



Sampling based path planning algorithm for UAV collision avoidance

A SARAVANAKUMAR*, A KAVIYARASU and R ASHLY JASMINE

Division of Avionics, Department of Aerospace Engineering, MIT Campus, Anna University, Chennai, India
e-mail: saravanakumar_a@yahoo.com; isrokavi@gmail.com; ashly1.1994@gmail.com

MS received 1 June 2020; revised 1 February 2021; accepted 13 April 2021

Abstract. Path planning of Unmanned Aerial Vehicles (UAVs) avoiding collisions with moving obstacles or other UAVs in motion, is one of the key functions to fulfill their mission. The current work is focused on the development of sampling-based path planning methods for UAV. Under this method, standard Rapidly exploring Random Tree algorithm (RRT) is chosen, but RRT algorithm faces some limitations. Thus few developments were made in RRT by simplifying the node connection strategy, to generate feasible path satisfying the operating environment constraints dictated. Simplified node connecting strategy as Modified RRT (MRRT) and collision avoidance using reachable sets is developed to avoid collisions along the path. It is demonstrated in the python window using Python software. The proposed algorithm can develop a path in a short time duration and guides the vehicle to bypass the obstacles to avoid collision.

Keywords. Path planning; obstacle avoidance; MRRT algorithm; reachable sets.

1. Introduction

Path planning of UAV focuses at automatically developing attainable paths from the predetermined start to the target point considering static and dynamic environmental constraints and uncertainties. Path planning in higher dimensions unlike 2-Dimensional environment has extensive potential, significantly becoming more complicated to understand [1]. Different UAV path planning algorithms and collision avoidance mechanisms have been proposed lately. These algorithms were sectioned into five segments as in [2] and contains different strategies which adapt to follow certain attributes. The subsections include sampling based path planning, node based optimal algorithm, mathematical model based algorithm, bio-inspired algorithm and multifusion based algorithm.

Lin *et al* presented the design and implementation of closed-loop RRT with modifications. Software and hardware experimental results demonstrated that the algorithm can continuously develop paths for UAVs, rid of obstacles of various numbers moving at different rates [3]. Liu *et al* proposed a hybrid crash evasion strategy based on fuzzy logic approach, alongside a modified course following methodology developed using RRT [4]. Hao *et al* presents a methodology for choosing the path with minimum distance, processed using grids on rasterized environment utilizing A* algorithm [5]. Mangal *et al* utilized a chance constraint to restrict the chances of limitation infringement to build up a real-time probabilistic path generator using rapidly exploring

random tree (RRT) algorithm to handle uncertain dynamic obstacles [6]. Jiawei *et al* developed an adaptable method to maximize the convergence rate in a cluttered 3D environment, by re-characterizing the search space depending on 2D Informed RRT* [7]. Be that as it may, these techniques do not computationally scale well to incorporate a number of obstacles, static as well as dynamic.

In this paper, sampling based methods to generate a feasible path for UAV collision avoidance with dynamic obstacle is proposed. Algorithms like RRT [8], RRT*[7], Probabilistic road map [9], Voronoi, visibility graphs, Artificial potential field [10] come under sampling based methods. Rapidly exploring random tree algorithm is chosen as it has advantages [2, 3] like capable to find a path attainable in brief time frame, ensures probabilistic fulfillment appropriate to various unique dynamic models, empowers trajectory-wise monitoring, so it does not require separate search algorithm and no explicit enumeration of constraints. But face some limitations such as: a) in the event that there exists a lot of obstacles in the state space, the possibility of finding a point that isn't lying in an obstacle region and the ability to link with the tree is little. b) A narrow path is also hard to follow. c) RRT is time consuming in case of large workspace and non-optimal with several unnecessary branches. To overcome these limitations, modification is been proposed by simplifying the node connection strategy of randomly generated nodes of standard RRT and has been coupled with reachable sets to detect the dynamic obstacles position for avoiding collision. The remainder of the paper is organized as problem statement, methods description, results and discussions.

*For correspondence
Published online: 07 June 2021

2. Problem statement

The path generated connecting the defined start (x_s, y_s, z_s) and goal (x_g, y_g, z_g) in the workspace must be collision free, is the main objective. Collision free means that no static or dynamic obstacles $(x_{oi}(t))$ come under the safety zone $(O(x(t)))$ of UAVs, at any instant of time. The path [11] is described as

$$f(x, y, z) = 0 \tag{1}$$

Where O_i is calculated by the constant velocity prediction model at t . The anticipated position of the obstacles is given by,

$$x_{oi}(t) = x_{oi}(t_o) + v_{oi}(t_o) * (t - t_o) \tag{2}$$

The shortest path length is derived by using the differential arc length along the paths,

$$f^* = \arg \min_f \int_f ds \tag{3}$$

The assumptions made are, a) obstacles move with constant velocity. b) positions, velocities of obstacles and UAV states are known to the path planner developed using the algorithm.

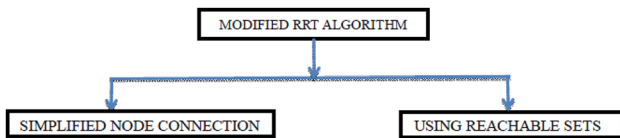


Figure 1. Modified RRT algorithm.

3. Algorithm description

3.1 Sampling-based path planning

These methods first randomly sample the workspace as a set of nodes, then haphazardly map the workspace to identify the best path. These methods have an uncomplicated framework, hence any modification can be made and they are more suitable for online planning conditions. Under sampling based methods RRT algorithm is chosen and it is modified to overcome limitations faced by basic RRT algorithm.

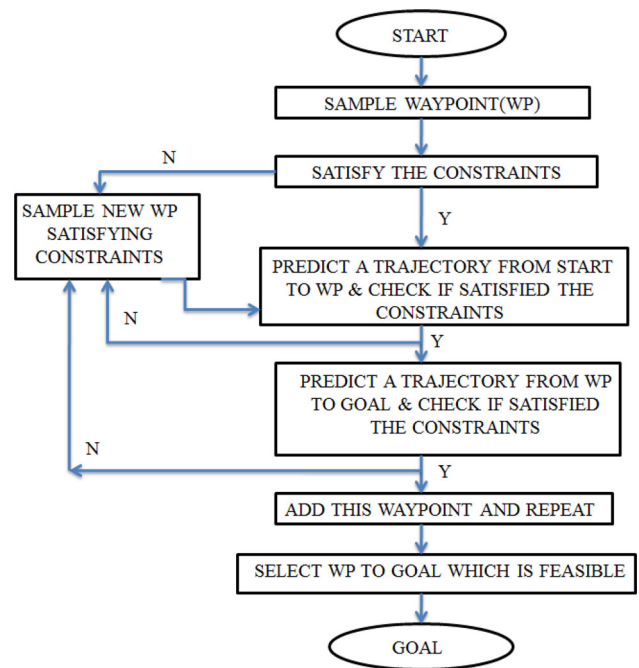


Figure 3. Flowchart of simplified node connection strategy.

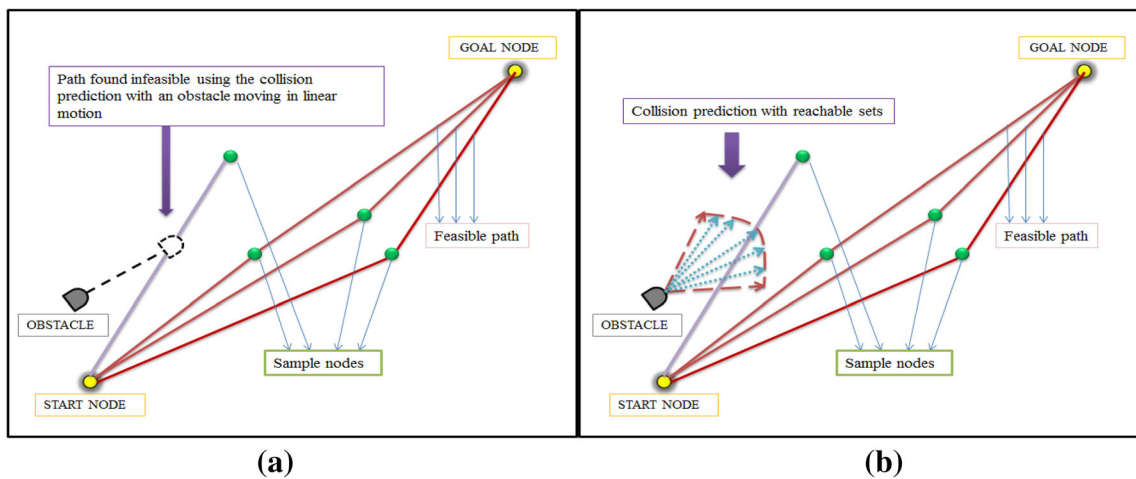


Figure 2. (a) Simplified node connection strategy – MRRT, (b) Representation of reachable set.

3.1a *Modified RRT algorithm*: The Modified Rapidly-exploring Random Tree algorithm is developed in this paper as shown in figure 1. Developments were made in RRT algorithm by modifying the node connecting strategy, generating a viable path in a short time. Additional paths were included by connecting the in-between waypoints in the developed paths to the goal and the reachable set concept is utilized to compute the states of obstacles in motion, for better reactionary obstacle detection.

3.1b *Simplified node connection strategy*: An approach to modify the node linking strategy is found to hasten the path planning rate. Contrasted with the standard RRT technique, sampled point is joined from the start point rather than the closest point of the RRT tree. In every sequence of simulation, a course is simulated from the start point to the sampled point and checked against limitations. If requirements are fulfilled, then another course from the sampled point to the objective (goal) point is simulated. An entire trajectory course linking the start and goal point is attained, only if the limitations formulated are satisfied as illustrated in figure 2(a). The proficiency of the method is further improved by exploiting the intermediate points obtained by discretizing the collision-free path generated by simplified node joining strategy. This process is repeated and all these trajectories are again simulated to check against constraints from intermediate points. Among the trajectories developed, the shortest is chosen. The features of the algorithm are shown as flowchart in figure 3.

3.1c *Collision Avoidance with Reachable Sets (CARS)*: The methods such as standard RRT and simplified node connection strategy predict the collision by assuming that obstacles move in linear motion. But in reality, obstacles may not follow the straight path consistently; this can be dealt using reachable sets. A reachable set of the moving obstacle is the region determined by set of all possible trajectories that can reach at any time instance. Reachable

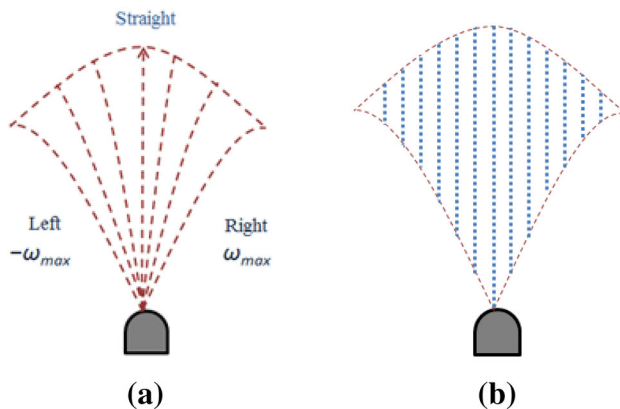


Figure 4. Illustration of Reachable set concept. (a) Possible trajectories of an obstacle in motion. (b) Set of possible trajectories from the reachable sets.

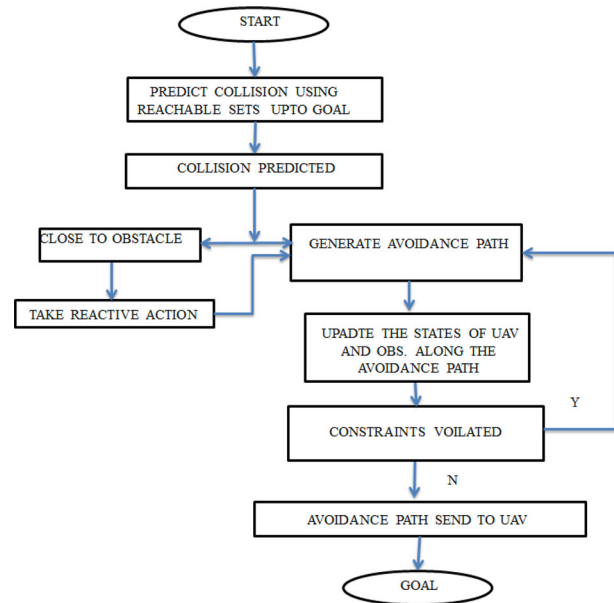


Figure 5. Flowchart of Collision avoidance using reachable sets.

sets are calculated by using the Dubins car model for obstacle in motion as of [12].

In figure 4 the obstacle have assumed to move at constant speed and with the permitted maximum turn rate of ω_{max} to the right and $-\omega_{max}$ to the left for the given time period figure 4(a). Thus a set of permitted trajectories are generated, composing the reachable sets figure 4(b). The developed approach checks, whether the selected course converges with the obstacles in motion or not, utilizing the concept of reachable sets as illustrated in figure 2(b). Frequent changes in the position of UAV and obstacles' is made available to the path generator. An evasion path will be generated by MRRT algorithm, if the obstacle is away from the UAV. Else, an avoidance maneuver such as vertical or horizontal maneuver is taken if the obstacle is too close or enters the safety region. The path developed is simulated and checked against limitations with upgraded states. The procedure is rehashed until the UAV arrives at its target node as shown in flowchart figure 5.

4. Results and discussion

The simulation of Standard RRT algorithm and modified RRT algorithm with collision avoidance using reachable sets are simulated using the software Python 3.7. Simulation results are presented as follows,

4.1 Implementation of RRT algorithm

A Path is developed autonomously from the Start to Goal State by randomly exploring the nodes in the given

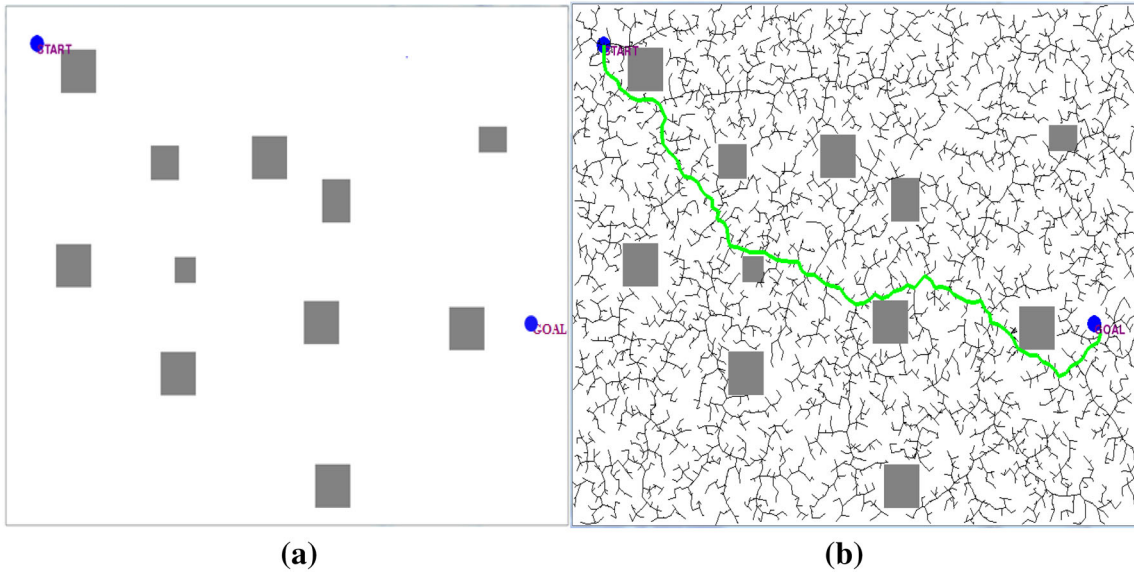


Figure 6. (a) Static obstacles, start and goal point is set. (b) Path created by RRT Algorithm.

Table 1. Workspace description consisting of obstacles and the sample nodes explored by RRT algorithm.

| | |
|--------------------------|------------|
| Workspace | [800, 600] |
| Start state | (37, 44) |
| Goal state | (740, 352) |
| Static obstacles | 11 |
| Total nodes explored | 2241 |
| Time taken to reach goal | 23.57 sec |

workspace by the algorithm of randomly exploring random tree and the path created is free of obstacles which are shown in figure 6 and the data are tabulated in table 1.

4.2 Implementation of modified RRT algorithm

4.2a *Simplified node connection strategy:* Simplified node connection strategy is simulated for the same start and goal node used in RRT algorithm simulation, and a path not colliding with obstacles, is generated through the single node in a short time compared to RRT, shown in figure 7 and the data are tabulated in table 2.

Comparison between the standard RRT and MRRT algorithm is made in terms of time taken by the algorithms in generating the feasible path using the same workspace dimension, start and goal node. MRRT more likely to generate the path in short time than standard RRT, plotted in figure 8 and the data are tabulated in table 3.

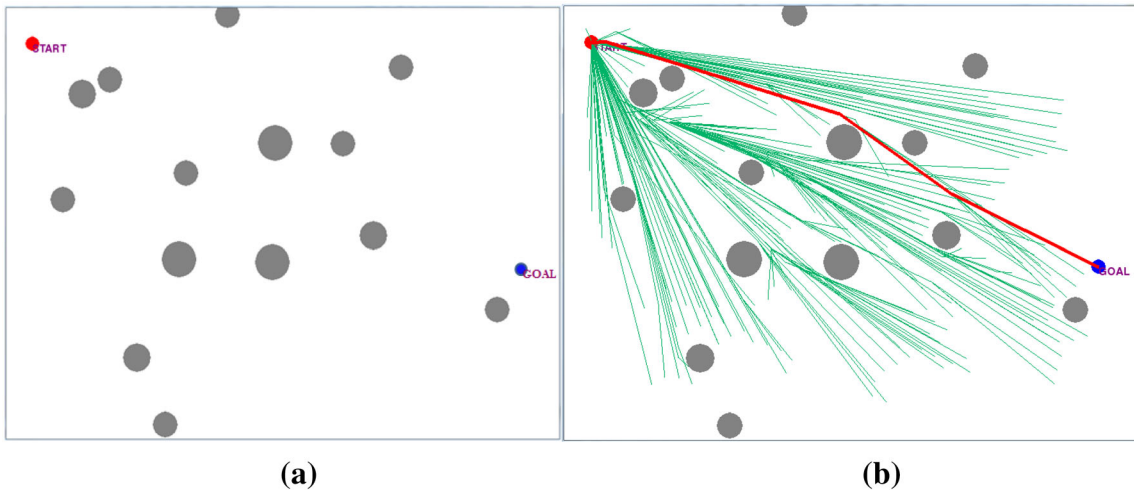


Figure 7. (a) Static obstacles start and goal point is set. (b) Path generated by modified RRT algorithm.

Table 2. Workspace description consisting of obstacles and the sample nodes explored by MRRT algorithm.

| | |
|--------------------------|------------|
| Workspace | [800, 600] |
| Start state | (37, 44) |
| Goal state | (740, 352) |
| Static obstacles | 14 |
| Node Chosen | 775.25 |
| Time taken to reach goal | 4.31 sec |

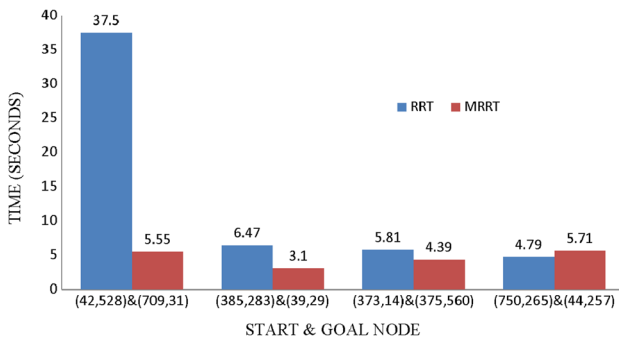


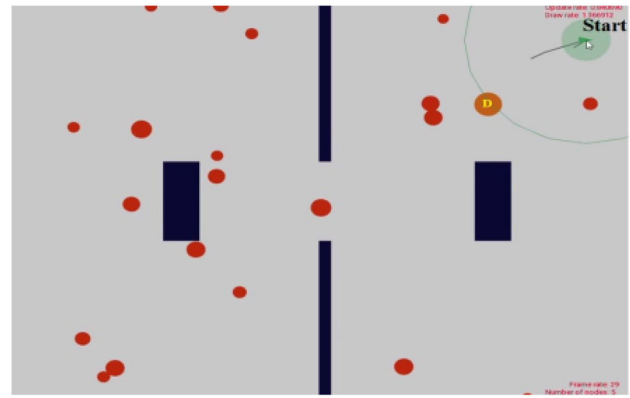
Figure 8. Comparison - RRT vs. MRRT.

Table 3. Time taken to generate path RRT vs MRRT for same start and goal node.

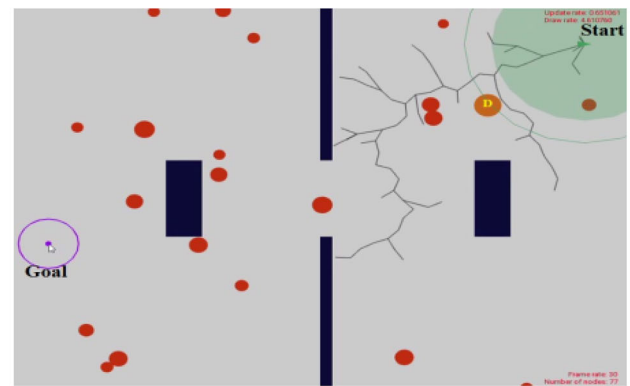
| Start point | Goal point | RRT | MRRT |
|-------------|------------|---------|--------|
| (42, 528) | (709, 31) | 37.35 s | 5.55 s |
| (385, 283) | (39, 29) | 6.47 s | 3.10 s |
| (373, 14) | (375, 560) | 5.81 s | 4.39 s |
| (750, 265) | (44, 257) | 4.79 s | 5.71 s |

4.3 Collision avoidance using reachable sets (CARS)

Trajectory simulated along the set path in 2D workspace, and the path gets updated based on the reachable sets of static obstacles (circular and rectangular objects) and



(a)



(b)

Figure 10. (a) Start point and (b) goal point set for simulation.

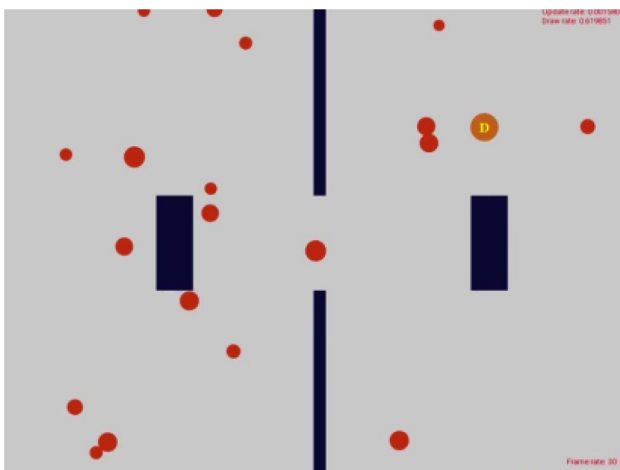


Figure 9. Workspace [800, 600] with Static and dynamic obstacles.

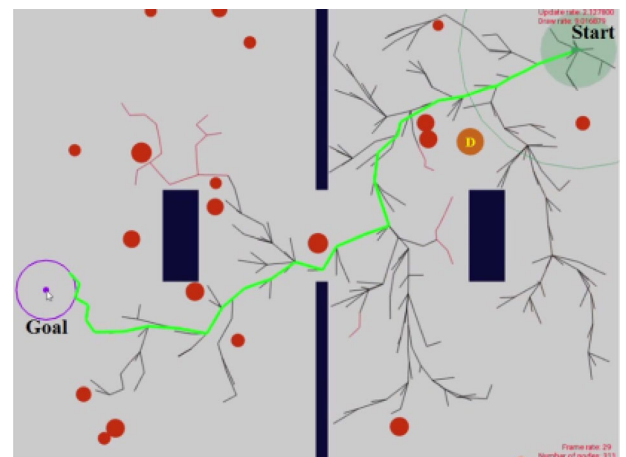


Figure 11. Trajectory defined by simplified node connection strategy.

dynamic obstacle (circular object with D) to avoid collision along the trajectory and the simulation results are shown in figures 9, 10, 11, 12, 13, 14, 15. Implementation steps of algorithm are as follows.

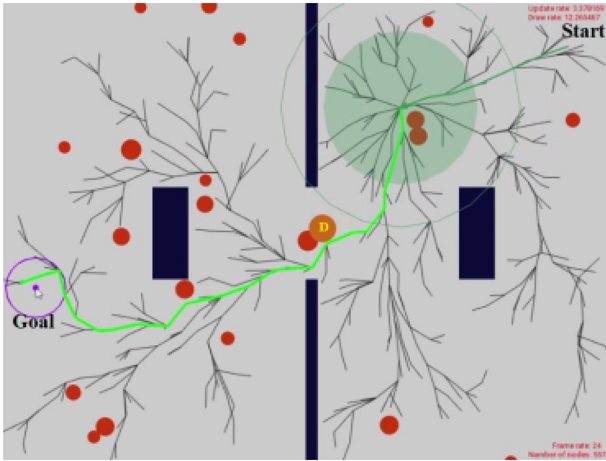
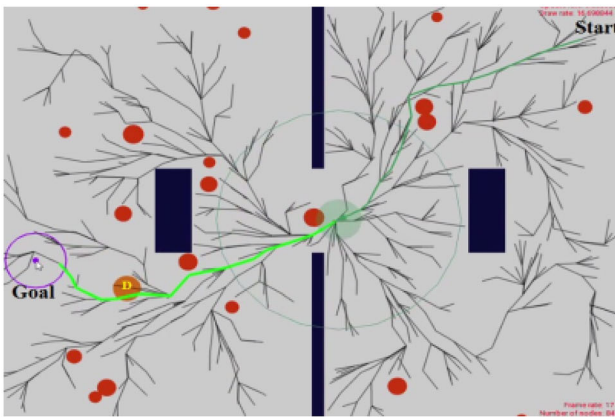
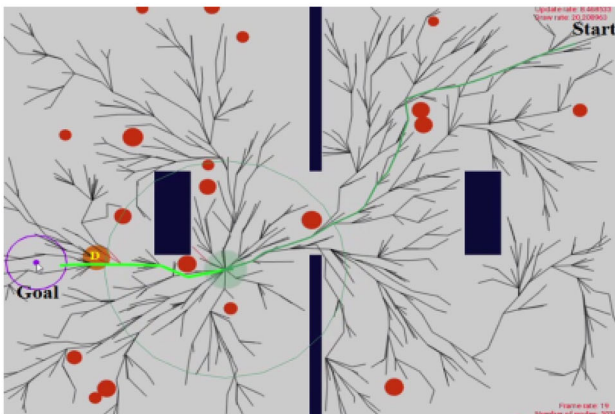


Figure 12. Holonomic object movement along the trajectory.



(a)



(b)

Figure 13. Trajectory changes with reachable sets of dynamic obstacle.

Step 1: Created workspace with static obstacles (circular and rectangular objects) and dynamic obstacle (circular object with D).

Step 2: Start and goal point set randomly, and started adding candidate paths, figures 10 (a) and (b).

Step 3: MRRT planner starts exploring the workspace and sets a defined initial path to the goal which (figures 11 and 12) describes simplified node connection strategy.

Step 4: Defined collision free trajectory changes with the movement of dynamic obstacles as following the path using the reachable sets formulation technique, figures 13 (a) and (b).

Step 5: Goal region is reached without colliding with the obstacles in figures 14 and 15.

Goal reached in 35 sec and initial path created at 10 sec avoiding the obstacles by identifying potential collisions along the path generated by MRRT technique. The intermediate points are used to set candidate paths among the total nodes explored of about 1294.

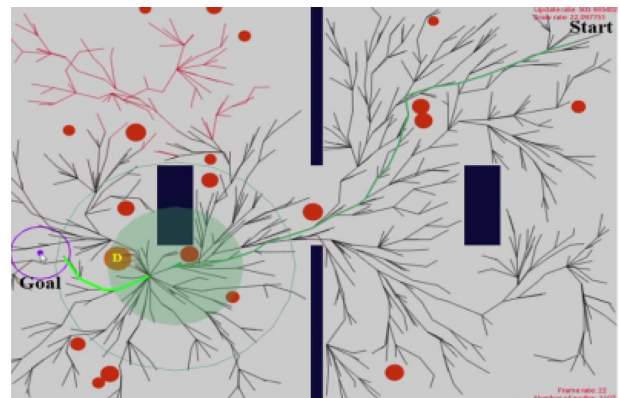


Figure 14. Green shadowing - safety region.

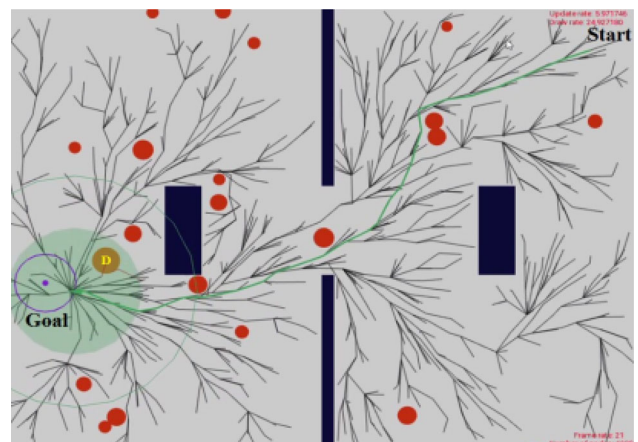


Figure 15. Connecting to the final goal.

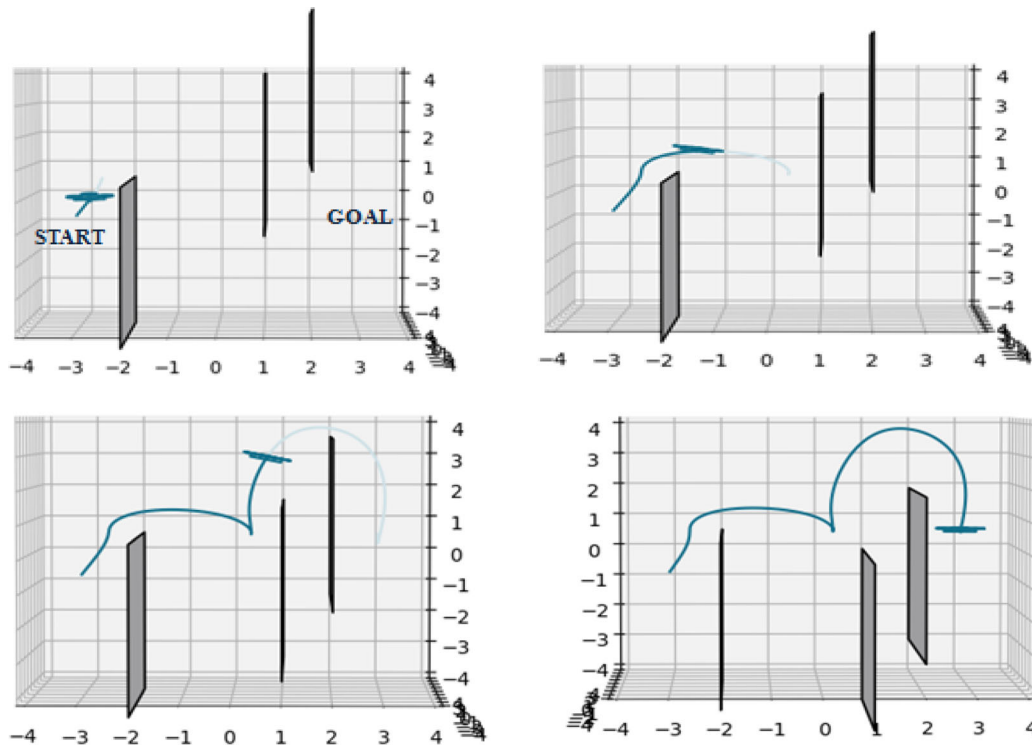


Figure 16. Simulation of CARS in 3D workspace.

Collision avoidance using reachable sets is simulated using the UAV module in 3D environment. UAV takes reactive action when found that the initial path is not feasible and then moves along the new path avoiding collision with moving plates to reach the goal, shown in figure 16.

5. Conclusion

In this paper, modified RRT algorithms with collision avoidance using reachable sets were developed. The developed algorithm were simulated in the Python software in 2D and 3D workspace by utilizing the python modules. Results show modified RRT algorithm develops collision free path within a short time compared to RRT when simulated with static and dynamic obstacles. Results show that the proposed method was more robust to deal with unexpected obstacle maneuvers, by defining the trajectory autonomously and following up without colliding with the obstacles along predicting the potential collision. Path effectively changes based on the prediction of obstacles in motion using reachable sets calculated using dubins model approach to avoid collision along the trajectory. The validation of this work in HILS (Hardware in Loop simulation) which is in progress.

References

- [1] Gasparetto A, Boscariol P, Lanzutti A and Vidoni R 2015 Path planning and trajectory planning algorithms: a general overview. Springer International Publishing, Motion and operation planning of robotic systems, pp 3–27
- [2] Yang, Liang, Juntong Qi, Jizhong Xiao and Xia Yong 2014 A literature review of UAV 3D path planning. In: *Proceeding of the 11th World Congress on Intelligent Control and Automation*. 2376–2381
- [3] Lin Yucong and Saripalli Srikanth 2017 Sampling-based path planning for UAV collision avoidance. *IEEE Transactions On Intelligent Transportation Systems* 18(11): 3179–3192
- [4] Liu Zhixiang, Zhang Youmin, Yuan Chi, Ciarletta Laurent and Theilliol Didier 2018 Collision avoidance and path following control of UAV in hazardous environment. *Journal of Intelligent & Robotic Systems*. 95: 193–210
- [5] Hao Xu, Xiangrong Xu, Yan Li, Xiaosheng Zhu, Liming Jia, and Dongqing Shi 2014 Trajectory planning of Unmanned Aerial Vehicle based on A* algorithm. In: *IEEE International Conference on Control and Intelligent Systems*. 463–468
- [6] Kothari Mangal and Postlethwaite Ian 2012 A Probabilistically Robust Path Planning Algorithm for UAVs Using Rapidly-Exploring Random Trees. *Journal of Intelligent and Robotic systems*. 71(2): 231–253
- [7] JiaweiMeng, Sebastian Kay, Angran Li and Vijay M. Pawar 2018 UAV Path Planning System Based on 3D Informed

- RRT* for Dynamic Obstacle Avoidance. In: *IEEE International Conference on Robotics and Biomimetics*. 1653–1659
- [8] Christain Zammit and Erik-Jan Van Kampen 2018 Comparison between A* and RRT algorithms for UAV path planning. In: *AIAA Guidance, Navigation, and Control Conference*. 1–23
- [9] Yan Fei, Liu Yi-Sha and Ji-Zhong 013 Path planning in complex 3D environments using a probabilistic roadmap method. *International Journal of Automation and Computing* 10(6): 525–533
- [10] Lee M C and Park M G 2003 Artificial potential field based path planning for mobile robots using a virtual obstacle concept. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. 2: 735–740
- [11] Yucong Lin and Srikanth Saripalli 2016 Sampling based collision avoidance for UAVs. In: *American Control Conference*. 1353–1358
- [12] Albert Wu and How Jonathan P 2012 Guaranteed infinite horizon avoidance of unpredictable, dynamically constrained obstacles. *Autonomous Robot* 32(3): 227–242