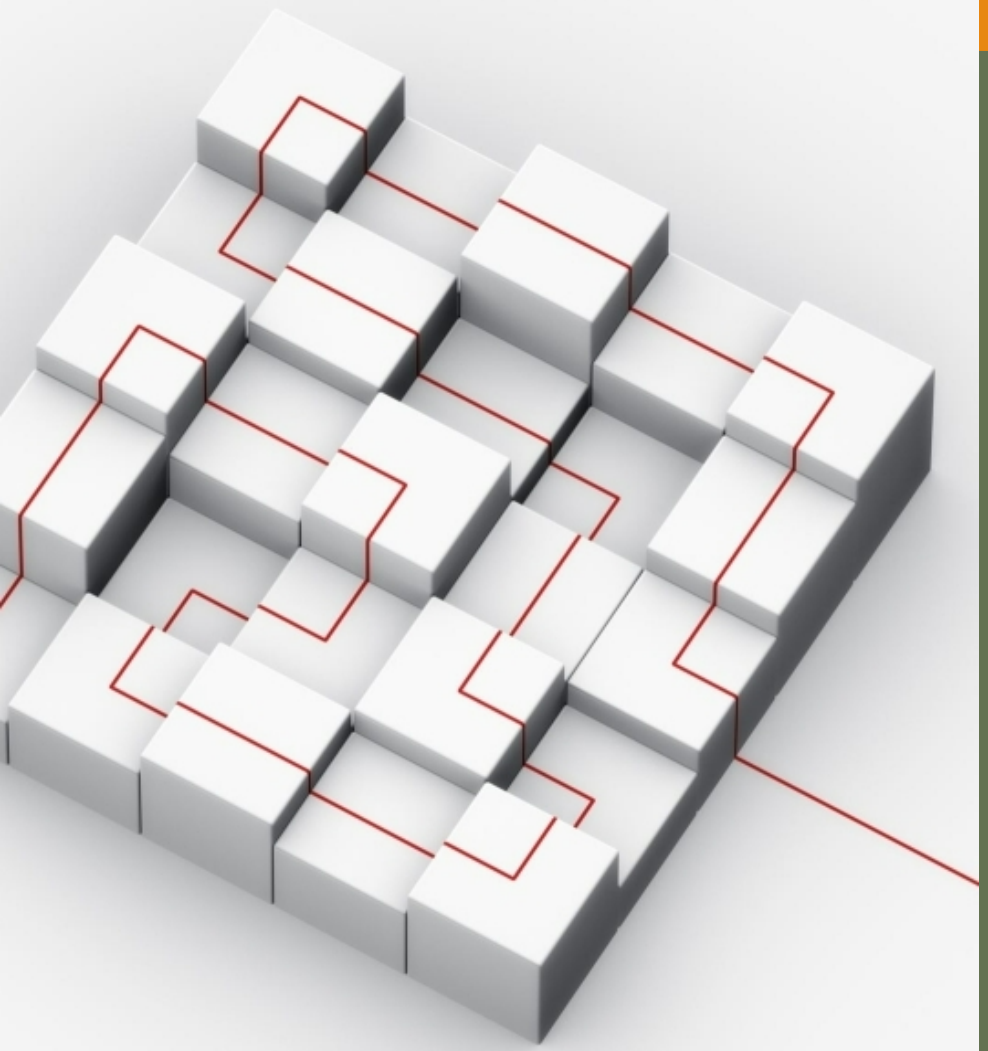
Path Planning for wheelchair navigation

Sharmita Ganesan

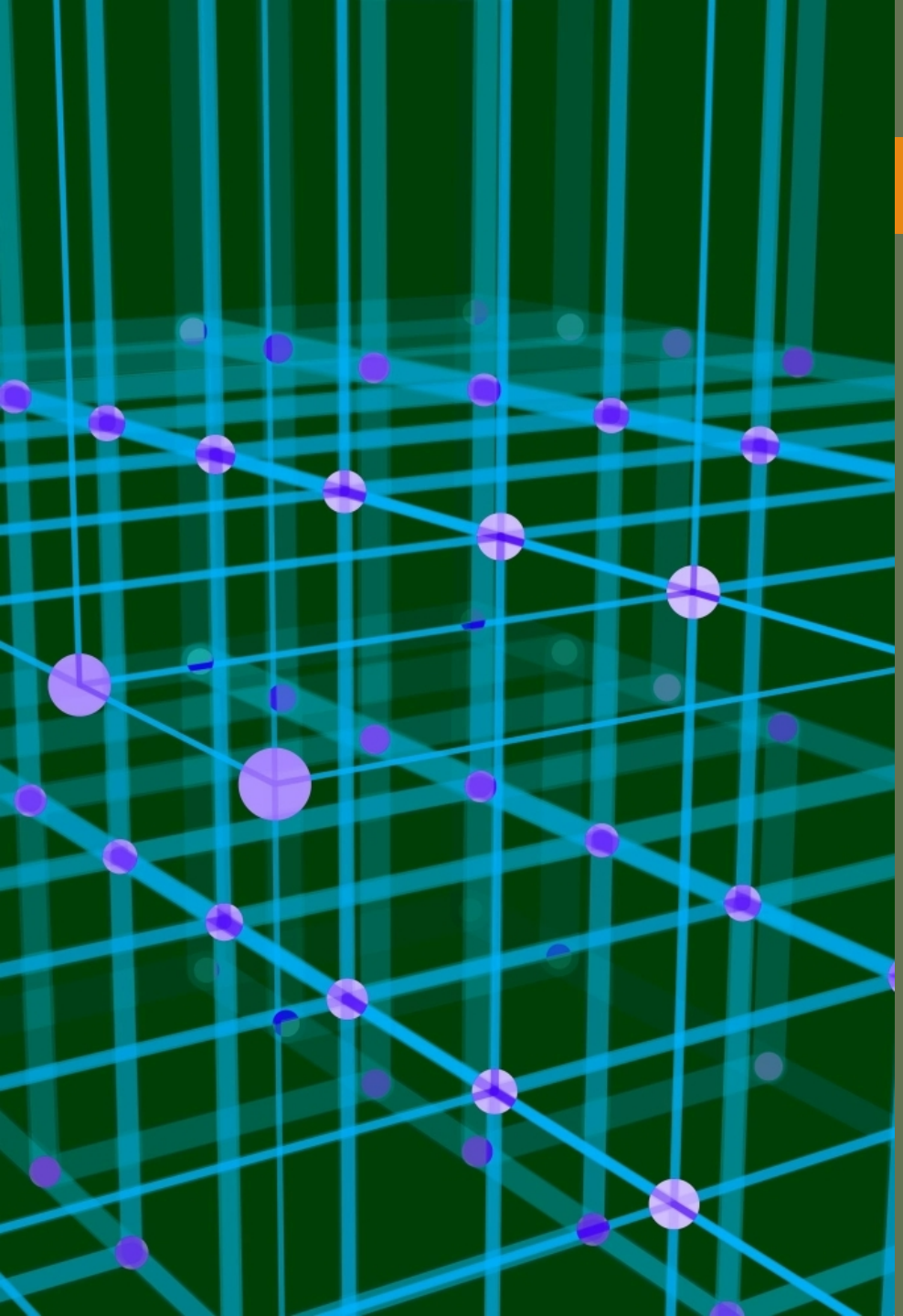
Shelvin Pauly

Contents



Introduction

- Path planning is finding a path from source to target avoiding obstacles. Building a path planner for Autonomous Robots needs to be able to do the same in an environment with dynamic obstacles. There are different types of path planning algorithms: search-based algorithms such as Depth First Search, Breadth First Search, Dijkstra, A* etc., and sampling-based algorithms as RRT (Rapidly exploring Random Tree), RRT*, Fast Marching Tree etc., where all of them focuses on efficient algorithm with fast convergence and low memory space. Search based algorithms computes path over discrete graphs. This technique can provide efficient solutions at the expense of computation time. Sampling based algorithms are fast but can provide inefficient or unusual paths.



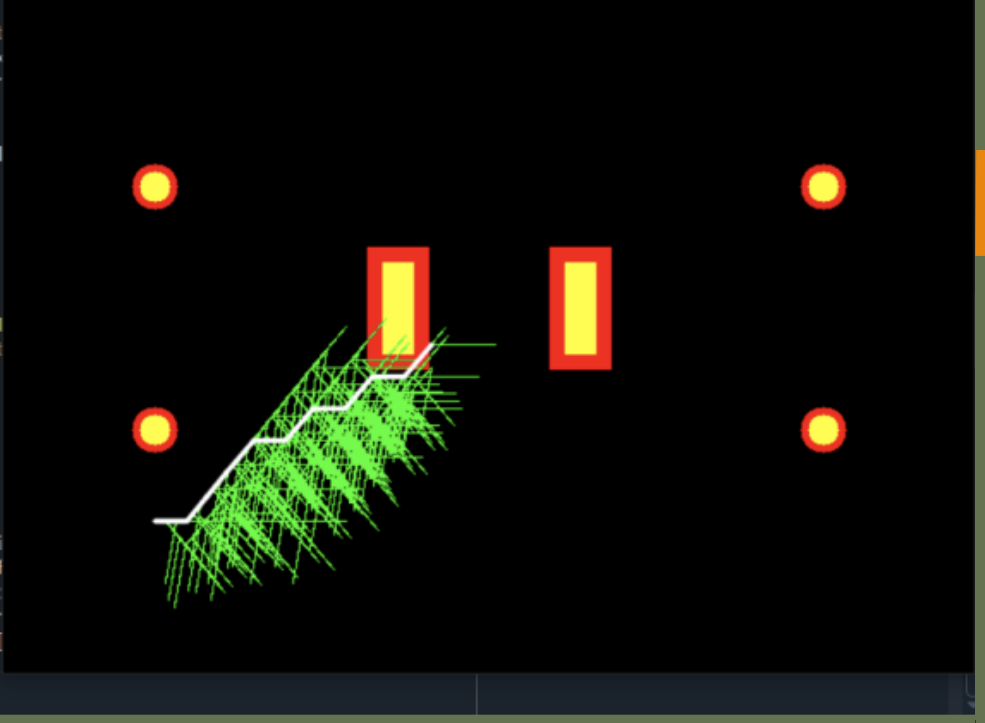
Methodology

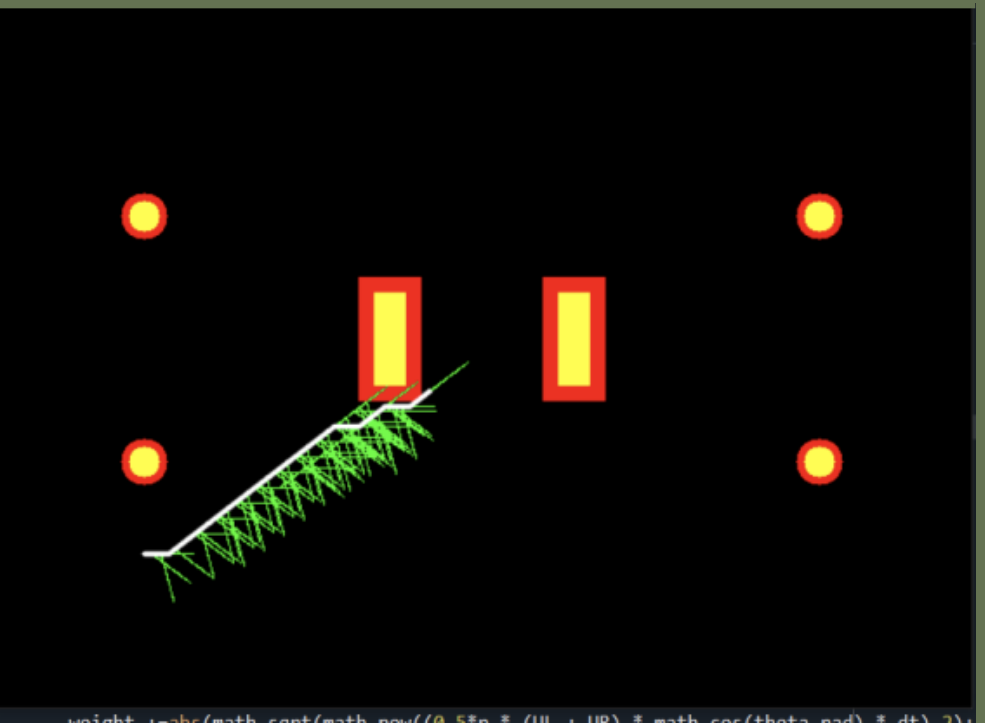
- In this project, we used two algorithms: A*, which is a search-based strategy and RRT* which is sampling-based algorithm. We implemented the code in @D as well as 3D, using both static and dynamic obstacles.

The environment for all simulations were based on an airport scene with pillars and seating areas, essentially a wheelchair navigation in an airport scenario. The dynamic obstacles account for the moving people, trolleys and bags in the airport.

Map size – 640 × 480 units

- Non-Holonomic Action Set – straight_near, straight_far, sharp_left, sharp_right, slight_left, slight_right.



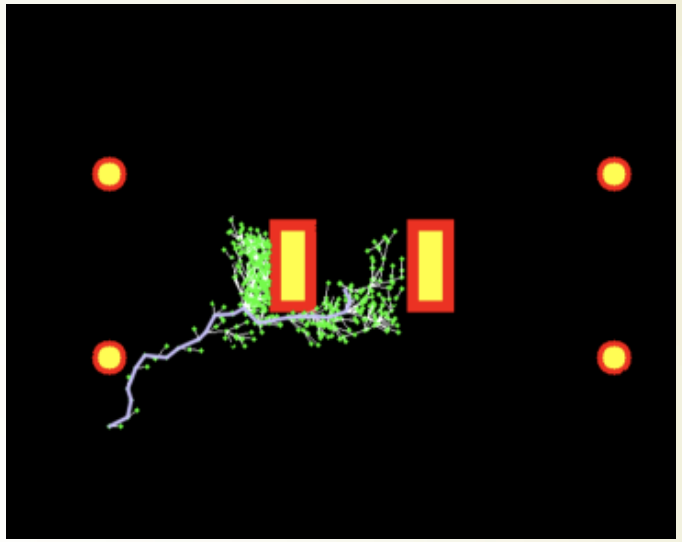


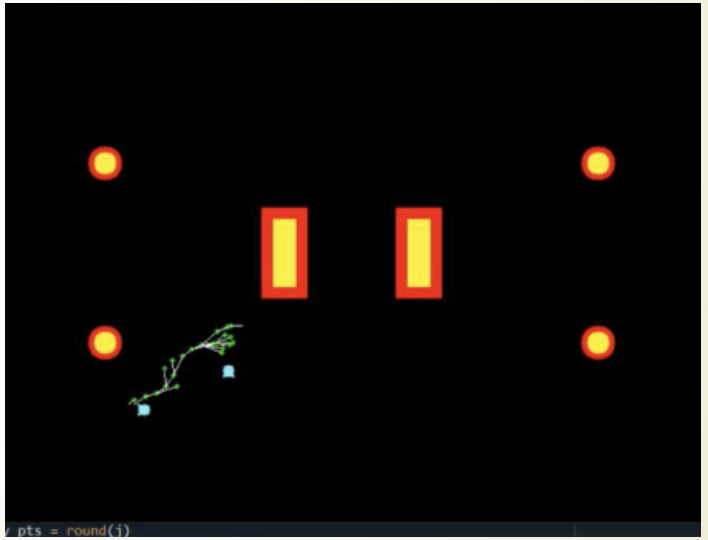
A*

- A* tries to find the shortest path to the goal from all the explored nodes. This is achieved by calculating the cost to reach the current node and the cost to go to goal from that specific node. All the new nodes are added to an open list, and the cheapest one is popped from there and is visited.
- And if a node has a cheaper alternative, the costs and parent is updated in the process.
- Step Size – 5, 10 units

RRT*

- RRT* is a modification over RRT. The basic difference between both is the rewiring of the graph based on the cost to come of the nearest nodes. These nearest nodes are found within neighboring radius of 20 units.
- Nodes are limited with 250 iterations each.
- Step Size – 10 units





Simulation

- Link for Gazebo sim

https://drive.google.com/file/d/16kQEY46So1Jm3abz1v4RAXHO4p8AkZX5/view?usp=sharing

- RTT* Dynamic obstacles outputs:

https://drive.google.com/file/d/10JxB_YCtUIeWnypENOMFvKWfJDPc_Ug5/view?usp=sharing

https://drive.google.com/file/d/1mQOPIylkV4jzDQ4FX-RVCWuuA58fpdd4/view?usp=sharing

Conclusion

- The project successfully simulated wheelchair navigation in an airport environment while avoiding both static and dynamic obstacles. The simulation clearly portrays the vast superiority sampling-based techniques have over search-based algorithms in a real-life situation.
- Our project depicts that this planner can be implement ed in real life with some modifications in the future. To implement this planner on a real-life basis, we need to include factors such as comfort of the user and model the obstacle’s behavior like they would behave in real life and simulate for more complex behaviors by obstacles. There should also be certain safety related factors that needed to be accounted for.

References

Y. Li, "An RRT-Based Path Planning Strategy in a Dynamic Environment," 2021 7th International Conference on Automation, Robotics and Applications (ICARA), 2021, pp. 1-5, doi: 10.1109/ICARA51699.2021.9376472.

J. Liu, B. Li, T. Li, W. Chi, J. Wang and M. Q. . -H. Meng, "Learning-based Fast Path Planning in Complex Environments," 2021 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2021, pp. 1351-1358, doi: 10.1109/ROBIO54168.2021.9739261.

Qinghua Li, Jihuai Wang and Haiming Li "Fast - RRT *: An Improved Motion Planner for Mobile Robot in Two - Dimensional Space", http://dx.doi.org/10.1002/tee.23502

Thank You