6-8-2016

3D Dynamic Motion Planning for Robot-Assisted Cannula Flexible Needle Insertion into Soft Tissue

Yanjiang Zhao  
*Harbin University of Science and Technology*

Wenqiang Wu  
*Harbin University of Science and Technology*

Yongde Zhang  
*Harbin University of Science and Technology*

Ruixue Wang  
*Harbin University of Science and Technology*

Jingchun Peng  
*Harbin University of Science and Technology*

*See next page for additional authors*

**Let us know how access to this document benefits you**

Follow this and additional works at: [http://jdc.jefferson.edu/radoncfp](http://jdc.jefferson.edu/radoncfp)

Part of the [Medicine and Health Sciences Commons](http://jdc.jefferson.edu/radoncfp)

**Recommended Citation**

Zhao, Yanjiang; Wu, Wenqiang; Zhang, Yongde; Wang, Ruixue; Peng, Jingchun; and Yu, Yan, "3D Dynamic Motion Planning for Robot-Assisted Cannula Flexible Needle Insertion into Soft Tissue" (2016). *Department of Radiation Oncology Faculty Papers*. Paper 81.  
http://jdc.jefferson.edu/radoncfp/81

This Article is brought to you for free and open access by the Jefferson Digital Commons. The Jefferson Digital Commons is a service of Thomas Jefferson University’s Center for Teaching and Learning (CTL). The Commons is a showcase for Jefferson books and journals, peer-reviewed scholarly publications, unique historical collections from the University archives, and teaching tools. The Jefferson Digital Commons allows researchers and interested readers anywhere in the world to learn about and keep up to date with Jefferson scholarship. This article has been accepted for inclusion in Department of Radiation Oncology Faculty Papers by an authorized administrator of the Jefferson Digital Commons. For more information, please contact: JeffersonDigitalCommons@jefferson.edu.
Authors
Yanjiang Zhao, Wenqiang Wu, Yongde Zhang, Ruixue Wang, Jingchun Peng, and Yan Yu
3D Dynamic Motion Planning for Robot-assisted Cannula Flexible Needle Insertion into Soft Tissue

Regular Paper

Yan-Jiang Zhao¹ *, Wen-Qiang Wu¹, Yong-De Zhang¹, Rui-Xue Wang¹, Jing-Chun Peng¹ and Yan Yu²

¹ Intelligent Machine Institute, Harbin University of Science and Technology, Harbin, China
² Department of Radiation Oncology, Thomas Jefferson University, Philadelphia, USA
*Corresponding author(s) E-mail: zhaoyj@hrbust.edu.cn
Received 25 January 2016; Accepted 11 May 2016
DOI: 10.5772/64199

© 2016 Author(s). Licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract
In robot-assisted needle-based medical procedures, insertion motion planning is a crucial aspect. 3D dynamic motion planning for a cannula flexible needle is challenging with regard to the nonholonomic motion of the needle tip, the presence of anatomic obstacles or sensitive organs in the needle path, as well as uncertainties due to the dynamic environment caused by the movements and deformations of the organs. The kinematics of the cannula flexible needle is calculated in this paper. Based on a rapid and robust static motion planning algorithm, referred to as greedy heuristic and reachability-guided rapidly-exploring random trees, a 3D dynamic motion planner is developed by using replanning. Aiming at the large detour problem, the convergence problem and the accuracy problem that replanning encounters, three novel strategies are proposed and integrated into the conventional replanning algorithm. Comparisons are made between algorithms with and without the strategies to verify their validity. Simulations showed that the proposed algorithm can overcome the above-noted problems to realize real-time replanning in a 3D dynamic environment, which is appropriate for intraoperative planning.

Keywords Dynamic Motion Planning, Rapidly-exploring Random Tree, Cannula Flexible Needle, Robot-assisted Surgery

1. Introduction

In minimally invasive surgeries, needle insertion is likely one of the most popular procedures, applied in tissue biopsies and radioactive brachytherapies for cancers. However, targeting is a challenge when targets are surrounded by anatomic obstacles or sensitive organs that must be avoided. Traditional rigid needles are not useful for this task. Therefore, a bevel tip flexible needle is proposed for overcoming this problem [1]. The conventional flexible needle is supposed to be flexible to a degree that it can bend inside tissue, due to lateral force being applied on the bevel tip at insertion. However, it still has some drawbacks: firstly, the curvature of the trajectory for a needle is supposed to be fixed in a tissue and cannot rectify its path once it has deviated. Secondly, it is difficult to precisely control the orientation of the bevel tip at
rotation, due to the torsional friction between the needle shaft and tissue. To overcome the first drawback, Minhas and Majewicz utilized the duty-cycling of the spinning motor to generate a series of curvatures [2, 3]. Datla et al. proposed an active, flexible needle that takes advantage of the characteristics of shape memory alloys (SMA), which can also generate different curvatures by supplying different electric currents to the SMA actuators [4-6]. However, the second drawback remains in place. We have been developing a cannula flexible needle, composed of a flexible cannula and a flexible bevel-tip stylet, as shown in Fig. 1. It can overcome both drawbacks of the conventional bevel tip flexible needle mentioned above. Firstly, it can generate a series of curvatures of trajectories by the different the length $d$ of the stylet out of the cannula. Secondly, it can improve the rotation precision of the bevel tip, because the cannula separates the stylet and the tissue, thereby reducing torsional friction.

**Figure 1.** Schematic of a cannula flexible needle

Motion planning of the cannula flexible needle in the soft tissue is challenging due to the nonholonomic motion of the needle and the presence of anatomic obstacles and sensitive organs [7]. It is even complicated in a dynamic environment, with uncertainties present due to errors in needle tip positioning and needle modelling, the inhomogeneity and deformation of tissue and the physiological movement of organs [8]. All of these disturbances and movements may cause the needle to deviate from its intended path, which can have fatal consequences.

Motion planning for the conventional bevel tip flexible needle has been extensively studied using different approaches, which can be used as references for the cannula flexible needle. Duindam et al. proposed motion planning using discretization of the control space in a 3D environment with obstacles and formulated the motion planning problem as a nonlinear optimization problem [7]. However, this planning is applied for a static environment. Alterovitz et al. depicted the motion planning problem as a Markov decision process, using a discretization of the control space in a 3D environment or a quasi-dynamic environment [12, 13]. Ever since Xu et al. first applied a RRT-based method to search for valid needle paths in a 3D environment with obstacles, the RRT algorithm has been commonly used in flexible needle motion planning [14]. However, these studies were only considered in a static environment. Patil et al. utilized improved RRTs called reachability-guided RRTs (RG-RRTs) and achieved orders of magnitude speed-ups compared to previous approaches, and relaxed the constraint of constant-curvature needle trajectories by relying on duty-cycling to realize bounded-curvature needle trajectories. This enabled the RRT method to be appropriate for dynamic intraoperative motion planning [15]. Caborni et al. proposed risk-based path planning for a steerable flexible probe based on the RG-RRTs [16]. In terms of path replanning, Patil et al. proposed a rapid replanning algorithm based on RG-RRTs and conducted experimental evaluations to show its validity [17]. Vrooojink et al. also developed a closed-loop replanning algorithm based on the RG-RRTs for a 3D non-static environment using 2D ultrasound images [18]. Both of these studies considered the uncertainty arising from tissue deformations, actuation errors and noisy sensing; however, they did not consider the significant displacement of targets and obstacles caused by physiological movement, which may cause the needle to fatally deviate from the planned path. Bernardes et al. presented a fast intraoperative replanning algorithm that considers disturbances including the movements of
obstacles and a target based on RRTs in both 2D and 3D environments; however, target movement was limited in 2D, despite having the potential to move in 3D [19, 20].

In summary, the advantage of the RRT method is that it is very fast (in milliseconds), easy to implement and probabilistic to complete, which is suitable for intraoperative planning. However, for dynamic replanning, all replanning algorithms may suffer from the large detour problem, convergence problem and/or accuracy problem. The large detour problem may be caused by the unpredicted movement of the obstacles and/or target; the convergence problem may be caused by nonholonomic motion and/or the minimum bending radius of the needle in the tissue, and the potential for the path adjustment becomes weaker when the needle gets close to the target; the accuracy problem may be caused by the termination conditions when the convergence problem occurs.

In this paper, we firstly calculate the kinematics of the cannula flexible needle, then introduce a fast motion planning algorithm based on RRT for a static environment (preoperative operation) for the cannula flexible needle in our previous work. Based on the proposed static motion planning algorithm, we propose dynamic (intraoperative operation) motion planning using replanning, taking into account any uncertainty caused by errors in needle tip positioning, the approximation of needle modelling, the inhomogeneity and deformation of tissue and physiological movement of the organs. Aiming at the large detour problem, the convergence problem and the accuracy problem that replanning encounters, we propose three novel strategies integrated into the intraoperative path replanning algorithm for overcoming these problems.

2. Kinematic Analysis of the Cannula Flexible Needle

Different from conventional bevel tip needles (with two DOFs: insertion and rotation) [1], the proposed cannula flexible needle has three DOFs: two translations \( u_1, u_2 \) for the cannula and the stylet, respectively, and rotation \( u_3 \) for the stylet, \( u_1, u_2 \) and \( u_3 \) can be input separately or simultaneously (see Fig.1).

2.1 Trajectory form analysis

We assume that the cannula flexible needle is pliably flexible and torsionally stiff, i.e., the needle shaft follows the needle tip, performing an approximate circular arc in the tissue when bending, while needle tip rotates at the same angle as the base [1, 7]. Similar to the conventional bevel tip flexible needles, it will be bent against the bevelled side when inserted into the tissue, due to the lateral force. The rotation of the stylet decides the orientation of the needle bending, so that the needle can achieve various paths in 3D by combining the three inputs.

1. Arc-based path: the radius of the needle path can vary according to the length \( d \) of the stylet which remains out of the cannula (see Fig.1). The more the stylet is able to get out of the cannula, the smaller radius the trajectory will receive. Thus, the relative velocity between \( u_1 \) and \( u_2 \) changes the radius of the path, and \( u_3 \) changes the orientation of the bending. As \( u_1 \) and \( u_2 \) are generally input simultaneously and with the same velocity, we can denote them as \( u_{12} \). If we input \( u_{12} \) and \( u_3 \) in a time-sharing way, the needle will generate an arc-based path (see Fig.2a). This method does not generate all radii of the path. The path will be in a range with the minimum bending radius \( r_{\text{min}} \) and the maximum bending radius \( r_{\text{max}} \) (as shown in Fig.3a). Both \( r_{\text{min}} \) and \( r_{\text{max}} \) are not only related to \( d \), but also to the mechanical properties of the needle and the tissue. For the paths with a bending radius larger than \( r_{\text{max}} \), we can also apply the duty-cycling method [2, 3]. By using a combination of both methods, the control expense is saved to some extent. The entire workspace is shown in Fig.3b.

2. Helix-based path: if we input \( u_{12} \) and \( u_3 \) simultaneously, and the speed of \( u_3 \) is much slower than that of \( u_{12} \), the needle will generate a helix-based path (see Fig.2b).

3. Linear path: if we input \( u_{12} \) and \( u_3 \) simultaneously, however, the speed of \( u_3 \) will be much faster than that of \( u_{12} \), and the needle will generate a linear path (see Fig.2c).

![Figure 2. Forms of path](image)
![Figure 3. Workspace of the needle](image)
control. The helix-based path, in contrast, is difficult to formulate and control and may also cause more trauma to biological organs, because its winding approach may lengthen the path.

2.2 Kinematic model

The kinematic model of the cannula flexible needle is formulated as shown in Fig.4. In the world frame $\Psi_w$, the configuration of the needle tip frame $\Psi_n$ can be described compactly by a $4 \times 4$ homogeneous transformation matrix:

$$\mathbf{g}_{wn} = \begin{bmatrix} R_{wn} & p_{wn} \\ 0 & 1 \end{bmatrix} \in \text{SE}(3) \quad (1)$$

where $R_{wn} \in \text{SO}(3)$ and $p_{wn} \in \mathbb{R}^3$ are the rotation matrix and the position of frame $\Psi_n$ relative to the frame $\Psi_w$, respectively.

In frame $\Psi_w$, the instantaneous linear velocity is $v = \begin{bmatrix} 0 & 0 & u_{12} \end{bmatrix}^T$, angular velocity is $\omega = \begin{bmatrix} u_{12} / r & 0 & u_3 \end{bmatrix}^T$, where $r$ is the radius of the path and the twist $\xi \in \text{se}(3)$ is formulated as:

$$\dot{\xi} = \left[ \begin{array}{c} v \\ \omega \end{array} \right]^\wedge = \begin{bmatrix} 0 & -u_3 & 0 & 0 \\ u_3 & 0 & -u_{12} / r & 0 \\ 0 & u_{12} / r & 0 & u_3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (2)$$

where $^\wedge$ is the wedge operator for forming a matrix in $\text{se}(3)$ out of a given vector in $\mathbb{R}^6$.

Then, the homogeneous transformation matrix can be formulated in exponential form as:

$$\mathbf{g}_{wn}(t) = \mathbf{g}_{wn}(0) \exp(\dot{\xi} t) \quad (3)$$

where $\mathbf{g}_{wn}(0)$ is the initial configuration, which is the configuration of the needle (frame $\Psi_n$) in frame $\Psi_w$ before insertion and $t$ is the execution time. Additional details can be found in [1].

However, different to other existing kinematics of the flexible needle [1, 7, 9, 11-20], because the planning algorithm considers the insertion pose of the needle, we have to add an insertion configuration to the kinematics:

$$\mathbf{g}_{in} = \mathbf{g}_{in}(0) \quad (4)$$

where $\mathbf{g}_{in}$ is the insertion configuration.

The entire path can be discretized by a few segments. Let $\mathbf{g}_i(t_i) = \exp(\dot{\xi}_i t_i)$ represent the transformation matrix of the $i$th segment, then the forward kinematics after $N$ segments of execution is formulated as:

$$\mathbf{g}_{wn}(T) = \mathbf{g}_{wn}(0) \prod_{i=1}^{N} \mathbf{g}_i(t_i) \quad (5)$$

where $t_i$ is the execution time of the $i$th segment, $T$ is the total execution time of the whole path, $T = \sum_{i=1}^{N} t_i$; $\mathbf{g}_{wn}(0)$ is the initial configuration and $\mathbf{g}_{in}$ is the insertion configuration.

To simplify the calculation, we make the $\Psi_n$ frame coincide with the $\Psi_w$ frame initially (see Fig.5). Thus, $\mathbf{g}_{wn}(0) = I_4$, where $I_4$ is a unit matrix with $4 \times 4$. The insertion pose can be regarded as frame $\Psi_w$ rotating $\alpha_0$ around axis $Z_w$ then rotating $\beta_0$ around axis $Y_w$; hence, configurations for the insertion configuration can be obtained by:

$$\mathbf{g}_{in} = \text{Rot}(Z_w, \alpha_0) \text{Rot}(Y_w, \beta_0) \quad (6)$$
3. Static Motion Planning Algorithm

We utilized the RRT method for the static motion planning, combined with the greedy heuristic method and reachable guided strategies, known as greedy heuristic and reachability guided rapidly-exploring random trees (GHRG-RRTs). We adopted variable but bounded curvatures for needle paths and also took account of linear segments and relaxation of insertion orientations where the trajectories were concerned. The superiority of this algorithm is attested to in our previous work [21, 22].

3.1 Outline of GHRG-RRTs algorithm

The outline of the GHRG-RRTs algorithm is shown in Algorithm 1. Further details can be obtained in [22].

Algorithm 1: GHRG-RRTs(qinit, qgoal, Q).

1. \(T \leftarrow \text{InitTree}(q_{\text{init}}); P \leftarrow \text{InitPath}(q_{\text{init}})\).
2. if LinearCheck(q_{\text{init}}, q_{\text{goal}}, Q) then
3. \(U \leftarrow \text{SolveLine}(q_{\text{init}}, q_{\text{goal}})\) \quad \text{Get a linear path}
4. \(P \leftarrow \text{AchivePath}(T, U_{q_{\text{goal}}})\) \quad \text{Get a path}
5. end if
6. \(U \leftarrow \text{SolveCurve}(T.g_{\text{init}}, q_{\text{goal}})\) \quad \text{Get a curve path}
7. if \(U \not\mathcal{R} r_{\text{cons}} \& \text{CollisionFree}(U, Q)\) then
8. \(P \leftarrow \text{AchivePath}(T, U_{q_{\text{goal}}})\) \quad \text{Get a path}
9. end if
10. while (\(r < \text{max\_path}\)) \& (\(i < \text{max\_iteration}\)) do
11. \(q_{\text{goal}} \leftarrow \text{RandomNode}(); \text{flag} \leftarrow \text{false}\) \quad \text{Generate a random point}
12. if LinearCheck(q_{\text{goal}}, q_{\text{goal}}, Q) then
13. \(U \leftarrow \text{SolveLine}(q_{\text{init}}, q_{\text{goal}})\) \quad \text{Get a linear tree}
14. \(T \leftarrow \text{ExtendTree}(T, U_{q_{\text{goal}}})\) \quad \text{Extend a tree}
15. \(U \leftarrow \text{SolveCurve}(T.g_{\text{goal}}, q_{\text{goal}})\) \quad \text{Get a curve tree}
16. if \(U \not\mathcal{R} r_{\text{cons}} \& \text{CollisionFree}(U, Q)\) then
17. \(P \leftarrow \text{GetPath}(T, U_{q_{\text{goal}}}); \text{flag} \leftarrow \text{true}\) \quad \text{Get a path}
18. end if
19. end if
20. \(U \leftarrow \text{SolveCurve}(T.g_{\text{goal}}, q_{\text{goal}})\) \quad \text{Get a curve tree}
21. if \(U \not\mathcal{R} r_{\text{cons}} \& \text{CollisionFree}(U, Q)\) then
22. \(T \leftarrow \text{ExtendTree}(T, U_{q_{\text{goal}}})\) \quad \text{Extend a tree}
23. end if
24. \(U \leftarrow \text{SolveCurve}(T.g_{\text{goal}}, q_{\text{goal}})\) \quad \text{Get a path}
25. if \(U \not\mathcal{R} r_{\text{cons}} \& \text{CollisionFree}(U, Q)\) then
26. \(P \leftarrow \text{GetPath}(T, U_{q_{\text{goal}}}); \text{flag} \leftarrow \text{true}\) \quad \text{Get a path}
27. end if
28. if \(\text{flag} = \text{false}\) then
29. \(q_{\text{proper}} \leftarrow \text{FindProperNode}(T, U_{q_{\text{goal}}}, p)\)
30. \(U \leftarrow \text{SolveCurve}(T.q_{\text{proper}}, q_{\text{goal}})\)
31. if \(U \not\mathcal{R} r_{\text{cons}} \& \text{CollisionFree}(U, Q)\) then
32. \(T \leftarrow \text{ExtendTree}(T, U_{q_{\text{goal}}})\)
33. \(U \leftarrow \text{SolveCurve}(T.g_{\text{goal}}, q_{\text{goal}})\)
34. if \(U \not\mathcal{R} r_{\text{cons}} \& \text{CollisionFree}(U, Q)\) then
35. \(P \leftarrow \text{GetPath}(T, U_{q_{\text{goal}}})\)
36. end if
37. end if
38. end while
39. \(p_{\text{opt}} \leftarrow \text{Optimization}(P)\)
40. return \(p_{\text{opt}}\) \quad \text{Local optimization}
41. GetPath(T, U_{q_{\text{goal}}})
42. \(1: g \leftarrow \text{GetConfig}(U, q)\)
43. 2: \(T.\text{add\_vertex}(q)\)
44. 3: \(T.\text{add\_edge}(q, g)\)
45. 4: return \(T\)

3.2 Optimization function

Among the candidate solutions, all of which have already met the required constraints, the optimal trajectory can be chosen based on a cost function. The objectives of the optimization take consideration of minimizing tissue trauma and the danger of the path, as well as the path shape evaluation.

\[
\min F(u, T) = \min[\alpha L + \alpha D + \alpha_3 S]
\]

where \(L\) is the length of path \(L = \sum_{i=1}^{N} L_i\), \(D\) is the degree of danger in the path, \(D = \min(D_i)\), where \(D_i\) is the distance between the path and the \(i\)th obstacle, \(S\) is the curve evaluation of the path and \(S = \sum_{i=1}^{N} r_{\text{min}} q^{\alpha_3}\); \(\alpha_1, \alpha_2, \alpha_3\) are the weighted coefficients. The result of the function is the comprehensive evaluation of the optimal path.

4. Dynamic Motion Planning Algorithm

Prior to surgery, we first used the GHRG-RRTs in preoperative planning to seek out an optimal path based on...
medical imaging (e.g., ultrasound, CT or MRI). During the surgery, we used online medical imaging to update the configurations of both the needle and the environment. We then used a fast intraoperative replanning algorithm to reform and adjust the path in order to realize a real-time closed-loop control up to the point where the needle reaches the target.

We acquired the optimal path using the preoperative motion planning algorithm. However, disturbances and uncertainties like model inaccuracy, tissue deformation and inhomogeneity, needle tip positioning errors, physiological movement, etc., could still deviate the needle from its intended path. Moreover, movement on behalf of the target and obstacles can render the planned path infeasible, leading to a collision with obstacles or misplacement of the target. To overcome this, we adopted the replanning idea [17], here referred to as conventional motion replanning (CMR), the outline of which is depicted in Algorithm 2. We attained the current state by using a 3D medical imaging system for replanning and execution in each cycle, in order to systematically rectify the path until the needle tip reaches target region $\delta$.

![Image](image.png)

**Figure 6. Problems of the CMR**

This algorithm appears reasonable; however, it generally encounters some intractable problems. Firstly, this algorithm makes each cycle of replanning independent and without learning from former cycles, because of the stochastic property of the RRT algorithm, the blind replanning may cause a large detour in the path when the environment changes (see Fig.6a). Secondly, it may cause the convergence problem, i.e., as the flexibility of the needle is minimized due to nonholonomic constraints, whenever the needle gets close to the target, it is likely that the target will be unreachable because of its movement or other uncertain errors. Furthermore, the replanning algorithm will replan a circle-like path to make the target reachable again and as a result, the needle will detour in a loop without ever reaching the target (see Fig.6b). Therefore, the CMR algorithm is not suitable in terms of the significant movement of obstacles and the target.

To overcome these problems, we propose an improved motion replanning (IMR) algorithm with two strategies, i.e., an old point tracking strategy (OPTS) and an extreme trend extension strategy (ETES). OPTS tracks the old point in the old path in order to learn from former planning in order to overcome the large detour problem. ETES bends in an extreme manner when the needle gets close enough to a target that is unreachable to overcome the convergence problem. The programming of the IMR-1 is depicted in Algorithm 3.

OPTS (line 3-6 of IMR-1) tracks and utilizes the old connecting points $q_i$ in the former path to replan a new path. If it cannot generate a new path using the old points, it will replan a new path by using a modified GHRG-RRTs. In this way, after several cycles of iteration, it will find a relatively stable path that is immune to changes in the environment to some extent, because it has learned from previous planning. This strategy overcomes the large detour problem.

where the modified GHRG-RRTs is the routine of lines 6-40 of Algorithm 1. Once the needle has been inserted into the tissue, the orientation at the needle tip can no longer be changed. Therefore, the linear path acquisition routine (lines 1-5 of Algorithm 1) are no longer appropriate for replanning.
will try its best to extend the needle to the target with the minimum radius until it gets to the closest position to the target. We therefore formulate IMR-2, as depicted in Algorithm 4.

10mm), which is intolerant for clinical surgery. To overcome this, it may cause a large error (sometimes above the specified area). If the needle gets close enough to a target that is unreachable (the radius of the last segment $r_{\text{max}}$), instead of replanning a new path, it will try its best to extend the needle to the target with the minimum radius until it gets to the closest position to the target. This strategy overcomes the reachability problem.

We performed different strategies according to different conditions. Here, $s_1$ is the switch between the proposed replanning strategies and $\Delta s$ is the length of the insertion we executed in each cycle. If the path was longer than $s_1$, the OPTS was performed; if the length of the path was between $s_1$ and $\Delta s$, the ETES was performed up to the point where the needle reaches the nearest position to the target. Although the problems noted above were well solved, the ETES introduces another problem to the planning. Since we force the needle to extend to the target in an extreme manner, it may cause a large error (sometimes above 10mm), which is intolerant for clinical surgery. To overcome the complication, we propose the hardest goal tracking strategy (HGTTS) and integrated it to the IMR-1 to form IMR-2, as depicted in Algorithm 4.

We found that the failure of reachability was badly influenced by movement of the target. Among all positions that the target has undergone, there is the hardest one to reach, which is contained in the path with the smallest radius. We called this the hardest goal. Theoretically, if the needle can achieve the hardest goal, it will be less difficult to achieve other positions linked to the goal. We therefore used the HGTTS (lines 7-18 of IMR-2) to track the hardest goal, regardless of the target movement, until the length of path was shorter than $s_2$.

ETES (lines 14-21 of IMR-1) prevents the planner from generating a circle-like path due to the failure of reachability. If the needle gets close enough to a target that is unreachable (the radius of the last segment $r$ is less than the minimum radius $r_{\text{min}}$), instead of replanning a new path, it will try its best to extend the needle to the target with the minimum radius until it gets to the closest position to the target. This strategy overcomes the reachability problem.

We performed different strategies according to different conditions. Here, $s_1$ is the switch between the proposed replanning strategies and $\Delta s$ is the length of the insertion we executed in each cycle. If the path was longer than $s_1$, the OPTS was performed; if the length of the path was between $s_1$ and $\Delta s$, the ETES was performed up to the point where the needle reaches the nearest position to the target. Although the problems noted above were well solved, the ETES introduces another problem to the planning. Since we force the needle to extend to the target in an extreme manner, it may cause a large error (sometimes above 10mm), which is intolerant for clinical surgery. To overcome the complication, we propose the hardest goal tracking strategy (HGTTS) and integrated it to the IMR-1 to form IMR-2, as depicted in Algorithm 4.

We found that the failure of reachability was badly influenced by movement of the target. Among all positions that the target has undergone, there is the hardest one to reach, which is contained in the path with the smallest radius. We called this the hardest goal. Theoretically, if the needle can achieve the hardest goal, it will be less difficult to achieve other positions linked to the goal. We therefore used the HGTTS (lines 7-18 of IMR-2) to track the hardest goal, regardless of the target movement, until the length of path was shorter than $s_2$.

5. Simulation and Discussion

5.1 Settings for simulation

We simulated the motion planner in MATLAB® (ver. 7.8.0, R2009a; MathWorks, Natick, MA) on a 2.5 GHz 4-core Intel® i5™ PC. We adopted an environment similar to [14], modelling it as a cubical region at 200mm along each axis. Six spherical obstacles, each with a radius of 20mm represent the pubic arch, the urethra and the penile bulb around the prostate. The goal is set to (0 0 195), in millimetre (see Fig.8a). We set the minimum radius $r_{\text{max}}=50mm$ and the specific metric at $p=10mm$ (in line 29 of Algorithm 1). The maximum number of the candidate paths was set to 100 and the maximum number of iterations to 10 000. In order to speed up computation, we assumed there was a relatively safe margin $m$ (here we set $m=3mm$) around the obstacle; as long as the needle did not puncture the safe margin, it would never puncture the obstacle; thus, the surgery was safe. The radius of the obstacle should also be enlarged by safe margin $m$ when planning to make it safe. As such, there was no need to care about the exact distance between the needle and the obstacles, as long as the surgery was sufficiently safe. Therefore, we could disregard the second term in (7) by setting $\alpha=0$. Other weighted coefficients were set to $\alpha_1=\alpha_2=1$, as we equally considered both of the remaining sub-objective functions. These settings were

![Algorithm 3 IMR-1](image)

![Algorithm 4 IMR-2](image)
applied to both preoperative planning and intraoperative replanning.

For replanning, we added some disturbances and movements. The disturbances such as model inaccuracy, tissue deformation, tissue inhomogeneity and needle tip positioning errors, were modelled as white noise and followed normal distribution \(N(\mu, \sigma)\). We let \(\mu=r\) and \(\sigma=10\) mm to the radii of the path, \(\mu=0\) and \(\sigma=1\) mm to the tip position and \(\mu=0\) and \(\sigma=0.01\) rad to the orientation of the needle tip. Movement of the target and obstacles were modelled as periodic sinusoidal motions in 3D, both with an amplitude of 5mm and at a period of 60s and 5s, respectively.

As noted, since \(s_i(i=1,2)\) are the switch for the replanning strategies and affect the validity of the planning, we need to discuss them in detail. For IMR-1, \(s_1\) is the switch between OPTS and ETES. Here we will track the old connecting points along the planned path. The convergence problem always occurs in the final segment, especially close to the target. We thus have to place \(s_1\) before the convergence problem occurs in the final segment. In order to attain switch \(s_1\), we have to study where the convergence problem generally occurs and how long the final segment of path is. We simulated for 20 times of the Algorithm 2 (CMR) in the specified environment. Among the 20 trials, the maximum length of path occurring in the convergence problem was 30.73 mm; the minimum length of the final segment was 92.01 mm. Thus, \(s_1\) should be between 30.73 mm and 92.01 mm; to be safe, we let \(s_1=40-80\) mm. Moreover, the convergence problem is also related to the environment and the value of minimum radius \(r_{\text{min}}\).

As for IMR-2, \(s_2\) is the switch between the first two strategies (OPTS and HGTS) and the last one (ETES). The analysis for IMR-2 is similar to that of IMR-1. We will perform the first two strategies until the needle gets close to the target and then perform the final strategy. However, the difference from the IMR-1 is that we do not have to worry about the convergence problem prior to releasing the HGTS, since the hardest goal is fixed. The critical matter is when to release the HGTS. If we release it too early, it will track the target thereafter and the effectiveness will be no better than for IMR-1; if we release it too late, the planning will have limited opportunity to adjust the path to the current target.

Thus, the value of \(s_2\) for IMR-2 is neither too large nor too small. In order to attain switch \(s_2\) for IMR-2, we simulated each different value 20 times to observe its performance, as is shown in Fig.7. From the figure, we can conclude that the best performance for IMR-2 is at \(s_2=10\). Moreover, \(s_2\) not only concerns the environment and \(r_{\text{min}}\) but also target movement.

Therefore, empirically, we set \(s_1=50\) mm and \(s_2=10\) mm for IMR-1 and IMR-2, respectively. We also set the insertion distance per cycle as \(\Delta s = 1\) mm.

### Table 1. Comparison of the three algorithms

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Large detour problem</th>
<th>Convergence problem</th>
<th>Error (mm)</th>
<th>Time for each cycle (ms)</th>
<th>Length of optimal path (mm)</th>
<th>Length of actual path (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMR</td>
<td>18</td>
<td>16</td>
<td>/</td>
<td>202.6±101.1</td>
<td>/</td>
<td>/</td>
</tr>
<tr>
<td>IMR-1</td>
<td>0</td>
<td>0</td>
<td>7.77±4.15</td>
<td>1.6±11.4</td>
<td>214.5±5.6</td>
<td>228.6±18.2</td>
</tr>
<tr>
<td>IMR-2</td>
<td>0</td>
<td>0</td>
<td>1.3±0.38</td>
<td>3.7±21.6</td>
<td>219.8±4.7</td>
<td></td>
</tr>
</tbody>
</table>

Figure 7. Average errors of different values of \(s_2\)

### 5.2 Simulation results and discussion

In order to verify the validity of each strategy, we compared Algorithm 2 (CMR), Algorithm 3 (IMR-1) and Algorithm 4 (IMR-2). We performed 20 trials using the same target for each algorithm. The validity of the OPTS and ETES is obvious in the comparison of CMR and IMR-1, as shown in Table 1. The validity of the HGTS is shown in the comparison of IMR-1 and IMR-2 (see Table 1). Table 1 shows that some of the results in the 20 trials have the formation “mean ± standard deviation”.

The results show that CMR suffers from the large detour problem, while the other algorithms avoid it. This reveals that OPTS is effective in terms of the large detour problem. CMR also suffers the convergence problem, while the other algorithms avoid this, which reveals the ETES is effective in terms of the convergence problem. Moreover, the time for each cycle for CMR is much longer than that for the other algorithms. This is because OPTS, which not only learns from former cycles of replanning and renders the path stable, but also enhances the cycling replanning speed.

On the other hand, from the comparison of IMR-1 and IMR-2, it is clear that the accuracy of IMR-2 is much better than that of IMR-1, which reveals the effectiveness of the HGTS. The error of IMR-2 is within 2mm, which meets the needs of clinical requirements. From the length of the actual path, we can see that IMR-2 is much more stable than
IMR-1. Although the replanning speed of IMR-2 is slightly slower than that of IMR-1 due to adding the extra strategy of HGTS, it is still in milliseconds (see the column “Time for each cycle” in Table 1), which is fast enough for intraoperative replanning.

We also obtained one of the simulation results and compared it with the result of the simulation without the replanning. The course of simulation is shown in Fig. 8. The original optimal path, the executed path, the ongoing/adjusted path and the path added with disturbances and without replanning or adjusting are depicted in the figure. The final errors were 0.75 mm and 43.15 mm for the replanning and without replanning paths, respectively.

In order to verify the robustness of the proposed algorithm (IMR-2), we also performed simulation for different goals. We randomly set two more goals as G1 (5 20 190) and G2 (10 -20 185), both in millimetre, and simulated each goal 20 times. The results are as shown in Table 2 and have the formation “mean ± standard deviation”.

From Table 2, we can see that the large detour or convergence problem did not occur, the error is within 2mm and speed remains high. As such, the conclusion can be drawn that the proposed algorithm has strong robustness.

### 6. Conclusion

In this work, we firstly calculated the kinematics for the cannula flexible needle. Based on the proposed RGHG-RRTs algorithm for static motion planning in our previous work, we propose a 3D dynamic motion planning algorithm for intraoperative surgery. We also propose three novel strategies to be integrated in the conventional

<table>
<thead>
<tr>
<th>IMR-2</th>
<th>Large detour problem (times)</th>
<th>Convergence problem (times)</th>
<th>Error (mm)</th>
<th>Time for each cycle (ms)</th>
<th>Length of optimal path (mm)</th>
<th>Length of actual path (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>0</td>
<td>0</td>
<td>1.56±0.53</td>
<td>2.5±18.3</td>
<td>204.9±6.1</td>
<td>208.2±8.7</td>
</tr>
<tr>
<td>G2</td>
<td>0</td>
<td>0</td>
<td>1.47±0.37</td>
<td>3.3±22.1</td>
<td>197.1±5.2</td>
<td>199.0±4.2</td>
</tr>
</tbody>
</table>

Table 2. The performance of the IMR-2 algorithm
replanning algorithm in order to improve it. By comparing the CMR, the IMR-1 and the IMR-2 algorithms, we can conclude that the proposed OPTS, ETES and HGTS are extremely effective for addressing the large detour problem, the convergence problem and accuracy problem, respectively. Moreover, the OPTS also benefits for the replanning speed of a cycle. Finally, we performed simulations for different goals. The results revealed the validity and robustness of the proposed replanning algorithm.

In future, we will integrate our planning with a real-time feedback controller to carry out experiments on the phantom tissue.

7. Acknowledgements

This research is supported in part by the National Natural Science Foundation of China (Grant#51305107), by the Natural Science Foundation of Heilongjiang Province of China (Grant#F20150509 and #2014448) and by the Science and Technology Project of the Education Department of Heilongjiang Province of China (Grant#12531110 and #12531122).

8. References


