Planning "Fireworks" Trajectories for Steerable Medical Needles to Reduce Patient Trauma

Jijie Xu[1], Vincent Duindam[2], Ron Alterovitz[3], Jean Pouliot[4], J. Adam M. Cunha[4], I-Chow Joe Hsu[4] and Ken Goldberg[2]

Abstract— Accurate needle insertion in 3D environment is always a grand challenge. When multiple targets are located in the tissue, a procedure of inserting multiple needles from a single small region on the patient's skin, so called "fireworks" insertion as shown in Fig. 1, can be executed to further reduce trauma on the patient. In this paper, we explore motion planning for "fireworks" needle insertion in 3D environments by developing an algorithm based on the Forest of Rapidly-exploring Random Trees (RRTs). Given a set of targets, we propose an algorithm to quickly explore the configuration space by building a forest of RRTs and find feasible plans for multiple steerable needles from a single entry region. With different optimality considerations, we present two path selection algorithms to optimize the final plan among all feasible outputs. Finally, we implement the algorithm in an approximate prostate cancer treatment environment and simulation results demonstrate the performance of the proposed algorithm.

Fig. 1. An example of "fireworks" steerable needle insertion: 5 needles are inserted from very small entry region to reach 5 targets in the workspace while avoiding collision with the obstacles.

I. INTRODUCTION

Due to lack of maneuverability, limited visibility, and possible obstructions between the needle entry point and the target zone, accurate insertion of flexible steerable needle is still a challenge [1]. Researchers at Johns Hopkins University and the University of California, Berkeley have been developing a new class of highly flexible, bevel-tip needles, which offers improved mobility and manipulability, which enable them to reach previously inaccessible targets while avoiding sensitive or impenetrable areas [2], [3].

Motion planning for bevel-tip steerable needle has been studied in many ways in two-dimensional image planes [3], [4]. Motions of the steerable needle in 3D workspace are both nonholonomic and underactuated because of its bevel-tip design. To generate its path, the needle need to execute more flexible rotations, instead of the bevel-left/bevel-right strategy in 2D. Moreover, when multiple targets are located in the workspace, a "fireworks" insertion treatment [5] may be executed to insert multiple needle from a single entry region to reach all targets, which will reduces trauma on the patient but increases the difficulty of motion planning. Whereas prior work has focused on motion planning for a single steerable needle from a single start state to a single goal state, here we will introduce an algorithm to efficiently plan paths for this "fireworks" insertion.

This paper aims to use sampling-based motion planning techniques to explore motion planning of "fireworks" insertion with the bevel-tip steerable needles in 3D environments with obstacles. Inspired by the well-known Rapidly-exploring Random Trees (RRTs), we develop a motion planning algorithm to quickly grow a forest of RRTs to explore the configuration space and find feasible paths for all targets. With different optimality considerations, we present two path selection algorithms to optimize the final plan among all feasible outputs. With an approximated real environment, we implement our algorithm and simulation results demonstrate its performance.

II. RELATED WORK

It has been shown that the bevel-tip needle design significantly affects the needle bending forces during insertion [6]. Based on this observation, Webster et al. [7] showed experimentally that steerable bevel-tip needles follow paths of constant curvature in the direction of the bevel-tip, and the radius of that curvature is not significantly affected by the insertion velocity. They also developed a nonholonomic model of the needle's motion in stiff tissues based on a generalized bicycle model and fit model parameters using experiments with tissue phantoms [8].

Incorporating the effects of tissue deformations and motion uncertainties, motion planning for steerable bevel-tip needles in a 2D workspace has been studied in many ways. By modeling the bevel tip needle's motion in a 2D workspace as a non-reversible Dubins car, Alterovitz et al. [9] formulated the 2D steerable needle motion planning problem

1. The author is with the B. Thomas Golisano College of Computing and Information Science, Rochester Institute of Technology, USA. E-mail: jijie.xu@rit.edu. 2. The authors are with the Department of IEOR & EECS, University of California, Berkeley, USA. 3. The authors are with the Department of Computer Science, University of North Carolina at Chapel Hill, USA. 4. The authors are with the Helen Diller Family Comprehensive Cancer Center, University of California, San Francisco,USA.

as a nonlinear optimization problem that uses a simulation of tissue deformation during needle insertion as a function in the optimization. To consider motion uncertainties due to needle/tissue interaction, they further formulated the motion planning problem as a Markov Decision Process (MDP) [3], [10] and proposed the Stochastic Motion Roadmap (SMR) to search for the plan with most probability of success [11].

For clinical implementation, research on steerable needle insertion has been extended to more complex 3D environments in many different ways [12][13]. By representing the bevel-tip needle's 3D motion as a screw motion, Duindam et al. [14] formulated 3D motion planning problem for the steerable needle as a dynamical optimization problem with a discretized control space. Based on the inverse kinematics and the self-motion manifold of the steerable needle, they also presented a local motion planning algorithm for steerable needle in 3D environment by solving the Paden-Kahan subproblems [15]. Inspired by the Rapidly-exploring Random Tree (RRTs) algorithms, Xu et al. [16] developed the first specific 3D sampling-based motion planner for the steerable needle insertion, which efficiently builds a global tree to quickly and probabilistically explore the entire workspace and search for a feasible plan.

The Rapidly-exploring Random Tree (RRTs), which was first introduced by LaValle [17], is a successful roadmap-based motion planning techniques and has shown its potential in dealing with motion planning problems for nonholonomc systems [18]. It incrementally grows a tree toward the target configuration by searching feasible paths in the configuration space, and provides an efficient and quick search in complex environments of high dimensions with different constraints [18]. Based on the original RRTs structure, many variants have been developed to improve the efficiency of searching [19], extend RRTs to more complex configuration space ($\mathcal{C}$-space) [20], and enhance the ability to explore difficult region in searching environment [21]. Knepper et al. [22] experimentally studied the relationship between path sampling strategy and mobile robot performance, and showed that different deterministic samplings of path sets led to different performances of motion planners for mobile robots. In this paper, we extend our RRT-inspired algorithms developed in [16] and propose an RRT-forest exploration strategy together with an plan selection algorithm to efficiently solve the motion planning problem for the "fireworks" insertion.

III. Problem Statement

We make the following reasonable assumptions in order to obtain a well defined problem. First, all needles are rigid, identical and sufficiently flexible, such that rotating the needle at the base will not change its position in the workspace. Second, the motion of each needle is fully determined by its tip, which means that the needle body follows the path of the needle tip. Third, the feasible workspace is a stiff 3D cuboid, the needles are 1 dimensional curve in the 3D workspace, and all spatial obstacles are approximated using 3D spheres with various radii to simplify computational expenses for collision detection.
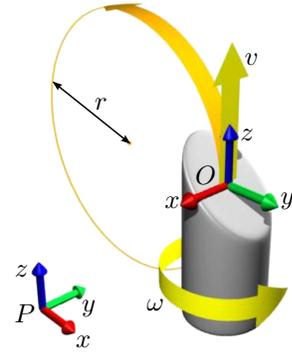
Fig. 2.   Model of the bevel-tip needle.

With the above assumptions, the steerable needle motion planning problem is well defined and stated as follows.

Problem *1*: **Motion Planning of Multiple Steerable Needles with Single Entry Region:** Given an entry region and a set of target configurations, determine a set of feasible paths and the corresponding sequences of controls (insertion depths and rotations for each needle) so that each target can be reached by a needle tip inserted from the entry region while avoiding obstacles and staying inside the workspace.

Input: Boundaries of the workspace, parameters of all needles, information of all obstacles, an entry region in the entry surface to insert all needles, and a set of target configurations to reach.

Output: A set of feasible entry points inside the entry region, a set of sequences of controls, with any of which there exists a needle from one entry point can reach one of the targets, or a report that no path is found.

Insertion planning with deformable environment and realistic obstacles with more complicated geometric shape will be considered in future work.

IV. Kinematics of Bevel-Tip Flexible Needle

Consider the bevel-tip needle shown in Fig. 2. Referring to the notations in [23], attach a spatial frame P to the base of the needle and a body frame O to the geometric center of the needle's bevel tip, respectively. The configuration of the needle tip can be represented homogeneously by the 4-by-4 transformation matrix of the object frame relative to the spatial frame,

$$g_{PO} = \begin{bmatrix} R_{PO} & p_{PO} \\ 0 & 1 \end{bmatrix} \in SE(3),$$

where $R_{PO} \in SO(3)$ is the rotation matrix and $p_{PO} \in T(3)$ is the position of frame O relative to frame S.

The motion of the needle is fully determined by two motions of the bevel tip: insertion with velocity $v(t)$ in the z direction and rotation with velocity $\omega(t)$ about the z axis of the body frame O [8], [14]. It has been experimentally shown [7] that the bevel-tip needle will follow a constantly curved path with curvature $\kappa = \frac{1}{r}$ while pushed with zero bevel rotation velocity, i.e. $\omega = 0$. The instantaneous velocity of the needle tip can be represented in the body frame O as

$$V_{PO}^b = [\mathbf{v}^T \quad \mathbf{w}^T]^T = [0 \quad 0 \quad v(t) \quad v(t)/r \quad 0 \quad \omega(t)]^T. \quad (1)$$
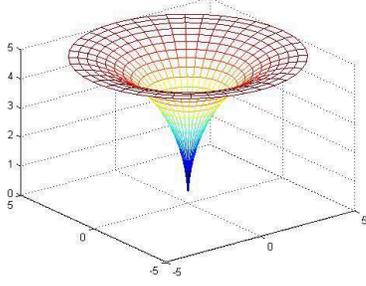
Fig. 3. The crateriform reachable region of local needle motion.

BUILD_FOREST($S_{\text{init}}, S_{\text{goal}}$)
1. for all $s_{goal}^i \in S_{goal}$
2. $\quad \mathcal{T}_i = \mathcal{T}_{\text{init}}(s_{\text{goal}}^i)$
3. $\quad \mathcal{F}_{add\_tree}(\mathcal{T}_i)$
4. while $No\_of\_Iteration < Max\_Iteration$
5. $\quad$ $s_{\text{rand}} \leftarrow$ RANDOM_STATE()
6. $\quad$ $\mathcal{F} \leftarrow$ EXTEND_FOREST($\mathcal{F}, s_{\text{rand}}$)
7. p*=SELECT_PATHS($\mathcal{F}$)
8. RETURN p*

TABLE I
THE SCENARIO OF ALGORITHM 1.

When V_{PO}^b is constant, i.e., $v(t)$ and $\omega(t)$ are constant, the configuration of the needle tip relative to the spatial frame after being pushed for a time interval t is

$$g_{PO}(t) = g_{PO}(0)e^{\hat{V}_{PO}^b t}, \quad (2)$$

where $g_{PO}(0)$ is the initial configuration of the needle frame relative to the spatial frame, and

$$\hat{V}_{PO}^b = \begin{bmatrix} 0 & -\omega(t) & 0 & 0 \\ \omega(t) & 0 & -v(t)/r & 0 \\ 0 & v(t)/r & 0 & v(t) \\ 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3)$$

According to the kinematics analysis in [14], the trajectory of the needle only depends on the ratio ω/v but not on the values of the two individual terms. Moreover, the nature of motion planning is to find a trajectory regardless the scaling in time, which means moving along a trajectory with different speed does not change its shape. Therefore, we discretize the entire insertion into N steps with N time segments $\{I_1, \cdots, I_N\}$ and assume that the velocity in each single step, $V_{PO}^b(I_n)$, is constant. The final needle tip configuration can be computed as a product of exponentials [23], [14]

$$g_{PO}(T) = g_{PO}(0)e^{\hat{V}_{PO}^b(I_1)I_1} \cdots e^{\hat{V}_{PO}^b(I_N)I_N}. \quad (4)$$

Note that the bevel-tip needle can only follow curved paths with a curvature $\kappa = 1/r$, as shown in Fig. 2. For this reason, reachable configurations of the needle are locally constrained inside the volume of a crateriform region (see Fig. 3) defined by

$$p_z \geq \sqrt{2r\sqrt{p_x^2 + p_y^2} - p_x^2 - p_y^2}, \quad (5)$$

where (p_x, p_y, p_z) are coordinates of a point in the body frame O.

V. MOTION PLANNING FOR MULTIPLE STEERABLE NEEDLE USING RRT FOREST

A. "Fireworks" needles insertion planning based on RRTs Forest

The configuration of the needle tip is determined by its position (x, y, z) together with its three Euler angles (ϕ, θ, ψ). Since the insertion task only requires the needle to reach a specific position in the 3 dimensional workspace,

the configuration space $\mathcal{C}$ for motion planning is equivalent to $\mathbb{R}^3$. Following its kinematics (2), the needle's trajectory to reach a target at $g_{PO}(t)$ can be computed backwardly from its target as

$$g_{PO}(t - \delta t) = g_{PO}(t)e^{-\hat{V}_{PO}^b \delta t}. \quad (6)$$

Given the workspace's boundaries, ($[x_{\min}, x_{\max}]$, $[y_{\min}, y_{\max}]$, and $[z_{\min}, z_{\max}]$), the obstacles information, the target configurations s_{goal} and the specified entry zone S_{entry}, a motion planning algorithm based on RRTs with backchaining has been developed to quickly explore the configuration space from the target to find feasible soultions for insertion tasks with single needle and single target (SNST) [16].

For "fireworks" insertion task, one can separatively execute the algorithm for SNST for each individual target. However, repeated sampling procedures for each needle unnecessarily increase the computational cost. Moreover, separated collision detection for a needle in an environment with previous solutions of other needles also requires extensive computational cost. By extending that algorithm, we propose a new algorithm to quickly grow a forest of RRTs to search feasible paths for all targets simultaneously.

Algorithm *1 (Forest of RRTs with backchaining):* For all $s_{\text{goal}}^i \in S_{\text{goal}}$, initialize the forest $\mathcal{F}$ by initializing all trees $\mathcal{T}_i$ rooting at s_{goal}^i. Randomly sample a state s_{rand} in the collision free configuration space $\mathcal{CS}_{\text{free}}$. For all trees in the forest, the reachable neighbor test is executed for all node $s_i \in \mathcal{T}_i$ to find a reachable set S_{reach}^i, which contains all nodes reachable from s_{rand}. The nearest neighbor search is executed inside S_{reach}^i to find the nearest neighbor of s_{rand}, denoted by $s_{\text{near}}^i \in S_{reach}^i$, which has the shortest distance to s_{rand}. The distance used in the nearest neighbor search can be defined in different ways by defining different metrics on the configuration space. Then we uniformly sample the negative control space $-\mathcal{U}$, and apply all sampled control inputs to s_{near}^i for a small time increment $\delta t > 0$ to generate a set of possible new states S_{new}^i from any $\mathcal{T}_i$. Again, the nearest neighbor of s_{rand}, $s_{\text{new}}^i \in S_{\text{new}}^i$, is added to each $\mathcal{T}_i$. Such strategy is repeated until the number of iteration reaches its predefined limit. Finally, a path selection process is executed to find a set of "optimal" feasible paths based on specific optimization criteria, or a failure report is output with no feasible plan found.

RANDOM_STATE()
 1. $p = rand(0, 1)$
 2. if $p < p_1$
 3. s_{rand} =UNIFORM_SAMPLE(S_{init})
 4. if $p_1 < p < p_2$
 5. s_{rand} =UNIFORM_SAMPLE(S_{goal})
 6. else
 7. s_{rand} =UNIFORM_SAMPLE($CS_{\text{free}}/(S_{\text{init}} \cup S_{\text{goal}})$)
 8. RETURN s_{rand}

EXTEND_FOREST($\mathcal{F}, s_{\text{rand}}$)
 1. for all $\mathcal{T}_i \in \mathcal{F}$
 2.   EXTEND_TREE($\mathcal{T}_i, s_{\text{rand}}$)
 3. RETURN $\mathcal{F}$

EXTEND_TREE($\mathcal{T}, s_{\text{rand}}$)
 1. $S_{\text{reach}}^i \leftarrow$ REACHABLE_NEIGHBORS($\mathcal{T}, s_{\text{rand}}$)
 2. $s_{\text{near}} \leftarrow$ NEAREST_NEIGHBOR($S_{\text{reach}}^i, s_{\text{rand}}$)
 3. $(s_{\text{new}}, u_{\text{new}}) \leftarrow$ NEW_STATE($s_{\text{near}}, s_{\text{rand}}, \mathcal{U}$)
 4. $\mathcal{T}$.add_vertex(s_{new})
 5. $\mathcal{T}$.add_edge($s_{\text{near}}, s_{\text{new}}, u_{\text{new}}$)
 6. RETURN $\mathcal{T}$

REACHABLE_NEIGHBORS($\mathcal{T}, s_{\text{rand}}$)
 1. For all $s_i \in \mathcal{T}$
 2.   if s_i is reachable from s_{rand}
 3.   add s_i to S_{reach}
 4. RETURN S_{reach}.

NEW_STATE($s_{\text{near}}, s_{\text{rand}}, \mathcal{U}$)
 1. $\mathcal{U}_{\text{rand}} \leftarrow$ CONTROL_SAMPLING($\mathcal{U}$)
 2. FOR all $u_i \in \mathcal{U}_{\text{rand}}$
 3. $s_{\text{new}}(i) = s_{\text{near}} + F_{q_{\text{near}}}(s, -u_i)\delta t$
 4. $S_{\text{new}} = \cup_i s_{\text{new}}(i)$
 5. $s_{\text{new}} \leftarrow$ NEAREST_NEIGHBOR($S_{\text{new}}, s_{\text{rand}}$)
 6. $u_{\text{new}} = -u_i$ such that $s_i = s_{\text{new}}$
 7. RETURN $s_{\text{new}}, u_{\text{new}}$

TABLE II
DETAILED PROCEDURE OF ALGORITHM 1.

One concern of RRT-based motion planning algorithm with backchaining is the efficient growth of the reversed RRTs toward the entry region. In order to explore the forest quickly toward the entry region, we apply a biased sampling strategy in $\mathcal{CS}_{\text{free}}$, with which the state s_{rand} is sampled with a higher density in S_{goal} than elsewhere in the configuration space. If s_{rand} collides with any obstacle, it is discarded and new states are sampled until one in $\mathcal{CS}_{\text{free}}$ is found.

The control inputs are sampled uniformly in the control space, using CONTROL_SAMPLING(), within a predefined range $[v_{\text{min}}, v_{\text{max}}] \times [\omega_{\text{min}}, \omega_{\text{max}}]$. By doing so, we not only explore the RRT toward all possible directions with same probability, but also extend the RRT toward the sampled states by various stepsize with same probability. The full scenario of Algorithm 1 is shown in Table. I and table. II.

B. Paths Selection

If a target s_{goal}^i is reachable from the entry region, Algorithm 1 will output a set of feasible paths for this target, denoted by P_i. SELECT_PATHS() will select and output the "best" plan, $P^* = \{p_1^*, \cdots, p_n^*\}$, for the given target set S_{goal}, where p_i^* is the "best" path for s_{goal}^i. With consideration of different optimization criteria, different path selection strategies can be applied.

1) Plan with minimal twists: While inserting a needle into the patient's body, rotating the needle will always cause damages to the tissue. Because of the discretization of the needle's kinematics, the more segments a feasible path contains, the more damage it will cause on the tissue. In addition, uncertainties in needles' motions will greatly affect the performance and accuracy of the insertion. Since the control of the needles is open loop, the more control inputs a feasible path has, the more uncertainties this path will involve. Let n_k be the number of control inputs corresponding to the kth path in P_i, the p_i^* for each s_{goal}^i can be selected using the following minimal twists strategy,

$$p_i^* = \arg \min_{p_k \in P_i} (n_k).$$

2) Plan with minimal insertion region: Inserting multiple needles to reach multiple targets may require more area of entry region. This will increase the complexity of the treatment as well as the trauma on the patient. As described in its scenario, Algorithm 1 consider this issue by exploring the connectivity between different targets so that possible plan may be found to reach multiple targets with only one needle. Besides that, the entry region can be further minimized based on the output path sets, P_is. Let e_k^i be the corresponding entry point of the kth path, $p_i^k \in P_i$, for target s^i. The optimal plan P^* can be selected by using the following minimal entry region strategy,

$$P^* = \arg \min_{k,l} (\max_{i,j} d(e_k^i, e_l^j)),$$

where $d(e_k^i, e_l^j)$ is the Euclidean distance between e_k^i and e_l^j. This strategy minimizes the maximal distance between possible entry points of any two needles.

Remark *1:* Because of the probabilistic characteristic of RRTs, the output are only feasible plans found in the iteration limits. Therefore, the proposed path selection strategies only reflet the sub-optimality inside the set of feasible plans.

VI. SIMULATION AND DISCUSSION

A. Simulation setup

We implement the proposed RRT-Forest-based motion planning algorithm for multiple steerable needles insertion using an approximated 3D prostate environment. The real prostate environment is very complex, which contains many obstacles with complicated shapes, such as the urethra, the pubic arch, the penile bulb, etc. To simplify the computational cost for obstacle collision avoidance, we use a set of spherical obstacles with different radii to approximate the real environment as shown in Fig. 4. The entire workspace
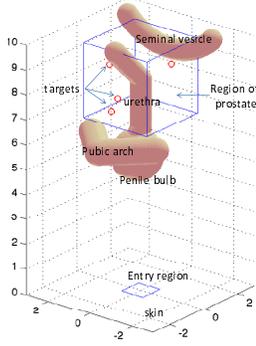
Fig. 4. The simulation setup: An approximated environment for multiple needle insertion.

(a)   (b)
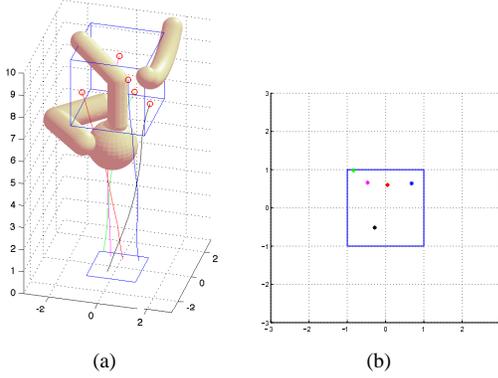
Fig. 5. A set of paths for simulation task 1: For all the 5 targets, only one path from each path set is selected with plan selection algorithm 1 to minimize twists of the path.

is defined as a cubiod with coordinates $(-3, 3) \times (-3, 3) \times (0, 10)$. The region of interest around the prostate is defined as a cube of $3 \times 3 \times 3$ located at the top of the workspace. The possible entry region is defined as a 2×2 square in the skin plane (X-Y plane). The assumption of spherical obstacles and 1D needles make it easy to implement a fast collision detection simply using Euclidean distance between the needle's trajectories and the obstacles' surfaces. Interference of different needles are not considered until selecting the final optimal insertion plan. Simulations are run on a laptop with the AMD® Turion$_{TM}$64 CPU @2.1 GHz, 4 GB memory, and the Microsoft® Windows Vista™ operating system.

B. Simulation Results

The insertion task is to insert multiple needles from the 2×2 rectangular entry region centering at $(0, 0, 0)$ in the skin plane to reach totally 5 targets, which are randomly generated in the region of interest. The range of the uniformly sampled control inputs are defined by insertion depth in the range of $[0.1, 0.5]$ and rotation angle in the range of $[0, 2\pi]$. We totally run 5 trials, all of which successfully find feasible plans within 10000 iterations with an average computational time of 57573.4 seconds. Figure 7 shows the feasible plans found for all 5 targets in one trial. For each target in this trail, we also implemented the SNST algorithm for individual needle. Feasible paths for each target are successfully found in 10000

(a)   (b)
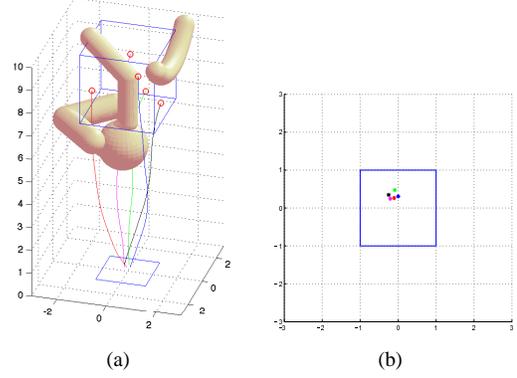
Fig. 6. A set of paths for simulation task 1: For all the 5 targets, only one path from each path set is selected with plan selection algorithm 2 to minimize region for insertion.

iterations with a computational time of 87755 seconds. This is because that Algorithm 1 used every random sample to explore all 5 random trees, which is more efficient that separately applying the SNST algorithm and significantly reduce the computational time required by repeated sampling process. Figure 5-(a) shows the "optimal" plan selected using minimal twist strategy, and Figure 5-(b) shows the corresponding entry points. Figure 6-(a) shows the "optimal" plan selected using minimal entry region strategy, and Figure 6-(b) shows the corresponding entry points.

C. Discussion

The RRT-Forest-based algorithm efficiently explores the collision-free configuration space and built its connectivity. With the minimal twist strategy, the entry points for all targets are selected in a way to minimize the path length to the corresponding target. However, this leads to a relatively larger entry region. With the minimal entry region strategy, the entry region is further minimized among all possible plans. As shown in Figure 6-(b), the "optimal" plan for this trial can be reduced into a circular region of radius less than 0.3. This can significantly reduce the complexity of the treatment as well as the patient's trauma. For example, with the precurved-tube continuum robots recently developed by Webster III, et. al. [24], this plan can be executed from only one entry hole to reach all targets.

VII. CONCLUSIONS AND FUTURE WORK

In this paper, we presented an algorithm for motion planning of multiple bevel-tip steerable needles to reach multiple targets in 3D environments with obstacles. This algorithm is inspired by the RRT-based motion planning algorithm with backchaining for solving feasible entry point planning problem for a single needle [16]. The proposed algorithm builds an RRT forest structure and provides a quick and more efficient exploration from all targets toward a single entry region to find feasible plans. By minimizing either the damage to the tissue or the final entry region, two path selection strategies were developed to further optimize the final "fireworks" insertion plan among all output feasible
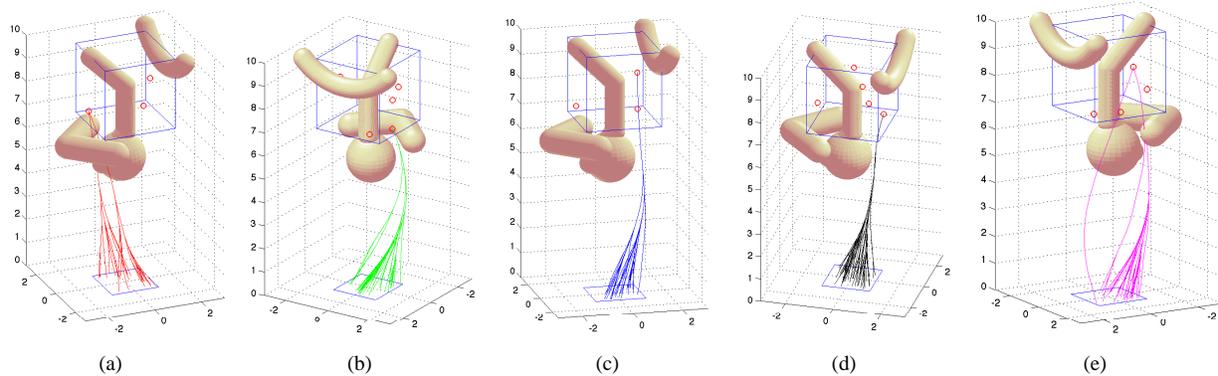
Fig. 7. Feasible paths found for one trial: For each of the 5 targets, a set of feasible paths is found. With any path in the path set, a steerable needle can be inserted from the single entry region to reach the target.

plans. Finally, we implemented this algorithm in an approximated prostate environment and provided simulation results to verify its performance.

In this paper, we only considered stiff environments with non-deformable spherical obstacles. In future work, we will explore the motion planning for steerable needle in deformable environments with obstacles of more complex shapes. Moreover, we will further explore the feasibility of reaching multiple targets with single needle in our future work.

ACKNOWLEDGEMENT

REFERENCES

[1] A. Henderson, R.W. Laing, and S.E.M. Langley. Identification of pubic arch interference in prostate brachytherapy: Simplifying the transrectal ultrasound technique. *Brachytherapy*, (4):240–245, 2003.

[2] Robert J. Webster III, Jin Seob Kim, Noah J. Cowan, Gregory S. Chirikjian, and Allison M. Okamura. Nonholonomic modeling of needle steering. *International Journal of Robotics Research*, 25:509–525, May 2006.

[3] R. Alterovitz, K. Goldberg, and A. M. Okamura. Planning for steerable bevel-tip needle insertion through 2d soft tissue with obstacles. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 1640–1645, 2005.

[4] R. ALterovitz and K. Goldberg. *Motion Planning in Medicine: Optimization and Simulation Algorithms for Image-Guided Procedures. Springer Tracts in Advanced Robotics Vol 50*. Springer, 2008.

[5] A. Cunha and J. Pouliot. Dosimetric equivalence of non-standard brachytherapy needle patterns. *Medical Physics*, (6):2737–2737, 2008.

[6] M. D. O'leary, C. Simone, T. Washio, K. Yoshinaka, and A. M. Okamura. Robotic needle insertion: Effects of friction and needle geometry. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2003.

[7] R. J. Webster III, J. S. Kim, N. J. Cowan, G. S. Chirikjian, and A. M. Okamura. Nonholonomic modeling of needle steering. *International Journal of Robotics Research*, 5/6:509–525, 2006.

[8] R. J. Webster III, J. Memisevic, and A. M. Okamura. Design consideration for robotic needle steering. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2005.

[9] R. Alterovitz, A. Lim, K. Goldberg, G. S. Chirikjian, and A. M. Okamura. Steering flexible needles under markov motion uncertainty. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 120–125, 2005.

[10] R. Alterovitz, M. Branicky, and K. Goldberg. Constant-curvature motion planning under uncertainty with applications in image-guided medical needle steering. In *Proceedings of workshop on the Algorithmic Foundations of Robotics*, 2006.

[11] R. Alterovitz, T. Siméon, and K. Goldberg. The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty. In *Proceedings of Robotics: Science and Systems*, June, 2007.

[12] V. Kallen and N. J. Cowan. Image-guided control of flexible beveltip needles. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 3015–3020, 2007.

[13] W. Park, J. S. Kim, Y. Zhou, N. J. Cowan, A. M. Okamura, and G. S. Chirikjian. Diffusion-based motion planning for a nonholonomic flexible needle model. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 4611–4616, 2005.

[14] V. Duindam, R. Alterovitz, S. Sastry, and K. Goldberg. Screw-based motion planning for bevel-tip flexible needles in 3d environments with obstacles. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 2483–2488, 2008.

[15] V. Duindam, J. Xu, R. Alterovitz, Ken Goldberg, and S. Sastry. 3d motion planning algorithms for steerable needles using inverse kinematics. In *Proceedings of International Workshop on Algorithmic Foundations of Robotics*, 2008.

[16] J. Xu, V. Duindam, R. Alterovitz, and Ken Goldberg. Motion planning for steerable needles in 3d environments with obstacles using rapidly-exploring random trees and backchaining. In *Proceedings of IEEE International Conference on Automation Science and Engineering*, 2008.

[17] S. M. LaValle. *Rapidly-exploring random trees: a new tool for path planning*. TR 98-11, Department of Computer Science, Iowa State University, 1998.

[18] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.

[19] J. Kuffner and S. LaValle. RRT-connect: An efficient approach to single-query path planning. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2000.

[20] M. S. Branicky, M. M. Curtiss, J. Levine, and S. Morgan. RRTs for nonlinear, discrete, and hybrid planning and control. In *Proceedings IEEE Conference Decision & Control*, 2003.

[21] S. Rodriguez, X. Tang, J. Lien, and N. Amato. An obstacle-based rapidly-exploring random tree. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 895–900, 2006.

[22] R. A. Knepper and M. T. Mason. Empirical sampling of path sets for local area motion planning. submitted to *the 11th International Symposium on Experimental Robotics*, 2008.

[23] R. Murray, Z.X. Li, and S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[24] R. J. Webster III, J. M. Romano, and N. J. Cowan. Mechanics of precurved-tube continuum robots. *IEEE Transactions on Robotics*, (1):67–78, 2009.