

ABSTRACT

Title of dissertation: DEVELOPMENT OF A REALITY-BASED,
HAPTICS-ENABLED SIMULATOR FOR
TOOL-TISSUE INTERACTIONS

Kevin Lister, Doctor of Philosophy, 2012

Dissertation directed by: Professor Jaydev P. Desai
Department of Mechanical Engineering

The advent of complex surgical procedures has driven the need for finite element based surgical training simulators which provide realistic visual and haptic feedback throughout the surgical task. The foundation of a simulator stems from the use of accurate, reality-based models for the global tissue response as well as the tool-tissue interactions. To that end, *ex vivo* and *in vivo* tests were conducted for soft-tissue probing and *in vivo* tests were conducted for soft-tissue cutting for the purpose of model development.

In formulating a surgical training system, there is a desire to replicate the surgical task as accurately as possible for haptic and visual realism. However, for many biological tissues, there is a discrepancy between the mechanical characteristics of *ex vivo* and *in vivo* tissue. The efficacy of utilizing an *ex vivo* model for simulation of *in vivo* probing tasks on porcine liver was evaluated by comparing the simulated probing task to an identical *in vivo* probing experiment. The models were then further improved upon to better replicate the *in vivo* response.

During the study of cutting modeling, *in vivo* cutting experiments were performed on porcine liver to derive the force-displacement response of the tissue to a scalpel blade. Using this information, a fracture mechanics based approach was applied to develop a fully defined cohesive zone model governing the separation properties of the liver directly in front of the scalpel blade. Further, a method of scaling the cohesive zone parameters was presented to minimize the computational expense in an effort to apply the cohesive based cutting approach to real-time simulators.

The development of the models for the global tissue response and local tool-tissue interactions for probing and cutting of soft-tissue provided the framework for real-time simulation of basic surgical skills training. Initially, a pre-processing approach was used for the development of reality-based, haptics enabled simulators for probing and cutting of soft tissue. Then a real-time finite element based simulator was developed to simulate the probing task without the need to know the tool path prior to simulation.

DEVELOPMENT OF A REALITY-BASED, HAPTICS-ENABLED
SIMULATOR FOR TOOL-TISSUE INTERACTIONS

by

Kevin Lister

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2012

Advisory Committee:

Professor Jaydev P. Desai, Chair/Advisor

Professor Amitabh Varshney, Dean's Representative

Professor Nikhil Chopra

Professor Abhijit Dasgupta

Professor Teng Li

© Copyright by
Kevin Lister
2012

- Portions reprinted, with permission, from: "K. Lister, A. Lau, and J.P. Desai. Simulation of surgical cutting through cohesive zone methods. In: 2012 27th International Congress and Exhibition of Computer Assisted Radiology and Surgery, Pisa, Italy, 27-30 June 2012, 189-191." ©2012 CARS.
- Springer and ABME allow for the inclusion of material from: "K. Lister, Z. Gao, and J.P. Desai. Development of in-vivo constitutive models for liner: application to surgical simulation. In: *Annals of Biomedical Engineering*, 39(3): 1060-1073, 2011." ©2011 Springer.
- Portions reprinted, with permission, from: "K. Lister, A. Lau, and J.P. Desai. Towards a soft-tissue cutting simulator using the cohesive zone approach. In: 2011 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Boston, MA, 30 August - 3 September 2011, 66916694." ©2011 IEEE.
- Springer and ABME allow for the inclusion of material from: "Z. Gao, K. Lister, and J.P. Desai. Constitutive modeling of liver tissue: experiment and theory. In: *Annals of Biomedical Engineering*, 28(2): 505-516, 2010." ©2010 Springer.
- Portions reprinted, with permission, from: "K. Lister, Z. Gao, and J.P. Desai. A 3D in-vivo constitutive model for porcine liver: matching force characteristics and surface deformations. In: 2010 IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics, Tokyo, Japan, 26-29 September 2010, 656-661." ©2010 IEEE.
- Portions reprinted, with permission, from: "K. Lister, and J.P. Desai. Real-time, haptics-enabled simulator for probing ex-vivo liver tissue. In: 2009 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Minneapolis, MN, 2-6 September 2009, 11961199." ©2009 IEEE.
- Portions reprinted, with permission, from: "Z. Gao, K. Lister, and J.P. Desai. Constitutive modeling of liver tissue: experiment and theory. In: 2008 IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics, Scottsdale, AZ, 19-22 October 2008, 477482." ©2008 IEEE.

Acknowledgments

I would like to extend my gratitude to everyone who supported my graduate studies over the past few years. Without their assistance, this study would not have been possible.

First, I would like to thank my advisor, Dr. Jaydev P. Desai for giving me with an opportunity to work in the Robotics, Automation and Medical Systems laboratory for the past five years. He has provided the support, constant guidance, and the environment required to focus on the research at hand. Throughout the past five years, he has made the time to provide advice and assistance to keep the research project on course, even when there was little time to be had. It has been a pleasure working with and learning from Dr. Desai over the years.

I would also like to thank Dr. Zhan Gao for her assistance and patience in working so closely with me over the past few years. She took time during the beginning of this study to teach me the fundamentals of the modeling process required for me to succeed in the subsequent stages of this project. Working closely with her has taught me more than I could have ever imagined.

Dr. Alan Lau has also been instrumental in guiding the scalpel cutting modeling efforts. I greatly appreciate the time and effort he put forth to hold weekly meetings over the phone. His expertise in the fracture process provided the reassurance that I was in fact working in the right direction throughout the past year.

My colleagues at the Robotics, Automation and Medical Systems lab have enriched my studies over the past few years and greatly aided in my success. Dr.

Anand Pillarisetti was instrumental in the first years of my study, teaching me what it meant to be a graduate student and how to thrive in the graduate environment. Yang Bo and Ankur Baheti have provided great help in aiding in the programming requirements over the years. Mingyen Ho and Dr. U-Xuan Tan were always there to help with my numerous computer problems and questions about LaTeX. It has also been a pleasure discussing various aspects of this project with Elif Ayvali, Chad Kessens and Rajarshi Roy. Thank you also to Yasamin Ekrami and Bethany Springer for their assistance with the *in vivo* experimental device. I would like to express my appreciation for everyone in the RAMS lab who aided in making the lab a fun and exciting place work over the past four years.

My gratitude extends also to all the members of the staff here at the University of Maryland who have aided in this research project. To Dr. Hugh Bruck and Dr. Balakumar Balachandran for their support as the director of graduate studies. Amarildo Damata, Fitz Walker and Lee Ellen Harper for keeping me on track as the graduate studies coordinators. Janet Woolery, Penny Komsat and Juanita Irvin for pushing through my numerous purchase requests so quickly throughout the project. Steve Wicker, Lita Brown and Erin Chen for their payroll and medical insurance support. Daniel Wysling and Melvin Fields for their overwhelming support with all of my computer issues. Finally, Howard Grossenbacher for his assistance in machining many of the components which made the development of the testing equipment possible.

I owe my deepest thanks to my family who have stood by me, encouraged and supported me throughout my entire education process. My parents have always been

there to aid in any way they possibly could over the years, for which I am eternally grateful. My sister could always be counted on for words of encouragement when difficulties arose. Genevieve has been nothing but supportive through the entire endeavour; thank you very much for having the patience to deal with me on a daily basis. You all have provided an environment that allowed me to focus and prosper; without your help all to this would not have been possible.

Lastly, I would also like to acknowledge the financial support of the National Institutes of Health, the National Science Foundation and the Link Foundation for the funds required to perform the research.

Table of Contents

List of Figures	vii
1 Introduction	1
1.1 <i>Ex Vivo</i> Modeling	2
1.2 <i>In Vivo</i> Modeling	4
1.3 Soft Tissue Cutting	8
1.4 Real-Time Finite Element Simulations	10
2 Probing of Porcine Liver	14
2.1 <i>Ex Vivo</i> Modeling	14
2.1.1 <i>Ex Vivo</i> Experimental Device	15
2.1.2 <i>Ex Vivo</i> Experiments	19
2.1.3 Analysis of <i>Ex Vivo</i> Experimental Data	27
2.1.4 <i>Ex Vivo</i> Model Development	35
2.2 <i>In Vivo</i> Modeling	37
2.2.1 <i>In Vivo</i> Experimental Device	38
2.2.2 <i>In Vivo</i> Experiments	48
2.2.3 Analysis of <i>In Vivo</i> Experimental Data	53
2.2.3.1 First Experiment	54
2.2.3.2 Second Experiment	63
2.2.4 Finite Element Simulation of <i>In Vivo</i> Experiment	72
2.2.5 Comparison of Experimental and Simulated Data	81
2.2.6 <i>In vivo</i> Model Development	89
3 Cutting Of Porcine Liver	104
3.1 Modeling of Cutting Process	104
3.1.1 Cohesive Zone Approach	105
3.1.2 <i>In Vivo</i> Experiments	109
3.1.3 Generation of Cutting Model Parameters	113
3.2 Simulation of Cutting	136
3.2.1 Cohesive Zone Model Verification	136
3.2.2 Cohesive Zone Scaling Study	150
4 Real-Time Simulation	156
4.1 Preprocessor Based Probing Simulator	157
4.1.1 Fundamentals	157
4.1.1.1 FEA Parameter Verification	158
4.1.1.2 FEA of <i>Ex Vivo</i> Tissue Sample	164
4.1.2 Implementation of the Simulator	166
4.2 Preprocessor Based Scalpel Cutting Simulator	170
4.3 Real-Time Simulation using SOFA Framework	173
4.3.1 Theory of TLED Algorithm	175
4.3.2 Second PK Stress for Four-Term Reduced Polynomial	177

4.3.3	Comparison of SOFA Implementation to ABAQUS	181
4.3.4	Effect of Reduced Polynomial on SOFA Frame Rate	184
4.3.5	Probing Simulator with SOFA Framework	188
4.3.6	SOFA Framework Discussion and the Implications on Cutting Simulation	200
5	Conclusions and Future Works	203
5.1	Future Work	206
A	TLED Code for Reduced Polynomial Model	208
B	Scene File for SOFA Probing Simulator	214
	Bibliography	216

List of Figures

2.1	Experimental device for <i>ex vivo</i> experiments. ©2010 Springer.	16
2.2	Configuration for a) tension and compression and b) pure shear tests.	18
2.3	Initial sample geometry for a) tension, b) compression, c) pure shear (clamp), d) pure shear (plates).	20
2.4	Schematic of the uniaxial tension test. ©2010 Springer.	21
2.5	Sequence of images obtained during uniaxial tension test.	22
2.6	Schematic of the compression test. ©2010 Springer.	22
2.7	Sequence of images obtained during unconfined compression test.	23
2.8	Schematic of the shear test using a) parallel plates and b) tissue clamp. ©2010 Springer.	23
2.9	Sequence of images obtained during pure shear test with sample glued to the parallel plates.	24
2.10	Sequence of images obtained during pure shear test with tissue clamps.	25
2.11	Imperfections in tissue samples. Pure shear sample a) before loading and b) after loading showing the blood vessel not previously seen. Tension sample c) before loading and d) after loading revealing the defect. e-f) Improper adhesion of tissue to the test fixture.	26
2.12	a) Original image, b) Image after brightening, c) Conversion to greyscale, d) Increase of contrast, e) Canny edge detection results, and f) Edge detection superimposed upon original color image	30
2.13	Stretch ratio in the compression direction, λ_1 , versus time.	32
2.14	Nominal stress in the tissue sample throughout the unconfined compression test.	33
2.15	Nominal stress in the tissue sample throughout the unconfined compression vs the stretch ratio.	34
2.16	Experimental results for tension, compression and pure shear tests. ©2010 Springer.	35
2.17	<i>In vivo</i> experimental device configured for the probing task. ©2010 IEEE.	39
2.18	Support assembly for linear rail systems with adjustment capability possible during experimentation.	41
2.19	Marker used in MicronTracker camera system.	46
2.20	Positioning of the support plate with liver placement during first pig lab.	50
2.21	Raw point cloud data obtained during first pig lab.	51
2.22	Location of markers during one test location in the first pig lab.	52
2.23	Force and probe displacement profile for one probing test during the first experiment.	55
2.24	Steps for organizing the raw point data stored by MicronTracker into an organized structure for each marker.	56
2.25	a) Marker location with labels defined for markers of interest b) Measured marker displacement throughout probing task after filtering data.	58

2.26	a) Sample marker displacement through the entirety of the test b) Sample marker displacement vs probe displacement.	59
2.27	Typical marker tracking data for first experiment.	61
2.28	Solid geometry developed from measured point cloud data for first experiment.	63
2.29	Measured force values and corresponding polynomial fits for the three probing locations.	64
2.30	(Top) Raw force data obtained during probing task (Bottom) Measured voltage levels throughout the probing task. ©2011 Springer. . .	65
2.31	Presentation of the raw marker displacement data along with the polynomial fits for first probe location.	69
2.32	Presentation of the raw marker displacement data along with the polynomial fits for second probe location.	70
2.33	Presentation of the raw marker displacement data along with the polynomial fits for third probe location.	71
2.34	a) Image of the <i>in vivo</i> liver segment tested. b) Corresponding three dimensional model developed from point cloud data with three probing locations labeled.	72
2.35	Mesh for lobe of liver used for finite element simulation. ©2010 IEEE. 73	
2.36	a) Original shape derived from point cloud data. b) Plot of shape and displacement magnitude after inverse gravitational loading. c) Point cloud representation of new tissue surface. d) New solid model for shape without gravity derived from point cloud data. e) Shape after applying gravity back to the model. f) Plot of shape after gravity process compared to point cloud data of gravity free shape. g) Plot of shape after gravity process compared to originally measured point cloud data.	78
2.37	Organ geometry for second experiment after gravitational loading process containing the required internal stresses. ©2011 Springer. . .	79
2.38	Displacement plots after probing simulation for the a) first probe location b) second probe location and c) third probe location.	81
2.39	Comparison of the experimentally measured force profile and the simulated force response for the a) first probe location b) second probe location and c) third probe location. ©2011 Springer.	82
2.40	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the first probe location.	85
2.41	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the second probe location.	86
2.42	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the third probe location.	87
2.43	Method for modification of constitutive model. ©2011 Springer.	90
2.44	Displacement response from FEM simulation of probing task for a) original <i>ex vivo</i> model b) first iteration c) second iteration d) third iteration and e) final iteration of material modification procedure. . .	91

2.45	Comparison of the experimentally measured force and the simulated force values for the various iterations of the material model modification procedure. ©2011 Springer.	92
2.46	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the first iteration of the material model modification procedure.	93
2.47	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the second iteration of the material model modification procedure.	94
2.48	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the third iteration of the material model modification procedure.	95
2.49	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the final iteration of the material model modification procedure.	96
2.50	Displacement response from FEM simulation of probing task for original <i>ex vivo</i> model and final iteration of material modification procedure for a) second probe location and b) third probe location.	98
2.51	Comparison of the experimentally measured force and the simulated force values for the various iterations of the material model modification procedure for the a) second probe location b) third probe location.	99
2.52	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the final iteration of the material model modification procedure at the second probing location.	101
2.53	Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the final iteration of the material model modification procedure at the third probing location.	102
3.1	Cohesive traction-separation curves for i) bilinear ii) trapezoidal and iii) exponential models. ©2011 IEEE.	106
3.2	Cohesive zone region. ©2011 IEEE.	108
3.3	Experimental setup for the <i>in vivo</i> cutting test.	110
3.4	Marker placement for cutting test.	111
3.5	Magnitude of total cutting force throughout the experiment.	112
3.6	Filtering of experimental force data in the a) x and b) z directions. Polynomial fits to represent the eighteen individual cut segments for the c) x and d) z directions.	116
3.7	Critical force values for each of the eighteen cut segments in the a) x and b) z directions.	117
3.8	Measured scalpel profile with corresponding polynomial fit.	118
3.9	Sectioning of the scalpel blade for the first cut segment.	119
3.10	Free body diagram of force distribution on scalpel blade. ©2011 IEEE.	121

3.11	Critical force values in the a) normal and b) tangential directions of each cut section using a 0.2mm section length. c) Graphical representation of the normal and tangential force values.	122
3.12	Critical force values in the a) normal and b) tangential directions of each cut section using a 0.8mm section length. c) Graphical representation of the normal and tangential force values.	123
3.13	Critical force values in the a) normal and b) tangential directions of each cut section using a 2.0mm section length. c) Graphical representation of the normal and tangential force values.	124
3.14	Cross section of the scalpel blade.	125
3.15	Critical traction values in the a) normal and b) tangential directions of each cut section using a 0.2mm section length. c) Graphical representation of the normal and tangential traction values.	128
3.16	Critical traction values in the a) normal and b) tangential directions of each cut section using a 0.8mm section length. c) Graphical representation of the normal and tangential traction values.	129
3.17	Critical traction values in the a) normal and b) tangential directions of each cut section using a 2.0mm section length. c) Graphical representation of the normal and tangential traction values.	130
3.18	Cohesive strength parameter for the a) normal and b) tangential direction for each of the eighteen cut segments using 0.2mm sectioning. c) Table of exact values.	131
3.19	Cohesive strength parameter for the a) normal and b) tangential direction for each of the eighteen cut segments using 0.8mm sectioning. c) Table of exact values.	132
3.20	Cohesive strength parameter for the a) normal and b) tangential direction for each of the eighteen cut segments using 2.0mm sectioning. c) Table of exact values.	133
3.21	Energy values for each cut segment from a) cutting force b) vertical force c) total cohesive energy. d) Exact values for the various energy calculations.	135
3.22	a) Image of liver from <i>in vivo</i> cutting experiments, b) point cloud data fit with liver surface and c) solid model used in meshing process.	139
3.23	a) Full mesh for the cohesive zone validation study, b) Close up of cutting path, c) Cross section of dense elements in the cutting path. .	141
3.24	a) Mesh for the cohesive zone validation study with contact regions displayed in yellow, b) Close-up of mesh directly in cutting path. . .	142
3.25	Progression of the scalpel and resultant tissue displacement through the cohesive zone verification simulation.	147
3.26	Force profile acting on the scalpel blade during the <i>in vivo</i> experiment and corresponding simulation. ©2012 CARS.	148
3.27	Depiction of stress distribution within tissue during scalpel cutting process.	149
3.28	Depiction of stress distribution within tissue during scalpel cutting process for a) coarse, b) normal and c) dense meshes.	151

3.29	a) Displacement and b) force plots for cohesive zone scaling test . . .	153
3.30	Stress distribution throughout cutting process for a) coarse b) normal and c) dense meshes.	155
4.1	Displacement plot for the unconfined compression test for a) original sample b) vertical displacement c and d) radial displacements. . . .	159
4.2	Resultant force vs. stretch ratio profile derived from ABAQUS sim- ulation. ©2009 IEEE.	160
4.3	Resultant force vs. stretch ratio profile derived from theoretical anal- ysis. ©2009 IEEE.	163
4.4	a) <i>Ex vivo</i> geometry solid model b) <i>Ex vivo</i> mesh derived from solid model. ©2009 IEEE.	164
4.5	Tissue displacement (m) in the vertical direction following the simu- lation of the a) gravitational loading and b) probing action. ©2009 IEEE.	166
4.6	Structure of the real-time, haptics-enabled simulator.	167
4.7	Force data and resultant exponential fit derived from ABAQUS simu- lation.	169
4.8	Cross section of the tissue sample captured from the real-time simu- lator display.	170
4.9	Preprocessor based real-time, probing simulator with realistic display and haptic feedback. ©2009 IEEE.	171
4.10	a) Internal view and b) system setup for preprocessor based cutting simulator.	172
4.11	Results for SOFA simulation of a cube under gravitational load. . . .	182
4.12	Results for ABAQUS simulation of a cube under gravitational load. .	183
4.13	Comparison of the SOFA and ABAQUS simulations.	184
4.14	Comparison of the simulation speeds for the Neo-Hookean and re- duced polynomial models.	187
4.15	a) Screen shot of simulator with realistic geometry and b) perfor- mance characteristics with realistic geometry.	189
4.16	a) Screen shot of simulator with additional components and b) perfor- mance characteristics with additional components.	191
4.17	a) Contact model setup and b) contact between probe and tissue surface.	193
4.18	a) Simulation with collision response for liver only and b) performance characteristics with collision response for liver only.	196
4.19	a) Simulation with collision response with torso and b) performance characteristics with collision response with torso.	197
4.20	Reaction force acting on probe during SOFA simulation and <i>in vivo</i> experiments.	199

Chapter 1

Introduction

The development of complex surgical procedures coupled with more stringent regulations on medical education has driven the need to develop reality-based surgical simulators for medical training. Regulations on the maximum working hours permitted for medical residents limits the amount of time available to practice common surgical tasks such as needle insertion, probing, cutting, dissecting and electro-cauterizing tissue. Additionally, fatalities related to surgical procedures constitute the seventh highest mortality rate in the United States. In light of these facts, simulators need to be developed to provide thorough training for common surgical tasks.

Surgical simulators are now commonly classified into three categories: anatomy, physics and physiology-based simulators [19]. The first category consists of simulators based on the geometrical structure of the anatomy involved in the simulation tasks. Simulators of this kind have been mainly used for endoscopic procedures where little interaction with the environment is required [20]. Anatomy-based simulators do not incorporate any physical components, such as tissue deformations. Physics and physiology based simulators have recently become the focus of several research groups due to the push for more comprehensive surgical simulations. It is essential to replicate a surgical procedure as accurately as possible; therefore,

physics-based approaches must be utilized to maximize the realism of the surgical task, since visualization alone is insufficient.

The foundation of a physics-based simulator is centered on the accurate modeling of the mechanical properties of the organ utilized in the simulation, as well as the tool-tissue interactions. To this end, this study focused on the development of reality-based models capable of accurately representing the force and deformation resultant from deformation of the tissue sample derived through experimentation on both *ex vivo* and *in vivo* porcine liver. Further, the scalpel cutting process has been studied to derive a physics-based model to define the interactions between the scalpel blade and the soft-tissue. This provided the foundation required for the development of a surgical simulator capable of producing a realistic representation of the forces and deformation involved with surgical procedures such as probing and cutting of tissue.

1.1 *Ex Vivo* Modeling

The study began with the development of constitutive models for *ex vivo* porcine liver through the analysis of tension, compression and pure shear (also known as constrained tension or planar tension) tests. The mechanical properties of very soft-tissues, i.e., tissues that do not bear mechanical loads, have only been extensively investigated over the last decade. Most of the reported experiments and models for soft-tissues under finite deformation were focused on compression [53][68], indentation [7][18][36], uniaxial tension [13][14][54] and aspiration [33][59] tests. Ad-

ditionally, a limited number of shear tests were conducted on porcine kidney cortex tissue [23] and human liver tissue [72]. The majority of the studies presented, however, were conducted via analysis of only one deformation mode. For a general surgical simulation, various loading states are involved. Therefore to develop a realistic soft-tissue model, characterization of the soft-tissue in various deformation modes under low strain rate is required.

Miller and Chinzei were one of the first to study the properties of very soft in tension [54]. Through this study they proposed a viscoelastic model based on the generalization of the Ogden strain energy function for brain tissue with strains up to 30% in compression and 60% in tension. Additionally, Chui *et al.* proposed a model for porcine liver tissue with a strain range of 40% in compression to 60% in tension. During this test the displacement of the tissue sample was taken directly from the fixture displacement upon which the tissue sample was attached. Studies have shown that the stress-strain state on the tissue around the fixture-tissue interaction is complicated, which can result in increased errors [9][31]. Additionally, the effect of gravity makes the deformation field non-uniform, even on the central portion of the tissue samples for tension, compression and pure shear tests, due to the very soft nature of the tissue. Therefore, a study was derived to use digital image correlation (DIC), a non-contact imaging technique, to accurately measure the non-uniform strain of porcine liver tissue undergoing tension and pure shear tests. Additionally, an image based approach was also utilized to determine the strain from measurement of the sample deformation directly for the compression test. Using all of this data, comprehensive models based on tension, compression and pure shear data were

generated which could be used for simulation of general tissue deformation due to the inclusion of all the deformation modes [27][28].

Studies have shown that the mechanical properties of liver tissue are strain rate dependent [35][41][53][79]. However, at relatively low strain rate, e.g. 0.05 s^{-1} , the liver tissue has been reported to be relatively rate insensitive. Non-linear viscoelastic modeling of liver tissue from *in vivo* experiments on Rhesus monkeys showed that the contribution of time dependence for experiments with the slowest strain rate, 0.2 s^{-1} , is negligible [53]. Also, varying the strain rate from 0.003 to 0.6 s^{-1} did not have a significant effect on the stress-strain data for compression and elongation experiments on porcine liver [13]. In this study, all *ex vivo* compression, tension and pure shear tests were conducted at a nominal strain rate of 0.05 s^{-1} , hence viscoelastic properties were not taken into consideration. The models developed in this paper are suitable for use as the quasi-static hyperelastic model, which is an essential component of a viscoelastic material model.

1.2 *In Vivo* Modeling

The models developed during the *ex vivo* testing provide an excellent starting point for development of a physics-based surgical simulator due to their generality. The end goal of the surgical training system is to mimic the haptic and visual depiction of a surgical task as precisely as possible. Therefore, the simulator should be built upon models that accurately reflect the response of tissue in the *in vivo* state. Previous studies have shown that the material properties of soft-tissue tend to

vary between the *in vivo* and *ex vivo* settings [6][29][41]. To further the accuracy of the simulator, an *in vivo* experiment was performed to study the variation between the properties of the liver tissue in the *ex vivo* and *in vivo* settings.

Developing models based on *in vivo* soft-tissue response requires the utilization of more accurate boundary conditions and approaches for non-contact methods of measuring the tissue deformation. *In vivo* experimental testing, in contrast to *ex vivo* studies, requires the use of the entire organ instead of a subsection of the appropriate size and shape, as is used for compression, tension and pure shear tests. In light of this fact, the techniques for *in vivo* tests are more varied than that of standard *ex vivo* studies. Miller *et al.* studied the applicability of using a hyperviscoelastic model derived from *ex vivo* experimentation for *in vivo* simulation of brain tissue [52]. Realistic organ geometry was obtained by utilizing MRI images to acquire an accurate representation of the brain. This study only utilizes the MRI images to obtain the organ geometry, where tracking of displacement under load is not achieved. Nava *et al.* developed a model based on an aspiration test of *in vivo* human liver through a 2D axisymmetric process based on the local tissue profile [59]. 3D ultrasound techniques were applied to an area directly under the probe to develop a model based on volumetric tissue deformation and measured force response of perfused *ex vivo* liver [39]. Ahn and Kim developed a material model utilizing force and surface displacement data for *ex vivo* liver whereby full-field deformation could be determined as well as accurate sample geometry [1]. The model was built upon data obtained from the probing force using an axisymmetric inverse finite element approach with simplified organ geometry. All of these techniques have

been utilized to develop more accurate models for the biological tissue deformation process. However, they do not present a method by which the force and displacement can be measured and compared to a simulation with accurate geometry and realistic boundary conditions throughout the entirety of the probing task.

The first question that arises when working with *in vivo* and *ex vivo* tissue models concerns the degree of variation between the two models. As the *ex vivo* experimental studies are much more straightforward, it would be beneficial to be able to use a model generated from *ex vivo* experimentation for *in vivo* surgical simulation. To this end, a method was developed for the acquisition of the surface deformation in conjunction with the force-displacement profile and accurate boundary conditions through the entirety of an *in-vivo* probing task on porcine liver. This data was used to simulate the experiment utilizing an Ogden model developed from *ex-vivo* compression, tension, and pure shear experiments. By comparing the simulation results to the experimental data it was possible to determine the efficacy of utilizing a model developed from *ex-vivo* techniques for *in-vivo* simulation of porcine liver.

The next step was to develop a material model for porcine liver which more accurately matched the *in vivo* tissue response. Currently reported *in vivo* and *ex vivo* soft-tissue whole-organ tests are mainly split into indentation [1][7][10][36][39][41][52][69][71] or aspiration [40][50][59] experiments. In these tests, the material parameters were usually extracted from finite element simulations coupled with an optimization algorithm to fit either force-displacement data of the indenter or the surface profile of the aspirated tissue. One exception to this trend is the study by

Jordan *et al.* where both volumetric deformation data and the force-displacement response of the indenter were used in fitting the material parameters [39]. Nevertheless, all of the above studies predefined a fixed format for the material model and determined the best fit to the material properties. The method presented in this study is different from the aforementioned approaches. Instead of modifying material parameters in each iteration, material properties (underlying stress-strain curves) were modified directly according to the comparison of the experiment and simulation results. As a consequence, the fitting process can be more efficient and the material model does not have to be a fixed format [45][46].

The modeling discussed thus far has focused on accurately representing general finite material deformations, which is the first requirement for a surgical simulator. If more complex tasks are required in the simulation, this level of modeling may not be sufficient. For simulation of simple surgical tasks such as probing, the only models required, as is shown in this study, are the general models developed from the *ex vivo* or *in vivo* experiments. However, most medical procedures involve more complex tasks such as cutting or dissecting the soft-tissue which can not be completely defined through the previously developed material models. Therefore, additional models must be developed to govern the specific tool-tissue interactions as required for the medical trainer. In this light, the details of the scalpel-soft-tissue interaction were studied in addition to the general tissue deformations.

1.3 Soft Tissue Cutting

In studying the simulation of soft-tissue cutting, the current research can be segmented in two distinct areas. From the simulation side, there is a focus on the element segmentation aspects while the material research has focused on the physics of the cutting process. In the simulation research, three main methods are utilized to replicate the cutting or fracture process of soft-tissue. The first approaches began by simply removing the elements that came into contact with the cutting implement [16]. This approach, however, is not physically sound as material is effectively being destroyed. Next, there were studies into the subdivision of elements [51][58] and more recently adaptive meshing was utilized to preserve the structure and volume of the mesh while maintaining elements with proper aspect ratios [61][74].

For the course of this study, we are first interested in the physics-based modeling of the scalpel cutting procedure. The available literature on determining physically accurate models of soft-tissue cutting comprises a much smaller group. The studies on fracture of soft-tissue are few and far between; also showing much variation in the methods used for the modelling process. Most of the models presented are based on deriving the fracture toughness of the biological tissue tested, or the energy required to create a fracture. The derivation of the fracture toughness has been conducted using scissor cutting of human skin [64] and sheep skin, abdominal muscle, liver and Achilles tendon [30]. Loading in a guillotine manner using a razor blade was conducted on porcine liver by Mahvash *et al.* [49] and needle insertion

into liver tissue was conducted by Azar *et al.* [3]. A slightly different approach was used to determine the local effective modulus for cutting of liver tissue with a scalpel blade [11]. These approaches do utilize fracture mechanics based methods, however, only the threshold of the fracture process can be determined from the measured values.

Cohesive zone modeling is a method based on fracture mechanics principles that simplifies the fracture process into a straightforward analysis of the region in front of the crack tip. Only three parameters define the model which governs the onset of fracture and the subsequent propagation [75]. Cohesive base models have been mainly utilized in studying the debonding process, where it represents the strength of the adhesive between two materials [4][57][81][82]. The heavy focus on the debonding problem stems from the inability of the cohesive zone model to relate information pertaining to the direction of the fracture. In fact, the model itself is designed under the assumption that all loading is confined to a line in front of the crack [21]. While this makes the cohesive zone approach difficult for many fracture-based studies, where the location and path of the fracture process is unknown, it does not limit the application to the scalpel-tissue interaction. Based on observation of experimental cutting studies, the fracture path during a scalpel cutting procedure is a line formed by the scalpel trajectory. Because the cohesive approach simplifies the physical mechanisms of the fracture process into a succinct model, it is well suited as a starting point for the modeling of the scalpel-tissue interactions where the parameters can be experientially determined through analysis of cutting experiments. After the cutting model is developed, the combination of

this model and the general deformation model derived from the *in vivo* experiments provide the framework upon which a surgical simulator can be developed.

1.4 Real-Time Finite Element Simulations

Finite element models have become the focus of many surgical simulation applications due to their increased accuracy and continuous representation of volumetric tissue deformation. The improvement in realistic deformation however, comes at the cost of computational complexity. New techniques are required to solve the finite element problem at speeds fast enough to produce real-time visual (30Hz) and haptic (1000Hz) displays. Linear elastic finite element methods have been implemented in various surgical simulators, including hepatectomy procedures [5][16]. However, simulations built upon linear elastic models can only be applied to small deformations and are not applicable to general surgery procedures.

Real-time finite element methods involving non-linear material characteristics have begun to gain attention in the research community. Picinbono *et al.* developed a simulator for laparoscopic liver surgery based on a St. Venant-Kirchoff model [66]. A Mooney-Rivlin model was utilized in the simulation of eye surgery and liver probing tasks [70][83]. An endoscopic simulator has been developed utilizing a Neo-Hookean material law for the surrounding tissue [77]. Joldes *et al.* utilized a Neo-Hookean material for the real-time simulation of brain tissue [38]. These studies all utilized simplified non-linear material models in order to achieve the real-time update rates required for simulation of the defined tasks. Although these models

are superior to the linear elastic approach, they still do not accurately reflect the true complexity of the soft-tissue deformation process.

The first generation simulator developed in the study presents a method to create real-time, haptics-enabled soft-tissue probing and cutting simulators. These simulators are governed by an Ogden model, derived directly from the *ex vivo* experiments, and a reduced polynomial model, developed during *in vivo* experiments. They produce a more accurate characterization of the soft-tissue deformation process [44]. This simulator utilizes a preprocessed finite element analysis derived from an accurate constitutive model of liver tissue and realistic geometry. The focus of the first generation simulator was to show an ability to utilize the more complicated non-linear models in conjunction with realistic tissue geometry to obtain a more accurate representation of soft-tissue response to probing and cutting stimuli. Preprocessing the tissue response is a means to incorporate the more detailed hyperelastic models and complex geometry in the simulation, resulting in improved accuracy of the tissue response to a probing action.

The approach of using preprocessed information as the backbone of a surgical simulator is limited in its applicability to many common surgical tasks due to the need for knowledge of the defined tool paths prior to the simulator development. This limitation prevents the method from being used for general, open surgical tasks. Therefore, many modern simulation systems have been focusing on utilizing the superior computational capabilities of modern computers to perform the finite element simulation processes itself in the real-time setting.

An encompassing review of the history and trends of the medical simulation

field, including the transition to real-time finite element approaches utilizing non-linear models can be found in [56]. More modern works, as the field is evolving quickly, are focused on the implementation and performance of real-time finite element approaches utilizing both non-linear material models and non-linear geometry. Miller *et al.* developed a total Lagrangian explicit dynamics (TLED) approach that has recently been adopted and applied in many real time simulators [55]. This approach allows for the calculation of all derivatives with respect to the original configurations, which provides the capability to apply precomputation techniques. Additionally, the method is structured to eliminate the need to assemble a global stiffness matrix. Another approach presented in [62] uses a proper orthogonal decomposition technique to reduce the order of the required stiffness matrix to decrease computation times. This approach was successful in replicating real-time palpation of human cornea models without the use of parallel architectures. A precomputation approach was also utilized by [65] to simulate St. Venant-Kirchhoff and Mooney Rivlin models in liver deformations.

Advances are not only being made in the algorithms for solving the non-linear finite element problems, but methods are simultaneously being made more efficient through the use of parallel processing on graphics processing units (GPU). The total Lagrangian explicit dynamics approach has been utilized to simulate the neurosurgical procedures developed by Joldes *et al.* using a Neo Hookean material law. Implementation of the process on a GPU resulted in a 20x overall increase of speed. A similar study for the TLED based approach was conducted for compression and pure shear simulations of a Neo Hookean material with anisotropic and viscoelastic

effects added. This project showed a increase in speed of 56.3 times on average. GPUs have also been adopted for other types of real-time finite element simulations including an iterative based method [17], which showed a varying degree of increased performance throughout the individual components of the code.

With these advances in the methodology for approaching the real-time finite element problem, combined with the efficiency provided by implementation on graphical processing units, the medical simulation field is taking large steps to close the gap between the trade-off of speed and accuracy in the simulation field. The TLED approach in combination with an open source finite element code developed for surgical simulation (Simulation Open Framework Architecture - SOFA) was adopted to further improve upon the first generation, preprocessor based simulators. This provided the means to study the possibility of developing real-time simulations using the more accurate hyperelastic models in which prior knowledge of the tool paths was not required.

Chapter 2

Probing of Porcine Liver

2.1 *Ex Vivo* Modeling

To develop a surgical simulator with superior accuracy requires a more precise foundation in the characterization of the underlying material properties of soft-tissue. In most commercially available real-time simulators, simplifications have been made which normally come down to the material model. Almost all simulation systems for medial training use linear elastic material models due to the fact that they simplify the computational complexity to a level that is manageable for real time implementation. However linear elastic models are not sufficient to accurately represent the loading and deformation of biological tissues. Therefore, to increase the realism of a medical training system, the most logical course of action would be to develop more accurate, reality-based constitutive models. In this context, we were interested in developing realistic models irrespective of the implications on real time simulation. As computational efficiency is continually increasing, there was a good chance that a model too complicated to implement in real time at the start of this project would be applicable after the modeling process was completed.

As a starting point, common tests used for material property characterization were applied to *ex vivo* samples of porcine liver. The goal was to develop a model based on tension, compression and pure shear (also known as constrained tension

or planar tension) tests which could be used to simulate general loading of soft-tissue, as is required for general surgical procedures. It is important to note that the majority of studies involving soft-tissue modeling aim to model one deformation mode at a time, for example, compression [53][68] or tension [13][14][54] of soft-tissue. In this chapter, details on the development of the experimental device for *ex vivo* studies, sample preparation, and experiments involving tension, compression and pure shear are described in detail. Soft-tissue modeling based on the *ex vivo* experimental data was done by Dr. Zhan Gao using some of the existing models in the literature [13][18][27][28][63][68].

2.1.1 *Ex Vivo* Experimental Device

A test apparatus, Figure 2.1 was designed and constructed to perform tension, compression and pure shear tests on samples of porcine liver. The device contains only one translational degree of freedom (DOF) in the direction of the desired applied load for the three tests. A rigid frame was constructed from aluminium to house the sensors and actuators required for the three experiments. A Maxon A-max 32 motor with a Maxon GP32 planetary gearhead connected to a lead screw assembly provided the controlled linear motion. The lead screw assembly utilized an antibacklash nut to prevent backdriving the motor under loading conditions. The motor was connected to the lead screw assembly by way of a 300mm threaded rod. Owing to inherent inaccuracies in machining practices, the alignment between the motor shaft and the threaded rod requires the use of a mechanical coupling between the two shafts to

account for the slight misalignment.

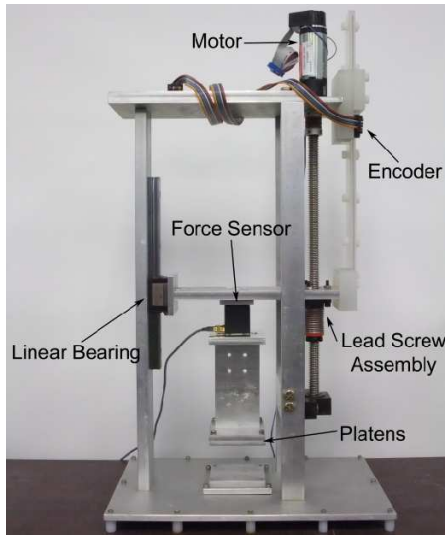


Figure 2.1: Experimental device for *ex vivo* experiments. ©2010 Springer.

The lead screw assembly had three major components attached: a linear bearing, a linear encoder system and the force sensor. The linear bearing (NSK Ltd.) was used in conjunction with the lead screw to translate the rotary motion of the motor into linear motion by restricting the rotation of the lead screw assembly. The bearing is constructed from hardened steel ball bearings with a corresponding linear rail to reduce the friction within the system.

As seen in Figure 2.1, the motor utilized in the device contained an attached encoder; however, this proved insufficient for the position measurement and an additional linear encoder (US Digital EMI-0-500, 500LPI) was utilized. The problem in using the rotary encoder attached to the motor arose from the use of the coupling between the motor unit and the threaded rod. In determining a suitable coupling for the task, it was discovered that all available options allow for a certain amount of axial compression. When the deformation mode of compression is analyzed, the

displacement of the lead screw assembly measured by the rotary encoder would be incorrect as it would not sense the magnitude of the coupling compression. To remedy this situation, a linear encoder was placed on the rigid frame while the optical strip was attached to the lead screw assembly. Therefore, any motion of the assembly relative to the rigid frame would be recorded, and compression or tension of the coupling would not effect the readings.

The final connection to the lead screw assembly was the force sensor. In this test, two different force sensors where used. For the compression test a 6-axis force/torque sensor (JR3 Model No. 20E12A-I25) with a range of 0.05 - 50N was used. For the tension and pure shear tests, where the loads were not as high, a Transducer Techniques MDB-2.5 load cell, with a range of 0.01-11N, was used. Both of the force sensors were attached to the lead screw assembly in the center between the linear guide rail and the threaded rod. This allowed for a even distribution of the load to the two support elements during testing to prevent any wedging or build up of friction within the system. The entire device was controlled via a dSPACE DS1103 controller board (dSPACE, Inc.) which also recorded the force and displacement data throughout the experimental runs.

For testing each tissue sample, slightly different configurations were utilized depending on the deformation mode. For compression and tension tests, two parallel platens, with the tissue sample placed between, were utilized (Figure 2.2a). One platen was directly connected to the rigid frame while the other was attached to the force sensor. A slightly different setup was used for the pure shear tests, whereby a tissue clamping system was added between the rigid frame and the moving force

sensor (Figure 2.2b). This clamp allowed for the support of the larger sample geometry required for the pure shear test. The fixture utilized serrated teeth to clamp onto the edges of the sample. The edges of the teeth were slightly rounded to prevent shearing of the sample while providing the required grip on the tissue. It was later determined that the clamping action on the tissue forced the sample around the clamp to deform, resulting in a preloading effect prior to testing. Therefore, additional shear tests were conducted using the parallel platen geometry as well.

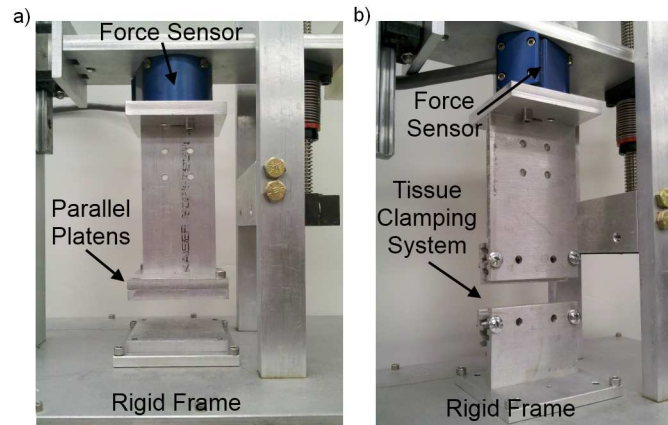


Figure 2.2: Configuration for a) tension and compression and b) pure shear tests.

To obtain the information on the tissue displacement during the tension, compression and pure shear tests, the experiments were recorded via still or video images. The tension and pure shear tests utilized a Nikon D40 camera, a high resolution camera which produced images sufficient for analyses by with digital image correlation to determined the strain in the tissue sample. However, the Nikon only could acquire images at 3 frames per second. For the compression test, this frame rate is insufficient. Owing to the fact that the compression distance is much less than the tension or pure shear tests, only a few frames would be captured with the low frame

rate camera for the compression test. Therefore, a Sony PCR-PC120E digital video camera was used to increase the number of frames captured (29.7 frames per second) in the compression test.

2.1.2 *Ex Vivo* Experiments

Sample Preparation

Twelve fresh pig livers were collected from a slaughter house and kept frozen in the laboratory. All tissue specimens were cut from frozen porcine liver and remained frozen until the time of the test. Careful attention was given to the orientation of the tissue samples. The samples were extracted perpendicular to the top surface of the livers and tested in that direction. The liver's outer layer of membrane (capsule) was carefully removed. Samples with large blood vessels or obvious pores were discarded in the experiments. Since the structural unit of liver, the hepatic lobule, is roughly 1 ~ 2mm in size [32], the sample side length was chosen in the range of 7 ~ 63mm to ensure that the tissue property was relatively uniform throughout the specimen volume and the average response of the tissue as a homogeneous continuum was measured.

The samples for the tension test were in the shape of rectangular prism (Figure 2.3a) and cut from a partially thawed liver by using a long blade knife. As the tissue thawed and deformed locally upon contacting the knife, it was difficult to obtain a precise 90 degree relationship between the two adjacent faces of the prism. Therefore, the dimensions of opposite faces of the samples could vary by 1 – 2mm.

The samples were then individually wrapped in plastic wrap and kept in freezer overnight until the tests were to be performed. The height of the sample for tension test was approximately 25mm, while the width and thickness was approximately 15 – 17mm.

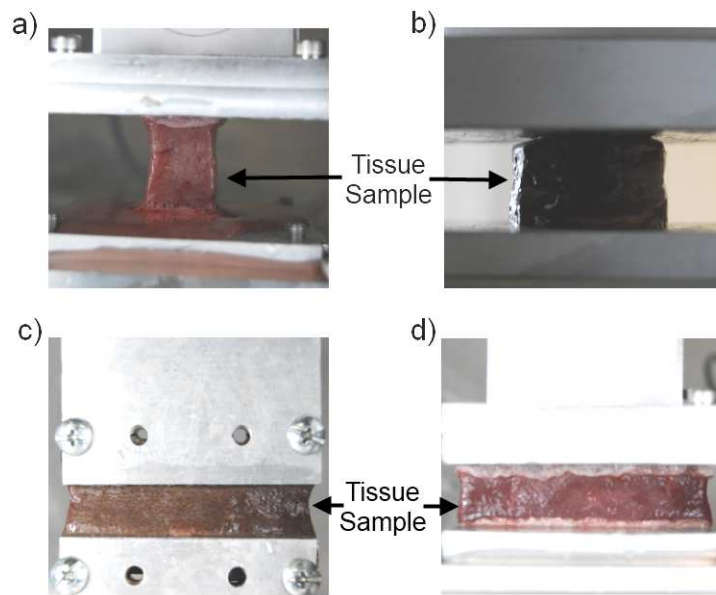


Figure 2.3: Initial sample geometry for a) tension, b) compression, c) pure shear (clamp), d) pure shear (plates).

For the compression test, cylindrical samples (Figure 2.3b) were cut by using an aluminum pipe (25.4mm diameter) with sharp edge. The faces of the cylindrical liver specimens were trimmed manually. The diameter of a sample was ~ 25.4 mm. The height was in the range of 9 – 17mm. The width of the sample for pure shear test (Figure 2.3c and d) was approximately 50 – 63mm with a thickness of 7 – 11mm, and a height of 13 – 15mm. The dimensions of the frozen samples were measured before the test. Each dimension of a sample was measured at three different locations using a Vernier caliper and the average value was recorded.

Uniaxial Tension Test

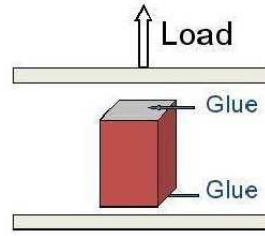


Figure 2.4: Schematic of the uniaxial tension test. ©2010 Springer.

As shown in Figure 2.4, a frozen sample was attached to the bottom platen by using cyanoacrylate (Super Glue). The glue was then applied to the top of the sample. The upper plate of the testing apparatus was moved down slowly so that it touched the upper surface of the sample. To assure proper attachment, the tissue sample was compressed by approximately 1 – 2mm . The specimen was left for about 15 – 30 minutes to thaw. Water mist was sprayed 2 or 3 times during that time to prevent dehydration of the tissue sample. Upon completion of the thawing process, the tension test was performed by moving the top platen at 1.25mm/s, corresponding to the nominal strain rate of approximately 0.05s^{-1} . The tests continued until the glue failed. Images of the deforming liver tissue samples in tension were acquired for later analysis. A subset of the images are displayed in Figure 2.5. The sprayed water mist may have some effect on the properties of the tissue, but due to the short period of testing time ($t < 30$ min), the potential influence was neglected.

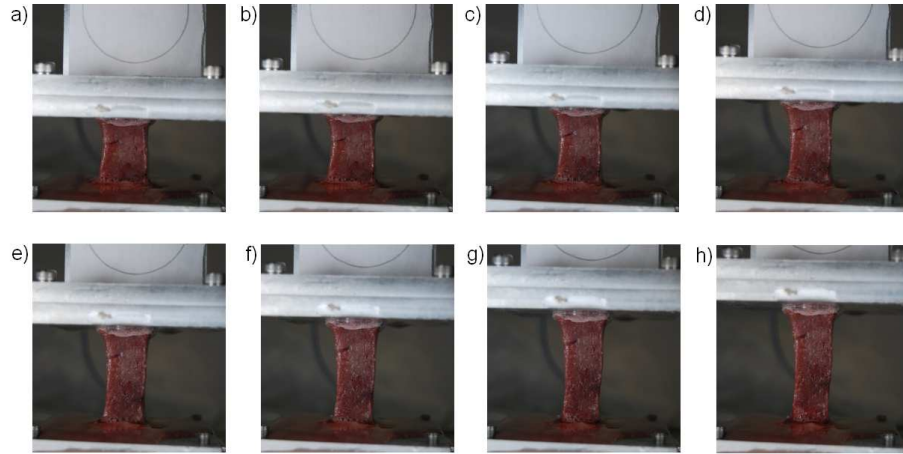


Figure 2.5: Sequence of images obtained during uniaxial tension test.

Unconfined Compression Test

The cylindrical tissue samples for the compression test were allowed to thaw in Saran Wrap and overturned several times prior to placement on the platens. This helped in maintaining the uniform shape of the sample prior to testing. The platens was covered with a thin layer of petroleum jelly to minimize friction between the plates and the tissue (Figure 2.6). The height of the sample before and after thawing was measured. Both the digital video camera and the still camera were used to obtain the image data from two perpendicular directions during the loading phase. The images were used to identify the samples that had uniformly expanded in radial

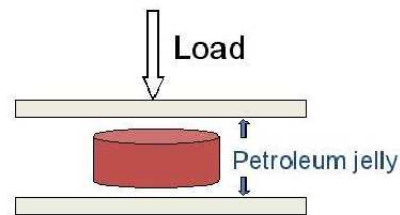


Figure 2.6: Schematic of the compression test. ©2010 Springer.

directions. A subset of the images obtained during one unconfined compression test are displayed in Figure 2.7.

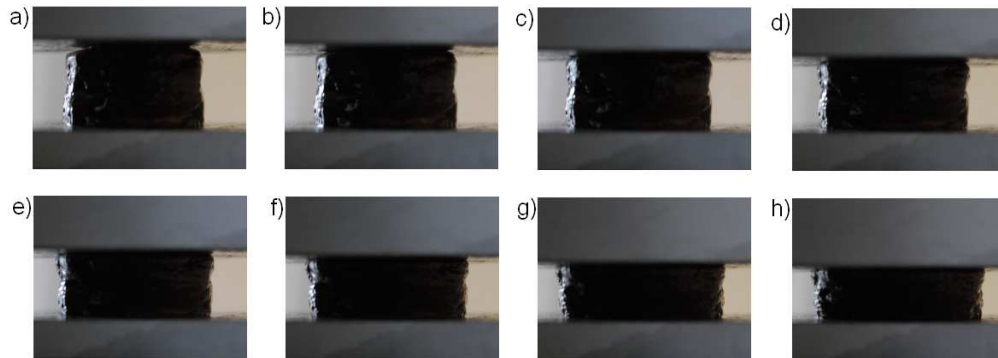


Figure 2.7: Sequence of images obtained during unconfined compression test.

Pure Shear Test

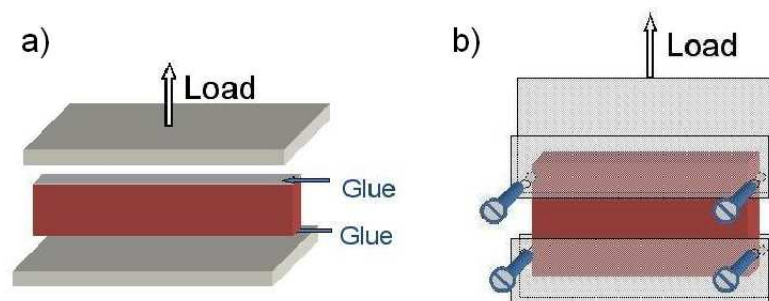


Figure 2.8: Schematic of the shear test using a) parallel plates and b) tissue clamp.

©2010 Springer.

The most significant aspect of the pure shear test is that the specimen is much shorter in the direction of stretching than in the width direction. This test has also been called a constrained tension test or a planar tension test. The objective is to create an experiment where the specimen is constrained in the lateral direction

such that all specimen thinning occurs in the thickness direction. The pure shear tests were conducted in two different ways, as illustrated in Figure 2.8. In the first setup, the frozen sample was attached to the platens using glue (cyanoacrylate). To assure proper attachment, the sample was compressed by approximately 2 – 3mm. In the second setup, the sample was held by two sets of custom-made clamps. The serrations inside the clamps provided a firm grasp to hold the tissue in place. The frozen specimen was placed between the clamps by slightly tightening the screws. Water mist was sprayed onto the specimen surface to prevent dehydration. During the process of thawing, the two sets of screws in the clamps were tightened gradually to ensure uniform clamping prestrain and also enough gripping force to hold the tissue. The specimen was left for about 15 – 30 minutes to thaw at room temperature ($\sim 22^\circ\text{C}$). The tests were performed at the nominal strain rate of 0.05s^{-1} , or roughly $0.65 - 0.75 \text{ mm/s}$, depending on the height of the samples. A depiction of the pure shear test using the parallel plates is shown in Figure 2.9 while the test conducted using the tissue clamps is displayed in Figure 2.10.

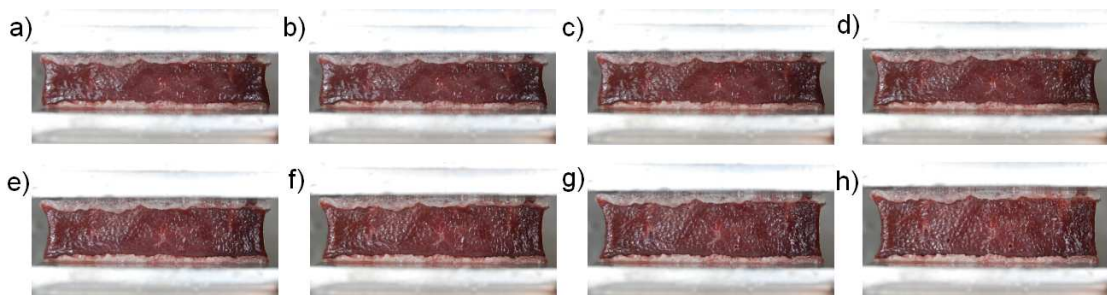


Figure 2.9: Sequence of images obtained during pure shear test with sample glued to the parallel plates.

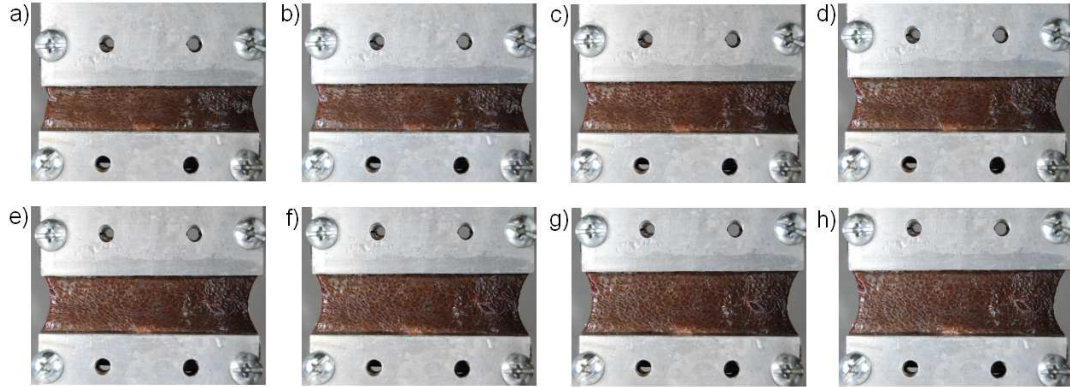


Figure 2.10: Sequence of images obtained during pure shear test with tissue clamps.

Testing Difficulties

Throughout the testing process many difficulties arose which greatly increased the number of tests required for each deformation mode. For the pure shear and the tension test, the tissue samples were being pulled apart. As previously mentioned, the samples for both of these tests were placed into the test apparatus while they were still frozen to assure proper adhesion. In many cases, it was difficult to determine if the sample was free of any defects, flaws or blood vessels while in the frozen state or after the tissue had naturally thawed. Therefore, often during an experimental run, an imperfection in the tissue sample would arise requiring the test to be conducted again. Figure 2.11a and c show tissue for the pure shear and tension tests after the thawing has completed that appear to be structurally valid samples; however after the test has commenced it is easily seen that large defect within the samples (Figure 2.11b and d) warrant them useless.

The other major problem that arose in this mode of testing came from the fixation method of the tissue samples. After many trials it was determined that

the best method of adhering the tissue to the test fixture was through the use of cyanoacrylate, while the samples were still frozen. Despite the fact that this was the best method of fixation, the bonding of the sample to the plates often failed under little load. As with the previous problem, there was no way to determine the validity of the bond prior to running the test. Figure 2.11e and 2.11f depict some of the bonding difficulties that occurred.

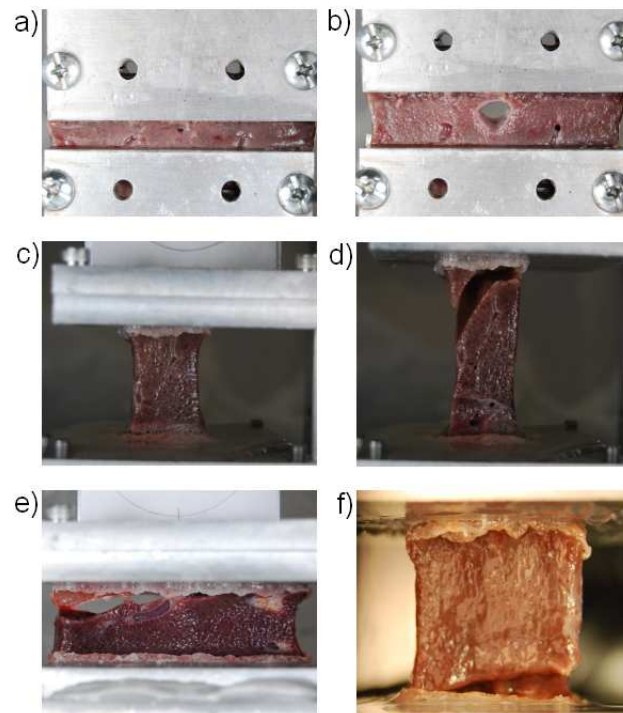


Figure 2.11: Imperfections in tissue samples. Pure shear sample a) before loading and b) after loading showing the blood vessel not previously seen. Tension sample c) before loading and d) after loading revealing the defect. e-f) Improper adhesion of tissue to the test fixture.

The compression test was more straightforward with respect to minimizing errors. The only difficulty encountered during the unconfined compression test was

obtaining a sample that could be analyzed properly. This consisted mainly of producing a run where the sample expanded uniformly in the radial direction without translating between the two plates. If the top and bottom surfaces were not cut parallel, or if internal imperfection existed in the tissue, the sample would translate to one side during the compression test. Therefore, like the other two deformation modes, many experimental runs were required to obtain a valid data set for analysis.

2.1.3 Analysis of *Ex Vivo* Experimental Data

The goal of the *ex vivo* experimentation was to determine a constitutive model that could be used to predict deformations and forces induced by general loading of the tissue samples. To derive the desired material models, an accurate stress-strain relationship was required upon which the model would be constructed. By performing the tension, compression and pure shear tests, individual components of the stress-strain relationship could be determined and combined to produce a complete curve which could then be used to generate an appropriate material constitutive model. In all of the experiments, the force induced on the load cells was collected in addition to the still or video images. By combining the force and image data it was possible to generate the stress-strain curve for each experiment.

The tension and pure shear tests were analyzed by Dr. Zhan Gao through the use of a global digital image correlation (DIC) technique [27][28][25]. Dr. Gao employed the non-contact DIC approach to determine the strain field within the tissue sample, whereby the natural patterns on the liver tissue were used to track

the tissue deformation. The compression test, on the other hand, could not use DIC due to the fact that the front surface area decreased during testing. Therefore, as the test progressed, the natural textures quickly became less distinguishable. Also, during the compression, the texture on the surface was often obscured by the excretion of blood from the tissue sample. To obtain the required strain information a Canny edge detection algorithm was developed to detect the boundary of the tissue and analyze the sample deformation.

The unit of measure for strain utilized throughout the analysis is the stretch ratio. In the case of the compression, test the stretch ratio in the direction of the compression is the required information. The stretch ratio is given by:

$$\lambda = \frac{l}{L}$$

where l is the current length of the sample and L is the original length. For the unconfined compression test there is a relationship between the three principle stretch ratios given by:

$$\lambda_2 = \lambda_3 = \lambda_1^{-\frac{1}{2}}$$

In this case λ_1 is the stretch ratio in the direction of the compressive motion. λ_2 and λ_3 are the principle stretch ratios in the radial direction. The desire is to find an accurate representation of λ_1 for the tissue sample throughout the experimental run. The simplest method to determine this quantity is to directly measure the motion of the platens and assume the tissue conforms to the distance between the two throughout the test. This approach, however, assumes uniform contact of the sample throughout the entirety of the test. From preliminary testing it was discovered that

difficulties arose in determining the starting point for the tissue-plate contact. An incorrect starting point would cause a shift in the resultant stress-strain curve which would decrease the model accuracy. Additionally, using only the plate displacement would not provide any information on the validity of the test because there would be no test for uniform radial expansion.

In light of these facts, a method was derived to calculate λ_1 from λ_2 which could be directly measured through the captured image sequence for each compression test. The relationship is given by:

$$\lambda_1 = \frac{1}{\lambda_2} \quad (2.1)$$

To acquire the information for the stretch ratio in the radial direction, λ_2 , a code was developed utilizing a Canny edge detection analysis to directly calculate the stretch ratio. Using the equation above, the stretch ratio in the principle compression direction could be determined.

The Canny edge detection algorithm is inherently sensitive to changes in the image intensity; therefore while performing the compression test all surfaces facing the camera were covered with white paper to maximize the contrast between the tissue sample and the background. The original image can be seen in Figure 2.12a. To assure robust detection of the outline of the tissue sample, a few steps were taken to modify the contrast of the image. First, the original color image was brightened; then the image was converted from color to greyscale as required for the Canny edge detection algorithm written in Matlab. The contrast of the image was then increased to make the detection of the outline of the tissue sample more robust

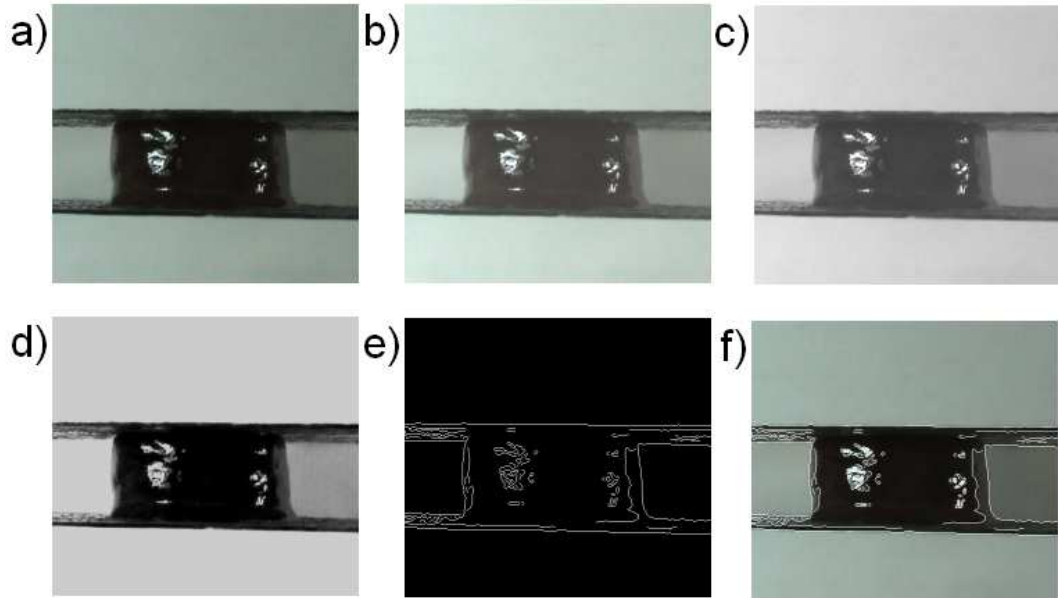


Figure 2.12: a) Original image, b) Image after brightening, c) Conversion to greyscale, d) Increase of contrast, e) Canny edge detection results, and f) Edge detection superimposed upon original color image

when applying the edge detection algorithm. Finally, the Canny edge detection algorithm was applied to the sample with the appropriate threshold level for the given experiment, thereby finding the outline of the tissue sample. The entirety of the process can be seen in Figure 2.12, where the final image shows an overly of the edges discovered in the Canny process on top of the original, unmodified color image. As can be seen, there is an excellent correlation between the two.

Upon completion of the edge detection process it was possible to determine the sample width at the midpoint between the two parallel plates. An algorithm was developed in Matlab that found the stretch ratio, λ_1 , by the following steps:

- Using the data from the edge detection, determine distance between two par-

allel plates in units of pixels

- Find the center plane as the midpoint between the two plates
- Find the intersection of the center plane with the tissue profile determined in the edge detection analysis in terms of pixels
- Convert the width of the sample to mm
- Calculate $\lambda_2 = \frac{\text{Width of Sample in Current Frame}}{\text{Width of Sample in First Frame}}$
- Determine λ_1 from Equation (2.1)

The conversion from pixels to millimeters was determined from the first image in the sequence along with some known camera information and calibration procedures. Prior to testing, a circle of known dimension was recorded to be used for calibration. Using this image and the camera resolution it was possible to get a correlation between the relative length of a pixel in the horizontal and vertical direction. This, together with a reference length in each particular experiment, would yield the conversion between pixels and millimeters. In the case of the compression test, the distance between the platens prior to the experiment was measured with a pair of callipers. The calibration information along with the first distance reading between the platens set the conversion factor for the experiment.

Using the above technique and the known frame rate of the camera system, it was possible to determine a plot of the stretch ratio in the compression direction versus time (Figure 2.13). As can be seen in the figure, there was a relatively linear relationship which was expected as the plates moved at a constant speed throughout

the compression test. The data is not smooth due to the limitations in the resolution of the digital video camera. When calculating the width of the sample, the distance is limited to steps of one pixel due to the image processing techniques.

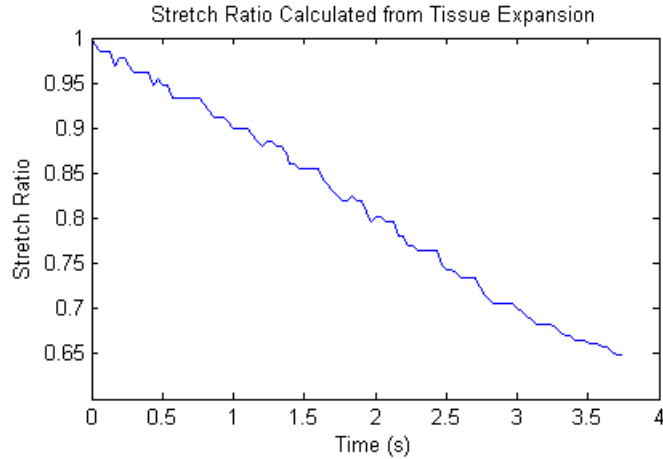


Figure 2.13: Stretch ratio in the compression direction, λ_1 , versus time.

To calculate the stress in the tissue sample during the unconfined compression test, the measured force data and information about the sample geometry were used.

The nominal stress is defined by:

$$\sigma = \frac{F}{A_o}$$

where F is the force and A_o is the original cross-sectional area of the tissue sample. To determine the original cross-sectional area of the sample, properties from the frozen sample and initial height of the thawed sample were used. An assumption was made that the volume of the tissue sample remained preserved through the thawing process. The following relationship determined the original cross-sectional

are of the thawed tissue sample:

$$V_f = \pi r_f^2 h_f$$
$$A_o = \frac{V_f}{h_o}$$

where V_f is the volume of the frozen sample, r_f is the radius of the frozen sample, h_f is the height of the frozen sample, and h_o is the original height of the thawed sample prior to compression. Using these relationships, it was possible to determine the nominal stress in the tissue sample throughout the unconfined compression test as seen in Figure 2.14.

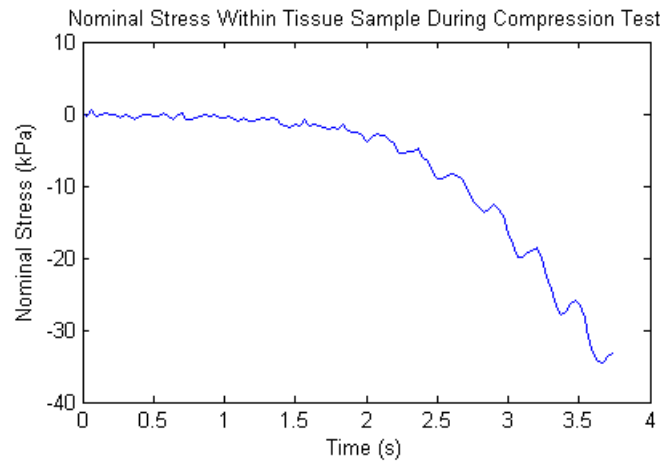


Figure 2.14: Nominal stress in the tissue sample throughout the unconfined compression test.

Combining the data from the stretch ratio and the nominal stress for each individual test resulted in the desired stress-strain curves for the compression portion of the total strain spectrum. Figure 2.15 shows the stress-strain curve for the compression test detailed in Figure 2.14. The process was repeated for multiple samples.

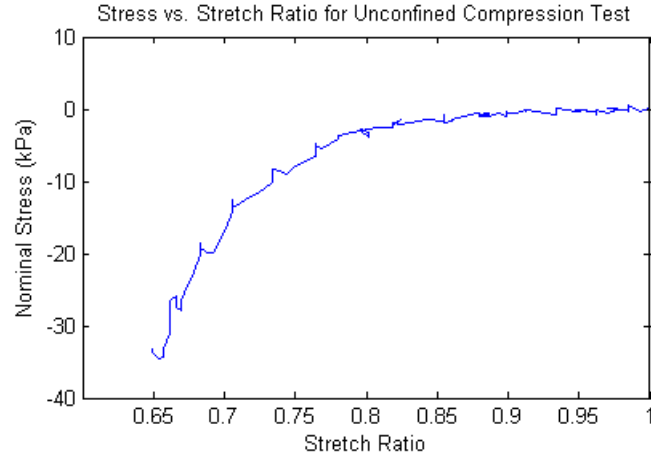


Figure 2.15: Nominal stress in the tissue sample throughout the unconfined compression vs the stretch ratio.

After determination of the individual segments of the stress-strain curve from each deformation mode, the three different data sets can be combined to formulate a complete characterization of the tissue response to loading conditions. Figure 2.16 shows the complete stress-strain curve for two sample runs for each tension, unconfined compression and pure shear deformation modes. As can be seen in the figure, the measured strain in the compression test was corrected by a factor which is approximated by the ratio of the height before and after sample thawing to compensate for the settling of the tissue due to gravity. This results in a shift of the compression curve to the left.

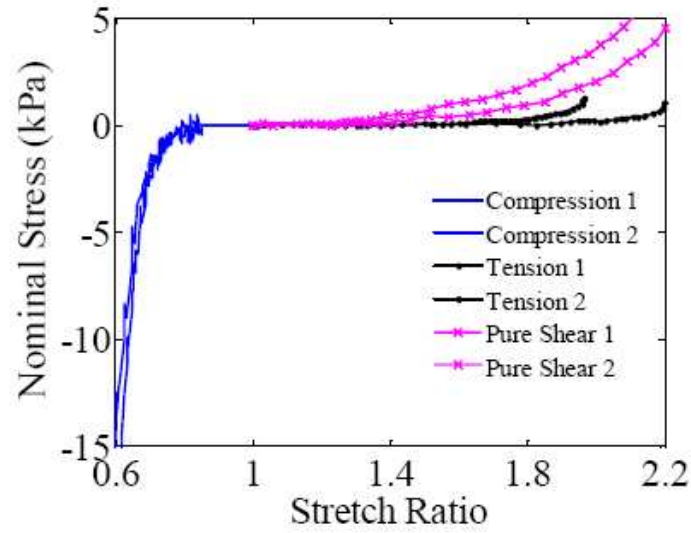


Figure 2.16: Experimental results for tension, compression and pure shear tests.

©2010 Springer.

2.1.4 *Ex Vivo* Model Development

After the experimental stress-stain curves were derived for the various tissue samples, the data was used to generate the hyperelastic constitutive models for *ex vivo* porcine liver. The modeling process, was developed and implemented by Dr. Zhan Gao using principles reported in [13][18][27][28][63][68]. Through this process three constitutive models of differing forms were developed which accurately represent the general loading of the tissue samples. The models are presented in Table 2.1

Table 2.1: Constitutive models for general loading of *ex vivo* porcine liver.

Model Type	Strain Energy Density Function	Model Parameters
Combined Logarithmic and Ogden Model	$-C_1 \ln(1 - C_2(\lambda_1^\alpha + \lambda_2^\alpha + \lambda_3^\alpha - 3))$	$C_1 = 60 \frac{N}{m^2}, C_2 = 0.632, \alpha = -2.544$
Combined Exponential and Ogden Model	$C_1(e^{C_2(\lambda_1^\alpha + \lambda_2^\alpha + \lambda_3^\alpha)} - 1)$	$C_1 = 10 \frac{N}{m^2}, C_2 = 1.6, \alpha = -3.068$
Ogden Model	$\sum_k \frac{2\mu_k}{\alpha_k} (\lambda_1^{\alpha_k} + \lambda_2^{\alpha_k} + \lambda_3^{\alpha_k} - 3)$	$\mu_1 = 7505.680 \frac{N}{m^2}, \alpha_1 = -2.773, \mu_2 = 0.155 \frac{N}{m^2}, \alpha_2 = 19.668$ $\mu_3 = -13856.8 \frac{N}{m^2}, \alpha_3 = -3.644, \mu_4 = 6421.31 \frac{N}{m^2}, \alpha_4 = -4.515$

2.2 *In Vivo* Modeling

When dealing with biological materials such as porcine liver, it is important to look into the differences between the tissue response in the *in vivo* and *ex vivo* states. Many previous studies have show that various biological materials behave differently in the *in vivo* state due to a variety of conditions; the variation between the response of the two states can vary drastically depending on the tissue type. For that reason, a study of the *in vivo* tissue response to loading was undertaken. In this study, two main goals were set forth. First, the efficacy of utilizing an *ex vivo* based material model for simulation of an *in vivo* probing test would be evaluated. This is an important aspect to research in that if the model is sufficiently accurate, it may be possible to only perform the more straightforward *ex vivo* test to generate the required material models used in surgical simulators. Second, if the response of the tissue using the *ex vivo* based tissue model does not accurately represent the actual tissue response, new models based on the *in vivo* experimental data should be generated to determine a more accurate representation of the tool-tissue interaction in the *in vivo* tests.

To test the validity of using the *ex vivo* based material models to simulate an *in vivo* experiment, a testing device was designed and constructed to perform a probing task on *in vivo* porcine liver with a hemispherical probe. The device is capable of recording the force imparted on the probe as well as the deformation of the tissue surface at discrete data points throughout the entirety of the probing task. Additionally, the camera incorporated with the probing device can be used

in a manner similar to a standard coordinate measuring machine (CMM) whereby the shape of the organ can be determined. Using this information it was possible to simulate the probing task in a standard finite element program, ABAQUS was used throughout this study, where the only variable in the analysis is the definition of the liver material characteristics. By using the *ex vivo* based constitutive model as the material parameters, the efficacy using the model for *in vivo* simulation was determined by comparing the simulation to the experimental force and displacement profiles. Further, the model was modified to better represent the properties of the *in vivo* tissue response.

2.2.1 *In Vivo* Experimental Device

The device designed and constructed for the collection of the *in vivo* experimental data (Figure 2.17) was developed to perform a controlled probing motion while recording the information required for accurate simulation of the probing task. When designing the device, many aspects had to be included that would allow the structure of the device to transform. This was due to the variability of many aspects pertaining to the animal which would interface with the device. For instance, the size of the pig could vary with each experiment, the shape and location of the liver would be different and the lighting conditions would change throughout the probing tests. In light of these facts, the design for the *in vivo* experimental set-up was made to be as versatile as possible. In addition, the device was also intended to be used for other tasks such as scalpel cutting and possibly needle insertion, therefore the design

allows for multiple initial configurations to accommodate the various desired tasks. Throughout this chapter, only the probing test will be explained, so the detailed description about the device will be confined to the probing mode configuration.

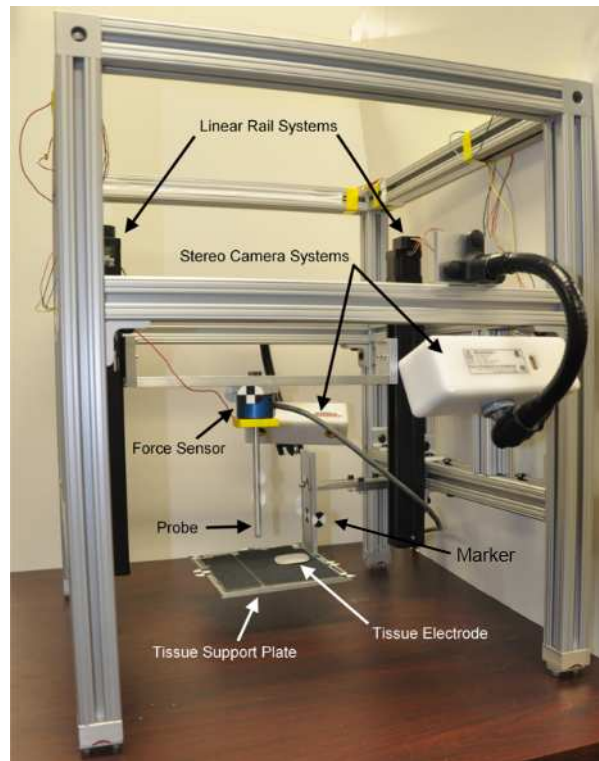


Figure 2.17: *In vivo* experimental device configured for the probing task. ©2010 IEEE.

The frame of the test device was constructed from 1515 Lite framing material (80/20 Inc.) to provide a rigid yet light support to connect the various sensors and actuators. The dimensions of the frame were determined to fit the standard surgical tables found in most pig laboratories. A clamp system was fabricated and attached to the bottom of each leg for fixation to the handles on the side of the tables. The device uses linear rails to provide the motion of the probe which will be discussed in detail in the next few paragraphs. To mount the linear rails a frame consisting of two

parallel supports with vertical reinforcements was utilized. This support assembly could be raised or lowered as needed to provide the proper starting location for the probe. An advantage of using the 80/20 framing material is the ability to alter the connections at individual locations without the need to completely dismantle the frame; providing the flexibility to raise and lower the linear rail starting positions as needed during each individual experiment. In addition to the vertical adjustment, the side rail assembly was designed such that the rails could slide horizontally to any location. The combination of these two motions provided the required flexibility to interface with animals of various sizes and organ locations. Figure 2.18 shows the support assembly which houses the linear rails in detail and shows the possible adjustments that can be made on the fly. On the sides perpendicular to the rails a single support bar was added. This support was designed to house the camera system and was also adjustable such that the bar could be raised or lowered without taking apart the device to properly orient the cameras.

Two linear rail systems (Haydon Motion Solutions), connected by a support beam containing the probing equipment, were rigidly mounted to the frame. The rails consist of a stepper motor (Size 17 double stack with 0.015875mm per step resolution), a wedge style anti-backlash nut, a optical encoder (US Digital E5S, 500 counts per revolution) and 305mm of total travel. They are controlled via a TTL-compatible motor control unit (DCM 8028) which directly drives the stepping motion of the motor. The speed of the motor is proportional to the frequency of the input signal. For the current setup, the frequency signal range is set between 0 and 2KHz which corresponds to a traveller speed of 0 to 100 millimeters per

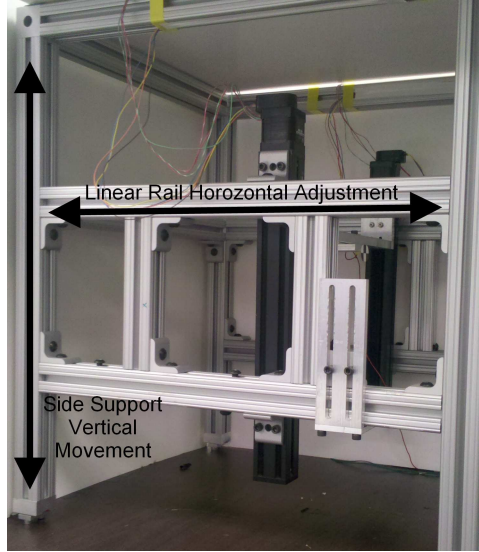


Figure 2.18: Support assembly for linear rail systems with adjustment capability possible during experimentation.

second. To generate a TTL-compatible signal, a control code was developed in C and implemented in conjunction with a Sensoray data acquisition system (Model 626). The two linear rail systems were connected with a rigid beam upon which the probe and sensors would be mounted.

The limitation of the proposed system occurs in the inability of the code to generate a signal in excess of 1KHz due to hardware limitations of the computer equipment. For the current application, an input signal in the range of 0 to 1KHz was sufficient for the desired probing speeds. To generate the proper linear motion of the probe itself, it was imperative that both linear rail systems moved in the same manner such that wedging of the rigid support did not lock up the motion of the system. Therefore, a proportional derivative based controller was implemented to assure the position and velocity of the rails remained constant throughout the

entire experiment. However, the addition of a control law used computational power which limited the maximum frequency of the output signal to be on the order of 100Hz, thereby greatly decreasing the maximum speed of the system. To counteract this limitation, a constant frequency is written to the motor controllers during a given experiment which will cause the traveller to move a specified distance. The distance of each rail is independently governed to assure proper movement in each rail regardless of loading conditions or differing frictional force. The speed between probing tests can be varied by changing the control signal frequency. This process was extensively tested in the lab to verify that the system would not lock up during experimentation.

Attached to the rigid beam connecting the two linear rail systems was the probe assembly which consisted of a force sensor and the rigid probe. The rigid beam was slotted such that the position of the probing assembly could be changed easily during the experimentation process. This flexibility, in combination with the ability to change the placement of the linear rails as previously mentioned, provided the capacity to move the starting location of the probe to almost any point within the frame of the testing device. The six-axis force/torque sensor (JR3 Model number 20E12A-I25) utilized in the experimental device records the force and torque data with a resolution of 0.002N and 0.00025Nm respectively. The force sensor signal was captured through the use of the Sensoray data acquisition board at a sampling frequency of 62.5Hz. Attached to the force sensor was the rigid hemispherical probe. Five different probes were constructed from aluminum with diameters of 6.5, 9.5, 12.5, 19 and 25mm respectively.

Beneath the probe, a support system was fabricated upon which the lobe of liver could be positioned. This support plate (200mm X 250mm) would isolate the liver from the surrounding internal organs. By including this rigid support plate, all measured tissue deformation could confidently be described as deformation of the liver itself instead of a combination of liver, lung and stomach. The support plate was connected to the rigid frame through a set of adjustable plates allowing for the ability to move the plate directly under the liver without the need to move the animal. By allowing the adjustments in all parts of the experimental device, the test apparatus was successfully designed and constructed with minimal knowledge of the size, shape or orientation of the animal with which it would interact.

The last major component of the experimental device was a camera system capable of recording the surface deformation throughout the probing task. In addition, it must also be capable of determining the shape of the liver before experimentation, as this information is required for the accurate simulation of the probing task. Various camera systems were evaluated for their applicability to the desired task including: digital image/volume correlation devices, structured light systems, motion capture systems and optical pose systems. The digital image/volume correlation devices provide a means for non-contact measurement of the full strain field as was implemented in the previous *ex vivo* study; however, problems would potentially arise with loss of data from reflections and the requirement of the fine texture for proper function. These issues in conjunction with the high price of the available units at the time resulted in their removal from the list of possible camera systems. Along the same approach, structured light systems project a unique light pattern

(often a laser grid or speckled flash) onto the surface of a substrate which can then be analysed with a stereo camera system to generate a three dimensional representation of the surface. These devices are capable of reproducing very accurate representations of the surface profile, however most systems are designed for quality control inspection and have acquisition times on the order of one to three seconds. The fastest structured light system contained a 2Hz imaging frequency, which was too slow for our system.

The selection of the camera system came down to two different options: a motion capture system by PhaseSpace Inc., and a optical pose system manufactured by Claron Technologies. The motion capture system was designed and used mainly for animation purposes. The system included a set of cameras mounted around the desired field of view which interact with light emitting diode (LED) markers which have a unique frequency profile. The spatial location of each marker can be determined from the combination of the individual locations in each of the camera frames. The unique frequency of each diode allows for active registration of the marker in each camera frame for easy correlation between the various cameras. This particular company developed markers that could be mounted on the face to aid in the animation of facial details, making the markers small enough to mount on the surface of the liver. The main advantage of using this system is the fact that it is an active system. Each marker is uniquely sensed and the spatial locations are automatically calculated for each point, making the surface tracking algorithm that would need to be written relatively simple. Problems arose, however, when studying the markers in more detail. Despite the fact that the makers were very small in size,

they were all LED's that needed to be wired for the system to work properly. When it came to mounting the markers on the surface of the liver, the wires themselves limited the number of markers that could be used before they began to be obscured by the wiring. Also, it was very time consuming and difficult to fix the markers to the liver surface making it infeasible for the current experimental protocol which limited the interaction time with the pig to five hours. Therefore, the PhaseSpace motion capture system was not selected for the imaging device.

The final camera chosen for the *in vivo* experiments was the MicronTracker stereo camera system (Model H40) developed by Claron Technologies. The system was similar to the motion capture system in that the spatial location of markers on the surface of the liver could be tracked in real time, however these markers were passive, meaning they did not emit a unique signature which distinguishes one from another, making the data processing more difficult. Also, instead of using a set of individual cameras, a stereo camera was utilized, which limited the field of view. This optical pose system utilizes unique markers (Figure 2.19) which could be detected by the camera system. Using the information for the location of the marker in each image, the global position with respect to the camera can be calculated. Testing of the markers revealed that the system worked properly with markers as small as 5mm in diameter and had robust tracking with markers 7mm and above. The markers themselves could be printed on paper and placed on the surface of the object being tracked. Therefore, since no wires were required, a large number of markers could be added to the surface of the liver without the registration problems seen in the active tracking systems.

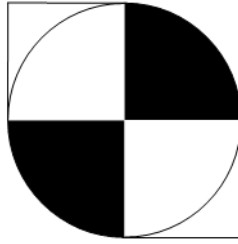


Figure 2.19: Marker used in MicronTracker camera system.

In addition to the individual point tracking, the MicronTracker system could be used to track tools. One such tool was developed that could be tracked and through a simple coordinate translation, the location of the stylus tip could be recorded. By using this technology with discrete data points it was possible to turn the camera system into a coordinate measuring machine (CMM). Using the MicronTracker in the CMM mode it was possible to generate a point cloud representation of the liver surface which could be transformed into a three dimensional model of the organ. Therefore, by choosing this imaging system it was possible to track the surface displacement of the liver at the marker locations in addition to obtaining an accurate representation to the starting organ geometry. The model purchased had a field of view which was acceptable for the size of the experimental device, an accuracy of 0.25mm on the volume and a sampling frequency of approximately 15Hz. Details pertaining to the specific use of imaging system in the experiments and data analysis are presented in Sections 2.2.2 and 2.2.3.

After performing an initial *in vivo* experiment to root out any issues with the device, two major modifications were made. First, as will be discussed in the next few sections, when the probe contacts the liver surface large deformations

occur which resulted in the rotation of some of the markers placed on the surface. Additionally, the probe itself acts to obscure the markers from the camera system. Because the imaging system is stereo camera based, the marker must appear in both frames to be tracked properly. Even a small diameter probe proved to cause a large gap in the field of vision for the markers on the back side of the probe. In light of these two facts, a second camera system was purchased that could be mounted on the opposite side (as can be seen in Figure 2.17) to collect the data which would be obscured or lost due to marker rotation. Also, it was difficult to determine the point at which the probe comes in contact with the tissue surface. Details about importance of the probe-tissue contact determination will be discussed in 2.2.3, however it is sufficient at this point to know that an accurate time of contact was required. Therefore, a contact determination circuit was added to the *in vivo* probing device.

Due to the soft nature of pig liver it is often difficult to determine the exact contact point directly from the force profile. The reaction force on the probe gradually increases as the probe is driven into the tissue and there is no discernible jump in the force profile upon initiation of probe-tissue contact. Various methods have been implemented to solve this problem and determine the precise instant the probe makes contact with the liver surface. Miller *et al.* utilized laser distance meter to measure the radial displacement during *ex vivo* compression tests to determine the initial contact location [52]. In this study, we propose an approach consisting of a contact-determination probe whereby the probe-tissue contact point could be realized similar to the method presented by Reilly *et al.* [67]. An electrode was attached

to the probe itself and a second electrode was placed between the tissue specimen and the supporting plate. Contact between the probe and the tissue resulted in the completion of an electrical circuit thereby producing a signal that could record the time of contact. The electrical circuit was designed such that low current would pass through the circuit to avoid damage to the tissue. Using this signal it was possible to compare the measured experimental force and surface displacement with the simulation of the probing test.

Through the use of this experimental setup it was possible to obtain, with a high level of accuracy, the parameters required for a complete simulation of the probing task as well as the experimental values with which to compare the simulation. Given the tissue geometry, boundary conditions, and probe location along with the force and displacement data, it was possible to perform a simulation of the *in vivo* experiment whereby the only variable in the analysis consisted of the material properties. Comparing the resultant force imparted to the probe and the deformation of the surface from the simulation with the measured values resulted in a method to validate the efficacy of the proposed constitutive model to simulation of an *in vivo* probing task.

2.2.2 *In Vivo* Experiments

Two *in vivo* experiments have been performed at the George Washington University Medical Center pig lab. The tests were conducted on female pigs weighing 18 and 20.5kg under an approved protocol by Institutional Animal Care and Use

Committee (IACUC). Throughout the entirety of each test, a trained veterinarian and anesthesiologist were on staff to handle the animal and supervise the interaction of the testing device with the liver. The animal was sedated and properly fixed to the surgical table where the abdominal cavity of the pig was opened by the veterinarian to expose the liver. During this process the anesthesiologist also supervised the administration of anesthesia as required, observed the vital signs and maintained proper function of the attached respirator used to control the pig's breathing throughout the entirety of the experiment. Once the animal was situated, the experimental device set up for the probing task was positioned above the animal and one of the lobes of liver was placed on top of the support plate (Figure 2.20). This plate was covered with a high grit, waterproof sand paper to prevent sliding of the liver during the experiments. The support plate was positioned slightly above the abdominal cavity leaving approximately a 3mm gap between the bottom of the plate and the abdominal cavity at the apex of the breathing motion. The proper positioning of the plate was important in that it allowed for the isolation of the liver from the surrounding organs and prevented the relative motion of the organ due to breathing. The liver was positioned such that a large portion of the lobe was housed on the plate, leaving enough slack in the end connected to the diaphragm to absorb the slight movements due to the breathing of the animal. During the first test, not enough slack was given to the organ during placement on the support which resulted in unwanted motion during the analysis process as will be shown in section 2.2.3.1. After the liver was properly positioned on the support plate, it remained in that orientation throughout the entirety of the testing. When the liver was not being

used for an experiment, a damp cloth was placed over the surface to prevent the organ from drying out, which could alter the response of the tissue to loading.

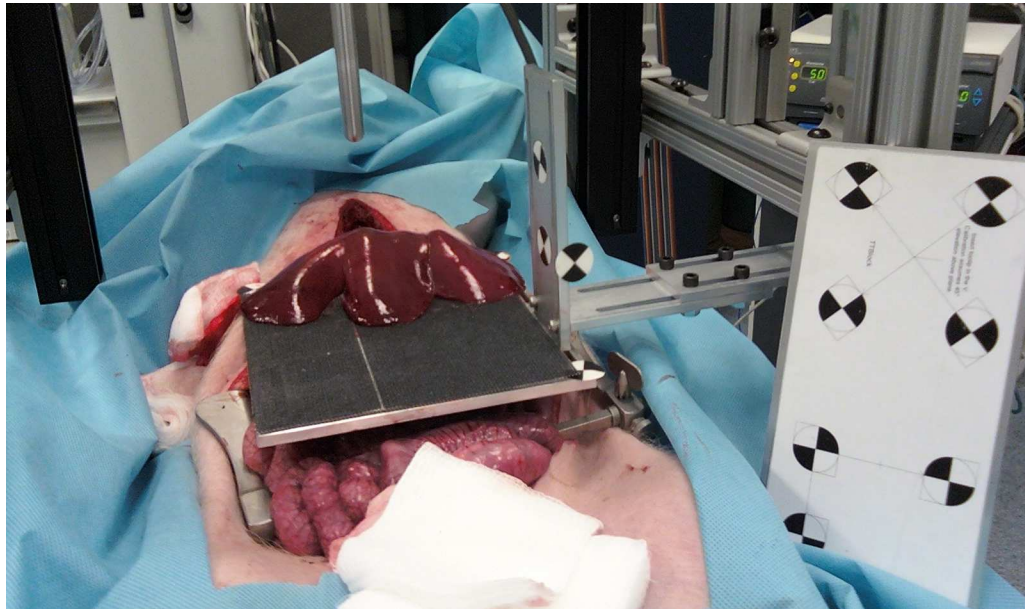


Figure 2.20: Positioning of the support plate with liver placement during first pig lab.

As previously mentioned, an accurate simulation of the probing task requires knowledge of the shape of the organ in addition to the proper boundary conditions present during the testing process. To obtain the organ geometry, the MicronTracker camera system was used in a manner similar to a CMM machine. A code interfaced the camera with Rhinoceros[®] 4.0 (a 3D drawing package manufactured by Robert McNeel & Associates) and stored the location of the points as they were recorded. A stylus was lightly traced by hand over the surface of the liver while the points corresponding to the tool tip were recorded. This process was continued until the entire surface of the liver was sufficiently sampled. The device displays the points in real time through the Rhinoceros[®] interface so it was possible during the reg-

istration process to assure a proper sampling of the organ for future use. After the process was completed, a point cloud representation of the liver surface represented the original shape of the organ. Figure 2.21 shows the raw point cloud data corresponding to the liver geometry from the first pig lab shown in Figure 2.20.

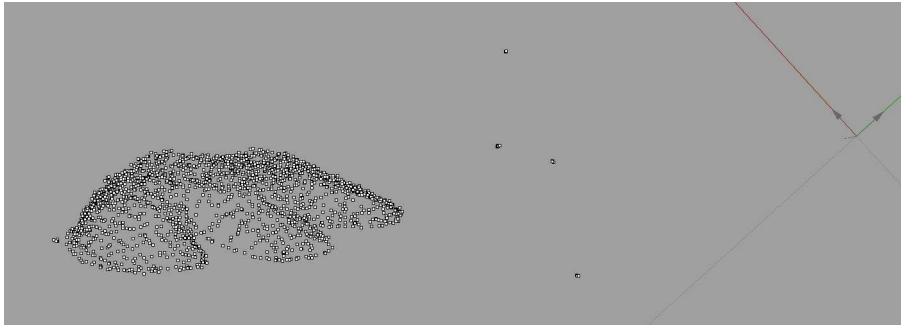


Figure 2.21: Raw point cloud data obtained during first pig lab.

During this process of obtaining the point cloud representation of the organ geometry, the camera needed to be located outside the testing frame to properly track the stylus for a complete representation of the liver. However, during the probing experiment, the camera needed to be moved inside the experimental device for robust tracking of the individual markers placed on the liver surface. Therefore additional reference markers were added to the system at locations on the support plate and the testing frame. These markers could be measured in each of the two camera locations to provide the information needed to transfer between the two reference frames. The reference markers and the corresponding point locations can be seen in Figures 2.20 and 2.21 respectively.

After completion of the geometry measurement, the device was set up to perform the probing task. Using the three adjustment options built into the experi-

mental device, the probe was positioned over the desired location for the particular test. Next, the markers that could be tracked with the MicronTracker system were fixed to the surface of the liver. Markers on the order of 7mm in diameter were printed on standard white paper and placed directly onto the surface of the liver. Due to the fact that the surface is naturally moist, and kept moist throughout the experiment, the papers markers naturally adhered and conformed to the shape of the liver. The natural adhesion of the markers is useful in that it will not alter the tissue response to the probing load as might be the case with the application of glue or another material to the liver surface. A large number of markers were added to the surface proximate to the probe location to maximize the number of points that could be tracked on the surface. The marker layout for one of the probing locations of the first experiment can be seen in Figure 2.22.



Figure 2.22: Location of markers during one test location in the first pig lab.

After the markers were properly positioned on the surface around the probe, the camera system was moved into position such that the surface markers and at least three of the reference markers were recorded in the field of view of both cameras constituting the stereo pair. This camera location provided the information needed to track the surface markers throughout the deformation process as well as correlate the displacement data with the initial organ geometry through the use of the reference marker location. When everything was set properly, the probing task was initiated whereby the probe was driven into the liver tissue, at a speed of 1.25 millimeters per second, while recording the displacement of the probe, the reaction force on the probe and the stereo camera images for surface marker analysis. As a safety precaution, the experiment would stop if the reaction force on the probe exceeded a threshold force set to be 3N. Upon completion of the probing task, the probe was moved to a different section of the liver and the process was repeated, including placing new markers as needed to obtain a proper sampling of the liver surface proximal to the new probe location.

2.2.3 Analysis of *In Vivo* Experimental Data

After the *in vivo* probing experiment was completed, the raw data recorded from the test was analyzed to extract the information required to set up the simulation and compare of the experimental and simulated results. In this context, the force and surface displacement data were acquired and the geometry for the liver sample was derived for the simulation. The acquisition of the force profile was

easily derived from the recorded raw force data. However, the analysis of the image data was much more involved. As mentioned in the previous sections, various changes were implemented in the device after the first experiment took place. The differences were namely in the introduction of the contact determination system and alterations in the imaging techniques. The reason for the changes will be discussed in the presentation of the data from the first experiment.

2.2.3.1 First Experiment

The calculation of the force profile was initially determined directly from the force sensor readings. During the first experiment there was no defined measure to determine the instant when the probing process began. In setting up the probing experiment, the probe was positioned slightly above the tissue surface to avoid beginning the probing process prior to recording the relevant information. Therefore the starting location for the probing test was estimated from the recorded still images of the experiment. The beginning of the test was defined as the first frame in which motion of any tissue was visually observable. A plot of the force profile in conjunction with the probe displacement is presented in Figure 2.23.

After the force profile was acquired the next main set of data to obtain was the displacement of the markers on the surface of the liver. Using the MicronTracker camera system, it was possible to record the global position of each marker with respect to the camera location. As previously mentioned, the system utilizes passive markers, therefore each marker is numbered depending on its location in each

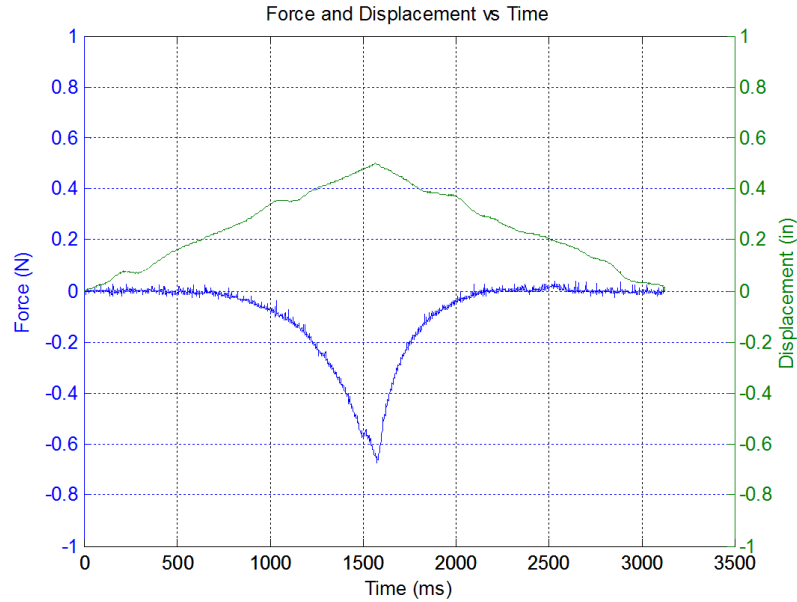


Figure 2.23: Force and probe displacement profile for one probing test during the first experiment.

individual frame. The number is assigned through a raster scan of the image; if the markers changed location with respect to one another the order could be altered frame by frame. Throughout the experiment, reflections on the surface of the liver or changes in the angle of the markers created many false reading for the markers or prevented recognition of a particular marker. Due to these complications, the data had to be filtered to determine the actual motion of each individual marker.

To determine the unique marker displacement, a code was developed in Matlab to filter the raw point data recorded by the MicronTracker. A flow chart describing the functionality of the code is presented in Figure 2.24 which is repeated for each individual marker. Starting at the beginning of the probing task, the initial position of the marker of interest was stored. For each frame a mask was established (a 6mm spherical region around the starting position) which would be used to determine

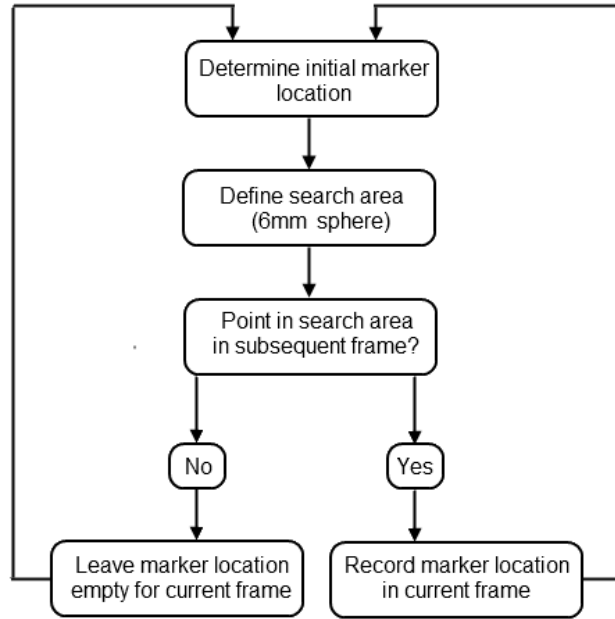


Figure 2.24: Steps for organizing the raw point data stored by MicronTracker into an organized structure for each marker.

its location in the subsequent frame. Using the known position of the mask, the program would look in the list of point locations for the next recorded frame for a point within the defined mask. If a point was located in this region, it was determined to be the next location of the marker and is stored as such. If no such point was found, no data was stored for the marker during that frame. The process was repeated for each frame using the last known location of the point as the center of the mask. At the end of the process an additional filter was added whereby if the marker was not recognized in a minimum number of frames (set to be at least 25 frames, which corresponds to approximately 20% of the duration of the probing test) it would be disregarded entirely. By this technique, the displacement of each marker could be uniquely determined and all spurious information relating to false

readings and renumbering of the markers could be filtered out. After the algorithm was implemented, a structured array contained the data for the location of each unique marker throughout the entirety of the probing task.

Using the above approach the data for the probing test was evaluated. As can be seen in Figure 2.25 the algorithm was successful in detecting and tracking the majority of the markers on the surface. Some of the markers, including many on the back side of the probe and a few in the foreground, were not recognized in enough of the frames and were therefore ignored by the tracking algorithm. Figure 2.25b shows a three dimensional plot of the displacement of the markers that were found through the filtering algorithm. As can be seen in the plot, there was apparent motion for the majority of the markers that were expected to move, whereas the reference markers and those far away from the probe appeared stationary. Included in the figure is a close up of the overall displacement of marker number five which showed a great deal of movement as would be expected for a marker so close to the location of the probe-tissue contact. It was not until each marker was looked at in detail that the problems begin to arise.

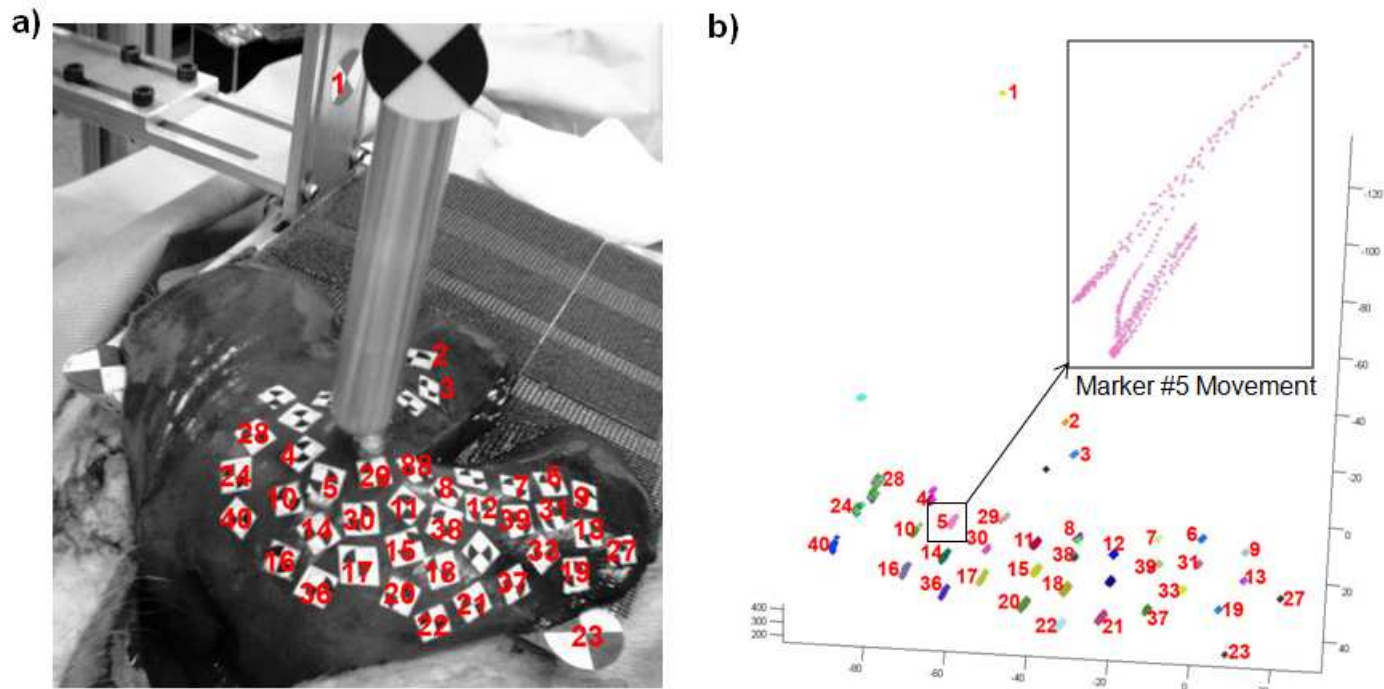


Figure 2.25: a) Marker location with labels defined for markers of interest b) Measured marker displacement throughout probing task after filtering data.

When looking into the details of the fifth marker displacement, it was apparent that there was motion present in addition to the motion induced through the probing task. Figure 2.26a shows the magnitude of the displacement of the marker with respect to the image frame number throughout the duration of the image recording process. From visual inspection of the recorded images, the estimate for the probe-tissue contact point was approximately frame number 120. However there was significant motion of the marker, approximately 4mm, at various points prior to the probe contact. This trend was seen in the majority of the markers. In all of the markers the displacement also displayed a cyclical pattern which correlated with the time period of the ventilator. Therefore, it was determined that the motion recorded was relative motion of the liver do to the pig's breathing. This fact led to the realization that the placement of the plate and the liver location on the support plate was a crucial part of the experimental set-up.

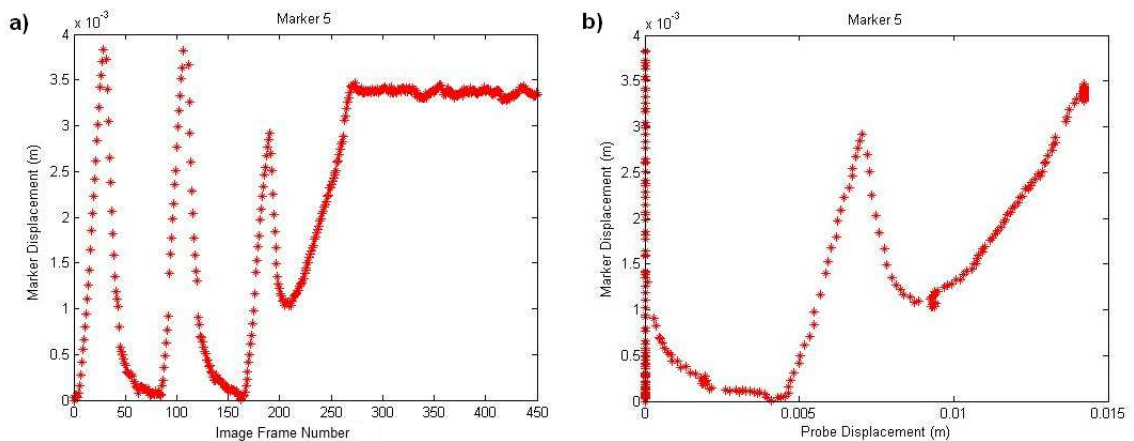


Figure 2.26: a) Sample marker displacement through the entirety of the test b) Sample marker displacement vs probe displacement.

Figure 2.26b shows a plot of the magnitude of the displacement of marker number five with respect to the motion of the probe. After removing the unnecessary information prior to the initiation of the probing experiment, it was observed that the overall displacement was a function of both the breathing and the induced tissue movement due to the probing task. There was once again a peak corresponding to the breathing motion; however, during the settling process, the marker did not return to the original location but began to displace due to the probing motion as was expected. Overall the marker moved a total of 3.5mm due to the relative motion of the tissue as a result of the probing process. After the probing was completed, the probe remained in place and the tissue remained compressed with the marker holding at 3.5mm of displacement even through the breathing cycle, as can be seen in Figure 2.26a. It should also be noted that the probe-tissue contact point was estimated from the recorded images. If this estimate is not correct there would be a shift to the right or left in figure 2.26b. Even with a small discrepancy in the starting frame number, the shift of the displacement curve would be substantial and could lead to inaccuracies in the validity of the comparison between the experimental and simulated data. Therefore, a method was devised to improve the accuracy of the probe-contact determination.

The final issue that arose with the original approach was due to the camera system itself. The MicronTracker system is not typically used with small, flexible markers like those in this experiment. Therefore, some difficulties occurred in tracking all of the markers throughout the entirety of the probing task. One of the major problems developed from the rotation of the markers due to large displacements

arising in markers close to the probe. In studying the images themselves it was apparent that the pattern used for the automatic recognition was severely altered after it rotated away from the ideal perpendicular configuration. During this process the marker displacement data was not collected automatically, however it was still possible to visually detect the center of the marker. A plot of the typical results for one of the markers during the test is shown in Figure 2.27. As can be seen, a substantial amount of data was lost which is essential for proper comparison to the simulated probing task. This dilemma was resolved by developing a new strategy for acquisition of the marker displacement for subsequent experiments.

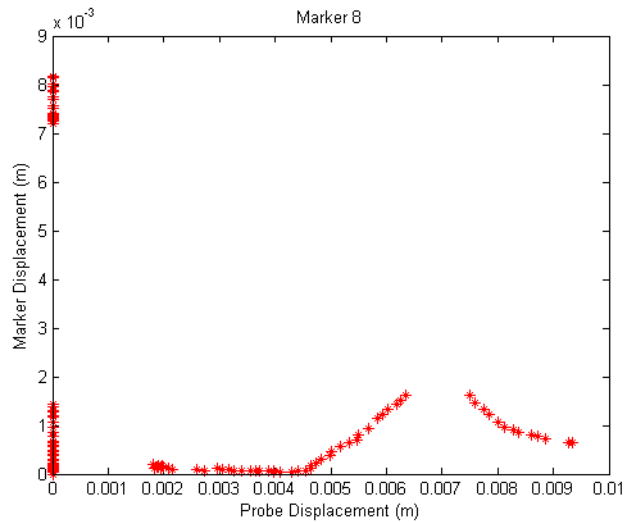


Figure 2.27: Typical marker tracking data for first experiment.

The next task that needed to be solved was the development of a geometrical model of the organ based on the original shape of the the liver which could be used for the foundation of the probing simulation task. This was generated by using the point cloud data obtained during the experiment with the MicronTracker camera system.

Upon completion of the surface registration procedure, a point cloud representation of the tissue surface was generated in Rhinoceros[®]. Using this point cloud data, it was possible to create a surface patch to best fit the data points. Being that the tissue is very soft, there is inherent error to the measurements and the accuracy of the surface can be determined to within only a few millimeters. By changing the parameters of the surface patch in Rhinoceros[®], different degrees of smoothing can be applied to the surface to obtain the proper shape. The function that generates the surface implements an algorithm which minimizes the error between the surface and the points, therefore, it results in the equivalent of a midplane along all the point cloud data. This is the ideal representation because there are unavoidable errors in the measurement process due to the soft nature of the liver. Therefore, during the generation of the point cloud, some of the points would be above the surface, when the stylus is slightly above the surface, and some would be below the surface, when the stylus slightly compresses the tissue during measurement.

Next, a plane was fit to the nodes that comprise the border of the tissue along the edge of the supporting plate. The surface representing the liver geometry was then extruded in the direction perpendicular to the plane, extending far enough that the entire liver section is solid at the cross section of the plane. By using a boolean split, the solid shape was split along the line of the plane creating two disjoint shapes, one representing the liver and the other which was disregarded. The solid model of the liver could then be used to generate the mesh for finite element simulation or could be imported directly into ABAQUS for the simulation of the probing task. A depiction of the point cloud data with the final three dimensional

geometry used during the simulation process for the first experiment can be seen in Figure 2.28

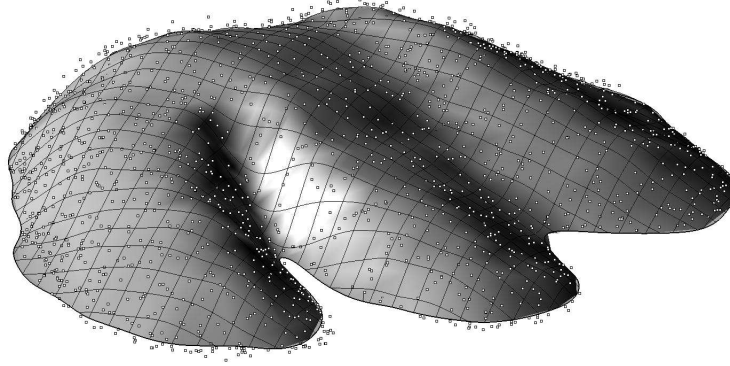


Figure 2.28: Solid geometry developed from measured point cloud data for first experiment.

2.2.3.2 Second Experiment

After resolving the problems that arose during the initial testing of the device mentioned above, a second *in vivo* porcine experiment was conducted. In this experiment probing was conducted at nine different locations on one lobe of liver. After a study of the data it was apparent that three of the probing locations were performed without any problems such as relative motion due to breathing or insufficient sample thickness at the probing location. All three of the tests that worked properly were analyzed in detail to test the efficacy of utilizing the *ex vivo* based models for *in vivo* simulation as well as implement improvements of the material model based on the recorded data. These three tests will be presented in detail from this point forth.

During the second *in vivo* experiment, the reaction force on the probe was recorded in the same manner as the initial test. Figure 2.29 shows a depiction of the reaction force incident on the probe for each of the three probe locations. The measured values that are plotted as the raw data points. As can be seen, the data had some fluctuation due to the inherent noise in the force sensor. To provide a more structured method for comparison to the outputs of the simulated probing tests, all force data was fit with a continuous polynomial curve. The raw force data along with the polynomial fits are both depicted in the Figure 2.29.

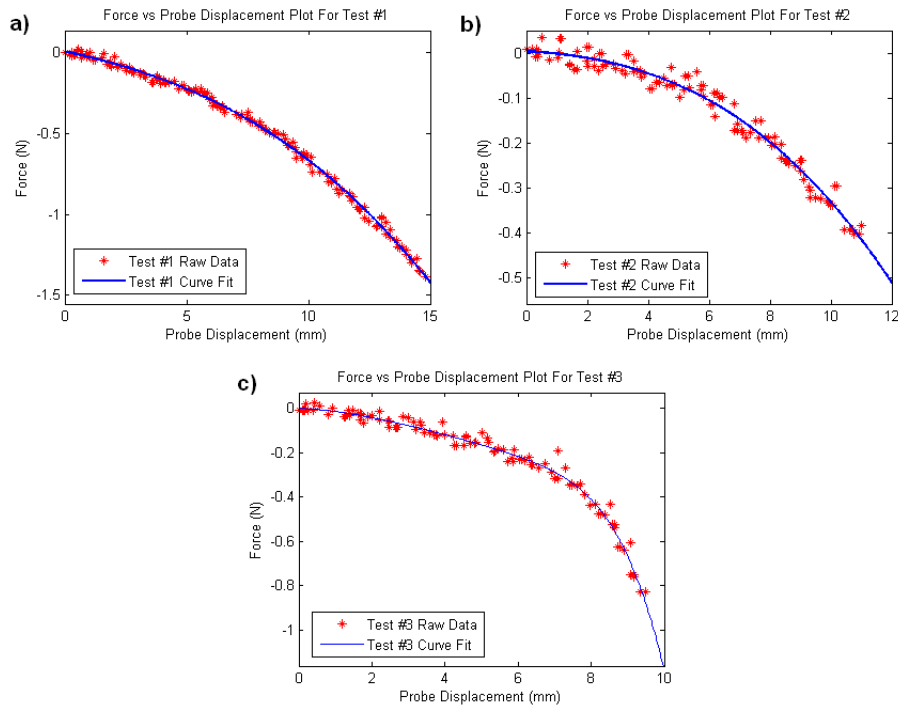


Figure 2.29: Measured force values and corresponding polynomial fits for the three probing locations.

One of the major problems with the data analysis during the first experiment was the inability to determine the instant at which the probe made contact with

the tissue. This is the time when the simulation and the experimental data should begin to be compared. To resolve this problem, an electrical circuit-based contact determination system (previously described) was utilized to provide superior localization of the tool-tissue contact. A depiction of the results from the circuit can be seen in Fig 2.30, plotted in conjunction with the raw force data. Seen in the plot, the voltage jump occurred immediately upon contact with the tissue surface. It should also be noted that there was no distinct change in the gradually increasing force profile that would have indicated the start of the probing action. Therefore, the use of the contact determination system allowed for an increase in accuracy when comparing the experimental and simulated results. From this point on, all data presented will be from the start of the probing task as defined by the contact determination probe.

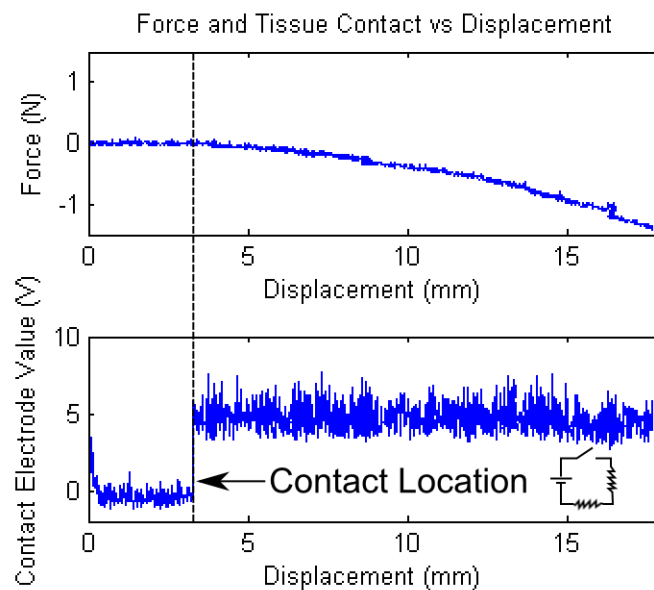


Figure 2.30: (Top) Raw force data obtained during probing task (Bottom) Measured voltage levels throughout the probing task. ©2011 Springer.

The other source of error that was overcome for the successive experiments was the determination of the accurate surface displacement. The MicronTracker camera system was still used to obtain the surface displacement information, however, modifications to the process of obtaining data was required. One of the problems that arose was the limitation of the marker locations that could be used while maintaining registration of the marker in both the left and right images of the stereo pair due to the probe itself. This limited placement of markers on the back side of the tissue surface. To overcome this problem a second MicronTracker camera system was purchased to mount on the opposite side of the probe. This camera, with the addition of a separate firewire card capable of running both cards simultaneously at 15Hz, provided a complete field of view of the entire liver surface. Modifications to the camera code were implemented to allow for the acquisition of both camera images.

The last obstacle arose from the lack of reliability of the MicronTracker system to automatically detect every marker location throughout the probing process regardless of lighting conditions or rotation of the marker itself. While it may have been possible to develop a more robust tracking code for the markers, or use a different shape altogether, the tracking algorithm implemented by MicronTracker could not be altered in the currently available programming framework. To improve this, a method for tracking the markers was developed whereby each marker location could be uniquely determined throughout the probing process. In addition, the approach allowed for the tracking of any point which was uniquely defined through the entire experiment, which led to the ability to track not only the center of the

markers but the corners of the paper themselves. As the paper adhered completely to the moist surface of the liver, this increased the number of points on the organ that were possible to track.

The method that was adopted for marker registration was a manual selection technique. It was decided that the variety of the errors that arose in the images due to changes in the light intensity of reflections on the surface, in addition to the possibility for large marker rotations close to the probe, would be very difficult to overcome in an automated image processing manner. Therefore a code was developed in Matlab whereby the pixel location of the points of interest, in both the right and left image frame, could be selected with the mouse. In this process it was possible to magnify the image to obtain a more accurate selection of the point location on the subpixel range to increase the selection accuracy. Through this process, a list of pixel locations was developed for each point of interest at every frame throughout the probing experiment. Then a second code was developed in C through the use of the MicronTracker programming framework that would import the pixel location data for each marker and transform the pixel information to the global location using the properties of the stereo camera configuration. After completing the process, the location of the points of interest with reference to the camera location were obtained as in the previous process; however the existence of data loss due to lighting issues or rotation was greatly reduced.

Due to the fact that the new process was performed visually, there was the possibility for larger variation in the accuracy of the pixel location throughout the probing task. This variability is observed when looking at the raw data for one of

the probing locations in the second attempt at the *in vivo* experiment. To make the comparison to the simulated displacement more straightforward, the displacement of each marker was fit with a polynomial curve which accurately represented the data similar to the recorded force response. The marker displacement information for each of the three probing locations is shown in Figures 2.31, 2.32, and 2.33 with the polynomial approximations included in the plots. For each of the three tests eight to eleven markers were chosen to represent the surface deformation in the area proximal to the probing location. Markers further away proved to have movements that were too slight to accurately track with the current imaging technique and were not included in the analysis.

The next task is the extraction of the final three-dimensional geometry from the point cloud representation recorded at the start of the experimental process. The procedure remained unchanged from that presented during the first pig lab. All three of the probe locations being analyzed occurred on the same lobe of liver; therefore, when reconstructing the geometry of the organ, only one representation is needed. Due to the limits imposed on the probing depth and maximum force for safety reasons, at the end of an individual probing task, the liver returns to its original starting shape. Figure 2.34a shows an image taken of the lobe of liver after the position was finalized on the support plate. Figure 2.34b depicts the final three dimensional solid model that was utilized as the basis for the simulation of the probing experiments. Additionally, the location of the three probing tests being analyzed are shown in the image.

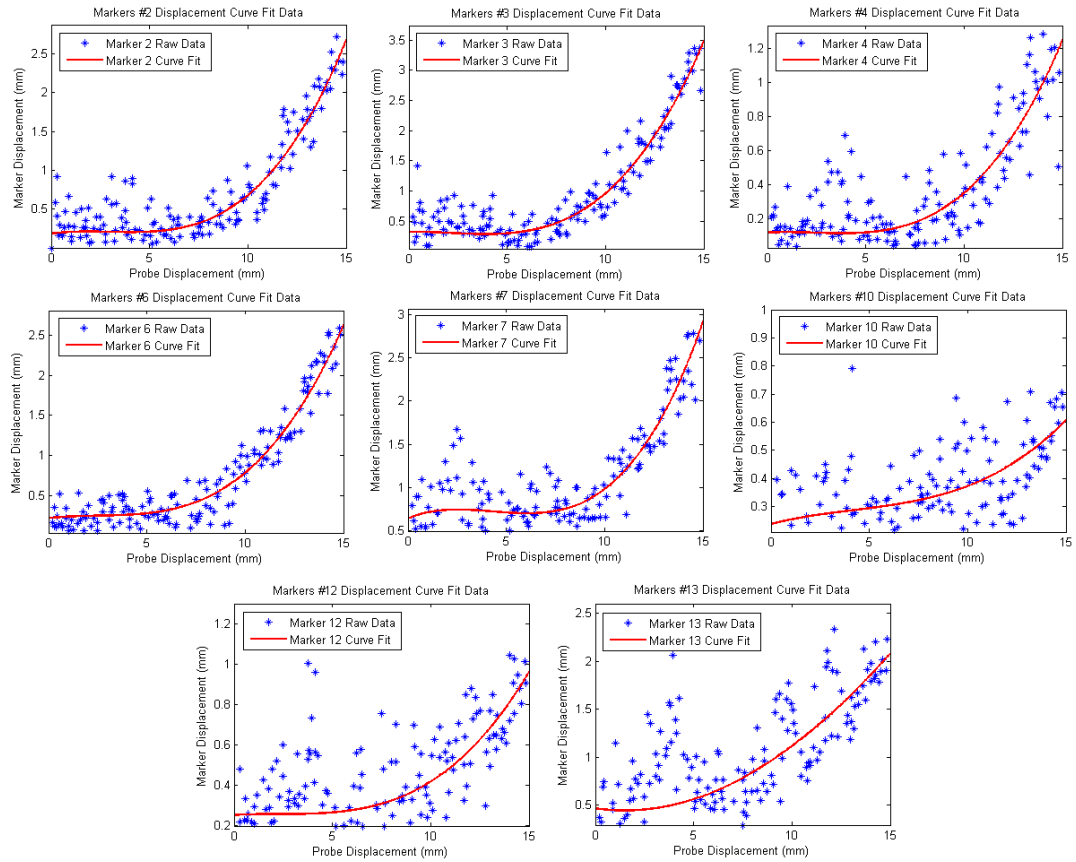


Figure 2.31: Presentation of the raw marker displacement data along with the polynomial fits for first probe location.

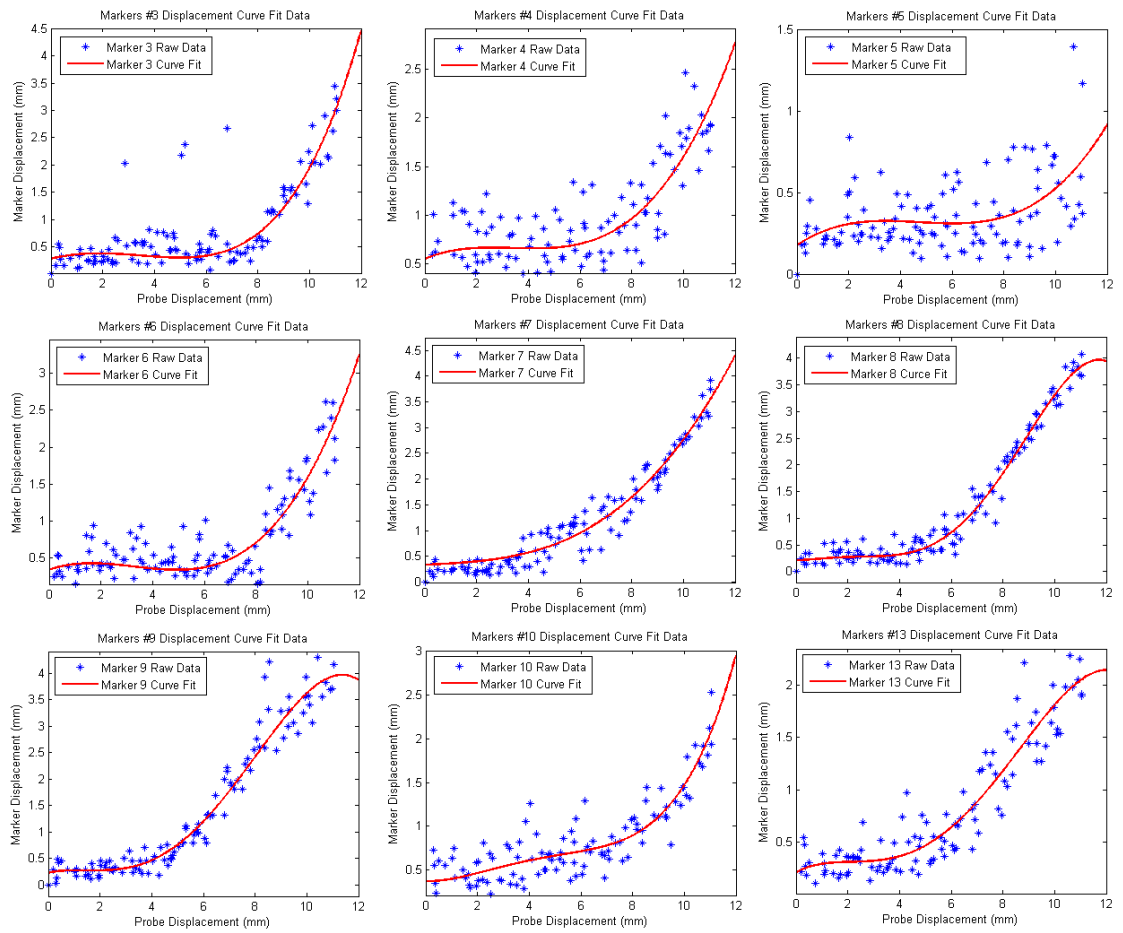


Figure 2.32: Presentation of the raw marker displacement data along with the polynomial fits for second probe location.

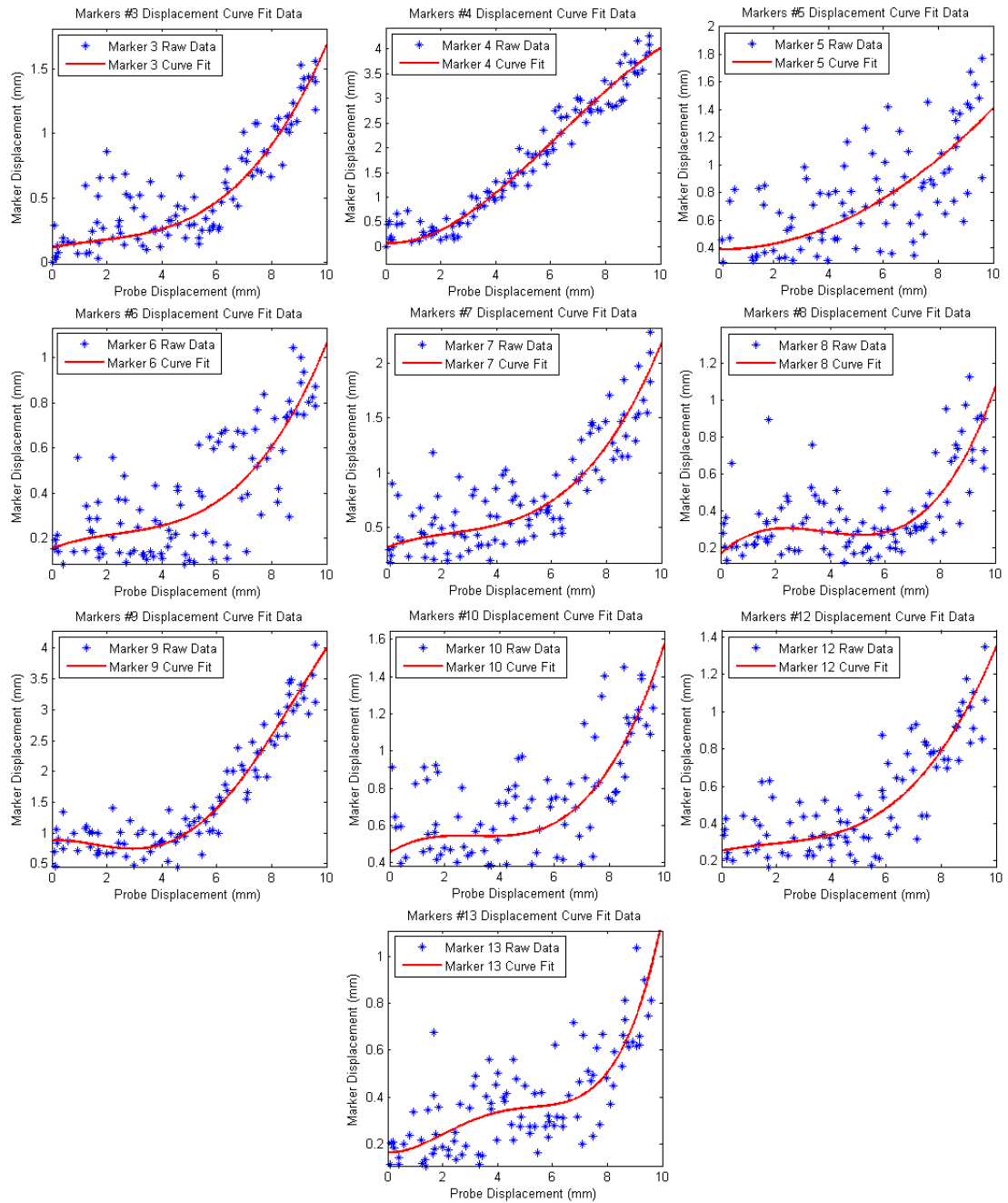


Figure 2.33: Presentation of the raw marker displacement data along with the polynomial fits for third probe location.

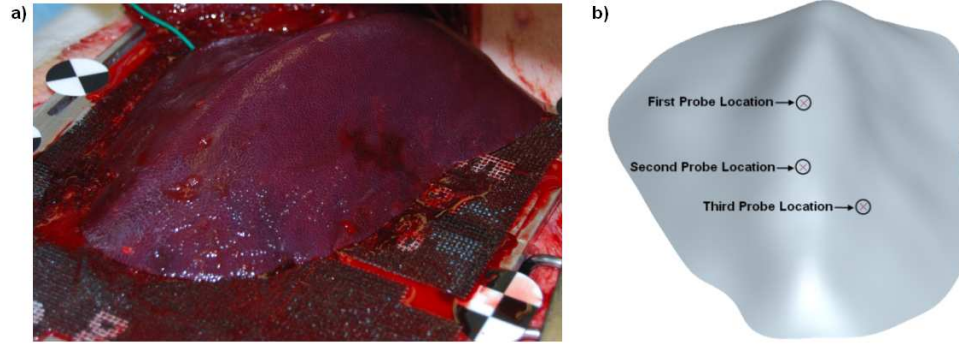


Figure 2.34: a) Image of the *in vivo* liver segment tested. b) Corresponding three dimensional model developed from point cloud data with three probing locations labeled.

2.2.4 Finite Element Simulation of *In Vivo* Experiment

Using the information obtained during the data analysis it was possible to reproduce the probing experiment in the context of a finite element simulation where the only variable in the analysis was the material properties of the liver tissue. To perform this analysis the shape of the liver, location of the probe, probing depth, boundary conditions and material properties all needed to be fully defined. Using the data recorded during the experimental probing run, along with the information processed after the experiment, it was possible to accurately define all of these required characteristics.

The shape of the liver was derived from the point cloud data as previously described. However, for proper use in a finite element simulation, a mesh must be generated for use in the analysis instead of a solid model. Due to the large nature of the liver organ and the small nature of the deformations and probe diameter (9.5mm in all three proing locations) a refined mesh is required throughout the model which

requires a large number of elements. To reduce the required computation time, and negative effects due to volumetric locking, linear hexahedral elements are utilized in the mesh. However, due to the complex nature of the organ, ABAQUS is not capable of automatically generating a hexahedral mesh for the required geometry. A meshing software, HyperMesh (manufactured by Altair Engineering), with a more powerful mesh generation interface was utilized. Dr. Zhan Gao was able to successfully mesh the accurate liver geometry with a total of 87,941 elements (Figure 2.35). The final mesh consisted of 84,339 linear hexahedral (C3D8RH) elements and 3,602 linear wedge (C3D6H) elements. Reduced integration and enhanced hourglass control were utilized to prevent volumetric locking and eliminate instabilities. Hybrid elements were chosen due to the incompressibility of biological tissues.

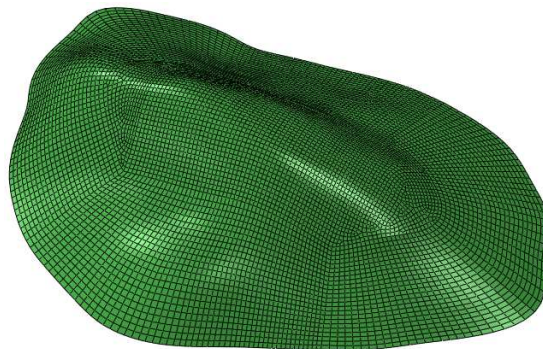


Figure 2.35: Mesh for lobe of liver used for finite element simulation. ©2010 IEEE.

Upon completion of the mesh, the information pertaining to the location of the probe and probe displacement must be entered into the simulation. As mentioned previously, the coordinate systems used for the generation of the liver geometry and the acquisition of the surface displacement information were different since the camera needed to be relocated for each probing run. However, reference markers

that remained stationary throughout were recorded in both coordinate systems. Therefore, a method for transferring the information between the reference frames developed by Arun *et al.* was utilized to obtain all information in one reference frame [2]. Due to the fact that transformation was required, an additional reference frame was set up to simplify the finite element simulation process. In this reference frame the X-Y plane was aligned with the bottom of the liver surface. This provided the means to define the movement of the probe directly along the Z axis, simplifying the set-up in ABAQUS. The method for representing the data in any coordinate system is summarized below based on [2]:

Let the set of coordinates of the reference points in two different coordinate frames be defined as $\{p_i\}$ and $\{p'_i\}$. With p_i and p'_i being 3x1 column matrices, the relationship between the two can be given by:

$$p'_i = Rp_i + T$$

where R is a 3x3 rotation matrix and T is a 3x1 translation vector. To find the proper R and T the following error function needs to be minimized:

$$\Sigma^2 = \sum_{i=1}^N \|p'_i - (Rp_i + T)\|^2$$

It has been previously shown that the centroid of $\{p_i\}$ and $\{Rp_i\}$ will be the same point [37]. Therefore let us define the centroid of each set as p' and p :

$$p' = \frac{1}{N} \sum_{i=1}^N p'_i$$

$$p = \frac{1}{N} \sum_{i=1}^N p_i$$

Next define the difference between the reference point location and the centroid as:

$$q'_i = p'_i - p'$$

$$q_i = p_i - p$$

Then calculate the 3 x 3 matrix, H :

$$H = \sum_{i=1}^N q_i q_i'^t$$

Find the singular value decomposition of H :

$$H = U \Lambda V^t$$

Using the information from the singular value decomposition of H , define a term, X , as:

$$X = UV^t$$

If the $\det(X) = +1$, then $R = X$. If the $\det(X) = -1$, then $R = V'U^t$. The translation vector T is found by:

$$T = p' - Rp$$

Using this approach it was possible to transform all information recorded pertaining to the shape of the lobe of liver, marker information, probe position and displacements to the newly defined simulation coordinate system. From this point on all information presented will be in the simulation coordinate system previously described. Additionally, the transformation of all data to the new coordinate system provided the information required to place the probe in the proper location and define the probe movement for the finite element simulation.

The definition of the boundary conditions for the analysis which becomes slightly more involved. Based on the fact that the lobe of liver was placed on the sandpaper to prevent sliding or motion during the probing process, the bottom surface of the liver was fixed in the simulation. The next information that needed to be included was the contact definition between the probe and the tissue. Seeing as how the surface of the liver was kept moist throughout the experimental only a small amount of friction would be present between the probe and the liver. Experiments were not conducted to determine the exact coefficient of friction which would vary between each liver and depends on the quantity of water and blood on the surface. For the purpose of this experiment, a coefficient of friction of 0.1 was assumed. Using this information, a contact pair was established between the probe and the liver, where the probe was defined as the master surface and the liver was the slave. The analysis used a finite sliding, node to surface based approach to solve for the contact iterations.

The final boundary condition that needed to be addressed was the effect of gravity on the tissue. As in the *ex vivo* case, the tissue naturally settles due to the existence of the gravitational force. Owing to the soft nature of the liver, the settling of the liver due to gravity could have a large effect on the resultant force and the accuracy of the surface displacements, depending on the liver geometry and probe location. The shape of the liver that was measured to produce the point cloud representation is the geometry after settling has occurred due to gravity. However, in the current state the internal loads that should arise due to the settling process are not represented. For an accurate portrayal of the probing task, the internal loads

need to be accounted for prior to applying the probing action in the finite element simulation.

An approach we developed to obtain the proper internal stress distribution was to first determine the shape of the liver in an environment free of gravitational forces. Then using this new geometry as the starting point, a gravitational load was applied to cause the tissue to settle back to its original position. This process was studied in detail after the first pig lab experiment and the data presented to validate the approach will be from that study. To determine the shape in a gravity free state, a simulation was run whereby a gravitational load was applied in the opposite direction of the normal loading. This had the effect of stretching out the tissue sample as can be seen in Figure 2.36b. From the output of the simulation it was possible to create a new point cloud representation of the surface through extracting the location of the surface nodes. This data can then be imported to Rhinoceros[®] and the process for defining a new solid model is repeated. The new shape that is derived represents the organ in a gravity free environment (Figure 2.36d). Next, the new shape is used as the starting point for a new simulation where the gravity is applied in the proper direction. This has the effect of settling the tissue as is observed in the *in vivo* experiment. Using a similar process, the final shape which includes the internal stress configurations can be compared to the original point cloud data (Figure 2.36g). In doing so it is apparent that the shape derived from the gravitational loading process is indeed the same as the originally measured and generated solid model.

A similar process was used for the gravity compensation in the second pig

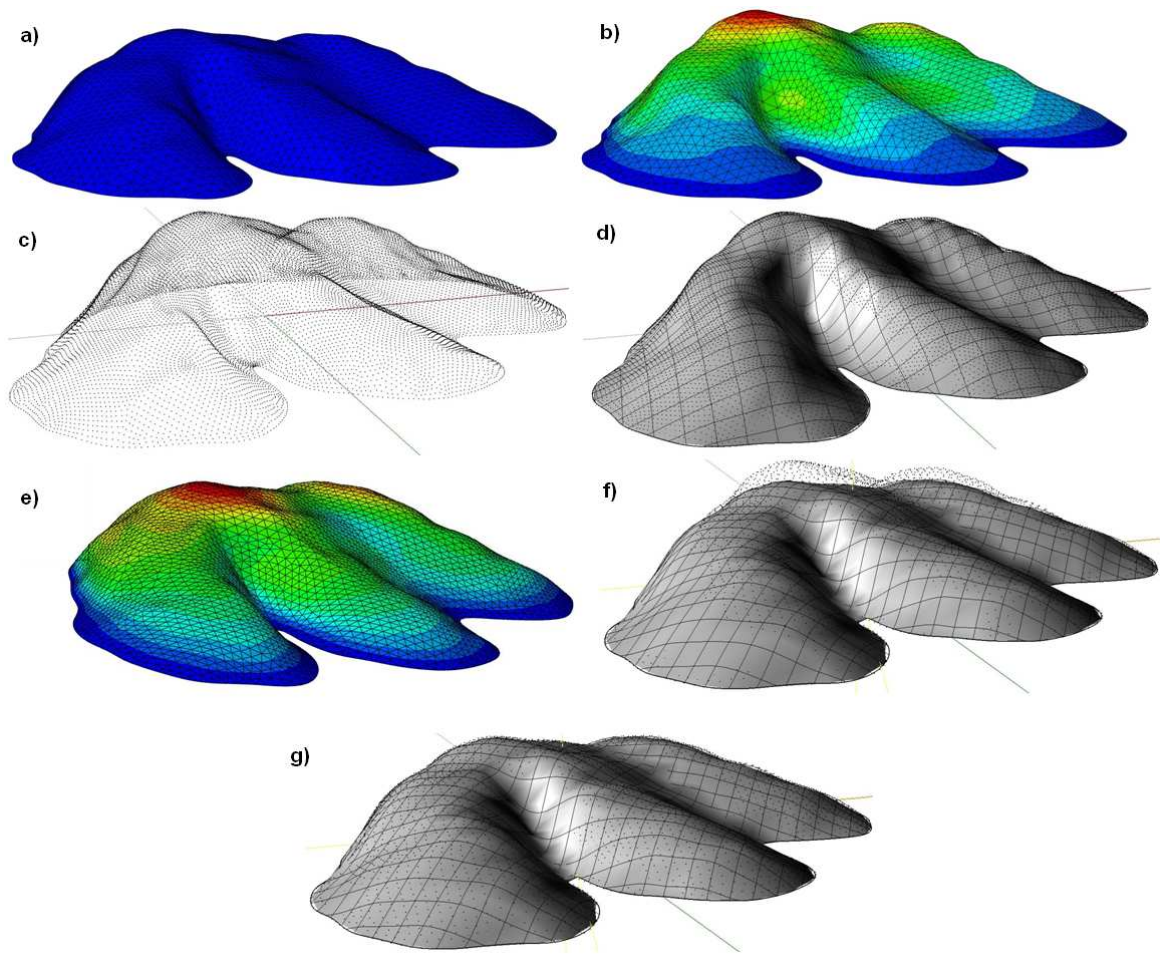


Figure 2.36: a) Original shape derived from point cloud data. b) Plot of shape and displacement magnitude after inverse gravitational loading. c) Point cloud representation of new tissue surface. d) New solid model for shape without gravity derived from point cloud data. e) Shape after applying gravity back to the model. f) Plot of shape after gravity process compared to point cloud data of gravity free shape. g) Plot of shape after gravity process compared to originally measured point cloud data.

lab probing tasks. However, after the detailed verification above, it was determined that the approach indeed worked as planned so the comparisons of the point cloud data in Rhinoceros[®] was unnecessary. To simplify the process, after the inverse gravitational load is applied the results from the analysis are used directly in the new simulation for the settling of the tissue. This can be achieved through some of the added functionality in ABAQUS, whereby the deformed mesh from the inverse gravity simulation can be imported directly and used as the starting mesh for the subsequent analysis. This approach bypassed the need to create a new solid model which would need to be remeshed prior to use. Figure 2.37 shows the stress distribution resultant in the tissue model following the gravitational loading process for the geometry used in the second experiment.

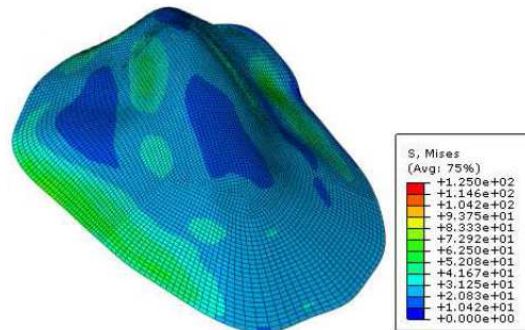


Figure 2.37: Organ geometry for second experiment after gravitational loading process containing the required internal stresses. ©2011 Springer.

For the second *in vivo* experiment, a separate method for verifying the accuracy of the starting shape after the gravitational loading process was derived. This process involved comparing the final shape after gravity loading to the initial position of the markers being tracked for the various probing tasks. The approach was a

more stringent check of the geometry formulation process because it uses data from the second camera position where the points were actually on the organ surface, instead of the data from the stylus which could be slightly above or below the tissue. In this method the location of the measured marker points were projected on the surface along the vertical (Z-axis) and the distance between the measured location and the projected locations was measured. The measured distance ranged from 0.01mm to 1.8mm over the locations of all surface markers used in the *in vivo* probing tests, with an average distance of 0.556mm. This process verified that the CMM measuring approach with the MicronTracker imaging system in conjunction with the gravitational compensation did in fact result in a highly accurate representation of the liver geometry.

Finally, the material model needs to be defined prior to the simulation of the probing task. For this experiment various models were utilized. The starting point for each of the probing locations was the four-term Ogden model, $U = \sum_{i=1}^N \frac{\mu_i}{\alpha_i} (\lambda_1^{\alpha_i} + \lambda_2^{\alpha_i} + \lambda_3^{\alpha_i} - 3)$, developed through the *ex vivo* experimentation as the bulk properties for the liver tissue. By using this model it was possible to determine the efficacy of utilizing a reality-based, *ex vivo* constitutive model for the simulation of *in vivo* tasks. After the material properties are defined in ABAQUS, the gravitational loading procedure is enacted. Then, after the tissue has settled to the proper state, the probe is driven into the tissue to the depth of the experimental test. It is important to repeat the gravitational loading procedure every time a different material model is used because the degree of settling will vary and the residual stress levels after the settling process will differ depending on the properties of the liver.

Inaccuracies in the force response as well as the surface deformations will arise if the gravitational process is not repeated with the proper model. Figure 2.38 depicts the displacement plots of the three probing locations after the probe was driven into the soft-tissue sample. Upon completion of the simulation, the information for the reaction force on the probe, probe displacement and surface deformation at each surface node was calculated. These values can be directly compared to the measured experimental values to determine the fit of the proposed material model.

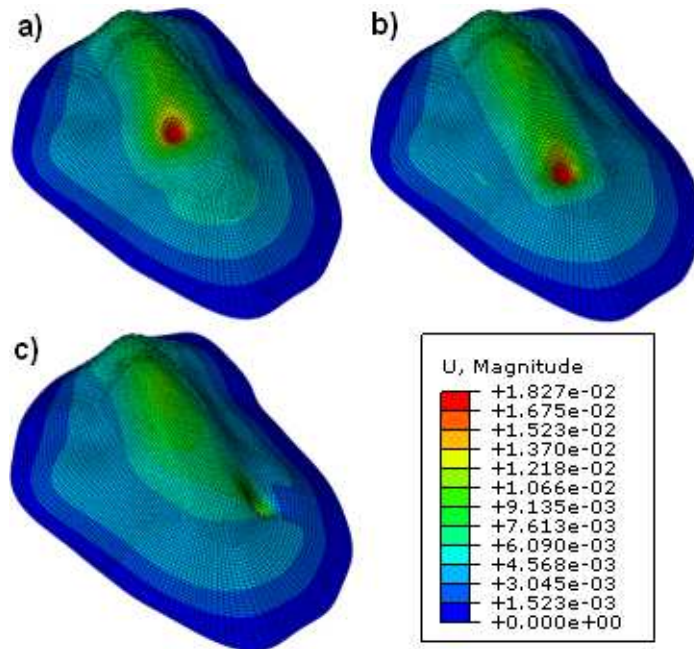


Figure 2.38: Displacement plots after probing simulation for the a) first probe location b) second probe location and c) third probe location.

2.2.5 Comparison of Experimental and Simulated Data

After all the data from the experimental side had been analyzed and the simulation of the probing task was completed, it was possible to evaluate the efficacy

of using a reality-based *ex vivo* material model for the simulation of an *in vivo* probing task on pig liver. To do so, both the force profile for the reaction force due to the probing task and the overall surface displacement were studied. The generation of the force profile was acquired through functionality built into ABAQUS, therefore this response was directly compared to the experimental force profile. A comparison of the force values for each of the three probing locations can be seen in Figure 2.39.

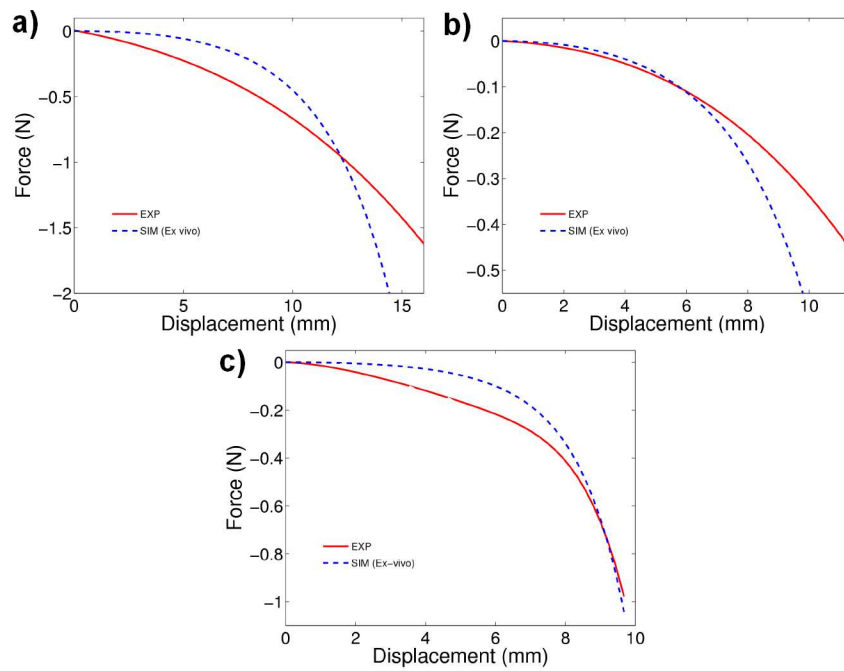


Figure 2.39: Comparison of the experimentally measured force profile and the simulated force response for the a) first probe location b) second probe location and c) third probe location. ©2011 Springer.

For the purposes of a reality-based simulation used for medical training purposes, a high degree of conformity between the experimental and simulated force response is desired due to the perceptive nature of human touch. In the case of using an *ex vivo* material model as the basis for a realistic simulation of *in vivo*

tissue response, the precision of the simulated force response is somewhat lacking. As can be seen in Figure 2.39, the simulation of the force is on the correct order of magnitude and the trends for each of the three probing locations are similar to the desired response. However, the error at certain locations can be as large as 0.5N. One additionally troubling aspect is the fact that the simulated and experimental force responses diverge so quickly in two of the three tests. Therefore, if the probing task were continued to an increased depth, the accuracy of the simulated force response would continuously degrade. Previous studies have shown that changes in force as small as 0.1N can be perceived by the human hand [8]. Therefore, in the context of a realistic training simulator, the *ex vivo* based models would not be able to accurately simulate the force response of an *in vivo* probing task.

To perform a comparison of the surface displacement data, some additional postprocessing was required for proper comparison to the experimental data. Using the coordinate transformation technique, the starting location of each surface marker of interest could be determined in the simulation reference frame. Using a code written in Matlab, it was possible to search through the list of all surface nodes utilized in the ABAQUS simulation for the node closest to each surface marker. Through this process a correlation between each individual experimental marker and the corresponding finite element node was generated. Then, the displacement of each node of interest was extracted from the output data developed during the ABAQUS simulation to construct the displacement information that was compared to the experimentally determined values. A comparison of the simulated surface node displacements corresponding to the experimentally measured marker movement is

presented in Figure 2.40, Figure 2.41 and Figure 2.42 for the first, second and third probing locations respectively.

Inspection of the surface displacement figures shows some variation between the fit of the simulation to the various markers. Many plots show the FEM nodes following the trends of the experimental markers well with accuracy in the submillimeter range, while others show more of a divergence from the proper experimental values and a large magnitude or displacement error. To better understand the comparison of the surface displacements, a quantitative root mean square (RMS) error approach was implemented. The RMS error was calculated for each of the marker-node pairs throughout the entirety of the probing distance. Thus:

$$RMSE_m = \sqrt{\frac{\sum_{i=0}^n (x_{s,i} - x_{m,i})^2}{n}} \quad (2.2)$$

where $RMSE_m$ is the RMS error of marker, m , n represents the total number of sample points in the data set, $x_{s,i}$ is the node location from the FEM simulation at sample point i , and $x_{m,i}$ is the marker location at sample point i . From these values the average RMS error, $RMSE_{ave}$, can be calculated to represent the fit of the simulated surface deformation to the measured surface deformation:

$$RMSE_{ave} = \frac{\sum_{i=0}^M RMSE_i}{M} \quad (2.3)$$

where M is the total number of markers analyzed.

Using the above approach with the markers selected for the analysis of the first probing location yields RMS values ranging from 0.118mm to 0.539mm for the individual marker-node pairs. An average of all individual marker-node pairs gives

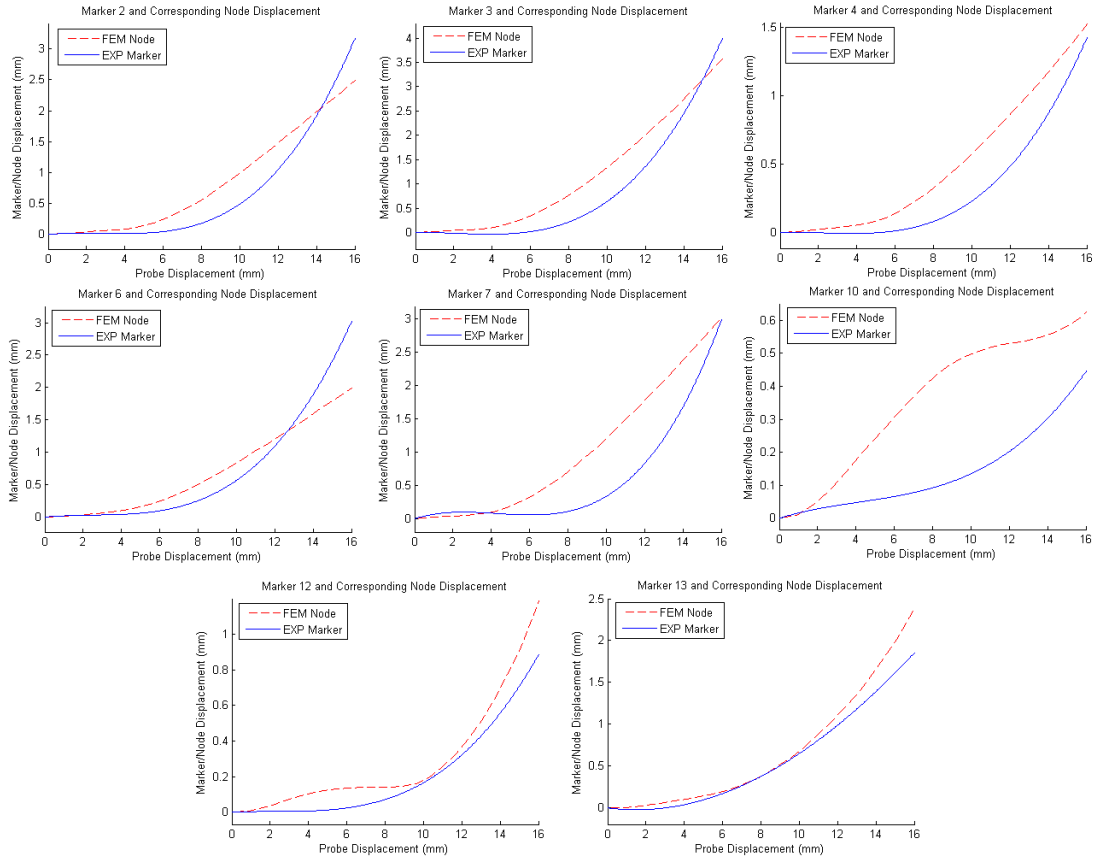


Figure 2.40: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the first probe location.

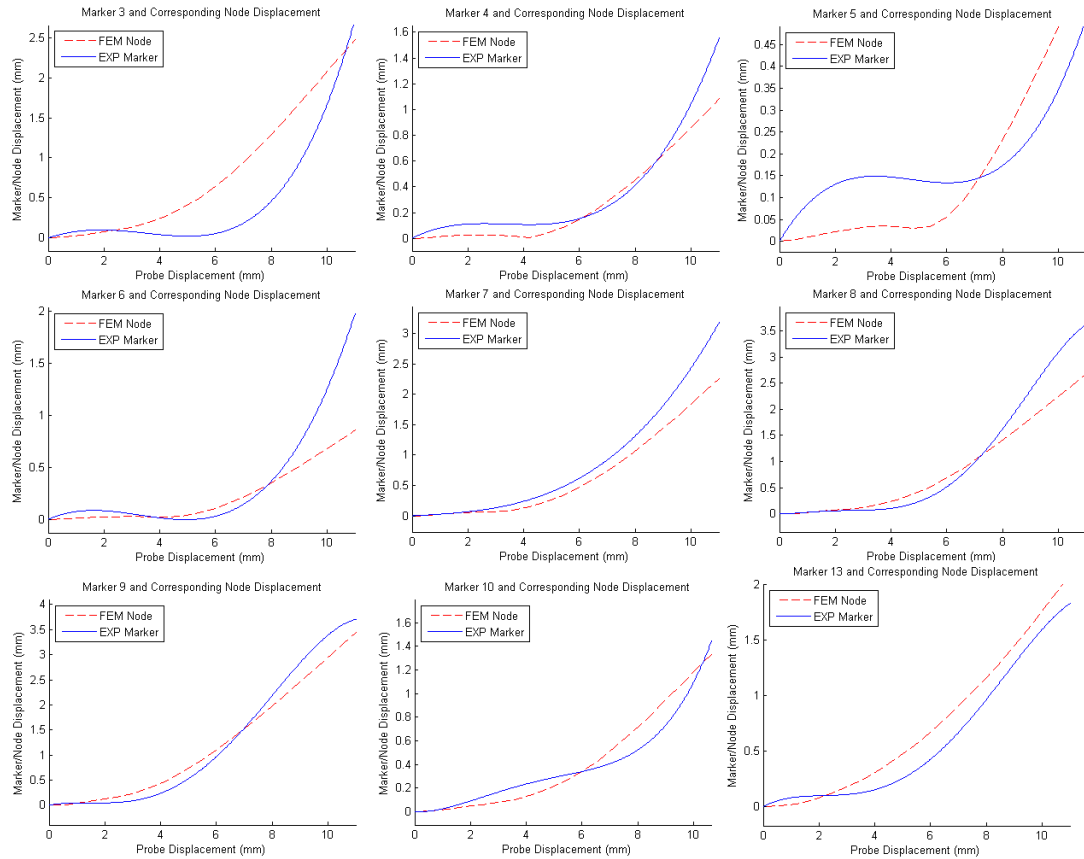


Figure 2.41: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the second probe location.

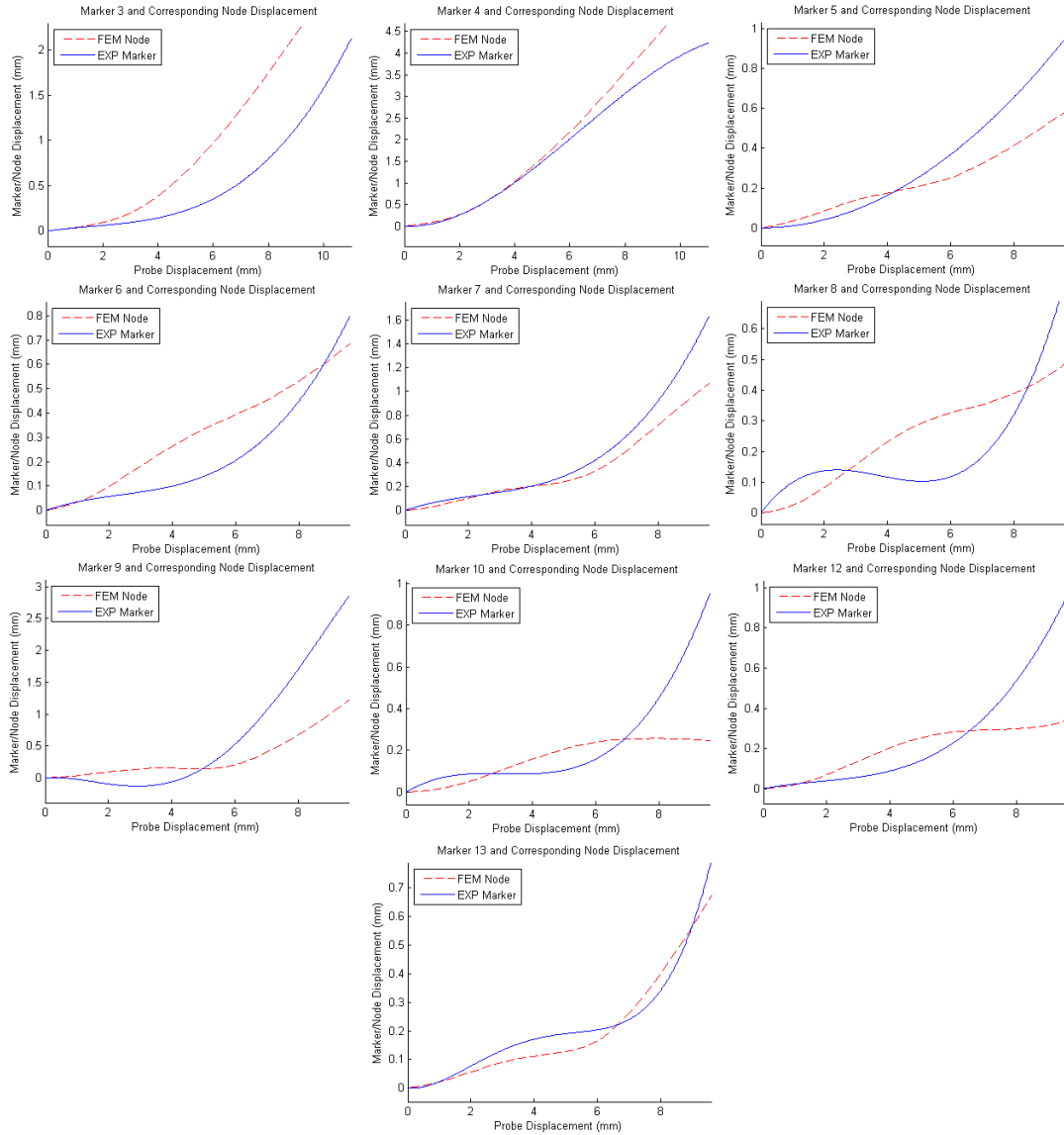


Figure 2.42: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the third probe location.

the overall fit of the simulated surface deformation to the experimentally measured surface deformation. In the case of the first probing location it is 0.307mm, with an average marker displacement of 0.564mm. Through previous studies, we have observed a significant amount of variation in soft-tissue response to tension, compression, and pure shear loading as well as tissue probing and cutting [26][25][36][11]. Hence, though the displacement error in this case may seem high, it is within the bounds of acceptable error for soft-tissue deformation. Also, for the purposes of simulation, it will be very difficult for the user to perceive a difference in tissue displacement of 0.3mm. The second and third probing locations respectively yield average RMS values of 0.237mm and 0.238mm.

The RMS error approach presented in Equation (2.2) is also used to develop a quantitative unit of measure for the accuracy of the force response throughout the entirety of the probing task. This approach yields RMS values of 0.42N, 0.17N and 0.08N for the three probing locations. In the context of a simulator, there is a large discrepancy between the magnitude of the errors that occur in the force and displacement domains. Due to the nature of the tissue, the surface displacement is on the order of a few millimeters in the region of interest surrounding the probe, however the error values that are witnessed are on the submillimeter range. Due to the relatively large reaction forces on the probe, the magnitude of the errors that arise can be as large as 0.4N. As previously mentioned a change in force of 0.1N is perceptible to the user, therefore the errors that arise due to the force response will be much more noticeable than the displacement response. Hence, a method has been developed to improve the system performance by focusing on improving the

representation of the *in vivo* force response.

2.2.6 *In vivo* Model Development

The compressive property modification and corresponding algorithm (Figure 2.43) were developed by Dr. Zhan Gao via the following:

$$T_a(\lambda) = T(\lambda) \frac{F_e(\lambda_e)}{F_s(\lambda_e)} \quad (2.4)$$

where $F_e(\lambda_e)$ stands for the force measured from the experiment and $F_s(\lambda_e)$ is the force obtained from the simulation and $T(\lambda)$ is the nominal stress. All the data fittings were generated by using the material evaluation capability in ABAQUS/CAE. The nominal stress and strain of compression data were entered as test data, and the model parameters were fitted for the selected model type. The material stability was checked to make sure the selected model was stable in the experimental strain range. There are more than five types of hyperelastic models one can choose from, namely, Arruda-Boyce, Mooney-Rivlin, polynomial, Ogden, reduced polynomial, etc. Ogden and reduced polynomial models have demonstrated their ability to fit the adjusted compression data and achieve stability at the same time.

This modification algorithm was applied to all three probing locations to improve upon the simulation response of the reaction force induced by the probe-tissue contact. Details for each step of the modification process for the first probing location follow. For each iteration of the algorithm, a new stress-strain curve was generated which would be used in ABAQUS to generate the new material model. Upon completion of the process, the new model was run through the probing sim-

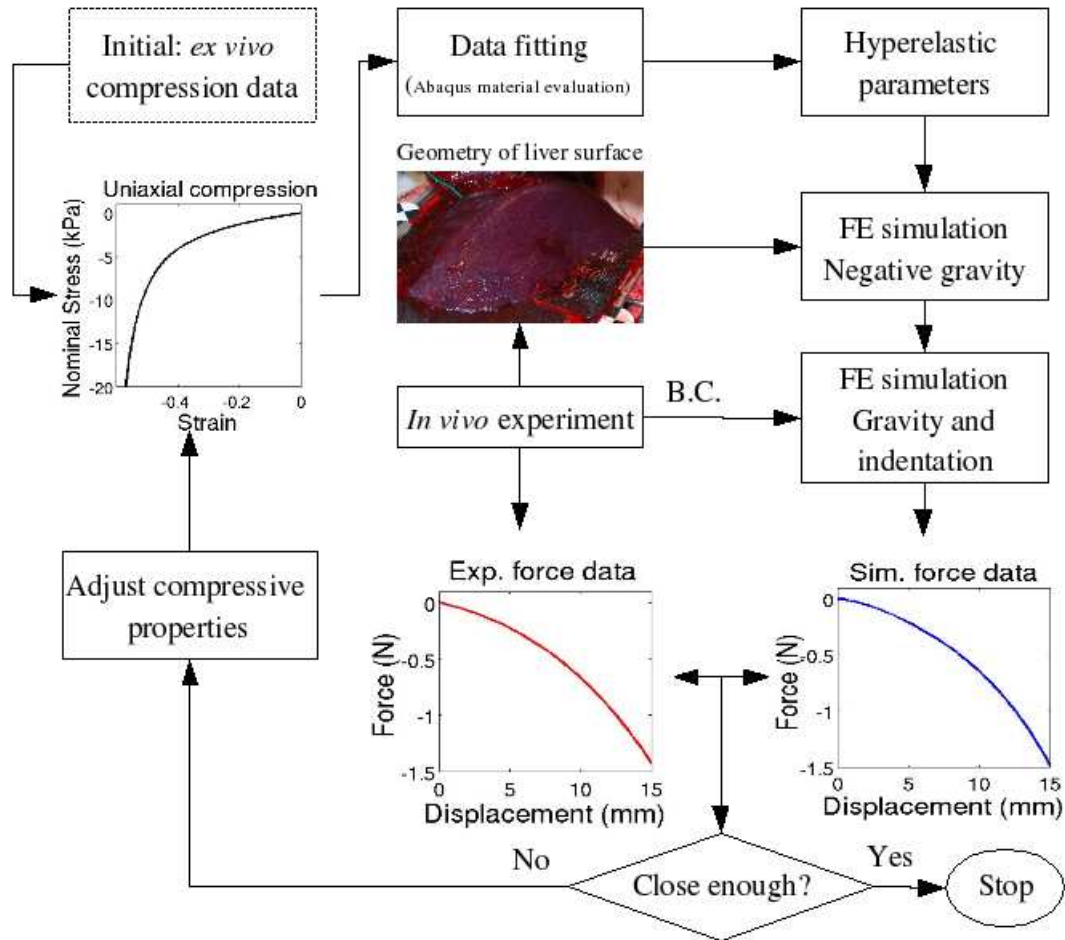


Figure 2.43: Method for modification of constitutive model. ©2011 Springer.

ulation process and the results were compared to the *in vivo* experimental results. Figure 2.44 shows the displacement results for the iterations of the first testing location. A total of four iterations of the modification algorithm were complete and are shown in addition to the original *ex vivo* model from the previous section. As can be seen, the process resulted in a drastic change of the deformation response in the tissue where the surface displacement becomes more localized as the modification process progresses.

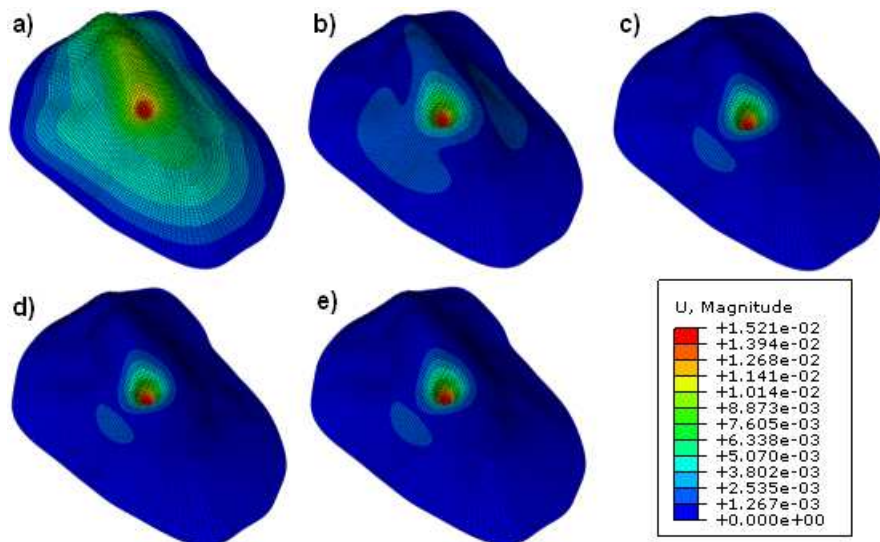


Figure 2.44: Displacement response from FEM simulation of probing task for a) original *ex vivo* model b) first iteration c) second iteration d) third iteration and e) final iteration of material modification procedure.

The resultant reaction force profiles, developed in collaboration with Dr. Zhan Gao, for the four steps in the material model modification process compared to the experimentally measured force response are depicted in Figure 2.45. As can be seen, even after the first iteration there is a drastic change in the resultant force

profile. Despite the fact that it does not entirely match the experimental values, it does follow the proper trend and does not diverge as quickly as the original *ex vivo* based simulation. As the modification process proceeds, the accuracy and trends exhibited in the force response tend to the experimental values. After a total of four iterations, the RMS error calculated from Equation (2.2) is reduced to 0.03N which is well within the sensible range of 0.1N.

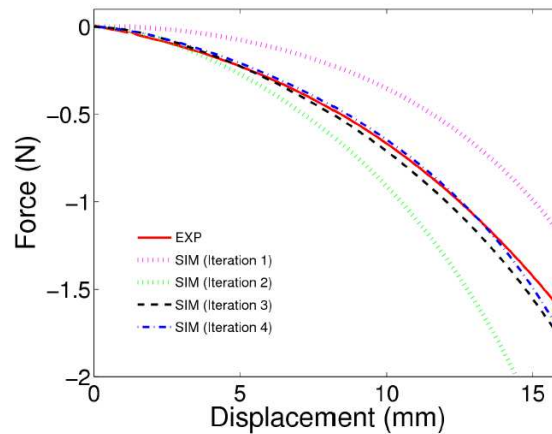


Figure 2.45: Comparison of the experimentally measured force and the simulated force values for the various iterations of the material model modification procedure.

©2011 Springer.

An assessment of the surface fit to the experimental marker displacements was also conducted using Equation (2.3) for each of the iterations in the modification algorithm. After the first iteration, comparison of the simulation and experiment (Figure 2.46) resulted in RMS errors ranging from 0.133mm to 0.541mm, with an average surface fit of 0.360mm. After the second iteration, comparison of the simulation and experiment (Figure 2.47) resulted in RMS errors ranging from 0.027mm

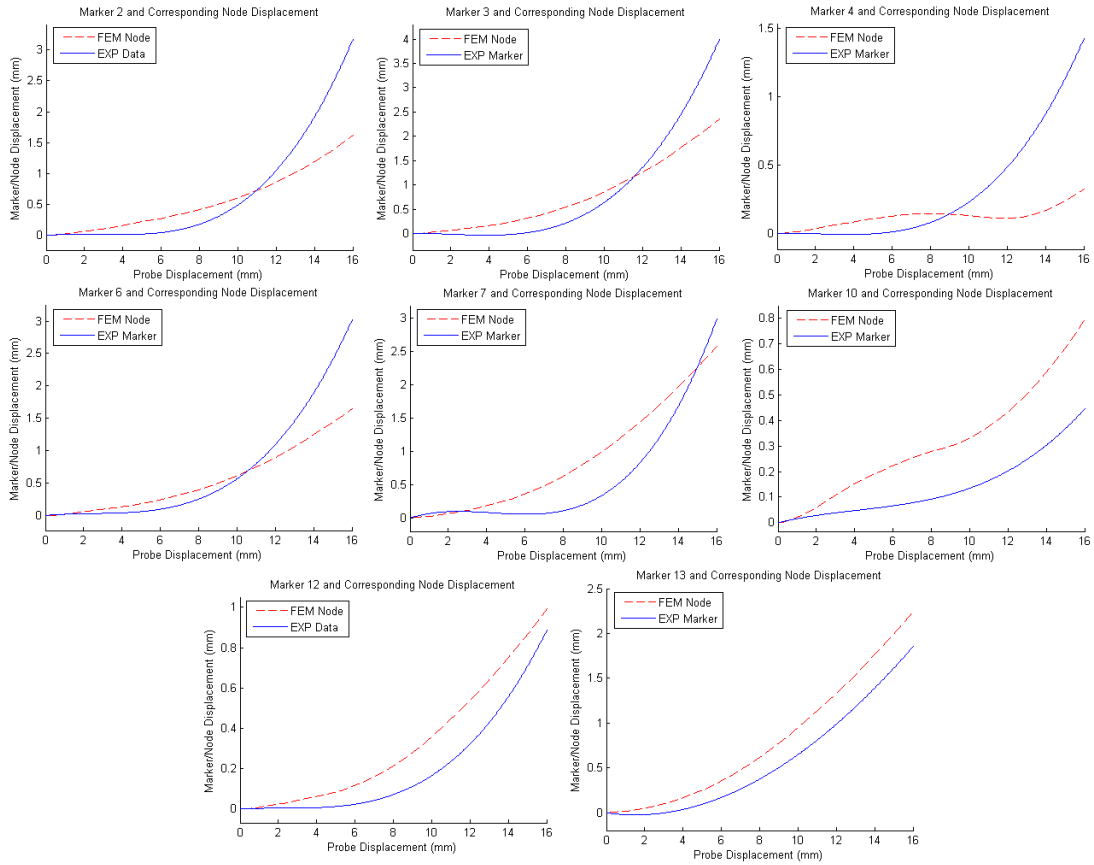


Figure 2.46: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the first iteration of the material model modification procedure.

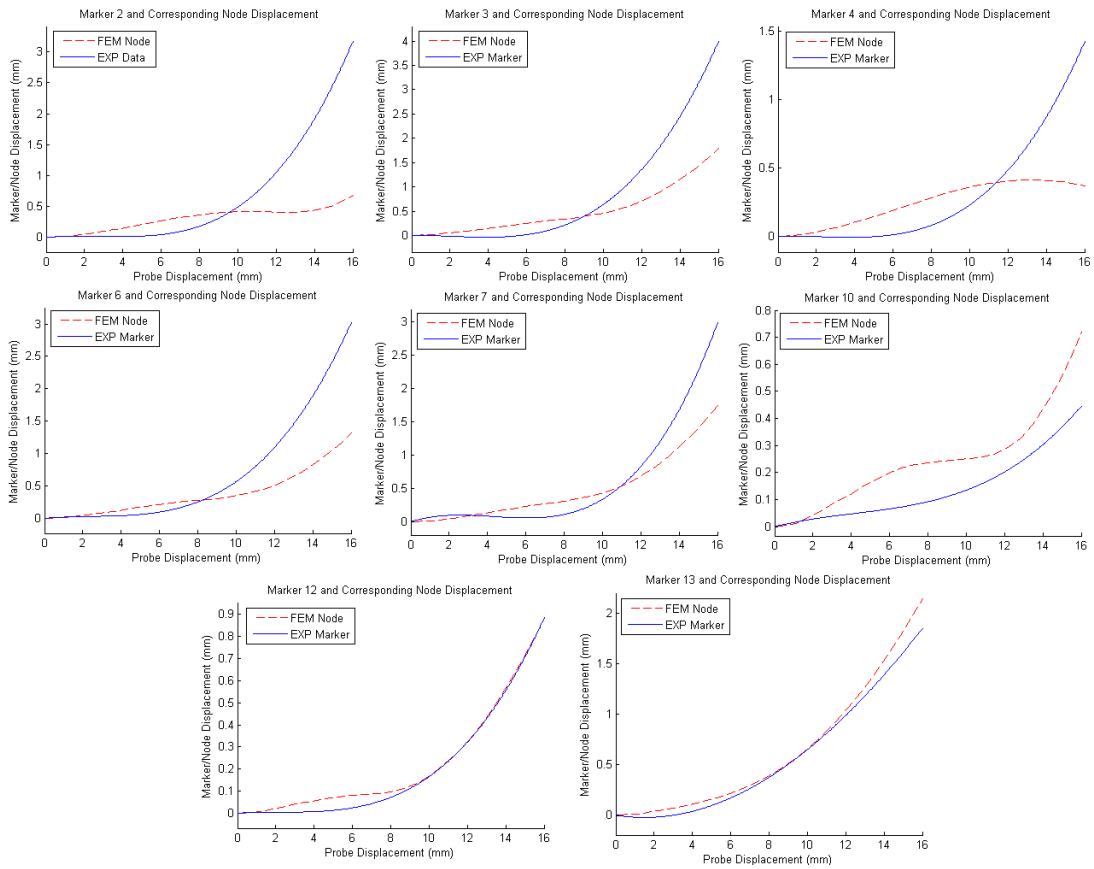


Figure 2.47: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the second iteration of the material model modification procedure.

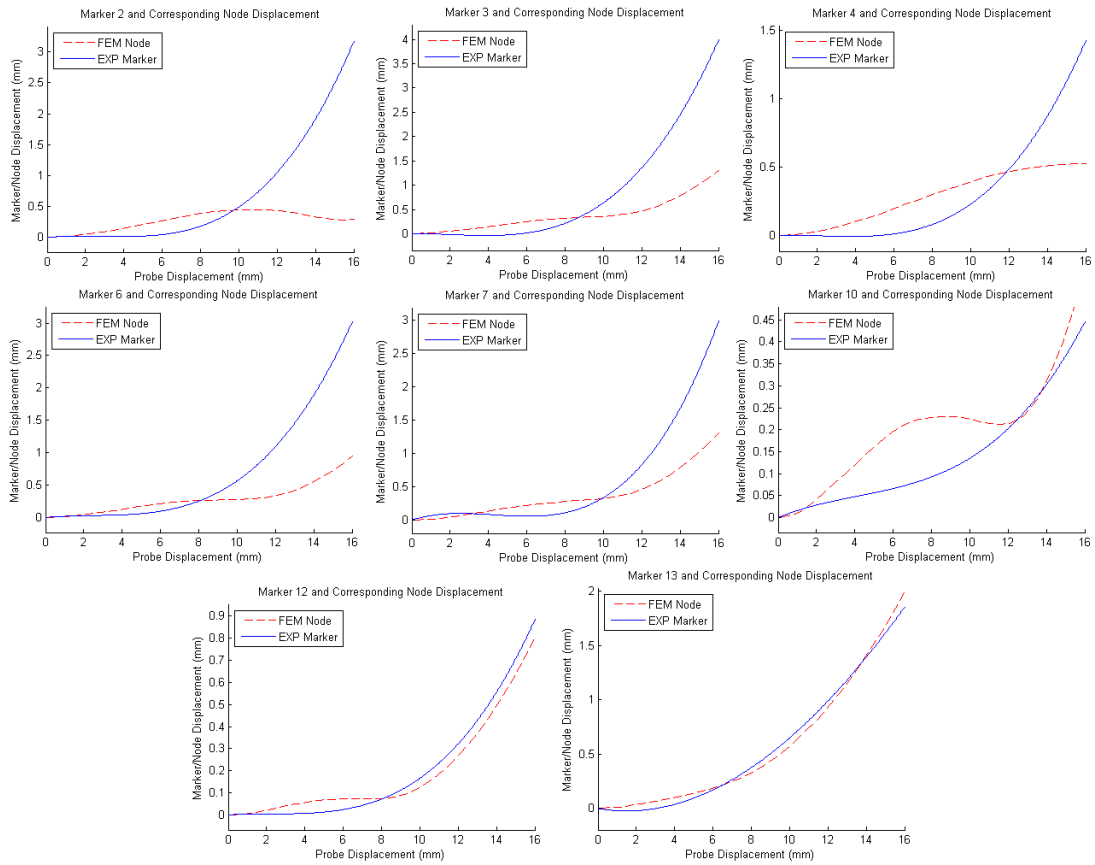


Figure 2.48: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the third iteration of the material model modification procedure.

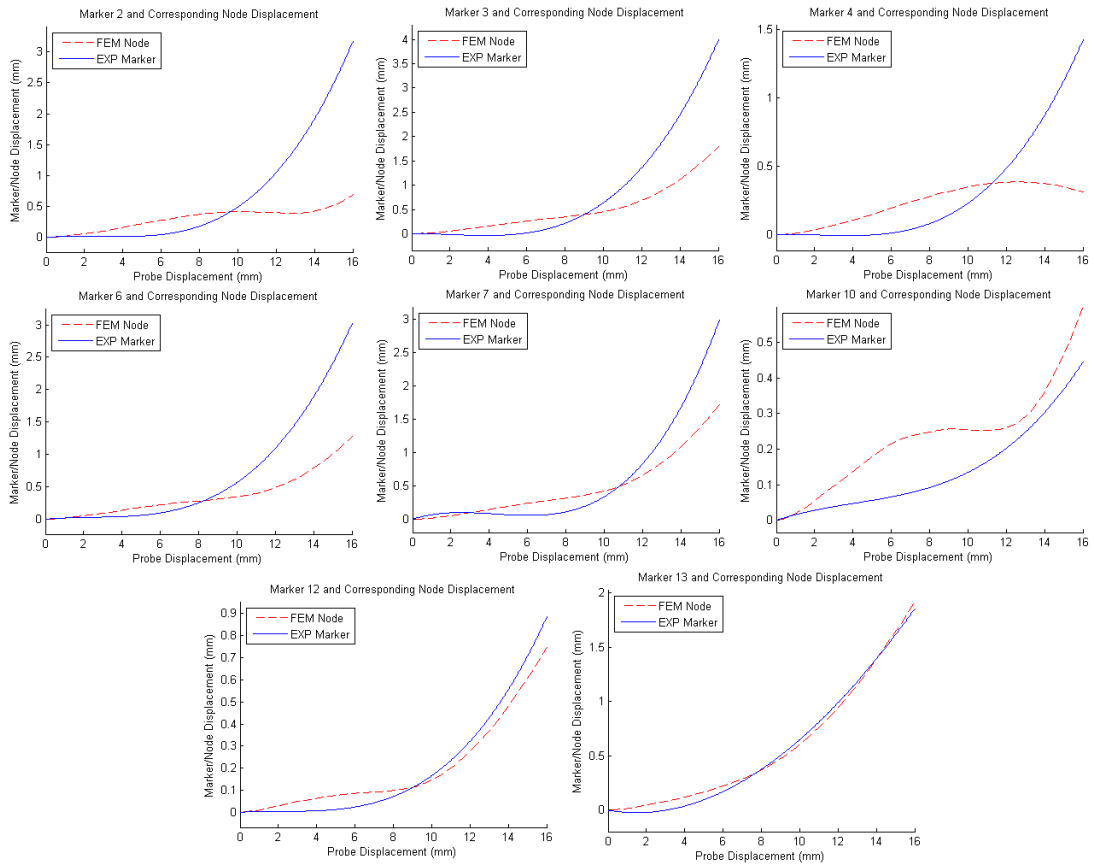


Figure 2.49: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the final iteration of the material model modification procedure.

to 0.890mm, with an average surface fit of 0.415mm. After the third iteration, comparison of the simulation and experiment (Figure 2.48) resulted in RMS errors ranging from 0.043mm to 0.847mm, with an average surface fit of 0.438mm. Finally, the last iteration (Figure 2.49) resulted in RMS errors ranging from 0.046mm to 0.702mm, with an average surface fit of 0.340mm. After the fitting was complete, the accuracy of the surface displacement due to probing only slightly decreased from 0.307mm to 0.340mm, which is comparable to the original *ex vivo* model. However, as previously mentioned, the force response accuracy was greatly improved. The new hyperelastic model, based on the *in vivo* experimental data, produces a much more accurate representation of the probing simulation than obtained using the *ex vivo* material model. The error values obtained for the first probing location are well within the values that would be noticeable to a user and are therefore applicable for use in an *in vivo* medical training simulator.

The material modification process was repeated for the next two probing locations to improve the performance to acceptable levels for *in vivo* simulation. Figure 2.50 shows the displacement results for the final iteration of the material model modification process compared to the original *ex vivo* based model for the two additional probing locations. As can be seen in the plots, the trends were similar to the previous test where the final material properties, after the modification procedure, reduced the magnitude of the surface displacement due to tissue probing.

The reaction force profiles obtained during the modification process for the next two probing locations are presented in Figure 2.51. Once again, the force profiles tend to the proper shape and magnitude over the course of only a few

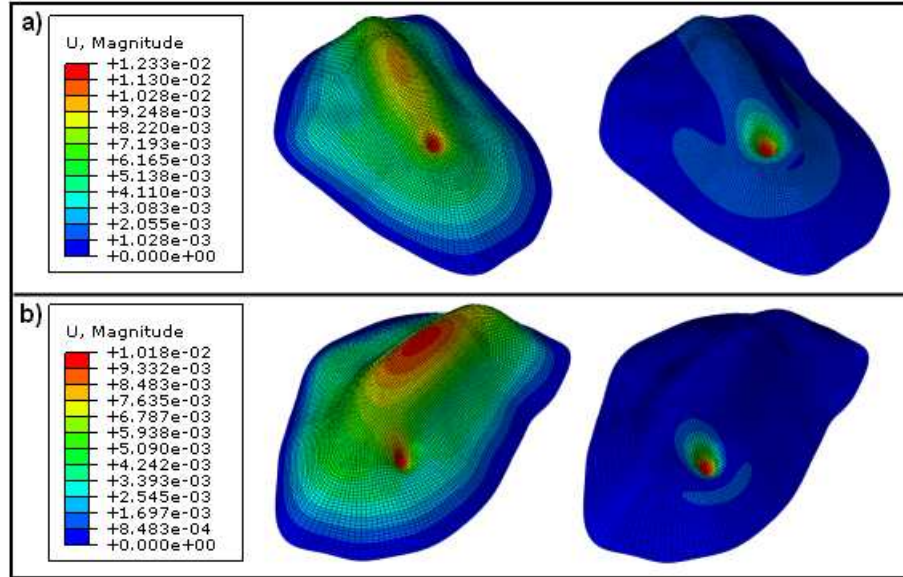


Figure 2.50: Displacement response from FEM simulation of probing task for original *ex vivo* model and final iteration of material modification procedure for a) second probe location and b) third probe location.

iterations. For the second location, the final iteration produced a reaction force response with a RMS error of 0.015N over the depth of the probing action. The third probe location yielded a final RMS value of 0.025N. Both of these values are once again far below the 0.1N threshold.

Using Equation (2.3) to determine the RMS error for the surface marker displacement information for the final iteration of the material model modification procedure at the second and third probe locations shows similar results to the first probe location. The final iteration for the second probing location (Figure 2.52) resulted in RMS errors ranging from 0.099mm to 0.810mm with an average surface fit of 0.402mm. The final iteration for the third probing location (Figure 2.53) resulted in RMS errors ranging from 0.066mm to 0.767mm with an average surface

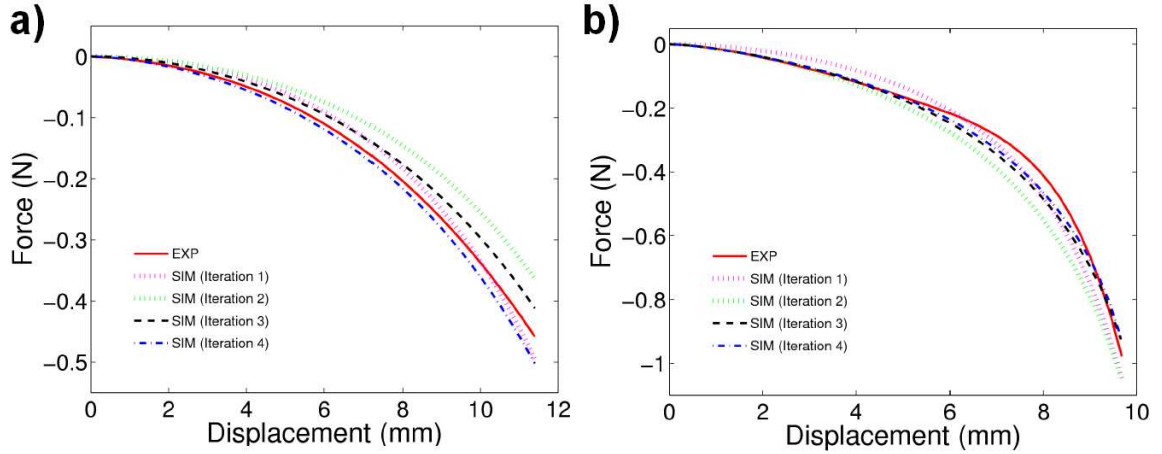


Figure 2.51: Comparison of the experimentally measured force and the simulated force values for the various iterations of the material model modification procedure for the a) second probe location b) third probe location.

fit of 0.242mm. These values reflect a slight increase over the original *ex vivo* based material model, however the difference is not significant and the overall surface displacement errors will be difficult for a user to notice.

A summary of the model accuracy data can be found in Table 2.2. The accuracy of the simulated force response is greatly improved due to the constitutive model modification process. The average errors decreased by an order of magnitude from 0.223N to 0.023N. In addition, while this drastic improvement could be seen on the force side, the induced inaccuracies in the surface displacement measurement only increased slightly from 0.261mm to 0.328mm. This amount of error would be difficult to detect in the context of a surgical simulator. Based on this information, the material models generated during the model modification procedure (detailed parameters given in Table 2.3) are well suited for use in surgical simulation of *in*

vivo based procedures on porcine liver.

Table 2.2: Comparison of *ex vivo* experimental data to the simulation of various constitutive models.

	<i>Ex Vivo</i> Material Model		Modified Material Model	
	Displacement RMS	Force RMS	Displacement RMS	Force RMS
First Probe Location	0.307mm	0.420N	0.340mm	0.030N
Second Probe Location	0.237mm	0.170N	0.402mm	0.015N
Third Probe Location	0.238mm	0.080N	0.242mm	0.025N
Average	0.261mm	0.223N	0.328mm	0.023N

Table 2.3: Constitutive models for general loading of *ex vivo* and *in vivo* porcine liver.

	Model Type	Model Parameters
<i>Ex vivo</i>	Four Term Ogden	$\mu_1 = 7505.68 \frac{N}{m^2}, \alpha_1 = -2.773, \mu_2 = 0.155 \frac{N}{m^2}, \alpha_2 = 19.668$ $\mu_3 = -13856.8 \frac{N}{m^2}, \alpha_3 = -3.644, \mu_4 = 6421.31 \frac{N}{m^2}, \alpha_4 = -4.515$
<i>In vivo</i> 1	Two Term Ogden	$\mu_1 = 1818.67 \frac{N}{m^2}, \alpha_1 = 3.277, \mu_2 = 4.752 \frac{N}{m^2}, \alpha_2 = -10.014$
<i>In vivo</i> 2	Three Term Ogden	$\mu_1 = -22740.362 \frac{N}{m^2}, \alpha_1 = 1.911, \mu_2 = 12578.797 \frac{N}{m^2}, \alpha_2 = 3.181$ $\mu_3 = 10686.170 \frac{N}{m^2}, \alpha_3 = 0.749$
<i>In vivo</i> 3	Reduced Polynomial	$C_{10} = 781.898 \frac{N}{m^2}, C_{20} = -747.631 \frac{N}{m^2}, C_{30} = 426.135 \frac{N}{m^2}$ and $C_{40} = 122.701 \frac{N}{m^2}$

Using the models developed from the *in vivo* experiments, simulations of tool-tissue interactions capable of accurately displaying both the tissue deformation and resultant reaction force were possible. In this study, three distinct models were de-

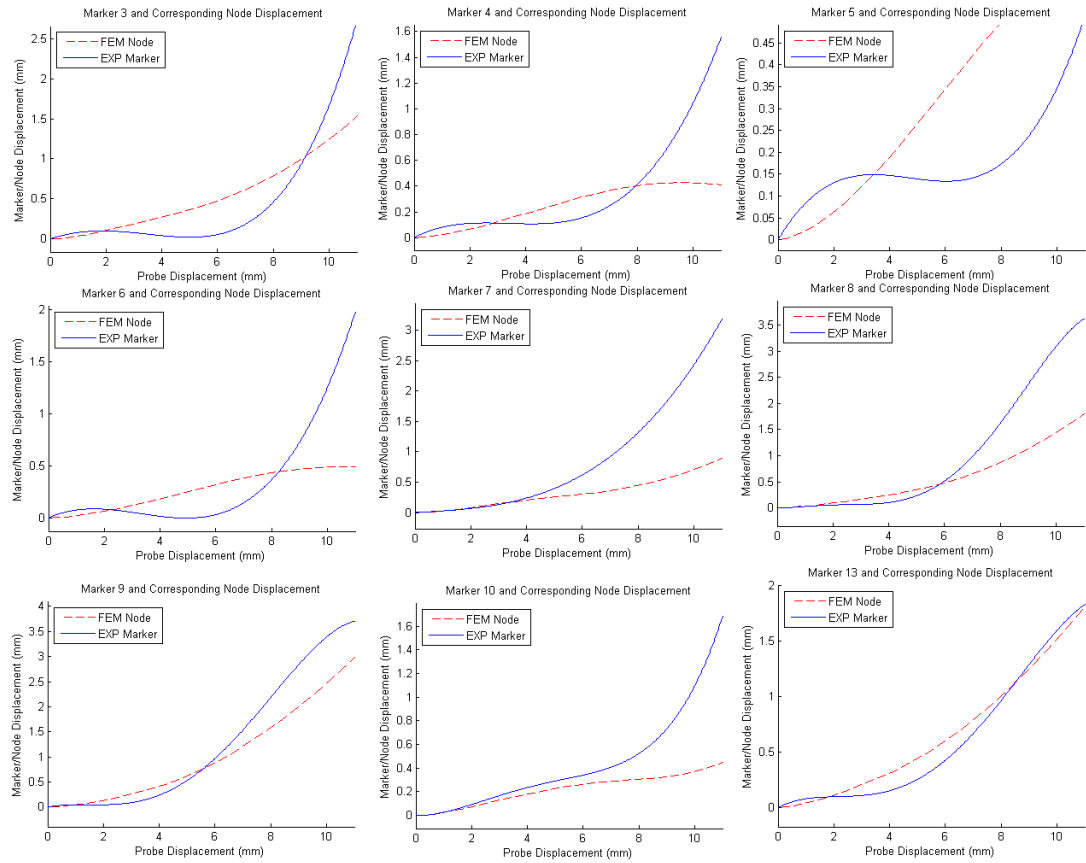


Figure 2.52: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the final iteration of the material model modification procedure at the second probing location.

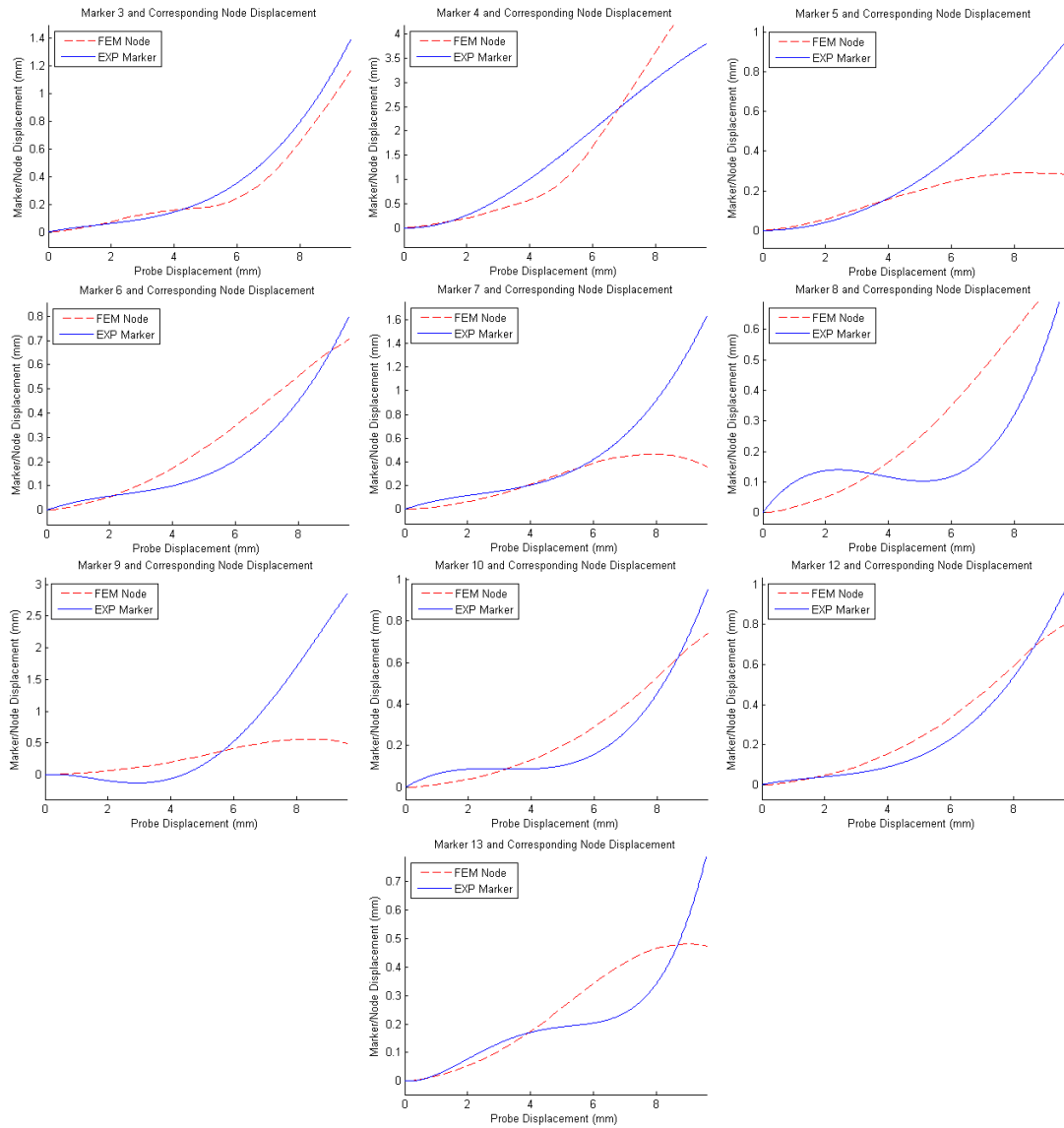


Figure 2.53: Comparison of the experimentally measured marker displacement and the simulated nodal displacement for the final iteration of the material model modification procedure at the third probing location.

veloped which all result in a slightly different response to tissue loading. Many of the previous soft-tissue characterization studies generate one model based on the average tissue response. However, biological tissue is inherently non-uniform in nature resulting in large variations in the tissue response between samples and even within the same organ. Therefore, in this work each test was independently analyzed and distinct models were developed for each probing location. By developing multiple models, all based on the measured tissue response, the variability witnessed in the tissue can be replicated in the simulator by randomly selecting the model at the beginning of the simulation, adding to the overall accuracy of the surgical training.

The models developed in this chapter are well defined to simulate the general deformation caused by the interaction between a surgical tool and the soft tissue. However, additional functionality needs to be added to the simulation to replicate more complex surgical tasks resulting in a physical change to the organ. Surgical techniques such as cutting, dissection, suturing, ablating or cauterizing tissue require additional models to govern the changes to the tissue morphology. The next chapter presents work on developing the additional techniques required to simulate scalpel cutting of soft tissue.

Chapter 3

Cutting Of Porcine Liver

One of the fundamental techniques used in surgical procedures is the act of cutting or dissecting soft-tissue with a scalpel. Providing the ability to virtually practice realistic scalpel cutting procedures will aid in preparing medical students for many common surgical procedures. However to benefit from the surgical training, the simulators must be as realistic as possible, both in the visual and haptic sense. The state of the art consists of simplistic, mesh based models for cutting procedures where the tissue separation occurs by simply removing or spiting elements. These approaches are not physics-based and do not provide the capability to represent the realistic reaction forces felt by the surgeon during the cutting process. To improve upon the current trends, a fracture mechanics approach was implemented to model and simulate soft-tissue cutting which takes into account the physical properties of the tissue and results in a proper graphical and haptic representation of the scalpel cutting process.

3.1 Modeling of Cutting Process

Selecting the appropriate means to model scalpel-tissue interaction was an interesting task to undertake as there were many possibilities to chose from, all with their own benefits and limitations. The general fracture process stems from many

different contributions all working together, from material based aspects such as the crystal structure, grain patterns, and intrinsic material imperfections, to external stimuli such as temperature effects and loading conditions. Being as there are many different types of materials exerting various behaviors under different conditions, many different fracture-based models have been generated over the years. The majority of these standard models, however, assume the use of common material; a category which does not include biological tissues. Therefore, many of the models are limited in their application to the scalpel-tissue interaction due to the inability to measure the required parameters from standard testing procedures like a notched bend test or a compact tension test [22]. Cohesive zone models, on the other hand, provide a proven method for simplifying the fracture process in a way that can be applied to soft-tissue.

3.1.1 Cohesive Zone Approach

The cohesive zone approach simplifies the fracture process into a form that allows for modeling of a crack initiation as well as crack propagation into the material [75]. It is a physics-based approach that utilizes three material specific parameters: the cohesive energy, cohesive strength and cohesive separation distance. From these parameters a traction-separation curve is generated which represents the tractions across the cohesive region from the initial loading through the material degradation to ultimate failure. A schematic of a standard traction-separation curve is displayed in Figure 3.1.

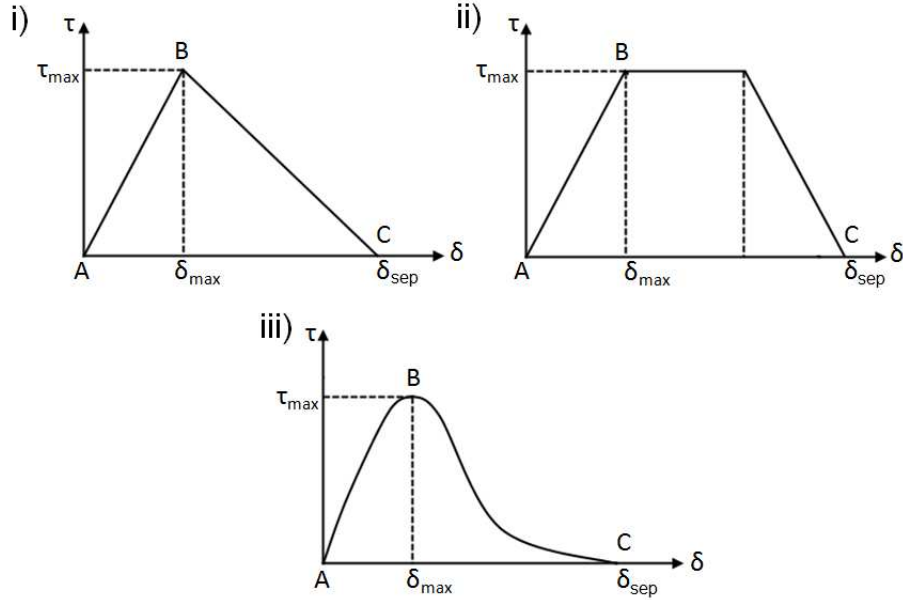


Figure 3.1: Cohesive traction-separation curves for i) bilinear ii) trapezoidal and iii) exponential models. ©2011 IEEE.

As can be seen in the depiction, various types of models have been used to describe the cohesive response by changing the shape of the traction-separation curves. The three most commonly used are the bilinear, trapezoidal and exponential models [75][80][43]. Despite the different shapes, the three models displayed all have similar features which correspond to the three specific parameters required to define the cohesive zone. The cohesive strength, τ_{max} , corresponds to the peak of the traction-separation curve. The cohesive separation distance, δ_{sep} , defines the total length of the curve (physically it is the width of the crack front at any given location) and the cohesive energy correlates to the total area under the curve. Differing views have arisen in the fracture mechanics community pertaining to the importance of the shape of the traction-separation curves with some saying the focus should be

placed on only the magnitude of the three characteristic parameters [12] and others stressing the importance of the curve shape [75]. The degree of the effects appear to be mainly material dependent and need to be investigated on an individual basis.

Physically speaking, the cohesive zone approach expresses all of the contributions that effect the fracture process in the simplified traction-separation characteristics. When a material is loaded, a cohesive zone develops in front of the crack tip. This zone starts at point A, in Figure 3.1, which correlates to the location in the material where loading due to the crack begins to become apparent. This point does not correspond with the location of the physical crack in the specimen but is located ahead of that point where the effect of loading begins, as can be seen in Figure 3.2. In the region from the physical crack tip, point C, to the mathematical crack tip, point A, the traction-separation curve defines the loading of the material. Point B is located between the mathematical and physical crack tips and represents the location of maximum stress within the cohesive region. From point A to point B, the sample is being loaded, at point B, the sample reaches it critical stress point and starts to fail. In the region from point B to point C the sample degrades until it reaches the point of complete physical separation at point C [75].

By using this approach it is possible to define the initiation and propagation of the crack front by studying the loading properties in the region directly in front of the crack tip. Once the material is loaded to a certain extent and the damage region completes, the crack physically opens in the material. Implementation through the use of the finite element method is direct as the parameters required for cohesive zone calculations (stress, strain and energy) are directly available in the existing

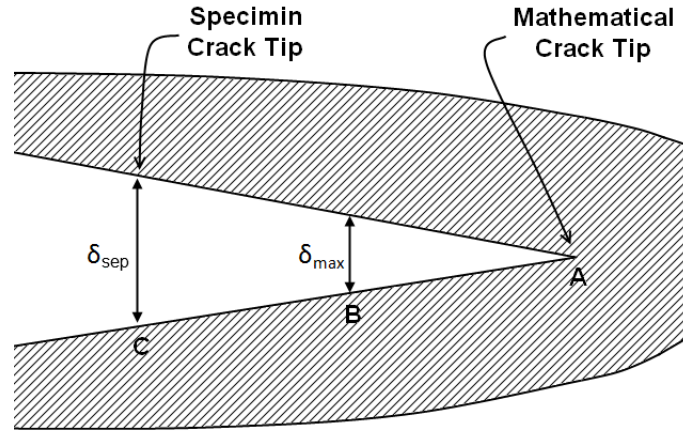


Figure 3.2: Cohesive zone region. ©2011 IEEE.

analysis [24]. By comparing the loading conditions in the finite element analysis to those of the defined cohesive zone, it is possible to determine the degree of damage within the material. At the critical level of damage, as defined by the cohesive mode, the material is known to be degraded to the extent that separation exists. This is then reflected in the simulation by physically separating the mesh itself, resulting in the formation or propagation of the crack.

One major limitation of the approach is this assumption that the crack direction must be known, as the cohesive zone is defined in along this line [21]. This issue is not detrimental to the scalpel loading problem because the direction of the crack is assumed to be in the direction of the applied load. This assumption will hold in the majority of cutting procedures on normal tissue barring any complications due to large defects in the tissue.

The cohesive zone approach was selected as the preferred model for the scalpel cutting process due to its ability to model the physical aspects of the fracture process in a simple yet realistic manner. It has been used effectively in the simulation of crack

growth in many different types of materials (metals, ceramic, polymers, etc.) and it naturally lends itself to implementation directly in a finite element formulation. By using the cohesive zone approach, an accurate representation of the tissue separation and resultant cutting force profile could be simulated. After selecting the cohesive zone method as the proper model to utilize, experiments were conducted on *in vivo* porcine liver to determine the cohesive parameters required for implementation of the model in a FEA simulation [47].

3.1.2 *In Vivo* Experiments

The experimental procedure for the collection of cutting data required for derivation of the cohesive zone parameters was similar to the previously discussed *in vivo* probing experiments. The test fixture developed for the *in vivo* experiments was used for the cutting analysis by rotating the linear rails to the horizontal position, as can be seen in Figure 3.3. An adapter was used to mount the force sensor and the scalpel blade in the vertical orientation conducive to the cutting motion. The cutting experiment was performed on the same 20.5kg pig as the second *in vivo* probing test. Following the completion of the probing experiments, a new lobe of liver, that had not been previously used for testing, was placed on the support plate and used for the cutting task.

Previous cutting experiments have shown that many variables affect the cutting process including the angle of the scalpel relative to the tissue, the depth of the scalpel, the speed of the cutting process as well as the sharpness of the scalpel

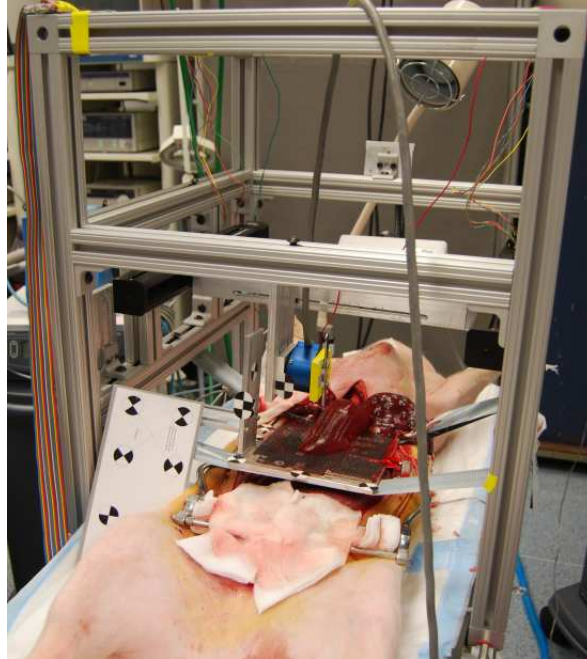


Figure 3.3: Experimental setup for the *in vivo* cutting test.

blade [11]. In order to reduce the effects caused by some of these conditions the test was conducted with the scalpel blade fixed in the vertical position at a constant speed of 1.25mm/s. A new scalpel blade was used during the test to avoid any differences that might arise from a dull cutting edge. Finally, to account for the depth of the blade, the tissue support system was oriented such that the tissue profile was as flat as possible. There were some changes in the scalpel depth that were unavoidable during the cutting tests, however, these changes in depth were accounted for during the analysis process.

After the system was configured to minimize the scalpel variables, the MicronTracker camera system was set up to record the cutting process. To obtain an accurate representation of the scalpel depth throughout the experiment, the shape of the liver was required. The procedure described in Section 2.2.2 was used once again

to record the point cloud representation of the organ geometry required to construct the three dimensional model. As before, the camera needed to be mounted outside the frame to acquire the surface information. Upon completion it was moved to the proper location to record the cutting task. Reference marker locations were measured to allow the transformation of data between the two reference frames. Finally, to record the tissue displacement information, markers were placed on the surface of the liver. The markers used for the cutting task were much smaller in an attempt to measure the local deformation around the scalpel blade to an extent that could be used to determine the cohesive separation distance. The markers were placed on the surface in two parallel lines encapsulating the scalpel blade (Figure 3.4). The tracking procedure developed for the *in vivo* probing tests was used without modification to determine the individual marker displacement for the cutting task. A marker was placed on the force sensor was used to track the scalpel displacement throughout the test.

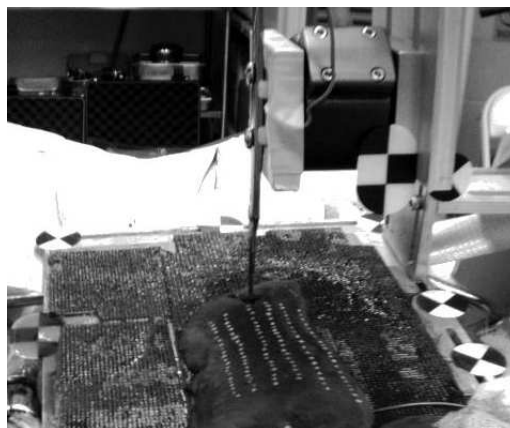


Figure 3.4: Marker placement for cutting test.

The goal of the cutting task was to extract the cohesive parameters required to

properly characterize the cohesive zone approach. Of the three parameters discussed, only two are required to fully characterize the cohesive zone. This allowed for some error in the experimental procedure, however the goal was to measure all three parameters directly. The cohesive separation distance was designed to be derived from the surface marker displacement. To obtain the cohesive stress and cohesive energy values, the force profile was recorded throughout the cutting task. The experimental force profile was recorded via the six axis JR3 force sensor. The total magnitude of the force imparted to the scalpel blade for one cutting location is depicted in Figure 3.5. The force profile was then used to determine the cohesive strength and the cohesive energy.

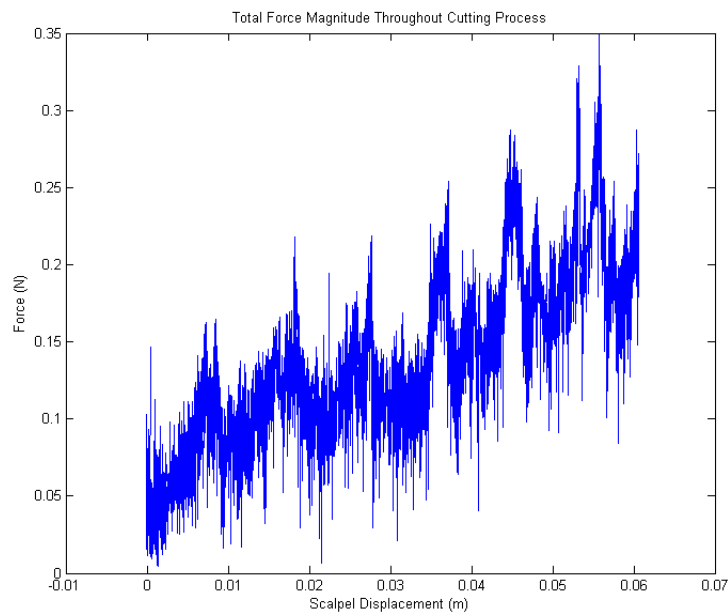


Figure 3.5: Magnitude of total cutting force throughout the experiment.

3.1.3 Generation of Cutting Model Parameters

After completion of the *in vivo* cutting tests, analysis of the collected data was used to derive the parameters required for the definition of the cohesive zone model. As was mentioned, only two of the three values defining the traction-separation curve are required for the complete statement of the model. Based on the data collected, it was apparent that the most accurate parameters to determine were the cohesive strength and the cohesive energy. The cohesive separation distance is defined as the critical width of the crack at the instant of complete failure. Despite the effort to measure this parameter with the smaller markers, the tissue exhibited a large amount of deformation due to the scalpel loading procedure. This deformation was a combination of the crack width, elastic deformation due to loading and motion due to settling processes influenced by the soft nature of the material under gravitational loading after the tissue fracture. By the nature of the scale of the tissue deformation, the crack width at the time of fracture was the smallest contributor, and the overall deformation was dominated by the other modes. These additional deformations made the measurement of the cohesive separation distance infeasible with the current mode of data collection. The cohesive strength and cohesive energy, however, could be derived from the measured force profile and tissue and scalpel geometries.

The approach utilized to determine the cohesive strength for the scalpel-tissue interaction was derived to solve for the local loading along the scalpel profile. The cohesive strength is a representation of the maximum normal and tangential tractions that occur along the scalpel blade during the cutting process. Due to the

non-linear nature of the scalpel geometry, the loading is different along the length of the blade, resulting in a variation in the magnitude of the tractions. Ideally, individual force sensors would be placed along the scalpel blade profile to provide a continuous representation of the force acting on the cutting region as it occurs during the actual cutting process. This would provide the ability to directly measure the maximum tractions along the blade. However, this approach is not feasible in the experimental realm so a method was developed to segment the measured cutting force profile into individual tractions acting along the blade. To do so, an assumption was made whereby the measured force is said to act evenly along the length of the scalpel blade in contact with the tissue. To mathematically distribute the measured forces properly, a few key factors need to be determined, namely, the depth of the blade at each fracture location, and a detailed representation of the scalpel geometry.

When speaking of the fracture process, it should be noted that simulation is desired to be as accurate as possible. During the scalpel cutting process, a distinct cutting pattern is observed where the force profile resembles a sawtooth pattern as can be seen in Figure 3.5. This pattern represents a series of local fracture processes whereby the tissue is loaded to a critical value at which point the actual fracture takes place. After the fracture propagation, which is nearly instantaneous, the surrounding tissue that has not been ruptured begins to load again until another critical point is reached. In this manner, the scalpel cutting procedure can be broken down into a series of individual local fracture processes. Combined, they form the sawtooth type force profile presented here and in other literature. The

goal of this study is to model the fracture process at the resolution of the individual loading segments to reproduce an accurate representation of the cutting force and resultant tissue separation. To do so, the parameters of the cohesive zone model are derived from each of the individual peaks observed in the fracture process. Therefore, one complete scalpel test measured during the *in vivo* experiment actually yields a significant number of fractures, providing a good data set to observe the variability in the cutting process. Throughout this study, one complete scalpel cut will be analyzed resulting in data for eighteen fracture segments.

The first step in the determination of the cohesive strength is to split the entire force profile into individual loading segments. The six axis force sensor used during the experiment recorded the force data in three directions: the direction of the cutting motion (defined as the x axis), along the vertical direction of the blade (defined as the z axis), and the direction perpendicular to the blade profile (defined as the y axis). In this case, due to the scalpel geometry, the dominant recorded forces are in the x and z directions. The measured forces in the y direction were more than an order of magnitude lower, as is expected with the symmetric geometry. With the goal being the derivation of the normal and tangential tractions, the force profiles are analyzed separately at the beginning of the analysis and will be used to solve for the required force values in the normal and tangential directions in the future. Therefore, both the force in the x direction and the force in the z direction are segmented into individual cutting segments and can be seen in Figure 3.6.

To perform the segmentation of the cutting forces, the profiles were first passed through a low-pass Butterworth filter to reduce the noise. The recorded raw data

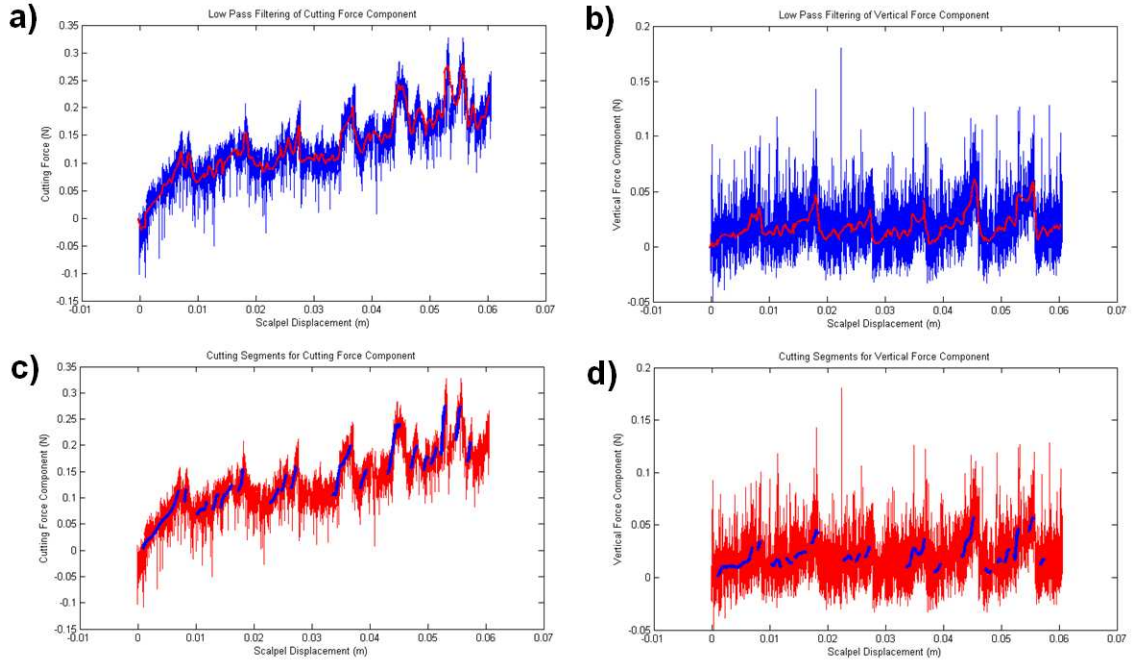


Figure 3.6: Filtering of experimental force data in the a) x and b) z directions. Polynomial fits to represent the eighteen individual cut segments for the c) x and d) z directions.

and resultant filtered force profiles can be seen in Figure 3.6a and b for the force in the x direction and z direction respectively. From this data, the individual cutting segments could be determined. Throughout the entirety of this cutting process, the force in the cutting, or x direction, is the dominate force. Therefore, it was used to determine the extents of the individual fracture segments. By studying the filtered force profile, eighteen segment of significant loading were determined as the main fracture segments. These segments were fit with a polynomial curve as a continuous mathematical representation made analysis later in the process more straightforward. The polynomial representations of the cutting segments in the x direction can be seen superimposed on the original measured force profile in

Figure 3.6c. From these cutting segments it was possible to determine the force acting in the cutting direction on the blade. Using the start and end points of the fracture process for each of the eighteen cut segments, polynomial fits to represent the forces in the vertical or z direction were derived for each segment, as can be seen in Figure 3.6d. As the end goal is to determine the critical strength, or maximum normal and tangential tractions, the critical force values were determined for each cut segment in both the x and z directions for each of the cutting segments at the time of fracture (Figure 3.7).

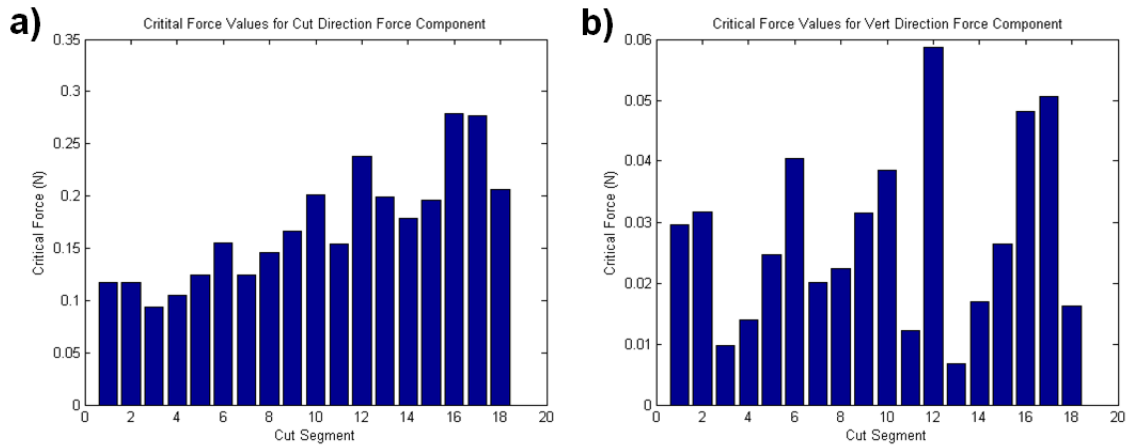


Figure 3.7: Critical force values for each of the eighteen cut segments in the a) x and b) z directions.

Next, to generate the tractions along the length of the blade, a representation of the profile is required. Using an image based approach to measure the shape of the scalpel profile, a series of discrete data points along the scalpel edge were determined. These points were fit with a curve to accurately represent the scalpel shape. The shape of the scalpel blade used during the *in vivo* cutting tests is shown in Figure 3.8

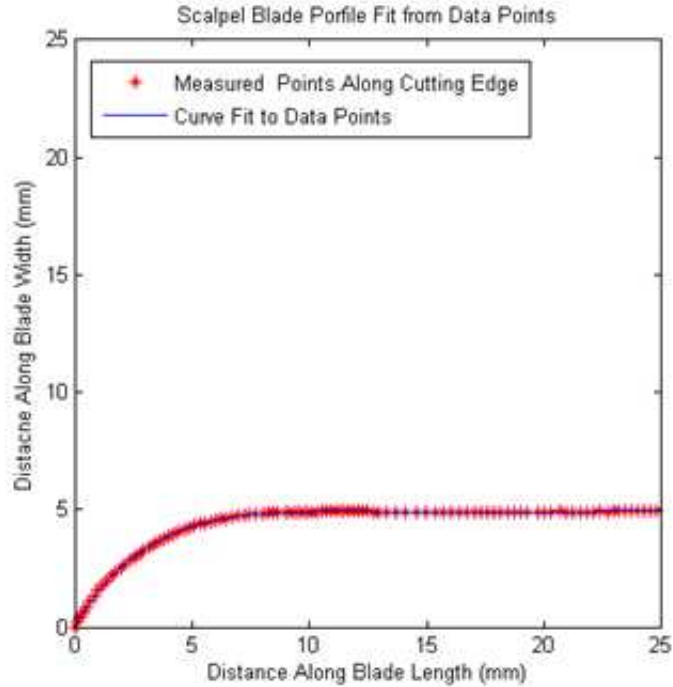


Figure 3.8: Measured scalpel profile with corresponding polynomial fit.

Finally, the depth of the blade at each of the cutting segments is required to define the contact region upon which the measured forces were distributed. To determine the scalpel depth, information was required about the scalpel location and the tissue geometry. Using the methodology presented in Section 2.2.2, a three dimensional representation of the organ surface was generated. In addition, the location of the markers placed on the surface of the liver were also determined. Using this information, it was possible to estimate the height of the tissue sample along the cutting path. To simplify the analysis, all data was transformed into a reference system with the bottom of the tissue geometry representing the x-y plane and the z direction pointing upward. This provided the ability to determine the tissue height along the cutting path directly from the z coordinates.

The second piece of information required to determine the depth was derived from the scalpel location. A transformation related the location of the tip of the scalpel blade to the reference marker fixed to the force sensor during the experiment. The tip location was converted into the new reference system; therefore, the value of the z coordinate at any time would give the distance from the x - y plane, or bottom of tissue sample, to the tip of the blade. The blade depth for each cutting location was determined by subtracting the distance from the bottom of the tissue sample to the tip of the blade from the height of the tissue sample at the location of maximum loading for each of the eighteen cut segments. This depth value corresponds to the vertical depth of the blade, or the length in the z direction and will hence forth be referred to as L_z . The length of the blade in the x direction, or the width of section in contact with the tissue was also required, defined as L_x

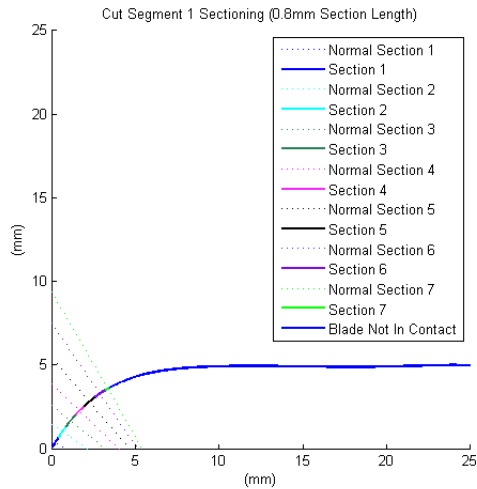


Figure 3.9: Sectioning of the scalpel blade for the first cut segment.

After the depth of the blade is determined for each of the cut segments, the contact length is segmented into subsections such that the force can be distributed

along the blade. This method provides the means to define the loading of the blade in a more realistic fashion. An example of the output from the sectioning procedure is depicted in Figure 3.9, where the first cut segment depth information is used. This approach segments the scalpel tissue contact length into six equal 0.8mm sections with the residual length representing the final section. During the process, the direction of the normal to the center of each section is also recorded. Finally, the length of each section is projected to the depth, or z , direction and the width, or x , direction. As mentioned previously, an assumption was made at this point to evenly distribute the force along the contact length. Using the projection of each section length, the forces in the cutting and vertical direction are distributed along the sections. This method provides the formulation to account for the scalpel shape, resulting in more of the load in the cutting direction being placed on section 6 than section 1 because section 6 has a larger projected length in the z direction. Therefore, the components of the force acting on each of the sections becomes:

$$F_c = (ProjectedSectionDepth) \times \frac{F_{c_{crit}}}{L_z} \quad (3.1)$$

and

$$F_v = (ProjectedSectionWidth) \times \frac{F_{v_{crit}}}{L_x} \quad (3.2)$$

Using this information, it is possible to determine the normal force and tangential force acting on each of the sections of the scalpel blade. Figure 3.10 displays a free body diagram of the force distribution of an individual section. Using this

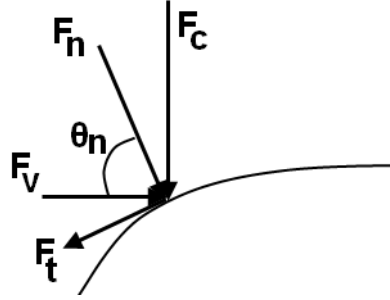


Figure 3.10: Free body diagram of force distribution on scalpel blade. ©2011 IEEE.

information for the direction of the section normal, θ_n , F_c , and F_v the magnitude of the critical normal and tangential forces can be defined as:

$$F_n = F_c \cos(90 - \theta_n) + F_v \cos(\theta_n) \quad (3.3)$$

and

$$F_t = F_c \cos(\theta_n) - F_v \cos(90 - \theta_n) \quad (3.4)$$

This approach allows for the magnitude of the critical forces to be determined for each section along the blade. Figures 3.11, 3.12 and 3.13 show the critical values depicted numerically and graphically for each of the cutting sections of the first cut segment. In this study, three different analyses were conducted to determine the effect of the length of the cut section. Therefore, the data presented represents sectioning of 0.2mm, 0.8mm and 2.0mm.

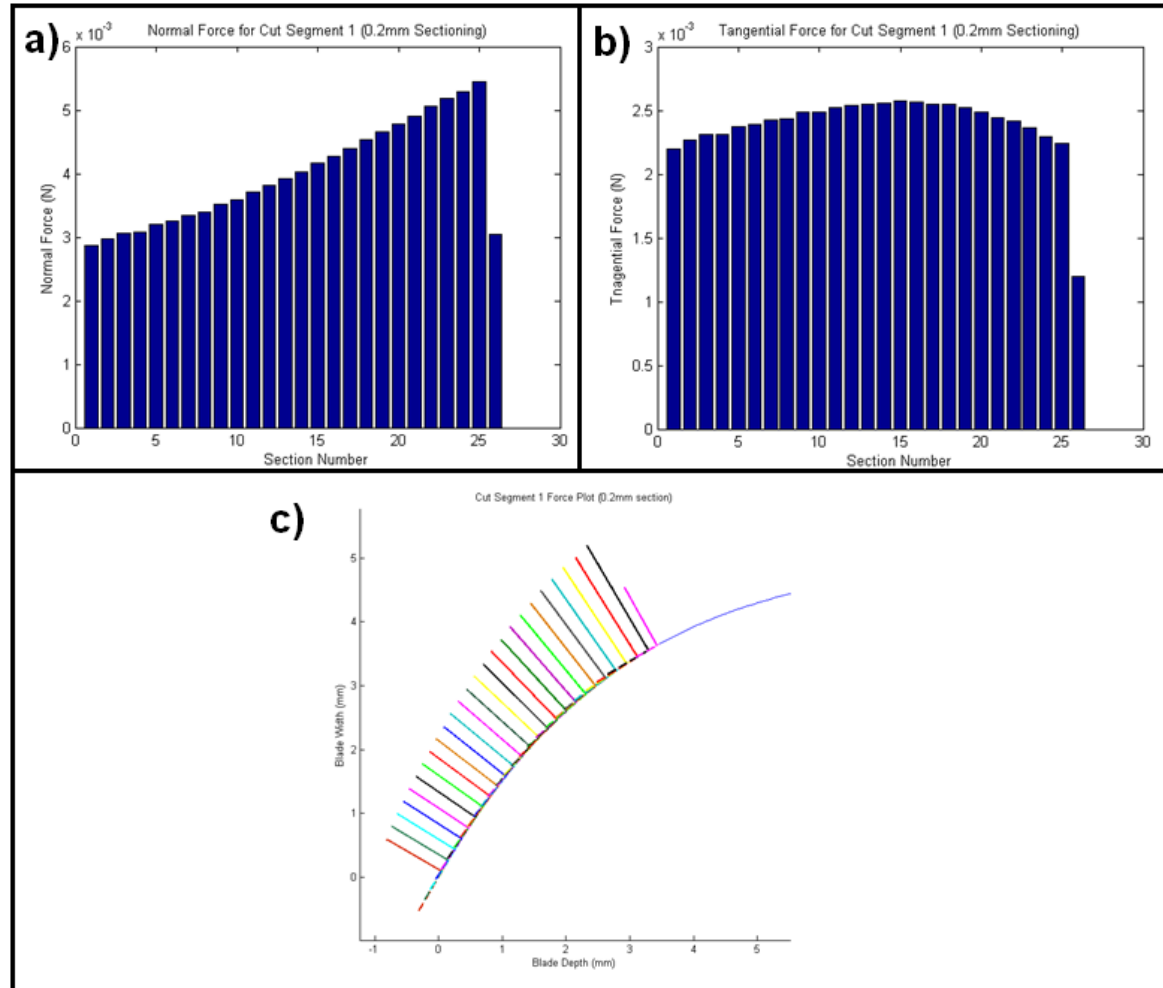


Figure 3.11: Critical force values in the a) normal and b) tangential directions of each cut section using a 0.2mm section length.

c) Graphical representation of the normal and tangential force values.

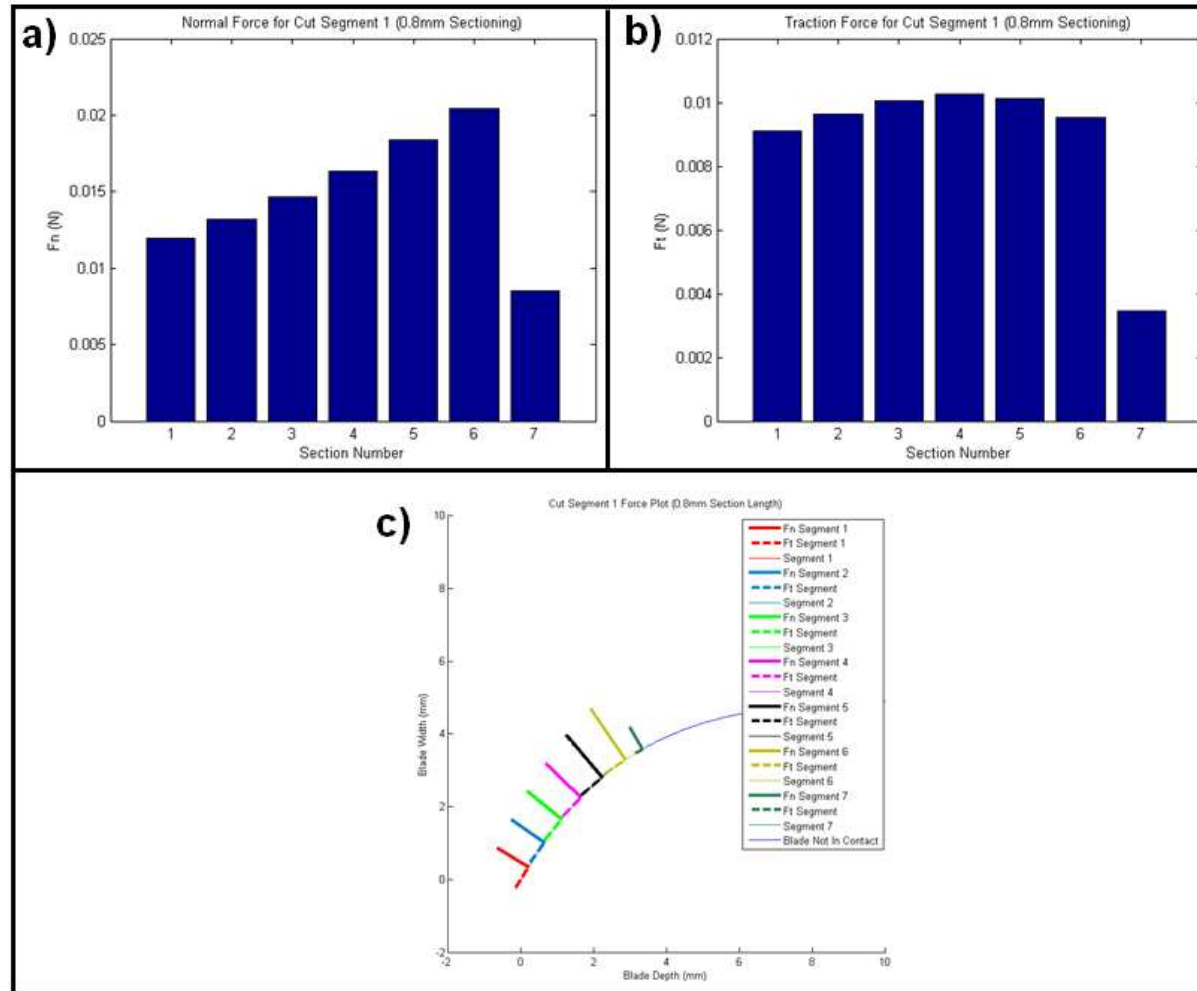


Figure 3.12: Critical force values in the a) normal and b) tangential directions of each cut section using a 0.8mm section length.

c) Graphical representation of the normal and tangential force values.

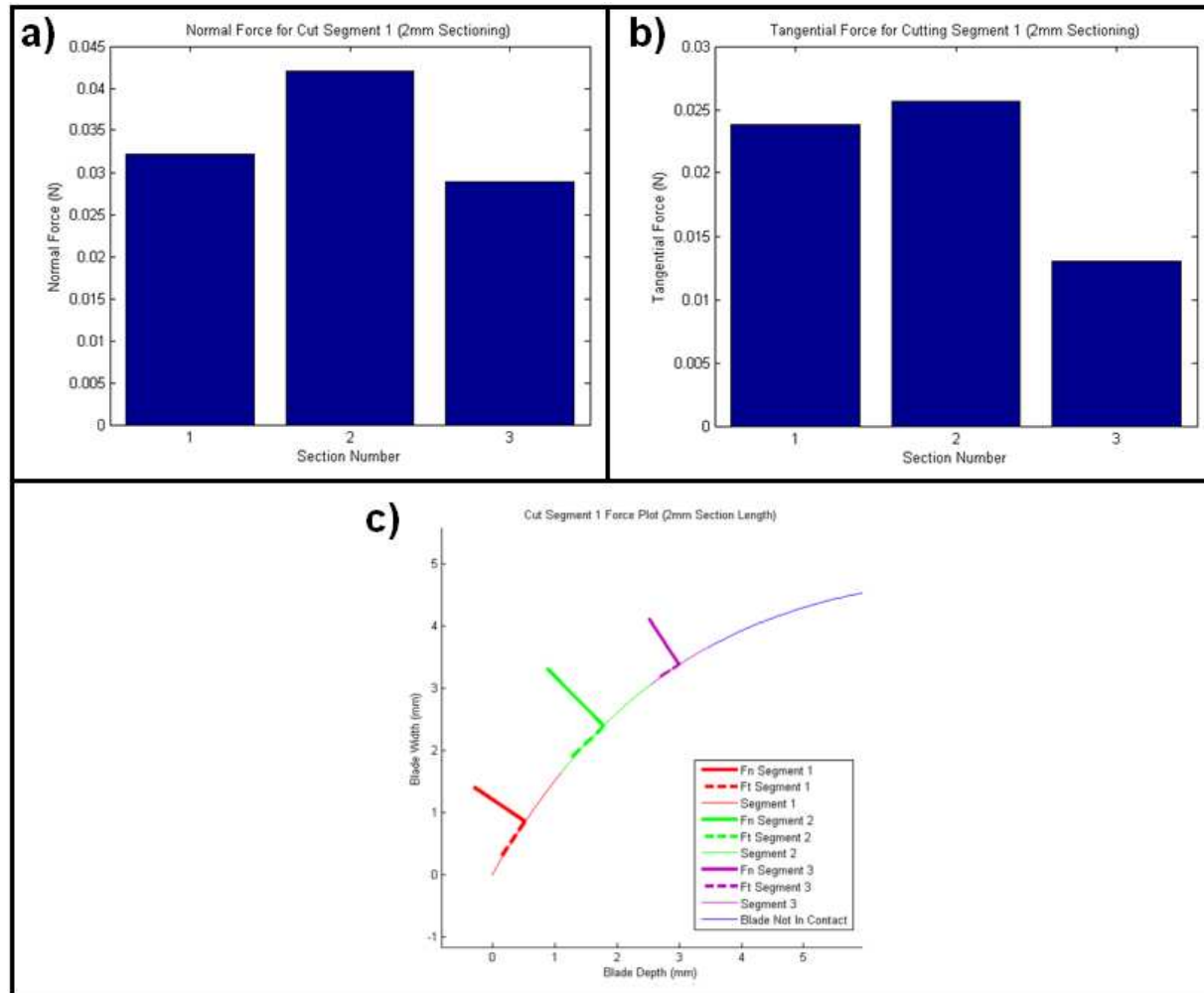


Figure 3.13: Critical force values in the a) normal and b) tangential directions of each cut section using a 2.0mm section length.

c) Graphical representation of the normal and tangential force values.

Next, the critical force values in the normal and tangential direction need to be converted to the critical normal and tangential tractions, as required by the cohesive zone modeling process. To do so, information about the cross-sectional area of the scalpel was required. Using an optical microscope setup capable of measuring distances from the acquired images, the cross-section of the scalpel blade was measured for a total of nine different scalpel blades. An image of the cross-section can be seen in Figure 3.14. The dimensions measured were the angle, the blade thickness and the length of each edge. Only the blade thickness was required for the stress analysis, however the other dimensions will be used in the future sections. Table 3.1 displays the values for each of the measurements. The average thickness of the scalpel blade was 0.365mm, so this value was used to calculate the cross sectional area required to transform the critical force values to the corresponding tractions.

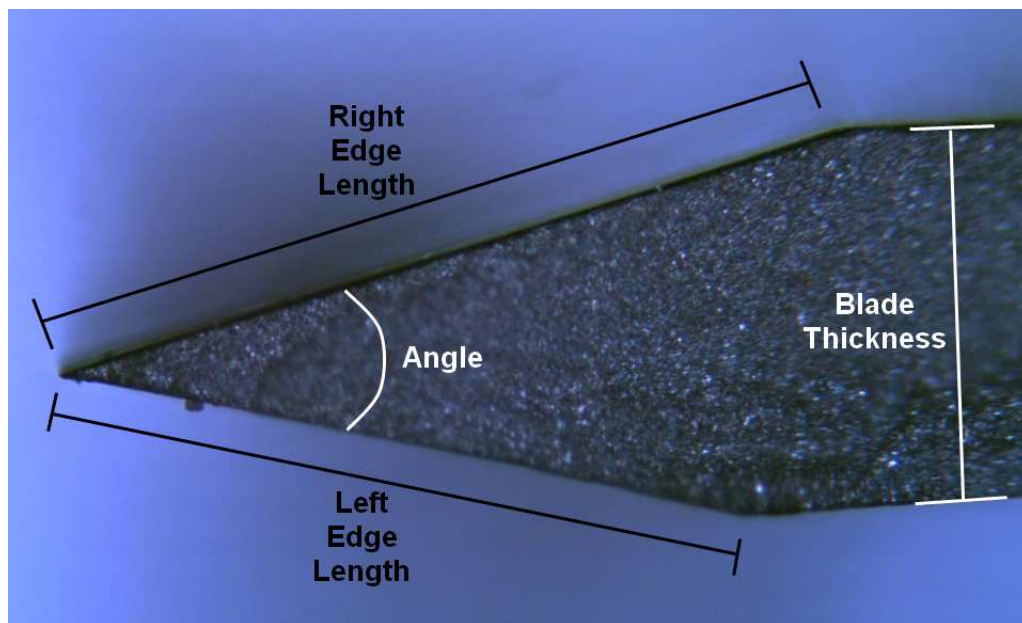


Figure 3.14: Cross section of the scalpel blade.

Table 3.1: Measurement of scalpel cross section data.

Scalpel Number	Angle (Degrees)	Blade Width (mm)	Left Edge Length (mm)	Right Edge Length (mm)
1	29.71	0.369	0.794	0.643
2	29.23	0.363	0.705	0.737
3	28.03	0.359	0.742	0.748
4	27.60	0.358	0.843	0.666
5	29.55	0.376	0.642	0.850
6	26.88	0.357	0.561	0.987
7	28.92	0.366	0.521	0.952
8	27.04	0.364	0.737	0.813
9	29.04	0.370	0.698	0.810
Average	28.44	0.365	0.693	0.800

To determine the tractions acting on each segment the following equations are used:

$$\tau_n = \frac{F_n}{SectionLength \times BladeThickness} \quad (3.5)$$

and

$$\tau_t = \frac{F_t}{SectionLength \times BladeThickness} \quad (3.6)$$

Using the force values calculated for each of the sections, the resultant normal and tangential tractions were determined for each segment of the eighteen cut segments. Figures 3.15 through 3.17 depict the magnitude of the traction distribution across the contact area of the scalpel blade for the first cut segment. Similar trends

were apparent in the other eighteen cut segments.

For the cohesive parameters, the largest normal and tangential traction values constitute the cohesive strength parameter. For each of the eighteen individual cut segments, the maximum tractions were determined for the sections of 0.2mm, 0.8mm and 2.0mm. The values extracted using the three different sections were similar in magnitude, however the values seem to begin to converge as the section size decreases. By splitting the cutting data into the eighteen segments and analyzing each of them individually a range for the critical strength value can be determined which is based on a realistic, *in vivo* cutting experiment. The final cohesive strength values for each of the eighteen cut segments for section lengths of 0.2mm, 0.8mm and 2.0mm are presented in Figures 3.18 through 3.20.

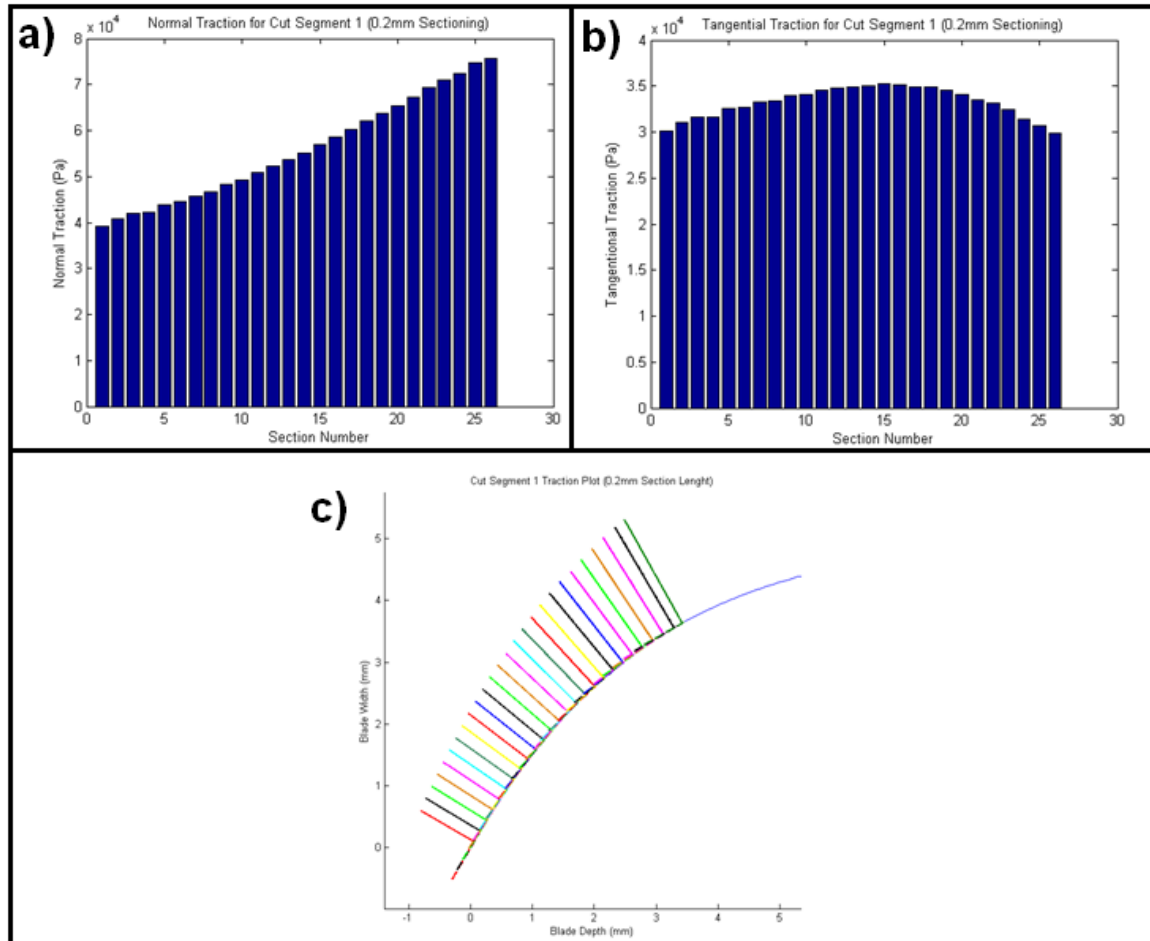


Figure 3.15: Critical traction values in the a) normal and b) tangential directions of each cut section using a 0.2mm section length. c) Graphical representation of the normal and tangential traction values.

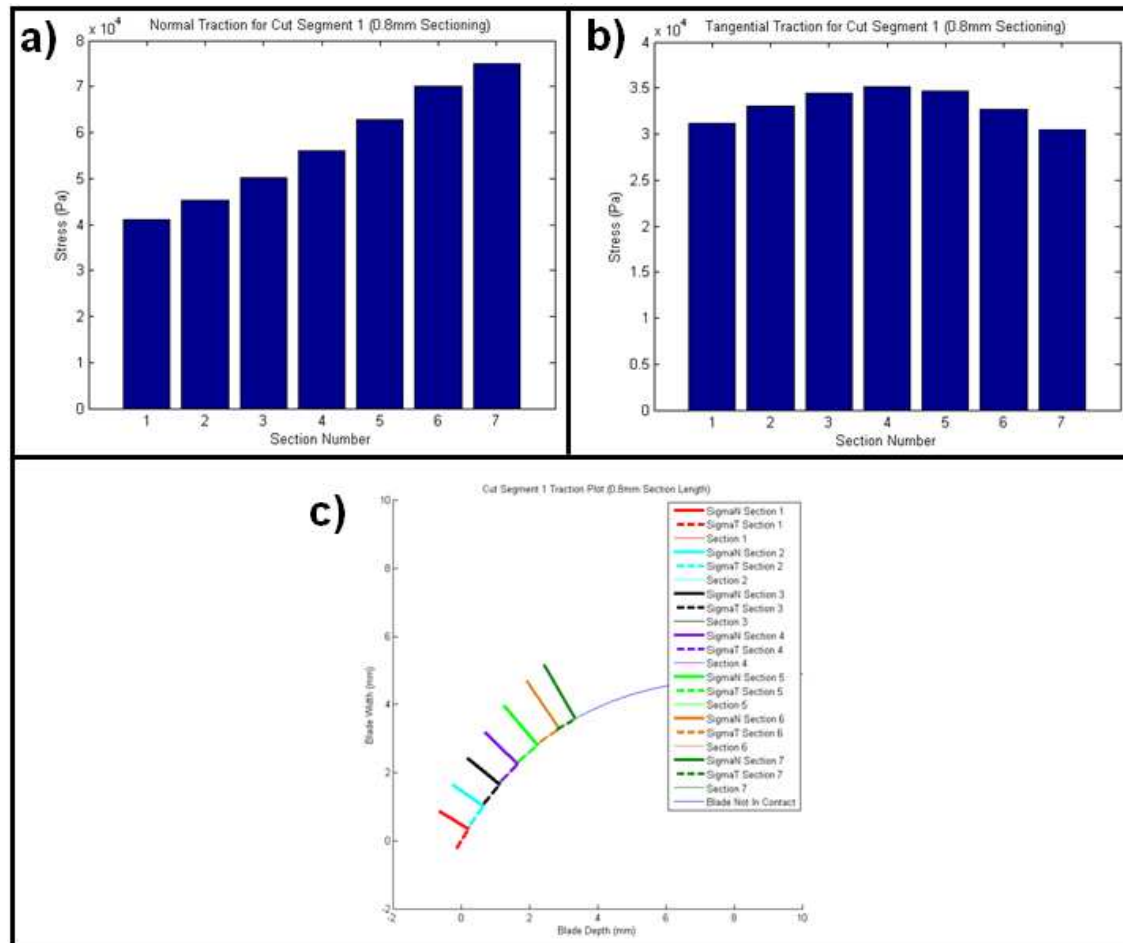


Figure 3.16: Critical traction values in the a) normal and b) tangential directions of each cut section using a 0.8mm section length. c) Graphical representation of the normal and tangential traction values.

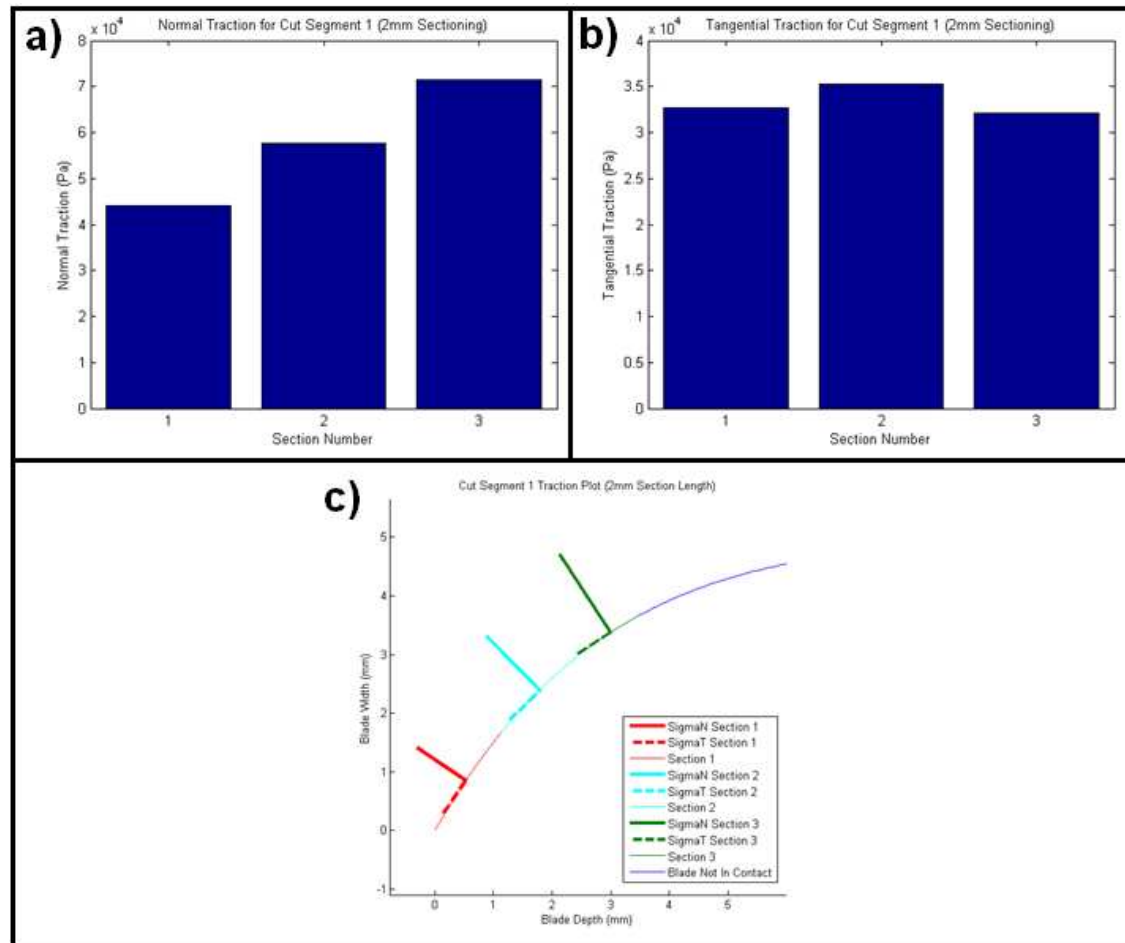


Figure 3.17: Critical traction values in the a) normal and b) tangential directions of each cut section using a 2.0mm section length. c) Graphical representation of the normal and tangential traction values.

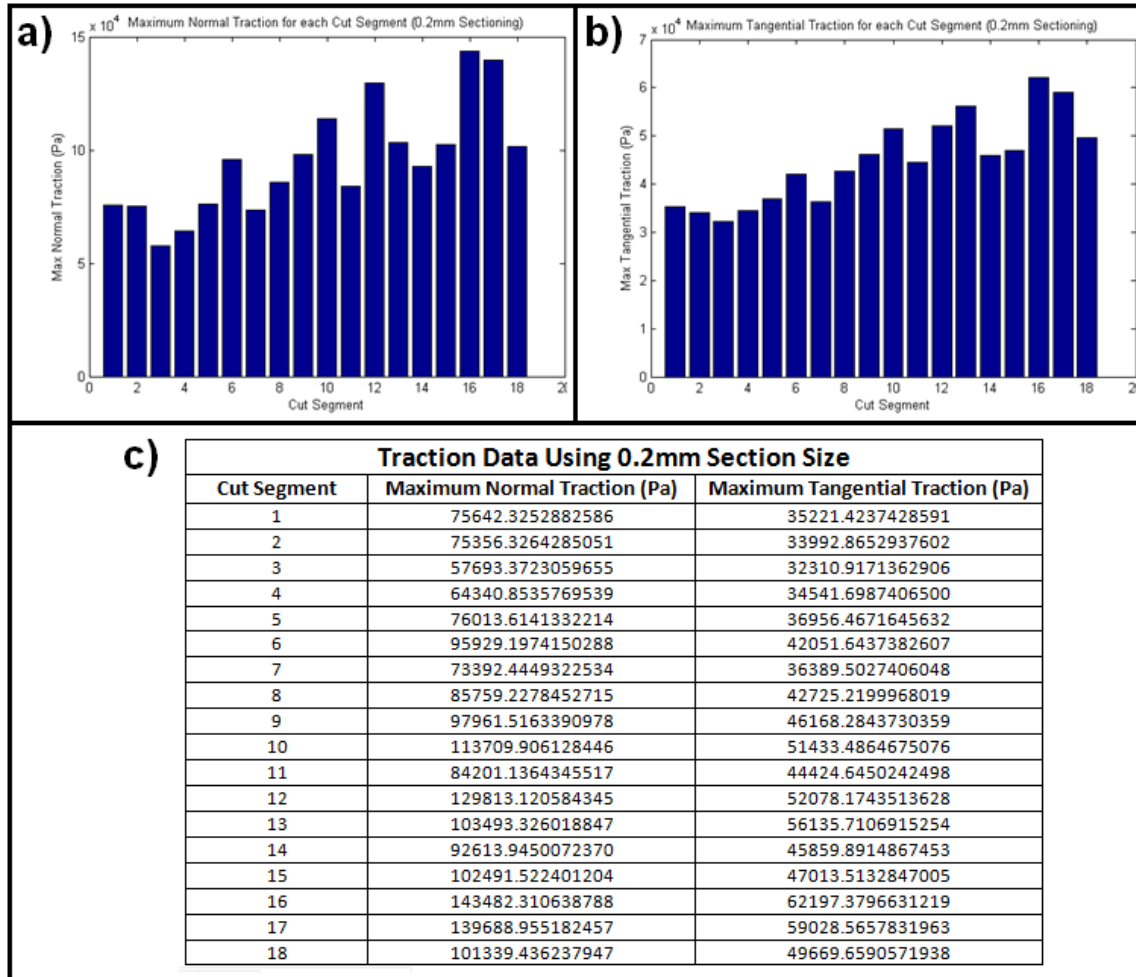


Figure 3.18: Cohesive strength parameter for the a) normal and b) tangential direction for each of the eighteen cut segments using 0.2mm sectioning. c) Table of exact values.

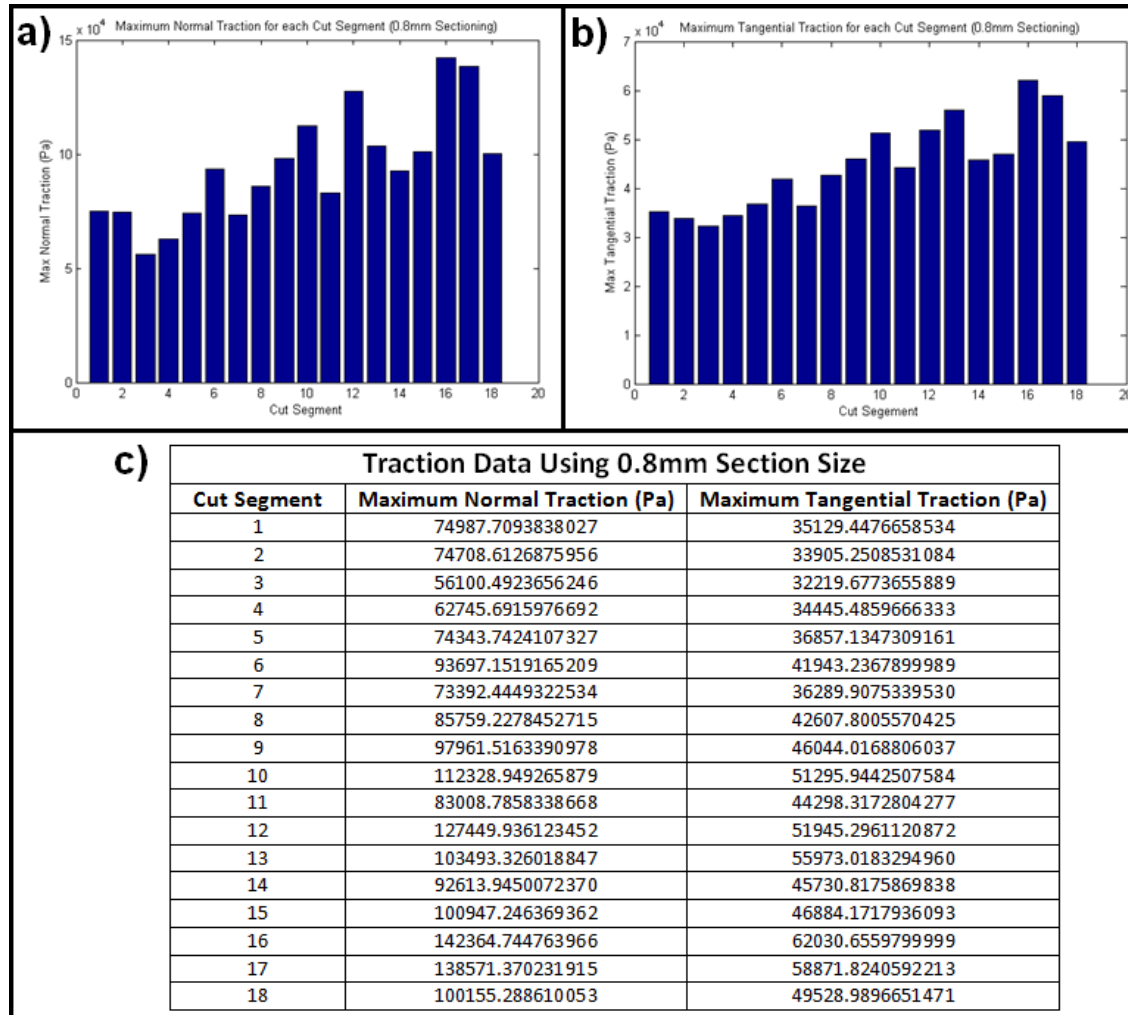


Figure 3.19: Cohesive strength parameter for the a) normal and b) tangential direction for each of the eighteen cut segments using 0.8mm sectioning. c) Table of exact values.

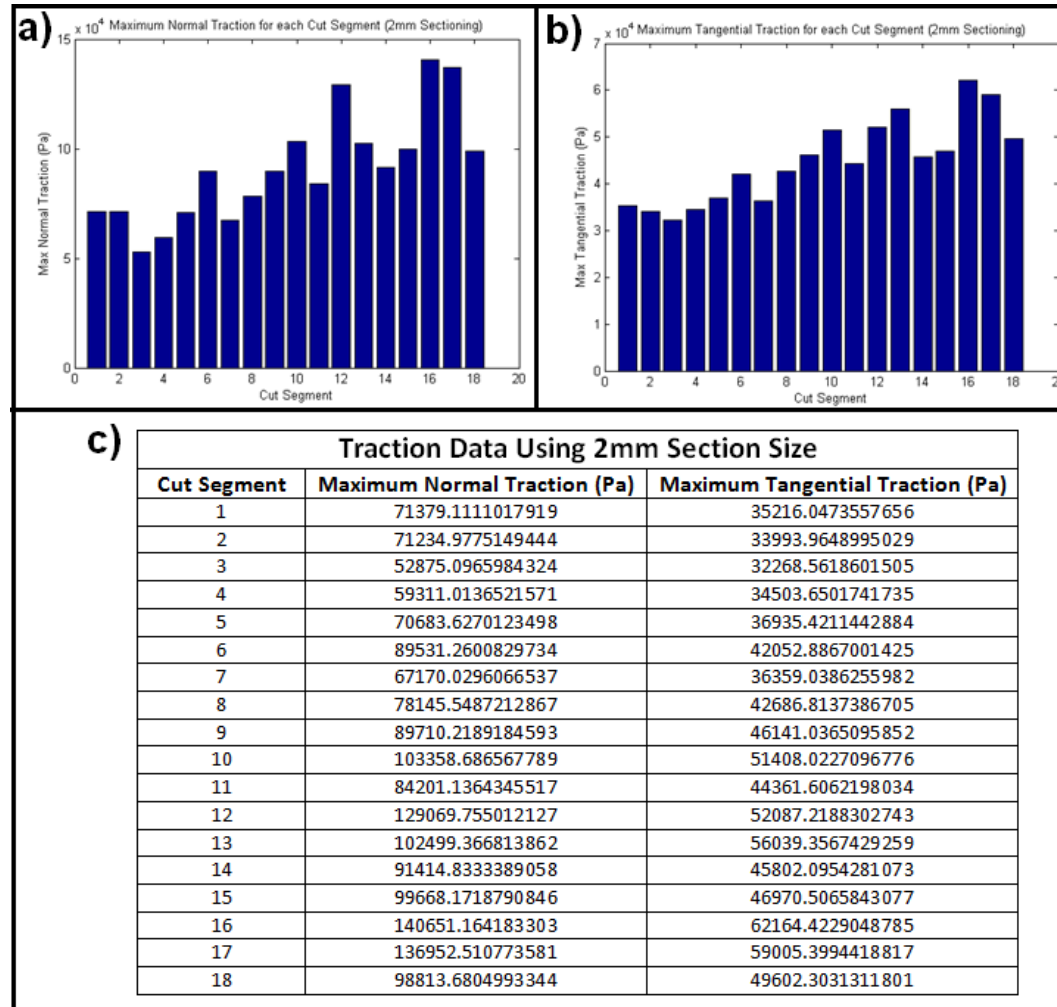


Figure 3.20: Cohesive strength parameter for the a) normal and b) tangential direction for each of the eighteen cut segments using 2.0mm sectioning. c) Table of exact values.

Next, the parameter for the cohesive energy was determined from the cutting force data as well. The cohesive energy is the energy required to complete the fracture process; thus it encompasses a combination of the forces measured in the cutting and vertical directions. Using the fits defined in Figure 3.6, the energy parameters were derived for each cutting segment by integrating each of the curves over the length of the individual fracture segment. This integration was performed for both the cutting and vertical force profiles. Figures 3.21a and b show the energy values for each of the segments for the cutting and vertical directions respectively. The total cohesive energy for each cutting segment is a combination of the individual energy parameters (Figure 3.21c). The numerical values for the energy parameters are given in Figure 3.21d.

Using this information in combination with the cohesive strength values results in a fully defined cohesive model for the scalpel-tissue cutting procedure. The analysis presented for the first *in vivo* scalpel cutting experiment was able to derive cohesive parameters required to generate eighteen individual traction-separation curves. Each of these curves will vary slightly as the biological tissue in and of itself is slightly non-uniform in nature. By generating these distinct models, variation in the resultant force profile can be implemented in the simulation by randomly selecting between the different traction-separation curves. This will enhance the realism of the surgical simulator as the fluctuations in the cutting response will mimic the trends seen in experimental procedures.

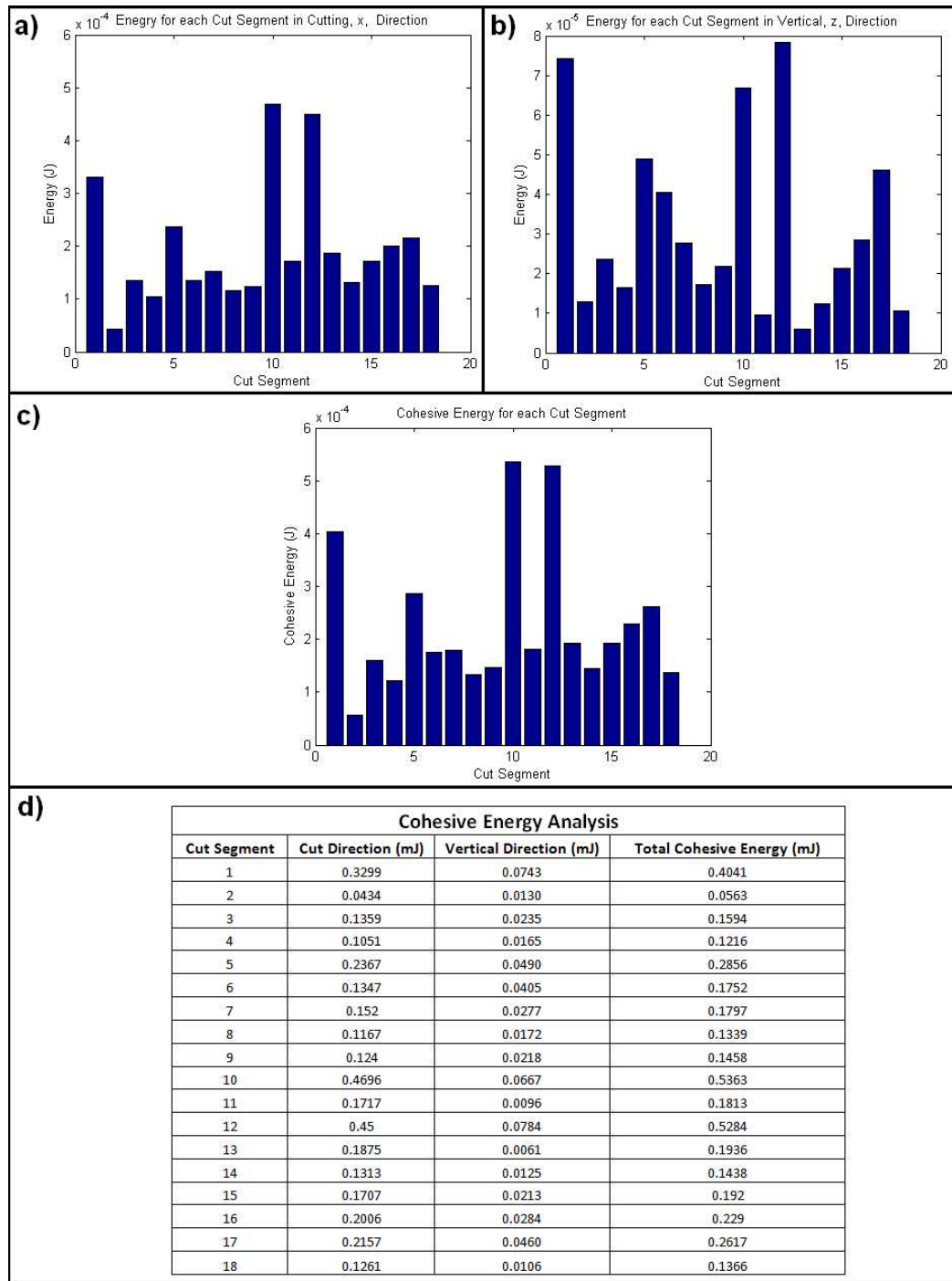


Figure 3.21: Energy values for each cut segment from a) cutting force b) vertical force c) total cohesive energy. d) Exact values for the various energy calculations.

3.2 Simulation of Cutting

After the determination of the cohesive zone parameters, verification of the force-displacement response arising from the implementation of the cohesive zone model was required. To validate the accuracy of the cohesive zone approach, a detailed study was conducted in ABAQUS [48]. Through this study, the functionality of the cohesive zone approach, as applied in finite element based simulations, is presented.

3.2.1 Cohesive Zone Model Verification

The simulation of the cutting process in ABAQUS consists of three distinct subsections that are combined to form the fracture process. The first issue centers around the definition of the material characteristics for the bulk tissue properties. As the cohesive model deals directly with the stress and strain characteristics of the material in front of the crack tip, a general material model must be specified to govern the overall response of the tissue. Second, the loading of the tissue must be conducted in the same manner as the scalpel cutting process. To assure the validity of these boundary conditions, the most accurate representation is obtained through the development of a contact analysis between realistic tissue samples and scalpel geometries. This problem is made more difficult by the small, sharp nature of the scalpel shape. Finally, the cohesive zone itself must be included in the finite element simulation. The cohesive zone has mainly been utilized for simulation of material delamination in ABAQUS, hence, the applicability to the direct loading of

the cohesive zone elements remains uncertain.

The first major component of the simulation process is the definition of the material properties for the bulk tissue response. For the purposes of this simulation, to maintain as much accuracy as possible, the reduced polynomial model, $\Psi = \sum_{i=1}^N C_{i0} (\bar{I}_1 - 3)^i + \sum_{i=1}^N \frac{1}{D_i} (J_{el} - 1)^{2i}$, derived from the *in vivo* experiments in Section 2.2.6 was used to define the material properties.

The reduced polynomial model is by definition non-linear and hyperelastic, which greatly effects the stability of the simulation. The nature of biological tissue, and the reduced polynomial utilized in this simulation, produces a hardening type of behavior under load, as can be seen in the model derivation sections previously presented. This does not present a problem in the small strain region. However, when large strains are required, the implementation utilizing finite element approaches becomes more difficult. Under large strain conditions, the material begins to increase in stiffness quickly enough that the material becomes infinitely stiff resulting in the inability of an element to deform past its current state. This problem is caused by a combination of the material model, the type and size of elements used, and the method of simulation. Unfortunately, when attempting to simulate the scalpel cutting process the large strain region can not be avoided. Implications pertaining to these issues will be discussed shortly.

The second requirement for the simulation of the scalpel-tissue interaction is the proper definition of contact for the analysis. In this case, contact properties need to be established between the scalpel blade and the tissue, as well as the tissue to itself. As the blade begins to pass through the tissue, it will part the liver

surface into two regions. After the scalpel has traveled an extended distance, the two surfaces will have the ability to contact one another in the wake of the scalpel. Therefore, a contact definition needs to include the scalpel-tissue contact as well as the tissue self intersection region.

Contact can be defined through multiple approaches in ABAQUS. The previous studies conducted under this research effort utilized a contact pair definition, which provided the context for the probing simulation. A similar approach was applied to the cutting simulation whereby a contact pair was defined between the tissue and the cutting surface of the scalpel blade. However, additional contact definitions were required to prevent the self intersection of the tissue after separation occurs. As one of the requirements for implementation of the cohesive zone approach in ABAQUS is prior knowledge of the crack path, the two surfaces that make up the new faces of the tissue following separation are known before the start of the simulation. Therefore, a second contact pair definition was defined between these two surfaces to prevent self intersection in the wake of the scalpel blade.

To obtain the proper contact between the scalpel and the tissue an appropriate geometry was required for the scalpel blade. During the analysis of the *in vivo* experimental cutting data, both the profile and the cross section of the scalpel blade were precisely measured. Based on these measurements a three-dimensional representation of the blade was created. To prevent problems in the contact algorithm due to sharp edges, the leading edge of the scalpel blade was filleted with a small radius. Upon completion of the shape, the geometry was imported into the ABAQUS simulation as a discrete rigid object due to the relative stiffness compared

to the soft-tissue.

In addition to the scalpel geometry, a realistic representation of the lobe of liver used during the *in vivo* cutting experiments was desired for the validation study to most accurately replicate the cutting process for comparison to the experimental results. The method of obtaining the shape of the lobe of liver for use in a finite element simulation is the same as presented in Section 2.2.2. In brief, during the *in vivo* experiments, the MicronTracker camera system was used to generate a point cloud representation of the liver surface. A picture taken during the experiment and the corresponding point cloud data are show in Figure 3.22a and Figure 3.22b, respectively. Next a solid model representing the liver was developed (Figure 3.22c). In this case, the solid model was split into two halves representing the two sections formed by the scalpel cutting process. This method was applied to aid in the meshing process as well as the definition of the cohesive zone parameters in ABAQUS.

After the geometry was developed, it was imported into HyperMesh, a powerful independent meshing software application. For simulation purposes, a dense mesh is desired in the region directly in front of the scalpel blade to fully define the stress

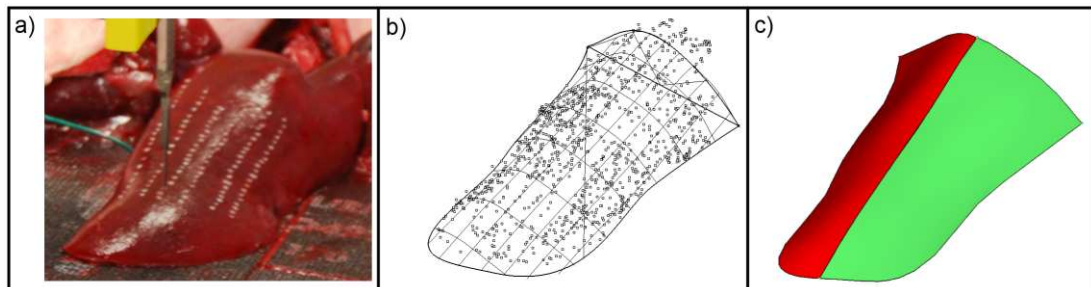


Figure 3.22: a) Image of liver from *in vivo* cutting experiments, b) point cloud data fit with liver surface and c) solid model used in meshing process.

concentrations imparted to the tissue during the scalpel loading process. However, the use of such small elements will increase the computation time required for the simulation and bring about convergence issues caused by excessive distortion related to the use of the hyperelastic constitutive model. Thus, a trade off in mesh density is required. In the case of this validation study, the solid model was split into four different sections; two corresponding to the bulk of the liver not directly loaded by the scalpel blade, and two comprising the direct loading region in the cutting path. Figure 3.23a shows the meshed liver geometry in its entirety while Figure 3.23b depicts the line of dense elements constituting the cutting path. The elements used in the larger bulk tissue sections were defined as C3D4 tetrahedral elements and the elements along the cutting path were C3D8 hexahedral elements. The size of the two different elements were determined through experimentation to maximize the stable deformation in front of the scalpel blade during the loading process.

Upon completion of the mesh generation, all files were imported into ABAQUS to begin the scalpel cutting simulation. Preliminary studies were conducted using a standard implicit approach, however the final data presented here resulted from a dynamic, explicit simulation. The switch from implicit to explicit methods was driven by two conditions. First, the implicit studies involving the scalpel-tissue contact resulted in stability issues regardless of mesh density. These instabilities were still present in the explicit approach, but appeared later when larger deformation and stress values were present. Second, the approach envisioned for the future real time simulators was intended to be used with the Total Lagrangian Explicit Dynamics methodology to decrease the computation time. Therefore, it was logical to

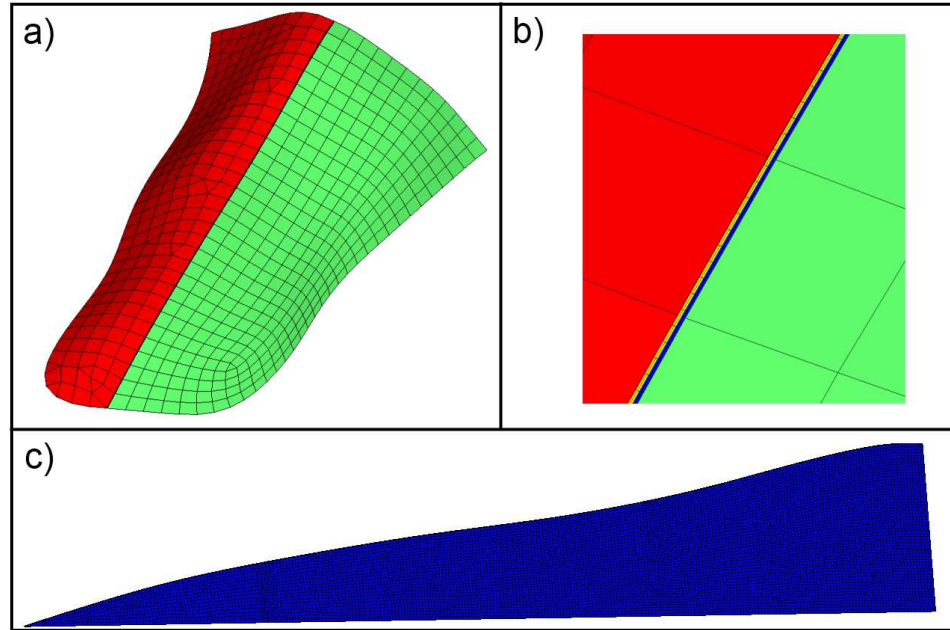


Figure 3.23: a) Full mesh for the cohesive zone validation study, b) Close up of cutting path, c) Cross section of dense elements in the cutting path.

switch to the explicit modeling scheme in ABAQUS to increase the stability as well as evaluate the ability to simulate the cohesive zone fracture approach using the methods desired for future real-time simulation.

The final mesh layout used in the validation study can be seen in Figure 3.24. It was modified slightly in ABAQUS to remove excess material in the liver prior to the location in which the cut begins. Also, the elements at the start of the cutting path were split into two C3D6 wedge element to facilitate the entry of the scalpel blade into the desired cutting path. Simulations were conducted without the wedge elements, but difficulties arose due to element buckling directly in front of the scalpel loading. This caused instabilities in the simulation before the tissue could reach a point where it would settle around the blade. The inclusion of the wedge elements

facilitated the parting of the mesh around the blade and eliminated the buckling instabilities. The final mesh consisted of 39,975 elements (35,344 hexahedral, 4,567 tetrahedral and 64 wedge).

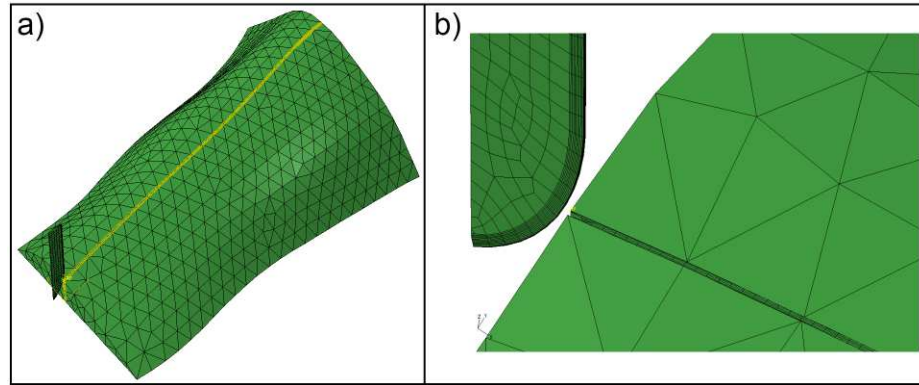


Figure 3.24: a) Mesh for the cohesive zone validation study with contact regions displayed in yellow, b) Close-up of mesh directly in cutting path.

With the constitutive model, mesh and contact conditions fully defined, the final major component to the simulation is the addition of the cohesive zone. As mentioned earlier, all previous parameters have a direct influence on the stability of the overall simulation. With the use of the hyperelastic reduced polynomial model, stability problems arise due to element distortion. Many different mesh designs of various element types and densities were used in order to facilitate the maximum stress and deformation within the tissue prior to the occurrence of instabilities in the simulation. The aim was to use elements small enough to reach the stress levels observed during the *in vivo* scalpel cutting experiments. However, to reach these high stress levels very small elements were required, which resulted in instability issues prior to achieving the proper stress and deformation levels.

During the verification process, the most acceptable configuration was chosen to maximize both the stress and deformation criteria. However, the element size in the region in front of the scalpel blade was larger than desired. This result yielded tissue deformations of the correct magnitude, but the internal stress conditions were lower than experimentally observed due to the larger characteristic size of the mesh in the region in front of the scalpel blade. To overcome this dilemma, the magnitude of the cohesive parameters were in turn scaled down proportionately to the level of the maximum stress within the cutting region prior to simulation instabilities. While this will result in inaccurate internal stress distributions within the tissue, it serves to relay accurate force and displacement characteristics through the cutting process. For the purposes of a surgical simulator, the force and displacement are the important signals to send to the end user, the internal stress will not be evident and thus need not be weighed as heavily. A study depicting the validity of scaling the cohesive zone parameters will be presented in Section 3.2.2.

The cohesive zone implementation in ABAQUS can be completed in two ways, an element-based or a surface-based approach. As four distinct mesh sections were already defined, including two making up the dense mesh region in front of the scalpel blade, the surface-based approach was selected due to its direct application to the existing model. A surface was defined along the cutting path on each of the mesh sections constituting the cutting zone (the yellow and blue sections of Figure 3.23b). Initially, the two surfaces coincide with one another, however, as loading begins the attachment of the surfaces to one another are governed by the cohesive parameters. Even though the surface definition is utilized to define the

cohesive region, the implementation of the methodology in ABAQUS is applied to the individual elements within the surfaces. Therefore, when the parameters of the cohesive region are exceeded, the effects are applied on an element by element basis to simulate individual local fractures, as naturally occur within soft-tissue cutting.

The complete setup for the ABAQUS simulation can be described with the following parameters:

- Parts Definition

- Left bulk tissue - 1,970 tetrahedral elements (C3D4)
- Right bulk tissue - 2,597 tetrahedral elements (C3D4)
- Left cutting path - 17,672 hexahedral (C3D8) and 32 wedge (C3D6) elements
- Right cutting path - 17,672 hexahedral (C3D8) and 32 wedge (C3D6) elements

- Material Definition

- Tissue - Reduced polynomial $\left(\Psi = \sum_{i=1}^N C_{i0} (\bar{I}_1 - 3)^i + \sum_{i=1}^N \frac{1}{D_i} (J_{el} - 1)^{2i}\right)$:
 $C_{10} = 781.898 \frac{N}{m^2}$, $C_{20} = -747.631 \frac{N}{m^2}$, $C_{30} = 426.135 \frac{N}{m^2}$, $C_{40} = 122.701 \frac{N}{m^2}$
and $D_1 = D_2 = D_3 = D_4 = 0.0001 \frac{m^2}{N}$. Mass Density - $981 \frac{kg}{m^3}$.
- Scalpel - Discrete rigid

- Step Definition

- Dynamic, explicit approach

- Establish contact step - 0.1s time period, move blade 0.005m to establish contact using a smooth step amplitude; Maximum permissible time step of $2.5e^{-5}s$
- Cutting step- 0.6s time period, move blade 0.03m to establish contact using a smooth step amplitude; Maximum permissible time step of $2.5e^{-5}s$
- Interaction Definition
 - General contact approach, selected surfaces (scalpel blade to cutting path surfaces), frictionless contact
 - Add individual property assignment to cutting surfaces for cohesive zone definition
- Interaction Properties (Cohesive Zone Parameters)
 - Normal Traction - 13.594kPa
 - Tangential Traction - 6.414kPa
 - Energy - 0.125mJ
- Constraint Properties
 - Tie back side of cutting patch elements to the bulk tissue elements
- Boundary Conditions
 - Fix blade rotation - constrain all DOF except translation in cutting direction

- Fix bottom of tissue - constrain all DOF along bottom tissue surface
- Move blade - blade motion as described in Step Definition above

Following the simulation, the main aspects of comparison between the simulation and experiment were the functionality of the cutting process and the correlation to the force profile trends seen in the *in vivo* cutting experiments. With regard to the functionality aspect of simulating the cutting process, the desire was to provide the capability for the scalpel blade to pass into the finite element mesh while separating the elements within the blade's path. In addition, the prevention of self intersection in the wake of the blade was also an important aspect of the simulation. As can be seen in Figure 3.25, the scalpel blade was able to replicate all aspects in a realistic manner. In fact, the simulation behaves similar to the actual *in vivo* cutting tests in that the tissue directly in front of the blade was locally loaded until failure where it relaxed around the scalpel blade. This process was repeated through the entirety of the cut to constitute individual loading and unloading segments as seen in real life.

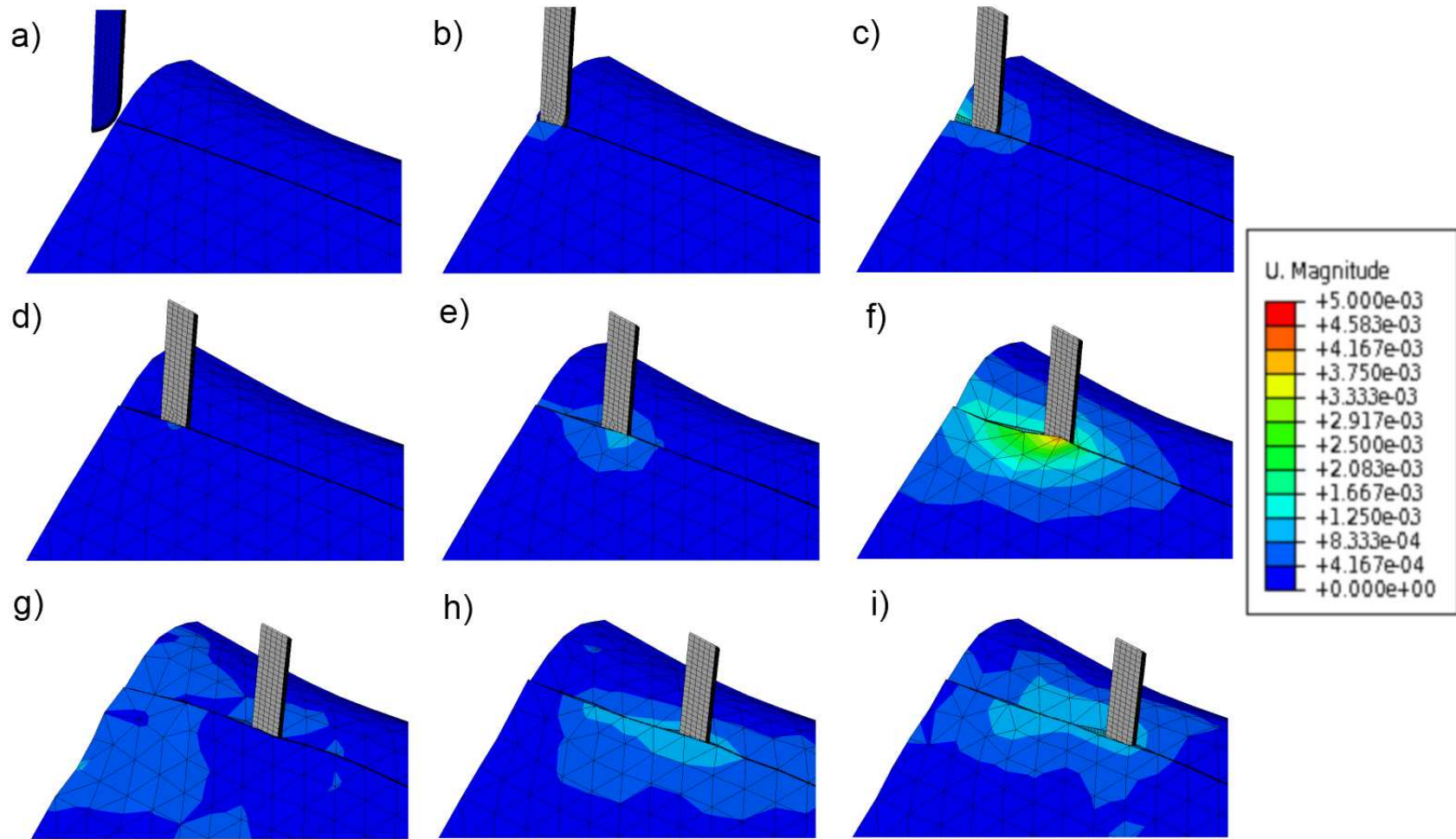


Figure 3.25: Progression of the scalpel and resultant tissue displacement through the cohesive zone verification simulation.

The next aspect to check was the similarity between the simulated and experimental force profile. It should be re-emphasized that the aim is not to replicate the exact force profile obtained in the *in vivo* experiment, but to develop a physically-based method capable of mimicking the important characteristics of the force profile itself. The three main aspects of the force profile required to accurately represent the trend of the force data are the average cutting force, the magnitude of the cutting peaks and the frequency of the individual loading segments. The force profile generated by the simulation is displayed in conjunction with the experimentally measured force in Figure 3.26.

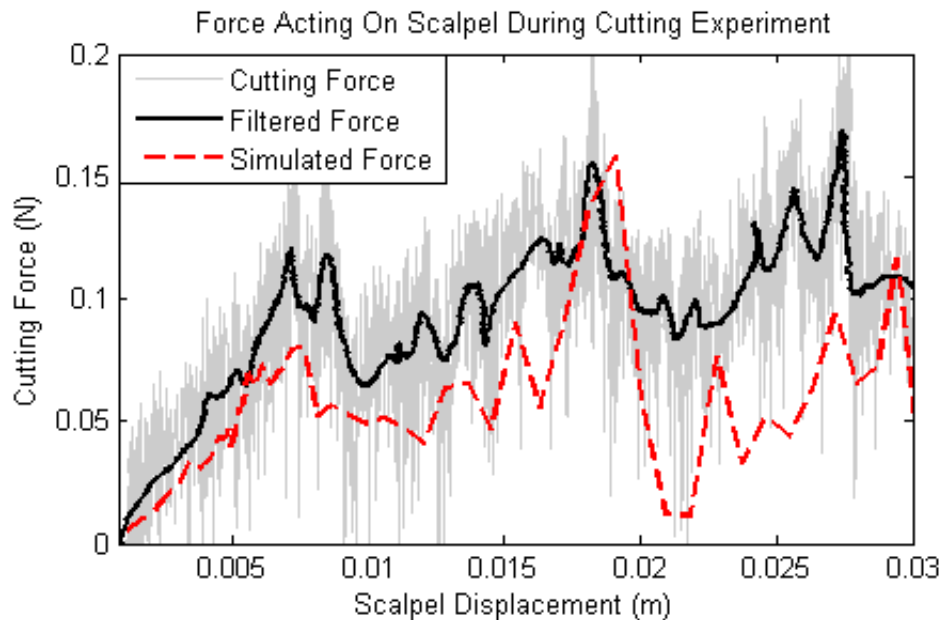


Figure 3.26: Force profile acting on the scalpel blade during the *in vivo* experiment and corresponding simulation. ©2012 CARS.

As can be seen, the use of the cohesive zone model in the finite element simulation resulted in a reaction force on the scalpel blade on a similar scale to the

experimental data. The individual loading segments replicate the general sawtooth pattern observed while cutting soft-tissue. The magnitude of the individual fracture occurrences vary throughout the course of the simulation and the peak-to-peak loading falls within the range of what is observed experimentally. Finally, the frequency of the fracture events also maintains a high correlation between the simulated and experimental data. Discrepancies between the two data sets are expected as the experiment was conducted on a continuous liver sample and the simulation represents a discrete approximation. The overall trends of the two force profiles, however, do project a similar response, especially in the three main areas of interest for medical training purposes.

The cohesive zone approach utilizes the stress, strain and energy values to update the current fracture status (a depiction of the stress distribution within the tissue during simulation is shown in Figure 3.27). The stress and strain values are

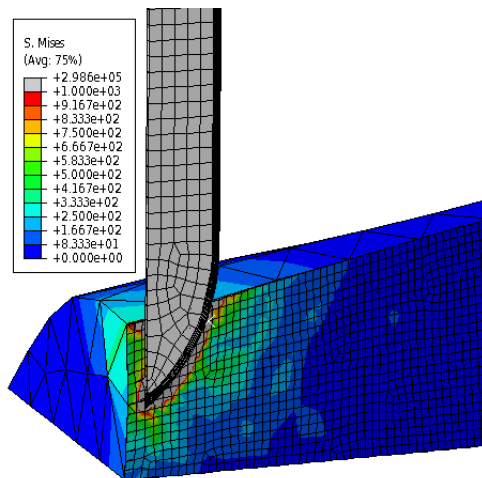


Figure 3.27: Depiction of stress distribution within tissue during scalpel cutting process.

already contained in the finite element simulation to update the nodal displacements. Therefore the addition of a cohesive region to an existing finite element simulation does not serve to greatly increase the complexity of the analysis. This promotes the promise of the method for real-time, finite element based approaches. The major limitation in the presented validation study to real-time application is the large number of elements utilized in the region in front of the scalpel blade. To that end, a study was conducted to determine the feasibility of scaling the parameters of the cohesive model to accommodate larger mesh densities in the cutting region.

3.2.2 Cohesive Zone Scaling Study

During the course of the cohesive zone validation study it was noted that modification in the mesh size proved to change the internal stress in the tissue region, however it did not significantly effect the reaction force response or maximum deformation. Therefore, the idea of scaling the parameters of the cohesive zone model based on the mesh density arose. To determine the feasibility of such an approach, a series of simulations were conducted to quantitatively measure the differences in the force, displacement and stress values arising in a cutting analysis due to changes in mesh density.

A simplified geometry (Figure 3.28) was utilized for the tissue substrate in this study to facilitate faster computation times. This shape was meshed with three different densities corresponding to 1467, 1671 and 16681 elements (referred to as coarse, normal and dense respectively). The constitutive model, once again, was

defined as the *in vivo* based reduced polynomial previously presented. Initially a simulation scheme was set up without utilizing the cohesive zone method. The surfaces instead were tied together to replicate loading of an intact tissue region. This provided the ability to measure the maximum stable deformation due to scalpel loading. At this point the the average stress value within the tissue and reaction force acting on the scalpel blade were also measured. A depiction of the stress distribution at maximum deformation for each mesh density is displayed in Figure 3.28.

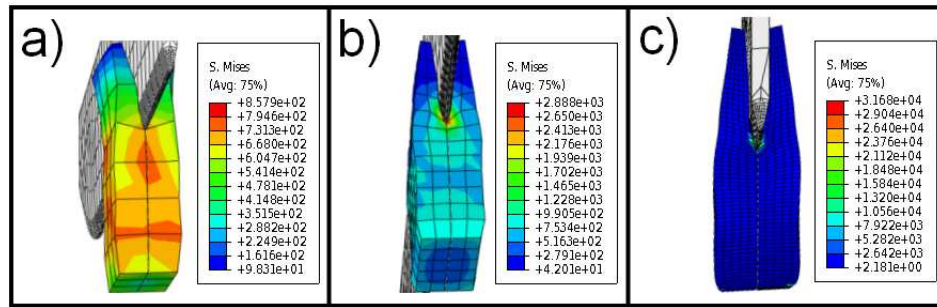


Figure 3.28: Depiction of stress distribution within tissue during scalpel cutting process for a) coarse, b) normal and c) dense meshes.

The results for the tissue displacement and the corresponding stress are shown in Table 3.2. As can be seen, the maximum displacement for the normal and dense element cases both showed a value of 1.5mm. In addition the reaction force acting on the blade in the final stable time step also resulted the same magnitude of 0.018N for these two mesh densities. The more coarse mesh, however, was not able to achieve these force and displacement levels prior to reaching the instability limit showing that there is a limit to the maximum size of the mesh that can be used for simulating scalpel contact. Despite the fact that the force and displacement

characteristics showed similar trends for the normal and dense elements, the stress values differed significantly. This is expected due to the change in element size.

Table 3.2: Maximum deformation and stress values for cohesive zone scaling study.

Mesh Density	Max Displacement	Stress	Reaction Force
1467 Elements	0.8 mm	0.3 kPa	1.3E-2 N
1671 Elements	1.5 mm	2 kPa	1.8E-2 N
16681 Elements	1.5 mm	20 kPa	1.8E-2 N

Using the results from the deformation simulation, the cohesive zone cutting approach was included in the simulation to allow for scalpel cutting. The cohesive parameters from the model derivation process were scaled proportional to the internal stresses that appear during the deformation analysis. This approach provided the means to obtain very similar scalpel cutting properties in both the normal and dense mesh configurations. Once again, some discrepancies arose with the coarse mesh due to the inaccurate nature of the force-displacement response. The tissue deformation during the cutting process is shown for the first node along the cutting path for all three cases in Figure 3.29a. The normal and dense mesh had a high degree of correlation in the nodal deformation prior to the point of fracture, which was located at a displacement of 0.31 mm for each case. In addition, the force values were very similar for the normal and dense mesh as can be seen in Figure 3.29b. The magnitude of the force at the point of initial fracture was 1.12E-3 N for the

normal case and $1.28\text{E-}3$ N for the dense case. The magnitude of the force in this study was lower than the previous simulations due to the smaller sample size and the lower threshold set for the cohesive zone parameters. The average magnitude cutting force throughout the test was also similar. Finally, the peak-to-peak individual loading segments and frequency of individual fractures also showed the same trends throughout the entirety of the simulated cutting process. As with the previous detailed cutting study, the exact force profile was not expected to match due to the different mesh densities, but the important aspects of the force profile for the simulation of scalpel cutting were preserved.

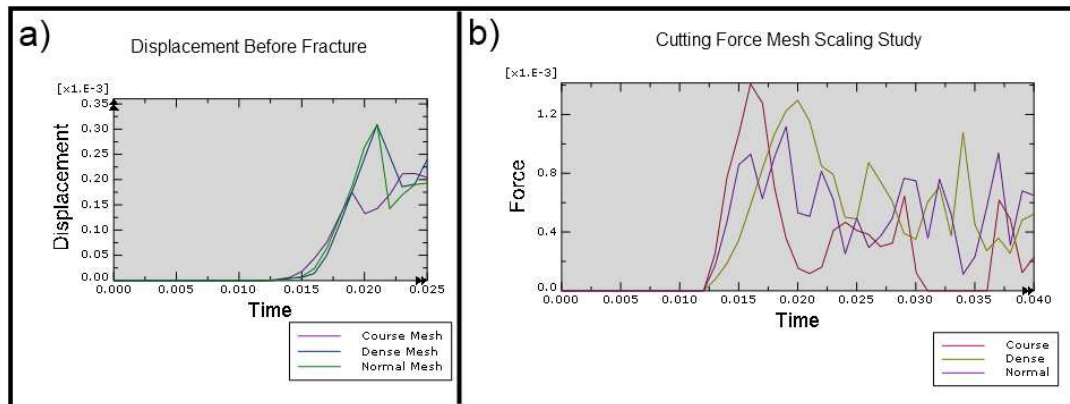


Figure 3.29: a) Displacement and b) force plots for cohesive zone scaling test

Through this study it can be noted that the cohesive zone properties can be scaled to fit a mesh of varying size presuming that the mesh is dense enough to produce a stable response through the entirety of the desired deformation path. If the element size is too large, as was the case in the coarse configuration, the initial parameters for the force and displacement will be inaccurate, which will lead to an incorrect cutting response. However, any mesh above the minimum threshold should

be sufficient for use with the cohesive zone cutting response. As is seen in this case, the size of the elements used in the normal density case is approximately twice the width of the scalpel blade. Use of this relatively large element size along the fracture line will be sufficient to fully define the cutting modality, while minimizing the total number of elements required for the cutting simulation. This will result in decreased computation times aiding in the ability to use the cohesive zone methodology in real-time, finite element based simulations. Figure 3.30 shows the full cutting simulation sequence for each of the three mesh densities used in this study.

The cohesive zone cutting methodology implemented in this section was required to add the scalpel cutting functionality to a surgical simulation system. Using the previously developed constitutive models for soft-tissue response to general loading in combination with the cohesive zone cutting models it is possible to simulate a scalpel cutting task on soft tissue. Being that both the constitutive model and the cutting model were developed directly from experimental data, the tissue deformation and forces induced by the scalpel blade are highly realistic. In addition, the development of the cutting model parameters was performed for many different cut segments. By randomly selecting one of these sets of model at the beginning of the simulation, the variability observed in soft tissue can easily be replicated in the training simulator further increasing the realism experienced by the user.

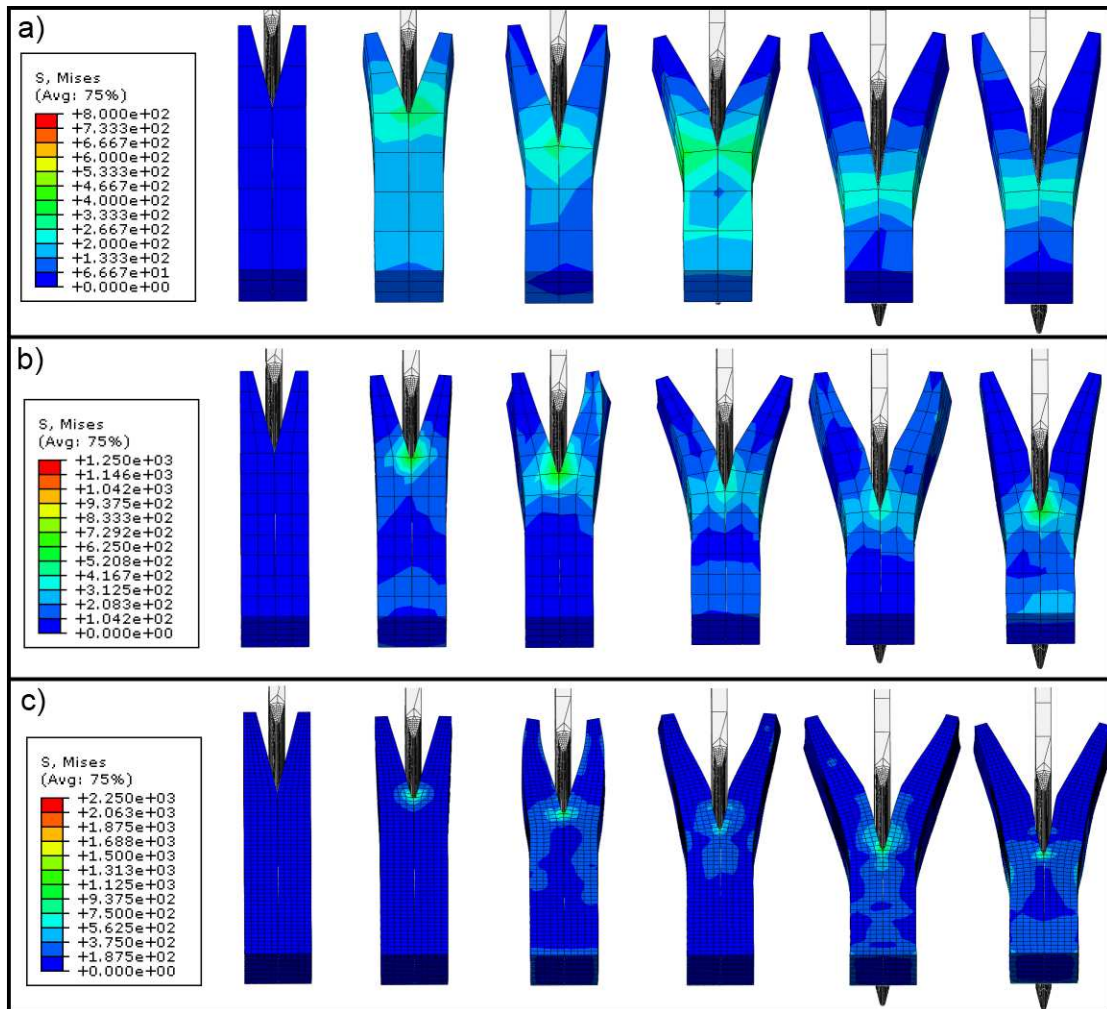


Figure 3.30: Stress distribution throughout cutting process for a) coarse b) normal and c) dense meshes.

Chapter 4

Real-Time Simulation

Many different methods have been utilized to develop surgical simulators for training of medical tasks; each have benefits as well as drawbacks. Surgical simulators often utilize mass spring-models due to the simple implementation and fast computation; however, these models do not provide the desired accuracy since they use a linear elastic approach to govern the material characteristics of the biological tissues, which is an inaccurate representation. This project aims to utilize the finite element method (FEM) in conjunction with the more realistic hyperelastic material models to develop more accurate, comprehensive simulation systems. Two approaches are presented to deal with the complexity of the issues that comprise a full FEM based approach with reality-based models and realistic geometry. First, a preprocessor based approach was developed in which a real-time response was developed for probing and cutting of liver tissue using experimentally determined hyperelastic constitutive models. This method provides the framework required to achieve real-time display rates for both visual and haptic information. Second, an open source, finite element based code was utilized to simulate soft tissue probing using the hyperelastic approach where computations are done in real-time. This provides more flexibility in the simulator in that all tool paths are not required prior to simulation.

4.1 Preprocessor Based Probing Simulator

The first approach to the solution of the reality-based, haptics-enabled simulation of liver probing in real-time makes use of preprocessed information. The goal of this simulator was to use the reality-based material models, developed in the previous *ex vivo* and *in vivo* experiments, as the basis for the probing simulation. Many modern simulators make concessions in the accuracy of the material model to simplify the complexity of the finite element approach. To attain the desired level of realism, accurate hyperelastic models previously derived were used to improve upon the overall system response. By using the reality-based models, the forces and deformations involved in the probing task will be much more realistic, thereby making for a superior training system. To use the accurate material models in real-time, a method of preprocessing the data has been developed for the implementation of an Ogden model in a real-time simulation of soft tissue probing [44].

4.1.1 Fundamentals

In this version of the probing simulator, the Ogden model developed from the *ex vivo* experimental process, $U = \sum_{i=1}^N \frac{\mu_i}{\alpha_i} (\lambda_1^{\alpha_i} + \lambda_2^{\alpha_i} + \lambda_3^{\alpha_i} - 3)$, presented in Section 2.1 was utilized. Despite the fact that the model is based on direct experimental tests of fresh *ex vivo* porcine liver, the finite element approach may not yield to correct results unless all other parameters involved in the analysis are accurately set as well. Therefore, it was essential to develop a method to check the validity of the settings used in the finite element approach. To do so, an analysis of an

unconfined compression test was conducted with a simple cylindrical sample prior to performing the probing task on a realistic organ geometry. The output of the finite element simulation was compared to the theoretical derivation to assure that all settings in the analysis, such as element type and size, were appropriate for use in the analysis of the realistic tissue geometry. All FEA simulations were conducted using ABAQUS software (Version 6.8 from Dassault Systèmes S.A.)

4.1.1.1 FEA Parameter Verification

The simulation set-up for the validation test consisted of a cylindrical test sample that undergoes loading identical to an unconfined compression test. To simplify the boundary conditions required for accurate analysis, only a quarter of the 2.54cm diameter cylindrical sample was modeled in ABAQUS. This is a technique commonly used when performing FEA on symmetric samples. The sample was constrained along the vertical planes to prevent motion perpendicular to each plane. This approach has the property of keeping the sample centered at its original state while allowing for radial expansion of the sample as is required for an unconfined compression test. The sample was also constrained in the vertical direction on the bottom surface to prevent movement while a displacement boundary condition (40% compression) was applied to the top surface. Material properties were defined by a two-term Ogden model developed for ex-vivo pig liver from the procedures presented in Section 2.1. The geometry was meshed with 1329 quadratic tetrahedral elements.

The analysis was performed in 20 steps to generate the force vs. displacement

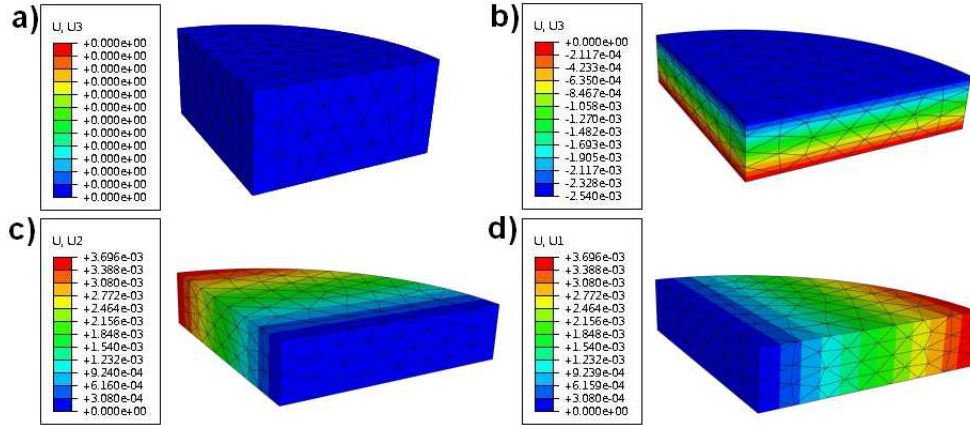


Figure 4.1: Displacement plot for the unconfined compression test for a) original sample b) vertical displacement c and d) radial displacements.

curves. Figure 4.1 depicts the displacement plots for the simulation of the unconfined compression test. Figure 4.1a shows the original geometry of the tissue sample. Figure 4.1b shows the vertical displacement after the loading of the sample has concluded. As expected, the tissue displaced a total of 2.54mm which corresponds to the desired 40% compression. Another important aspect to check to assure the test is working properly is the radial expansion of the sample. As can be seen in Figures 4.1c and 4.1d, the sample did expand uniformly in the radial direction as the two deformation values are the same. The force vs. stretch ratio plot can also be determined from the simulation (Figure 4.2). In this case, the final force value must be increased by a factor of four to account for the whole sample diameter.

In the theoretical analysis, which was compared to the output of the ABAQUS simulation to validate the accuracy of the approach, the Ogden strain energy density

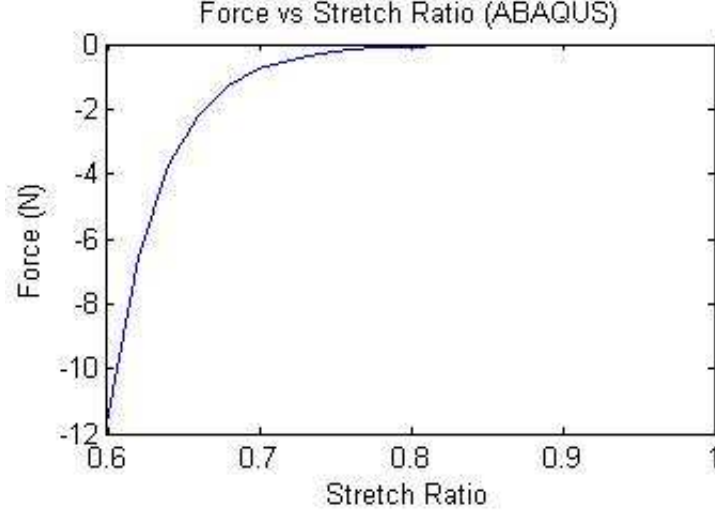


Figure 4.2: Resultant force vs. stretch ratio profile derived from ABAQUS simulation. ©2009 IEEE.

function is defined as [76]:

$$U = \sum_{i=1}^N \frac{\mu_i}{\alpha_i} (\lambda_1^{\alpha_i} + \lambda_2^{\alpha_i} + \lambda_3^{\alpha_i} - 3) \quad (4.1)$$

where μ_i and α_i are constants. For incompressible material, the principal Cauchy stresses can be determined by:

$$\sigma_j = \lambda_j \frac{\partial U}{\partial \lambda_j} - p \quad (j = 1, 2, 3) \quad (4.2)$$

Since the material is incompressible, the hydrostatic pressure, p , is decoupled from the deformation and has to be calculated directly from the equilibrium equations. Substituting Equation 4.1 into Equation 4.2, the principal Cauchy stresses for the Ogden model become:

$$\sigma_j = \sum_{i=1}^N \mu_i \lambda_j^{\alpha_i} - p \quad (j = 1, 2, 3) \quad (4.3)$$

For uniaxial compression, let $\lambda_1 = \lambda$ be the stretch ratio in the direction of compression, and $\sigma_1 = \sigma$ the corresponding principal Cauchy stress. The other two

principal stresses are *assumed to be* zero, since no lateral forces are applied in the analysis. Because the material is incompressible, we have:

$$\lambda_2 = \lambda_3 = \lambda^{-\frac{1}{2}} \quad (4.4)$$

The stress can be expressed as:

$$\sigma = \sum_{i=1}^N \mu_i \lambda^{\alpha_i} - p \quad (4.5)$$

and

$$0 = \sum_{i=1}^N \mu_i \lambda^{\frac{-\alpha_i}{2}} - p \quad (4.6)$$

Elimination of p from Equation 4.5 and Equation 4.6 yields:

$$\sigma = \sum_{i=1}^N \mu_i \left(\lambda^{\alpha_i} - \lambda^{\frac{-\alpha_i}{2}} \right) \quad (4.7)$$

In this case $N = 2$, as defined by the two-term Ogden model being utilized, which yields:

$$\sigma = \mu_1 \left(\lambda^{\alpha_1} - \lambda^{\frac{-\alpha_1}{2}} \right) + \mu_2 \left(\lambda^{\alpha_2} - \lambda^{\frac{-\alpha_2}{2}} \right) \quad (4.8)$$

The nominal stress (defined as force per unit unstrained area of cross-section) is given by:

$$T = \sigma \lambda^{-1} \quad (4.9)$$

Therefore:

$$T = \mu_1 \left(\lambda^{\alpha_1-1} - \lambda^{\frac{-\alpha_1}{2}-1} \right) + \mu_2 \left(\lambda^{\alpha_2-1} - \lambda^{\frac{-\alpha_2}{2}-1} \right) \quad (4.10)$$

Using the nominal stress we are able to determine the theoretical nominal force applied to the tissue sample (where A_{CS} is the original cross sectional area of the

tissue sample and F is the nominal force):

$$F = A_{CS} \left(\mu_1 \left(\lambda^{\alpha_1-1} - \lambda^{\frac{-\alpha_1}{2}-1} \right) + \mu_2 \left(\lambda^{\alpha_2-1} - \lambda^{\frac{-\alpha_2}{2}-1} \right) \right) \quad (4.11)$$

Using this theoretical analysis it is possible to determine the validity of the settings used for the ABAQUS simulation. First, the simulated sample displacement can be compared to the theoretical analysis. As previously mentioned, the sample deformed the proper amount in the vertical direction as was prescribed through the boundary condition input. However, for an unconfined compression test the stretch ratio in the principle directions follow the strict relationship represented by Equation 4.4. Using this relationship, the 40% compression the stretch ratio in the radial direction λ_r can be defined as:

$$\begin{aligned} \lambda_r &= \lambda^{-\frac{1}{2}} \\ \lambda_r &= (0.6)^{-\frac{1}{2}} \\ \lambda_r &= 1.291 \end{aligned} \quad (4.12)$$

Using the radial stretch ratio in conjunction with the original sample diameter θ_D it is possible to determine the length of sample expansion in the radial direction Δ_r :

$$\begin{aligned} \Delta_r &= \lambda_r \theta_D - \theta_D \\ \Delta_r &= (1.291)(25.4) - 25.4 \\ \Delta_r &= 7.3914\text{mm} \end{aligned} \quad (4.13)$$

Due to the fact that symmetry was used in the ABAQUS simulation, the total magnitude of the radial expansion needed to be divided by two. Therefore,

the radial expansion displayed in the simulation should be 3.6957mm, which is the magnitude of the displacement shown in Figures 4.1c and 4.1d. This check shows that the settings chosen in the ABAQUS simulation were appropriate for the sample deformation. To further validate the settings, the force vs. stretch ratio has been determined from Equation 4.11. The resultant plot is shown in Figure 4.3. There is a high correspondance between the simulated force profile (Figure 4.2) and the theoretical force profile. In fact, the largest error that occurs through the unconfined compression task is 0.0003N. This shows that the ABAQUS simulation is quite accurate and the parameters that were used in the analysis (such as element type, boundary conditions, material properties, etc.) can be used for the simulation of the whole liver geometry with a high degree of certainty.

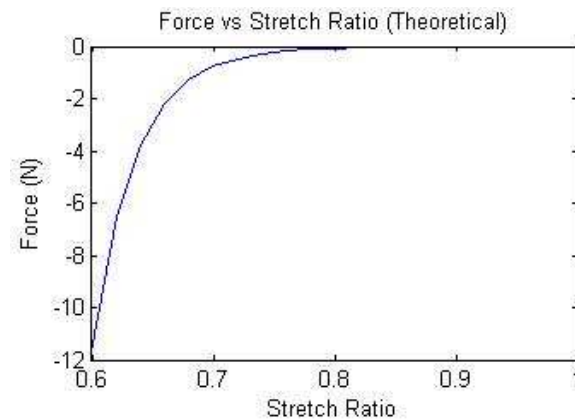


Figure 4.3: Resultant force vs. stretch ratio profile derived from theoretical analysis.

©2009 IEEE.

4.1.1.2 FEA of *Ex Vivo* Tissue Sample

A lobe of soft-tissue was modeled using Pro/Engineer to develop a geometric representation of the tissue sample to be analyzed (Figure 4.4a). The size and shape of the sample was within the range of lobes obtained during the experimental tests (approximately 180cm X 115cm X 40cm at extremes).

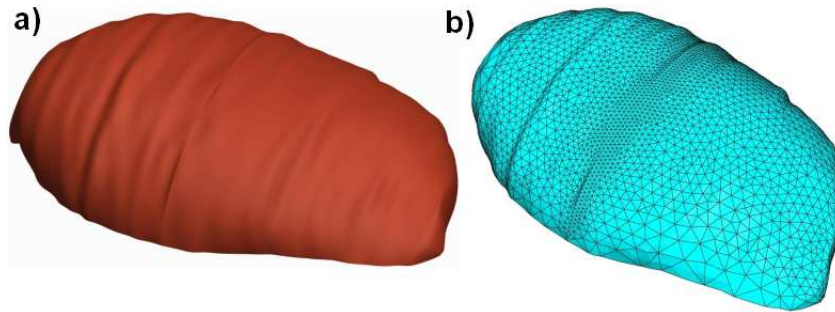


Figure 4.4: a) *Ex vivo* geometry solid model b) *Ex vivo* mesh derived from solid model. ©2009 IEEE.

The geometry was imported into ABAQUS and assigned material properties defined by the Ogden model. It was constrained in all directions along the bottom surface and meshed with 28,820 quadratic tetrahedral elements. The mesh was implemented with an edge bias located along the centerline of the tissue sample along which the probe would be contacting the tissue. This allowed for a higher resolution deformation analysis in the proximity of the probe without the need for refined mesh throughout the entirety of the sample, which would greatly increase the simulation complexity. A 10mm hemispherical probe was then modeled as a rigid body in ABAQUS. The rigid body assumption was utilized due to the fact that a metallic probe will be much stiffer than the soft-tissue. The probe was aligned

perpendicular to the surface of liver in the center of the sample.

A contact analysis was performed whereby the probe was displaced in the vertical direction and introduced displacement conditions on the surface of the liver. The contact between the probe and the tissue was assumed to be frictionless, which mimics surgical procedures where blood would be present between the probe and the tissue resulting in very low frictional effects. The simulation was implemented in two steps. Initially, a uniform gravitational load was applied to the tissue sample. This proved to be an essential step during the analysis because it allowed for the settling of tissue that naturally occurs due to the compliant nature of the material. Without this step, the calculated forces would be much lower than the actual measured values. Finally, the probe was moved into contact with the tissue and continued to a distance of 1.55cm which corresponds to 47.5% strain. The simulation was run in 40 equal steps corresponding to 0.3875mm displacements where the equilibrium position is found at each displacement condition. Upon completion of the analysis, the nodal displacement values and the reaction force applied to the probe can be determined.

The results from the probing simulation procedure are presented in Figure 4.5. It was apparent through analyzing the displacement plots that a great deal of settling had occurred throughout the tissue sample (an average of 0.5cm) during the gravitational loading process as was expected for a soft biological material. Additionally, it was observed that through the probing deformation process, the area in the vicinity of the probe experienced some local deformation and the tissue displacement in the area surrounding the probe was increased. The reaction force on the probe was also determined for each of the 40 steps solved for during the simulation.

This displacement and reaction force information was then used for the formulation of the reality-based, haptics-enabled *ex vivo* probing simulator.

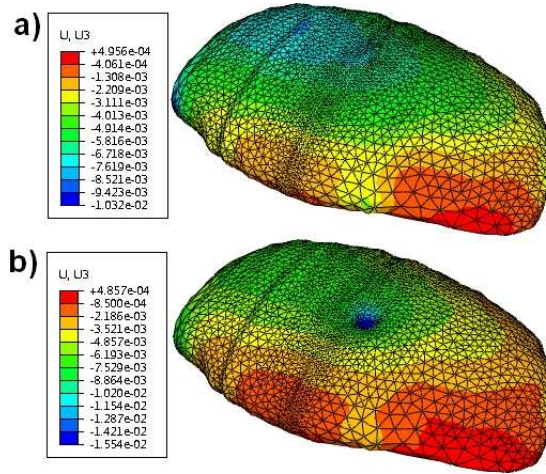


Figure 4.5: Tissue displacement (m) in the vertical direction following the simulation of the a) gravitational loading and b) probing action. ©2009 IEEE.

4.1.2 Implementation of the Simulator

A real-time, haptics-enabled probing simulator was developed using the data collected during the finite element analysis. The simulator consists of a graphical display, implemented through OpenGL programming framework, and a haptic display, through the use of a PHANToM haptic device (SensAble Technologies, Version 1.5A). The structure of the program is graphically depicted in Figure 4.6.

The graphical display is developed by rendering the surfaces of the elements generated in ABAQUS. A collection of the nodal locations and connectivity information at each step of the analysis is read into the OpenGL framework and displayed on the screen. By selecting a step size of 0.3875mm in the finite element analysis,

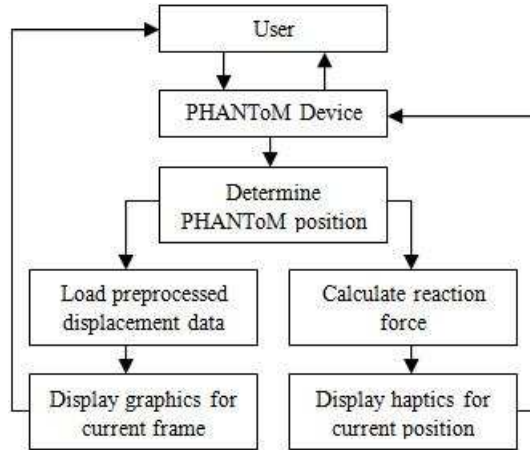


Figure 4.6: Structure of the real-time, haptics-enabled simulator.

the variation between steps is very small and visually approximates a continuous representation of the tissue deformation. This prevents noticeable jumps between states during the probing motion. A probe of the same shape as that used in the finite element analysis is also displayed through the OpenGL framework. Simple lighting conditions were added to the code to enhance the three dimensional aspect of the simulation as well as provide additional depth information that aids in displaying the tissue deformation.

The simulator has been programmed to allow the user to toggle the display of the probe on and off. This allows the user to get the detailed depiction of the tissue deformation without being hindered by the display of the probe. The tissue can also be rotated about all three axes and translated in all directions to provide the user with all possible views of the tissue.

The PHANToM haptic device is the interface between the user and the simulator. By moving the stylus the user is able to control the position of the probe in the simulation. Additionally, the forces acting on the probe are relayed to the user

in real-time to provide the sensation that the user would feel when probing a real piece of tissue. The force data that was collected during the finite element simulation needed to be modified prior to haptic display. Due to the sensitivity of human touch, a simulation using the 40 discrete force states could result in a discontinuous sensation that would not accurately represent the proper tissue response. Therefore the discrete dataset was fit with an exponential curve that relates the reaction force output to the displacement of the probe:

$$F = -0.009991e^{-373.6x} \quad (4.14)$$

F is the reaction force in Newtons and x is the displacement in meters. This model has an R-square value of 0.9998, corresponding to a very close fit to the data generated in ABAQUS. A plot of the function is shown in Figure 4.7 along with the force values at each discrete step. Equation 4.14 is used in the simulation code to determine a continuous representation of the force as the PHANToM stylus is moved.

The detailed ABAQUS simulation is the foundation for the representation in the real-time framework, and the preprocessing aspect allows for the inclusion of many details that would not otherwise be possible. As previously mentioned, simplifications are often made to the material model for real-time simulation purposes. Another method used to reduce the computational complexity is to solve only for the details of the surface deformation in the organ of interest. While this may be a valid approach for some surgical tasks, the loss of information about the inter-

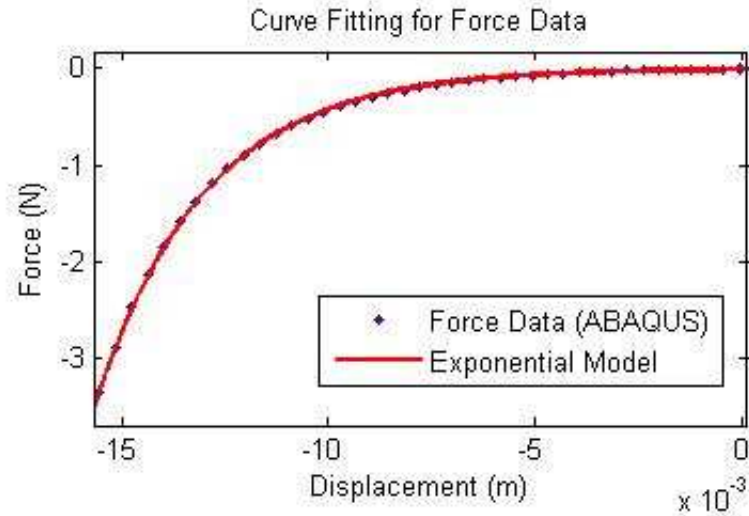


Figure 4.7: Force data and resultant exponential fit derived from ABAQUS simulation.

nal structure may be important for different modes, such as the tracking of tumor movement in response to applied loads. This simulation framework, however, allows for use of the detailed models in a full volumetric deformation study, as can be seen in Figure 4.8. This image shows a cross section of the tissue sample obtained directly from the real-time simulator display. The internal elements can easily be viewed; when the probe loads the surface of the organ, the corresponding loads and displacements are translated into the internal structure to produce a full volumetric response.

The final version of the simulator is successful in displaying the graphics and force feedback to the user in real time. An image of the set-up is shown in Figure 4.9, where a user is controlling the simulation through the PHANToM haptic device. The current simulator provides the framework to efficiently display prepro-

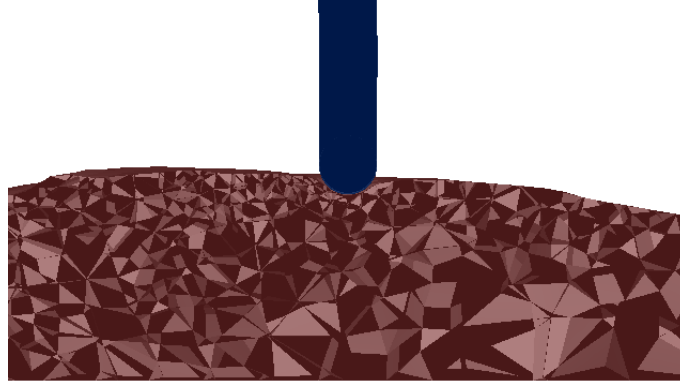


Figure 4.8: Cross section of the tissue sample captured from the real-time simulator display.

cessed displacement and force data obtained from finite element analysis in real-time. This approach greatly increases the accuracy of the tactile sensations and visual deformation when compared to the current trend of mass-spring models commonly being utilized in surgical simulators.

4.2 Preprocessor Based Scalpel Cutting Simulator

The above simulation approach was utilized in a similar manner to develop a real-time, preprocessor based simulation method for scalpel cutting of soft tissue. In this case, an experimentally derived hyperelastic model was again the framework for the soft tissue response. In the cutting simulator, however, the reduced polynomial model derived from *in vivo* testing, $\Psi = \sum_{i=1}^N C_{i0} (\bar{I}_1 - 3)^i + \sum_{i=1}^N \frac{1}{D_i} (J_{el} - 1)^{2i}$, was adopted. The simulator was based on the detailed execution of the ABAQUS based finite element simulation of scalpel cutting presented in Section 3.2.1.

As in the probing simulator, the mesh deformation was updated by loading the



Figure 4.9: Preprocessor based real-time, probing simulator with realistic display and haptic feedback. ©2009 IEEE.

simulation step and connectivity information into the preprocessor code. Additional functionality was added to handle multiple types of elements because tetrahedral, hexahedral and wedge elements were used in the ABAQUS simulation. The haptic loop of the code was also modified for direct display of the force at each simulation step to a PHANTOM Omni haptic device. In this case, the shape of the force profile obtained during the simulation was that of a saw-tooth pattern common to surgical cutting tasks. Therefore, the development of an equation to govern the force values at discrete steps was not an intuitive solution. Instead, the code directly implemented the reaction force acting on the probe obtained from the ABAQUS

simulation steps. When running the preprocessor based simulation, no lags or jumps in the force are detectable to the user other than those intentionally instilled by a local cutting action corresponding to a jump in the saw-tooth profile.

Once again, the simulator is able to display the full volumetric information calculated in the ABAQUS simulation. A depiction of an interior tissue view can be seen in Figure 4.10a. A full view of the entire setup with the user interfacing with the PHANToM Omni haptic device is also seen in Figure 4.10b.

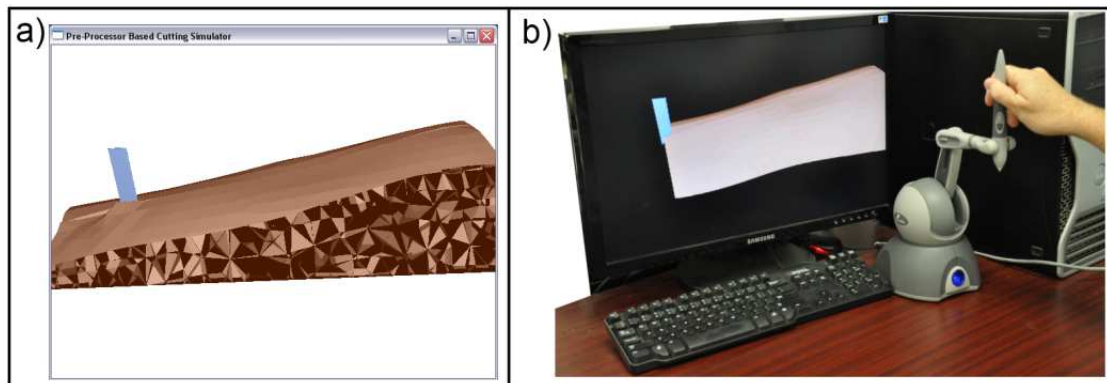


Figure 4.10: a) Internal view and b) system setup for preprocessor based cutting simulator.

The technique presented for the preprocessor based simulators is well suited for surgical procedures with a finite number of tool paths. The preprocessed approach, however, is limited in its scope for general surgical simulation because all desired tool paths must be known prior to constructing the simulator. Thus, the proposed simulator framework would not be advantageous for the majority of complex surgical procedures. To allow for generality in the tool paths, real-time finite element methods must be applied.

4.3 Real-Time Simulation using SOFA Framework

The next logical evolution of the simulation process is to develop a real-time finite element based approach for performing surgical simulation. Many of the aspects of a finite element solver will remain unchanged when altering only the parameters of the underlying material model. To allow for focus on the material model implementation itself, a standard open source framework is proposed for the foundation of the real-time simulator. The Simulation Open Framework Architecture (SOFA) was developed formulated for this reason. It is a highly segmented architecture designed for the development of medical training systems. The SOFA framework presents a method whereby the individual core components of a medical simulation are independently defined to allow for easy implementation of various configurations. For example, there are subsections of the code responsible for handling the underlying material model, the collision detection, integration schemes, mappings, haptics, parallel processing, OpenGL based display, etc. Each of the subsections interact with one another to create a complete simulation scheme, however the options for an individual component can easily be modified without effecting the other aspects. For example, the collision models utilized in a simulator can be changed from spheres to triangular meshes or distance fields without modifying any other aspects of the simulation code.

To this end, the SOFA framework provides an excellent starting point for the desired real-time probing simulator. The framework contains the implementation of many aspects that will be unaffected by the type of material model used, such as the

method of collision detection, graphical user interface and haptic implementation. This allows for modification of the material model segment without necessitating changes the other sections.

When the framework was first defined, it was configured to solve linear elastic finite element problems in real-time with geometric nonlinearities [60]. In this case, the approach decomposed the deformation of each element into a rigid motion and a pure deformation to provide the capability to solve at real-time rates without parallel processing. The linear elastic materials were utilized to create a general deformation simulation of liver. Another linear based finite element method has also been used to simulate respiratory motion of the lungs in real-time [42]. More recently, a non-linear material based finite element simulation has also been conducted using the SOFA framework [15]. In this study, the Total Lagrangian Explicit Dynamics (TLED) approach was used to simulate a transversely isotropic visco-hyperelastic model. In addition, the system was modified to allow for execution on a graphics processing unit (GPU) through CUDATM programming. This study shows that the SOFA framework is not only well conditioned for non-linear finite element simulation, it can also be used along with GPU implementation to further increase the system response. Finally, work on the haptic side has included the ability to utilize haptic devices such as the PHANToM (SensAble) in the simulations [73].

4.3.1 Theory of TLED Algorithm

The majority of the changes required for solving the finite element problem in a real-time setting focused on updating the TLED approach to work with more more complex hyperelastic models. The TLED algorithm was modified to include the reduced polynomial model derived from the *in vivo* experimentation. The previous work, and portions of code obtained contained in SOFA, utilized the Neo-Hookean model. Changes were required to simulate the more accurate and complex reduced polynomial mode. A detailed description of the TLED algorithm is presented in [55], and an overview of the important concepts will be presented here.

The TLED algorithm utilizes the total Lagrangian method of solving finite element problems. This allows the computations to proceed by relating all variables to the original model configuration. By referencing the original configuration, as opposed to the current configuration used in the updated Lagrangian formulation, many of the variables required may be precomputed. Also, the assembly of a global stiffness is not required, which greatly increases the speed of the calculations.

Using the approach presented in [55] and [78], the TLED algorithm can be described in two steps. First, a pre-computation step is conducted whereby the variables referenced to the original configuration which will remain constant throughout the simulation are calculated. Specifically:

- Load mesh information
- Calculate spatial derivatives of shape functions
- Calculate linear strain-displacement matrix B_L for each element

- Calculate mass matrix

Second, a time-loop phase is initiated, within which all components required to update the nodal displacements are determined. The details of the second step are as follows:

- Apply updated nodal load and boundary conditions
- Calculate deformation gradient \mathbf{F} for each element
- Calculate full strain-displacement matrix \mathbf{B}_L^T for each element ($\mathbf{B}_L^T = \mathbf{B}_L \mathbf{F}^T$)
- Calculate second Piola-Kirchoff stress \mathbf{S} for each element
- Calculate total force acting on each node $\tilde{\mathbf{f}}$ for each element
- Sum all nodal force components to achieve total force \mathbf{f}
- Calculate new nodal displacements using central difference method

Specifically, the element nodal force components, $\tilde{\mathbf{f}}$, are found from:

$$\tilde{\mathbf{f}} = \int \mathbf{B}_L^T \mathbf{S} dV$$

where V is the element volume. This expression can be modified for specific types of elements, namely the reduced integration linear hexahedral and linear tetrahedral elements to become:

$$\tilde{\mathbf{f}}_{HEX} = 8 \mathbf{B}_L^T \mathbf{S} \det(\mathbf{J}) \quad (4.15)$$

$$\tilde{\mathbf{f}}_{TET} = V \mathbf{B}_L^T \mathbf{S} \quad (4.16)$$

where \mathbf{J} is the Jacobian matrix.

4.3.2 Second PK Stress for Four-Term Reduced Polynomial

The majority of the computational complexity arises in the determination of the second Piola-Kirchoff (PK) stress for each individual element. In this algorithm, and code, the methods presented in [34] were used to split the strain energy density function, Ψ , into two components, the volumetric and isochoric components:

$$\Psi(\mathbf{C}) = \Psi^{VOL}(J) + \Psi^{ISO}(\bar{\mathbf{C}})$$

where J is the Jacobian ($J = \det \mathbf{F}$), \mathbf{C} is the right Cauchy-Green deformation tensor ($\mathbf{C} = \mathbf{F}\mathbf{F}^T$) and $\bar{\mathbf{C}}$ is the modified right Cauchy-Green deformation tensor ($\bar{\mathbf{C}} = J^{-\frac{2}{3}}\mathbf{C}$). This expression can also be presented in terms of the invariant as:

$$\Psi(\mathbf{C}) = \Psi^{VOL}(J) + \Psi^{ISO}(\bar{I}_1, \bar{I}_2)$$

where $\bar{I}_1 = \text{tr} \bar{\mathbf{C}}$ and $\bar{I}_2 = \frac{1}{2} \left[(\text{tr} \bar{\mathbf{C}})^2 - \text{tr}(\bar{\mathbf{C}}^2) \right]$ are the first and second invariant of $\bar{\mathbf{C}}$. Using the above representation of the strain energy density function, the second PK stress can be decomposed into the volumetric and isochoric components as well resulting in the following representation:

$$\mathbf{S} = 2 \frac{\partial \Psi(\mathbf{C})}{\partial \mathbf{C}} = \mathbf{S}_{VOL} + \mathbf{S}_{ISO} \quad (4.17)$$

The two components of the second PK stress can be analyzed individually. Beginning with the volumetric response:

$$\mathbf{S}_{VOL} = 2 \frac{\partial \Psi^{VOL}(J)}{\partial \mathbf{C}}$$

$$\mathbf{S}_{VOL} = 2 \frac{\partial \Psi^{VOL}(J)}{\partial J} \frac{\partial J}{\partial \mathbf{C}}$$

where

$$\frac{\partial J}{\partial \mathbf{C}} = \frac{1}{2} J \mathbf{C}^{-1}$$

so S_{VOL} becomes:

$$\mathbf{S}_{VOL} = J \frac{\partial \Psi^{VOL}(J)}{\partial J} \mathbf{C}^{-1} \quad (4.18)$$

For the four-term reduced polynomial model, the volumetric component of the strain energy density function is:

$$\Psi^{VOL}(J) = \sum_{i=1}^4 \frac{1}{D_i} (J_{el} - 1)^{2i} \quad (4.19)$$

In this case, $J_{el} = J$ as defined by ABAQUS due to the fact that there is no thermal expansion present. Hence:

$$\begin{aligned} \frac{\partial \Psi^{VOL}(J)}{\partial J} &= \frac{\partial}{\partial J} \left(\sum_{i=1}^4 \frac{1}{D_i} (J - 1)^{2i} \right) \\ \frac{\partial \Psi^{VOL}(J)}{\partial J} &= \sum_{i=1}^4 \frac{2i}{D_i} (J - 1)^{2i-1} \end{aligned}$$

So, the volumetric component of the second PK stress for a four-term reduced polynomial becomes:

$$\mathbf{S}_{VOL} = J \left(\sum_{i=1}^4 \frac{2i}{D_i} (J - 1)^{2i-1} \right) \mathbf{C}^{-1} \quad (4.20)$$

Next the isochoric component of the second PK stress can be derived for the reduced polynomial model. Using the approach presented in [34] again, the isochoric component can be derived from:

$$\mathbf{S}_{ISO} = 2 \frac{\partial \Psi^{ISO}(\bar{\mathbf{C}})}{\partial \mathbf{C}}$$

or

$$\mathbf{S}_{ISO} = J^{-\frac{2}{3}} Dev(\bar{\mathbf{s}}) \quad (4.21)$$

where

$$Dev(\bullet) = (\bullet) - \frac{1}{3} [(\bullet) : \mathbf{C}] \mathbf{C}^{-1} = (\bullet) - \frac{1}{3} tr [(\bullet)^T \mathbf{C}] \mathbf{C}^{-1} \quad (4.22)$$

and

$$\bar{\mathbf{s}} = 2 \frac{\partial \Psi^{ISO}(\bar{\mathbf{C}})}{\partial \bar{\mathbf{C}}} = \bar{\gamma}_1 \mathbf{I} + \bar{\gamma}_2 \bar{\mathbf{C}}$$

with $\bar{\gamma}_1$ and $\bar{\gamma}_2$ defined as:

$$\begin{aligned} \bar{\gamma}_1 &= 2 \left(\frac{\partial \Psi^{ISO}(\bar{I}_1, \bar{I}_2)}{\partial \bar{I}_1} + \bar{I}_1 \frac{\partial \Psi^{ISO}(\bar{I}_1, \bar{I}_2)}{\partial \bar{I}_2} \right) \\ \bar{\gamma}_2 &= -2 \frac{\partial \Psi^{ISO}(\bar{I}_1, \bar{I}_2)}{\partial \bar{I}_2} \end{aligned}$$

For the four-term reduced polynomial case, the isochoric component of the strain energy density function is defined as:

$$\Psi^{ISO}(\bar{I}_1, \bar{I}_2) = \sum_{i=1}^4 C_{i0} (\bar{I}_1 - 3)^i \quad (4.23)$$

For the above reduced polynomial model, the $\bar{\gamma}_1$ and $\bar{\gamma}_2$ coefficients can be derived as follows:

$$\begin{aligned} \bar{\gamma}_1 &= 2 \left(\frac{\partial \Psi^{ISO}(\bar{I}_1, \bar{I}_2)}{\partial \bar{I}_1} + \bar{I}_1 \frac{\partial \Psi^{ISO}(\bar{I}_1, \bar{I}_2)}{\partial \bar{I}_2} \right) \\ &= 2 \frac{\partial}{\partial \bar{I}_1} \left(\sum_{i=1}^4 C_{i0} (\bar{I}_1 - 3)^i \right) \\ &= 2 \left(\sum_{i=1}^4 i C_{i0} (\bar{I}_1 - 3)^{i-1} \right) \\ \bar{\gamma}_2 &= -2 \frac{\partial \Psi^{ISO}(\bar{I}_1, \bar{I}_2)}{\partial \bar{I}_2} \\ &= 0 \end{aligned}$$

resulting in the following version for $\bar{\mathbf{s}}$:

$$\bar{\mathbf{s}} = 2 \left(\sum_{i=1}^4 i C_{i0} (\bar{I}_1 - 3)^{i-1} \right) \mathbf{I} \quad (4.24)$$

Using 4.22 and 4.24, the expression for $Dev(\bar{\mathbf{s}})$, particular to the four-term reduced polynomial, can be determined:

$$\begin{aligned} Dev(\bar{\mathbf{s}}) &= \bar{\mathbf{s}} - \frac{1}{3} tr[\bar{\mathbf{s}}^T \mathbf{C}] \mathbf{C}^{-1} \text{ but } \bar{\mathbf{s}} = \bar{\mathbf{s}}^T \text{ since } \bar{\gamma}_1 \text{ is a scalar} \\ &= \bar{\gamma}_1 \mathbf{I} - \frac{1}{3} tr[(\bar{\gamma}_1 \mathbf{I}) \mathbf{C}] \mathbf{C}^{-1} \\ &= \bar{\gamma}_1 \left(\mathbf{I} - \frac{1}{3} tr[\mathbf{I} \mathbf{C}] \mathbf{C}^{-1} \right) \\ &= \bar{\gamma}_1 \left(\mathbf{I} - \frac{1}{3} tr[\mathbf{C}] \mathbf{C}^{-1} \right) \\ &= \bar{\gamma}_1 \left(\mathbf{I} - \frac{1}{3} I_1 \mathbf{C}^{-1} \right) \\ &= \left(\sum_{i=1}^4 i C_{i0} (\bar{I}_1 - 3)^{i-1} \right) \left[\mathbf{I} - \frac{I_1}{3} \mathbf{C}^{-1} \right] \end{aligned}$$

which, used in conjunction with 4.21, results in the final expression for the isochoric component of the second PK stress:

$$\mathbf{S}_{ISO} = J^{-\frac{2}{3}} \left[\sum_{i=1}^4 i C_{i0} (\bar{I}_1 - 3)^{i-1} \left(\mathbf{I} - \frac{I_1}{3} \mathbf{C}^{-1} \right) \right] \quad (4.25)$$

Finally, a full representation of the second Piola-Kirchoff stress can be defined using 4.20 and 4.25 as:

$$\mathbf{S} = J \left(\sum_{i=1}^4 \frac{2i}{D_i} (J-1)^{2i-1} \right) \mathbf{C}^{-1} + J^{-\frac{2}{3}} \left[\sum_{i=1}^4 i C_{i0} (\bar{I}_1 - 3)^{i-1} \left(\mathbf{I} - \frac{I_1}{3} \mathbf{C}^{-1} \right) \right] \quad (4.26)$$

All calculations specified above for the second PK stress terms are required for each element. In addition, the deformation gradient, full strain-displacement matrix and the total force acting on each node are all required for each individual

element. As all of these terms can be generated directly from manipulating the nodal displacement for each element independently, the code can be generated in a manner in which all elements are processed in parallel. This is possible due to the fact that everything is computed for individual elements without need for passing information between elements during the TLED calculations. Therefore, these sections of the code were developed using CUDATM programming to run on a graphics processing unit (GPU). This greatly increases the computational efficiency as all elements can be analyzed simultaneously instead of sequentially. A copy of the TLED code implementing the four-term reduced polynomial response is available in Appendix A.

4.3.3 Comparison of SOFA Implementation to ABAQUS

After the portions of the TLED algorithm were modified, a simulation implemented in SOFA was compared to an identical analysis solved via ABAQUS to validate the accuracy of the SOFA approach. In this case, a cube was instantaneously subjected to a gravitational force and the transient response was recorded using both simulation systems for comparison.

The geometry of the test sample consisted of a cube measuring 10cm along each side. The cube was split into a uniform mesh of 1,000 reduced integration hexahedral elements (C3D8R) with 1,331 corresponding nodes. The material model used was that of the *in vivo* based, four-term reduced polynomial presented in Section 2.2.6. A dynamic, explicit approach was used in both simulations with a time step of

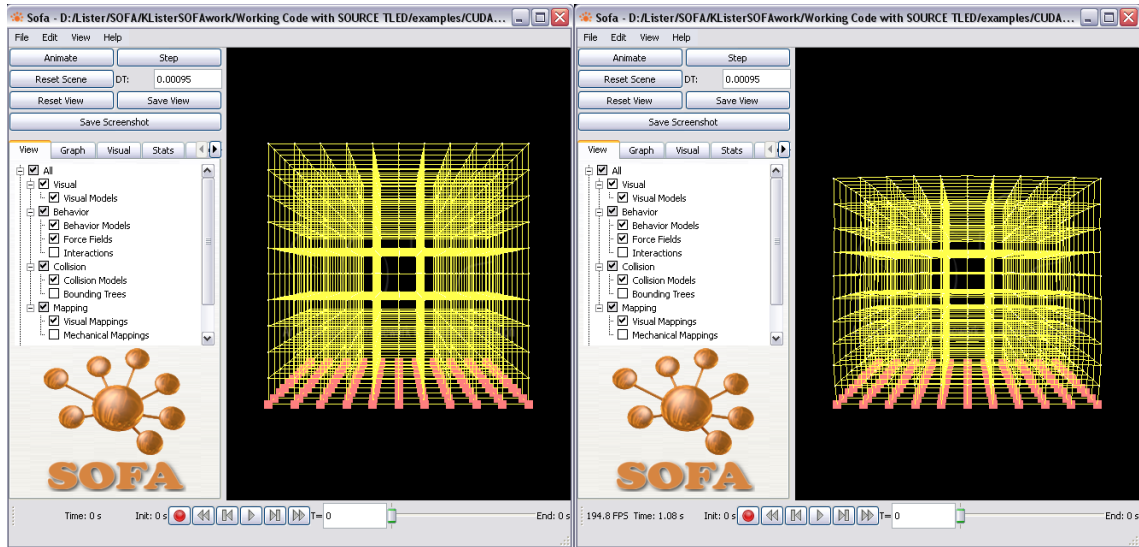


Figure 4.11: Results for SOFA simulation of a cube under gravitational load.

0.00095s. Displacement boundary conditions were applied to the nodes along the bottom face of the cube to prevent motion in any direction. Rayleigh damping was applied with $\alpha = 25$ and $\beta = 0$. Finally a gravitational load was applied instantaneously at the first time step with a magnitude of -9.81N . The simulation was allowed to progress for a total of 1s until the new equilibrium condition was achieved. A depiction of the cube deformation resultant from the SOFA simulation is presented in Figure 4.11. In the figure, the mesh is displayed in wire-frame with the yellow elements while the pink squares represent the nodes along the bottom surface where the static displacement condition was applied. As expected, the mesh compressed a significant amount under the influence of the gravitational load. This was similar to the effects discovered during the model generation testing due to the highly compliant nature of soft tissue.

Using ABAQUS as the simulation engine, a similar response can be observed

(Figure 4.12). A contour plot of the displacement magnitude is displayed where it can be seen that the bottom surface remained stationary and the top of the cube moved approximately 12mm due to the gravitational load.

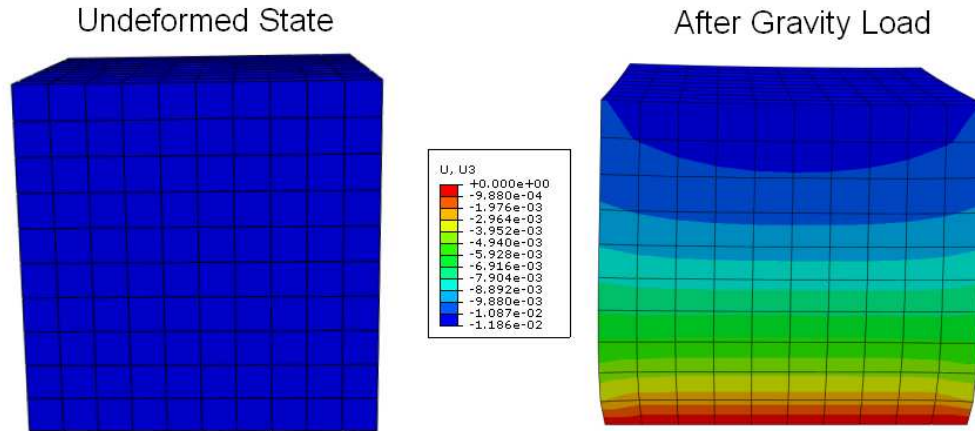


Figure 4.12: Results for ABAQUS simulation of a cube under gravitational load.

From the two simulations, data was extracted for individual node locations to compare the tissue response throughout the entirety of the motion induced by the gravitational loading. A depiction of the nodal response is shown in Figure 4.13. Displacement values in the X,Y and Z directions as well as the total magnitude of the nodal displacement are compared. The two different simulators showed similar trends in the tissue response, with slight differences as seen in the figure. The ABAQUS simulation resulted in a final displacement of 11.85mm while the SOFA simulation yielded a displacement value of 11.69mm. This results in an error of 1.35% for the steady state response after equilibrium is reached. The largest error observed is during the transient response at the peak of the tissue displacement. This resulted in difference of 0.534mm for a short moment of time. As the simulation progressed, the error stabilized around the steady state point. Differences are

expected as SOFA utilizes the total Lagrangian method and ABAQUS is based on the updated Lagrangian method, however the errors are small and within the limits that would be noticeable to the operator of the simulator.

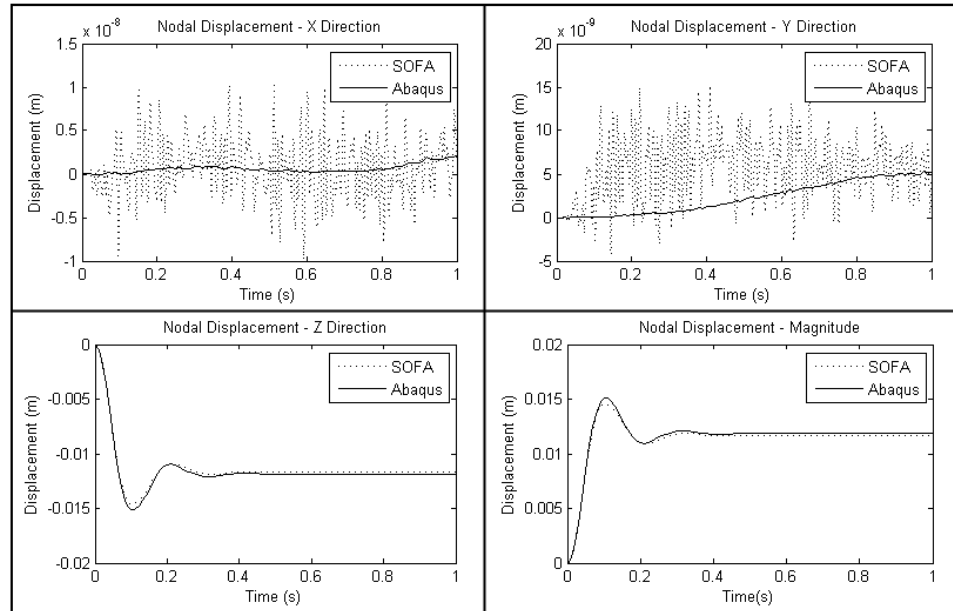


Figure 4.13: Comparison of the SOFA and ABAQUS simulations.

4.3.4 Effect of Reduced Polynomial on SOFA Frame Rate

With the accuracy of the reduced polynomial model implemented using the TLED algorithm established, the next step was to determine the change in computation time required for the additional complexity of utilizing the hyperelastic representation of the tissue. To determine the speed decrease an example code distributed with the SOFA framework for testing the functionality of the TLED algorithm was utilized. This code also simulates the response of a cube to gravitational loading, but only the nodal response is displayed. In this test, the speed of

the simple Neo-Hookean model distributed with the code was compared to that of the more accurate four-term reduced polynomial. Two units of measure were used for comparison of the simulation speed. First, the average frequency at which the entire code runs is calculated for each of the two models. Then an investigation into the time required to implement the modified TLED portion of the code is observed.

A study of the frame rate at which the simulation as a whole runs will determine the ability to update both the display and haptic feedback in real-time. The simulation frame rate will be discussed in detail in the next few sections when the complexity of the task increases, however they will be presented here as well for comparison. The simulation computer used throughout this study consisted of a 2.33GHz Dual Core Processor, 2GB RAM, running Windows XP with a NVIDIA Quadro 570 programmable graphics card. This computer configuration is far from optimal from the standpoint of computational ability, but it provides a starting point for analysis into the real-time simulation of medical tasks. For the most simplistic simulation possible, the deformation of a unit cube (1000 nodes), the simulator based on the Neo-Hookean model is able to run at an average speed of 301Hz with our current computer configuration. When switching to the more complex reduced polynomial the update rate reduces slightly to 283Hz.

A more detailed investigation can be conducted by using some of the evaluations metrics built into the SOFA code. A diagnostic tool allows for the display of the total time taken for each component in the simulation during each time step or over the entirety of the simulation. An image of the diagnostic tool as well as the simulation screen shot for both the Neo-Hookean and reduced polynomial model are

included in Figure 4.14. Through this analysis, the average time required to compute the TLED portion of the code was determined. In the figure, this correlates to item number 5, `CudaHexahedronTLEDForceField(FEM)`, and the cyan color on the pie chart breakdown of total step time. The average time required for the TLED code in the Neo-Hookean case is 0.1186ms. The reduced polynomial implementation does increase the total amount of time required for the TLED algorithm, however it is a very small increase to 0.1205ms. This is a total increase of 0.0019ms per simulation step. When it comes to simulation accuracy, the reduced polynomial model is far superior with regard to simulation accuracy, and as shown here, the increase in computational complexity is minimal.

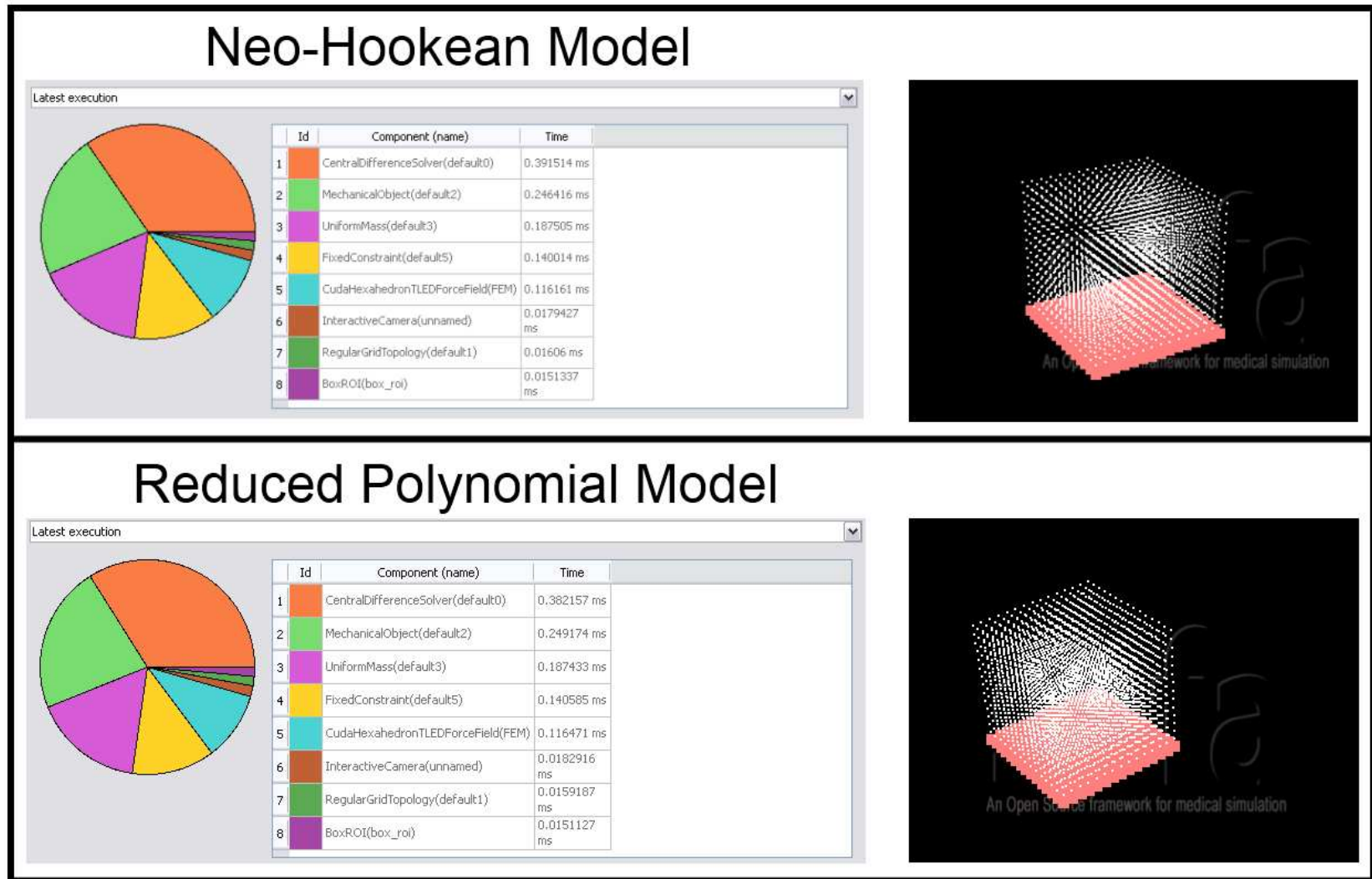


Figure 4.14: Comparison of the simulation speeds for the Neo-Hookean and reduced polynomial models.

4.3.5 Probing Simulator with SOFA Framework

After the accuracy and the speed of the reduced polynomial version of the TLED code was validated, the next step was to develop a simulation system for the probing of soft tissue. The first part of this process was the development of an appropriate geometry for the organ of interest. Throughout this study all work was performed on the liver, therefore a three dimensional model of the liver was developed and meshed for use in this real-time simulator. The mesh consisted of 708 reduced integration hexahedral elements (C3D8R) with a total of 1010 nodes. This mesh was then imported into the SOFA code through use of a mesh loader. As previously stated, the SOFA framework was built in a modular manner to allow for versatility when developing a surgical simulator. Therefore much of the functionality required to develop the probing simulator was used directly. The mesh loader acted as a means to load the nodal locations and connectivity information required for the full definition of the mesh characteristics. Using this information, along with a fixed boundary condition module, simulation of the liver deformation due to gravity was achieved.

A screen shot of the simulator including the realistic liver shape is displayed in Figure 4.15a. Not only has the shape been updated, but the surface of the model has been skinned to provide a more realistic display. The wire frame mesh is included and displayed in yellow in addition to the organ surface. Also, the pink boxes along the bottom edge represent the fixed boundary condition applied along the bottom surface of the organ. The inclusion of the additional sections of code required to

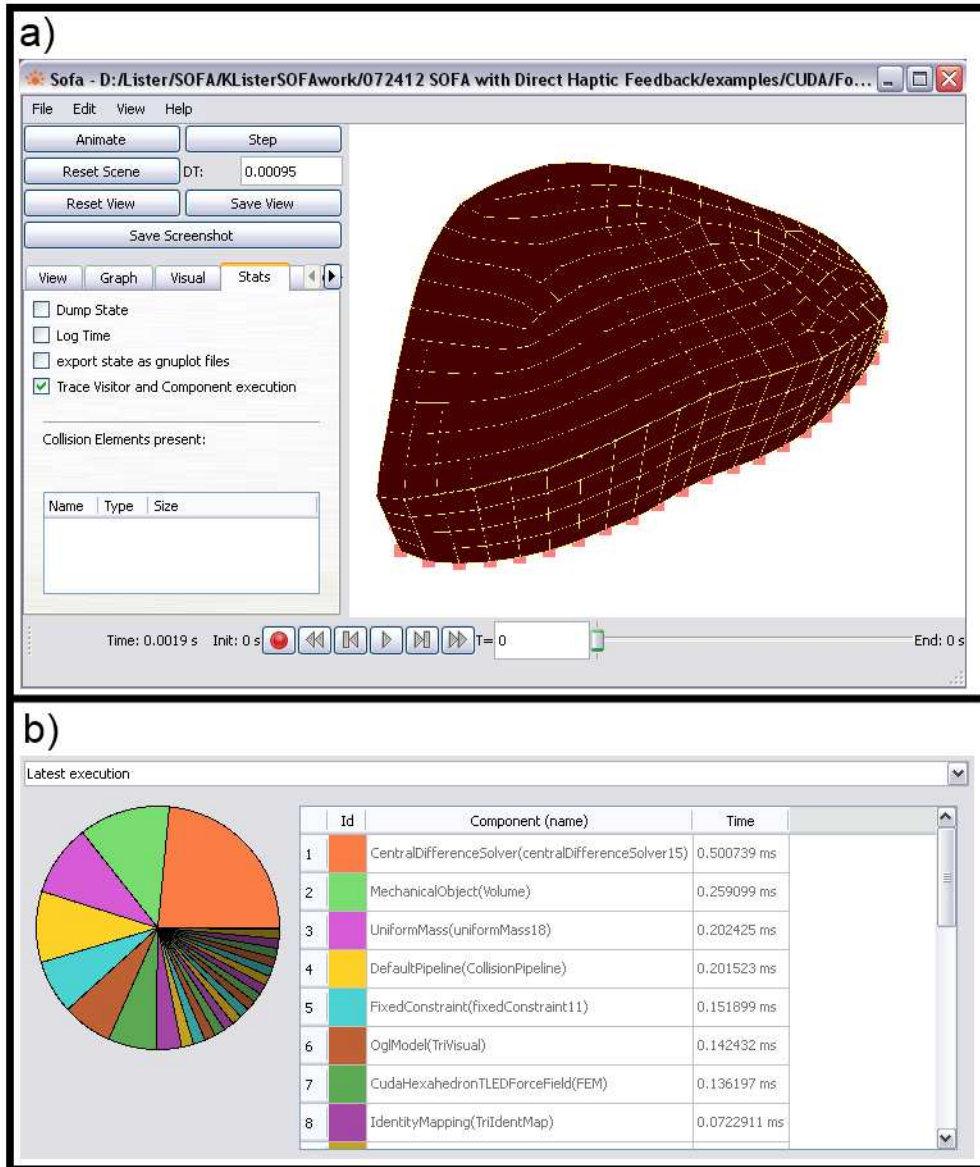


Figure 4.15: a) Screen shot of simulator with realistic geometry and b) performance characteristics with realistic geometry.

set up this version of the simulator do come at the expense of the simulation speed. Using the same methods as the previous examples, the timing for the new simulator results in an average refresh rate of 234 frames per second. Additionally, utilizing a full hexahedral mesh representation of the liver, instead of simply using the node locations as in the speed test, proves to slow down the TLED loop within the code to 0.1298ms. Despite being somewhat slower than the previous simple example, the user is still unable to perceive any lag in the display as it is updated at well above the 30 frame per second requirement.

To make the simulation appear more realistic, additional components were added to the scene around the liver organ (Figure 4.16). A skeleton, with the breast plate partially removed to grant complete access to the liver was included along with a components resembling a human torso. The additional components greatly enhance the feel of the simulator and aid in the orientation of the user to the shape, size and orientation of the liver. However, it comes at the cost of computational speed. The average update rate when adding the reference geometry drops to 170 frames per second. In this case, the additional computation time comes from displaying the new components and the average speed of the TLED implementation loop remains similar at 0.1290ms.

The next step was to add the functionality required for the probing action. This consisted of developing the probe geometry, including it in the scene, controlling the probe position, establishing the collision parameters required to simulate the tool-tissue interaction, and relaying the haptic feedback to the user. The probe itself was developed in a computer aided design (CAD) package and saved as a

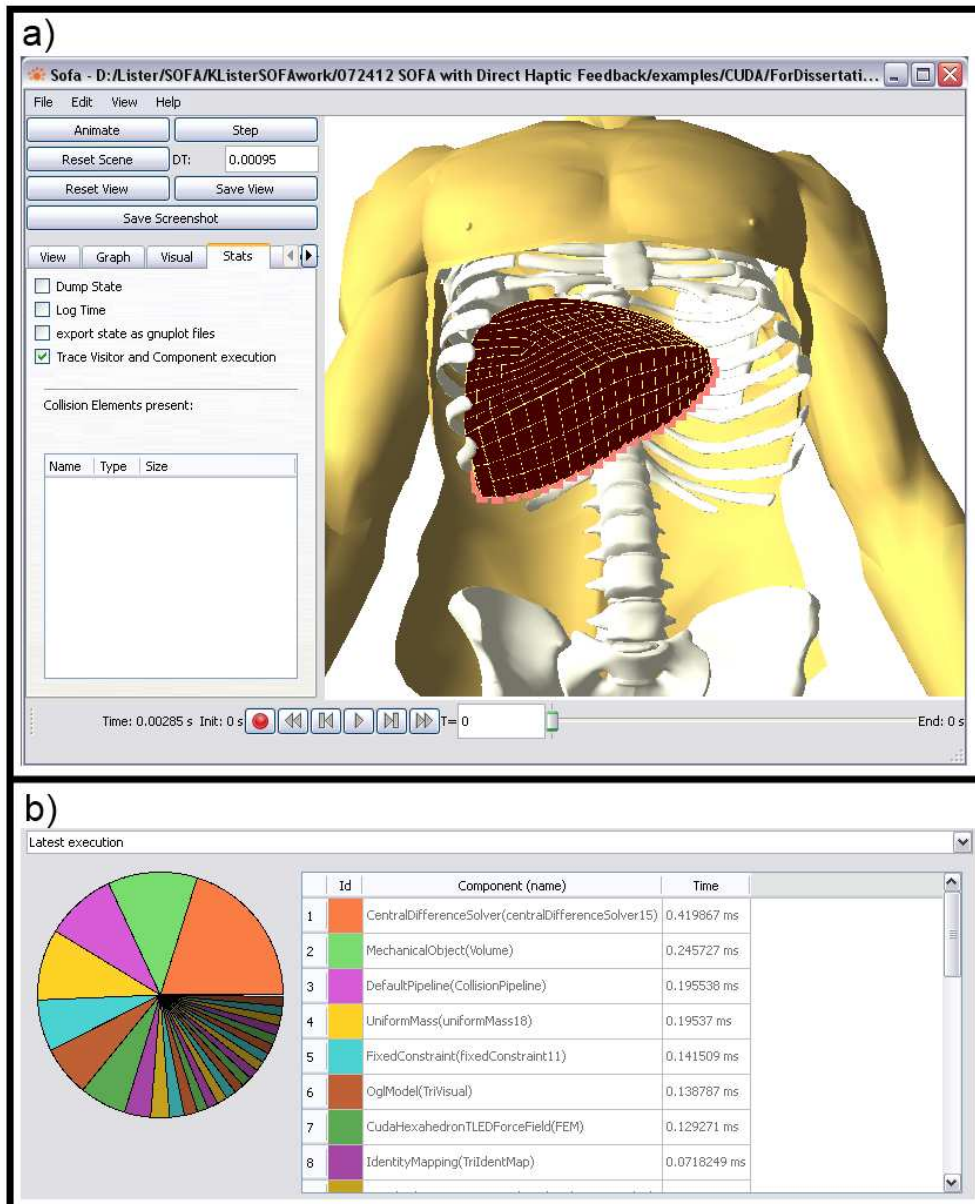


Figure 4.16: a) Screen shot of simulator with additional components and b) performance characteristics with additional components.

Wavefront object file. The Wavefront object file was directly imported into the scene through the mesh loader built into the SOFA framework. Once loaded into the simulation, the probe location was set to be controlled through the use of a PHANToM Omni haptic device. The PHANToM was linked to the code and set up similar to a mouse input where the location and orientation of the stylus was mapped to the location and orientation of the probe within the simulator display. The linking of the PHANToM Omni to the virtual probe provided the means for the user to interact with and control the location of the probe in a logical manner. The haptic device was programmed to align the physical stylus orientation to that of the probe so the movements on the screen directly follow the users input.

The development of the interaction between the probe and the tissue was a more complex procedure. Collision parameters must be added to the probe and the tissue independently. After the collision models were implemented, the rules for contact between the two were added which govern the tool-tissue interaction. The contact definition used for the probe was a spherical based collision algorithm which was added to each node on the probe geometry. The main parameter utilized in spherical based contact in the SOFA framework was the diameter of the sphere at each nodal location. For this surgical simulator, the sizes of the spheres were chosen so they were large enough to slightly overlap one another. This assured the entire probe surface that would interact with the liver was covered to achieve proper contact. The distribution of the spheres can be seen in Figure 4.17.

When determining the collision definitions for the liver model, some additional aspects must be considered. First, to speed up the simulator it is important to note

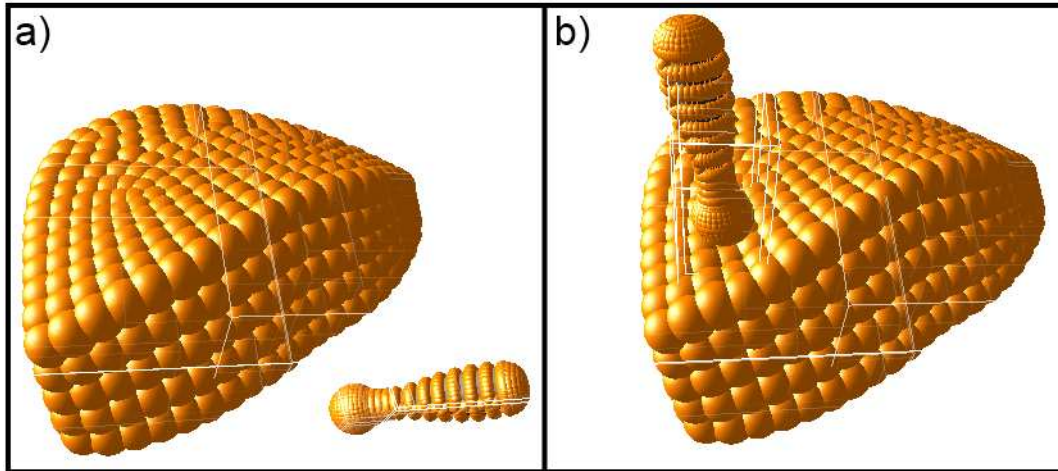


Figure 4.17: a) Contact model setup and b) contact between probe and tissue surface.

that in the case of a probing simulator the tool-tissue contact region will always be on the surface of the liver. Therefore, the collision model should only be defined on the surface to minimize the additional computations added to the simulator as a whole. This approach was utilized in drawing the surface representation of the liver. The hexahedral element can be automatically segmented into quadrilaterals representing the liver surface and then split into triangles for display purposes. Ideally, the triangles representing the surface of the liver would then be used for a surface based collision response. This would provide the support needed to analyse the collision response between the spheres on the probe and the corresponding organ surface. However, the functionality required for more complex collision models, such as the triangular surface interaction, has not yet been achieved in SOFA for codes using GPU acceleration. Since the TLED algorithm implemented on the GPU is essential to achieve the update rates required for a real-time simulator, the more

sophisticated contact model can not be used. Therefore, the collision definition for the liver was also defined by using a spherical approach as shown in Figure 4.17. Using this approach, the collision spheres were attached to each node, including the interior nodes.

The effect that the collision models had on the performance of the overall simulation was drastic compared to the other aspects of the code. When the collision models were defined in the code, a large decrease in the simulator frame rate was observed. Even when the probe was not in contact with the tissue, while the code was only working to verify if contact had occurred, the simulation rates dropped to 137 and 83 frames per second for the liver only and whole body simulation respectively. This represented approximately one half of the speed observed without the collision definitions. However, it was still sufficient for graphical display.

An even larger disturbance was observed when a contact state existed between the probe and the tissue. In this case, additional computations were required to determine the effect of the collision response and impose the displacements to the effected areas. Figures 4.18 and 4.19 show a depiction of the simulator and corresponding performance characteristics for both the liver only and full graphical display cases. As can be seen, the spherical contact models did interact as intended. Yet this functionality came at a very high cost to the frame rate. In the simulation containing only the liver, the update frequency dropped to a range of 23-50 frames per second, depending on the severity of the probe penetration depth. When the torso representation was included the range dropped to 17-42 frame per second. This decrease moved the simulation, in both cases, to the range where visual artifacts

may be noticeable to the user if the update rate drops below 30 frames per second.

Looking into the details during the contact phase (Figures 4.18b and 4.19b), the reason for the drastic slow down was evident. The collision pipeline in both cases fell with the 20-30ms range. This was approximately thirty times larger than the next longest component in the SOFA based simulation. A loop in the code running at 25ms would limit the frame rate to a maximum of 40Hz. Adding the fact that many additional computations were required in addition to the collision pipeline, the ability to simulate the task in real-time was questionable.

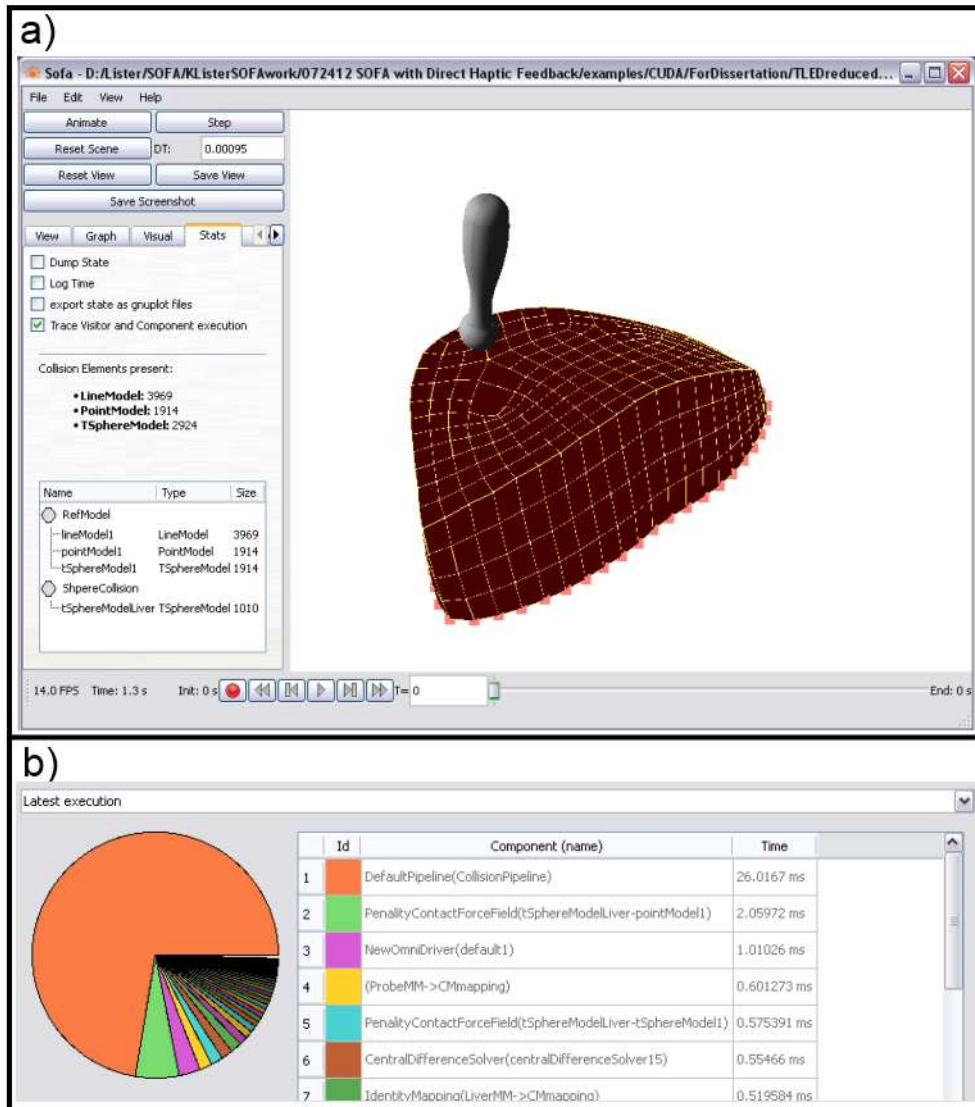


Figure 4.18: a) Simulation with collision response for liver only and b) performance characteristics with collision response for liver only.

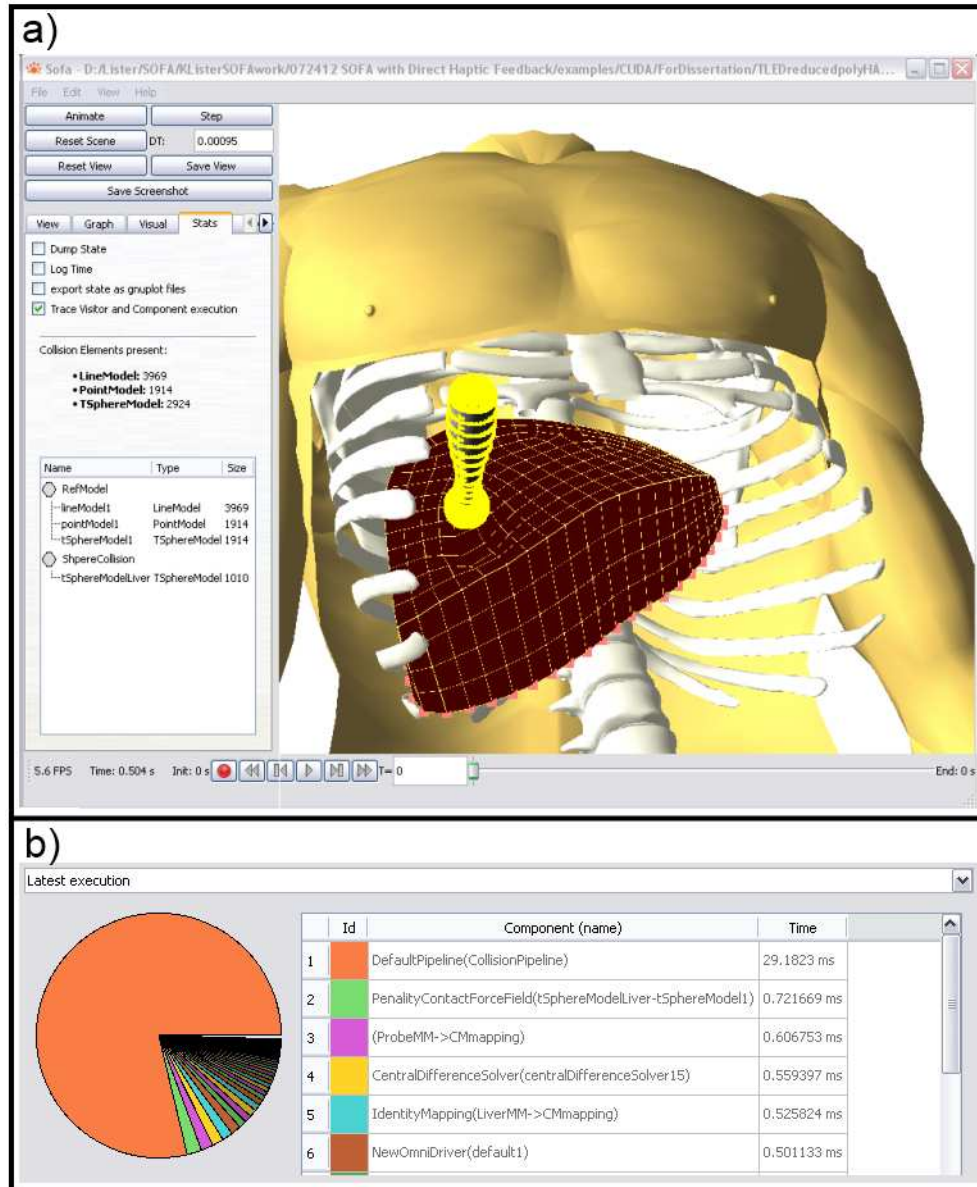


Figure 4.19: a) Simulation with collision response with torso and b) performance characteristics with collision response with torso.

In addition to the change in computation time induced by the simulation of the collision response, the accuracy of the tool-tissue interaction was also evaluated. The reaction force acting on the probe due to the tool-tissue interaction for a standard probing task was recorded and compared to the previous *in vivo* experiments (Figures 4.20). As shown in the figure, the response in the SOFA simulation resulted in a stiffer tissue response and a higher force on the probe. This inaccuracy may have been caused by errors induced by the spherical collision definition. Due to the sphere-to-sphere based setup, the spheres representing the tissue surface must be large enough to overlap. Without some overlapping of the spheres, the probe pushed the tissue spheres apart until it penetrated into the organ. With the selection of spheres of this size, an accurate representation of the surface geometry was not possible. In addition, due to the fact that the spheres were required to overlap to prevent penetration into the tissue, an additional force is imparted to each tissue node defined by the collision between the neighboring nodes. This may result in increasing the reaction force observed on the probe. As the probing occurs the tissue is compressed from the probe from above and the internal collision spheres from below, resulting in additional compression and higher reaction forces. To prevent the apparent increase in tissue stiffness, a more appropriate collision response is required.

The final aspect to add to the probing simulator was the haptic response. When contact between the probe was established, the reaction forces acting on the probe were relayed directly to the PHANToM Omni haptic device. The forces were only calculated when the probe makes contact with the tissue and the simulator

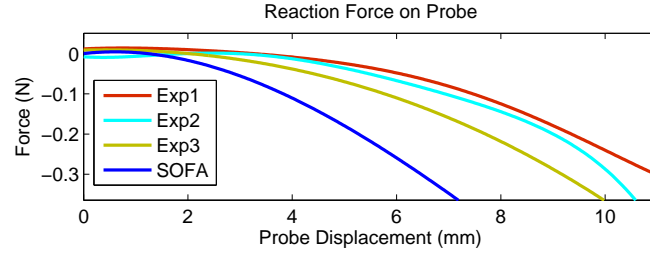


Figure 4.20: Reaction force acting on probe during SOFA simulation and *in vivo* experiments.

began slowing down below the 50Hz range. This resulted in the display of unwanted and inaccurate vibrations to the haptic device. In order to operate properly, the update rate for real-time haptic simulation must be greatly increased.

Overall, the SOFA based simulation was able to accurately represent the deformation of tissue governed by the reduced polynomial model, as shown by the SOFA-ABAQUS comparison study. Implementation of realistic geometry for both the liver shape and reference information in the body was possible without significant detriment to the overall simulator performance. However, the addition of the collision detection algorithms to define the contact between the probe and the soft tissue had a negative effect on both the speed and accuracy of the simulated response. Details of the limitations of the SOFA approach are presented in the next section. The complete .xml file controlling the SOFA based probing simulator with haptic response is presented in Appendix B.

4.3.6 SOFA Framework Discussion and the Implications on Cutting Simulation

The SOFA framework is a very useful tool for developing finite element based simulation systems. The highly modular design adds versatility to the code that is required for simulation of various surgical tasks. A vast array of options are available for material models, solvers, boundary conditions, graphical display and collision models when working towards developing a simulation based on linear models. The major limitation in the code comes when using the TLED algorithm and the GPU based methodology. The SOFA framework is an open source code developed by researchers in the surgical simulation field as a tool for others to use to quickly and efficiently develop their own simulation systems. The work to include GPU based methods has only recently begun, so all of the various modules existing for the linear support have yet to be converted to interface properly with the TLED algorithm. Once the additional functionality becomes GPU compatible, the ability to reach real-time, haptics enabled simulation rates with the SOFA framework will be achievable.

The major limitations that arose in the development of the probing simulator came from the contact models. The general deformation response using the TLED algorithm with the reduced polynomial model worked well for both accuracy and speed of calculations, even with an outdated processor and GPU. However, the addition of collision to the simulation drastically decreased the speed of the simulation to a point at which it was not feasible to use with haptic feedback. An alternative for

smoothing out the haptic response was presented in the SOFA documentation. The approach involved placing a fictitious spring between the simulated probe and the mapping of the haptic device. By including this spring force in the simulation, there would in effect be two springs in series, the tissue itself and the fictitious spring. If the spring constant of the fictitious spring is set high enough, the response will be purely from the deformation of the tissue. The aim is to tune the spring constant parameters to each simulation to effectively dampen out the vibrations felt on the haptic display. While this approach may smooth the haptic display, it will greatly effect the accuracy of the force response felt by the user, negating the benefit of using a more accurate hyperelastic material model.

When looking to implement the cohesive zone based cutting models into the SOFA simulation, additional complications arise in the area of collision response. Neglecting the large decrease in speed, the sphere-to-sphere based contact is simply not accurate enough to be used for cutting simulation. As seen in Section 3.2.1, the cutting model requires a high resolution contact response between the scalpel blade and the tissue surface. While sphere based collisions are relatively simple to implement, they have a major limitation in that the spheres are not able to replicate the proper surface shape unless a very dense mesh is utilized. By using such a dense mesh, any increase in computational speed achieved from the use of the GPU accelerated TLED approach will be negated. Additionally, using sphere-to-sphere contact will act to skew the direction of the applied stress as the two sphere shift around one another trying to reach the minimum energy state. An inaccurate stress distribution will result in an incorrect response in the cohesive zone and will

alter the desired force output. For these reasons, the cutting methods previously developed were not implemented in the SOFA framework.

Once the problem of solving collisions in an accurate and timely method is developed, the addition of cohesive zone based cutting should be relatively simple to include in existing simulation systems. The parameters required for using the cohesive zone are simply the stress, strain and/or energy terms. All of these values are calculated to some extent during each time step of the finite element based simulation. An additional segment of the code would be needed only to compare the stress, strain and/or energy terms to the cohesive model. If they fall below the damage threshold, the simulation progresses normally. However, if they are above the threshold, the new nodal displacement can be directly calculated from the cohesive zone theory and passed into the subsequent step of the TLED algorithm. This would provide the ability to add cutting simulation in the real-time system with very little added computational expense.

Chapter 5

Conclusions and Future Works

The aim of this project was to develop a surgical simulator capable of accurately representing the haptic and visual information required for training medical professionals in probing and cutting tasks. The foundations of this project are generally split into two research areas: the derivation of reality-based models for biological tissue deformation and cutting tasks, and the development of real-time simulations of the medical procedures. A list of the current contributions in these two areas of study are listed below.

- *Ex Vivo* Soft Tissue Modeling
 - Designed and fabricated a test apparatus capable of performing compression, tension and pure shear tests on biological samples.
 - Generated an image-based method to track tissue deformation during compression tests to verify uniform expansion of tissue sample.
 - Analyzed data from the compression test, which was used by Dr. Zhan Gao, to develop models which describe the tissue response to general loading for the purpose of surgical simulation.

- *In Vivo* Modeling of Probing
 - Developed and constructed a testing fixture for probing and cutting of

soft tissue used in the operating theater.

- Devised a method to measure the shape of a liver sample for the generation of a full three dimensional model through the use of a stereo camera system which resulted in geometric realism with errors on the order of 1.5mm.
- Formulated a technique to accurately account for the effects of tissue settling due to gravity during simulation of surgical tasks for more realistic results.
- Established an approach for validating the efficacy of proposed constitutive models through both force verification and surface deformation assessment.
- Collaborated with Dr. Zhan Gao to establish a method whereby existing *ex vivo* material models could be improved upon, resulting in increased model accuracy.

- Cutting of Porcine Liver

- Performed *in vivo* scalpel cutting tests on porcine liver to obtain data required for modeling of the scalpel cutting process on soft biological tissues.
- Developed a method that utilizes a cohesive zone, fracture mechanics approach to describe the fracture process occurring during scalpel-tissue contact.

- Extracted model parameters from experimental cutting data to fully characterize the cohesive zone model.
 - Validated the cohesive zone approach for cutting simulation through a dynamic, explicit analysis which was able to show a high correlation between the force and displacement characteristics of an experimental and simulated scalpel cut.
- Real-Time Simulation
 - Pre-Processor Base Simulator
 - * Developed a reality-based, haptics-enabled simulator for probing of *ex vivo* porcine liver based on an experimentally determined nonlinear constitutive model through the use of pre-processing techniques.
 - * Volumetric deformations and reaction forces were calculated directly from hyperelastic material models without need for simplification which resulted in increased realism.
 - * Extended pre-processor based code to simulation of scalpel cutting using the cohesive zone approach capable of real-time haptic and graphic display.
 - Real-Time Simulation using SOFA Framework
 - * Implemented the GPU based Total Lagrangian Explicit Dynamics approach for the *in vivo* based reduced polynomial model using the SOFA framework to increase the accuracy of tissue deformations in

surgical simulators.

- * Using the Simulation Open Framework Architecture (SOFA), developed a finite element based probing simulator capable of real-time visual display for any arbitrary probe position and orientation.

5.1 Future Work

While the presented work was able to more accurately replicate the tool-tissue interactions seen in common surgical tasks due to the foundation in physics based modeling, the simulations can be further improved upon with some additional focus in the following areas:

- Model Development
 - The inclusion of viscous terms to the existing hyperelastic models will further increase the simulation accuracy. Surgical procedures are undertaken at various rates and soft tissue has been shown to have a rate dependent response.
 - The dynamic effects arising due to the scalpel-tissue contact should be studied in detail to improve upon the overall resolution of the cutting model.
 - Frictional effects between the scalpel blade and the soft-tissue during cutting can be analyzed to determine its effect on the cutting process.

- Real-Time Simulation

- The computational efficiency of the collision response in the SOFA framework should be increased to a state that will allow for real-time haptic display, possibly through GPU programming.
- Methods of collision detection are needed in addition to sphere-to-sphere contact that are compatible with the TLED algorithm.
- Following the improvements to the contact definitions, the cohesive zone method of scalpel cutting should be added to the SOFA framework.

Once the above mentioned advancements are developed, the ability to simulate probing and cutting tasks on liver tissue with real-time graphics and haptic display could be achieved using the SOFA framework. It should be noted that many aspects of the future works listed in this section are SOFA framework specific. The models developed throughout this study are general in nature and can be applied in any finite element based simulation code. Therefore, inclusion of the reality-based models governing the soft-tissue response to tool-tissue interactions may also be applied to existing commercially available simulators to improve the overall system performance.

Appendix A

TLED Code for Reduced Polynomial Model

```
--global-- void CudaHexahedronTLEDForceField3f_calcForce_kernel0(float Lambda,
float Mu, int nbElem, float4* F0_gpu, float4* F1_gpu, float4* F2_gpu, float4*
F3_gpu, float4* F4_gpu, float4* F5_gpu, float4* F6_gpu, float4* F7_gpu)
{
    int index0 = umul24(blockIdx.x,BSIZE);
    int index1 = threadIdx.x;
    int index = index0+index1;

    if (index < nbElem)
    {
        // Shape function derivatives matrix
        float4 Dh0_a = tex1Dfetch(texDhC0, 2*index);
        float4 Dh0_b = tex1Dfetch(texDhC0, 2*index+1);
        float4 Dh1_a = tex1Dfetch(texDhC1, 2*index);
        float4 Dh1_b = tex1Dfetch(texDhC1, 2*index+1);
        float4 Dh2_a = tex1Dfetch(texDhC2, 2*index);
        float4 Dh2_b = tex1Dfetch(texDhC2, 2*index+1);

        int4 NodesPerElement = tex1Dfetch(texNodesPerElement, 2*index);
        CudaVec3f Node1Disp = getX(NodesPerElement.x) - getX0(NodesPerElement.x);
        CudaVec3f Node2Disp = getX(NodesPerElement.y) - getX0(NodesPerElement.y);
        CudaVec3f Node3Disp = getX(NodesPerElement.z) - getX0(NodesPerElement.z);
        CudaVec3f Node4Disp = getX(NodesPerElement.w) - getX0(NodesPerElement.w);

        NodesPerElement = tex1Dfetch(texNodesPerElement, 2*index+1);
        CudaVec3f Node5Disp = getX(NodesPerElement.x) - getX0(NodesPerElement.x);
        CudaVec3f Node6Disp = getX(NodesPerElement.y) - getX0(NodesPerElement.y);
        CudaVec3f Node7Disp = getX(NodesPerElement.z) - getX0(NodesPerElement.z);
        CudaVec3f Node8Disp = getX(NodesPerElement.w) - getX0(NodesPerElement.w);

        /**
        * Computes the transpose of deformation gradient
        *
        * Transpose of displacement derivatives = transpose(shape function
        derivatives) * ElementNodalDisplacement
        * Transpose of deformation gradient = transpose of displacement derivatives
        + identity
        */
        float XT[3][3];

        //Column 1
        XT[0][0] = Dh0_a.x*Node1Disp.x + Dh0_a.y*Node2Disp.x + Dh0_a.z*Node3Disp.x
+ Dh0_a.w*Node4Disp.x +
Dh0_b.x*Node5Disp.x + Dh0_b.y*Node6Disp.x + Dh0_b.z*Node7Disp.x
+ Dh0_b.w*Node8Disp.x + 1.0f;
        XT[1][0] = Dh1_a.x*Node1Disp.x + Dh1_a.y*Node2Disp.x + Dh1_a.z*Node3Disp.x
+ Dh1_a.w*Node4Disp.x +
Dh1_b.x*Node5Disp.x + Dh1_b.y*Node6Disp.x + Dh1_b.z*Node7Disp.x
+ Dh1_b.w*Node8Disp.x;
        XT[2][0] = Dh2_a.x*Node1Disp.x + Dh2_a.y*Node2Disp.x + Dh2_a.z*Node3Disp.x
+ Dh2_a.w*Node4Disp.x +
Dh2_b.x*Node5Disp.x + Dh2_b.y*Node6Disp.x + Dh2_b.z*Node7Disp.x
+ Dh2_b.w*Node8Disp.x;

        //Column 2
        XT[0][1] = Dh0_a.x*Node1Disp.y + Dh0_a.y*Node2Disp.y + Dh0_a.z*Node3Disp.y
+ Dh0_a.w*Node4Disp.y +
```

```

        Dh0_b.x*Node5Disp.y + Dh0_b.y*Node6Disp.y + Dh0_b.z*Node7Disp.y
        + Dh0_b.w*Node8Disp.y;
XT[1][1] = Dh1_a.x*Node1Disp.y + Dh1_a.y*Node2Disp.y + Dh1_a.z*Node3Disp.y
        + Dh1_a.w*Node4Disp.y +
        Dh1_b.x*Node5Disp.y + Dh1_b.y*Node6Disp.y + Dh1_b.z*Node7Disp.y
        + Dh1_b.w*Node8Disp.y + 1.0f;
XT[2][1] = Dh2_a.x*Node1Disp.y + Dh2_a.y*Node2Disp.y + Dh2_a.z*Node3Disp.y
        + Dh2_a.w*Node4Disp.y +
        Dh2_b.x*Node5Disp.y + Dh2_b.y*Node6Disp.y + Dh2_b.z*Node7Disp.y
        + Dh2_b.w*Node8Disp.y;

//Column 3
XT[0][2] = Dh0_a.x*Node1Disp.z + Dh0_a.y*Node2Disp.z + Dh0_a.z*Node3Disp.z
        + Dh0_a.w*Node4Disp.z +
        Dh0_b.x*Node5Disp.z + Dh0_b.y*Node6Disp.z + Dh0_b.z*Node7Disp.z
        + Dh0_b.w*Node8Disp.z;
XT[1][2] = Dh1_a.x*Node1Disp.z + Dh1_a.y*Node2Disp.z + Dh1_a.z*Node3Disp.z
        + Dh1_a.w*Node4Disp.z +
        Dh1_b.x*Node5Disp.z + Dh1_b.y*Node6Disp.z + Dh1_b.z*Node7Disp.z
        + Dh1_b.w*Node8Disp.z;
XT[2][2] = Dh2_a.x*Node1Disp.z + Dh2_a.y*Node2Disp.z + Dh2_a.z*Node3Disp.z
        + Dh2_a.w*Node4Disp.z +
        Dh2_b.x*Node5Disp.z + Dh2_b.y*Node6Disp.z + Dh2_b.z*Node7Disp.z
        + Dh2_b.w*Node8Disp.z + 1.0f;

/**
 * Computes the right Cauchy–Green deformation tensor C = XT*X (in fact we
 * compute only 6 terms since C is symmetric)
 */
float C11, C12, C13, C22, C23, C33;
C11 = XT[0][0]*XT[0][0] + XT[0][1]*XT[0][1] + XT[0][2]*XT[0][2];
C12 = XT[0][0]*XT[1][0] + XT[0][1]*XT[1][1] + XT[0][2]*XT[1][2];
C13 = XT[0][0]*XT[2][0] + XT[0][1]*XT[2][1] + XT[0][2]*XT[2][2];
C22 = XT[1][0]*XT[1][0] + XT[1][1]*XT[1][1] + XT[1][2]*XT[1][2];
C23 = XT[1][0]*XT[2][0] + XT[1][1]*XT[2][1] + XT[1][2]*XT[2][2];
C33 = XT[2][0]*XT[2][0] + XT[2][1]*XT[2][1] + XT[2][2]*XT[2][2];

/**
 * Computes determinant of X
 */
float J = XT[0][0]*( XT[1][1]*XT[2][2] - XT[2][1]*XT[1][2] )
        - XT[1][0]*( XT[0][1]*XT[2][2] - XT[2][1]*XT[0][2] )
        + XT[2][0]*( XT[0][1]*XT[1][2] - XT[1][1]*XT[0][2] );

/**
 * Computes second Piola–Kirchhoff stress
 */
float SPK[6];
float RPD10=0.001;
float RPD20=0.001;
float RPD30=0.001;
float RPD40=0.001;
float RPC10=781.8981;
float RPC20=-747.6308;
float RPC30=426.1348;
float RPC40=122.7012;

// Determinant of C
float invdetC = _fdivdef(1.0f, C11*(C22*C33 - C23*C23)
        - C12*(C12*C33 - C23*C13)
        + C13*(C12*C23 - C22*C13) );

// C inverse
float Ci11, Ci12, Ci13, Ci22, Ci23, Ci33;
Ci11 = (C22*C33 - C23*C23)*invdetC;
Ci12 = (C13*C23 - C12*C33)*invdetC;
Ci13 = (C12*C23 - C13*C22)*invdetC;
Ci22 = (C11*C33 - C13*C13)*invdetC;

```

```

Ci23 = (C12*C13 - C11*C23)*invdetC;
Ci33 = (C11*C22 - C12*C12)*invdetC;

// 4Term Reduced Polynomial
float J23 = _powf(J, -(float)2/3);
float I1bar = J23*(C11+C22+C33);
float RPP1 = I1bar-3.0f;
float RPP2 = pow((I1bar-3.0f), 2.0f);
float RPP3 = pow((I1bar-3.0f), 3.0f);
float x1 = J23*2.0f*(RPC10 + 2.0f*RPC20*RPP1 + 3.0f*RPC30*RPP2 + 4.0f*RPC40
    *RPP3);
float x4 = _fdivdef(-x1*(C11+C22+C33), 3.0f);

float RPj1 = J-1.0f;
float RPj3 = pow((J-1.0f), 3.0f);
float RPj5 = pow((J-1.0f), 5.0f);
float RPj7 = pow((J-1.0f), 7.0f);
float Rpv1 = _fdivdef(2.0f, RPD10);
float Rpv2 = _fdivdef(4.0f, RPD20);
float Rpv3 = _fdivdef(6.0f, RPD30);
float Rpv4 = _fdivdef(8.0f, RPD40);
float x5 = J*(Rpv1*RPj1 + Rpv2*RPj3 + Rpv3*RPj5 + Rpv4*RPj7);

// Elastic component of the response (isochoric part + volumetric part)
float SiE11, SiE12, SiE13, SiE22, SiE23, SiE33;
SiE11 = x4*Ci11 + x1;
SiE22 = x4*Ci22 + x1;
SiE33 = x4*Ci33 + x1;
SiE12 = x4*Ci12;
SiE23 = x4*Ci23;
SiE13 = x4*Ci13;

float SvE11, SvE12, SvE13, SvE22, SvE23, SvE33;
SvE11 = x5*Ci11;
SvE22 = x5*Ci22;
SvE33 = x5*Ci33;
SvE12 = x5*Ci12;
SvE23 = x5*Ci23;
SvE13 = x5*Ci13;

SPK[0] = SiE11 + SvE11;
SPK[1] = SiE22 + SvE22;
SPK[2] = SiE33 + SvE33;
SPK[3] = SiE12 + SvE12;
SPK[4] = SiE23 + SvE23;
SPK[5] = SiE13 + SvE13;

// Gets the Jacobian determinant
float detJ = tex1Dfetch(texDetJ, index);
SPK[0] *= 8*detJ;
SPK[1] *= 8*detJ;
SPK[2] *= 8*detJ;
SPK[3] *= 8*detJ;
SPK[4] *= 8*detJ;
SPK[5] *= 8*detJ;

/**
 * Computes strain-displacement matrix
 */
F0_gpu[index] = computeForce_hex(0, Dh0_a, Dh0_b, Dh1_a, Dh1_b, Dh2_a,
    Dh2_b, Node1Disp, Node2Disp, Node3Disp, Node4Disp,
    Node5Disp, Node6Disp, Node7Disp, Node8Disp, SPK
    , index);

F1_gpu[index] = computeForce_hex(1, Dh0_a, Dh0_b, Dh1_a, Dh1_b, Dh2_a,
    Dh2_b, Node1Disp, Node2Disp, Node3Disp, Node4Disp,
    Node5Disp, Node6Disp, Node7Disp, Node8Disp, SPK
    , index);

```



```

F2_gpu[index] = computeForce_hex(2, Dh0_a, Dh0_b, Dh1_a, Dh1_b, Dh2_a,
    Dh2_b, Node1Disp, Node2Disp, Node3Disp, Node4Disp,
    Node5Disp, Node6Disp, Node7Disp, Node8Disp, SPK
    , index);

F3_gpu[index] = computeForce_hex(3, Dh0_a, Dh0_b, Dh1_a, Dh1_b, Dh2_a,
    Dh2_b, Node1Disp, Node2Disp, Node3Disp, Node4Disp,
    Node5Disp, Node6Disp, Node7Disp, Node8Disp, SPK
    , index);

F4_gpu[index] = computeForce_hex(4, Dh0_a, Dh0_b, Dh1_a, Dh1_b, Dh2_a,
    Dh2_b, Node1Disp, Node2Disp, Node3Disp, Node4Disp,
    Node5Disp, Node6Disp, Node7Disp, Node8Disp, SPK
    , index);

F5_gpu[index] = computeForce_hex(5, Dh0_a, Dh0_b, Dh1_a, Dh1_b, Dh2_a,
    Dh2_b, Node1Disp, Node2Disp, Node3Disp, Node4Disp,
    Node5Disp, Node6Disp, Node7Disp, Node8Disp, SPK
    , index);

F6_gpu[index] = computeForce_hex(6, Dh0_a, Dh0_b, Dh1_a, Dh1_b, Dh2_a,
    Dh2_b, Node1Disp, Node2Disp, Node3Disp, Node4Disp,
    Node5Disp, Node6Disp, Node7Disp, Node8Disp, SPK
    , index);

F7_gpu[index] = computeForce_hex(7, Dh0_a, Dh0_b, Dh1_a, Dh1_b, Dh2_a,
    Dh2_b, Node1Disp, Node2Disp, Node3Disp, Node4Disp,
    Node5Disp, Node6Disp, Node7Disp, Node8Disp, SPK
    , index);
}

/**
 * Function to be called from the device to compute forces from stresses
 * => allows us to save registers by recomputing the deformation gradient
 */
__device__ float4 computeForce_hex(const int node, const float4 Dh0_a, const float4
    Dh0_b, const float4 Dh1_a,
    const float4 Dh1_b, const float4 Dh2_a, const
    float4 Dh2_b, const float3 Node1Disp,
    const float3 Node2Disp, const float3
    Node3Disp, const float3 Node4Disp,
    const float3 Node5Disp, const float3
    Node6Disp, const float3 Node7Disp,
    const float3 Node8Disp, const float * SPK,
    const int tid)
{
    float XT[3][3];

    //Column 1
    XT[0][0] = Dh0_a.x*Node1Disp.x + Dh0_a.y*Node2Disp.x + Dh0_a.z*Node3Disp.x +
        Dh0_a.w*Node4Disp.x +
        Dh0_b.x*Node5Disp.x + Dh0_b.y*Node6Disp.x + Dh0_b.z*Node7Disp.x +
        Dh0_b.w*Node8Disp.x + 1.0f;
    XT[1][0] = Dh1_a.x*Node1Disp.x + Dh1_a.y*Node2Disp.x + Dh1_a.z*Node3Disp.x +
        Dh1_a.w*Node4Disp.x +
        Dh1_b.x*Node5Disp.x + Dh1_b.y*Node6Disp.x + Dh1_b.z*Node7Disp.x +
        Dh1_b.w*Node8Disp.x;
    XT[2][0] = Dh2_a.x*Node1Disp.x + Dh2_a.y*Node2Disp.x + Dh2_a.z*Node3Disp.x +
        Dh2_a.w*Node4Disp.x +
        Dh2_b.x*Node5Disp.x + Dh2_b.y*Node6Disp.x + Dh2_b.z*Node7Disp.x +
        Dh2_b.w*Node8Disp.x;

    //Column 2
    XT[0][1] = Dh0_a.x*Node1Disp.y + Dh0_a.y*Node2Disp.y + Dh0_a.z*Node3Disp.y +
        Dh0_a.w*Node4Disp.y +
        Dh0_b.x*Node5Disp.y + Dh0_b.y*Node6Disp.y + Dh0_b.z*Node7Disp.y +
        Dh0_b.w*Node8Disp.y;

```

```

XT[1][1] = Dh1_a.x*Node1Disp.y + Dh1_a.y*Node2Disp.y + Dh1_a.z*Node3Disp.y +
Dh1_a.w*Node4Disp.y +
Dh1_b.x*Node5Disp.y + Dh1_b.y*Node6Disp.y + Dh1_b.z*Node7Disp.y +
Dh1_b.w*Node8Disp.y + 1.0f;
XT[2][1] = Dh2_a.x*Node1Disp.y + Dh2_a.y*Node2Disp.y + Dh2_a.z*Node3Disp.y +
Dh2_a.w*Node4Disp.y +
Dh2_b.x*Node5Disp.y + Dh2_b.y*Node6Disp.y + Dh2_b.z*Node7Disp.y +
Dh2_b.w*Node8Disp.y;

//Column 3
XT[0][2] = Dh0_a.x*Node1Disp.z + Dh0_a.y*Node2Disp.z + Dh0_a.z*Node3Disp.z +
Dh0_a.w*Node4Disp.z +
Dh0_b.x*Node5Disp.z + Dh0_b.y*Node6Disp.z + Dh0_b.z*Node7Disp.z +
Dh0_b.w*Node8Disp.z;
XT[1][2] = Dh1_a.x*Node1Disp.z + Dh1_a.y*Node2Disp.z + Dh1_a.z*Node3Disp.z +
Dh1_a.w*Node4Disp.z +
Dh1_b.x*Node5Disp.z + Dh1_b.y*Node6Disp.z + Dh1_b.z*Node7Disp.z +
Dh1_b.w*Node8Disp.z;
XT[2][2] = Dh2_a.x*Node1Disp.z + Dh2_a.y*Node2Disp.z + Dh2_a.z*Node3Disp.z +
Dh2_a.w*Node4Disp.z +
Dh2_b.x*Node5Disp.z + Dh2_b.y*Node6Disp.z + Dh2_b.z*Node7Disp.z +
Dh2_b.w*Node8Disp.z + 1.0f;

float BL[6];
float FX, FY, FZ;
float4 HG_read;

float Dh0, Dh1, Dh2;
switch(node)
{
case 0 :
Dh0 = Dh0_a.x;
Dh1 = Dh1_a.x;
Dh2 = Dh2_a.x;
break;
case 1 :
Dh0 = Dh0_a.y;
Dh1 = Dh1_a.y;
Dh2 = Dh2_a.y;
break;
case 2 :
Dh0 = Dh0_a.z;
Dh1 = Dh1_a.z;
Dh2 = Dh2_a.z;
break;
case 3 :
Dh0 = Dh0_a.w;
Dh1 = Dh1_a.w;
Dh2 = Dh2_a.w;
break;
case 4 :
Dh0 = Dh0_b.x;
Dh1 = Dh1_b.x;
Dh2 = Dh2_b.x;
break;
case 5 :
Dh0 = Dh0_b.y;
Dh1 = Dh1_b.y;
Dh2 = Dh2_b.y;
break;
case 6 :
Dh0 = Dh0_b.z;
Dh1 = Dh1_b.z;
Dh2 = Dh2_b.z;
break;
case 7 :
Dh0 = Dh0_b.w;
Dh1 = Dh1_b.w;

```

```

        Dh2 = Dh2_b.w;
        break;
    }

    // Computes X component
    BL[0] = Dh0 * XT[0][0];
    BL[1] = Dh1 * XT[1][0];
    BL[2] = Dh2 * XT[2][0];
    BL[3] = Dh1 * XT[0][0] + Dh0 * XT[1][0];
    BL[4] = Dh2 * XT[1][0] + Dh1 * XT[2][0];
    BL[5] = Dh2 * XT[0][0] + Dh0 * XT[2][0];
    FX = SPK[0]*BL[0] + SPK[1]*BL[1] + SPK[2]*BL[2] + SPK[3]*BL[3] + SPK[4]*BL[4] +
        SPK[5]*BL[5];

    // Computes Y component
    BL[0] = Dh0 * XT[0][1];
    BL[1] = Dh1 * XT[1][1];
    BL[2] = Dh2 * XT[2][1];
    BL[3] = Dh1 * XT[0][1] + Dh0 * XT[1][1];
    BL[4] = Dh2 * XT[1][1] + Dh1 * XT[2][1];
    BL[5] = Dh2 * XT[0][1] + Dh0 * XT[2][1];
    FY = SPK[0]*BL[0] + SPK[1]*BL[1] + SPK[2]*BL[2] + SPK[3]*BL[3] + SPK[4]*BL[4] +
        SPK[5]*BL[5];

    // Computes Z component
    BL[0] = Dh0 * XT[0][2];
    BL[1] = Dh1 * XT[1][2];
    BL[2] = Dh2 * XT[2][2];
    BL[3] = Dh1 * XT[0][2] + Dh0 * XT[1][2];
    BL[4] = Dh2 * XT[1][2] + Dh1 * XT[2][2];
    BL[5] = Dh2 * XT[0][2] + Dh0 * XT[2][2];
    FZ = SPK[0]*BL[0] + SPK[1]*BL[1] + SPK[2]*BL[2] + SPK[3]*BL[3] + SPK[4]*BL[4] +
        SPK[5]*BL[5];

    // Writes into global memory
    return make_float4( FX, FY, FZ, 0);
}

```

Appendix B

Scene File for SOFA Probing Simulator

```
<?xml version="1.0"?>
<Node name="Root" dt="0.00095" showBehaviorModels="1" gravity="0 -9.81 0">

  <LCPCConstraintSolver name="LCPCConstraintSolver1" displayTime="0" initial_guess="1"
    build_lcp="1" tolerance="1e-010" mu="0.7" group="0" />
  <BackgroundSetting color="1.0 1.0 1.0" />
  <DefaultPipeline name="CollisionPipeline" verbose="0" />
  <BruteForceDetection name="N2" />
  <DefaultContactManager name="collision response" response="default" />
  <NewOmniDriver scale="1" omniVisu="false" permanent="1" forceScale="10" />
  <MinProximityIntersection name="proximity" alarmDistance="0.0001" contactDistance
    ="0.00005" />
  <CentralDifferenceSolver name="centralDifferenceSolver15" rayleighMass="50" />

  <Node name="Omni" gravity="0 -9.81 0" showVisualModels="1" >
    <MechanicalObject template="Rigid" name="DOFs" />
    <UniformMass template="Rigid" name="OmniMass" showAxisSizeFactor="0"
      separateGravity="1" />
    <MechanicalStateController template="Rigid" name="default40" listening="1"
      handleEventTriggersUpdate="1" mainDirection="0 1 0" />

    <Node name="USProbeVisualModel">
      <OglModel template="ExtVec3f" name="Probe" fileMesh="mesh/
        USProbePhantomSmall.obj" />
      <RigidMapping template="Rigid,ExtVec3f" name="ProbeVisualMapping"
        input="@../DOFs" output="@Probe" />
    </Node>

    <Node name="RefModel" >
      <MeshLoader name="default41" filename="mesh/USProbePhantomSmall.
        obj" />
      <MeshTopology name="default42" />
      <MechanicalObject template="Vec3d" name="instrumentCollisionState"
        />
      <LineModel name="lineModel1" group="1" />
      <PointModel name="pointModel1" group="1" />
      <TSphereModel template="Vec3f" name="tSphereModel1" radius="0.0025"
        contactStiffness="0.01" />
      <RigidMapping template="Rigid,Vec3d" name="ProbeMM->CMmapping"
        mapForces="1" mapConstraints="1" mapMasses="1" index="0" />
    </Node>
  </Node>

  <Node name="Liver" >

    <MeshGmshLoader name="loader" filename="mesh/SimplifiedLiverMeters.msh" />
    <MechanicalObject template="CudaVec3f" name="Volume" position="@loader.
      position" showObjectScale="0.1" showIndicesScale="0.0001"
      translation="@loader.translation" rotation="@loader.rotation" scale3d
      ="@loader.scale3d" />
    <UniformMass template="CudaVec3f" name="uniformMass18" mass="0" totalmass
      ="1.0" />
    <HexahedronSetTopologyContainer name="HexaContainer" fileTopology="mesh/
      SimplifiedLiverMeters.msh" hexahedra="@loader.hexahedra" />
    <HexahedronSetGeometryAlgorithms template="CudaVec3f" name="HexaGeoAlgo"
      showIndicesScale="0.0001" drawHexa="1" drawColorHexahedra="1 1 0.3"

```

```

/>
<FixedConstraint template="CudaVec3f" name="fixedConstraint11" indices="1
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27
28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51
52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98
99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116
117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133
134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151
152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168
169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186
187 188 189 190 191 192 193 194 195 196 197 198 199 200 201" />
<CudaHexahedronTLEDForceField name="FEM" youngModulus="3000" poissonRatio
="0.45" isViscoelastic="0" isAnisotropic="0" preferredDirection="0
0.707 0.707"/>

<Node name="ShpereCollision" >
  <MeshLoader name="MeshLoaderSphere" filename="mesh/
  SimplifiedLiverMeters.obj" />
  <MeshTopology name="MeshTopologySphere" />
  <MechanicalObject template="Vec3d" name="CollisionStateSphere" />
  <TSphereModel template="Vec3d" name="tSphereModelLiver" radius
  ="0.009" contactStiffness="0.01" />
  <IdentityMapping template="CudaVec3f,Vec3d" name="LiverMM->
  CMmapping" mapForces="1" mapConstraints="1" mapMasses="1" />
</Node>

<Node name="QuadMesh" >
  <QuadSetTopologyContainer name="QuadContainer" />
  <QuadSetTopologyModifier name="QuadModifier" />
  <QuadSetGeometryAlgorithms template="CudaVec3f" name="QuadGeomAlgo"
  showIndicesScale="0.0001" />
  <Hexa2QuadTopologicalMapping name="HexaQuadMapping" input="@../
  HexaContainer" output="@QuadContainer" />
  <Node name="TriangleMesh" >
    <TriangleSetTopologyContainer name="TriContainer" />
    <TriangleSetTopologyModifier name="TriModifier" />
    <TriangleSetGeometryAlgorithms template="CudaVec3f" name="
    TriGeomAlgo" showIndicesScale="0.0001" />
    <Quad2TriangleTopologicalMapping name="QuadTriMapping"
    input="@../QuadContainer" output="@TriContainer" />
    <Node name="SurfVisual" >
      <OglModel template="ExtVec3f" name="TriVisual" />
      <IdentityMapping template="CudaVec3f,ExtVec3f" name
      ="TriIdentMap" input="@.././././Volume" output
      ="@TriVisual" />
    </Node>
  </Node>
</Node>
</Node>

<Node name="TorsoSkel" >
  <OglModel template="ExtVec3f" name="SkeletonVisMod" fileMesh="mesh/
  torsoskelcutribsmetersphantom.obj" />
</Node>

<Node name="TorsoSkin" >
  <OglModel template="ExtVec3f" name="SkinVisMod" fileMesh="mesh/
  torsoskinopenmetersphantom.obj" />
</Node>

</Node>

```

Bibliography

- [1] Bummo Ahn and Jung Kim. Measurement and characterization of soft tissue behavior with surface deformation and force response under large deformations. *Medical Image Analysis*, 14:138–148, 2010.
- [2] K.S. Arun, T.S. Huang, and S.D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700, 1987.
- [3] Toufic Azar and Vincent Hayward. Estimation of the fracture toughness of soft tissue from needle insertion. In *4th International Symposium on Biomedical Simulation (ISBMS)*, pages 166–173, 2008.
- [4] B.R.K Blackman, H. Hadavinia, A.J. Kinloch, and J.G. Williams. The use of a cohesive zone model to study the fracture of fibre composites and adhesively-bonded joints. *International Journal of Fracture*, 119:25–46, 2003.
- [5] Morten Bro-Nielsen and Stéphane Cotin. Real-time volumetric deformable models for surgery simulation using finite elements and condensation. *Computer Graphics Forum*, 15:57–66, 1996.
- [6] Jeffrey D. Brown, Jacob Rosen, Yoon Sang Kim, Lily Chang, Mika N. Sinaman, and Blake Hannaford. In-vivo and in-situ compressive properties of porcine abdominal soft tissues. In *Medicine Meets Virtual Reality, MMVR 2003*, pages 26–32, 2003.
- [7] Jeffrey D. Brown, Jacob Rosen, Mika N. Sinaman, and Blake Hannaford. In-vivo and postmortem compressive properties of porcine abdominal organs. In *Medical Image Computing and Computer-Assisted Intervention, MICCAI 2003*, number 2878 in Lecture Notes in Computer Science, pages 238–245, 2003.
- [8] Grigore C. Burdea. *Force and Touch Feedback for Virtual Reality*. John Wiley & Sons, Inc, New York, NY, 1996.
- [9] David L. Butler, Edward S. Groom, Frank R. Noyes, Ronald F. Zernicke, and Kim Brackett. Effects of structure and strain measurement technique on the material properties of young human tendons and fascia. *Journal of Biomechanics*, 17(8):579–596, 1984.
- [10] F.J. Carter, T.G. Frank, P.J. Davies, D. McLean, and A. Cuschieri. Measurements and modelling of the compliance of human and porcine organs. *Medical Image Analysis*, 5:231–236, 2001.
- [11] T. Chanthasopeephan, J.P. Desai, and A. Lau. Modeling soft tissue deformation prior to cutting for surgical simulation: finite element analysis and study of cutting parameters. *IEEE Transactions of Biomedical Engineering*, 54(3):349–359, 2007.

- [12] C.R. Chen, O. Kolednik, I. Scheider, T. Siegmund, A. Tatschl, and F.D. Fischer. On the determination of the cohesive zone parameters for the modeling of microductile crack growth in thick specimens. *International Journal of Fracture*, 120:517–536, 2003.
- [13] C. Chui, E. Kobayashi, X. Chen, T. Hisada, and I. Sakuma. Combined compression and elongation experiments and non-linear modelling of liver tissue for surgical simulation. *Medical and Biological Engineering and Computing*, 42(6):787–798, 2004.
- [14] C. Chui, E. Kobayashi, X. Chen, T. Hisada, and I. Sakuma. Transversely isotropic properties of porcine liver tissue: experiments and constitutive modelling. *Medical and Biological Engineering and Computing*, 45(1):99–106, 2007.
- [15] Olivier Comas, Zeike Taylor, Jérémie Allard, Sébastien Ourselin, Stéphane Cotin, and Josh Passenger. Efficient nonlinear fem for soft tissue modelling and its gpu implementation within the open source framework sofa. *Lecture Notes In Computer Science*, 5104:28–39, 2008.
- [16] Stéphane Cotin, Hervé Delingette, and Nicholas Ayache. A hybrid elastic model for real-time cutting, deformations, and force feedback for surgery training and simulation. *The Visual Computer*, 16(8):437–452, 2000.
- [17] Hadrien Courtecuisse, Hoeryong Jung, Jérémie Allard, Christian Duriez, Doo Yong Lee, and Stéphane Cotin. Gpu-based real-time soft tissue deformation with cutting and haptic feedback. *Progress in Biophysics and Molecular Biology*, 103:159–168, 2010.
- [18] Penny J. Davies, Fiona J. Carter, and Alfred Cuschieri. Mathematical modelling for keyhole surgery simulations: a biomechanical model for spleen tissue. *IMA Journal of Applied Mathematics*, 67:41–67, 2002.
- [19] Hervé Delingette. Toward realistic soft tissue modeling in medical simulation. In *Proceedings of the IEEE: Special Issue on Surgery Simulation*, pages 512–523, 1998.
- [20] Hervé Delingette and Nicholas Ayache. Hepatic surgery simulation. *Communications of the ACM*, 48(2):31–36, 2005.
- [21] M. Elices, G.V. Guinea, J. Gomez, and J. Planas. The cohesive zone model: advantages, limitations and challenges. *Engineering Fracture Mechanics*, 69:137–163, 2002.
- [22] H.L. Ewalds and R.J.H. Wanhill, editors. *Fracture Mechanics*. Edward Arnold Ltd, Baltimore, MD, 1984.
- [23] Mehdi Farshad, Michel Barbezat, Peter Flueler, Franz Schmidlin, Pierre Graber, and Peter Niederer. Material characterization of the pig kidney in

- relation with the biomechanical analysis of renal trauma. *Journal of Biomechanics*, 32:417–425, 1999.
- [24] J.W. Foulk, D.H. Allen, and K.L.E. Helms. Formulation of a three-dimensional cohesive zone model for application to a finite element algorithm. *Computer Methods In Applied Mechanics And Engineering*, 183:51–66, 2000.
- [25] Zhan Gao and Jaydev Desai. Estimating zero-strain states of very soft tissue under gravity loading using digital image correlation. *Medical Image Analysis*, 14(2):126–137, 2010.
- [26] Zhan Gao, Kevin Lister, and Jaydev Desai. Constitutive modeling of liver tissue: experiment and theory. *Annals of Biomedical Engineering*, 38(2):505–516, 2010.
- [27] Zhan Gao, Kevin Lister, and Jaydev P. Desai. Constitutive modeling of liver tissue: experiment and theory. In *Second Biennial IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics BioRob 2008*, pages 477–482, October 2008.
- [28] Zhan Gao, Kevin Lister, and Jaydev P. Desai. Constitutive modeling of liver tissue: experiment and theory. *Annals of Biomedical Engineering*, 38(2):505–516, 2009.
- [29] Amit Gefen and Susan S. Margulies. Are in vivo and in situ brain tissues mechanically similar? *Journal of Biomechanics*, 37:1339–1352, 2004.
- [30] Stephanie Greenish, Vincent Hayward, Vanessa Chial, Allison Okamura, and Thomas Steffen. Measurement, analysis and display of haptic signals during surgical cutting. *Presence*, 11(6):626–651, 2002.
- [31] Deng-Lin Guo, Bo-Shao Chen, and Nai-Shang Liou. Investigating full-field deformation of planar soft tissue under simple-shear tests. *Journal of Biomechanics*, 40(5):1165–1170, 2007.
- [32] Arthur W. Ham. *Histology*, chapter Pancreas, Liver and Gallbladder, pages 711–717. J. B. Lippincott Company, Philadelphia and Toronto, sixth edition, 1969.
- [33] Marc Hollenstein, Alessandro Nava, Davide Valtorta, Jess G. Snedeker, and Edoardo Mazza. *Biomedical Simulation*, volume 4072 of *Lecture Notes in Computer Science*, chapter Mechanical characterization of the liver capsule and parenchyma, pages 150–158. Springer, Berlin/Heidelberg, 2006.
- [34] Gerhard Holzapfel. *Nonlinear Solid Mechanics*, chapter Hyperelastic Materials. John Wiley and Sons, LTD, West Sussex, England, 2000.
- [35] Tie Hu and Jaydev P. Desai. Modeling large deformation in soft-tissues: Experimental results and analysis. In *EuroHaptics*, 2004.

- [36] Tie Hu, C.W. Lau, and Jaydev P. Desai. Instrumentation for testing soft-tissue undergoing large deformation: ex vivo and in vivo studies. *ASME Journal of Medical Devices*, 2(4):041001, 2008.
- [37] T.S. Huang, S.D. Bolstein, and I.A. Margerum. Least-squares estimation of motion parameters from 3-d point correspondences. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1986.
- [38] G.R. Joldes, A. Wittek, and K. Miller. Suite of finite element algorithms for accurate computation of soft tissue deformation for surgical simulation. *Medical Image Analysis*, 13:12–24, 2009.
- [39] P. Jordan, S. Socrate, T.E. Zickler, and R.D. Howe. Constitutive modeling of porcine liver in indentation using 3d ultrasound imaging. *Journal of the Mechanical Behavior of Biomedical Materials*, 2:192–201, 2009.
- [40] M. Kauer, V. Vuskovic, J. Dual, G. Szekely, and M. Bajka. Inverse finite element characterization of soft tissues. *Medical Image Analysis*, 6:275–287, 2002.
- [41] Amy E. Kerdok, Mark P. Ottensmeyer, and Robert D. Howe. Effects of perfusion on the viscoelastic characteristics of liver. *Journal of Biomechanics*, 39:2221–2231, 2006.
- [42] Mirna Lerotic, Su-Lin Lee, Jennifer Keegan, and Guang-Zhong Yang. Image constrained finite element modeling for real-time surgical simulation and guidance. In *IEEE International Symposium on Biomedical Imaging: From Nano to Macro (IBSI)*, pages 1063–1066, 2009.
- [43] H. Li and N. Chandra. Analysis of crack growth and crack-tip plasticity in ductile materials using cohesive zone models. *International Journal of Plasticity*, 19:849–882, 2003.
- [44] Kevin Lister, Zhan Gao, and Jaydev P. Desai. Real-time, haptics-enabled simulator for probing ex vivo liver tissue. In *31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society EMBC 2009*, pages 1196–1199, September 2009.
- [45] Kevin Lister, Zhan Gao, and Jaydev P. Desai. A 3d in-vivo constitutive model for porcine liver: Matching force characteristics and surface deformations. In *2010 IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 656–661, September 2010.
- [46] Kevin Lister, Zhan Gao, and Jaydev P. Desai. Development of in vivo constitutive models for liver: Application to surgical simulation. *Annals of Biomedical Engineering*, 39(3):1060–1073, 2011.

- [47] Kevin Lister, Alan Lau, and Jaydev P. Desai. Towards a soft-tissue cutting simulator using the cohesive zone approach. In *33rd Annual International Conference of the IEEE EMBS*, pages 6691–6694, September 2011.
- [48] Kevin Lister, Alan Lau, and Jaydev P. Desai. Simulation of surgical cutting through cohesive zone models. In *Proceedings of the 26th International Congress and Exhibition of Computer Assisted Radiology and Surgery (CARS)*, pages 189–191, June 2012.
- [49] Mohsen Mahvash and Vincent Hayward. Haptic rendering of cutting, a fracture mechanics approach. *Journal of Haptics Research*, 2:2001, 2001.
- [50] Edoardo Mazza, Alessandro Nava, Dieter Hahnloser, Wolfram Jochum, and Michael Bajka. The mechanical response of human liver and its relation to histology: An in vivo study. *Medical Image Analysis*, 11:663–672, 2007.
- [51] C. Mendoza and C. Laugier. Simulating soft tissue cutting using finite element models. In *International Conference on Robotics and Automation (ICRA)*, pages 1109–1114, September 2003.
- [52] K. Miller, K. Chinzei, G. Orssengo, and P. Bednarz. Mechanical properties of brain tissue in-vivo: experiment and computer simulation. *Journal of Biomechanics*, 33:1369–1376, 2000.
- [53] Karol Miller. Constitutive modelling of abdominal organs. *Journal of Biomechanics*, 33:367–373, 2000.
- [54] Karol Miller and Kiyoyuki Chinzei. Mechanical properties of brain tissue in tension. *Journal of Biomechanics*, 35(4):483–490, 2002.
- [55] Karol Miller, Grand Joldes, Dane Lance, and Adam Wittek. Total lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation. *Communications in Numerical Methods in Engineering*, 23(2):121–134, 2007.
- [56] S Misra, K T Ramesh, and A M Okamura. Modeling of tool-tissue interactions for computer-based surgical simulation: a literature review. *Presence: Teleoperators and Virtual Environments*, 17(5):463–491, 2008.
- [57] Ilyas Mohammed and Kenneth M. Liechti. Cohesive zone modeling of crack nucleation at bimaterial corners. *Journal of the Mechanics and Physics of Solids*, 48:735–764, 2000.
- [58] Andrew B. Mor and Takeo Kanade. Modifying soft tissue models: Progressive cutting with minimal new element creation. In *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 598–607, 2000.

- [59] A. Nava, E. Mazza, M. Furrer, P. Villiger, and W.H. Reinhart. In vivo mechanical characterization of human liver. *Medical Image Analysis*, 12:203–216, 2008.
- [60] Matthieu Nesme, Maud Marchal, Emmanuel Promayon, Matthieu Chabanas, Yohan Payan, and François Faure. Physically realistic interactive simulation for biological soft tissues. *Recent Research Developments in Biomechanics*, 2:1–22, 2005.
- [61] Han-Wen Nienhuys and A. Frank van der Stappen. A surgery simulation supporting cuts and finite element deformation. *Lecture Notes In Computer Science*, 2208:145–152, 2001.
- [62] S. Niroomandi, I. Alfaro, E. Cueto, and F. Chinesta. Real-time deformable models of non-linear tissues by model reduction techniques. *Computer Methods and Programs In Biomedicine*, 91:223–231, 2008.
- [63] R.W. Ogden. Large deformation isotropic elasticity - on the correlation of theory and experiment for incompressible rubberlike solids. *Proceedings of the Royal Society of London Series A*, 326:565–584, 1972.
- [64] Barry P. Pereira, Peter W. Lucas, and Teoh Swee-Hin. Ranking the fracture toughness of thin mammalian soft tissues using the scissofrs cutting test. *Journal of Biomechanics*, 30(1):91–94, 1997.
- [65] I Peterlik, M Sedef, C Basdogan, and L Matyska. Real-time visio-haptic interaction with static soft tissue models having geometric and material nonlinearity. *Computers and Graphics*, 34(1):43–54, 2010.
- [66] Guillaume Picinbono, Herve Delingette, and Nicholas Ayache. Non-linear anisotropic elasticity for real-time surgery simulation. *Graphical Models*, 65(5):305–321, 2003.
- [67] Matthew A. Reilly, Gavin Perry, and Nathan Ravi. A dynamic microindentation device with electrical contact detection. *Review of Scientific Instruments*, 80(1):015105, 2009.
- [68] Esra Roan and Kumar Vemaganti. The nonlinear material properties of liver tissue determined from non-slip uniaxial compression experiments. *Journal of Biomechanical Engineering*, 129:450–456, 2007.
- [69] Jacob Rosen, Jeffery Brown ans Smita De, Mika Sinanan, and Blake Hannaford. Biomechanical properties of abdominal organs in vivo and postmortem under compression loads. *Journal of Biomechanical Engineering*, 130(2):021020, 2008.
- [70] Mark Sagar, David Bullivant, Gordon Mallinson, and Peter Hunter. A virtual environment and model of the eye for surgical simulation. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 205–212, 1994.

- [71] Evren Samur, Mert Sedef, Cagatay Basdogan, Levent Avtan, and Oktay Duzgun. A robotic indenter for minimally invasive measurement and characterization of soft tissue response. *Medical Image Analysis*, 11:361–373, 2007.
- [72] H. Saraf, K.T. Ramesh, A.M. Lennon, A.C. Merkle, and J.C. Roberts. Measurement of the dynamic bulk and shear response of soft human tissues. *Experimental Mechanics*, 47:439–449, 2007.
- [73] Guillaume Saupin, Christian Duriez, and Stephane Cotin. Contact model for haptic medical simulations. *Lecture Notes In Computer Science*, 5104:157–165, 2008.
- [74] D. Serby, Matthias Harders, and Gabor Szekely. A new approach to cutting into finite element models. In *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 425–433, 2001.
- [75] C. Shet and N. Chandra. Effect of the shape of $t - \delta$ cohesive zone curves on the fracture response. *Mechanics of Advanced Materials and Structures*, 11:249–275, 2004.
- [76] Dassault Systemes, editor. *Abaqus Theory Manual*, chapter Mechanical Constitutive Theories. Dessault Systemes, 2010.
- [77] G. Szekely, M. Bajka, Ch. Brechbuhler, J. Dual, R. Enzler, U. Haller, J. Hug, R. Hutter, N. Ironmonger, M. Kauer, V. Meier, P. Niederer, V. Vuskovic, and G. Troster. Virtual reality based surgery simulation for endoscopic gynaecology. *Presence*, 9:310–333, 2000.
- [78] Z.A. Taylor, A Comas, M. Cheng, J. Passenger, D.J. Hawkes, D. Atkinson, and S. Ourselin. On modeling of anisotropic viscoelasticity for soft tissue simulation: Numerical solution and gpu execution. *Medical Image Analysis*, 13:234–244, 2009.
- [79] Davide Valtorta and Edoardo Mazza. Dynamic measurement of soft tissue viscoelastic properties with a torsional resonator device. *Medical Image Analysis*, 9:481–490, 2005.
- [80] M.J. van den Bosch, P.J.G. Schreurs, and M.G.D. Geers. An improved description of the exponential xu and needleman cohesive zone law for mixed-mode decohesion. *Engineering Fracture Mechanics*, 73:1120–1234, 2006.
- [81] M.J. van den Bosch, P.J.G. Schreurs, and M.G.D. Geers. A cohesive zone model with a large displacement formulation accounting for interfacial fibrillation. *European Journal of the Mechanics*, 26:1–19, 2007.
- [82] J.G. Williams and H. Hadavinia. Analytical solutions for cohesive zone models. *Journal of the Mechanics and Physics of Solids*, 50:809–825, 2000.

- [83] Xunlei Wu, Michael Sownes, Tolga Goktekin, and Frank Tendick. Adaptive nonlinear finite elements for deformable body simulations using dynamic progressive meshes. *Computer Graphics Forum*, 20(3):349–358, 2001.