

Feedback Control for Steering Needles Through 3D Deformable Tissue Using Helical Paths

Kris Hauser*, Ron Alterovitz†, Nuttapon Chentanez*, Allison Okamura‡, and Ken Goldberg*

*IEOR and EECS Departments, University of California, Berkeley, {hauser, nchentan, goldberg}@berkeley.edu

†Computer Science Department, University of North Carolina, Chapel Hill, ron@cs.unc.edu

‡Mechanical Engineering Department, Johns Hopkins University, aokamura@jhu.edu

Abstract—Bevel-tip steerable needles are a promising new technology for improving accuracy and accessibility in minimally invasive medical procedures. As yet, 3D needle steering has not been demonstrated in the presence of tissue deformation and uncertainty, despite the application of progressively more sophisticated planning algorithms. This paper presents a feedback controller that steers a needle along 3D helical paths, and varies the helix radius to correct for perturbations. It achieves high accuracy for targets sufficiently far from the needle insertion point; this is counterintuitive because the system is highly underactuated and not locally controllable. The controller uses a model predictive control framework that chooses a needle twist rate such that the predicted helical trajectory minimizes the distance to the target. Fast branch and bound techniques enable execution at kilohertz rates on a 2GHz PC. We evaluate the controller under a variety of simulated perturbations, including imaging noise, needle deflections, and curvature estimation errors. We also test the controller in a 3D finite element simulator that incorporates deformation in the tissue as well as the needle. In deformable tissue examples, the controller reduced targeting error by up to 88% compared to open-loop execution.

I. INTRODUCTION

Needles are used in medicine for a wide range of diagnostic and therapy delivery procedures. Many applications, such as biopsies and prostate brachytherapy, require the needle tip to be positioned accurately at a target in the tissue. Such procedures require a great deal of clinician skill and/or trial and error, and can be difficult even under image guidance. Errors are introduced when positioning the needle for insertion, and the needle can be deflected from its intended path by tissue inhomogeneities. Furthermore, the target may shift due to tissue deformations caused by the needle or patient motion.

Asymmetric-tip flexible needles are a new class of needles, developed in collaboration between Johns Hopkins and U.C. Berkeley, that can steer through soft tissue [24]. As it is inserted, the needle travels along a curved path due to the asymmetric cutting forces generated by either a bevel cutting tip or a pre-bent kink. The arc direction can be controlled by twisting the base of the needle, which allows the needle to travel in circular arcs [24] (by holding twist constant) and helical trajectories (by simultaneously twisting and inserting the base at constant velocity) [12]. In principle, these inputs can be used to correct for needle placement errors and deflections during insertion. They may also steer around obstacles such as bones or sensitive organs to reach targets that are inaccessible to standard rigid needles.

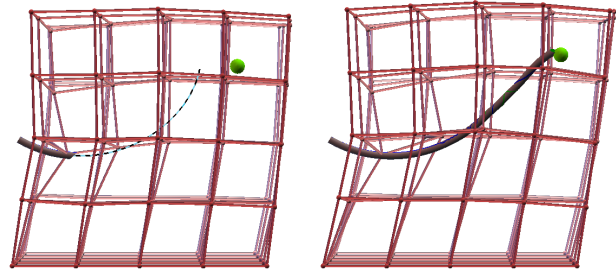


Fig. 1. Needle steering in a soft block of tissue. Left: the needle pierces the surface. Dotted curve is the predicted helical path of the needle. Right: feedback control corrects for errors due to deformation. More examples can be found at <http://automation.berkeley.edu/projects/needlesteering/>.

A variety of techniques have been used for 3D path planning of steerable needle trajectories [12, 13, 20, 26]. But these techniques assume an idealized rigid tissue model; in practice, the needle will deviate from the planned path due to deformation effects (of both tissue and needle), tissue inhomogeneities, and patient variability. Planners that address uncertainty and deformations are currently limited to 2D needle steering, and perform extensive precomputations [1, 2].

Our work breaks with the trend toward increasingly precise modeling and more sophisticated (and computationally expensive) planning techniques. We present a closed-loop, model predictive controller that steers the needle along helical paths to reach a desired target position. Perturbations are sensed using imaging feedback, and the controller reacts by adjusting the radius and heading of the helix to pass as close as possible to the target. A fast branch and bound technique is used to find the closest point on a helix, which enables trajectory corrections to be computed at kilohertz rates.

We evaluate the controller on a wide variety of simulation experiments in both rigid tissue and in a finite element deformable tissue simulator (Figure 1). In light of the fact that steerable needles are underactuated and non-controllable, our results are surprising. Assuming no perturbations, the controller achieves high accuracy (less than 1% of the needle's radius of curvature r) for all targets sufficiently far from the insertion point (approximately twice r). It attains reasonable accuracy (less than 5% of r) under a variety of simulated Markovian and non-Markovian perturbations, including imaging noise, needle deflection, and curvature estimation errors. In

deformable tissue, the controller can compensate for deformation effects to achieve much higher accuracy than open-loop execution. In the example of Figure 1, the controller reduces targeting error by 88%.

II. RELATED WORK

Several mechanisms for needle steering have been proposed. Symmetric-tip flexible needles can be steered by translation and rotation at the needle base [10, 14]. Bevel-tip flexible needles achieve steering with a tip that produces asymmetric cutting forces, causing them to bend when inserted into soft tissue [24, 25]. Recent experiments with pre-bent needle tips have achieved up to a 6.1cm radius of curvature [22]. New needle designs are expected to further decrease this radius.

Bevel-tip and pre-bent needles are controlled by two degrees of freedom: insertion and twisting at the needle base. When restricted to a plane, the needle moves like a Dubins car that can only turn left or right. In 3D, the needle moves in approximately helical trajectories. The radius of curvature of the path can be adjusted during insertion using duty cycling, enabling the needle to be driven straight forward [18].

Planning motions of 1D flexible objects has been studied in the context of redundant manipulators [6, 7, 8], ropes [23] and symmetric-tip flexible needles [10, 14]. Several researchers have developed motion planners for steerable needles in 3D tissues [12, 13, 20, 26]. These methods have all assumed that tissue is rigid. Duindam et al. derived a closed-form inverse kinematics solution for reaching a desired position orientation in 3D [13]. The solution uses up to three stop-and-turn motions in which the needle, between turns, follows a circular arc. Inverse kinematics solutions for helical paths have not been found, and this is challenging because of the mixed trigonometric and polynomial terms required to specify these paths. Planning with helices in 3D has been achieved using numerical optimization [12], an error diffusion technique [20], and a sampling-based planner [26].

Needle and tissue deformation will deflect the needle from an idealized trajectory. In such cases a controller may be used to keep a needle on its intended path. Kalleem and Cowan developed controllers to stabilize a needle to a given plane [17]. For 2D, Alterovitz et al. developed an optimization-based planner that predicts tissue deformations using finite element simulation [2], although this assumes no uncertainty in the motion model or tissue properties. Alterovitz et al. developed a 2D motion planner for needle steering with obstacles and Markov motion uncertainty [1], but did not model large-scale tissue deformations or curvature estimation errors.

III. PROBLEM STATEMENT AND MODELING ASSUMPTIONS

A. Problem Statement

We wish to place the needle tip at a certain goal location \mathbf{p}_g in the tissue. The desired needle orientation is not specified. We assume we have access to an imaging system that periodically provides an estimate of the world-space coordinate of \mathbf{p}_g as it moves due to tissue deformations, as well as the position, heading, and bevel direction of the needle tip. We

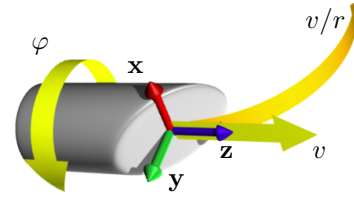


Fig. 2. The coordinate frame attached to the needle tip. Figure adapted with permission from [13].

also assume a given workspace $\mathcal{W} \subseteq \mathbb{R}^3$ in which the needle is permitted to travel. In this work, \mathcal{W} is taken to be either all of \mathbb{R}^3 or a bounding box.

We assume an insertion velocity is chosen for the duration of the procedure (on the order of 1cm/s). The controller outputs a twist rate to be applied at the needle base. We impose a maximum twist rate φ_{max} (specified in rad/m) because fast spinning may damage the tissue. A robot or physician will simultaneously insert and twist for Δt seconds at the twist rate computed by the controller. The controller then repeats.

B. Rigid Tissue Motion Model

Though the controller is designed to operate in deformable tissue, it uses a purely kinematic rigid tissue motion model [24] for computational efficiency. This model assumes the dynamics of the needle are fully determined at the needle tip, such that the tip travels with constant curvature $1/r$ in the bevel direction. Thus, if inserted without twisting, the needle trajectory is a circle with radius of curvature r . Experiments suggest that this assumption approximately holds in homogeneous materials [24]. The model also assumes that a twist at the base of the needle is transmitted directly to the needle tip, and does not deflect the tip position. This assumption neglects torsional friction along the needle shaft, which causes a “lag” before twists are felt at the tip. Control techniques have been developed to counteract these effects [21].

We place a frame $\mathbf{x}, \mathbf{y}, \mathbf{z}$ at the needle tip such that \mathbf{z} is forward and \mathbf{y} is the bevel direction (Figure 2). Inserting the needle causes the tip to move along \mathbf{z} with velocity $v(t)$ and rotate about \mathbf{x} with angular velocity $v(t)/r$. A twist rotates the tip about \mathbf{z} with angular velocity $\varphi(t)$. In screw notation, the instantaneous velocity of the frame is given by:

$$\hat{V}(t) = \begin{bmatrix} 0 & -\varphi(t) & 0 & 0 \\ \varphi(t) & 0 & -v(t)/r & 0 \\ 0 & v(t)/r & 0 & v(t) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

If $v(t) \neq 0$, then we can factor out $v(t)$ so that the needle travels at unit velocity, and the only variable to control is the twist rate $\varphi(t)$. If the twist rate is held constant, then after inserting the needle a distance d , the tip frame is transformed by the quantity $T(d) = \exp(d\hat{V})$ [12].

We model a trajectory as a sequence of screw motions with piecewise constant twist rates $\varphi(d)$ parameterized by insertion distance. Denote $\mathbf{t}_{\varphi(d)}(d)$ as the trajectory of the tip position

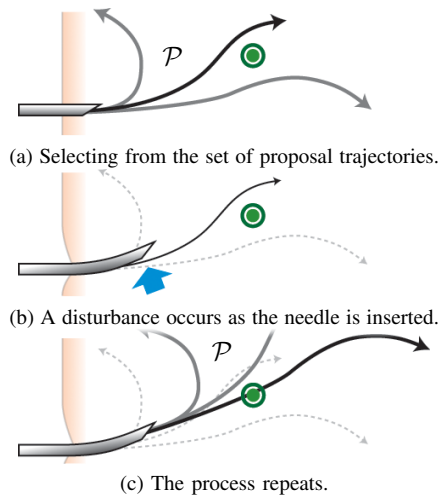


Fig. 3. Overview of the control policy. Curves emanating from the needle depict proposal trajectories. Circle depicts the target.

under twists $\varphi(d)$. We drop the subscript if the twists are unimportant.

IV. FEEDBACK CONTROLLER WITH HELICAL PATHS

A. Controller Framework

We assume a model predictive control (MPC) framework where the following steps are executed every iteration (Figure 3):

- 1) *Propose*. Generate a set \mathcal{P} of proposal trajectories that start from the current estimated needle state.
- 2) *Select*. Find the trajectory in \mathcal{P} with control $\varphi(d)$ that achieves the minimal distance to the target.
- 3) *Execute*. The needle is then inserted with twists according to $\varphi(d)$ and constant velocity for time Δt .

The process repeats until no trajectory can improve the distance to the target by more than ϵ (we set $\epsilon=0.2\%r$).

The controller meets basic stability criteria: first, it will terminate, because the predicted distance to the target must decrease by at least ϵ every iteration; second, the final error will be no larger than the distance from the target to \mathcal{P} . Convergence is difficult to prove, and depends on the selection of \mathcal{P} . Empirical results in Section VI suggest that setting \mathcal{P} to the set of constant-twist-rate helices is sufficient to reach most target locations with high accuracy, even under perturbations.

This is surprising and counterintuitive for two reasons. First, the system is not locally controllable, even if the needle were allowed to move backward [17]. Even though our controller cannot reduce errors asymptotically to zero, it often has low error upon termination. Second, traditional MPC techniques use expensive numerical optimizations at every time step to compute \mathcal{P} (e.g. receding horizon optimal control). Our controller performs well even without considering future twist rate changes, so each time step is computed efficiently.

In the remainder of this section, we derive a closed form expression for the helical trajectories followed by the needle tip when inserted with constant velocity and twist rate φ . We let the set of proposal trajectories \mathcal{P} contain all such helices

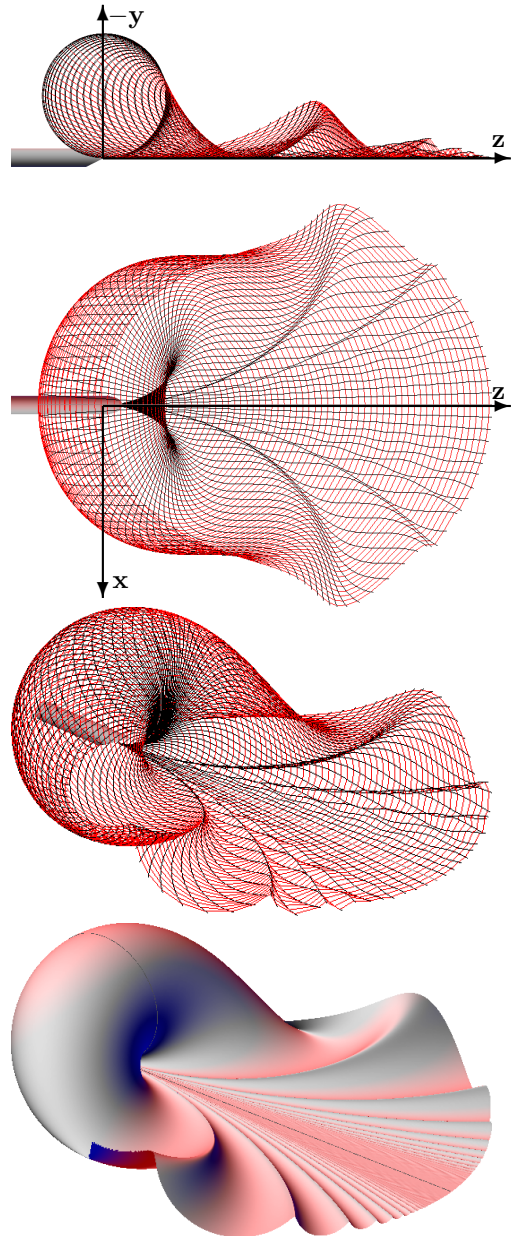


Fig. 4. Four views of the constant-twist-rate reachable set: $z, -y$ plane, z, x plane, perspective, and surface shaded. Twist rate in $(-\infty, \infty)$, insertion distance in $[0, 2\pi r]$.

for $\varphi \in [-\varphi_{max}, \varphi_{max}]$. Using a visualization of \mathcal{P} , we argue that the controller converges for sufficiently distant targets.

B. Constant-Twist-Rate Helical Paths

Let $\mathbf{t}_\varphi(d)$ be the needle tip trajectory followed under a constant twist rate φ . We wish to find the helix that describes this trajectory relative to the initial frame of the tip. A helix with radius a , slope θ , and oriented in the z axis has the arc-length parameterization

$$\mathbf{h}(d) = \begin{bmatrix} a \cos((d/a) \cos \theta) \\ a \sin((d/a) \cos \theta) \\ d \sin \theta \end{bmatrix}. \quad (1)$$

Here we derive the helix parameters a, θ and a rigid transformation A such that $A\mathbf{h}(d) = \mathbf{t}_\varphi(d)$. Denote $\mathbf{h}_x, \mathbf{h}_y, \mathbf{h}_z$, as the coordinate axes of A , and denote \mathbf{h}_o as its origin. We proceed by matching derivatives:

$$A\mathbf{h}(0) = \mathbf{t}_\varphi(0) \Rightarrow a\mathbf{h}_x + \mathbf{h}_o = [0, 0, 0]^T \quad (2)$$

$$A\mathbf{h}'(0) = \mathbf{t}'_\varphi(0) \Rightarrow \cos\theta\mathbf{h}_y + \sin\theta\mathbf{h}_z = [0, 0, 1]^T \quad (3)$$

$$A\mathbf{h}''(0) = \mathbf{t}''_\varphi(0) \Rightarrow -\cos^2\theta/a\mathbf{h}_x = [0, -1/r, 0]^T \quad (4)$$

Let $\omega = [1/r, 0, \varphi]^T$ be the angular velocity of the needle tip. The axis of rotation must be aligned with the helix axis, so $\mathbf{h}_z = \omega/|\omega|$. Let $q = |\omega| = \sqrt{1/r^2 + \varphi^2}$. Taking the dot product of both sides of (3) with \mathbf{h}_z , we get

$$\sin\theta = \varphi/q \quad \text{and} \quad \cos\theta = 1/(rq).$$

From (4) we see

$$a = r \cos^2\theta = 1/(rq^2) \quad \text{and} \quad \mathbf{h}_x = [0, 1, 0]^T.$$

Setting \mathbf{h}_y orthogonal to \mathbf{h}_x and \mathbf{h}_z , we get

$$\mathbf{h}_y = [-\varphi, 0, 1/r]^T/q.$$

Finally, from (2) we have $\mathbf{h}_o = [0, -a, 0]^T$.

In summary, letting $\tan\theta = r\varphi$, we have

$$\mathbf{t}_\varphi(d) = \begin{bmatrix} -a \sin\theta \sin((d/r) \csc\theta) + d \sin\theta \cos\theta \\ a \cos((d/r) \csc\theta) - a \\ a \cos\theta \sin((d/r) \csc\theta) + d \sin^2\theta \end{bmatrix} \quad (5)$$

Where $a = r \cos^2\theta = r/(1/r^2 + \varphi^2)$ is the helix radius, and $[\cos\theta, 0, \sin\theta]^T$ is the helix axis. The tip moves at the speed $\sin\theta = \varphi/\sqrt{1/r^2 + \varphi^2}$ with respect to the helix axis.

C. Constant-Twist-Rate Reachable Set

Let $\mathcal{R}(\varphi_{max}) = \{\mathbf{t}_\varphi(d) \mid \varphi \in [-\varphi_{max}, \varphi_{max}], d \geq 0\}$ be the set of all needle tip positions reachable under constant twist rate, given twist rate bounds $[-\varphi_{max}, \varphi_{max}]$. Figure 4 plots a portion of $\mathcal{R}(\infty)$.

$\mathcal{R}(\varphi_{max})$ is scale-invariant w.r.t. radius of curvature, in the following sense. Let $\tilde{\mathbf{t}}_{\alpha\varphi}(d)$ denote the tip trajectory with curvature α/r and constant twist rate $\alpha\varphi$. It is easily verified that $\tilde{\mathbf{t}}_{\alpha\varphi}(\alpha d) = \alpha\mathbf{t}_\varphi(d)$. Thus, $\mathcal{R}(\alpha\varphi_{max}) = \alpha\mathcal{R}(\varphi_{max})$.

This visualization helps understand why the controller should converge. The projection of $\mathcal{R}(\infty)$ on the x, z plane covers the entire $z > 0$ halfplane, and all twists produce axes of rotation that are parallel to the x, z plane. Thus, if $\mathcal{R}(\infty)$ is rotated by a full turn about any axis, it will sweep out a large portion of \mathbb{R}^3 . Most trajectories make a full turn quickly — in fact, constructing a trajectory that does not ever make a full turn is extremely difficult. For any target point in this swept volume, errors will converge to zero.

D. Boosting Accuracy with Alternating-Twist Maneuvers

For a finite maximum twist rate φ_{max} , $\mathcal{R}(\varphi_{max})$ has a gap of coverage along the z axis, shaped approximately like a wedge of angle $|\arctan r\varphi_{max}|$. To help fill in the gap, we consider the maneuver that makes a full turn of the helix with twist rate φ , and another with twist rate $-\varphi$ (Figure 5).

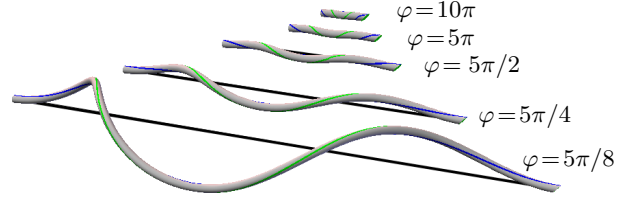


Fig. 5. Alternating-twist maneuvers for various twist rates. Units in rad/r.

After each full turn, the needle tip is displaced by $(2\pi r \sin\theta \cos^2\theta, 0, 2\pi r \sin^2\theta \cos\theta)$ where $\theta = \arctan r\varphi$. Orientation is unchanged. Executing both turns places the needle tip at $(0, 0, 4\pi r \sin^2\theta \cos\theta)$. This maneuver requires that the needle be inserted by distance $d = 4\pi r \cos\theta$.

We augment \mathcal{P} with a trajectory that contains repetitions of this maneuver, using twist rate φ_{max} . Experiments demonstrate that the augmented proposal set improves accuracy (Section VI-A).

V. FAST TRAJECTORY SELECTION

The speed of computing the proposal and selection steps limit the speed at which the needle can be inserted. If more than Δt time is spent, the needle must be halted while the computation finishes. Thus, it is critical for the closest point optimization in the selection step to be fast. In this section we present an efficient branch-and-bound technique for computing the closest point. We also present a bounding volume method for incorporating workspace constraints.

A. Branch-and-Bound Closest Point Computation

As can be seen in Figure 4, the reachable set $\mathcal{R}(\varphi_{max})$ has many valleys and ridges, so local optimization techniques easily get stuck in local minima. Instead, we use branch-and-bound optimization to ensure that the global closest point is found.

We seek to minimize the function $f(\varphi, d) = \|\mathbf{t}_\varphi(d) - \mathbf{p}\|$ over the space $\mathcal{S} = \{(\varphi, d) \mid |\varphi| \leq \varphi_{max}, d > 0\}$. Here \mathbf{p} denotes the coordinates of \mathbf{p}_g relative to the needle tip frame. We use an auxiliary function $f_L(R)$, described in Section V-C, that computes a lower bound on the function value attained in a region $R \subseteq \mathcal{S}$. A search tree is then built by recursively splitting the space into subregions which may contain the optimum. When a subregion is split, we also test the value of f at its midpoint. We maintain the helix φ^* and insertion distance d^* that give the minimum value of f seen thus far. If $f_L(R)$ is found to be larger than \underline{f} , then subregion R does not contain the optimum and can be safely pruned. The process continues until \underline{f} achieves an ϵ tolerance.

In subsequent iterations of the control loop, the closest point does not change very much, so we can exploit this temporal coherence to help prune many intervals quickly. We keep the optimal twist φ^* and insertion distance d^* from the previous iteration, and initialize $\underline{f} = f(\varphi^*, d^*)$. In our experiments this reduces overall running time by about a factor of four.

The efficiency of branch and bound depends greatly on the tightness of the lower bound. Interval analysis could be used

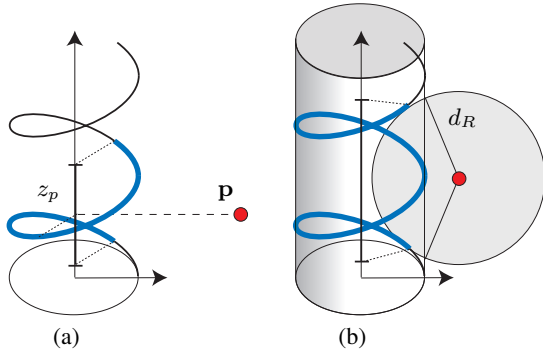


Fig. 6. Two initial intervals for finding the closest point from \mathbf{p} to a helix. (a) The interval containing half a turn of the helix, centered about the $z = z_p$ plane. (b) The interval containing the z -range of the intersection between a sphere of radius d_R centered around \mathbf{p} , and a cylinder surrounding the helix.

to automatically compute the bounds [16], but the bounds can be extremely poor, depending on how (5) is formulated. In our experiments, the running time of a standard interval analysis implementation ranged from milliseconds to several seconds, and was sensitive to the location query point.

Our implementation is as follows. We partially discretize \mathcal{S} by selecting $n = 100$ possible values for φ . For a more uniform coverage of the reachable set given a fixed n , we discretize evenly in the $\theta = \arctan r\varphi$ coordinate. To include other proposal trajectories in the search, we divide them into helix segments, and add the segments to \mathcal{S} . Then, for each helix, we narrow \mathcal{S} further by computing a narrow initial interval of insertion distances that is guaranteed to contain the closest point. We then employ a good lower bound on the distance from a helix segment to a target point.

B. Initial intervals for helix closest point

We derive two intervals that must contain the closest point using geometrical reasoning; their intersection provides a tight initial interval for the search. Reparameterize (1) such that $u = (d/a)\cos\theta$, and let $b = a\tan\theta$. Write the helix and point \mathbf{p} in cylindrical coordinates with respect to the helix origin. Denoting cylindrical coordinates with the superscript \cdot^c , we have $\mathbf{h}^c(u) = [a, u, ub]^T$. Let $\mathbf{p}^c = [r_p, \theta_p, z_p]^T$.

First, the helix attains z -coordinate z_p at $u = z_p/b$. The closest point to \mathbf{p} must be within a half turn of this parameter; that is, $|u^* - z_p/b| \leq \pi$ (Figure 6a). The proof is by contradiction: if u^* were actually outside this interval, then a full turn around the helix could move the z coordinate closer to z_p while keeping x and y fixed. Therefore u^* is not optimal.

Second, we examine the cylinder bounding the helix. If d_R is the distance from \mathbf{p} to any point on the reachable set (say, $\mathbf{h}(z_p/b)$), then the closest point must lie within a ball of radius d_R around \mathbf{p} . Projecting the intersection of the bounding cylinder with the sphere onto the z axis, we see that the z -coordinate z^* of the closest point must satisfy $|z_p - z^*| \leq \sqrt{d_R^2 - (r_p - a)^2}$ (Figure 6b). This bound performs well when d_R is close to the actual helix-point distance.

C. Lower bounding the helix-point distance

Now we describe a lower bound f_L for $u \in [\underline{u}, \bar{u}]$ and φ given. We compute the distance from the point to a patch (in cylindrical coordinates) that contains the portion of the helix $[a, a] \times [\underline{u}, \bar{u}] \times [ub, \bar{u}b]$. Let $C(x, [a, b])$ denote the clamping function $\max(a, \min(b, x))$, and implicitly assume operations in the θ coordinate are modulo 2π . Then the closest point to this patch is given by

$$\mathbf{q}^c = \begin{bmatrix} a \\ C(\theta_p, [\underline{u}, \bar{u}]) \\ C(z_p, [ub, \bar{u}b]) \end{bmatrix} \quad (6)$$

The Cartesian distance $\|\mathbf{q} - \mathbf{p}\|$ is a lower bound.

D. Workspace Bounds

We consider workspace constraints using a simple extension to the selection step. Before computing the helix-point distance, we truncate the trajectory $\mathbf{t}(d)$ if it collides with the workspace boundary. In other words, we find the maximum D for which $\mathbf{t}(d)$ remains in \mathcal{W} . Then, the initial branch-and-bound search interval for the closest point (Section V-B) is restricted to lie in $[0, D]$.

We use an recursive bisection to find D up to tolerance ϵ . We start the recursion with interval $I = [0, \bar{d}]$, where \bar{d} is the upper bound on the insertion distance from Section V-B. We then recurse as follows. If the size of I is less than ϵ we set D to its lower bound, as we cannot rule out the possibility of collision in I . Otherwise, we compute the axis-aligned bounding box containing the portion of $\mathbf{t}(d)$ for all $d \in I$ using interval computations. If the box lies within \mathcal{W} , the trajectory segment lies entirely in the workspace. If not, we bisect I , and recurse on the lower half. If no collision is reported in the lower half, then we recurse on the upper half.

E. Running Time

Since no interval will have size less than ϵ , the number of intervals examined by the branch and bound algorithm is no more than $\lceil 2L/\epsilon \rceil$, where L is the total length of all initial intervals. Each of the n initial intervals has length at most $2\pi r$, so the number of intervals is bounded by $\lceil 4\pi r n / \epsilon \rceil$. In practice, this bound is extremely loose; the number of examined intervals rarely exceeds a few hundred.

We performed timing experiments on a 2GHz laptop PC. With an unbounded workspace, the average controller iteration was computed in 0.11ms, with standard deviation of 1.3ms. On rare occasions, particularly as the target nears the needle tip, it is much slower, at one point taking 63ms. More computation is required for bounded workspaces with obstacles. In a box domain with 10 spherical obstacles, each iteration averaged 0.41ms, with standard deviation 2.52ms.

VI. SIMULATIONS IN RIGID TISSUE

This section performs a set of simulation experiments to evaluate the accuracy of the controller, as a function of the target position. We simulate the needle using the rigid tissue motion model, first under varying simulation parameters, and

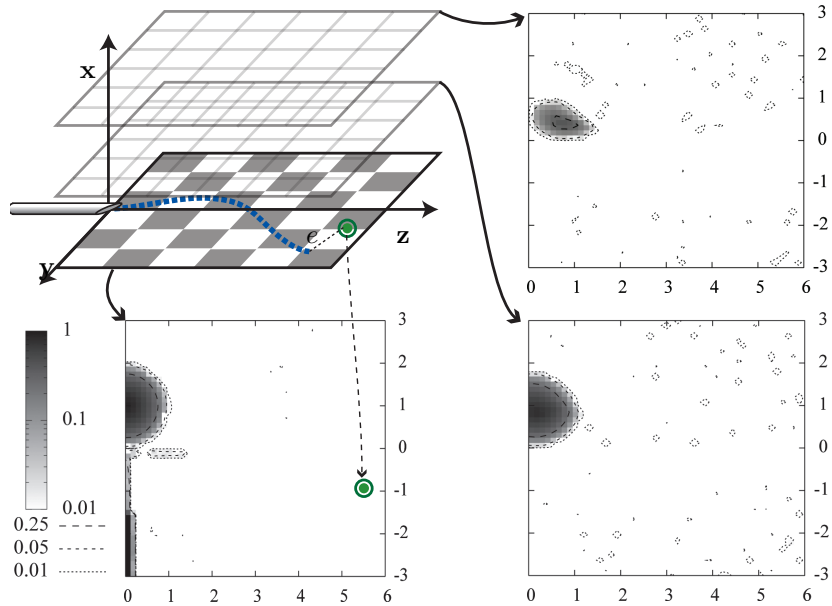


Fig. 7. Controller accuracy in rigid tissue as a function of target location. For a given target, the reported accuracy is the final error e achieved by the controller when the needle is inserted at $(0, 0, 0)$. Targets are varied over a 50×50 grid in the region $[0r, 6r] \times [-3r, 3r]$ in the $z, -y$ plane, and in three slices along the x -axis at $x=0r, 1r$, and $2r$. Darker shading indicates higher error. Axes are labeled in multiples of r .

then under a variety of perturbations. We define accuracy as the final distance from the needle tip to the target when the controller terminates.

A. Accuracy Without Perturbations

For comparison, we instantiate a *reference controller* that uses the following parameters: a refresh occurs every 2% of insertion distance, maximum twist rate $\varphi_{max} = 10\pi\text{rad}/r$, and the workspace \mathcal{W} is all of \mathbb{R}^3 .

Our first experiment evaluates how accuracy of the reference controller varies over the space of targets, under the rigid-tissue motion model. Figure 7 plots accuracy as the target position varies in 2D slices of space. Grids are defined parallel to the $z, -y$ plane (see Figure 7). We then exhaustively enumerate each grid cell, set the target \mathbf{p}_g to the cell center, and then run the controller from the initial configuration to completion. The final error from the needle tip to the target is recorded as the accuracy for that cell. The plots show convergence to sub-millimeter distances almost everywhere, except for a narrow strip along the $+y$ axis, and a circular spot directly above the needle tip in the $-y$ direction. As the x value deviates further from 0, the radius of this spot shrinks. Comparing to Figure 4, we see this region corresponds to the large “lobe” of the reachable set.

The next experiments evaluated accuracy when the controller still has an exact motion model, but the following control parameters are varied:

- 1) Maximum twist rate reduced to $2.5\pi\text{rad}/r$.
- 2) Without alternating-twist maneuvers (see Section IV-D).
- 3) In a bounded workspace $[-3r, 3r] \times [-3r, 3r] \times [0r, 6r]$.

For each setting of parameters, we evaluate the accuracy distribution for targets in a region R in space. We chose R

TABLE I
ACCURACY UNDER VARYING CONTROLLER PARAMETERS.

Simulation	Parameters	Accuracy (std), % r
Reference	—	0.2 (0.2)
Reduced twist rate	$\varphi_{max} = 2.5\pi$	2.9 (1.6)
No maneuver	—	1.3 (0.8)
Bounded workspace	—	1.0 (2.1)

to be the rectangle $[0, 0] \times [-3r, 3r] \times [1.5r, 6r]$, which is the rightmost $3/4$ of the $x = 0r$ plot of Figure 7. This region was chosen to exclude the circular spot where the reference controller does not converge. For each point in a 50×50 grid over R , we set the target to the point, and ran the controller from start to finish. Table I reports the average accuracy over all grid points and its standard deviation.

B. Under Perturbations and Modeling Errors

The second set of experiments introduce a variety of perturbations and modeling errors into the simulation. The perturbations are discussed in detail below. Table II summarizes the results. Accuracy is measured exactly as in Table I.

All perturbations degrade accuracy to some extent, but rarely drive the error above 10% . This suggests that the controller is stable. Noise in the target position and needle pose estimation can cause significant reductions in accuracy, but in practice a smoothing filter will help mitigate these effects. Twist lag causes high variance in accuracy, in some cases exceeding 20% error.

1) *Noise introduced into the target position estimation:* Tracking the target position may require the use of deformable image registration techniques (e.g. [3]), so target position estimates will contain errors due to imaging noise and deformation

TABLE II
ACCURACY UNDER SIMULATED PERTURBATIONS.

Simulation	Parameters	Accuracy (std), % r
Reference	—	0.2 (0.2)
Target noise	$\sigma = 2\%r$	2.8 (1.5)
Target noise	$\sigma = 4\%r$	6.0 (4.6)
Needle pose noise	$\sigma_p = 4\%r, \sigma_o = 2^\circ$	5.4 (4.3)
Needle deflections	$\sigma_p = 40\%, \sigma_o = 25^\circ/r$	3.2 (3.1)
Curvature errors	$\tilde{r} = 80\%r$	0.9 (1.2)
Curvature errors	$\tilde{r} = 120\%r$	0.7 (3.8)
Twist lag	$\eta = 6^\circ/r$	2.2 (12)

modeling errors. To simulate, we randomly perturb the target position estimate at each time step with isotropic Gaussian noise with standard deviation σ . We performed trials with $\sigma = 2\%r$ and with $\sigma = 4\%r$. The random number generator is seeded with the result of the C time procedure at the beginning of each trial.

2) *Noise introduced into the needle pose estimation:*

Needles show up clearly in medical imaging, but limited resolution (approximately 0.8mm for modern ultrasound [11]) will cause errors in estimating the tip position. The direction of the tip may be estimated by fitting a line (or curve) to the image. Since the bevel tip is too small to be seen, more sophisticated state estimation models are needed to estimate the twist about the needle shaft [17]. To simulate, we randomly perturb the needle position by isotropic Gaussian noise with standard deviation $\sigma_p = 2\%r$, and perturb the orientation by a rotation about a random axis and a random rotation angle with standard deviation $\sigma_o = 2^\circ$.

3) *Random deflections of the needle tip:* We simulate random deflections of the needle tip that accumulate over time, and can be mathematically described as a Wiener process. The standard deviation of the position and orientation deflections are respectively $\sigma_p = 0.4r$ and $\sigma_o = 5^\circ$ per each r distance the needle is inserted.

4) *Curvature Estimation Errors:* We set the actual radius of curvature to $\tilde{r} = 0.8r$ and $\tilde{r} = 1.2r$, while the estimated radius remains at r .

5) *Twist Lag:* As a needle is twisted inside tissue, needle-tissue friction causes internal torsion to build up along the length of the needle. This causes twist lag, a phenomenon where twists applied at the base are not directly commanded to the tip [21]. We model twist lag by nullifying all base twists if $|\theta_b - \theta_t| < \eta L$, where θ_b and θ_t are the base and tip angle, L is the length of needle inside the tissue, and η is a constant (with units of angle over distance). In other words, we evolve θ_b and θ_t with the equations $\dot{\theta}_b = \varphi$, and $\dot{\theta}_t = 0$ if $|\theta_b - \theta_t| < \eta L$, or φ otherwise. We set $\eta = 6^\circ/r$, which is consistent with the experimental data in [21] assuming $r=5\text{cm}$.

VII. SIMULATIONS IN DEFORMABLE TISSUE

We evaluate the convergence of our controller in a finite element (FEM) simulation of a flexible needle in deformable tissue. The simulator is presented in detail in [5]. Past work has addressed simulation of rigid needles in 2D and 3D deformable FEM models [9, 15, 19], and symmetric tip flexible needles

using a linear beam model [14]. Our simulator couples a discrete elastic rod flexible needle model with a FEM tissue simulation. The needle is modeled as a piecewise linear curve connecting a sequence of nodes. These nodes are coupled to nodes of the tissue, which is modeled as a tetrahedral mesh. Viscous and sticking friction along the needle shaft is simulated, but torsional friction is not.

We performed experiments with an $r=5\text{cm}$ needle in a 10cm cube of tissue constrained on the bottom face. The tissue mesh is discretized at 1.25cm intervals, and contains 729 nodes and 3072 tetrahedra. Tissue constitutive properties were set to approximately those of prostate tissue (e.g., elastic modulus 60kPa) [27]. The friction coefficient between the tissue and needle was set to 0.1 and the minimum cutting force was set to 1N. We tuned needle stiffness to achieve a turning radius of approximately 5cm ($\pm 0.3\text{cm}$) on a small number of experiments, where we varied the insertion point and angle. The tissue cube is used as the workspace \mathcal{W} .

A single trial is shown in Figure 8. The needle tip, initially at (0cm,5cm,5cm) and pointing horizontally, reaches the target (8cm,2cm,8cm) with 0.88mm accuracy. Open-loop execution of a trajectory planned in rigid tissue achieved 5.9mm error, so the controller’s improvement is 85%. Figure 9.a plots accuracy for targets in the x, z slice at $y = 5\text{cm}$, showing good agreement with accuracy in rigid tissue (Figure 9.b).

We also tested our controller under exaggerated deformation. In Figure 1 we increased the cutting force, friction coefficient, and viscous damping by factor of 10. The tip entered the tissue at a downward angle of 30° . The final error achieved by open loop execution is 13.5mm, while the controller achieved 1.59mm, an 88% improvement.

VIII. CONCLUSION

We presented a closed-loop control policy for steering bevel-tip flexible needles in deformable tissue. It rapidly iterates a single rule: execute the helical trajectory that minimizes the distance to the target. Fast branch and bound techniques that enable the controller to run in real-time on standard PC hardware. We evaluated the accuracy of the controller in rigid-tissue simulations under simulated disturbances as well as simulations of 3D finite element deformable tissue. The controller reaches relatively distant targets with high accuracy, which is surprising because steerable needles have nonholonomic and non-controllable dynamics.

A primary goal for future work is integration and evaluation on robotic hardware and tissue phantoms. We also plan to address obstacles in the workspace. Preliminary work suggests that the presented controller can already avoid simple obstacles by excluding them from the workspace. Complex workspaces might be addressed by decomposition into convex regions, or more sophisticated planners to generate obstacle-avoiding proposal trajectories. We will also seek to understand the effects of different proposal trajectories, which may enable better workspace coverage, or enable reaching targets with orientation specified. Finally, traditional asymptotic stability criteria (e.g., [4]) do not apply to our system, so we are

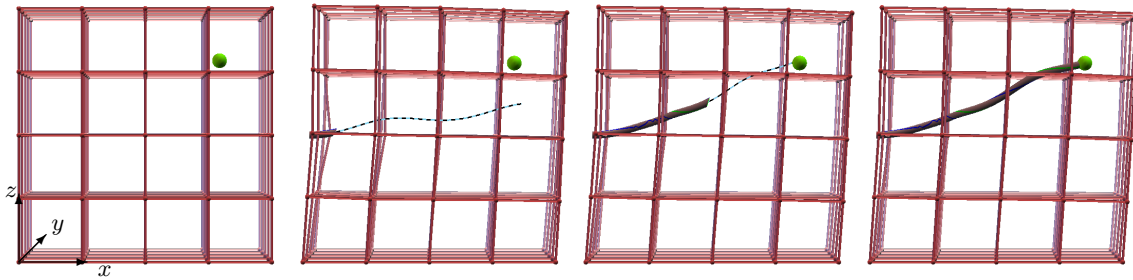


Fig. 8. A simulated trial of the controller in deformable tissue: initial setup, cutting through the surface, tracking the target, final needle configuration. Bottom plane is fixed. Grid used for visualization is at half the resolution of the simulation mesh. Dotted line depicts the trajectory predicted by the controller.

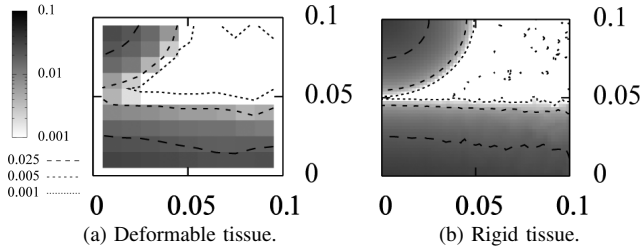


Fig. 9. Accuracy images for a deformable 10cm cube compared to a rigid cube. Needle was inserted at (0cm,5cm). Slice taken down the middle of the cube. Because deformable tissue simulation is computationally expensive, a lower resolution grid was used. Units in meters.

pursuing alternate methods to further understand why and when the controller converges.

ACKNOWLEDGMENT

This work is supported in part by NIH grant R01 EB 006435. We thank the reviewers for their helpful comments. We thank our collaborators and colleagues N. Cowan and G. Chirikjian of Johns Hopkins, J. O'Brien, J. Shewchuk, V. Duindam, and R. Jansen of U.C. Berkeley, and J. Pouliot, I-C. Hsu, and A. Cunha of UCSF for their valuable insights and suggestions.

REFERENCES

- [1] R. Alterovitz, T. Simeon, and K. Goldberg, "The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty," in *Robotics: Science and Systems*, June 2007.
- [2] R. Alterovitz, K. Goldberg, and A. M. Okamura, "Planning for steerable bevel-tip needle insertion through 2D soft tissue with obstacles," in *IEEE Intl. Conf. on Robot. and Autom.*, Apr. 2005, pp. 1652–1657.
- [3] A. Bowden, R. Rabbitt, and J. Weiss, "Anatomical registration and segmentation by warping template finite element models," in *Proc. SPIE (Laser-Tissue Interaction IX)*, vol. 3254, 1998, pp. 469–476.
- [4] H. Chen and F. Allgöwer, "Nonlinear model predictive control schemes with guaranteed stability," in *R. Berber and C. Kravaris, editors, Nonlinear Model Based Process Control*. Dordrecht: Kluwer Academic Publishers, 1998, pp. 465–494.
- [5] N. Chentanez, R. Alterovitz, D. Ritchie, L. Cho, K. Hauser, K. Goldberg, J. Shewchuk, and J. O'Brien, "Interactive simulation of surgical needle insertion and steering," in *ACM SIGGRAPH*, Aug 2009.
- [6] G. Chirikjian and J. Burdick, "A geometric approach to hyper-redundant manipulator obstacle avoidance," *ASME Journal of Mechanical Design*, vol. 114, pp. 580–585, December 1992.
- [7] —, "A modal approach to hyper-redundant manipulator kinematics," *IEEE Trans. Robot. and Autom.*, vol. 10, pp. 343–354, 1994.
- [8] G. Chirikjian and I. Ebert-Uphoff, "Numerical convolution on the euclidean group with applications to workspace generation," *IEEE Trans. Robot. and Autom.*, vol. 14, no. 1, pp. 123–136, February 1998.
- [9] S. DiMaio and S. Salcudean, "Needle insertion modelling and simulation," in *IEEE Intl. Conf. on Robot. and Autom.*, vol. 2, 2002, pp. 2098–2105.
- [10] —, "Needle steering and motion planning in soft tissues," *IEEE Trans. Biomedical Engineering*, vol. 52, no. 6, pp. 965–974, 2005.
- [11] M. Ding and H. Cardinal, "Automatic needle segmentation in three-dimensional ultrasound image using two orthogonal two-dimensional image projections," *Medical Physics*, vol. 30, no. 2, pp. 222–234, 2003.
- [12] V. Duindam, R. Alterovitz, S. Sastry, and K. Goldberg, "Screw-based motion planning for bevel-tip flexible needles in 3d environments with obstacles," in *IEEE Intl. Conf. on Robot. and Autom.*, May 2008, pp. 2483–2488.
- [13] V. Duindam, J. Xu, R. Alterovitz, S. Sastry, and K. Goldberg, "3d motion planning algorithms for steerable needles using inverse kinematics," in *Workshop on the Algorithmic Foundations of Robotics*, 2008.
- [14] D. Glozman and M. Shoham, "Flexible needle steering for percutaneous therapies," *Computer Aided Surgery*, vol. 11, no. 4, pp. 194–201, 2006.
- [15] O. Goksel, S. E. Salcudean, S. P. DiMaio, R. Rohling, and J. Morris, "3d needle-tissue interaction simulation for prostate brachytherapy," in *Medical Image Computing and Computer-Assisted Intervention MIC-CAI 2005*, 2005, pp. 827–834.
- [16] E. R. Hansen, *Global Optimization Using Interval Analysis*. New York: Marcel Dekker, 1992.
- [17] V. Kallem and N. J. Cowan, "Image-guided control of flexible bevel-tip needles," in *IEEE Intl. Conf. on Robot. and Autom.*, 2007.
- [18] D. Minhas, J. A. Engh, M. M. Fenske, and C. Riviere, "Modeling of needle steering via duty-cycled spinning," in *Proc. 29th Annual Intl. Conf. of the IEEE Eng. in Medicine and Biology Society*, August 2007, pp. 2756–2759.
- [19] H.-W. Nienhuys and A. van der Stappen, "A computational technique for interactive needle insertions in 3d nonlinear material," in *IEEE Intl. Conf. on Robot. and Autom.*, vol. 2, May 2004, pp. 2061–2067.
- [20] W. Park, Y. Liu, Y. Zhou, M. Moses, and G. S. Chirikjian, "Kinematic state estimation and motion planning for stochastic nonholonomic systems using the exponential map," *Robotica*, vol. 26, pp. 419–434, July 2008.
- [21] K. B. Reed, "Compensating for torsion windup in steerable needles," in *Proc. 2nd Biennial IEEE/RAS-EMBS Intl. Conf. on Biomedical Rob. and Biomechanics*, Scottsdale, Arizona, 2008.
- [22] K. B. Reed, V. Kallem, R. Alterovitz, K. Goldberg, A. M. Okamura, and N. J. Cowan, "Integrated planning and image-guided control for planar needle steering," in *Proc. 2nd Biennial IEEE/RAS-EMBS Intl. Conf. on Biomedical Rob. and Biomechanics*, Scottsdale, Arizona, 2008.
- [23] M. Saha and P. Isto, "Manipulation planning for deformable linear objects," *IEEE Trans. Robot.*, vol. 23, no. 6, pp. 1141–1150, Dec. 2007.
- [24] R. J. Webster III, N. J. Cowan, G. Chirikjian, and A. M. Okamura, "Nonholonomic modeling of needle steering," in *9th International Symposium on Experimental Robotics*, June 2004.
- [25] R. J. Webster III, J. Memisevic, and A. M. Okamura, "Design considerations for robotic needle steering," in *IEEE Intl. Conf. on Robot. and Autom.*, April 2005, pp. 3588–3594.
- [26] J. Xu, V. Duindam, R. Alterovitz, and K. Goldberg, "Motion planning for steerable needles in 3d environments with obstacles using rapidly-exploring random trees and backchaining," in *Proc. IEEE Conf. on Automation Science and Engineering*, August 2008, pp. 41–46.
- [27] H. Yamada, *Strength of Biological Materials*. Baltimore: Williams & Wilkins., 1990.