3D Simulation of Needle-Tissue Interaction with Application to Prostate Brachytherapy

Orcun Goksel[1], *Septimiu E. Salcudean*[2], *Simon P. DiMaio*[3]

Degrees: [1] M.A.Sc., [2,3] Ph.D.
Affiliations:
[1,2] University of British Columbia, Vancouver, Canada
[3] Brigham and Women's Hospital, Boston, USA

Running header: 3D Simulation of Needle-Tissue Interaction

Corresponding Author: Orcun Goksel
2356 Main Mall
Vancouver, BC, V6T 1Z4
Canada
Lab Phone: (604) 822-9215
email: orcung@ece.ubc.ca

Abstract

This work presents a needle-tissue interaction model that is a 3D extension of prior work based on needle and tissue models discretized using the Finite Element Method. The use of flexible needles necessitates remeshing the tissue during insertion since simple mesh-node snapping to the tip can be detrimental to the simulation. In this paper, node *repositioning* and node *addition* are the two methods of mesh modification examined for coarse meshes. Our focus is on numerical approaches for fast implementation of these techniques. Although the two approaches that are compared, namely the Woodbury formula (matrix inversion lemma) and the *boundary condition switches*, have the same computational complexity, the Woodbury formula is shown to perform faster due to its cache-efficient order of operations. Furthermore, node addition is applied in constant time for both approaches, whereas node repositioning requires longer and variable computational time. A method for rendering the needle forces during simulated insertions into a 3D prostate model has been implemented. Combined with a detailed anatomical segmentation, this will be useful in teaching the practice of prostate brachytherapy. In this paper, issues related to discretization of such coupled (e.g., needle-tissue) models are also discussed.

Keywords: Fast mesh modification, haptic simulators, needle insertion, tetrahedral remeshing, finite element method, prostate brachytherapy

1 Introduction

Tissue deformation is common during many medical interventions, therefore an accurate simulation of these procedures necessitates counting in these tissue displacements by modeling tissue deformation and medical tool interaction. Deformation modeling has been an active topic in computer graphics for the fitting of noisy data and simulation of clothing, facial expressions, and human/animal characters [1]. Although mass-spring meshes were successfully implemented in [2, 3], the discretization of continuum mechanics equations using the Finite Element Method (FEM) still has significant advantages in deformation modeling. The *condensation* technique introduced in [4] allows fast surface interactions. Computational advances made real-time rates possible for large meshes using quadratic strain and dynamic FEM [5]. Some recent work includes attempts on other elasticity formulations, comparison of iterative solution techniques, offline mesh refinement methods, and contact handling algorithms [6, 7].

The simulation of needle insertion differs from the simulation of other medical tool-tissue interactions in several ways: the needle does not manipulate only the organ surface; friction is a significant force during its interaction; and fine needles are flexible unlike many rigid surgical tools. These three issues were addressed in 2D with a finite-element-based model using the condensation approach, a stick-slip friction model, and local frame rotations with low-rank matrix updates, respectively [8, 9]. Fluid pockets were then introduced onto this system [10]. Earlier needle insertion force modeling mainly focused on the needle base forces. Recently, tip cutting and (velocity-dependent) shaft friction force have been separately modeled [11]. The tip bevel deflection of *highly flexible* needles has also been studied using a non-holonomic bicycle model [12].

Generally, online mesh modification is a common challenge for real-time implementations (e.g., incision in [7, 25]). To retain the tissue mesh nodes on the needle, element subdivision and node repositioning techniques were implemented in [7, 24]. There exists a real-time haptic needle-insertion implementation developed over the work in [11], however it is in 2D and only for rigid needles to save simulation computation time and avoid some instability issues.

This paper presents a 3D needle-tissue interaction model with flexible needles and a specific application of it on prostate brachytherapy, which is a treatment scheme for prostate cancer. To be able to render kinesthetic feedback to user, such a model needs to be simulated at very high rates, typically over 250 Hz. We focus on the computational challenges in achieving such high rates, specifically when the tissue mesh needs to be updated online in order to conform with the penetrating needle.

Prostate cancer is the most common cancer among men in the US with 232,090 new cases and 30,350 deaths estimated in 2005. Low-dose-rate brachytherapy, which is the permanent implantation of low energy radioactive pellets (seeds), is often the treatment of choice for early-stage locally confined prostate cancer [14]. This treatment option has a high probability of eradicating the cancer while preserving healthy tissue, contingent upon good planning and proper application. During a brachytherapy procedure (see in Fig. 1), loaded needles are inserted through the holes of a template grid into the prostate according to a pre-plan. Meanwhile, the physician refers to transrectal ultrasound (TRUS) images to find the needle tip. During a procedure, 80 to 150 seeds are implanted using approximately 25 needles. The brachytherapy needles are 20 cm in length, are quite flexible, and have a significant tip bevel, using which the physician can steer them to desired locations. For instance, target locations behind the pubic arch necessitate such steering.

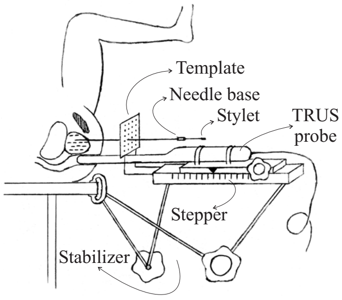


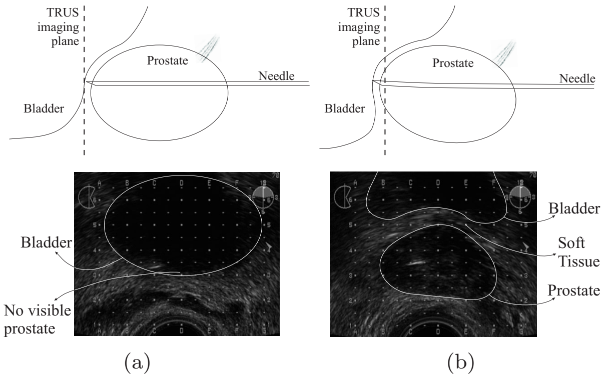
Figure 1: A conceptual overview of the brachytherapy procedure

Despite the low risk of brachytherapy, seed placement errors are still common even by experienced physicians [15]. A sub-optimal application resulting in an undesired dose distribution at the target volume not only decreases the effectiveness of the treatment, but it may also lead to subsequent complications such as impotence [16]. This identified need led us to consider alternative training schemes and consequently to propose a prostate brachytherapy simulator on which medical residents can train.

The prostate is shifted and torqued during needle insertions. In [17], prostate torquing of $\pm 12°$ were recorded on the coronal plane. Figure 2 demonstrates its shift on *in-vivo* TRUS data. The given two images were taken at the same transducer depth within one second, before and after a force was applied on the needle which was already in the prostate. Initially, only the bladder is visible (see Fig. 2(a)) at the TRUS depth of the prostate *base* (its superior side). When the needle is pushed in by the doctor, the prostate is observed to consequently shift into the plane of view in Fig. 2(b). As a result, a 3D deformation model is needed for brachytherapy simulation. The haptic (kinesthetic) feedback at the needle base (e.g., prostate membrane puncture, or pubic arch collision) should also be rendered to the user for a realistic procedure simulation.

There exist graphical interfaces designed for prostate intervention simulations [18, 19] and graphical rendering of prostate surface modeled using the 3D ChainMail method [20], which is not a physically-based approach. Nevertheless, a physically-based 3D prostate simulator has not yet been achieved.

In this work, we present a 3D extension of the needle-tissue interaction model in [9]. The modification techniques, node repositioning and node addition, are compared for coarse meshes using this model. Two numerical approaches to inverted matrix update, the boundary condition switches and the Woodbury formula (matrix inversion lemma), are presented and it is shown that the latter can perform faster due to its operations being cache-efficient. Furthermore, node addition is shown to require constant computational time regardless of the local mesh structure, whereas node repositioning performs in longer and local-mesh dependent time. In the rest of this paper, first obtaining a valid coarse mesh from prostate segmentation data is presented. Then, two remeshing techniques and possible numerical approaches are demonstrated. Next, the extension to 3D needle-tissue interaction model is introduced and our application to needle insertion for prostate brachytherapy is described while exemplifying with a sample insertion data. Finally, discretization related issues, which may deteriorate the realism of such a model unless handled properly, are discussed.



(a) (b)

Figure 2: The observed prostate shift during needle insertion

2 Materials and Methods

2.1 Mesh Generation

Simplices are the most common type of elements in FEM domain discretization. This led us employ tetrahedra in our anatomical model. For finite element analysis, the first constraint is having *high quality* tetrahedra [21] in order to minimize errors caused by the discretization geometry. These error can simply be reduced by using smaller elements. However, many mesh-processing algorithms (including our routines for model deformation, needle-tip element interception, and possible future extensions for visualization) are negatively affected by the increasing number of nodes, which sets the second constraint. Finally, the third one is posed by the need to model the anatomical boundaries: Different anatomies should be separated by tetrahedra surfaces so that one element does not enclose more than one tissue type.

The segmentation data from the British Columbia Cancer Agency (BCCA) is in the form of polygons delineating prostate boundary on each parallel slice of TRUS volume. First, Nuages [22] is applied on this set of polygons to obtain the

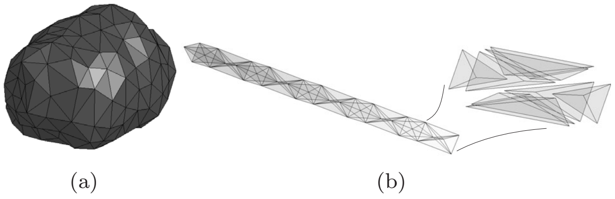


Figure 3: (a) Prostate surface obtained by Nuages, and (b) the axially symmetric 3D needle-mesh design

prostate surface definition shown in Fig. 3(a). Then, the interior and the exterior prostate were meshed using an off-the-shelf meshing application, GiD [23], utilizing the advancing fronts technique.

The needle bending is modeled using FEM with a quadratic strain assumption due to large strains generated by even modest bending. Needle deformation for given forces is solved using the iterative approach described in [24]. Since the speed of this technique significantly depends on the needle mesh size, the needle is modeled with as few elements as possible. On the other hand, for many simple tetrahedral mesh designs, the deflection of this model was not axially symmetric. Thus, a special needle segmentation having axially symmetric force-response and small number of elements is designed as shown in Fig. 3(b) . In terms of the tissue interaction, since the needle has negligible thickness, it is assumed to be a set of line segments passing through the center of the actual 3D needle model.

2.2 Tissue Remeshing

2.2.1 The Need for Remeshing

A finite element discretization of tissue can only be interacted via its nodes. Thus, a needle model can apply boundary conditions only on a subset of these tissue mesh nodes, namely *contact nodes.* Ideally, contact nodes should coincide with the needle shaft so the constraints can be applied to the tissue at the right locations. However, the tissue nodes by nature do not exist exactly on the needle path, where our simulation employs remeshing to obtain one. *Remeshing* refers to recomputing the tissue stiffness matrix for a new local discretization, so that a tissue node will coincide with the needle tip making its addition on the shaft as a contact node possible.

In the literature, many studies avoided the computational burden of remeshing. In the suturing implementation of [25], constraints were interpolated over the corners of each intercepted element surface. This becomes very difficult, if not invalid, for needle insertion due to boundary condition changes and local frame rotations of the constraints. Other attempts include snapping the closest node to the tip without remeshing [13]. This results in high force discontinuities, which may incur model instabilities with flexible needles. Besides, not applying the constraints at the correct locations will reduce simulation accuracy, let alone its visual plausibility. As a result, remeshing the tissue is a necessary action prior to introducing a new contact node, to achieve a physically valid model with one-to-one correspondence between the needle shaft and the contact nodes. In our model, each new contact node is introduced when a new element surface is penetrated.

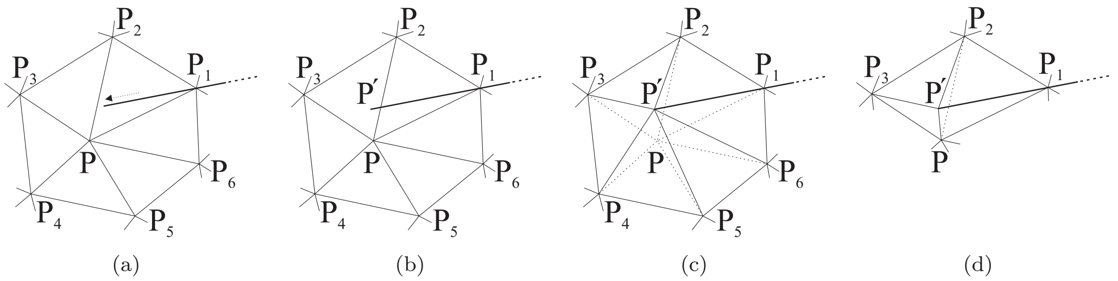
2.2.2 Remeshing Techniques: Repositioning vs. Addition

One approach to remeshing is recursively refining the mesh (locally) until a node is close enough to the tip [7, 24]. This can easily be implemented similar to widely known octree techniques. However, it is a slow process since the number of recursions is indefinite for coarse anisometric meshes. Two other methods of obtaining a node at a desired (tip) location are *repositioning* and *addition* (see Fig. 4). In the former method, surrounding elements are removed and new ones are added such that a corner of them coincides with the desired position. In the latter treatment a new node (consequently new rows and columns to the tissue stiffness matrix) are added and the elements are rearranged locally in order to accommodate this new node.

In our 3D simulation, remeshing is done at the tip for every new element encounter by modifying only the immediate neighbouring elements. *Repositioning* and *addition* are shown in Fig. 4 in 2D for simplicity. In Fig. 4(a) the needle tip is advancing inside an element. Although P is depicted with 6 neighbouring triangles, the number of its neighbours depends on the mesh topology and ranges up to 40 tetrahedra for some nodes in our particular 3D prostate mesh. Computationally, a new interception is identified by checking neighbouring elements for whether the tip is in them. Fig. 4(b) shows the tip which has just crossed a new element boundary at a simulation iteration.

Note that in *repositioning*, the closest corner of the penetrated surface is chosen to minimally disturb the tissue mesh geometry. Still, one possible problem arises if this node belongs to an anatomical boundary. For instance, if P_4PP_6 in Fig. 4 were the surface of a structure, then *repositioning* would degenerate it. To better visualize this severe drawback of *repositioning* which does not exist in *addition.* Assume that this neighbouring anatomy is a bone and that a coarse mesh is used, then the bone could be severely "reshaped" and the needle may be even collided with the bone, when there is in fact a clear pass.

The 3D simulation uses the inverse of $3n \times 3n$ tissue stiffness matrix $\boldsymbol{K}$ for deformation modeling



(a) (b) (c) (d)

Figure 4: The simulation step (a) before and (b) after a new element penetration, where the tissue is remeshed using (c) node repositioning and (d) node addition techniques

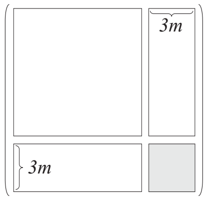
where n is the number of tissue mesh nodes. Considering the dimensions of $\boldsymbol{K}^{-1}$, a real-time full-rank inversion is impractical for remeshing. In this paper as a solution to this, two methods of fast $\boldsymbol{K}^{-1}$ treatment are implemented and compared. The first one, taken from [24], employs boundary condition switches whereas the second new approach is a one-step algebraic solution. Both approaches require the differential $3m \times 3m$ stiffness matrix ΔK where m is the total number of nodes involved in remeshing. ΔK is the desired change in $\boldsymbol{K}$ to obtain the remeshed stiffness matrix $\boldsymbol{K'}$. So,

$$\boldsymbol{K}'_{3n\times 3n} \;=\; \boldsymbol{K}_{3n\times 3n} \bigoplus \Delta K_{3m\times 3m} \qquad (1)$$

where $\bigoplus$ adds the small matrix on the right onto the corresponding elements of interest on the left.

In *repositioning*, m is equal to the node *valence* (which is the number of edges emanating from the node) of P plus 1 (e.g., 7 for the 2D case in Fig. 4(c)). Node valance may be quite large in 3D as it is in our particular mesh. Here comes a significant advantage of *node addition*, that is m being constant for any mesh topology such that it is always 6 in 3D (and 5 in 2D as seen in Fig. 4(d)). ΔK can easily be found by subtracting the stiffness matrices of the elements to be removed from the ones to be added. This procedure requires negligible time compared to the rest of both algorithms that need to apply this change on the *inverted* matrix,

$$\mathbf{K}^{-1}_{(3n\times 3n)} = \qquad . \qquad (2)$$



For simplicity, let us assume that the rows/columns are reordered to have the m nodes of interest at the end.

2.2.3 Numerical Approaches: BCS vs. Woodbury

The first method for fast $\boldsymbol{K}^{-1}$ treatment performs BCS on the $3m$ corresponding rows/columns of $\boldsymbol{K}^{-1}$ so that the shaded submatrix of $\boldsymbol{K}^{-1}$ is entirely switched to displacement boundary conditions. A boundary condition switch on row/column i can computationally be achieved using the following low-rank update:

$$\boldsymbol{K}' = \boldsymbol{K} - \frac{1}{k_{(i,i)}}\,\boldsymbol{c}_i\,\boldsymbol{r}_i \qquad (3)$$

where $\boldsymbol{c}_i$ and $\boldsymbol{r}_i$ are the corresponding i^{th} column and i^{th} row of $\boldsymbol{K}$, and $k_{(i,i)}$ is the pivot i. This is similar to the formula employed in [8] for stick-slip friction model of contact nodes.

Then, ΔK can be added onto this shaded region and their boundary conditions may be switched back. Considering each BCS takes $(3n)^2$ operations, this approach requires $2(3m)(3n)^2$ total operations. Throughout this paper, *operations* refer to floating-point multiplications and divisions only.

The second treatment employs the Woodbury formula, also known as the matrix inversion lemma, which is an extension of Sherman-Morrison formula. It allows a perturbed inverse $(\boldsymbol{K}')^{-1}$ to be computed for a given change ΔK to the original matrix as follows:

$$\left(\mathbf{K} + V\,\Delta\mathrm{K}\,V^T\right)^{-1} = \mathbf{K}^{-1} - \Big(\underbrace{\mathbf{K}^{-1}V}_{\boldsymbol{\Lambda}_{(3m\times 3m)}} \cdot \underbrace{\Delta\mathrm{K}\big(\mathrm{I} + \underbrace{V^T\,\mathbf{K}^{-1}\,V}_{3m\times 3m}\,\Delta\mathrm{K}\big)^{-1}}_{\boldsymbol{Q}_{(3m\times 3m)}} \cdot \underbrace{V^T\,\mathbf{K}^{-1}}_{\boldsymbol{\Lambda}^T_{(3m\times 3m)}}\Big) \qquad (4)$$

where V is a $3n\times 3m$ zero matrix with $3m$ corresponding rows taken from the identity matrix $I_{3m\times 3m}$. In (4), $\boldsymbol{K}^{-1}V$ and $V^T\boldsymbol{K}^{-1}$ are computationally-costless column and row selections, respectively. These basically obtain the last $3m$ columns and rows in (2). The rest operates on $V^T\boldsymbol{K}^{-1}V$ which is again a costless selection of elements shaded in (2). The matrix computations in the middle costs approximately $3(3m)^3$ operations considering the low-rank inverse computed via LU factorization using Gaussian elimination. Adding the costs of final matrix multiplications shown in (4), the full-blown Woodbury approach takes $3(3m)^3 + (3m)^2 3n + 3m(3n)^2$ operations.

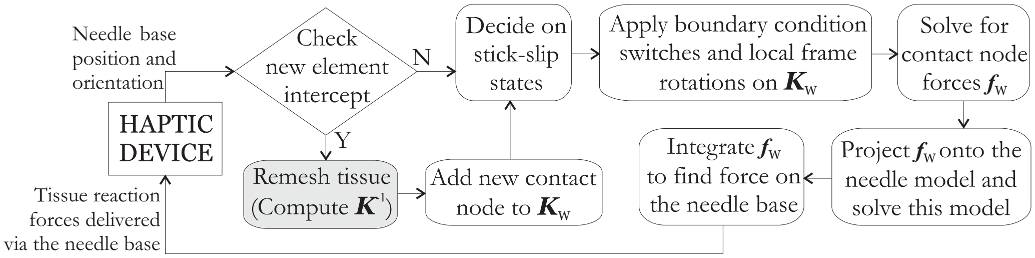


Figure 5: The flowchart of a typical iteration during 3D needle-tissue interaction simulation

2.2.4 Cache-efficiency of the Woodbury formula

On modern computers, the performance of programs is often limited by memory latency rather than by processor cycle time [26]. A cache access can be up to 10 times or more faster than a RAM access depending on the architecture. Cache-aware algorithms with increased locality and block processing were proposed in the literature for various large scale computational processes and were utilized in many popular linear algebra packages, such as LAPACK [27]. Below, it is shown that the Woodbury update has a cache-efficient structure compared to the BCS.

Considering $n \gg m$, both the BCS and the Woodbury update have $\mathcal{O}(mn^2)$ complexity. Nevertheless, the Woodbury formula in practice has a major computational advantage, since it can update each individual element of K^{-1} independently. In other words, once Q is computed, K_{ij}^{-1} can be updated given the i^{th} row and Λ and the j^{th} column of Λ^T. On the other hand, each BCS needs to completely traverse all elements of K^{-1} before the next BCS can start. Therefore, the full-rank $K_{3n\times 3n}^{-1}$ has to be updated $2\times 3m$ times (switching forth and back) in order to get one and hence all the elements of the new K^{-1}. As a result, is the Woodbury update is a cache-effective computational approach: Provided that Λ (and Q) fits in the (L2) cache memory, the entire K^{-1} can be traversed with only $(3n)^2$ cache misses. Whereas, the BCS causes $6m(3n)^2$ cache misses unless the entire K^{-1} fits in cache, which is practically not possible.

2.3 3D Needle-Tissue Interaction

In-vivo tissue deformation is accepted to be nonlinear and anisotropic. Although the organic tissue properties have been studied extensively in biomechanics, the proposed models for its complex behaviour (e.g., dynamic implementations of quadratic strain) have not been rigorously validated. In our quasi-static model, the Hookean linear-elasticity with linear-strain is assumed for the tissue and this continuum equations are discretized over a tetrahedral mesh using the FEM.

Most techniques in our simulation have been incorporated from [9] and extended to 3D. A flexible needle is coupled to the deformable tissue mesh via *contact nodes*. For fast computation, a condensed system matrix is employed [4]. This system matrix is maintained at each iteration to accommodate the shaft orientation changes [9]. It is further processed each time a contact node changes its frictional (*stick-slip*) state on the shaft, so that the condensed system can compute for the proper boundary constraint [9]. A typical simulation cycle is given in Fig. 5. The shaded block in Fig. 5 marks the remeshing operation, in which this paper mainly focuses on. Note that the boundary condition switch operation that is applied on the condensed matrix for stick-slip friction model [9] is in this paper further utilized for remeshing as described in the previous section. Throughout this paper, BCS refers to this latter remeshing scheme, which targets the entire $3n \times 3n$ inverse tissue stiffness matrix $\boldsymbol{K}^{-1}$ (rather than the condensed matrix as in [9]).

3 Results

Two remeshing techniques, BCS and the Woodbury update, were implemented in C on a P4 3.2 GHz system with 768 MB 333 MHz DDR SDRAM and 1 MB L2 cache. Figure 6(a) presents the expected quadratic relation of the processing time needed to reposition one single node (with $m = 8$) for varying mesh sizes (n). It is observed that the Woodbury update takes, on average, half the time of BCS due to its cache-efficient structure. Figure 6(b) demonstrates the superiority of node addition regarding computational time. For instance, in our sample prostate mesh, some nodes have up to 23 neighbours, which results in 2.5 s remeshing time even using the Woodbury update. Furthermore, the number of neighbours is different for each node in an unstructured mesh. On the other hand, using addition fixes m at 6 regardless of the mesh. For the given mesh size, this leads to implementation under 400 ms per remeshing.

The prostate region obtained using advancing fronts consists of 570 nodes and 2801 tetrahedra. The quality of this mesh for FEM simulation was assessed

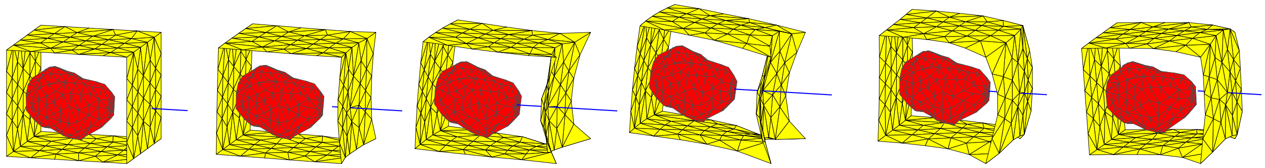


Figure 7: The tissue mesh while the needle is inserted, moved upwards, and retracted

observing some measures [21]: The shortest tetrahedra edge is 3.06 mm and the longest one is 24.04 mm; the minimum and the maximum dihedral angles in this mesh are 9.94° and 159.49°, respectively. Furthermore, only a couple elements have these borderline values showing that the geometry of elements are acceptable for a reasonable FEM approximation. To fix this model in space, four inferior corners of the meshed tissue region were set as the fundamental displacement boundary condition. This should ultimately be the pubic arch segmentation.

The 3D needle insertion simulation has been implemented in C for the ease of a future port on a real-time operating system for fast haptic control. For testing and debugging purposes, a Matlab interface, where the needle can be manipulated and the tissue deformation can be projected, has been developed (see Fig. 7) on top of the fast C implementation. In Fig. 8, the three axis components of the needle base force are plotted for a base manipulation of pushing in the tissue, moving upwards, and retracting, in order (as seen in Fig. 7). The only significant torque component, which is on the sagittal plane due to the upward movement, is also plotted in this figure. The forces during the penetration into the skin and the prostate are observed after 0.1 s and 0.6 s, respectively, after the simulation starts. The needle base was started to be moved upwards at 1.3 s. As it was being pulled back between 1.4 s and 2.1 s, the tissue bulk can be observed to stay stuck on the shaft (due to stick-slick friction model) while bulging outwards.

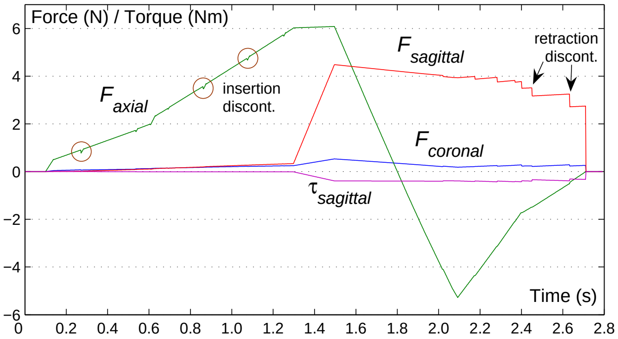
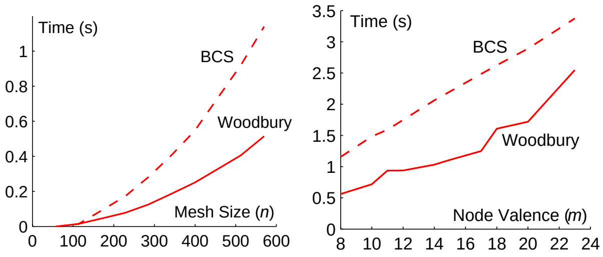


Figure 8: Axial, sagittal, and coronal forces on the needle base, and the sagittal torque

The force discontinuities observed during the insertion and the retraction are discussed in the next section. Although they can be damped further prior to the application on the user's hand, they are plotted untouched here in order to give an idea of their magnitude.



(a) constant m=8

(b) constant n=570

Figure 6: Computation time of node repositioning for varying (a) mesh size and (b) node valence

4 Discussion

4.1 Remarks on Node Addition

Aside from the constant short computational time, node addition has another major advantage that the node-added mesh complies with the initial anatomical boundaries, whereas node repositioning may disturb them. For instance, assuming that P_1PP_3 is an anatomy surface in Fig. 4, the given repositioning would alter it. To avoid populating the mesh with extra nodes using addition, the added ones are later removed during retraction. A copy of K^{-1} is saved prior to every mesh modification, so this removal can easily be performed by restoring this copy. The implementation of node addition using the Woodbury update, which has been concluded to be the best remeshing combination, requires attention to the point that new rows/columns are needed in $\boldsymbol{K}^{-1}$ for the added node. Using BCS, this can easily be achieved after the first set of switches (prior to ΔK addition) by introducing three new all-zero rows and columns. On the other hand, the Woodbury formula necessitates an initial matrix with those rows/columns, some elements of which need to be theoretically assigned to *infinity*. Considering very large numbers result in poor $\boldsymbol{K}^{-1}$ conditioning, an empirically-determined large number is used deliberating this numerical error trade-off.

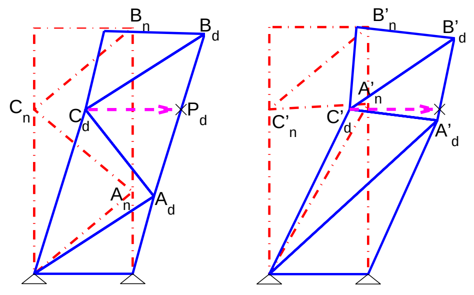
4.2 Remeshing Point

In remeshing, the new node position is preferred to be found by transforming the needle tip position in the deformed tissue into the nominal (pre-deformed) mesh using *basis functions*. However, since the tissue discretization is changed during remeshing, the previous contact node displacements/forces do not necessarily deform the tissue in the same way as in the previous iteration. This causes small (negligible) force spikes during the needle insertion. Let us show this effect on an example, where the needle penetrates into a tissue sample at C_n and deforms it as seen in Fig. 9(a) just before touching the edge A_dB_d (to simulate the needle shaft, node C is constrained on a horizontal line while a rightward force is being applied). In this example, the tissue is modeled employing quadratic-strain assumption. Intuitively, point P_d, which lies right in front of the tip, would be chosen as the desired remeshing location for the new node. First, the corresponding point A'_n in the nominal mesh is to be found. This requires a linear transformation from the deformed to the nominal mesh (e.g., here, from line segment A_dB_d to A_nB_n). Once the remeshing is performed and the constraints are re-applied onto this new discretization, the new node A'_n may adversely not deform onto the tip (see Fig. 9(b)). Consequently, snapping this node to the tip in the next iteration causes a force discontinuity at the needle tip. Although, this example is presented using repositioning here, the argument is still valid for both remeshing techniques.

A node that would end up at the tip position could be computed using iterative optimization techniques. However, there exists another side-effect of remeshing. Once the constraints of the previous iteration is re-applied on the new discretization, the previously-added contact nodes will not generate the same transversal forces and the same axial deformations along the shaft as in the previous iteration (this can be observed on C'_d in Fig. 9(b)). Thus, in order to theoretically avoid all discontinuities, remeshing only the tip node for each contact node addition is not enough. Instead, all contact node positions should be remeshed using an optimization scheme at each new contact node addition. This would require a significant computational time, which cannot be afforded in a real-time application. Furthermore, as seen in our simulation results, its effect is negligible. Therefore, remeshing only one node and simply onto the current needle tip position was implemented and the possible discontinuities were damped in our model.

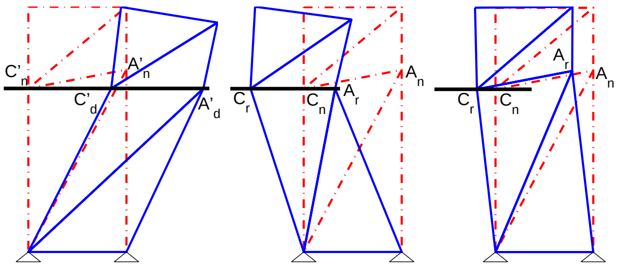
4.3 Contact Node Removal

During retraction, another type of force discontinuity occurs due to the tissue discretization. As seen in Fig. 8, these retraction force discontinuities are of relatively high magnitude and may consequently be detrimental to the simulation. They occur due to the sudden removal of the transversal force from the tip when a contact node slips off the needle tip. Such a transversal force might be induced on the shaft if the needle base is shifted transversally after the insertion. Furthermore, this force can also be very easily induced with the rotation of surrounding elements due to the stick-slip behaviour of the needle. Shift of the tissue during the stick state between the insertion and the retraction. If a couple of elements rotate even slightly during this action, relatively large torques can be created on the needle shaft. For instance, assume that the insertion is stopped and the needle starts being retracted as in Fig. 10(a). Here, the nominal position A'_n is computed using iterative *repositioning* so that initially no force is exerted by A'_d on the needle complying to the discussing in the previous subsection. Fig. 10(b) shows the moment that



(a) before *remeshing* (b) after *remeshing*

Figure 9: Tissue mesh (a) before and (b) after remeshing (note that A'_d does not end up at the intended position P_d without iterative optimization)



(a) insertion (b) retraction (c) node removal

Figure 10: The torque accumulated in the tissue between (a) the insertion and (b) the retraction can be observed in (c) when the tip node slips off

the shaft overcomes the friction and starts slipping. Note the force induced on the tip by the lower elements (since the shaft is taken to be solid here, it does not bend). This force can be better visualized observing how far the node A_r jumps, once it slips off the tip (see Fig. 10(c)). Similar forces may be induced on the needle by the torquing of the prostate around the pubic arch. Thus, when the node slips off, sudden removal of this tip force may cause instabilities in the coupled system, which also needs to be taken into account and damped.

5 Conclusions

In this work, the first physically-based 3D interaction model of insertion of a flexible needle into a soft deformable body has been introduced. The proposed system successfully couples the FEM discretization of a linearly-elastic tissue model with a flexible needle. For coupling of coarse tissue meshes with the needle shaft, two remeshing techniques, node repositioning and node addition, have been studied. The latter is demonstrated to require shorter and constant computational time. When comparing two numerical approaches, namely the boundary condition switches and the Woodbury formula (matrix inversion lemma), for fast remeshing implementation, the Woodbury formula performs faster due to its cache-efficient order of operations. Using the techniques presented in this paper for 3D needle-tissue interaction modeling, a prostate brachytherapy simulator has been designed using a prostate mesh with anatomical boundaries extracted from contours segmented on parallel ultrasound image slices.

In the future, the pubic arch needs to be segmented to be used as a key displacement constraint for the prostate. The model will be validated after incorporating further anatomical information. TRUS data taken from clinical cases will be used for this validation (e.g., the prostate shift caused by the needle insertion). Ultimately, the simulation will be implemented on a haptic device. For brachytherapy seed placement, real-time acquisition of tissue elasticity parameters using elastography combined with the 3D simulation presented in this paper, will make accurate intra-operative plan adjustments possible.

References

[1] S. Gibson and B. Mirtich, "A survey of deformable modeling in computer graphics," Tech. Rep. TR-97-19, MERL – Mitsubishi Electric Research Laboratory, 1997.

[2] M. Downes, M. C. Cavusoglu, W. Gantert, L. W. Way, and F. Tendick, "Virtual environments for training critical skills in laparoscopic surgery," in *Proceedings of 6th Medicine Meets Virtual Reality (MMVR)*, (San Diego, CA), pp. 316–322, 1998.

[3] D. Terzopoulos and K. Waters, "Physically-based facial modeling, analysis and animation," *Visualization and Computer Animation*, vol. 1, no. 2, pp. 73–80, 1990.

[4] M. Bro-Nielsen and S. Cotin, "Real-time volumetric deformable models for surgery simulation using finite elements and condensation," *Computer Graphics Forum (EUROGRAPHICS'96)*, vol. 15, no. 3, pp. 57–66, 1996.

[5] Y. Zhuang and J. Canny, "Real-time simulation of physically realistic global deformation," in *SIGGRAPH99 Sketches and Applications*, (Los Angeles, CA), 1999.

[6] G. Hirota, S. Fisher, and A. State, "An improved finite element contact model for anatomical simulations," *The Visual Computer*, vol. 19, no. 5, pp. 291–309, 2003.

[7] H. Nienhuys, *Cutting in deformable objects*. PhD thesis, Utrecht University, 2003.

[8] S. P. DiMaio and S. E. Salcudean, "Needle steering and motion planning in soft tissues," *IEEE Transactions on Biomedical Engineering*, vol. 52, pp. 965–974, June 2005.

[9] S. P. DiMaio and S. E. Salcudean, "Interactive simulation of needle insertion models," *IEEE Transactions on Biomedical Engineering*, vol. 52, pp. 1167–1179, July 2005.

[10] A. H. Gosline, S. E. Salcudean, and J. Yan, "Haptic simulation of linear elastic media with fluid inclusions," *Haptics-e: The Electronic Journal of Haptics Research*, vol. 3, March 2005.

[11] S. P. DiMaio and S. E. Salcudean, "Needle insertion modeling and simulation," *IEEE Trans. Robot. Automat.*, vol. 19, pp. 864–875, 2003.

[12] R. J. WebsterIII, N. J. Cowan, G. Chirikjian, and A. M. Okamura, "Nonholonomic modeling of needle steering." 9th International Symposium on Experimental Robotics, June 2004.

[13] R. Alterovitz, J. Pouliot, R. Taschereau, I.-C. J. Hsu, and K. Goldberg, "Needle insertion and radioactive seed implantation in human tissues: Simulation and sensitivity analysis," in *Proc. of IEEE Int. Conf. on Rob. and Autom. (ICRA)*, pp. 1793–1799, Sep 2003.

[14] W. M. Butler, "Review of modern prostate brachytherapy," in *Proc. of the 22nd Annual EMBS Int. Conf.*, (Chicago, IL), pp. 748–752, July 2000.

[15] R. Taschereau, J. Pouliot, J. Roy, and D. Tremblay, "Seed misplacement and stabilizing needles in transperineal permanent prostate implants," *Radiotherapy and Oncology*, vol. 55, pp. 59–63, 2000.

[16] G. S. Merrick, W. M. Butler, K. E. Wallner, R. W. Galbreath, R. L. Anderson, B. S. Kurko, J. H. Lief, and Z. A. Allen, "Erectile function after prostate brachytherapy," *Int. J. Radiation Oncology Biol. Phys.*, vol. 62, no. 2, pp. 437–447, 2005.

[17] V. Lagerburg, M. A. Moerland, J. J. W. Lagendijk, and J. J. Battermann, "Measurement of prostate rotation during insertion of needles for brachytherapy," *Radiotherapy and Oncology*, vol. 77, pp. 318–323, 2005.

[18] J. K. Hahn, M. J. Manyak, G. Jin, D. Kim, J. Rewcastle, S. Kim, and R. J. Walsh, "Cryotherapy simulator for localized prostate cancer," in *Proceedings of 10th Medicine Meets Virtual Reality (MMVR)*, 2002.

[19] A. N. Kimura, J. Camp, R. Robb, and B. Davis, "A prostate brachytherapy training rehearsal system – simulation of deformable needle insertion," in *Proc. of Med Image Comp. and Computer-Assisted Interv. (MICCAI)*, pp. 264–271, 2002.

[20] X. Wang and A. Fenster, "A haptic-enhanced 3D real-time interactive needle insertion simulation for prostate brachytherapy," in *Proc. SPIE Medical Imaging: Vis., Image-Guided Procedures, and Display*, vol. 5367, pp. 781–789, 2004.

[21] J. R. Shewchuk, "What is a good linear element? interpolation, conditioning, and quality measures," in *Eleventh International Meshing Roundtable*, 2002.

[22] B. Geiger, *Three-dimensional modeling of human organs and its application to diagnosis and surgical planning.* PhD thesis, École Nationale Supérieure des Mines de Paris, France, 1993.

[23] R. Löhner, "Progress in grid generation via the advancing front technique," *Engineering with Computers*, vol. 12, pp. 186–210, 1996.

[24] S. P. DiMaio, *Modelling, Simulation and Planning of Needle Motion in Soft Tissues.* PhD thesis, Univ. of British Columbia, 2003.

[25] J. Berkley, G. Turkiyyah, D. Berg, M. A. Ganter, and S. Weghorst, "Real-time finite element modeling for surgery simulation: An application to virtual suturing," *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, no. 3, pp. 314–325, 2004.

[26] I. Kodukula and K. Pingali, "Data-centric transformations for locality enhancement," *International Journal of Parallel Programming*, vol. 29, no. 3, pp. 319–364, 2001.

[27] E. Anderson, Z. Bai, and C. B. *et al.*, *LAPACK Users' Guide.* SIAM, Philadelphia, second edition ed., 1995.