

Multiple Query PRM APF Planner - Documentation

Instructions

- **NOTE:** I have done only upto checkpoint 2 (included), i.e., until PRM-based global planning.
- The program to be run is `main.ipynb` .
- I have provided necessary documentation in the notebook wherever necessary, and also displayed the required plots.

Process and Methods Used

- Matplotlib library was used for visualising the grid
- First the obstacles were drawing using the values given in `obstacles.txt`
- Global path planning was done using Probabilistic Road Map (PRM) method:
 - A list of 100 nodes was generated randomly, and it was checked whether they lied within any obstacle or not.
 - After following the above method, the valid nodes were stored separately for use, whereas the others were rejected.
 - Each valid node was connected to its K (taken as 3) nearest neighbors connected by straight lines if there was no obstacle between them - this was checked using incremental method.
 - Now the queries from `queries.txt` were sampled one-by-one.
 - They were augmented into the roadmap, then connected to the existing roadmap through its K nearest neighbors.
 - A^* algorithm was then used to find the shortest path between them:
 - Starting from the `start` node, each K neighboring node was explored and its distance was updated following the principles of A^* algorithm.
 - The heuristic function chosen was the distance of the current node from the `goal` node.
 - **Note:** Distance was just calculated as the sum of the square of the difference of x and y co-ordinates.

Results

All the relevant results are displayed in the Jupyter Notebook.

References Used

- [Principles of Robot Motion](#)
- [A* Search Algorithm](#)