Proceedings of the 10th World Congress on Electrical Engineering and Computer Systems and Sciences (EECSS'24)
Barcelona, Spain - August 19 - 21, 2024
Paper No. CIST 157
DOI: 10.11159/cist24.157

Path Planning with RRT*M Algorithm in Simulated Human Respiratory Environment

Yuhao Huang[1], Xiwen Fan[1], Kunpeng Wang[1], Zheng Yang[1], Sai Cheong Fok[1]
[1]Sichuan University-Pittsburgh Institute
Sichuan University, Chengdu, China
2020141520159@stu.scu.edu.cn; fanxiwen@stu.scu.edu.cn; kunpeng.wang@scu.edu.cn; zhengyang2018@scu.edu.cn; saicheong.fok@scupi.cn

Abstract - Image-guided percutaneous insertion is widely applied in lung biopsy surgeries. Traditional procedure used rigid needle, which may lead to post-operation complications such as pneumothorax and hemothorax. Recent research have investigated the application of curvature-controllable steerable bevel-tip needle with pre-surgery path planning to overcome this problem. This work focuses on improving the RRT* algorithm (in terms of path length, search time and redundant random nodes) for the pre-operation path planning based on a constrained search environment. An RRT*M algorithm is proposed via probability distribution of the generation of random nodes around the ideal path connecting the start and goal points within a constraint region while avoiding the obstacles in the simulated human respiratory system. The performance of the proposed algorithm is compared with informed RRT* algorithm based on a similar 2D environment. The result indicated the potential of the proposed algorithm for lung biopsy path planning in a 3D virtual environment based on the human anatomy.

***Keywords*:** RRT*, RRT*M, Lung biopsy, Steerable needle, Moving boundary, Constraint search region, Normal distribution

1. Introduction

In the aftermath of the COVID-19 pandemic, diseases associated with the human lung and respiratory systems have attracted much attention. Among lung diseases, cancer had the highest incidence and mortality rate [1]. The key to improve the survival rate of cancer patients is screening and early diagnosis of the disease. In traditional clinical settings, CT-guided lung biopsy is commonly used to extract the tumour for malignant diagnosis. The process uses a percutaneous needle and is noted for its accuracy which normally ranges between 90.0% to 96.7% [2]. This technique is the gold standard for the early detection of lung cancer and the procedure is currently performed manually. Suggestions to improve the procedure include the use of flexible needles to avoid unnecessary damage to surrounding tissues [3] and the use of robots. These improvements require pre-surgery path planning of the bevel-tip needle in human tissues.

The RRT* algorithm is one of the more popular approaches used in path planning. However, the classical RRT* algorithm still have many areas that can be improved, including 1) the generation of many redundant nodes during the search for a feasible path through obstacles; 2) the selection of relevant random nodes to generate a valid path that will not pass through obstacles. These issues will increase the computation time. Solutions to shorten the search time may compromise the path length and planning success rates. Many enhancements to the RRT* algorithm [4-6] have been proposed in recent years, aiming to optimize the search performance with higher computational efficiency. These improvements revolve around producing more centralized random nodes and reducing random redundancy via more centralized distributions in a constraint region of the global environment.

This paper presents an improved path planner, RRT*M, for lung biopsy. RRT*M algorithm provides feasible paths in the simulated human respiratory environment with shorter length, faster convergence and less redundant points via employing a normally distributed random generator and producing the random nodes within a preselected rectangular region admitting feasible paths. Notably, the constrained region can be selected by medical doctors in real clinical pre-surgery procedure. The algorithm aims to improve the search efficiency to find the shortest path. The performance of RRT*M algorithm is evaluated via the comparisons with the existing RRT* [7] in the simulated human respiratory 2D environment generated from CT scans, in which irregular shapes of bones are used as obstacles. Moreover, the compassion with informed

RRT* is conducted in the environment presented in [4]. The result shows that RRT*M algorithm is comparatively more superior in terms of path length, search time and redundant points.

2. The RRT*M algorithm

The original RRT* algorithm has already been modified for the lung biopsy in our previous work [7]. The RRT*M is named with M since the boundary of the constraint search region can move along with the generation of random nodes towards the goal points. In general, the RRT*M includes the three main changes:

(1) A rectangular constraint region is introduced to refine the search and reduce the redundancy. Notably, we assume that the employed constraint region includes a feasible path.

(2) The planner employs a normally distributed generator such that the random nodes will distribute more compactly around the ideal straight path connecting the start and goal points.

(3) The boundary of the constraint region can move along with the generation of the random nodes towards the goal point. The backwards search of the random points is avoided in this treatment and the planner only samples the nodes towards to the goal point to assist the growth of the feasible path.

2.1. Obstacles in lung biopsy

In lung biopsy, the main obstacle that needs to be considered are the bones in the upper body, which include the rib cage and the spinal cord. Other obstacles such as major blood vessels can be included if needed.

Prior to the surgical operation, the patient must undertake a CT scan. The scan will give a 3D view of the anatomy of the upper body. The locations and geometries of the bones were obtained from the CT scan. It is common to separate the 3D view into three 2D views like those used in engineering drawings. These are the top view, front view, and side view. Figures 1 and 2 show the bone (i.e. obstacles) in the side view and front view respectively. The side view is normally used by the surgeon for path planning. In the side view, where only the cross section of the bone can be seen. Figure 1 shows that the cross section of the bone simplified with circles, which resembles ellipse due to the aspect ratio. The front and side views used by the surgeon contain the planes that pass through the tumour.

The surgeon plans for the manual biopsy operation by searching for a suitable path for the needle using the side view. The first step is to confirm a confined region where there exists at least one possible path for the needle to be inserted from outside the body to the target tumour within the lung. Due to the shape of the CT scan, the constraint search is mostly rectangular in shape and oriented at an angle. Also, the rectangle is more computationally efficient than other shapes. The RRT*M algorithm will be described using the side view assuming that the surgeon has specified this constraint search region.

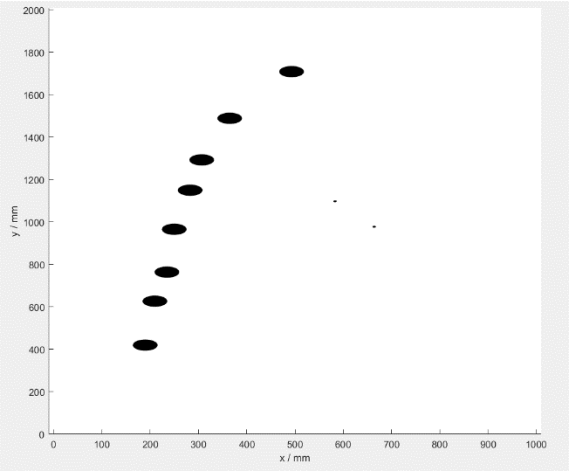


Fig. 1: Side view of the bone in the CT scan.

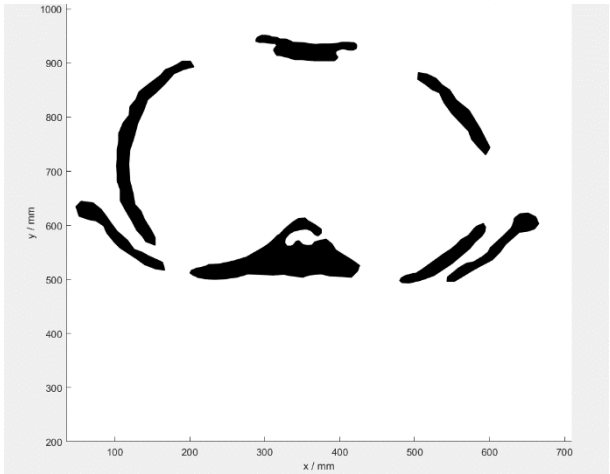


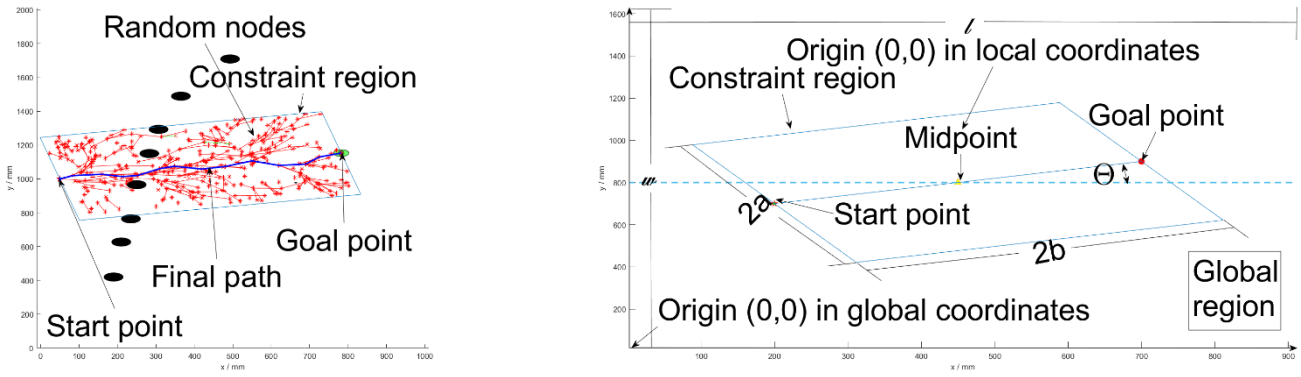
Fig. 2: Top view of bone in the CT scan.

After the constraint search area has been identified, the goal of the path planning algorithm is to search for an optimal path to navigate the obstacles within the constraint region. The most common criterion for optimization is the path length. The efficiency of the search algorithm can be measured by the time needed to find a valid path. The RRT*M algorithm aims

to find the shortest path with a higher search efficiency. The modification will be examined in three aspects of improvement: search time, path length and the number of redundant nodes under certain conditions.

2.2. Analysis of the rectangular constraint region

In [4], Gammel et al. employed an elliptical constraint search region to reduce the search time, path length and redundant nodes. However, the constraint search region specified by the surgeon in the 2D side view is mostly rectangular



(a) (b)

Fig. 3: Illustration of the RRT*M in the side view. (a) An example of RRT*M in the side view. (b) The principle of the constraint region in RRT*M.

in shape and oriented at an angle θ. The use of the rectangular constraint search region is investigated in our algorithm.

Figure 3 shows the constraint search region within the global search region. The global coordinate is illustrated in Figure 3(a) and will be denoted by the capital letter (X, Y). The global coordinates are provided by the CT image processing software. The local coordinate system (denoted by (x, y)) is used for the constraint search region. The start and goal points of the needle is shown in Figure 3(b). The midpoint of the line joining the start and goal points is defined as the origin of the local coordinate system. The lower left corner of the constraint region is used as a referenced point for the transformation between the local and global coordinate systems. The random points (x_{random}, y_{random}) are generated inside the constraint rectangle region with respect to the origin of the local coordinate system using:

$$x_{random} = rand * 2a + X - a, \tag{1a}$$
$$y_{random} = rand * 2b + Y - b, \tag{1b}$$

where $2a, 2b$ are the length and width of the constraint rectangle region, respectively, $rand$ stands for the uniform random number generator, X, Y are the global coordinates of the origin of the constrained region. Note the inclined angle θ is defined as

$$\theta = arctan\ arctan\ \left(\frac{Y - Y_{start}}{X - X_{start}}\right). \tag{2}$$

The global coordinate of the random points inside the constraint rectangle are obtained using:

$$[X_{random}\ Y_{random}] = [cos\ cos\ (\theta)\ \ -sin\ sin\ (\theta)\ \ sin\ sin\ (\theta)\ \ (\theta)\ ]\left[x_{random}\ y_{random}\right] + [X\ Y\ ]. \tag{3}$$

Figure 4 illustrates the generation of a valid node. The subfunction $modified\_formula\ (\chi, O)$ in Figure 4 employs the equation (3) to generate the global coordinate of the random points within the constraint region. The algorithm will then select the node in the search tree that is nearest to the random point and calculate the distance

between them. The nearest random point will become a new node if the distance is shorter than the prespecified step length S. Otherwise, X_{new} will be a point in the direction of X_{rand} such that the distance is S.

Algorithm 1: Generation of X_{new}

Input: χ, T, O, S
Output: X_{new}
1: $X_{rand} = modified\_formula\ (\chi, O)$
2: $X_{nearest} = seclect\_nearest\_node\ (X_{rand}, T)$
3: if $get\_distance\_between\ (X_{nearest}, X_{rand}) \leq S$
4: $X_{new} = X_{rand}$;
5: else
6: $X_{new} = extend\ (X_{rand}, X_{nearest}, S)$
7: end if
8: Return X_{new}

Fig. 4: The pseudo algorithm for the random node generation in RRT*M.

2.3. Normally distributed search nodes

Instead of using the uniform random generator for both the *x* and *y* coordinates, the normal distribution can make the random nodes more compact and closely distributed around the assumed ideal straight line connecting the start and goal points. This involves the generation of the random nodes as follow:

$$x_{random} = rand * l, \tag{4a}$$
$$y_{random} = normrnd\left(\frac{1}{2}, \sigma\right) * w, \tag{4b}$$

where $rand$ stands for a random number from 0 to 1, $normrnd\left(\frac{1}{2}, \sigma\right)$ stands for a random number following normal distribution, of which the mean equals $\frac{1}{2}$, deviation equals σ. A mean value of $\frac{1}{2}$ is imposed such that the random nodes will distribute around the line $y = \frac{1}{2}w$, which equally divides the global search region in the y-direction. Afterwards, (x_{random}, y_{random}) are substituted into the equation (3) to obtain the global coordinate of random nodes. The value of σ can be adjusted based on the complexity and location of the obstacles. In an environment with less considered obstacles, a small σ (for example, $\sigma = 0.25$) can be assigned. In an environment with more obstacles, a larger value of σ is expected so that the random points are spread over wider region to facilitate the search.

The normally distributed search nodes can be generated by combining equations (1) and (4):

$$x_{random} = rand * 2a + X - a, \tag{5a}$$
$$y_{random} = sin\ sin\left(normrnd(0, \sigma)\right) * b + Y, \tag{5b}$$

where $2a, 2b$ are the length and width of the constraint search area respectively, $sin(normrnd(0, \sigma))$ is employed to restrict the random number within $[-1,1]$.

2.4 Dynamic search boundaries

As the feasible path extends towards the target point, it may be more cost effective to focus the search on the target. This concept of forward searching is explored by a dynamic constraint search region, i.e., the search region will become smaller and move towards the target. Figure 5 illustrates the concept of the moving boundary, where the left boundary keeps approaching as the path develops towards the target. This forward generation of random nodes can reduce the computation power of cost comparisons of line 3 in Figure 4.

The local coordinate of the random nodes can be determined by:

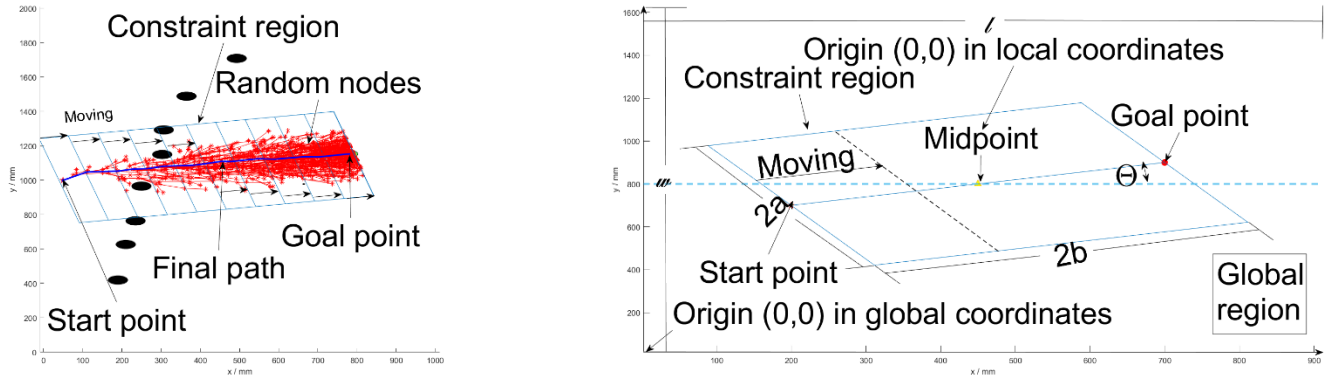
$$x_{rand} = rand * 2a + x - a + iter * S, \tag{6a}$$
$$y_{rand} = sin\ sin\ \left(normrnd(0, \sigma)\right)\ * b + y, \tag{6b}$$

where $iter$ represents the time that random nodes have been generated, S is the step length. The global coordinate can be obtained via substituting (x_{rand}, y_{rand}) into the equation (3).

The step length function S can be further investigated according to the search environment. In our experiments, the moving boundary is designed to stop at the line $x = X_m$ in local coordinates. The step length is then defined by the following formula

$$S = \frac{X_m - (X - a)}{n},$$

where n denotes the number of the new nodes X_{new}, X is the midpoint, a is one half of the width of the constraint region and X_m is the goal position of the moving boundary. Note S can be further investigated to facilitate a non-constant speed moving boundary based on the environment.



(a) (b)

Fig. 5: Illustration of RRT*M with moving boundaries. (a) An example of the moving boundary. (b) The principle of the moving boundary treatment.

3. Results and comparisons

All path planning were executed on a 2.60GHz Inter core i7-10750H processor with 16GB RAM.

3.1 Path planning in the side-view and top view environments

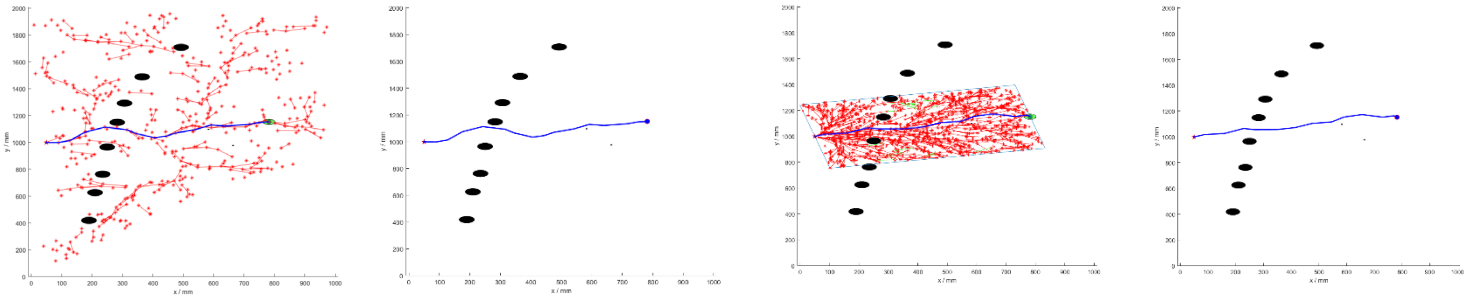
The performances RRT*M algorithm with and without moving boundaries were first investigated and compared with the RRT* in [7] using realistic obstacles in the human anatomy. The global configuration environment was obtained from the CT scans of a human subject and segmented into side and top views with obstacles shown in Figures 1 and 2. The start and goal points in both views are listed in Table 1.

Table 1: Start and goal settings for simulations.

Environment	Start point	Goal point
Side view	(50,1000)	(782,1153)
Top view	(300,300)	(400,700)

The investigation first used the side view. The modified RRT* presented in [7], and RRT*-M with and without moving boundaries, were each executed 100 times with different numbers of total random nodes x_{rand} (denoted by n varying between 300 and 2000). For each algorithm, the shortest search time to find the first valid path in the 100 trials was recorded. The average path length (without smoothing) generated for all 100 trails was determined. Let k denote the points on the feasible path. Then the number of redundant points is $n - k$. The required nodes to facilitate a feasible path are also recorded.

The process was then repeated using the top view. The results of the average value of the 100 trials with $n = 2000$ are summarized in Table 2. The findings indicated that the RRT*-M algorithm outperform the RRT* in both search environments in terms of search time, path length, redundant points, and valid nodes for tree branching. Notably, the RRT*-M algorithm



with moving boundaries showed good performance in the top view environment. With limited number of random nodes, RRT* algorithm in [7] cannot always converge, whereas the success rate of RRT*-M algorithm is around 99%. Comparison of the side and top views indicated that the RRT*-M with moving boundaries can result in slightly longer search time. Hence the effectiveness of the moving boundaries could depend on the obstacles and this will need further investigation.

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

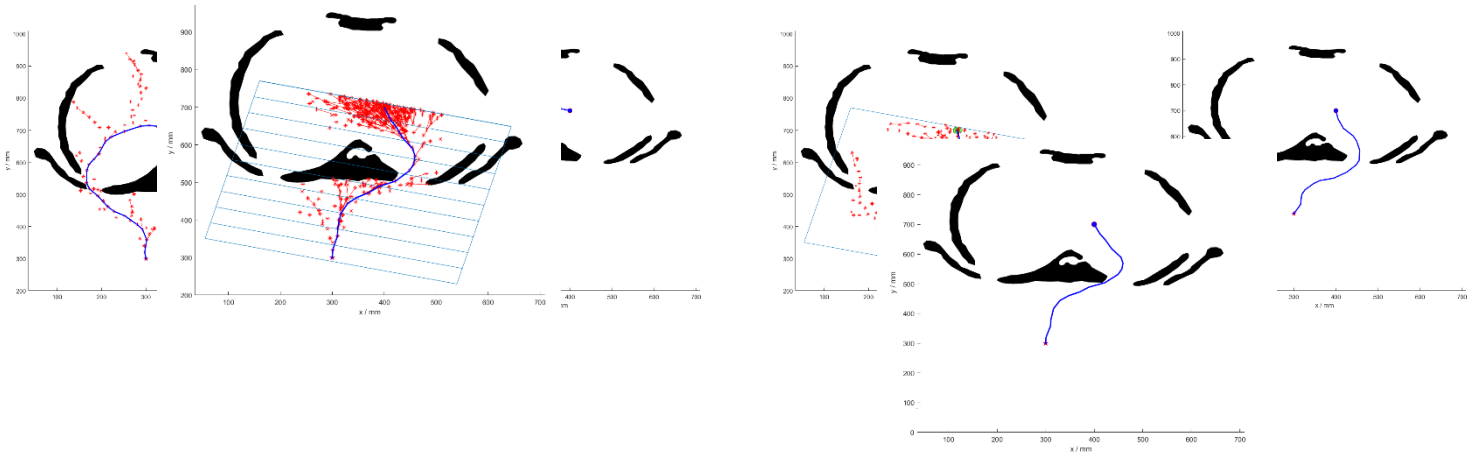
(e) (f)

Fig. 6: Comparison between RRT* and RRT*M in the side view. (a) Search tree of RRT*. (b) Path of RRT*. (c) Search tree of RRT*M without moving boundary. (d) Path of RRT*M without moving boundary. (e) Search tree of RRT*M with moving boundary. (f) Path of RRT*M with moving boundary.

Table 2: Comparisons of results RRT*M with and without moving boundaries with RRT* [7].

	Algorithm	Search time (s)	Path length (mm)	Redundant points	Required X_{new}
Side view	RRT*	2.11	994.50	430.34	957.16
	RRT*M (without moving boundary)	0.17	787.51	38.51	125.23
	RRT*M (with moving boundary)	0.29	788.64	34.86	206.02
Top view	RRT*	0.72	531.10	919.23	1048.77

	RRT*M (without moving boundary)	1.12	475.75	632.17	861.88
	RRT*M (with moving boundary)	0.87	474.32	549.35	723.94



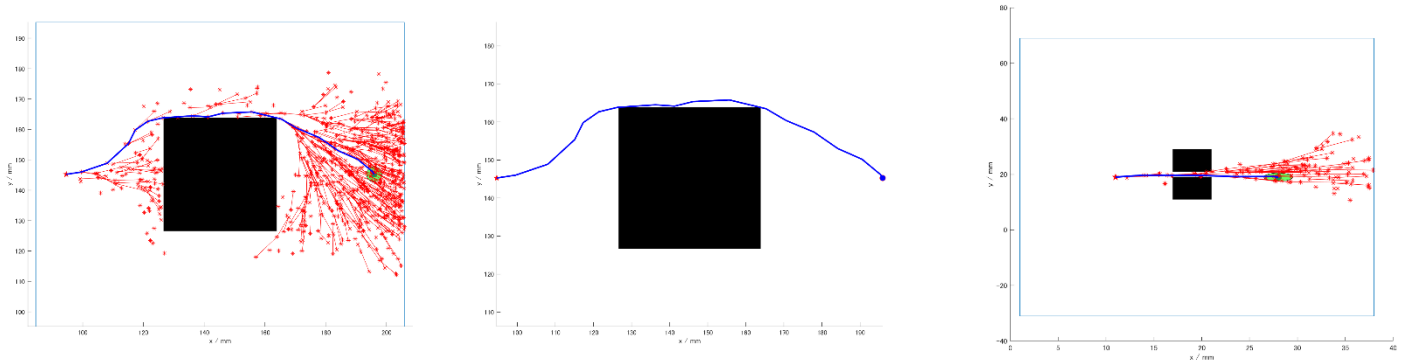
(a) (b) (c) (d)

(e) (f)

Fig. 7: Comparison between RRT* and RRT*M in the top view. (a) Search tree of RRT*. (b) Path of RRT*. (c) Search tree of RRT* without moving boundary. (d) Path of RRT*M without moving boundary. (e) Search tree of RRT*M with moving boundary. (f) Path of RRT*M with moving boundary.

3.2 Comparisons of RRT*M and informed RRT*[4]

The RRT*M with moving boundary was also compared with the informed RRT* algorithm using the environment with obstacles presented in [4]. The configuration space is set to $[0, 290.5] \times [0, 290.5]$ (mm). The start and goal point are $(94.59, 145.25)$ and $(195.94, 145.25)$. The obstacle in Figure 8(a)(b) is a square block with length $37.16mm$ centered at $(145.25, 145.25)$. Our algorithm was executed 100 times. The best path length obtained from informed RRT*M is 115.00 mm while the result of the informed RRT*M in [4] is 112.52 mm. Although RRT*M gave a slightly longer path, the search time is only $0.31s$, which is much shorter than $5s$ needed by informed RRT presented in [4]. The possible reason of the different path length may be due to the variation in the environment. With the Bezier curve smoothing (see more details in [7]), the final path can be refined to 112.58 mm with computation time of $2.17s$, which is still shorter than $5s$. The RRT*M algorithm is also tested with narrow opening environment in [4] and proves to generate a successful path (see Figure 8(c)).



(a) (b) (c)

Fig. 8: A result of RRT*M in the environment [4] with road-blocking squares. (a) Search trees of RRT*M. (b) Initial path of RRT*M. (c) An example of the RRT*M finding a successful path in the narrow opening environment [4].

4. Conclusion

In this paper, we proposed an RRT*M for pre-surgery lung biopsy path planning. The constraint environment is identified by the surgeon during pre-operation procedure. Based on defined constraint environment, the algorithm incorporated normal distribution probability function to better generate the random points and moving boundary to facilitate the forward search. The performances of the RRT*M algorithm were compared in the 2D simulated human respiratory environment with the RRT* algorithm of [7]. It was also compared with the informed RRT* algorithm in [4].

Results and comparisons suggested that the RRT*M is a promising method capable of producing a shorter path with faster convergence. The performance demonstrates the potential applications of the algorithm in real-time lung biopsy surgeries using realistic virtual environment resembling the human anatomy. Nevertheless, we find that RRT*M algorithm can be further improved. Potential improvements include the investigation of the speed of the moving boundary.

Acknowledgements

This work was supported by the Young Scientist Fund under Grant No.2022NSFSC1855 and the Key Research Program under Grant NO. 2023YFG001, both of the Provincial Natural Science Foundation of Sichuan Province.

References

[1] Q. Fu, Y. Zhou and Y. Zhang. “Lung cancer screening strategy for non-high risk individuals a narrative review.” *Translational Lung Cancer Research*, 2021,1(10):452-461.

[2] J. Zhong, F. Zhong, L.Tang, J. Zhang, H. Feng, J. Deng and He, L. “A Virtual Surgery System for Lung Biopsy,” in *Proceedings of the IEEE International Conference on Communication, Image and Signal Processing (CCISP)*, Chengdu, Sichuan, China, 2023, pp. 372-378.

[3] 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 *2008 IEEE International Conference on Automation Science and Engineering*, Arlington, VA, USA, 2008, pp. 41-46.

[4] A. H. Qureshi and Y. Ayaz. “Intelligent bidirectional rapidly-exploring random trees for optimal motion planning in complex cluttered environments,” *Robotics and Autonomous Systems*, vol. 68, 2015, pp. 1-11.

[5] D. Gammell, S. Siddhartha and B. Timothy. “Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Chicago,IL,USA, 2014, pp. 2997-3004.

[6] S. Karaman and E. Frazzoli, “Sampling-based algorithms for optimal motion planning,” IJRR,30(7):846–894,2011.

[7] Y. Dong., K. Wang, Z. Yang, S. C. Fok and H. Wang. “Path Planning with Modified RRT* Algorithm for Lung Biopsy,” in *Proceedings of the 8th World Congress on Mechanical, Chemical, and Material Engineering (MCM'22)*, Prague, Czech Republic, 2022, pp. 1-8.