University of Twente

EEMCS / Electrical Engineering
Robotics and *Mechatronics*

Finite Element Modelling of Prostate Deformation and Needle-Tissue Interactions

M. (Mark) Herink

MSc Report

Committee:

Prof.dr.ir. S. Stramigioli
Dr. S. Misra
A. Jahya, MSc
Dr.ir. J.J. Homminga

January 2013

Report nr. 004RAM2013

Robotics and Mechatronics
EE-Math-CS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Contents

1 Introduction **1**

2 Finite Element Modelling of Prostate Deformation and Needle-Tissue Interactions **4**

A Model preparation **13**

Model preparation

A.1 Introductory 13
A.2 Overview of mould 13
A.3 Time line 13
A.4 Removing (parts of the) mould 17
A.5 Adding markers 17
A.6 Stiffness verification 17
A.7 Overview of material composition 19

B Marker tracking **21**

B.1 Pseudo-code 21
B.1.1 Movement 21
B.1.2 Data acquisition 21
B.1.3 Data output 22
B.1.4 Notes 22

C Matlab scripts **23**

C.1 Compare experimental, Abaqus and SOFA results 23
C.2 Create scene file and *.msh files 23

D SOFA Installation manuals **25**

D.1 SOFA CUDA installation 25
D.1.1 Notes 27
D.2 SOFA Boost Installation 27

E SOFA Model **31**
E.1 Introduction . . . 31
E.2 Mesh . . . 31
E.3 Model . . . 32
E.3.1 Model structure . . . 33
E.3.2 Constraints . . . 33
E.3.3 Collision . . . 34
E.3.4 Attach constraints . . . 34
E.3.5 Material properties . . . 36
E.4 Optimization . . . 36

F Used materials **39**
F.1 Gelatine . . . 39
F.2 Wacker SilGel . . . 40
F.3 PVC . . . 40

G Adding markers **43**
G.1 Spray paint . . . 43
G.2 Paint . . . 43
G.3 Laser cutting . . . 43
G.4 Holes . . . 44

H Running experiments **45**
H.1 Prostate movement . . . 45
H.2 Needle insertion . . . 47

I Abaqus model **49**
I.1 Geometry import and mesh . . . 49
I.2 Boundary conditions . . . 49
I.3 Results . . . 49

J Friction experiments **53**
J.1 Friction . . . 53
J.2 Penetration force . . . 53

1

Introduction

Prostate cancer is a hot topic in the medical world. In the Netherlands, the most common type of cancer for men is prostate cancer (20% of total). The disease mostly affects elderly men. Since the 1990's, the increase of prostate cancer is best noticable in the group of men between 50 and 59. The diagnosis of cancer is also better than before: nowadays the diagnosis is often localised prostate cancer while the diagnoses of a metastatic disease have decreased.

To diagnose prostate cancer, TransRectal UltraSound-guided (TRUS) needle biopsy is nowadays the best method. TRUS-guided biopsy is a relative inexpensive diagnostic test and is operator dependent. Several systematic samples of the prostate are taken during the biopsy, mostly of the locations where prostate cancer occurs most often. Figure 1.1 [1] shows how such an ultrasound probe is used. For biopsy a biopsy gun is mounted above the ultrasound probe and the position of the needle tip is tracked using the probe.

For the treatment of prostate cancer, brachytherapy can be applied. This method uses the same TRUS method but instead of taking a sample, the operator puts small radioactive seeds inside the prostate to treat the cancer. This is a difficult and time-consuming process and requires an experienced operator.

[1] http://www.weisshospital.com/Files/Images/biopsy.jpg

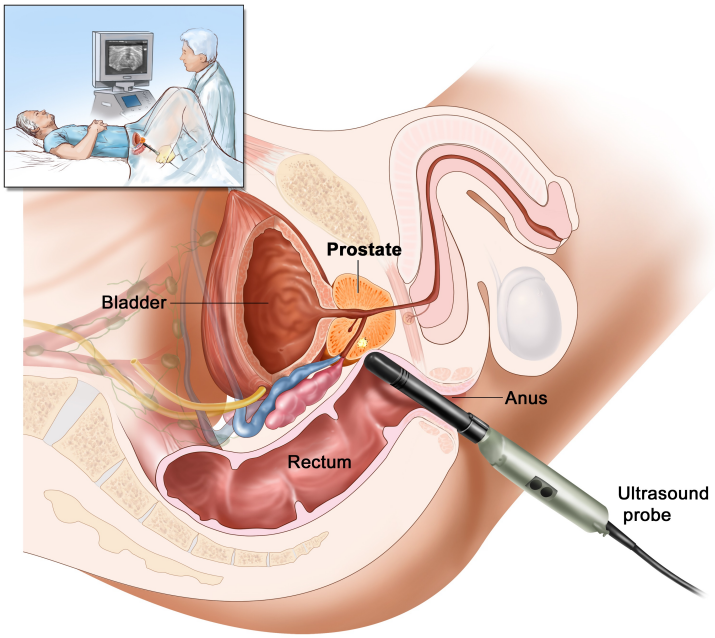

Figure 1.1: Schematic overview of the use of TRUS

Since both TRUS-guided biopsy and brachytherapy are highly dependent on the skills of the operator, creating a model which predicts the movement of the prostate and deflection of the needle could increase the speed and accuracy. In this study it has been tried to create two Finite Element (FE) models. One model calculates the movement of the prostate due to a needle guide in a model, the other model predicts the deflection of a needle with an asymmetric bevel tip inside the prostate. An overview of this model is shown in Figure 1.2. If these models run real-time, it is possible to automate the process and use the operator for another procedure.

This thesis consists of a main thesis in paper format which summarizes methods and results of this study to create a real-time FE model of the prostate and its surrounding organs. Furthermore, 10 appendices are available which describe the experiments, data analysis and FE models in more detail.

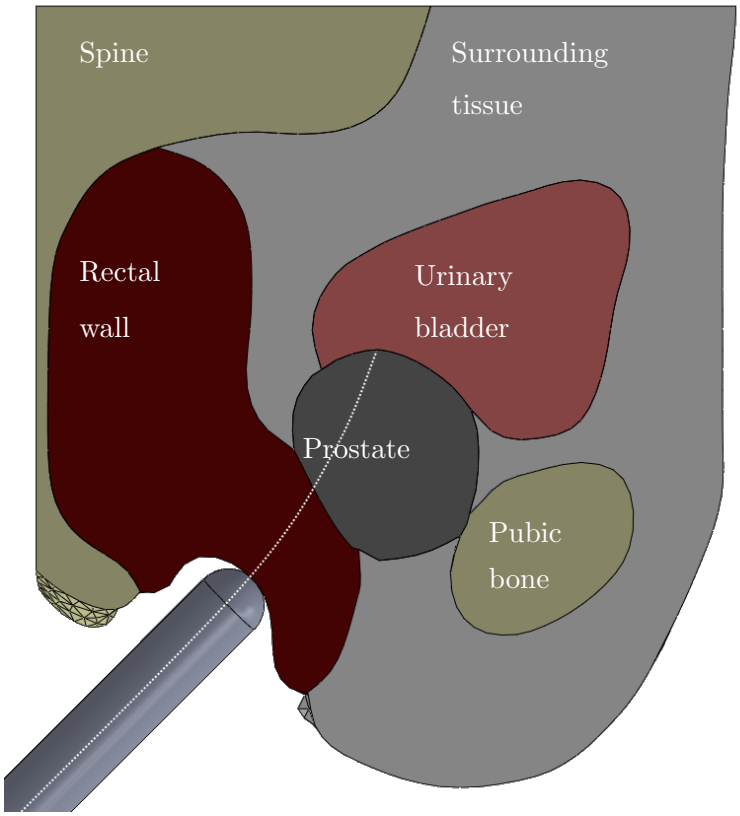

Figure 1.2: Top view of the prostate and its surrounding tissues, a needle penetrates the rectal wall and prostate (dashed line)

Finite Element Modelling of Prostate Deformation and Needle-Tissue Interactions

M. Herink
University of Twente, The Netherlands

***Abstract*—During brachytherapy and biopsy, significant prostate motion (including deformation) can occur, causing the target lesion to move during the procedures. One method to improve the accuracy of needle tip placement during these percutaneous procedures is to use a 3D Finite Element (FE) model to estimate the amount of needle deflection. This model is based on the available mechanical properties of the material (shape, stiffness and density). In this study, the needle deflection during brachytherapy will be modelled using an FE model and compared with experimental results.**

In order to predict a correct needle deflection using FE, boundary conditions have to be defined accurately. Anatomically correct prostate models that include urethra can improve the accuracy of the FE model. However, computational time of an FE model increases with the complexity of the model. The first aim of this project is to make a real time FE model of the prostate and the surrounding tissues and deform it using a needle guide. During the experiments, several markers are put on top of the prostate, will be tracked and compared with the FE model. Finally, a needle is inserted and the deflection will be compared between the experimental results and an FE model.

To create an FE model which runs in real-time, the open source framework SOFA has been used. SOFA targets at real-time medical simulations and allows the user to model the deformation of the prostate. The needle insertion is done using a plug-in which enables SOFA to calculate needle deflection. This study shows that both prostate deformation and needle insertion can be predicted which can be used for e.g. autonomous robotic surgery.

I. Introduction

A. Previous works

Prostate cancer is a disease which costs many men in developed countries their lives [1], [2]. Biopsy and brachytherapy are two procedures which require needle insertion. These methods are used to diagnose, and treat prostate cancer by placing small radioactive sources within the prostate, respectively [2], [3], [4], [5]. If a tumor has been discovered in an early stage, the size of the sample is rather small [6]. Precise needle placement is required to place the seeds and sometimes multiple insertions at different angles have to be performed [7] [8]. In this paper, the goal is to predict the movement of the prostate due to a needle guide and predict the needle placement.

Since Terzopoulos *et al.* started modelling soft tissue deformations [9], several studies have shown that it is possible to model deformation of soft tissue using an FE model [10], [11]. To simulate the deformation of a prostate, the model should consist of multiple body parts since the deformation of the prostate depends on accurate modelling of geometry and boundary conditions [12]. Needle-tissue interactions have been researched [13] and an overview has been written by Abolhassani *et al.* [14]. The target of this paper is to investigate the feasibility of SOFA to predict prostate deformation and needle insertion. The results of the SOFA (RC1, INRIA, Rocquencourt, France) model are compared with an Abaqus (6.12, Dassault Systèmes, Vélizy-Villacoublay Cedex, France) model and experimental data.

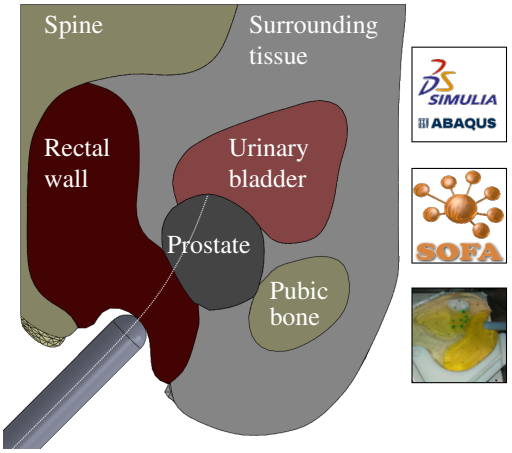

Fig. 1. Top view of the prostate and its surrounding tissues, a needle penetrates the rectal wall and prostate (dashed line)

SOFA is an Open Source framework primarily targeted at real-time simulation, with an emphasis on medical simulation [15]. Several models have been made in SOFA [16] [17], including interactive needle insertion. Duriez *et al.* [18] focused on the simulation of needle insertion in geometries with different elasticities which are not based on human tissue geometries. The model used in our paper is derived from a Magnetic Resonance Imaging (MRI) scan since this method visualises soft tissue contrast better than ultrasound [19] [20]. Each of the geometries found by the MRI scan have their own stiffness, just like in the real human body [12], [21], [22].

B. Novelty of the work

This research combines several of the above mentioned studies and more: FE models, needle insertion, MR images based geometries, experimental data and soft-tissue deformation. Where current research stops at combining two or three of these topics, this study combines all five. In this study, two FE models based on MR images of the prostate and its surrounding which consists of multiple soft-tissue geometries are created. Both models simulate the deformation of the prostate due to a needle guide and needle insertion. The Abaqus model will be used as the truth model for the prostate deformation since Abaqus is proven to be capable of modelling soft tissues [23], [24], [25]. For the needle insertion, experimental data and a SOFA model are used since an Abaqus model is not created yet. The experimental data will be acquired using a model which is based on the same MR images and includes several soft tissues with different elasticities. These elasticities are verified using the Acoustic Radiation Force Impulse (ARFI) technique, like Assaad *et. al* did in [26].

C. Summary

In Chapter II, the method is discussed in which the experiments and models will be explained. Chapter III shows the results, followed by a discussion with a brief error analysis. Finally, Chapter IV lists recommendations and reveals ideas for future work.

II. Method

The method section contains 3 parts, first the experiments are described, followed by a description of model in Abaqus. The other FE model is mentioned in the last paragraph.

A. Experiments

1) Model: To make it possible to verify the experiments, a flat model has been made of the prostate and the surrounding organs, as shown in Figure 1. To create a model which uses the geometries of an MRI scan, the scan has been post processed by triangulating the images. ScanIP (Simpleware, Exeter, United Kingdom) is used to do this triangulation, the sides of the triangles have an average length of 4.5 mm. A finer mesh would not add significant detail and gives more possibilities to mesh it.

Using this output, a mould has been designed using SolidWorks (SolidWorks 2012, Dassault Systèmes, Vélizy-Villacoublay Cedex, France). Using this mould, each geometry can have different material properties. The model has a height of 16 mm. The bottom plate of the mould has small bumps so the prostate gets small cubical cavities to add markers. Figure 2 shows the experimental model after using the mould.

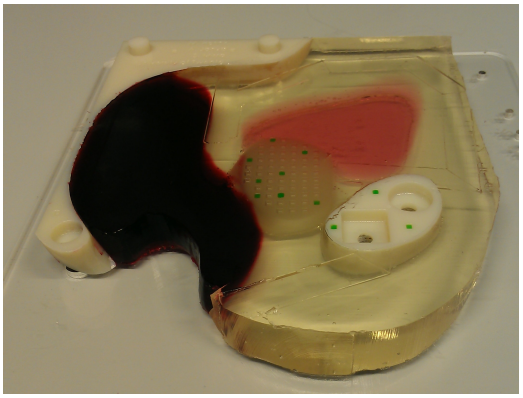

Fig. 2. Phantom used for experiments in which the markers and different elasticities are visible

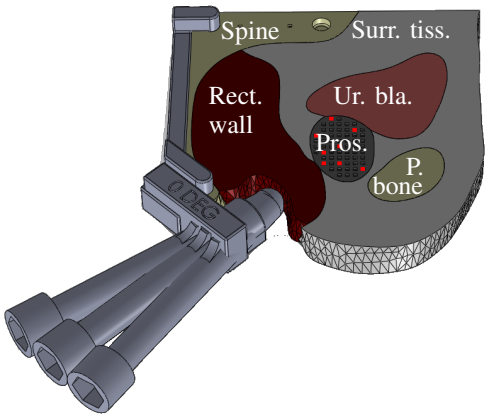

Fig. 3. Needle guide positioner at 0° deg, also the -10° deg and +10° deg needle guides are shown. The markers on the prostate are shown in red

This mould enables us to give the four soft parts different stiffness properties. The model has been printed with a 3D printer (Objet EDEN 250, Stratasys Ltd., Rehovot, Israel). The MRI images have been modified by increasing the thickness of the model from 9 mm to 16 mm. This has been done since the needle guide, by which the displacement is caused, has a thickness of 16 mm. By making the phantom as thick as the needle guide, the needle guide is easily positioned at the correct height. The position of the needle guide was determined by printing additional parts which can be put on the spine part of the phantom. This part has a tunnel which constraints the needle guide in the remaining directions, Figure 3 shows this part connected to the phantom. For more information see Chapter II.

Another modification of the model is that there are markers added on the pubic bone and spine. This is done to verify that some parts of the model are fixed; the bone parts should not move.

2) Marker detection: The phantom has several markers to visualize the movement of the prostate. This movement

is quantified by an imaging system. A program has been written to get the positions of the markers using a Sony XCD-X710CR colour camera (Sony Corporation, Tokyo, Japan). The resolution of this camera is 1024 x 768 pixels. The height of the camera and the lens (type 16 mm 1:1.6 TV lens, PENTAX RICOH IMAGING CO., LTD., Tokyo, Japan) make a pixel to mm ratio of 0.175. This is calculated by measuring the distance (in mm) between the first pixel and the last pixel of one line of the camera image at the height of the markers.

The markers are made green to easily distinguish them from the rest of the model. The camera software uses a region of interest to reduce the processing time, filters by hue values, erodes the image and calculates the centre of mass of the markers. The program is written in C++ using the OpenCV library (version 2.3.1) [27] [28]. The markers are numbered and the position at each time step is printed to a text file for post processing. The program consists of multiple threats and is therefore also capable of controlling the position of the needle guide.

3) Needle and needle guide movement: The same program as described above, not only tracks the markers but also controls the motor which drives the needle guide. The needle guide is positioned using the part described previously in Section II. Once the needle guide is in position, it is pushed 5 mm to the rectal wall and therefore deforming the prostate. The velocity of the needle guide is 0.5 mm/s. The needle guide is pressed from five different angles to the rectal wall: -10°, -5°, 0°, 5° and 10° degrees. These angles are relative to the original needle guide position which is rotated 45° with respect to the long side of the spine. The same setup is used to drive the needle inside the phantom with a velocity of 1 mm/s. The needle is positioned using the needle guide which is put at the desired position using the same needle guide position system.

The needle guide and needle are moved by the setup used by Roesthuis *et al.* [29]. This setup has two Degrees Of Freedom (DOF): a translational along the axis of the needle and a rotation about the same axis. For the experiments described in this paper, only the translational DOF is used. If the needle is inserted, another device is used to position the needle guide which guides the needle. The needle is inserted for 60 seconds with a velocity of 1 mm/s.

4) Needle bending quantification: During the insertion of the needle, an ultrasound probe is used to track the needle. The used ultrasound machine is a Siemens Acuson 2000 (Siemens AG, Erlangen, Germany) with an 18L6 HD probe. This setup is also used to make a sweep after the needle is inserted into the phantom [30].

5) Materials: The used needle is made from nitinol (E = 75 GPa) [31]. This solid needle has a diameter of 1 mm and a bevel tip of 45 degrees. The bevel tip-gel interaction results in needle deflection.

For the phantom, two materials have been used: gelatine and PVC. Gelatine (type Gelatinepoeder, Dr. Oetker, Amersfoort, The Netherlands) was used because of its availability and easy adjustable stiffness. Gelatine, however, only lasts for several hours/days. After that, it starts to decompose which has not the original dimensions and stiffnesses anymore. The markers in the gelatine model are made by colouring gelatine with the same Young's modulus as the prostate. Once the phantom is prepared, the holes are filled with this green coloured gelatine.

PVC (types Assouplissant Plastileurre, Plastileurre Standard and Plastileurre Super Hard, Bricoleurre, Mont-Saint-Aignan, France), was chosen to be its replacement for the long term, PVC lasts longer but requires a little different preparation method. The markers on the PVC model are created using a permanent marker (type green permanent marker medium, 3M Corporate, St. Paul, Minnesota). Table I shows the ingredients and ratio's of the used materials.

The stiffness of the soft materials is verified by measuring the shear wave velocity [32] using the same Siemens Acuson 2000 ultrasound machine. The shear wave velocity c_T relates as follows to the shear modulus G and density ρ of the tissue:

$$c_T = \sqrt{\frac{G}{\rho}} \tag{1}$$

Using the following relation between Young's modulus E, G and Poisson's ratio μ

$$E = 2 \cdot G \cdot (1 + \mu) \tag{2}$$

Results in the following relation between E and c_T:

$$E = 2 \cdot \rho \cdot c_T^2 \cdot (1 + \mu) \tag{3}$$

For the prostate deformation, both gelatine and PVC have been used, for needle insertions only gelatine is used. Now the experiments are explained, the reference FE model will be discussed.

B. Abaqus

To make a Finite Element truth model, Abaqus is used. First the model is imported into Abaqus, see Figure 4. Before this leads to reliable results, the basic FE steps have to be done.

Several meshes have been tested to determine the correct mesh size. The final mesh is shown in Figure 5. This mesh size (10-node quadratic tetrahedron elements) is chosen based on the amount of change of the results with decreasing element size. The mesh is generated automatically by giving a preferred element size (seed) of 3, the needle guide is meshed with a seed of 0.5 to maintain the round shape. The

Geometry	Ideal		Gelatine			PVC			
	E [KPa]	Poisson's ratio []	Gelatine [%]	Water [%]	Silica [%]	Softener [%]	Standard [%]	Super Hard [%]	Silica [%]
Rectal wall	172	0.495	25.7	74.3	1.0	0	10	90	1.0
Urinary bladder	98	0.495	19.0	81.0	1.0	0	90	10	1.0
Prostate	75	0.495	17.0	83.0	1.0	5	95	0	1.0
Surrounding tissue	10	0.495	5.3	94.7	1.0	50	50	0	1.0

TABLE I
USED SOFT MATERIALS FOR THE PHANTOM

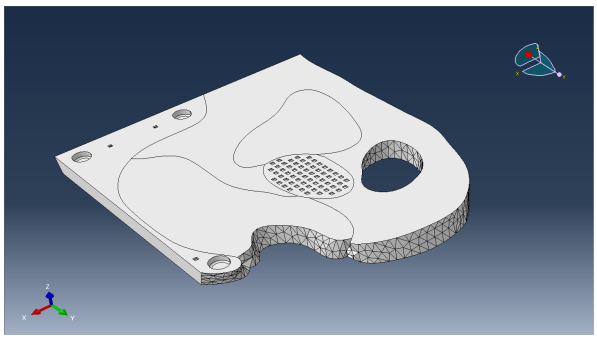

Fig. 4. Imported model in Abaqus

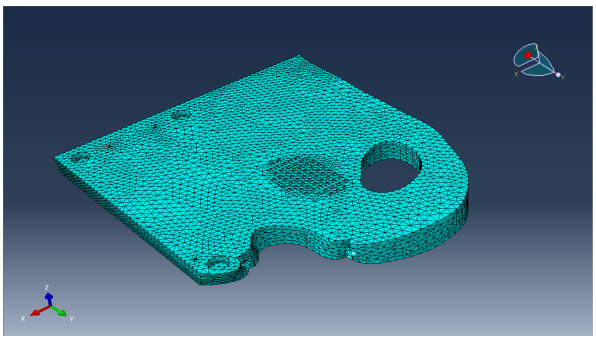

Fig. 5. Meshed model in Abaqus

needle guide is modelled as a hemisphere in order to reduce the amount of elements.

1) Boundary conditions: The model has 3 boundary conditions:

- Frictionless constraints
- Fixed constraints
- Displacement

The frictionless constraint is for the bottom of the phantom, this might not be the closest to reality but everything in between frictionless and fixed is difficult to use or determine in an experiment. Since the prostate should move because of the movement of the needle guide, frictionless is preferred. The node selection criteria for this constraint is the z component. If the z coordinate of a node is 0, then frictionless support is applied.
The fixed constraints are used to fix the soft tissue to the bone parts.
The displacement constraint is used for the needle guide, it moves 5 mm towards the rectal wall at 5 different angles.

The biggest drawback of this simulation is the required time: 2 hours for a 5 mm displacement. Once the simulation is done, the 'XYData' option in the results section of Abaqus is used to get the displacement of each individual marker printed to a text file. The way of creating an Abaqus model is - except for the user interface - almost the same for every FE package. In the next section, a less known framework will be used to create a faster FE model. This framework is called SOFA.

Geometry	In plane [mm]	Out of plane [mm]
Rectal wall	160	1895
Prostate	183	1566
Surrounding tissue	1305	15000

TABLE II
NEEDLE BENDING RADII OF CURVATURE OF A NITINOL NEEDLE OF 1 MM DIAMETER AND A 45 DEGREES BEVEL TIP IN GELATINE

C. SOFA

1) Prostate deformation: SOFA is an Open Source framework primarily targeted at real-time simulation, with an emphasis on medical simulation [33]. Since creating a model in SOFA is different than most commercial FE packages, the SOFA model will be discussed in more detail.

As the target of SOFA is real-time simulation, the simulator should be fast. One feature is that the models are interactive: users can use the mouse or a haptic device to 'touch' and influence the model. SOFA has, however, not its own meshing algorithm, nor advanced post-processing tools. For post-processing, there is an option to write the desired outputs to a text file which can be used as an input for another program. The mesh (4-node tetrahedron) is generated by using a custom ANSYS Parametric Design Language (APDL) script for ANSYS (ANSYS 14.0, ANSYS Inc., Canonsburg, Pennsylvania USA) and a Matlab (v7.13, Mathworks Inc., Natick, USA) script.

SOFA uses so called scene (*.scn) files, these files are written in XML. These scene files can be modified by hand or the so called 'Modeler' can be used, this is a Graphical User Interface (GUI) to build scene files. It's basically a library which knows all the available solvers, constraints,

forcefields, mesh loaders, elements etc. The first model was built using Modeler, once the process was clear, a Matlab script was built to create automatically a model which meets the experiments. This Matlab script uses the intersection points between the different geometries (urinary bladder, rectal wall etc.) to create a scene file which is used by SOFA. The results are printed to several text files which are analysed by Matlab. To model needle insertion, a special plug-in is used.

2) Needle insertion: Most of the FE simulations involving needle insertion use either element deletion or element splitting [34] [35]. Element deletion means that an element is deleted if the stress is too high. A drawback of this method is that the crack propagation in terms of both orientation and velocity is highly mesh-dependent [36]. Another drawback is that the elements should be very small because otherwise there will be holes when an elements is deleted. Once an element is deleted, it lets the remaining nodes move a little bit in the wrong direction. An increasing amount of elements require more computational time which is not desirable. During element splitting, the risk of creating odd shaped elements exists, this might lead to unreliable results. This can be solved by remeshing but this also takes time. The main reason for these methods for being slow is that these methods - based on topology changes - usually require modification of the system matrix. The matrix changes due to added/suppressed nodes. This process can be time consuming and thus incompatible with real/time computations, a different approach has been implemented in SOFA.

First, the needle is physically modelled as a series of linked 1D beam elements, based on Timoshenko beam theory, whereas the tissue is represented by a 3D mesh of tetrahedra elements, on which an elastic FE model is built based on corotational formulation. Second, the interaction between the needle and tissue is modelled using a constraint-based approach [18]: as the needle is being inserted into the tissue, kinematic constraints are being created along the needle axis. There are two types of constraints: *sliding constraints* are placed along the shaft of the needle, constraining the lateral motion of the needle w.r.t. the tissue whereas a *tip constraint* is coupled with the tip of the needle, determining the direction of the needle motion.

From the numerical point of view, in each step of the simulation, first a predictive motion of both bodies (needle and tissue) is computed based on constraint resolution computed in the previous time step. Based on the predictive motion, a violation is updated for each constraint. Further, a constraint resolution is iteratively computed within a Gauss-Seidel method, resulting in a set of Lagrange multipliers. These are finally applied in a correction phase using the compliance matrix (which is homogeneous to the inverse of stiffness matrix), leading to a configuration where all the constraints are satisfied. This approach proved to be fast and computationally efficient enough for a wide range on tool-tissue interactions allowing also for the haptic interaction [37].

During the constraint resolution, the dynamic friction along the shaft is taken into account, being determined by friction coefficient specified for the sliding constraints. Similar parameter called 'cutting force' can be specified for the tip constraint; it represents the force needed to cut the tissue as the needle advances. In order to model needles with an asymmetric bevel, a geometric parameter can be specified for the tip constraint resulting in a 'deviation' of the needle tip, thus leading to curved-motion of the needle (i.e. needle steering). To model needles with different steering curvature, the tip deviation parameter can be adjusted using experimental data.

The radii of curvature were determined by doing experiments in gelatine blocks with the same needle insertion device as described above. These results are shown in Table II. In SOFA, the tip deviation parameter of the needle insertion plug-in is per geometry adjusted to get the same behaviour as the needle bending experiments. Once the parameters are found, they are used in the complex geometry.

D. Analysis and summary

Matlab is used to analyse the results from both experiments and computational models. For the experiments, Matlab applies a lens correction function to compensate for lens distortions. This calibration matrix is generated using a checkerboard [38].

For the rest of the post processing, Matlab reads the generated text files and puts them in a matrix. These matrices are then compared with each other. So, a lot of steps have been done to be able to compare the experimental results with the FE results. A brief overview is shown in Figure 6.

III. RESULTS

Experiments to determine the deformation of the prostate have been done with both a model made of gelatine and PVC. Not all the markers are compared, only a small selection. This selection of markers is shown Figure 7.

The experimental results of the gelatine phantom are shown in Table III. The results of the PVC phantom are shown in Table IV. The displacements of the markers of the PVC model are generally a little bit smaller than the gelatine markers. This is most probably caused by the shrinkage of PVC. The used PVC shrinks a little bit after being processed which results in a slightly smaller phantom. Because the needle guide positioning system does not account for this effect, the needle guide might not touch the PVC phantom as much as it touches the gelatine rectal wall.

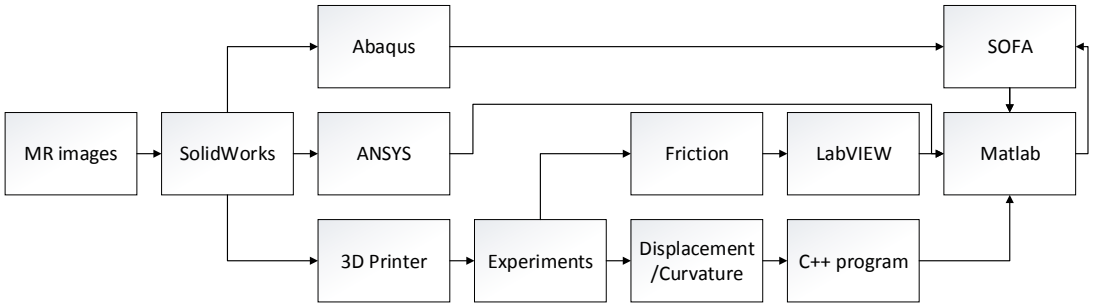

Fig. 6. Work flow

	Insertion angles [°]				
#	-10	-5	0	5	10
1	1.47 ±0.04	1.50 ±0.04	1.55 ±0.03	1.48 ±0.05	1.44 ±0.05
2	1.82 ±0.02	1.77 ±0.04	1.87 ±0.02	1.77 ±0.04	1.69 ±0.03
3	2.37 ±0.01	2.28 ±0.03	2.40 ±0.03	2.31 ±0.03	2.24 ±0.02
4	1.39 ±0.02	1.50 ±0.04	1.57 ±0.03	1.50 ±0.05	1.52 ±0.04
5	1.90 ±0.02	1.99 ±0.04	1.97 ±0.01	2.01 ±0.04	1.99 ±0.02
6	2.63 ±0.02	2.65 ±0.03	2.62 ±0.01	2.70 ±0.03	2.66 ±0.04
7	2.21 ±0.02	2.29 ±0.03	2.22 ±0.01	2.38 ±0.03	2.37 ±0.03
8	1.37 ±0.01	1.56 ±0.02	1.41 ±0.01	1.55 ±0.06	1.64 ±0.02

TABLE III
AVERAGE ABSOLUTE DISPLACEMENT OF MARKERS (FIG. 7) IN [MM] DURING 5 EXPERIMENTS WITH A GELATINE PHANTOM FOR DIFFERENT INSERTION ANGLES

	Insertion angles [°]				
#	-10	-5	0	5	10
1	1.34 ±0.01	1.33 ±0.02	1.40 ±0.02	1.39 ±0.06	1.29 ±0.02
2	1.63 ±0.01	1.59 ±0.01	1.70 ±0.02	1.67 ±0.05	1.56 ±0.01
3	2.16 ±0.01	2.13 ±0.02	2.18 ±0.02	2.16 ±0.05	2.13 ±0.03
4	1.30 ±0.01	1.30 ±0.01	1.35 ±0.01	1.38 ±0.05	1.32 ±0.03
5	1.79 ±0.01	1.76 ±0.01	1.84 ±0.02	1.89 ±0.05	1.85 ±0.01
6	2.39 ±0.01	2.50 ±0.01	2.65 ±0.01	2.58 ±0.04	2.62 ±0.05
7	1.98 ±0.01	2.11 ±0.02	2.14 ±0.03	2.18 ±0.04	2.25 ±0.02
8	1.24 ±0.01	1.33 ±0.01	1.44 ±0.03	1.44 ±0.04	1.46 ±0.01

TABLE IV
AVERAGE ABSOLUTE DISPLACEMENT OF MARKERS (FIG. 7) IN [MM] DURING 5 EXPERIMENTS WITH A PVC PHANTOM FOR DIFFERENT INSERTION ANGLES

	Insertion angles [°]				
#	-10	-5	0	5	10
1	1.55	1.55	1.58	1.58	1.56
2	1.81	1.83	1.84	1.84	1.81
3	2.20	2.23	2.27	2.27	2.26
4	1.49	1.52	1.54	1.54	1.53
5	1.89	1.93	1.96	1.97	1.96
6	2.45	2.51	2.56	2.58	2.58
7	2.07	2.13	2.18	2.21	2.22
8	1.28	1.33	1.37	1.39	1.40

TABLE V
ABSOLUTE DISPLACEMENT OF MARKERS IN [MM] CALCULATED BY ABAQUS FOR DIFFERENT INSERTION ANGLES

	Insertion angles [°]				
#	-10	-5	0	5	10
1	1.79	1.78	1.77	1.78	1.79
2	2.14	2.12	2.11	2.13	2.13
3	2.66	2.63	2.61	2.61	2.61
4	1.73	1.71	1.70	1.70	1.71
5	2.26	2.23	2.22	2.21	2.21
6	2.99	2.95	2.92	2.90	2.90
7	2.53	2.49	2.45	2.44	2.42
8	1.56	1.52	1.49	1.48	1.47

TABLE VI
ABSOLUTE DISPLACEMENT OF MARKERS IN [MM] CALCULATED BY SOFA

The movements of the markers on the pubic bone are not shown in the tables, the maximum total displacement of these markers is 0.12 mm where the average displacement is 0.058 mm with an average standard deviation of 0.01. Since this is lower than the resolution of the camera system it can be considered fixed.

The phantom results can be compared with the computational models of Abaqus and SOFA. The results of these models are shown in Table V and VI, respectively. The difference between these two models is most probably caused by the mesh difference and because of the proximity detection in SOFA which makes the probe 0.4 mm in diameter larger. This leads to a larger contact point and therefore the net result is that more of the phantom will move. Also, since the SOFA model has a completely different mesh (element type and number of elements), it is expected that the result is different. The relative error comparison is done by comparing all the results with the Abaqus simulation, which is used as the truth model. Only the maximum percentage differences are given. The maximum relative errors are calculated using the following formula:

$$e_{relative} = max \left| \frac{Disp_{Abaqus} - Disp_{Other}}{Disp_{Abaqus}} \right| \quad (4)$$

Using this relation (Disp means displacement), Table VII is produced, Table VIII shows the maximum absolute displacements. As mentioned before, the Abaqus model takes 2 hours to compute. The SOFA model does it within 20 minutes. These results are shown in IX.

Compared to	Maximum relative error [%]	Angle	Marker
SOFA	22.2	-10	7
Gelatine	17.29	-5	8
PVC	17.49	+10	1

TABLE VII
RELATIVE ERRORS COMPARED TO THE ABAQUS MODEL

Compared to	Maximum absolute error [mm]	Angle	Marker
SOFA	0.54	-10	6
Gelatine	0.24	+10	8
PVC	0.27	+10	1

TABLE VIII
ABSOLUTE ERRORS COMPARED TO THE ABAQUS MODEL

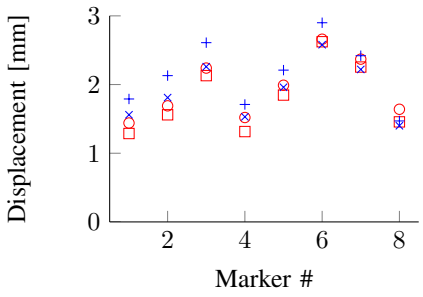

Fig. 8. Average absolute displacements of the markers at plus 10 Deg, PVC = red □ (mean std. 0.02), gelatine = red o (mean std. 0.03), Abaqus = blue x, SOFA = blue +

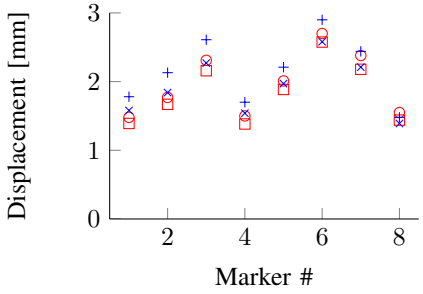

Fig. 9. Average absolute displacements of the markers at plus 5 Deg, PVC = red □ (mean std. 0.05), gelatine = red o (mean std. 0.04), Abaqus = blue x, SOFA = blue +

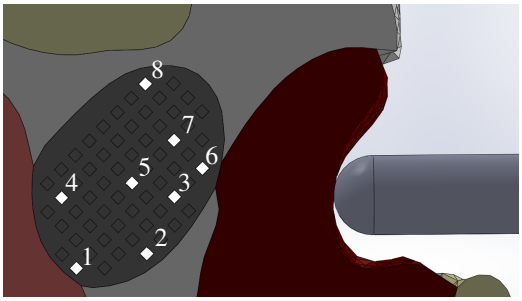

Fig. 7. Selection of markers

Program	Prostate deformation [s]	Needle insertion [s]
Abaqus	7200	XX
SOFA	1200	60

TABLE IX
CALCULATION TIMES FOR DIFFERENT MODELS AND PROGRAMS

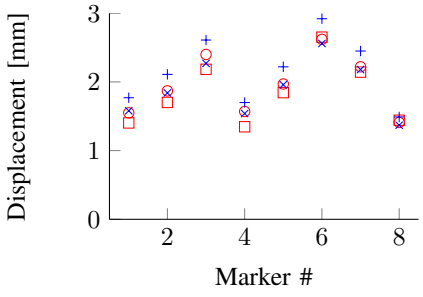

Fig. 10. Average absolute displacements of the markers at 0 Deg, PVC = red □ (mean std. 0.02), gelatine = red o (mean std. 0.02), Abaqus = blue x, SOFA = blue +

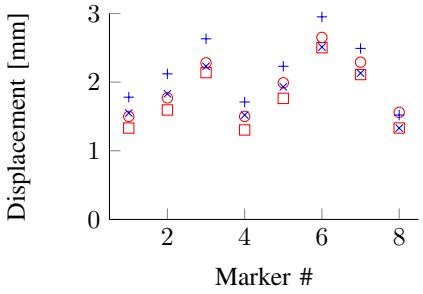

Fig. 11. Average absolute displacements of the markers at min 5 Deg, PVC = red □ (mean std. 0.01), gelatine = red o (mean std. 0.03), Abaqus = blue x, SOFA = blue +

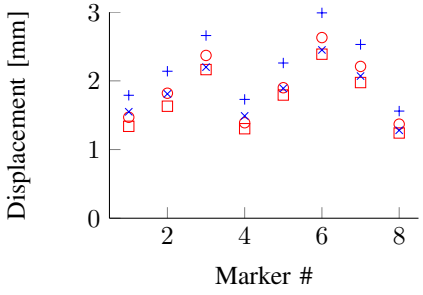

Fig. 12. Average absolute displacements of the markers at min 10 Deg, PVC = red □ (mean std. 0.01), gelatine = red o (mean std. 0.02), Abaqus = blue x, SOFA = blue +

1) Needle insertion: SOFA is used to model needle insertion. This model uses the same plug-in as Duriez *et. al* used [18]. This plug-in creates constraints within the material for the flexible needle. To model the needle, the plug-in uses a tip deviation parameter which determines the bending of the needle. For our paper the radius of curvature is used,

the friction and penetration force are not taken into account, this could be used in future research. The modelled needle has a length of almost 80 mm and is driven 60 mm inside the phantom. This result was calculated in 60 seconds, just as long the needle insertion took. The results of both the experiment and SOFA model are shown in Figure 13 and Table IX. The result of this simulation can be varied by changing the time step. For now, the computer can calculate one time step per second. Using a time step (dt) of 1 second results in an insertion of 1 mm per second, a smaller time step would result in a longer simulation time.

The two plots are almost similar, the difference is maximum at an insertion depth of 19 mm: 0.4 mm. The small difference between the plots might be caused by different properties of the gelatine: the blocks used to get the radii of curvature was from another batch than the actual gelatine phantom. This process has to be improved. Another method is to use for instance PVC for modelling: this material has a higher sustainability and the temperatures are easier to match since this material lasts much longer outside the refrigerator.

Before it is possible to use this method in a human body, a large library of needles and material properties should be created. SOFA is not calculating the actual needle-tissue interaction but adding constraints to the needle. This limits the use for unknown materials but once the properties of the material are determined the model can be used. The properties of an unknown material can be found by using for instance the ARFI technique, it is possible to create a phantom to determine the properties (radius of curvature) of the material. Once this table is built, it does not have to be updated unless other needles are used. Only if a reliable radius of curvature is known, the model is usable. Since SOFA uses a parameter which has nothing to do with the tip of the needle, it is necessary to create such a table.

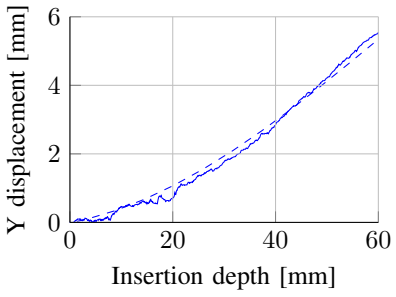

Fig. 13. Comparison between needle insertion experiment and SOFA model, gelatine = -, SOFA = - -

IV. Conclusion

In the previous section, it is shown that the displacement of the prostate of the gelatine phantom, PVC phantom and SOFA model differ at most 0.54 mm with the Abaqus model. The SOFA model gives the worst results, this is caused by the limited amount of calculation time, the different mesh and the proximity detection of SOFA which results in a slightly larger needle guide. For the needle insertion, the results show a maximum difference of 0.4 mm which is most likely caused by different material properties. Finally some recommendations are given.

A. Recommendations and future work

Since the shown solution is not perfect yet, several recommendations and ideas for future work are put together. To start, the experiments could be improved by creating a better phantom: the material properties are not completely known yet. The process of creating a phantom should be better defined so a more accurate phantom can be produced. To do this, the properties of PVC have to be better determined; compression tests to better characterize the material, influence of temperature and humidity, shrinkage of PVC and long term property changes. Also, the frictionless condition can be improved, now a mixture of soap and water is used but there might be fluids which make the support closer to frictionless. Of course, other phantom material is also an option.
For the recording of the displacement, a better camera system could be used, another lens and a higher resolution could make the system more accurate. Together with a subpixel accurate algorithm, the resolution could be several factors higher.

With regard to needle insertion, more experiments are required, because for now, the needle is not rotated. For a practical application it is important that the needle rotation can also be simulated. This means that many experiments have to be done before the prediction can be verified. After this, needle steering towards a target is important. Finally, obstacle avoidance should be applied to avoid veins and other parts of the body that should not be harmed.

For the SOFA model, the deformation model should be faster and more accurate. The differences in plane of the needle insertion are pretty good, but the experiments should be more accurate in order to say what is the truth. Also a better look at the out of plane movement is necessary. The mesh should be optimized, the influence of the geometries behind the prostate should be investigated: removing some of it will speed up the simulation and the friction of the needle could be modelled to match the model even better with the experiments. Finally, a needle guide should be added in the needle insertion model.

Acknowledgment

The author thanks Gerben te Riet o/g Scholten and Alfred de Vries for help with phantom mould manufacturing. Also Philip Muis of Van Son Liquids is thanked for supplying the ink for the markers. Furthermore Igor Peterlik and Christian Duriez are thanked for helping with the needle insertion

modelling.

REFERENCES

[1] K. Engelhard, H. Hollenbach, B. Kiefer, A. Winkel, K. Goeb, and D. Engehausen, "Prostate biopsy in the supine position in a standard 1.5-t scanner under real time mr-imaging control using a mr-compatible endorectal biopsy device," *European radiology*, vol. 16, no. 6, pp. 1237–1243, 2006.

[2] C. P. Evans and J. E. Busby, *Prostate Cancer*. John Wiley & Sons, Ltd, 2001. [Online]. Available: http://dx.doi.org/10.1038/npg.els.0003856

[3] H. Ragde, G. L. Grado, B. Nadir, and A.-A. Elgamal, "Modern prostate brachytherapy," *CA: A Cancer Journal for Clinicians*, vol. 50, no. 6, pp. 380–393, 2000. [Online]. Available: http://dx.doi.org/10.3322/canjclin.50.6.380

[4] H. Ragde, J. Blasko, P. Grimm, G. Kenny, J. Sylvester, D. Hoak, W. Cavanagh, and K. Landin, "Brachytherapy for clinically localized prostate cancer: Results at 7-and 8-year follow-up," in *Seminars in surgical oncology*, vol. 13-6. Wiley Online Library, 1998, pp. 438–443.

[5] V. Lagerburg, "A robotic device for mri-guided prostate brachytherapy," Ph.D. dissertation, Utrecht University, 2008.

[6] R. C. Susil, C. Ménard, A. Krieger, J. A. Coleman, K. Camphausen, P. Choyke, G. Fichtinger, L. L. Whitcomb, C. N. Coleman, and E. Atalar, "Transrectal prostate biopsy and fiducial marker placement in a standard 1.5t magnetic resonance imaging scanner." *J Urol*, vol. 175, no. 1, pp. 113–120, Jan 2006. [Online]. Available: http://dx.doi.org/10.1016/S0022-5347(05)00065-0

[7] D. Beyersdorff, A. Winkel, B. Hamm, S. Lenk, S. Loening, and M. Taupitz, "Mr imaging–guided prostate biopsy with a closed mr unit at 1.5 t: Initial results1," *Radiology*, vol. 234, no. 2, pp. 576–581, 2005.

[8] Y. Yu, T. Podder, Y. Zhang, W. Ng, V. Misic, J. Sherman, L. Fu, D. Fuller, E. Messing, D. Rubens *et al.*, "Robot-assisted prostate brachytherapy," *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006*, pp. 41–49, 2006.

[9] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer, "Elastically deformable models," in *ACM Siggraph Computer Graphics*, vol. 21. ACM, 1987, pp. 205–214.

[10] S. DiMaio and S. Salcudean, "Needle insertion modeling and simulation," *Robotics and Automation, IEEE Transactions on*, vol. 19, no. 5, pp. 864–875, 2003.

[11] S. Zachow, E. Gladiline, H. Hege, and P. Deuflhard, "Finite-element simulation of soft tissue deformation," in *Proc. CARS*. Citeseer, 2000, pp. 23–28.

[12] S. Misra, K. Macura, K. Ramesh, and A. Okamura, "The importance of organ geometry and boundary constraints for planning of medical interventions," *Medical engineering & physics*, vol. 31, no. 2, pp. 195–206, 2009.

[13] S. Misra, K. B. Reed, B. W. Schafer, K. T. Ramesh, and A. M. Okamura, "Mechanics of flexible needles robotically steered through soft tissue." *Int J Rob Res*, vol. 29, no. 13, pp. 1640–1660, Nov 2010. [Online]. Available: http://dx.doi.org/10.1177/0278364910369714

[14] N. Abolhassani, R. Patel, and M. Moallem, "Needle insertion into soft tissue: a survey." *Med Eng Phys*, vol. 29, no. 4, pp. 413–431, May 2007. [Online]. Available: http://dx.doi.org/10.1016/j.medengphy.2006.07.003

[15] J. Allard, S. Cotin, F. Faure, P.-J. Bensoussan, F. Poyer, C. Duriez, H. Delingette, and L. Grisoni, "Sofa an open source framework for medical simulation," in *Medicine Meets Virtual Reality (MMVR'15)*, Long Beach, USA, February 2007.

[16] O. Comas, Z. Taylor, J. Allard, S. Ourselin, S. Cotin, and J. Passenger, "Efficient nonlinear fem for soft tissue modelling and its gpu implementation within the open source framework sofa," *Biomedical Simulation*, pp. 28–39, 2008.

[17] S. Cotin, H. Delingette, and N. Ayache, "Real-time elastic deformations of soft tissues for surgery simulation," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 5, no. 1, pp. 62–73, 1999.

[18] C. Duriez, C. Guébert, M. Marchal, S. Cotin, and L. Grisoni, "Interactive simulation of flexible needle insertions based on constraint models," *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2009*, pp. 291–299, 2009.

[19] D. Cheng and C. M. Tempany, "Mr imaging of the prostate and bladder." *Semin Ultrasound CT MR*, vol. 19, no. 1, pp. 67–89, Feb 1998.

[20] J. E. Husband, A. R. Padhani, A. D. MacVicar, and P. Revell, "Magnetic resonance imaging of prostate cancer: comparison of image quality using endorectal and pelvic phased array coils." *Clin Radiol*, vol. 53, no. 9, pp. 673–681, Sep 1998.

[21] T. A. Krouskop, T. M. Wheeler, F. Kallel, B. S. Garra, and T. Hall, "Elastic moduli of breast and prostate tissues under compression." *Ultrason Imaging*, vol. 20, no. 4, pp. 260–274, Oct 1998.

[22] H. Yamada and F. Evans, *Strength of biological materials*. Williams & Wilkins Baltimore:, 1970, vol. 1000.

[23] A. E. Kerdok, S. M. Cotin, M. P. Ottensmeyer, A. M. Galea, R. D. Howe, and S. L. Dawson, "Truth cube: establishing physical standards for soft tissue simulation." *Med Image Anal*, vol. 7, no. 3, pp. 283–291, Sep 2003.

[24] M. Oldfield, D. Dini, G. Giordano, and F. Rodriguez y Baena, "Detailed finite element modelling of deep needle insertions into a soft tissue phantom using a cohesive approach," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 0, no. 0, pp. 1–14, 2011, pMID: 22229447. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/10255842.2011.628448

[25] M. Oldfield, D. Dini, and F. Rodriguez y Baena, "Predicting failure in soft tissue phantoms via modeling of non-predetermined tear progression," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, 28 2012-sept. 1 2012, pp. 6305 –6308.

[26] W. Assaad and S. Misra, "Combining ultrasound-based elasticity estimation and fe models to predict 3d target displacement." *Med Eng Phys*, Dec 2012. [Online]. Available: http://dx.doi.org/10.1016/j.medengphy.2012.11.003

[27] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[28] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, ser. Software that sees. O'Reilly Media, Incorporated, 2008. [Online]. Available: http://books.google.nl/books?id=seAgiOfu2EIC

[29] R. Roesthuis, Y. van Veen, A. Jahya, and S. Misra, "Mechanics of needle-tissue interaction," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 2557–2563.

[30] G. Vrooijink, "Real-time three-dimensional tracking and steering of flexible needles using two-dimensional ultrasound images," master's degree in electrical engineering - measurement and control engineering, University of Twente, 2012.

[31] R. Roesthuis, M. Abayazid, and S. Misra, "Mechanics-based model for predicting in-plane needle deflection with multiple bends," in *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on*. IEEE, 2012, pp. 69–74.

[32] K. Nightingale, L. Zhai, J. Dahl, K. Frinkley, and M. Palmeri, "4k-5 shear wave velocity estimation using acoustic radiation force impulsive excitation in liver in vivo," in *Ultrasonics Symposium, 2006. IEEE*, oct. 2006, pp. 1156 –1160.

[33] INRIA. Simulation open framework architecture. [Online]. Available: http://www.sofa-framework.org/

[34] H.-W. Nienhuys and A. F. van der Stappen, "Interactive needle insertions in 3d nonlinear material," 2003.

[35] ——, "A computational technique for interactive needle insertions in 3d nonlinear material," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation, ICRA 2004, April 26 - May 1, 2004, New Orleans, LA, USA*. IEEE, 2004, pp. 2061–2067.

[36] P. R. Johnson, P. N, and S. E., "Element-splitting for simulation of fracture in 3d solid continua," 2005.

[37] I. Peterlik, M. Nouicer, C. Duriez, S. Cotin, and A. Kheddar, "Constraint-based haptic rendering of multirate compliant mechanisms," *Haptics, IEEE Transactions on*, vol. 4, no. 3, pp. 175 –187, may-june 2011.

[38] J. Bouguet, "Camera calibration toolbox for matlab," 2004.

A

Model preparation

A.1 Introductory

This appendix shows how to prepare the phantom including the prostate and the surrounding organs/tissues.

A.2 Overview of mould

Figure A.1 shows the mould and the part names. This view is as if the model is still in the mould. In the following section the time line will refer to this figure.

A.3 Time line

The time line is shown in Table A.1 and Table A.2. This manual is used for preparing the gelatine phantom. Since there is less experience with PVC, the time line is not as defined as the gelatine. Since the water evaporates during the heating process, it is important to notice the times. The time of heating PVC does not affect the properties as much. This was tested using small and large amounts of PVC: a small amount heats much faster and therefore there is not much time to evaporate. The large amount of PVC needed to heat up for an hour during it could evaporate a lot, the properties of the PVC were the same using the ARFI technique. The most important part of creating a PVC phantom is to heat the PVC up such that it is thin and it does not solidify while pouring it.

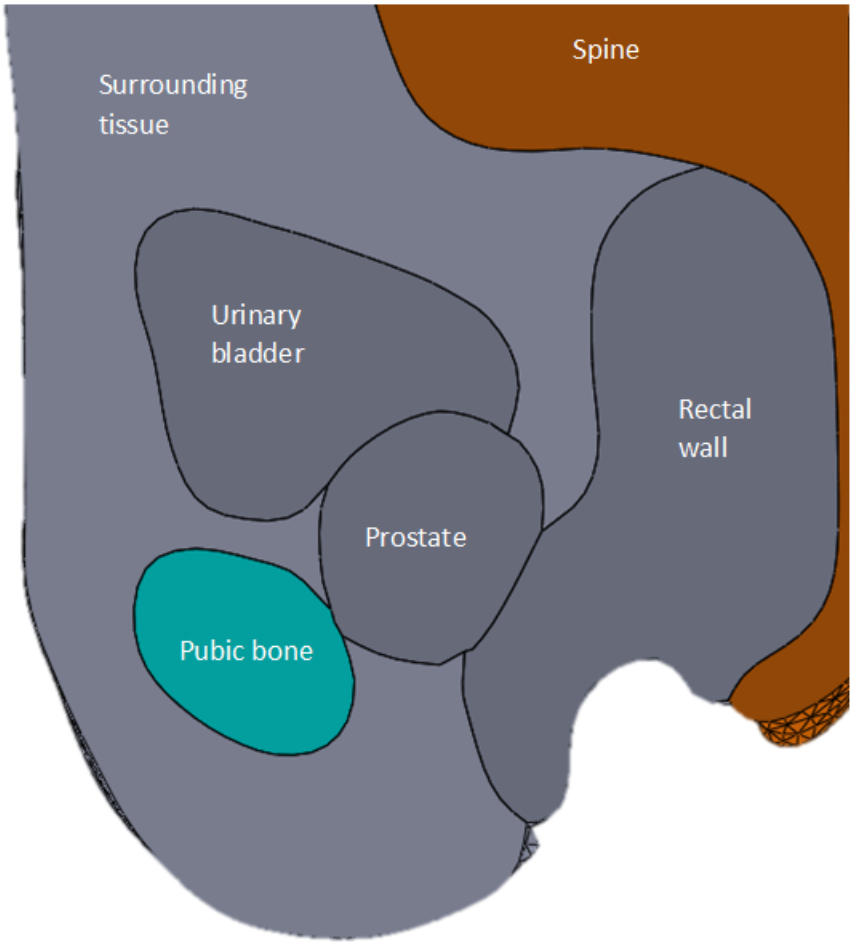

Figure A.1: Overview of the model

Start time		Addit. time	Description
0:00	+	0:05	Put mould in freezer and start cleaning a beaker
0:05	+	0:04	The water (0.45 L) should be around 41° C, the set point for the heating plate is 70° C and the bottom plate is set to 175° C (195° C for plastic beaker)
0:09	+	0:17	Now the water should be around 47° C and the gelatine (25,7%) should be added
0:26	+	0:04	Once the gelatine is 65° C the silica can be added (1% of the total mass in the beaker)
0:30	+	0:13	Change the set point to 75° C
0:43	+	0:38	The gelatine is 70° C, now it should be poured in the mould (rectal wall), it should be left on the desk as this is level
1:21	+	0:05	Now the mould can be put in the refrigerator
1:26	+	0:20	The mould can be put in the freezer
1:46	+	0:04	Start heating up water (0.4 L) in a beaker: SP 65° C, 175° C (195° C for plastic beaker)
1:50	+	0:07	When the water is 40° C add the gelatine (19%)
1:57	+	0:03	Add 1% (of total mass in beaker) silica @ 54° C
2:00	+	0:14	Put a foil on top of the beaker
2:14	+	0:13	The gelatine is 63° C, put the gelatine in the mould (urinary bladder)
2:27	+	0:07	Put the mould in the refrigerator
2:34	+	0:19	Put the mould in the freezer

Table A.1: First part of preparing the gelatine phantom

Start time		Addit. time	Description
2:53	+	0:01	Start heating up water (0.4 L) in beaker: SP 65° C, 175° C (195° C for plastic beaker) and add gelatine when the water is 40° C (17%)
2:54	+	0:10	Put foil on top of it
3:04	+	0:03	Add 1% (of total mass in beaker) silica @ 60° C
3:07	+	0:04	Put foil back on
3:11	+	0:02	Get mould out of freezer and remove prostate parts
3:13	+	0:03	Put the mould back in the freezer and set the set point to 0° C
3:16	+	0:02	Get mould out of freezer
3:18	+	0:12	Put the gelatine in the mould (prostate), also fill an Erlenmeyer flask with about 50 mL until 100 mL to make the markers
3:30	+	0:09	Put foil on top of mould and put in freezer
3:39	+	0:41	Foil on top of box of the 25% gelatine and 19% gelatine
4:20	+	0:38	Foil on top of box and flask of the 17% gelatine and put in refrigerator
4:58	+	0:15	Removed parts from mould an put back in refrigerator
5:13	+	0:15	Start heating up water (0.6 L) in beaker: SP 50° C, 175° C (195° C for plastic beaker) (5.3%)
5:28	+	0:01	Get mould from refrigerator
5:29	+	0:19	Add 1% (of total mass in beaker) silica @ 48° C
5:48	+	0:07	Pour a thin layer in the mould to prevent leaking
5:55	+	0:12	Put mould back in refrigerator covered with foil. SP 0° C
6:07	+	0:10	Take mould out refrigerator and heat gelatine up to 50° C using SP 55° C
6:17	+	0:40	Pour the gelatine in mould (>50° C) (surrounding tissue)
6:57	+	1:03	Put mould in refrigerator covered with foil.
8:00			Remove mould from refrigerator

Table A.2: Second part of preparing the gelatine phantom

A.4 Removing (parts of the) mould

Removing the mould is a very precise job. One has to be careful not to break the gelatine. Some of the inner parts are a little difficult to remove especially the ones between the prostate and rectal wall. These parts should be removed with care, it helps to lift the part which is not under the gelatine up.
After the solidifying of the surrounding tissue the mould should be put upside down on the Perspex plate. Now it is possible to press the pubic bone and spine to the Perspex plate while using a knife to slowly separate the gelatine from the mould.

A.5 Adding markers

The markers can be made using the gelatine in the Erlenmeyer flask. First heat the gelatine up to 70° C (SP 80° C, 175° C), once this is done, add the green ink. 1% is enough, one can use a syringe or a pipette. Please clean it immediately because this ink is meant to colour plastics and if it is in contact with the syringe, pipette or sink too long it will stay forever green. Once it is mixed, take the Erlenmeyer flask and take it to the mould. Take a pipette (Rainin, Pipet-Lite, available at ECTM of the University of Twente) with a maximum of 100 μL and set the maximum amount to about 40 μL. This is enough to keep the gelatine in the pipette hot long enough to make several markers. If this is set too low, the gelatine will almost instant solidify. Please be aware that the gelatine cools down pretty fast, so if the gelatine almost solidifies in the pipette it is time to heat it up again. The pattern of the markers is shown in Figure A.2.

A.6 Stiffness verification

The stiffness of the gelatine can be verified by using the Acuson S2000 ultrasound machine from Siemens. Using virtual touch, the shear wave velocity can be determined. This is typically a number between 1 and 9. From this velocity, the Young's Modulus can be calculated using the following formula:

$$E = 2(1 + \mu) \cdot v_s^2 \cdot \rho \tag{A.1}$$

Where E is the Young's Modulus (Pa), μ is Poisson's ratio, v_s the shear wave velocity and ρ is the density of the material. Typically these values can be used:

$$E = 2 \cdot 1.495 \cdot 1000 \cdot v_s \cdot v_s = 2990 \cdot v_s \cdot v_s[Pa] \tag{A.2}$$

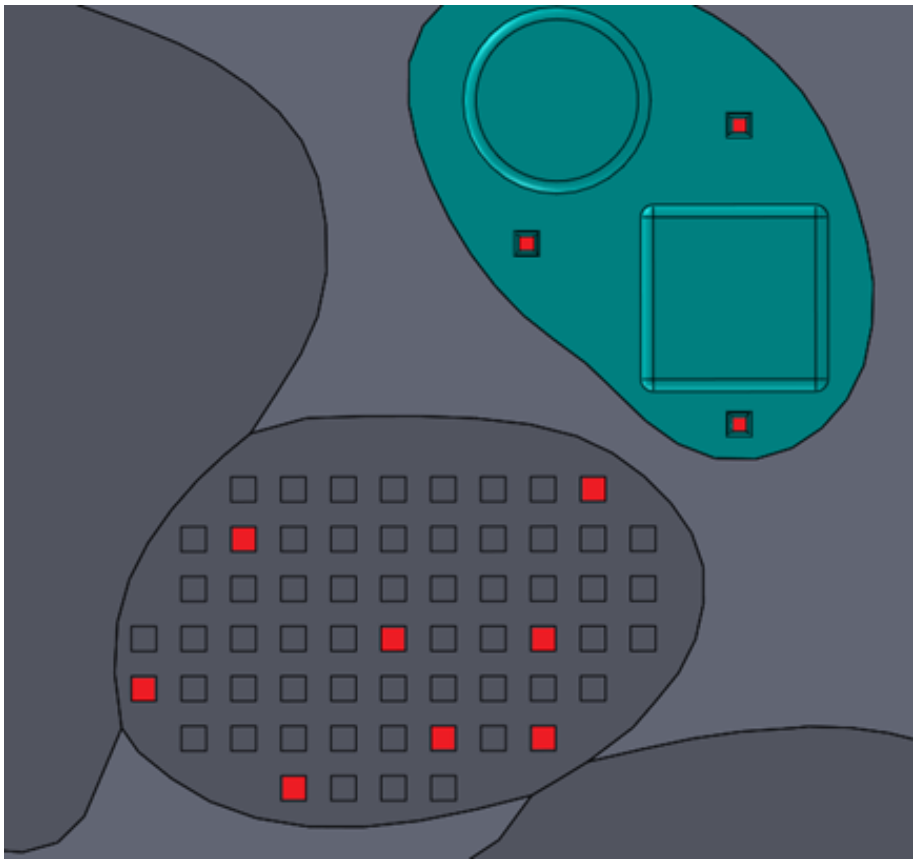

Figure A.2: Marker positions

Geometry	E[kPa]
Rectal wall	172.0
Urinary bladder	98.0
Prostate	75.0
Surrounding tissue	10.0

Table A.3: Used Young's modulus for materials

$$E = 2.99 \cdot v_s \cdot v_s[kPa] \tag{A.3}$$

Table A.3 shows an overview of the desired Young's modulus per geometry.

A.7 Overview of material composition

This section gives an overview of the materials inside the different geometries. Table A.4 shows the percentages of the gelatine for each part and Table A.5 shows the same table for PVC.

Type	Total volume [L]	Mass % gelatine	Mass % water
Rectal wall	0.45	25.7	74.3
Urinary bladder	0.4	19	81.0
Prostate	0.4	17	83.0
Surrounding Tissue	0.6	5.3	94.7

Table A.4: Gelatine percentages

Type	Total volume [L]	Softener [%]	Standard [%]	Rigide [%]
Rectal wall	0.45	0	10	90
Urinary bladder	0.3	0	90	10
Prostate	0.3	5	95	0
Surrounding Tissue	0.5	50	50	0

Table A.5: PVC percentages

B

Marker tracking

The program to track the markers on top of the model is described in this appendix. It is written in C++ and uses the OpenCV library for computer vision. First the pseudo-code is given.

B.1 Pseudo-code

A brief overview of the C++ code will be given to make it easier to understand the code. The purpose of this program is to move the needle guide to the model and record the motion of the markers. So, there are three parts which should be put together: needle guide movement, data acquisition and data output.

B.1.1 Movement

The linear stage which moves the needle guide is connected to a CAN bus. Some files have been written by Guus Vrooijink to control the linear stage which is connected via a CAN bus. Parts of this code are used to move the needle guide towards the prostate.

B.1.2 Data acquisition

Data acquisition is the most difficult part of this program. The first step is to get a colour image and convert it to the HSV colour space. The HSV space is preferred because the image is less influenced by the amount of light. After some experiments, the green colour of the markers is determined and used in a threshold function. After the image is cleared from all the non-green objects, a search for contours is started. This OpenCV function lists all the contours. The contours are filtered by size (minimum and maximum)

to make sure only the markers are found.

Next, the centre of mass of all the markers has to be found. After these so called image moments are determined, they are listed and the next frame is captured.

The new frame is treated the same way but now the markers should be matched to the previous set. This is done using several if-statements and for-loops. The result of this methods is verified by plotting the marker numbers at a window and see if the markers have the same number even if the block at which the markers are positioned is rotated. There are multiple optional windows, one with the markers and numbering and several windows to fine tune the threshold process. All these windows can be disabled if desired.

B.1.3 Data output

The data of the program is a Comma Separated Value (CSV) file which includes the timestamp, the needle guide position and the marker positions for each measurement. A part of such a measurement is shown below:

50046.1; 2.00023; 743.197; 434.925; 822.973; 418.256;

The first number (50046.1) is the time, 2.00023 is the position of the needle guide in mm. Finally, the x and y of the first two markers are shown. The first marker has coordinates (743.197; 434.925) and the second (822.973; 418.256).

B.1.4 Notes

Image rectification is not done in C++, this is done in Matlab. This is explained in the next Appendix.

C

Matlab scripts

In this appendix, some larger Matlab scripts - besides the scripts for needle bending and force analysis - are explained to obtain the results found in this document.

C.1 Compare experimental, Abaqus and SOFA results

'GetDataFromMonitorsProstate.m' compares the results of the experiments, Abaqus and SOFA. This script reads several types of files to be able to compare it. First, it reads the data from the experiment and recalculates the actual positions of the markers by using the result of a checkerboard calibration using the 'Camera Calibration Toolbox for Matlab'.

The displacement of the markers in the Abaqus model are given in another text file. This file contains all the x- and y-displacements for each marker. This is put in a matrix. Finally, the displacement of the SOFA model is read and put in a matrix.

Once the results are put in a matrix they can be compared: show the results in graphs and run statistical analyses.

C.2 Create scene file and *.msh files

SOFA requires an xml file to describe the scene (boundary conditions, materials, movement etc.) and *.msh files for geometry (mesh) defition, see Appendix E. The creation of these files has been automated. A Matlab script and two Matlab functions have been written in order to automatically create a scene file. This will briefly be described in this sub paragraph.

The script should create a scene file with several variables:

- Mesh size
- Insertion time
- Insertion angle

At first the text files of the geometries - created in ANSYS - are imported and the common nodes are found. A problem is that SOFA is not able to connect 3 nodes at the same point. So this situation must be dealt with by deleting those double attachments (tens of nodes). After all these nodes are matched, the list of fixed node numbers is used to make a list of the fixed node index number. Another list which is created, is the marker list. The node numbers of the markers is known but the node index number is unknown. The script searches inside the prostate node numbers the numbers of the markers and stores the index number. Finally, the 'CreateSCNFileFunction' and the 'EditTxtFilesToMSHFunction' are called to create the actual scene file and the corresponding mesh files. 'CreateSCNFileFunction' uses 13 parameters to built the actual custom scene file and 'EditTxtFilesToMSHFunction' creates meshes from the text files provided by ANSYS.

D

SOFA Installation manuals

D.1 SOFA CUDA installation

The following manual has been tested on a Windows 7 64 bit computer with an Nvidia GTX 560 Ti GPU.

1. Install Visual Studio 2008 (32 bit) from this website http://download.microsoft.com/download/A/5/4/A54BADB6-9C3F-478D-8657-93B3FC9FE62D/vcsetup.exe

2. Install latest Nvidia driver (64 bit) http://www.nvidia.co.uk/Download/Scan.aspx?lang=en-uk

3. Install CUDA (64bit) http://www.nvidia.com/content/cuda/cuda-downloads.html

4. Go to `C:\ProgramData\NVIDIA Corporation\NVIDIA GPU Computing SDK 4.2\C\common` and open `cutil_vs2010.sln`. Don't click on the green play button but press right mouse button on the `cutil` folder (left in the solution explorer) en click rebuild for both debug and release. This should give no errors.

5. Now you can test `deviceQuery_vs2010.sln` from `C:\ProgramData\NVIDIA Corporation\NVIDIA GPU Computing SDK 4.2\C\src\deviceQuery` This doesn't do much but it shouldn't give an error. Now CUDA is installed successfully.

6. Download all the required files from this website http://www.sofa-framework.org/download In total 5 files need to be downloaded: framework, modules, applications, patch, dependency package of Visual Studio 2008 @ 32 bit.

7. Unzip all the files to `C:\SOFAGPU` in this order: framework, modules, applications, patch and dependency package. The patch should replace several files.

8. The folder structure looks like this:

```
sofa
    applications
    bin
    examples
    extlibs
    features
    framework
    include
    lib
    licences
    modules
    scripts
    share
    tests
    tools
```

9. Make a copy of `sofa-default.prf` and rename it to `sofa-local.prf`

10. Change `sofa-local.prf` to make a solution you want. First, to know what to change, one could use this website: http://developer.nvidia.com/cuda/cuda-gpus to find the compute capability of the GPU.
So in the case of a GTX 560 the version is 2.1 →21. The total change in the GPU section is:

```
# Uncomment if you want to compile CUDA GPU...
DEFINES += SOFA_GPU_CUDA
# Compute capabilities (sm_10 for G80, sm_11 for G92,
sm_13...
CUDA_FLAGS += --ptxas-options=-v -arch sm_21
```

    In total, two lines are uncommented. Save the file.

11. Start `Project VC9.bat`, this will generate a solution for Visual Studio 2008 @ 32 bit using the settings of the `sofa-local.prf` file.

12. If necessary, rename the solution to SofaGPU (handy if you wish to also make a non GPU version to compare the performance).

13. Start Visual Studio 2008 Express and go to Tools - Options - Projects and Solution - Build and Run - Set maximum number of parallel project builds to 1.

14. Edit the properties of `sofagpucuda_1_0` (right mouse button - Properties): Configuration Properties - Linker - Input - add 'libcmt' to the 'Ignore Specific Library'.

15. Open the solution and build both release and debug versions (right click upper file in the solution explorer called 'Solution 'SofaGPU'(70 projec...)'.

16. Once it is built with no errors, in the `C:\SOFAGPU\bin` folder there exist several executables: `runSofa.exe` and `Modeler.exe`. Try these to test whether the build was successful. If the name is `*.*d.exe` it means it is the debug version (slower).

D.1.1 Notes

- I didn't get it running in Visual Studio 2010, therefore Visual Studio 2008 was used.

- The installation paths are of course free to choose, it is however wise to make the path as short as possible and don't use any spaces.

D.2 SOFA Boost Installation

The following manual has been tested on a Windows 7 64 bit computer with an Nvidia GTX 560 Ti GPU.

1. Install BOOST 1.46.1 from this website: http://www.boostpro.com/download/ and use the settings shown in Figure D.1.

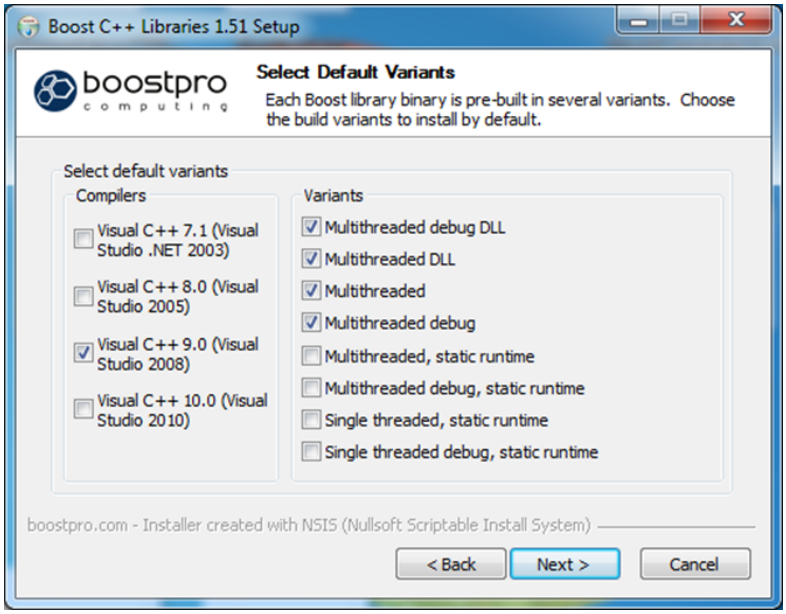

Figure D.1: Boost settings

2. Install Visual Studio 2008 (32 bit) from this website `http://download.microsoft.com/download/A/5/4/A54BADB6-9C3F-478D-8657-93B3FC9FE62D/vcsetup.exe`

3. ownload all the required files from this website `http://www.sofa-framework.org/download`.
   In total 5 files need to be downloaded: framework, modules, applications, patch, dependency package of Visual Studio 2008 @ 32 bit.

4. Unzip all the files to `C:\SOFAGPU` in this order: framework, modules, applications, patch and dependency package. The patch should replace several files.

5. Make a copy of `sofa-default.prf` and rename it to sofa-local.prf.

6. Edit the preference file and uncomment the following lines:

   # Uncomment if you want to use Boost lib for multithread computing
   DEFINES += SOFA_HAVE_BOOST

   # On Windows, the path where boost is install and the suffix of the dlls should be specified
   BOOST_ROOT = 'C:\Program Files (x86)\boost\boost_1_50'
   BOOST_SUFFIX = -vc90-mt-1_50

```
# Activate multithreading support (requires SOFA_HAVE_BOOST)
DEFINES += SOFA_MAX_THREADS=3
```

Just make sure the version in red is correct and the number of threads should be around the number of cores + 1 (not really tested).

7. Start `Project VC9.bat`, this will generate a solution for Visual Studio 2008 @ 32 bit using the settings of the `sofa-local.prf` file.
8. If necessary, rename the solution to SofaBOOST (handy if you wish to also make a non-BOOST version to compare the performance).
9. Start Visual Studio 2008 Express and go to Tools - Options - Projects and Solution - Build and Run - Set maximum number of parallel project builds to 1.
10. Open the solution and build both release and debug versions (right click upper file in the solution explorer called 'Solution 'SofaGPU'(70 projec...)'.
11. Once it is built with no errors, in the `SOFABOOST\bin` folder there exist several executables: `runSofa.exe` and `Modeler.exe`. Try these to test whether the build was successful. If the name is `*.*d.exe` it means it is the debug version (slower).

Finally, sometimes the `boost_thread-vc90-mt-1_46_1.dll` file should be added to the `\bin` directory because `sofatest.exe` gives error that it could not find this `*.dll`.

E

SOFA Model

E.1 Introduction

SOFA is an Open Source framework primarily targeted at real-time simulation, with an emphasis on medical simulation. Since creating a model in SOFA is completely different than most commercial FE packages, the SOFA model will be discussed in more detail.

E.2 Mesh

SOFA is not capable of creating its own mesh and therefore another program is used. The mesh is created using ANSYS. An APDL (ANSYS Parametric Design Language) script writes all the nodes and elements to a file in a format which can be read by Matlab to create a file format which can be loaded by SOFA (see Appendix C for more information). A small piece of such a mesh file is shown below.

```
1   $NOD
2   444
3   1 52.808 135.984 98.292sd
4   2 58.465 130.327 90.292
5   ...
6   444 53.596 128.5 99.368
7   $ENDNOD
8   $ELM
9   1731
10  1 4 1 1 4 117 73 118 81
11  2 4 1 1 4 117 119 120 81
12  ...
13  1731 4 1 1 4 330 376 272 377
14  $ENDELM
```

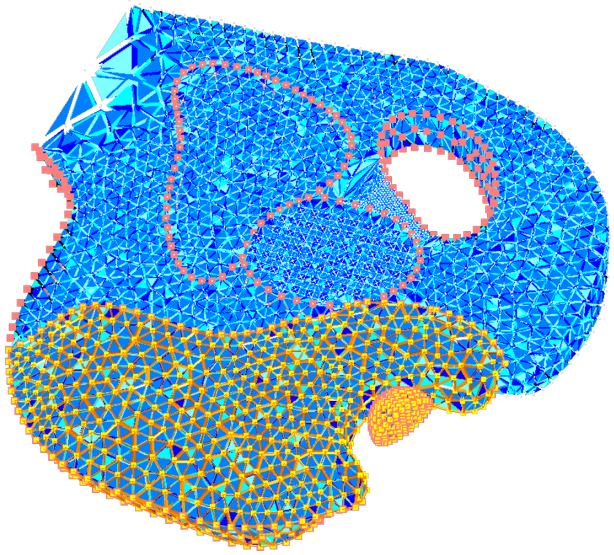

Figure E.1: Mesh in SOFA

The first part of the mesh file contains the node number and position of the node; the first node is located at ('52.808, 135.984, 98.292'; x, y, z). The second part creates elements by defining the type ('4 1 1 4'; a 4-node tetrahedron) and which nodes are involved per element ('117 73 118 81' for the first element). This file type is used to import the geometries in SOFA. Table E.1 shows the number of nodes and elements for each geometry. Figure E.1 shows the mesh when all the parts are imported in SOFA.

Geometry	Number of nodes	Number of elements
Bladder	1245	5472
Prostate	5185	28813
Surrounding	7410	32775
Rectal	1865	8135
Needle guide	444	1731
Total	16149	76926

Table E.1: Number of nodes and elements per geometry

E.3 Model

As mentioned before, a model in SOFA is created in a different way than most commercially available FE packages. This section describes how the boundary conditions, materials, movements etc. are defined.

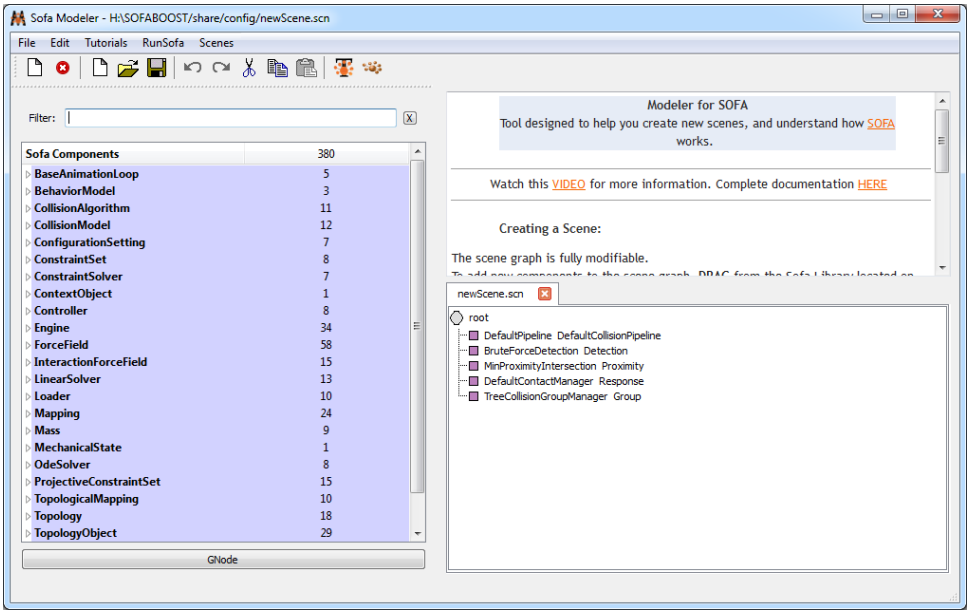

Figure E.2: To create a SOFA model, 'Modeler' can be used

E.3.1 Model structure

Where an Abaqus model can be created using a GUI, SOFA uses files which are written in XML format. To create these files in an easy and simple way, a so called 'Modeler' is available (Figure E.2). Using Modeler, the user can browse the available constraints, solvers, mesh loaders etc. For many of these functions there is an example available to learn how it should be used. A model consists of several sub groups (Figure E.3). This nested structure makes that a 'lower' group uses the properties of its parent group. Using this structure, it becomes clear that the 'Bladder' group can use the attach constraint of its parent. By choosing the right tree structure, dependencies between groups can be defined. Now the structure of the entire model is clear, the implementation of constraints will be discussed.

E.3.2 Constraints

The model requires several fixed constraints which are implemented as shown in Figure E.4. Such a constraint is implemented inside a geometry, as shown in Figure E.5. This figure shows that an object has several properties such as fixed constraint. It is possible to constrain nodes by specifying a list of

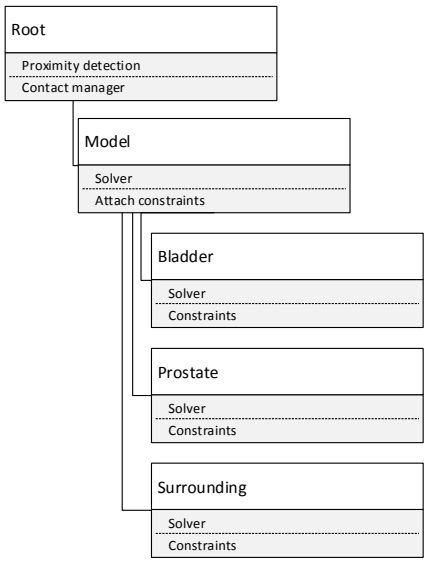

Figure E.3: Nested structure in SOFA

node numbers.

At the bottom of the model is a frictionless constraint applied. This is modelled in SOFA by applying gravity and putting a plane force field underneath the geometries. Matlab is used to find the nodes which have this constraint.

E.3.3 Collision

The model should also deal with collision: this is where the needle guide collides with the rectal wall. Due to this interaction, the whole model starts to move. The collision detection happens at different levels: triangles, lines and points. In SOFA it is possible to set a so called 'contact distance'. This is the distance below which a contact is created. The contact distance is set to 0.2 mm as otherwise the points where not colliding and there was no interaction at all between the needle guide and rectal wall.

E.3.4 Attach constraints

There is one way to connect geometries in SOFA: by using an 'AttachConstraint'. This method should be put in a parent group of the geometries which should be attached, see Figure E.3. The attach method should be

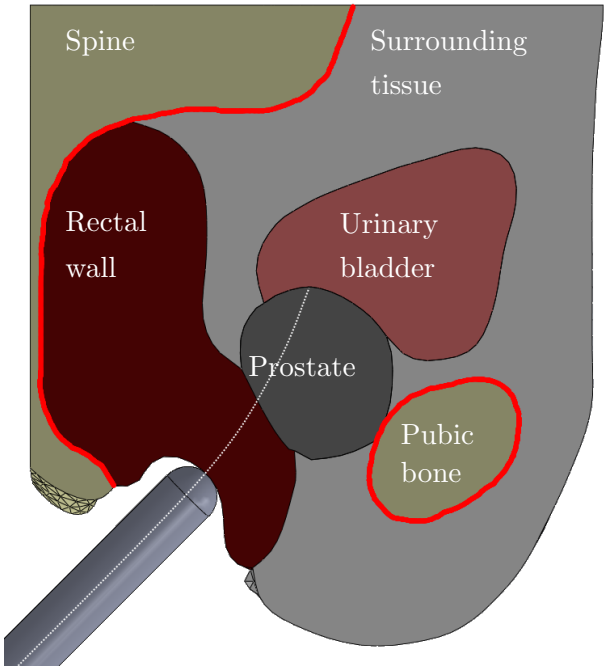

Figure E.4: Fixed boundary conditions of the model are shown in red

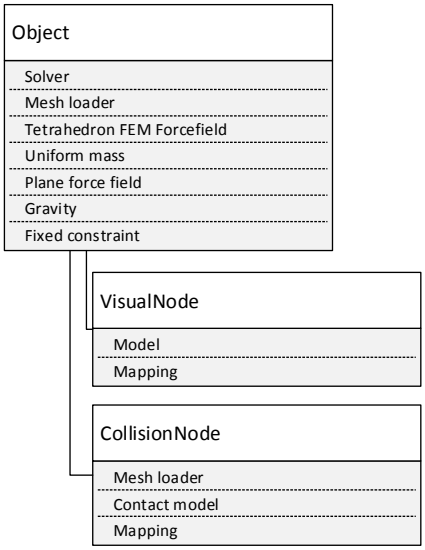

Figure E.5: Object structure in SOFA

able to automatically find the nearest nodes to apply an attach constraint but this mechanism was not working. Another way of using this method is by doing it manually, node-by-node. In the AttachConstraint method, two lists of nodes are required. These lists contain the nodes of the geometries which should be attached. The order of this list is important as the nodes are connected based on their index of the list. If the nodes are not at the same position at the start of the simulation, they will be pulled to each other. A node mismatch is detected if there is movement without any input.

E.3.5 Material properties

The geometries have different properties in SOFA. The most important properties for this research are the Young's Modulus (E) [kPa] and Poisson's ratio. The used material properties are shown in Table E.2. These two properties are given in the object group of Figure E.5. Here is a method called 'Tetrahedron FEM Forcefield' which defines the mechanical properties of the geometry. Without this method, the elements would fall apart because there is no relation between them.

Geometry	Young's Modulus [kPa]	Poisson's ratio
Bladder	98.0	0.495
Prostate	75.0	0.495
Surrounding	10.0	0.495
Rectal	172.0	0.495
Needle Guide	10 000	0.3

Table E.2: Material properties used for FE model

E.4 Optimization

As SOFA is open source, it is possible to read and change the source code. To be able to use SOFA with different hardware, the developers provide a preference file which can be edited. This file is basically a large list of flags which can be enabled to use for instance CUDA or Boost support. CUDA is a parallel programming framework by NVIDIA. Developers can use this framework to use the computational power of the GPU. Boost is a set of C++ libraries which should be able to speed up a program, these libraries are optimized and reviewed by many programmers.

It was not possible to create the complex model in SOFA using the CUDA framework since there are several functions not available for CUDA. One of them is the possibility to attach nodes. Since the entire model uses

the fact that several geometries are bonded, it is important that this method is available.
To address this problem, it was tried to use the localStiffnessFactor option in 'Tetrahedron FEM Forcefield'. This option makes it possible to give each element a different stiffness based on its element number. A Matlab script has been written to list the stiffness per element to use the local stiffness factor option. For some reasons the model was not working with the GPU, one of the problems was the contact part. For future work, it might be worth taking a better look at the local stiffness factor option to speed up the model.

For the needle insertion, the use of GPU is less useful: most of the computational time is solving the linear system which is computed by a direct solver. The direct solver is needed for the needle insertion plug-in to handle constraint corrections. An iterative solver can make use of the CUDA cores but is not useful because the needle cannot be positioned correctly. There is not a direct solver implemented in CUDA yet.

SOFA has been compiled using the Boost libraries but there was no noticeable speed up. Since this is a flag in the preferences file which enables or disables the use of Boost libraries, nothing more could have been done.

The compilation of SOFA is explained in Appendix D.

F

Used materials

This appendix gives an overview of the tested materials to use for a phantom. The material of the phantom should originally have the following properties:

- Not sticky
- Durable, same properties even after weeks/months
- Linear elasticity
- Variable elasticity of 10 kPa to 200 kPa
- Not dangerous for people
- Solidifies at room temperature

F.1 Gelatine

Gelatine was available as this research started. Gelatine satisfies most of the desired properties:

- Not sticky
- Variable elasticity of 10 kPa to 200 kPa
- Not dangerous for people
- Solidifies at room temperature

But it has one major drawback: it is not durable. This means that a model of the phantom can only be used several hours. After that, the gelatine shrinks and gets dehydrated. So, another materials had to be found.

F.2 Wacker SilGel

To find other materials, several companies have been contacted and they came up with Wacker SilGel 612 A/B. This is a two component silicone rubber which vulcanizes at room temperature. The stiffness of this rubber can be varied by varying the mixing ratio between component A and B.

The main problem of this material is that it gets stickier if the stiffness gets lower. SilGel consists of two components and B is the softener. So, to get a higher stiffness, the A component should be higher. For high stiffnesses, the material doesn't stick as much as it does at a 1:1 ratio but it is not perfect and since the lower ratios are also needed, an additive has been tried to find.

After some contact with BYK-Cera (company for additives), a sample of 'ceraflour 1000' was sent. This should reduce the stickiness. After some testing, it was concluded that it made it only worse. This was reported to the company and they suggested another 'Hordamer PE 02' but this required more caution since this additive is much more dangerous. Since there was not enough lab experience it was chosen not to use this.

Together with the group Inorganic Materials Science of the University of Twente, it was tried to add bees was to the mould in order to prevent the Wacker to stick to it. This has also failed and it was decided not to continue with SilGel.

F.3 PVC

In papers, people used material from M-F manufacturing. This material is used for fishing lures and the stiffness can be easily adjusted. Since the shipping was very expensive - M-F manufacturing is an American company - a European company was found. This company is located in France and is called bricoleurre. This company sells the same components and it was decided to buy it from them. After some tests, the material seems to be very good. There are, however, several drawbacks to this material:

- The properties are not specified and must be determined by hand
- It shrinks
- It has a high melt temperature

The Young's modulus of the material is determined using a compression test, after that, the material properties are verified by using the ARFI

method (ultrasound). Many other properties, density, Poisson's ratio, toxicity, etc. are unknown since it is a material for customers and not an industrial material.
Since the material shrinks, it was tried to minimize this. The shrinking process can be reduced by putting soft PVC in the refrigerator.
The melting temperature depends on the material properties: a high Young's modulus has a higher melting point. For the rectal wall with the highest Young's modulus, this temperature is above 130° C which is higher than room temperature. At first, it was assumed that the 3D printed material could not handle this temperature. After some tests, it was shown that the material does not melt but only gets soft. To add an extra heat shield and make it easier to remove the PVC, a thin layer of nail polish was added (Hema, Longlasting, Miss Helen). Once the phantom was completely poured, it was not bonding very well to the bone parts. To fix these parts, some super glue was added.

G

Adding markers

This appendix describes methods to add markers. Many possibilities have been tried on gelatine, one successful. All these experiments were done on gelatine.

G.1 Spray paint

First, it was tried to use a piece of paper with square holes which were cut by a laser. This way, the distance between the holes was equal. The result was that the ink got everywhere: gelatine does not absorb the paint. The paint just flows under the paper and colours the entire gelatine block. Since this might be cause by the flexibility of the paper, it was also tried with a Perspex plate with laser cut holes. This gave the same result and spray paint was therefore not usable.

G.2 Paint

Instead of using spray paint, it was tried using a paintbrush. First with the paper as described above and later with the Perspex plate, this also gave bad results. Another version was to use a stamp, a simple stamp has been created and the same Perspex plate was used for the marker positions. The result was more controllable, but still not good enough: the ink went everywhere.

G.3 Laser cutting

Since a laser heats material up, it was tried to colour the gelatine by heating it up. The only result it gave were small wet spots with no visible result.

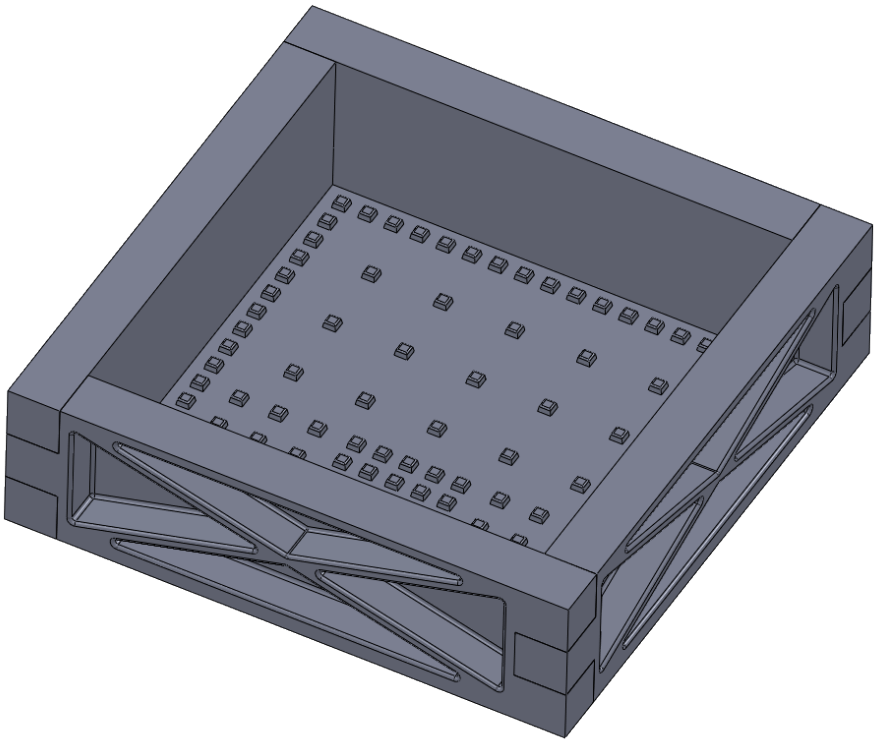

Figure G.1: Box with small bumps on the bottom

G.4 Holes

The laser cutting method was the first method which actually changed the gelatine. Instead of adding ink on top of the gel, it was tried to put it inside the gel using small holes. Since the laser cutter was not successful in creating holes, a mould had to be used. A small box has been printed and the bottom consisted of small bumps, see Figure G.1. This results in a gelatine block with small holes which can be filled with a coloured material. At first, it was tried with ink from Van Son Liquids (FF-Aquabase 51530) but this ink was too hard and popped out of the gelatine holes.

The next idea was to fill the holes with coloured gelatine. Three methods have been tried to colour the gelatine: pigment, food colouring and the Van Son ink. The pigment had too many small granules and the colour of the food colouring was not good enough. The Van Son ink was perfect, a small amount of this ink was enough to colour the markers green (<1% volume).

H

Running experiments

This appendix describes how to do the experiments. First the prostate movement experiments are described, and later the needle insertion.

H.1 Prostate movement

Assuming the phantom has been created using the manual in Appendix A, the phantom should be put on top of the Perspex plate with holes for the spine and pubic bone, Figure H.1. Once the phantom is positioned at this plate, this plate is put on top of a (white) Perspex plate (thickness of 3 mm) to make sure the bottom of the phantom is at the exact same height as the bottom of the needle guide.

Plug the camera in and mount it above the phantom. Once this is done, start Matlab and run the image acquisition toolbox (imaqtool). This toolbox gives you several options to use the camera, use the Y8_1024x768 settings and set gray scale to Bayer. This means the camera is using a Bayer pattern to record colours. The settings should be as follows:

Brightness 256
FR 30
FT 5000
Gain 384
Gain mode Manual
Hue 37
Opticial Filter 0
Shutter 45
Shutter mode: manual
WB [128 128]

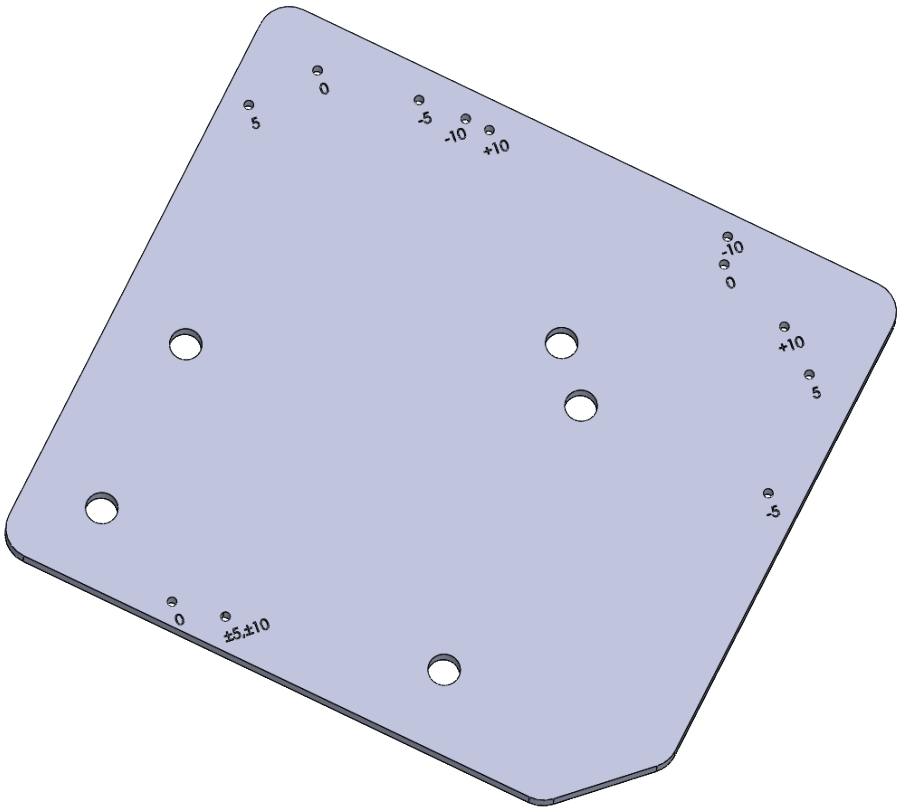

Figure H.1: Perspex plate with holes to fix the bone parts

Where the most important parameters are marked bold. Now, the green markers should be very clear. Most settings can be tuned by hand if necessary but especially the 'Gain' should be put at 384 and the shutter mode should be absolute/'manual'. Using a higher gain results in a brighter but noisier image and a relative shutter introduces al kind of automatic processes to 'improve' the image including increasing the gain. To give a starting point to get better images, the hue might be the best solution to change colours. To get a darker or brighter image, the shutter absolute should be changed.
Once the camera is working, close Matlab, turn on the power supplies of the Elmo's and start Visual Studio.
Open 'VisualTracking' which is pinned and run it. Several screens are opened and if the result is not OK, the HSV filter should be adjusted. To get the value of the hue, saturation of value, there are three screen which show bars which can be used to find the correct threshold. This threshold can be adjusted at line 453 of the program. Once the program has finished, the results can be found in `E:\Mark_x64\Documents\Visual Studio 2010\Projects\VisualTracking\VisualTracking\Metingen`.

H.2 Needle insertion

The setup for the needle insertion is easier, attach the 18L6HD probe to the setup and set the profile on the ultrasound machine to 'NeedleTrack'. Put a 1 mm nitinol needle with a bevel tip of 45° degrees inside the needle insertion device and rotate it such that it moves towards the spine. Put the phantom on the table and run the experiments using the Visual Studio project 'Needle Insertion'. The program asks for several inputs which can be found at the display of the ultrasound machine, if necessary, the brightness of the image can be adjusted on the ultrasound machine by turning the '2D' knob. Sometimes, the display is too bright or dark to see the needle.

I

Abaqus model

This appendix will briefly discuss the Abaqus model. It basically summarizes the settings used in Abaqus.

I.1 Geometry import and mesh

The geometries are imported using the ACIS file format. Abaqus can read this file format by default. The mesh is created using the built-in mesher. The element type is a 10-node quadratic tetrahedron. This element is best suited for 3D stress simulations, the settings are shown in Figure I.1.

The settings for the mesh size for the needle guide is shown in Figure I.2, the setting of the rest of the geometries is shown in Figure I.3. The mesh size has been analysed by applying several meshes. The result of this is shown in Figure I.4. This figure shows that a seed of 2 does not change the result much, therefore a seed of 3 has been chosen.

I.2 Boundary conditions

The boundary conditions of the Abaqus model are the same as the boundary conditions of the SOFA model. The spine is fixed ('Encastre') and the elements which touch the pubic bone are also fixed. Another constraint it used to keep the nodes at the bottom of the phantom at the table. Finally, the needle guide is moved with a displacement.

I.3 Results

The results are exported by using the 'XYData' tool in the results section of Abaqus. This tool allows a user to export any property of a node to a

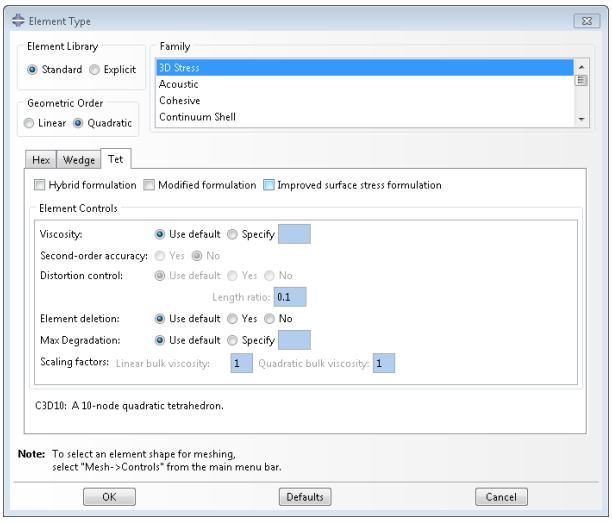

Figure I.1: Used element type in Abaqus

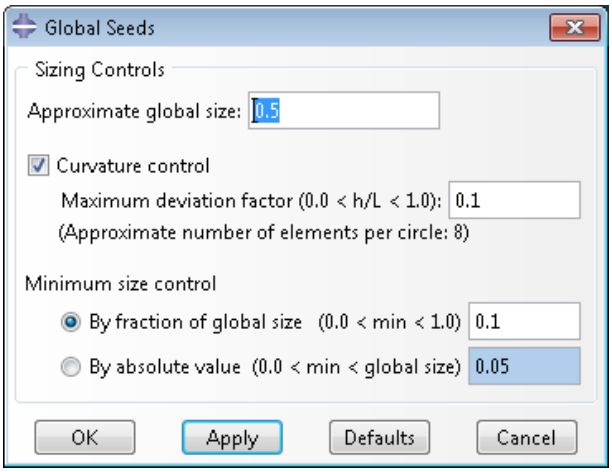

Figure I.2: Used mesh settings for the needle guide in Abaqus

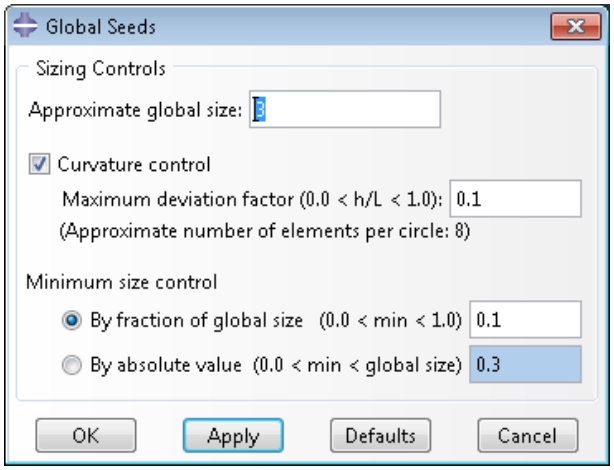

Figure I.3: Used mesh settings for the phantom in Abaqus

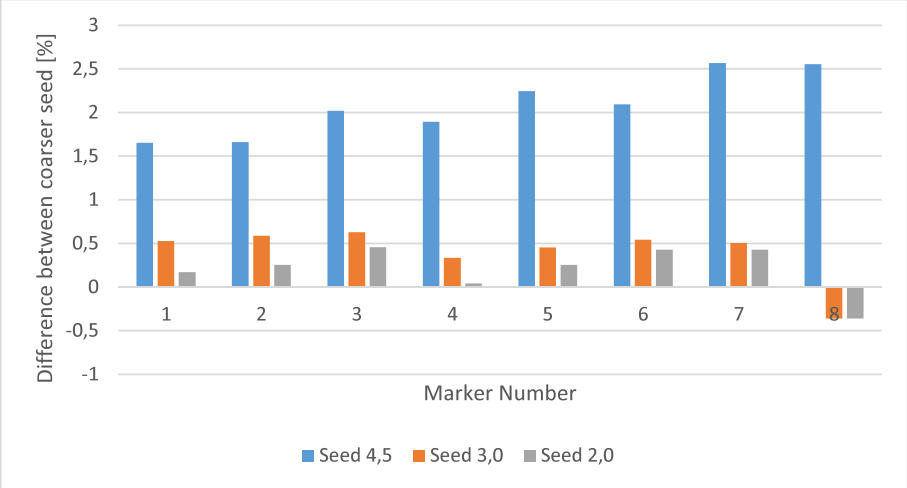

Figure I.4: Verifying mesh size in Abaqus

text file. In this case, the x and y displacement are chosen. This text file is read by Matlab for post processing.

J

Friction experiments

Since the needle insertion plug-in can also handle penetration force and needle-tissue friction. Several experiments have been done to determine these parameters for gels with different elasticities.

J.1 Friction

To determine the friction between needle and gelatine, the micro needle insertion setup is used. This setup is equipped with a six-axis Nano 17 force/torque sensor (ATI Industrial Automation, Apex, USA) located at the base of a nitinol needle with a bevel tip of 45° Deg. The entire setup is driven by Labview, so a Labview model was built to do several tests. Labview sends a command to the motor to drive the needle through a cubical gelatine block with edges of 12 mm and performs a sinusoidal displacement with a amplitude of 1 mm. The amplitude of this motion is small because of the size of the setup and the length of the needle. From these results, the friction for 12 mm gelatine is determined, which is recalculated to N/mm by dividing the result by 12. This was done for three parts of the phantom: rectal wall, surrounding tissue and prostate. The choice for the surrounding tissue might not be obvious but it is penetrated for several mm's between the rectal wall and prostate.

J.2 Penetration force

To determine the penetration force, the same setup was used, the needle was positioned just before the gelatine and then driven 0.5 mm inside the gelatine. The force measured at that point is assumed to be the penetration force. Increasing the displacement resulted in a penetration, this was only done for the rectal wall. Another method would be to use the friction force

and do a linear movement, since the friction per mm is known, the penetration force can be calculated. These linear force measurements have also been done. The data is not processed.