Simulation of Needle Insertion and Tissue Deformation for Modeling Prostate Brachytherapy

Nuttapong Chentanez
Department of Electrical Engineering and Computer Science, U.C. Berkeley

Ron Alterovitz
Department of Computer Science, University of North Carolina at Chapel Hill

Daniel Ritchie
Department of Electrical Engineering and Computer Science, U.C. Berkeley

Lita Cho
Department of Electrical Engineering and Computer Science, U.C. Berkeley

Kris K. Hauser
School of Informatics and Computing Indiana University

Ken Goldberg
Department of Industrial Engineering and Operation Management, U.C. Berkeley

Jonathan R. Shewchuk
Department of Electrical Engineering and Computer Science, U.C. Berkeley

James F. O'Brien
Department of Electrical Engineering and Computer Science, U.C. Berkeley

Abstract

Accurate simulation of needle insertion during Brachytherapy can be used both for training and in automated planning to reduce errors between intended and actual placement of the needle tip. We have developed a 3D tetrahedral finite element simulation that models tissue deformation, needle flexation, and their coupled interaction. This system addresses the following applications:

- Training physicians to compensate for tissue deformation during needle insertion.
- Incorporating a model of tissue deformation into manual and automated planning for seed placement and dose distribution.
- Developing automated steering procedures for robotic devices

We model tissue elasticity with constitutive equations discretized over a 3D tetrahedral mesh by a finite element method. The needle is modeled as a stiff elastic rod. The two systems are coupled together by shared nodes, and the tissue and needle are dynamically remeshed to allow needle insertion and withdrawal. Nodes are dynamically positioned along a curvilinear needle path in a volumetric mesh, enabling the simulation to apply accurate cutting and frictional forces along the needle shaft and at the needle tip.

Core Contributions

Fully 3D real-time needle insertion simulator.

- Support for both stiff and highly flexible needle shafts with symmetric or bevel tips.
- Novel algorithms for local remeshing that enforce the conformity of a tetrahedral mesh to a curvilinear needle path, enabling accurate computation of contact forces.
- An efficient method for coupling a 3D finite element simulation with a 1D inextensible rod with stick-slip friction.
- Optimizations to allow real-time performance using high resolution models.

Contact:
Prof. James F. O'Brien
Electrical Engineering and Computer Science
University of California, Berkeley
Email: job@berkeley.edu
Project Website:
http://graphics.cs.berkeley.edu/papers/Chentanez-ISN-2009-08

Simulation Overview

Nonlinear Finite Element Model for Tissue
- Tetrahedral elements
- Co-rotational method to cope with large deformations
- Invertible elements for robust stability

Discrete Elastic Rod Model for Needle
- Series of connected one dimensional elastic rod elements
- Torsion and bending forces computed at segment connections

Coupling Between Needle and Tissue
- Conforming meshes with shared nodes and dynamic remeshing
- Coupling handled by hard constraints
- Friction and cutting force using unilateral constraints

Simulation Loop
- Compute forces and Jacobians for tissue and needle systems
- Solve coupled system for friction states and nodal accelerations
- Update positions and velocities of tissue and needle
- If the needle tip has moved through tissue, remesh around the tip to ensure good element quality
- Reparameterize the needle along FEM mesh edges

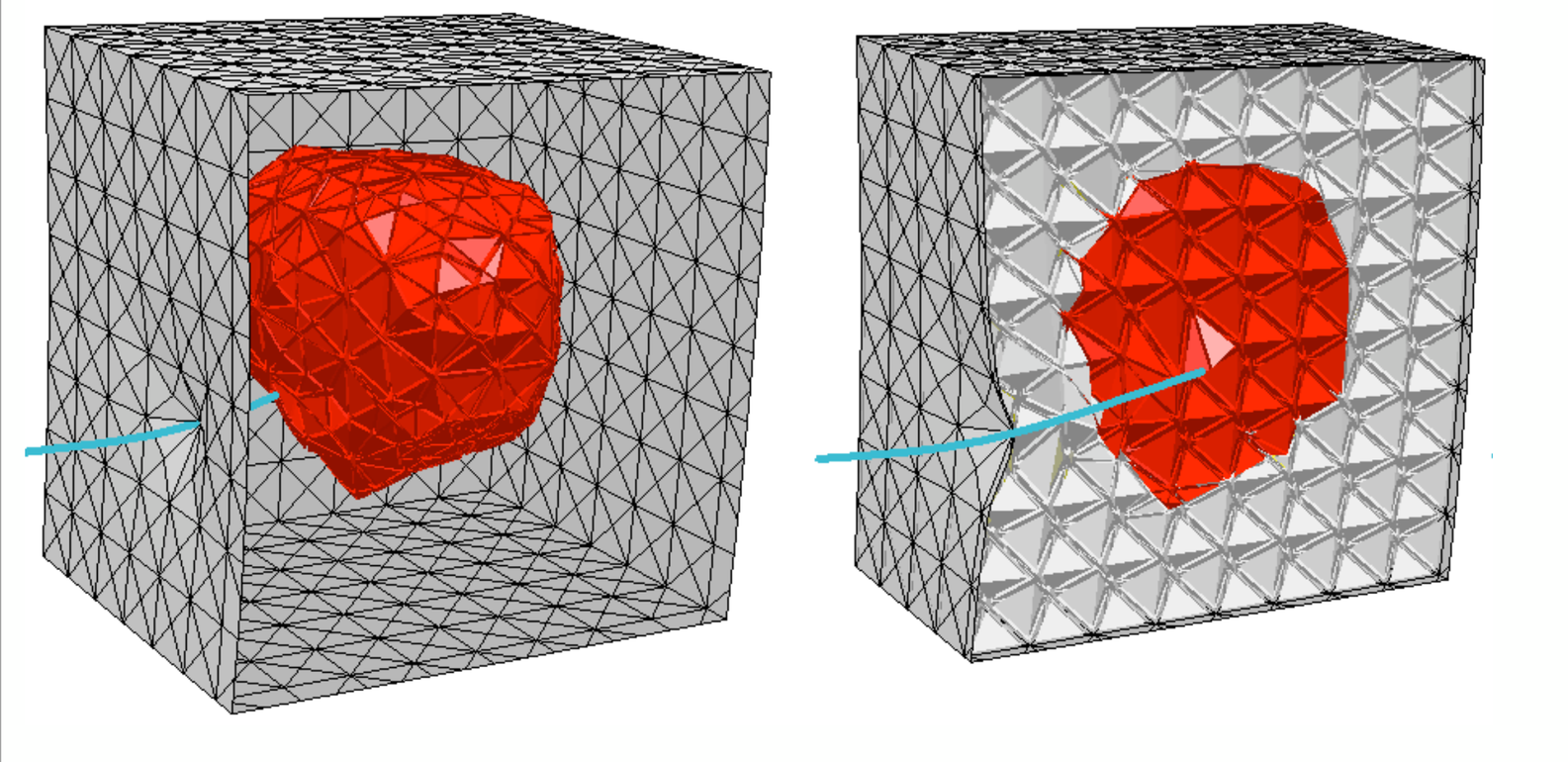
Performance Optimizations

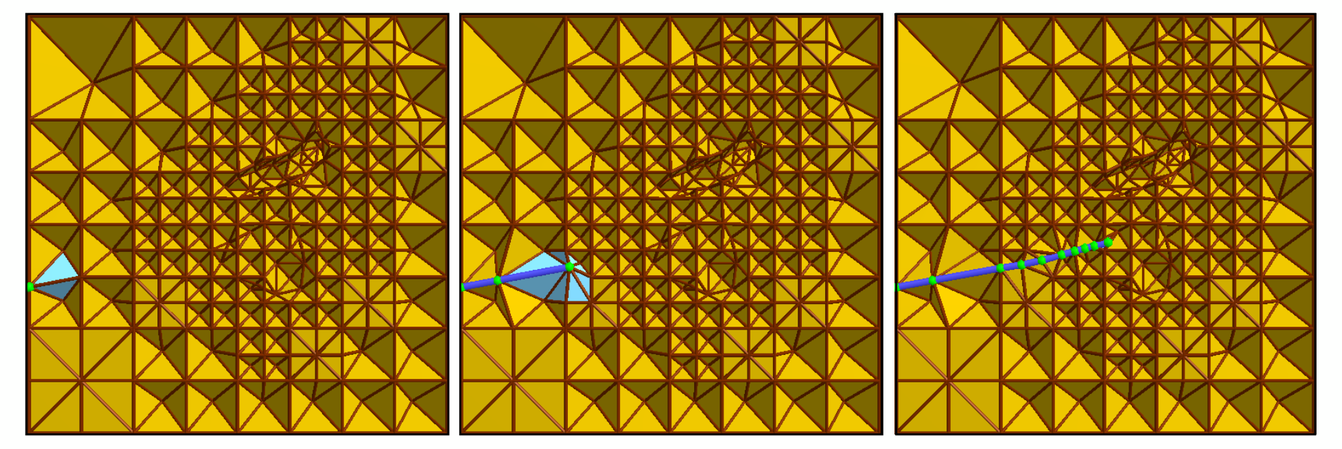
Accelerating Search for LCP Solution
- Initialize contact states using previous trial solutions
- Abort CG if tolerance excludes feasible solution
- Restart CG prom previously aborted solutions

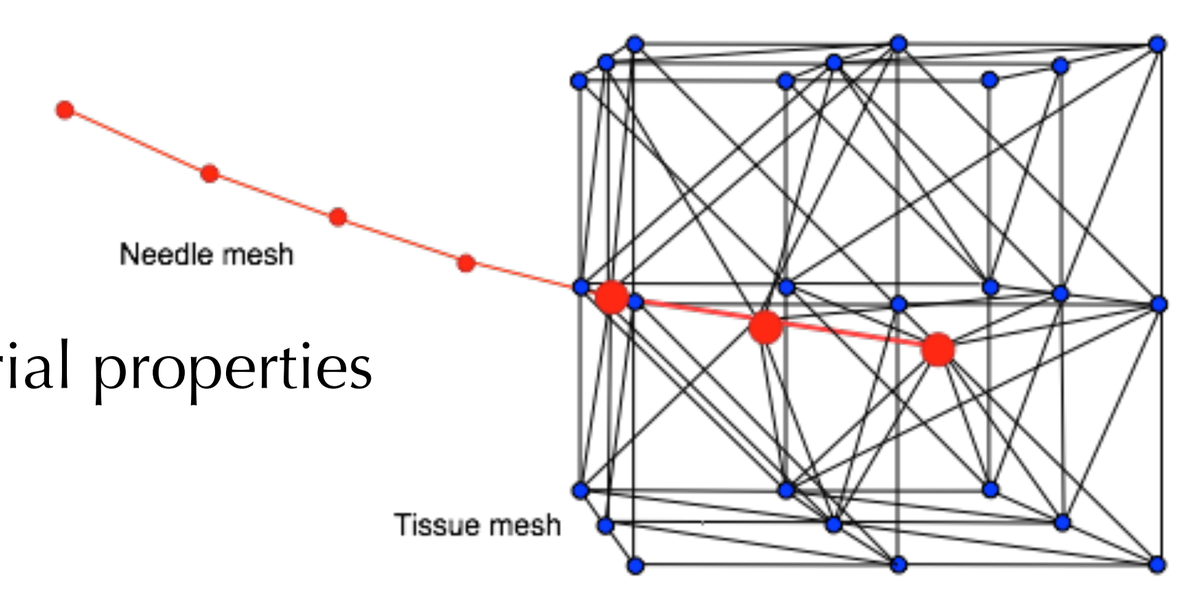
Accelerating CG Iterations
- Multi-threaded CG and Jacobian computation
- Arrange nodes to improve cache coherency
- Arrange tetrahedra to minimize lock congestion

Approximate Jacobian
- Perform partial updates only when bounds on element matrices exceed a threshold
  [Lazy update: 83% reduction in computation cost with no more than a 0.2% relative error in the needle tip position.]
- Avoid reassembly due to remeshing by zeroing unused entries

Discretization



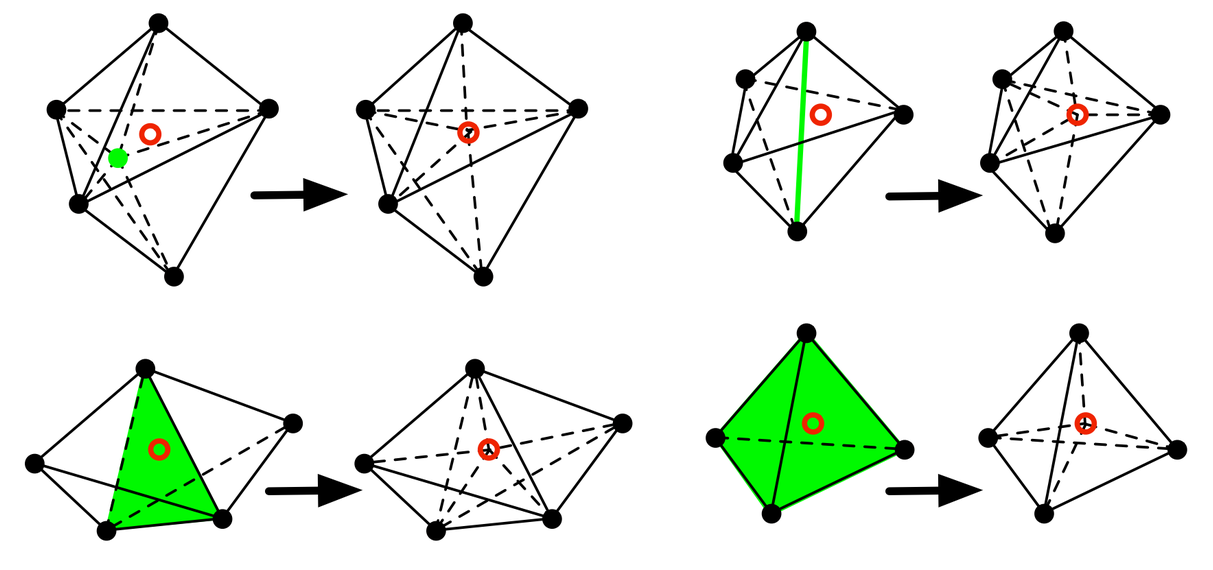




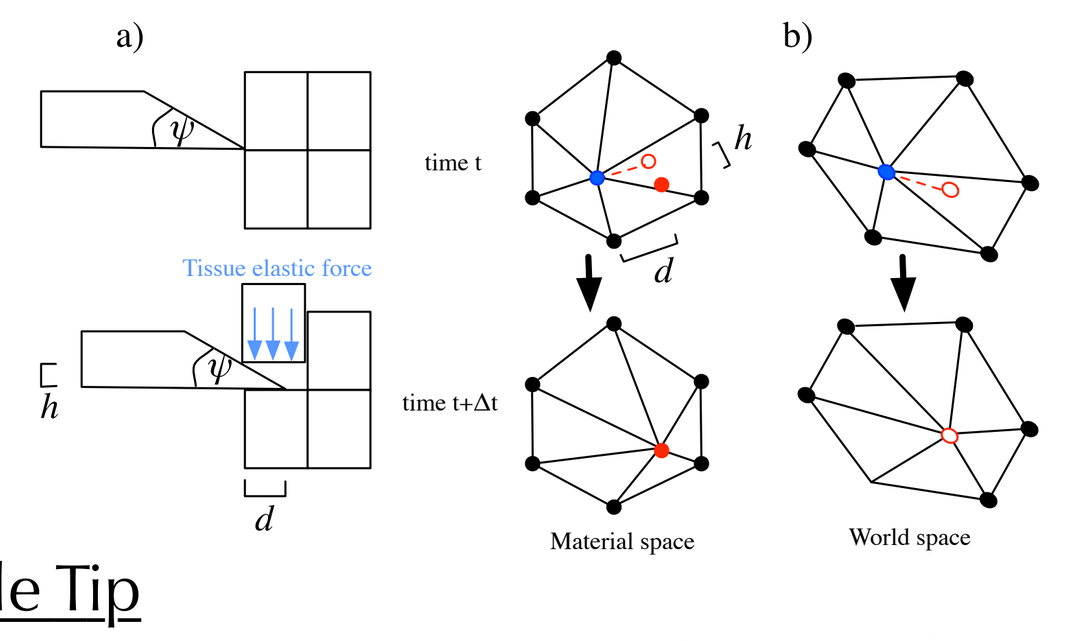
- Hybrid regular/Irregular tetrahedral mesh
- Internal boundaries separating regions with distinct material properties
- Embedded triangular membranes
- Needle represented by edges with shared nodes
- Dynamic remeshing to accomodate needle movement

Remeshing

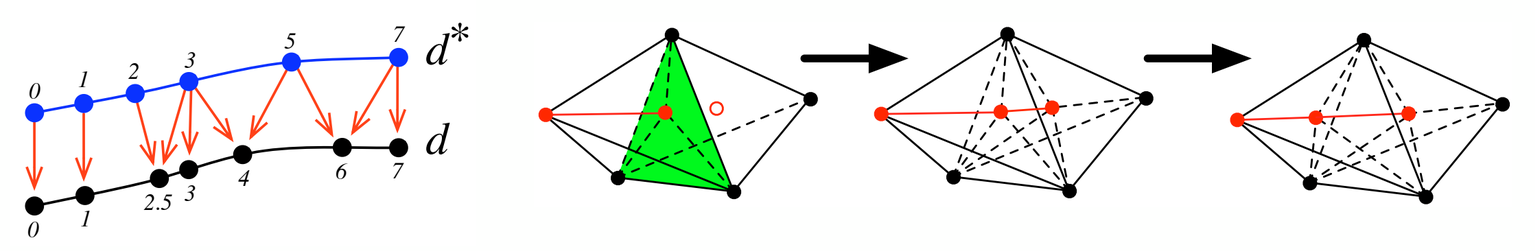
After a simulation step, nodes no longer align. Need to locally remesh tissue and reparameterize needle discretization.



Tissue Remeshing
- Place a node at the new tip location
- Maintain good mesh quality v/l^3_{rms}
- Simple tracking until node quality below threshold
- Select between tet-split, edge-split, face-split, and node snap
- Keep track of all topological changes in stack
- Consider undoing previous operations before performing new ones
- Undo them as the needle is retraced



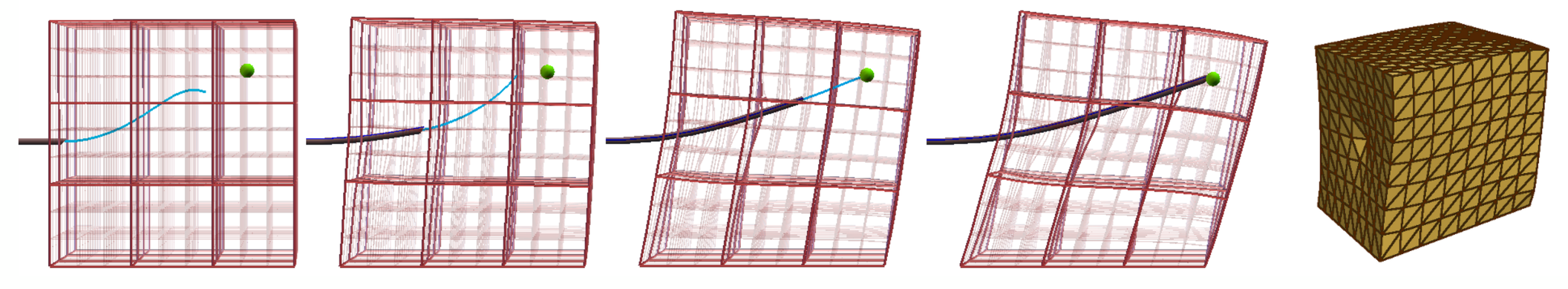
Needle Reparameterization
- Resample needle to match new node positions



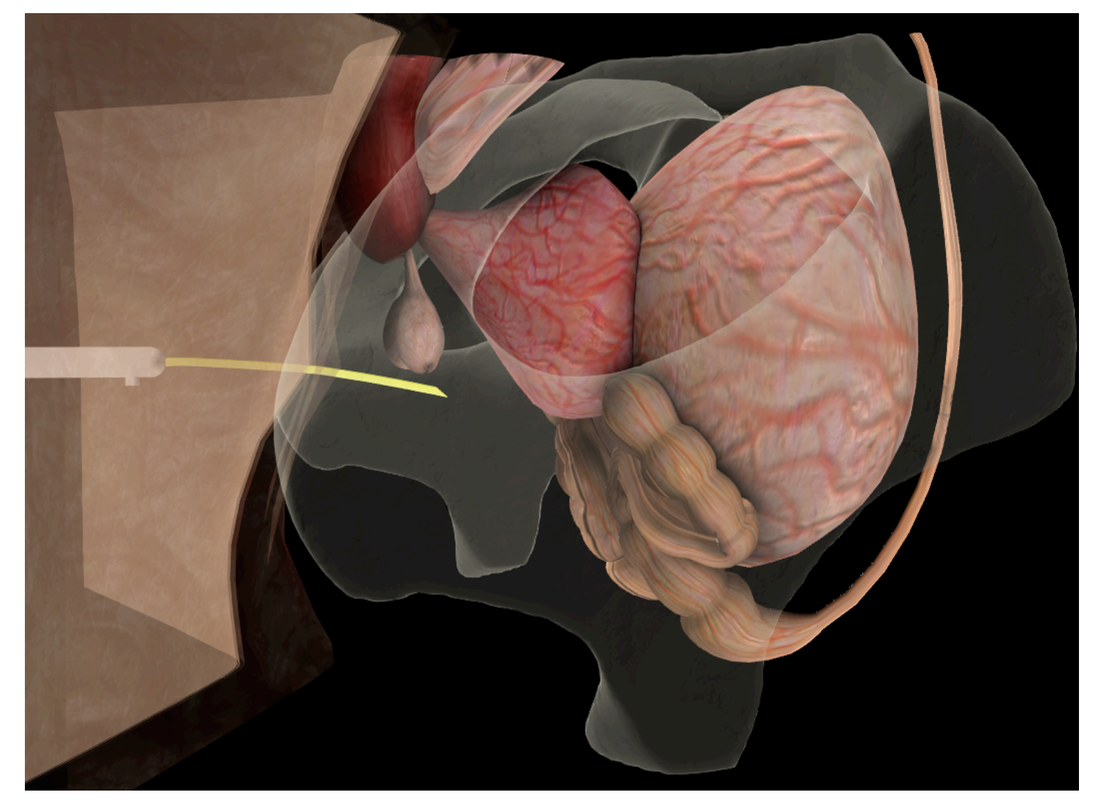
Needle Tip
- Create asymmetric cutting force by biased adjustment of the material-space parameterization

Planning and Steering

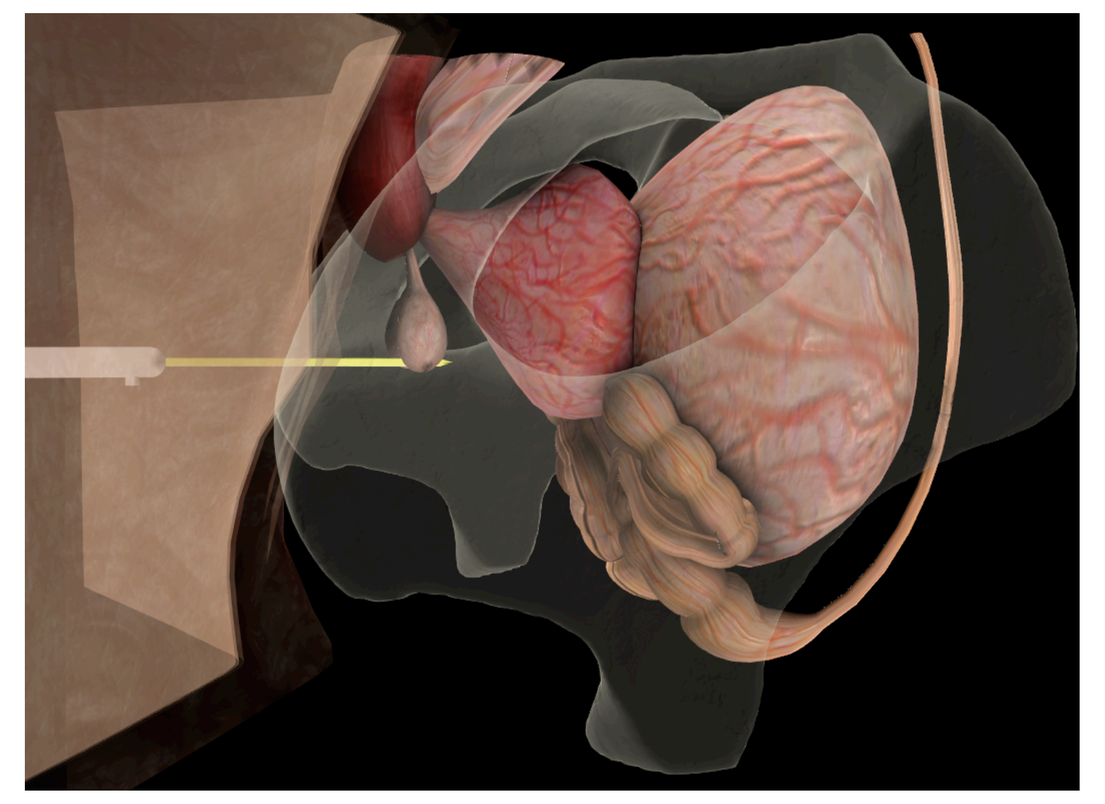
We plan to use this simulation code for surgical planning. Members of our project group have developed feedback control software that uses our simulator to maneuver a steerable needle to a target in a block of deformable tissue [Hauser et al., 2009]. These images show several states during needle insertion. The light blue helical paths are trajectories chosen by the planner at different instances in time. As the needle is inserted, our simulator predicts how the tissue deforms, and the planner uses this feedback to make trajectory corrections. Our future plans are to integrate into our planner the ability to avoid obstacles and to achieve coverage of a prostate gland with radioactive seeds using as few needle insertions as possible.



Simulated Scenario: Prostate Brachytherapy



Flexible Needle



Stiff Needle

Flexible Needle Demo Sequence

Timing Information:

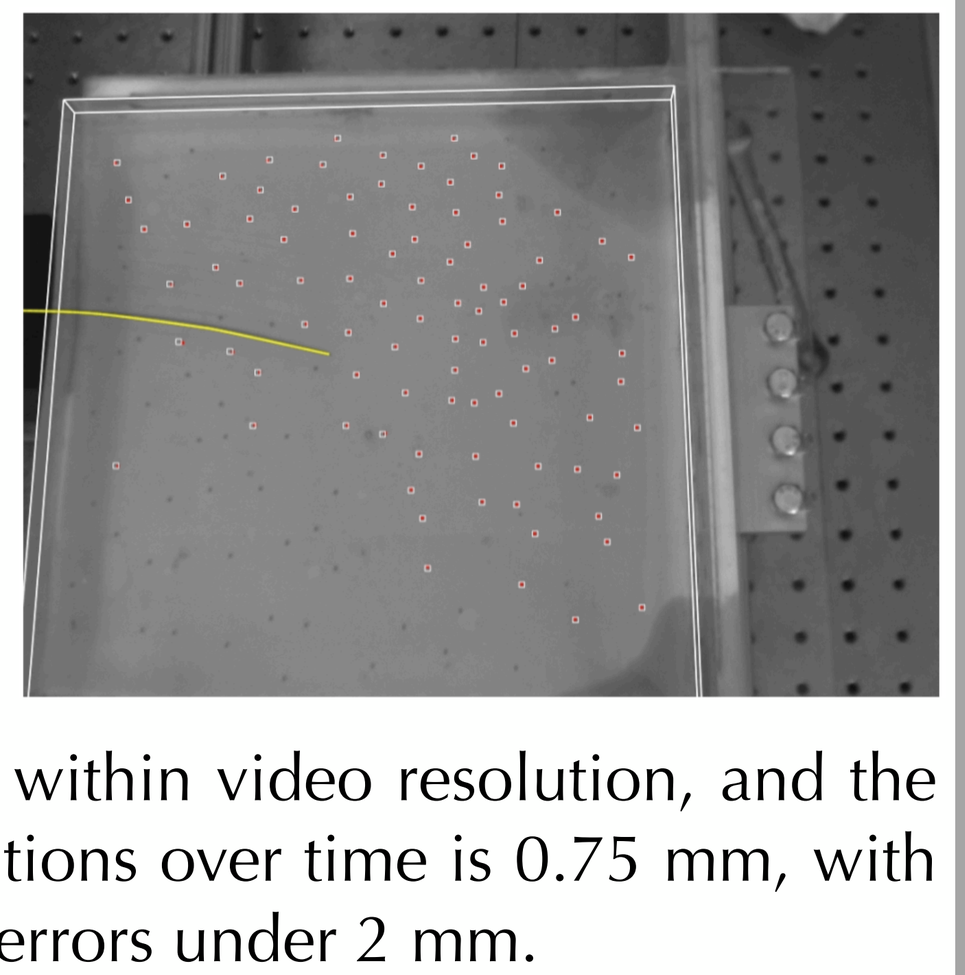
- All times in milliseconds averaged over simulation run.
- Integration time step was constant at 20 ms.
- Measurements taken on PC with Intel Xeon x8 3GHz, 16GB RAM, and Nvidia Quadro FX 5800.
- The number of tetrahedra/vertices for prostate mesh and tissue phantoms are 13,375/2,763 and 2,280/672, respectively.

Name	#	Total	LCP	Tissue	Needle	Remesh
ProsFB	1	130.9	108.8	13.4	1.3	0.5
ProsFB	2	77.7	62.3	7.7	1.6	0.5
ProsFB	4	56.6	44.8	3.9	1.4	0.7
ProsFB	7	38.5	28.3	2.2	1.4	0.3
ProsSS	7	38.2	28.7	2.1	1.1	0.5
1bend	7	22.8	13.2	1.1	0.4	0.9
2bends	7	33.0	23.6	1.1	0.5	0.5

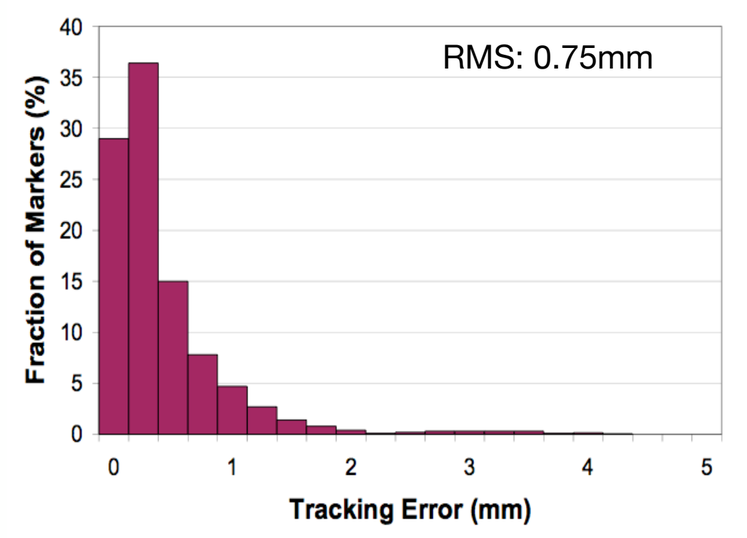
- Min/avg/max number of trials required for the LCP solver and number of CG iterations for each linear solution for examples running on seven threads

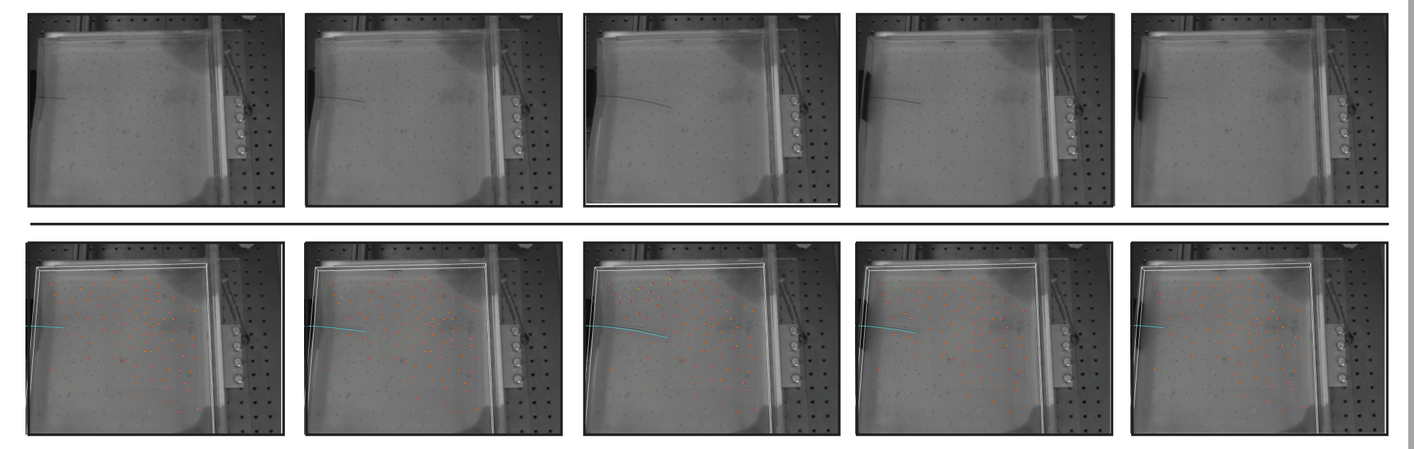
Name	LCP	CG
ProsFB	1 / 1.41 / 10	18 / 205 / 652
ProsSS	1 / 1.23 / 4	12 / 225 / 536
1bend	1 / 1.17 / 4	46 / 234 / 343
2bends	1 / 1.04 / 3	54 / 413 / 686

Accuracy Assessment



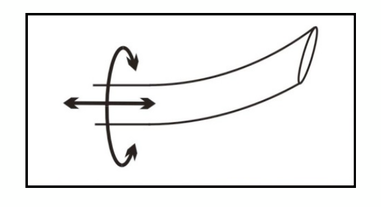
To evaluate the accuracy of the simulation we compare against experiments in which flexible, nitinol needles of diameter 0.83 mm were robotically inserted into a 27.1×26.5×3.9 cm gelatin tissue phantom. Video showing the needle and motion of fiducial markers was recorded. We then simulated the same configuration and compared the recorded and simulated markers. The needle trajectories match to within video resolution, and the root-mean-squared error of the marker positions over time is 0.75 mm, with 88.3% of errors under 1 mm and 97.8% of errors under 2 mm.

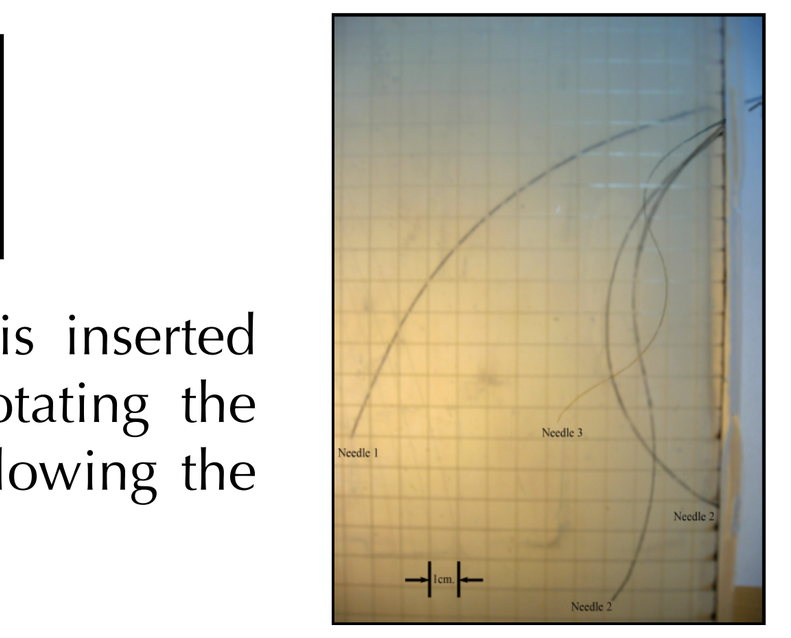




Bevel Tip Needle

A motivation for modeling needle elasticity is a new class of flexible, steerable needles recently developed in collaboration between researchers at U.C. Berkeley and Johns Hop-kins University [Webster III et al., 2005; Webster III et al., 2006]. These bevel-tip steerable needles have a flexible shaft that curves as it penetrates soft tissue, due to asymmetric forces exerted at the needle's bevel tip. By twisting the needle as it is inserted, a physician can steer its tip around obstacles to reach clinical targets in soft tissues [Alterovitz et al., 2005; Alterovitz et al., 2007]. It is not easy to learn how to control steerable needles, and realistic training simulations will accelerate their deployment in clinical practice. Simulation will also allow the development of automated planning algorithms for robotic insertion devices.





The beveled tip exerts an asymmetric force as it is inserted causing the needle to follow a curved path. Rotating the needle's base changes the direction of curvature allowing the needle to be steered through the tissue.

Acknowledgements

- Berkeley Graphics Group the Automation Lab members for their helpful criticisms and comments
- Allison Okamura, Noah Cowan, Robert Webster III, Landon Unninayar and Vinutha Kallem at Johns Hopkins for the tissue phantom video and marker tracking code
- Judy Hoffman for marker tracking code
- James Demmel and Samuel Webb Williams for discussions about sparse matrix solvers
- Vincent Duindam of Intuitive Surgical and Jean Pouliot, Joe Hsu and Adam Cunha of UCSF for advice on physiology and prostate treatment
- **Project Funding:**
  - NSF Award CCF-0635381
  - NIH Award 1R01EB-006435-01A1
  - UC Lab Fees Research Program Grant 09-LR-01-118889-OBRJ
  - California MICRO 08-077
  - California Discovery Grant COM09S-156646
  - NVIDIA Corporation
  - Pixar Animation Studios
  - Autodesk
  - Intel Corporation
  - Sony Computer Entertainment America

Simulation Details

We model tissue elasticity with constitutive equations discretized over a tetrahedral mesh by a finite element method. The needle has a small diameter, so we model it as a 1D elastic rod, following. We denote the tetrahedral tissue mesh by T. The needle is represented by a mesh $\widetilde{T}$ comprising a subset of the edges of T, plus additional edges that represent the portion of the needle outside the tissue. We dynamically update the tetrahedral mesh so that it always conforms to the needle.

We use a stick-slip model of the friction between the tissue and the needle shaft. Each node shared by the two meshes is in either a static or dynamic friction state. Static friction implies that the needle and tissue are moving in lockstep at the node; dynamic friction implies that they are sliding against each other. The needle tip is a special case, because when it is moving into the tissue with dynamic friction it is cutting tissue, and therefore encounters much greater resistance than accounted for by dynamic friction alone.

Throughout the simulation, we maintain for each node of T both a material coordinate (recording the geometry of the undeformed mesh) and a world coordinate (recording the deformed mesh). Let $\mathbf{u}_i^k$, $\mathbf{x}_i^k$, $\mathbf{v}_i^k$, $\mathbf{a}_i^k \in \mathbb{R}^3$ denote the material position, world position, velocity, and acceleration of the i^{th} node at time index k. We omit the node index to refer to a vector of properties for all the nodes. We omit the time index to refer to the current time. We use carets ($\hat{}$) to denote tissue properties, and tildes ($\tilde{}$) for needle properties.

Implicit Time Integration

Let n be the number of nodes in the (tissue or needle) mesh. We integrate the node positions $\mathbf{x} \in \mathbb{R}^{3n}$ and velocities $\mathbf{v} \in \mathbb{R}^{3n}$ over time with Newmark's method,

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \triangle t \mathbf{v}^k + \triangle t^2 \left(\left(\frac{1}{2} - \beta \right) \mathbf{a}^k + \beta \mathbf{a}^{k+1} \right), \quad (1)$$

$$\mathbf{v}^{k+1} = \mathbf{v}^k + \triangle t \left((1-\gamma) \mathbf{a}^k + \gamma \mathbf{a}^{k+1} \right), \quad (2)$$

where $\triangle t$ is the time step, $0 \leq \beta \leq 0.5$, and $0 \leq \gamma \leq 1$. (All our simulation results use $\beta = 0.25$ and $\gamma = 0.5$, equivalent to integration by the trapezoid rule.) We obtain the accelerations $\mathbf{a}^{k+1} \in \mathbb{R}^{3n}$ by solving

$$\mathbf{F}(\mathbf{x}^{k+1}, \mathbf{v}^{k+1}) = \mathbf{M}\mathbf{a}^{k+1}, \quad (3)$$

where $\mathbf{M} \in \mathbb{R}^{3n \times 3n}$ is the mass matrix and $\mathbf{F}(\cdot) \in \mathbb{R}^{3n}$ is the sum of all internal forces such as stiffness and damping forces (discussed in the next section) and external forces such as gravity. Because Equation (3) is nonlinear, we linearize it with one Newton–Raphson iteration; *i.e.* by solving

$$\mathbf{F}(\mathbf{x}^k, \mathbf{v}^k) + \frac{\partial \mathbf{F}}{\partial \mathbf{x}}(\mathbf{x}^{k+1} - \mathbf{x}^k) + \frac{\partial \mathbf{F}}{\partial \mathbf{v}}(\mathbf{v}^{k+1} - \mathbf{v}^k) \approx \mathbf{M}\mathbf{a}^{k+1}, \quad (4)$$

where $\partial \mathbf{F}/\partial \mathbf{x}$, $\partial \mathbf{F}/\partial \mathbf{v} \in \mathbb{R}^{3n \times 3n}$ are the Jacobian matrices of force with respect to position and velocity, evaluated at $(\mathbf{x}^k, \mathbf{v}^k)$.

Ignoring for now the coupling between needle and tissue, we substitute (1) and (2) into (4) to obtain sparse linear systems

$$\hat{\mathbf{A}}\hat{\mathbf{a}}^{k+1} = \hat{\mathbf{b}}, \quad (5)$$

$$\tilde{\mathbf{A}}\tilde{\mathbf{a}}^{*k+1} = \tilde{\mathbf{b}} \quad (6)$$

for the tissue and needle nodes' accelerations, respectively. The asterisk indicates that $\tilde{\mathbf{a}}^{*k+1}$ is a temporary quantity, for reasons explained in Section . Having solved for $\hat{\mathbf{a}}^{k+1}$ and $\tilde{\mathbf{a}}^{*k+1}$, we obtain $\hat{\mathbf{x}}^{k+1}$, $\hat{\mathbf{v}}^{k+1}$, $\tilde{\mathbf{x}}^{*k+1}$, and $\tilde{\mathbf{v}}^{*k+1}$ from Equations (1) and (2).

The sparsity of $\hat{\mathbf{A}}$ is unstructured, whereas $\tilde{\mathbf{A}}$ has bandwidth 5 (measured in 3×3 blocks) because each needle node has nonzero entries for the two nodes before and after it on the needle. We assemble both matrices and store $\hat{\mathbf{A}}$ with a block compressed sparse row format and $\tilde{\mathbf{A}}$ with a block banded format.

Needle-Tissue Coupling and Cutting

Each node i of $\widetilde{T}$ has a friction state s_i which is one of FREE, STATIC, DYNAMIC−, or DYNAMIC+. The sign of dynamic friction indicates the direction in which the needle is sliding along the tissue at that node. The FREE nodes are not in the tissue. The other nodes, which are shared by the tissue mesh T and the needle mesh $\widetilde{T}$, are called *coupling nodes*. Assume that each coupling node has the same index in both meshes.

If we know every state s_i, we can solve the coupled equations with Lagrange multipliers, introducing for each coupling node i an additional variable $\mathbf{c}_i \in \mathbb{R}^3$, the constraint force required to satisfy the stick-slip constraint. This force acts upon tissue and needle nodes i in equal magnitude but opposite directions. To accommodate dynamic friction and sliding of the needle, we express $\mathbf{c}_i$ in a local coordinate system of the needle, in which the first axis is the unit vector $\mathbf{t}_i$ that is tangential to the needle at node i. Let $\mathbf{R}_i \in \mathbb{R}^{3\times3}$ be the rotation matrix that transforms from local coordinates to world coordinates, so that $\mathbf{R}_i\mathbf{c}_i$ is the constraint force at the coupling node i in world coordinates. For a DYNAMIC node, we set the first column of $\mathbf{R}_i$ to zero, thus ignoring the tangential constraint force in direction $\mathbf{t}_i$.

The coupled system is

$$\begin{bmatrix} \hat{\mathbf{A}} & 0 & \hat{\mathbf{W}}\mathbf{R} \\ 0 & \tilde{\mathbf{A}} & -\tilde{\mathbf{W}}\mathbf{R} \\ (\hat{\mathbf{W}}\mathbf{R})^\mathsf{T} & -(\tilde{\mathbf{W}}\mathbf{R})^\mathsf{T} & \mathbf{Z} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{a}}^{k+1} \\ \tilde{\mathbf{a}}^{*k+1} \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{b}} + \hat{\mathbf{W}}\mathbf{R}\mathbf{d} \\ \tilde{\mathbf{b}} - \tilde{\mathbf{W}}\mathbf{R}\mathbf{d} \\ \mathbf{e} \end{bmatrix}. \quad (7)$$

The first two rows are Equations (5) and (6) augmented with the constraint forces. Here, $\mathbf{d}_i$ is $[0,0,0]^\mathsf{T}$ for a STATIC node and $[s_i f_i, 0, 0]^\mathsf{T}$ for a DYNAMIC node, s_i is 1 for DYNAMIC+ or -1 for DYNAMIC−, f_i is the magnitude of dynamic friction (possibly including a cutting force f_{cut} at the needle tip), and $\hat{\mathbf{W}}$ and $\tilde{\mathbf{W}}$ are 0–1 matrices that map coupling nodes to the tissue nodes and needle nodes, respectively. Thus, for a dynamic node i, the unknown tangential component of the constraint force $\mathbf{c}_i$ on the left-hand side is supplanted by the known friction $\pm f_i$ on the right.

The third row of Equation (7) constrains the coupling nodes to have the same positions in the tissue and needle meshes, except that a node in a dynamic friction state permits the needle to slide tangentially relative to the tissue. If s_i is STATIC, we constrain $\hat{\mathbf{x}}_i$ and $\tilde{\mathbf{x}}_i$ to be identical. If it is DYNAMIC, we constrain them to agree in the directions orthogonal to $\mathbf{t}_i$. Thus, $\mathbf{Z}$ is a diagonal matrix in which a diagonal entry is 1 for the tangential component of a DYNAMIC node and 0 otherwise, and $\mathbf{e}$ is the right-hand side of the equation found by substituting Equation (1) into

$$(\hat{\mathbf{W}}\mathbf{R})^\mathsf{T}(\hat{\mathbf{x}}^{k+1} - \hat{\mathbf{x}}^k) - (\tilde{\mathbf{W}}\mathbf{R})^\mathsf{T}(\tilde{\mathbf{x}}^{*k+1} - \tilde{\mathbf{x}}^{*k}) = \mathbf{0} \quad (8)$$

and moving the terms that include $\hat{\mathbf{a}}^{k+1}$ and $\tilde{\mathbf{a}}^{*k+1}$ to the left-hand side. Note that the columns having a 1 on $\mathbf{Z}$'s diagonal are the same columns of $\mathbf{R}$ that we set to zero.

An advantage of our formulation (7) is that we can update it quickly if the friction states s_i change. The tricky part of stick-slip friction is that the states s_i are not known in advance (except the FREE ones). We must guess them, then guess again if we are wrong. We have guessed right if they satisfy the following constraints. For a STATIC node i except the needle tip,

$$-f_i \leq [1\ 0\ 0]^\mathsf{T} \cdot \mathbf{c}_i \leq f_i, \quad (9)$$

where f_i is the static friction threshold, which experimentally is the same as the dynamic friction magnitude. For a STATIC needle tip i,

$$-(f_i + f_{\text{cut}}) \leq [1\ 0\ 0]^\mathsf{T} \cdot \mathbf{c}_i \leq f_i. \quad (10)$$

For a DYNAMIC node i (needle tip or not),

$$s_i \mathbf{t}_i \cdot \left((\hat{\mathbf{x}}_i^{k+1} - \hat{\mathbf{x}}_i^k) - (\tilde{\mathbf{x}}_i^{*k+1} - \tilde{\mathbf{x}}_i^{*k}) \right) \geq 0; \quad (11)$$

that is, the relative tangential movement between tissue and needle DYNAMIC coupling nodes must not change direction.

The equations and constraints together form a linear complementarity problem (LCP). Given q coupling nodes, there are 3^q possible settings of the friction states. The LCP has the potential to take exponential running time. Each wrong guess requires us to solve the linear system again, so even a moderate number of wrong guesses can kill real-time performance. Fortunately, the system has temporal coherence, and a good guess is to take the friction states from the previous time step. If these are wrong, we make local changes (driven by the constraints that are not satisfied) and usually find the correct states within a few trials.

Remeshing and Reparameterization

After a simulated time step, some of the DYNAMIC coupling nodes of T and $\widetilde{T}$ are no longer coincident. However, our coupling method requires coupling nodes to have the same positions in both meshes. Thus, we dynamically adapt the meshes after each time step. Our mesh adaptation consists of two steps: needle tip remeshing and needle reparameterization.

Needle Tip Remeshing

The goals of needle tip remeshing are to make T conform to the needle, to have tetrahedra of as high quality as possible, and to do so quickly. We remesh in material space by applying one of the candidate operations: node snap, edge split, face split, and tetrahedron split.

Two ideas govern our remeshing algorithm. First, we choose among candidate operations by directly measuring the quality of the tetrahedra that would be created by each operation, and selecting the operation that maximizes the quality of the worst tetrahedron. Second, we maintain a stack of all the operations that have changed the mesh topology (*i.e.* all operations except the node snap), and we consider undoing the most recent operation before applying a new one. The stack is particularly important when the needle is retracted; our procedure is designed so that once the needle is fully withdrawn from the body, the tissue mesh will have returned to its original topology. We thereby prevent the accumulation of mesh quality degradation when the needle is inserted and withdrawn multiple times. Even when the needle is being inserted, the ability to undo the previous operation and replace it with a new one often offers better mesh quality.

For each candidate operation, we evaluate each new tetrahedron with a quality measure equal to its signed volume divided by the cube of its root-mean-squared edge length. This measure is zero for a degenerate tetrahedron, and maximized by an equilateral tetrahedron. We have found this measure to be both a good reflection of a tetrahedron's fitness for finite element simulation and amenable to numerical optimization. Tetrahedron quality is always computed from material (not world) coordinates. Let R be the mesh region comprising the union of all tetrahedra that can be deleted or changed by the candidate operations. The *quality vector* of the tetrahedra in R is a list of the tetrahedron qualities, sorted from worst to best. Our algorithm chooses the operation that lexicographically maximizes that quality vector (that is, it maximizes the worst tetrahedron, breaking ties by maximizing the second-worst, then the third-worst, etc.).

If the needle is penetrating tissue, we generate a set of candidate operations as follows. Let $\mathbf{u}_{\text{new}}$ be the new position of the needle tip in material space. (We obtain $\mathbf{u}_{\text{new}}$ by barycentric interpolation from the needle tip position in world space, and we obtain a tissue velocity and acceleration for it the same way.) We consider fifteen *standard operations* that transform the tetrahedron that contains $\mathbf{u}_{\text{new}}$: four node snaps, six edge splits, four face splits, and one tetrahedron split. Each operation places the new node or snapped node at $\mathbf{u}_{\text{new}}$. We also consider composite operations that first undo the operation atop the stack (if that operation created the needle tip node), then apply a new standard operation.

Candidate operations that would change the mesh boundary are discarded, except in the moment where the needle first penetrates the skin. Operations that fail to properly connect the needle nodes are also discarded. Let $\mathbf{u}_1$ and $\mathbf{u}_2$ be the old positions in material space of the needle tip and the needle node adjoining the tip, respectively. The needle nodes remain properly connected by the operations that snap $\mathbf{u}_1$ to $\mathbf{u}_{\text{new}}$, that create an edge connecting $\mathbf{u}_1$ to $\mathbf{u}_{\text{new}}$, or that delete $\mathbf{u}_1$ and create an edge connecting $\mathbf{u}_2$ to $\mathbf{u}_{\text{new}}$. (Undoing the top stack operation entails deleting $\mathbf{u}_1$ from both T and $\widetilde{T}$.)

Because the needle moves only a small distance during a time step, $\mathbf{u}_{\text{new}}$ tends to be close to $\mathbf{u}_1$ or $\mathbf{u}_2$, often producing a short edge that compromises the mesh quality. We avoid this pitfall by moving $\mathbf{u}_1$ to the optimal position on the segment $\mathbf{u}_{\text{new}}\mathbf{u}_2$, or (if the operation deletes $\mathbf{u}_1$) by moving $\mathbf{u}_2$ to the optimal position on the segment $\mathbf{u}_{\text{new}}\mathbf{u}_3$. The "optimal" position is the one that maximizes the minimum quality among the tetrahedra that adjoin the moved node. This repositioning is part of the candidate operation, and is taken into account when the best operation is chosen.

Needle retraction uses somewhat different candidate operations. The *only* operation we consider that does not delete the needle tip $\mathbf{u}_1$ from the needle mesh is a node snap that moves $\mathbf{u}_1$ to $\mathbf{u}_{\text{new}}$. The other candidate operations delete $\mathbf{u}_1$ as follows. If $\mathbf{u}_1$ was created by the operation on top of the stack, then the stack is popped and that operation is undone, deleting $\mathbf{u}_1$ from both T and $\widetilde{T}$; otherwise, $\mathbf{u}_1$ was placed by a node snap, in which case we delete it from the needle mesh only. In either case, one of the standard operations subsequently creates a node at $\mathbf{u}_{\text{new}}$, or snaps a node there. If $\mathbf{u}_1$ survives in the tissue mesh, it tends to be close to $\mathbf{u}_{\text{new}}$, so we subsequently optimize the position of $\mathbf{u}_1$ (but not $\mathbf{u}_2$) as part of the candidate operation. Because $\mathbf{u}_1$ no longer lies on the needle, it can move freely.

Needle Reparameterization

Our needle tip remeshing procedure ensures that the tissue mesh has a sequence of nodes and edges that corresponds to the part of the needle inside the tissue. These nodes will be the coupling nodes in the next time step, and their positions are determined by the tissue mesh—that is, $\tilde{\mathbf{x}}_i^{k+1} = \hat{\mathbf{x}}_i^{k+1}$. The FREE nodes' positions are determined by the solution to Equation (7); that is, $\tilde{\mathbf{x}}_i^{k+1} = \tilde{\mathbf{x}}_i^{*k+1}$.

Needle sliding at a DYNAMIC node implies that it no longer represents the same point on the needle as it did before the time step. Moreover, remeshing creates and deletes nodes, and the simulation does not keep the needle perfectly inextensible, so the needle length varies slightly. Therefore, we reparametrize the needle and interpolate physical quantities from before to after the time step.

We parametrize each node i existing before the time step by its distance d_i^* from the base of the needle, and each node j existing after by its distance d_j from the base after the time step. (We compute these distances as sums of line segment lengths, but one could use the arc length of an interpolating curve instead.) Because the needle is not perfectly inextensible, we scale all the distances after the time step so the values of d^* and d at the needle tip are equal.

To compute the acceleration $\tilde{\mathbf{a}}_j$ at node j after a time step, we build an interpolating function $g(\cdot)$ such that $g(d_i^*) = \tilde{\mathbf{a}}_i^*$, where the right-hand side comes from the solution of Equation (7), then set $\tilde{\mathbf{a}}_j = g(d_j)$.

A needle edge outside the tissue can become too short or too long in two places: where the needle exits the guide sleeve, and where the needle enters the tissue. Thus, we merge nodes that are too close together (shorter than half the minimum initial edge length), moving a FREE node onto the node on the surface of the tissue or the end of the sleeve; and we split edges that are too long (over four times the maximum initial edge length), all before reparameterizing. To split an edge, we place a new node at the midpoint of an interpolating cubic curve.