Path Planning system

By Mahmoud Elgendy

9/8/2023

1 Introduction to Path Planning

In the robotics, path planning is a critical component of determine the localization and the navigation to robot, addition to identify safe and efficient route , collision-free, and least-cost travel paths from an origin to a destination. Path planning is essential for robots operating in dynamic and complex environments.

2 Types of Path Planning Algorithm

the choosing right path planning algorithm is important to ensure safe and efficient navigation in various environments and to less the time.

1. **Dijkstra's Algorithm**:is a popular algorithms for solving many single-source shortest path problems having non-negative edge weight in the graphs i.e., it is to find the shortest distance between two vertices on a graph.

2. **A* Algorithm**: Combines the strengths of Dijkstra's algorithm and it directs its search toward the most promising states, potentially saving time.and It is commonly used in static environments but also dynamic environments.

3. **D* Algorithm**: Path planning in partially known and dynamic environments, such as for automated vehicles, is becoming increasingly important. The D* (or Dynamic A*) algorithm generates a collision-free path among moving obstacles to solving this problem. D* is a cost map repair algorithm that uses informed incremental search to partially repair the cost map and the previously calculated cost map.

4. **Rapidly-exploring Random Trees (RRT)**: is an algorithm designed to efficiently search nonconvex, high-dimensional spaces by randomly building a space-filling tree. The tree is constructed incrementally from samples drawn randomly from the search space and is inherently biased to grow towards large unsearched areas of the problem.

5. **Probabilistic Roadmap (PRM)**: Builds a roadmap of the configuration a path between a starting configuration of the robot and a goal configuration while avoiding collisions.

2.1 Dijkstra's Algorithm

2.1.1 how it work

- Basically, Dijkstra's algorithm starts at the node source node we choose and then it analyzes the graph condition and its paths to find the optimal shortest distance between the given node and all other nodes in the graph.

- Dijkstra's algorithm keeps track of the currently known shortest distance from each node to the source node and updates the value after it finds the optimal path once the algorithm finds the shortest path between the source node and destination node then the specific node is marked as visited.

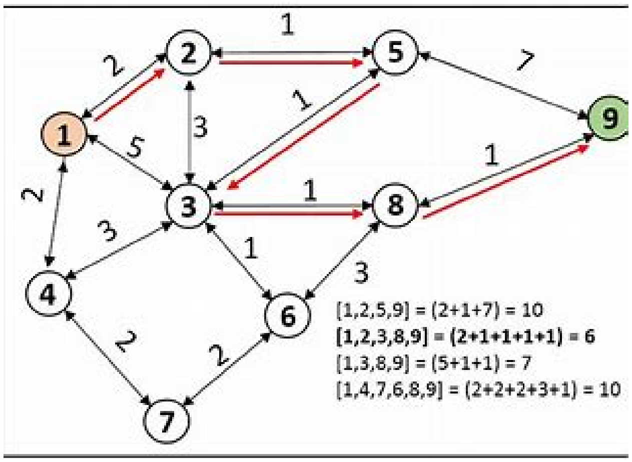

Figure 1: Illustration of Dijkstra's algorithm.

2.2 A* Algorithm

2.2.1 how it work :

The A* algorithm is a heuristic algorithm that finds the best path using heuristic information. The A* algorithm must search the map for nodes and apply appropriate heuristic functions to provide guidance. The two factors that govern an algorithm are the efficient resources used to perform the task and the response time or computation time taken to perform the task. When using the A* algorithm, there is a trade-off between speed and accuracy. We can reduce the algorithm's time complexity in exchange for more memory or consume less memory for slower executions. We find the shortest path in both cases. The A* algorithm can be used to find the shortest path to an empty parking space in a crowded parking lot.

2.3 Probabilistic Roadmap (PRM)

2.3.1 how it work

- it grows a tree starting from an initial configuration by using random samples from the search space. It attempts to connect each sample to the nearest state in the tree, adding the new state if the connection is feasible and passes through free space. The algorithm prefers to expand into large unsearched areas, making it space-filling.

- The length of connections is limited by a growth factor, and if a random sample is too far from the nearest state, a new state is created along the line to the sample. Random samples influence the direction of tree growth, while the growth factor controls its rate, ensuring incremental growth.

- RRT growth can be biased by increasing the probability of sampling states from specific areas, often used to guide the search toward the desired goal in practical implementations. A higher probability of sampling the goal leads to a more greedy expansion of the tree toward the goal.

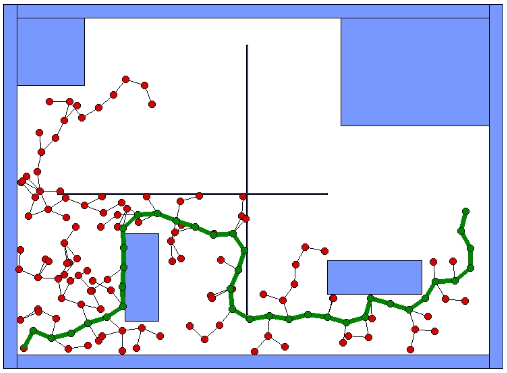

Figure 2: Probabilistic Roadmap (PRM)

2.4 Probabilistic roadmap

2.4.1 The basic idea

- is to take random samples from the configuration space of the robot, testing them for whether they are in the free space, and use a local planner to attempt to connect these configurations to other nearby configurations. The starting and goal configurations are added in, and a graph search algorithm is applied to the resulting graph to determine a path between the starting and goal configurations.
- The probabilistic roadmap planner consists of two phases: a construction and a query phase. In the construction phase, a roadmap (graph) is built, approximating the motions that can be made in the environment. First, a random configuration is created. Then, it is connected to some neighbors, typically either the k nearest neighbors or all neighbors less than some predetermined distance. Configurations and connections are added to the graph until the roadmap is dense enough. In the query phase, the start and goal configurations are connected to the graph, and the path is obtained by a Dijkstra's shortest path query.

3 Local Planner and Global Planner

Path planning is often divided into two main components: local planning and global planning.

- **Local Planner**:
  - it focus in the immediate surroundings from all obstacles like cars , during the movement, making quick decisions .
  - It deals with real-time obstacles (both moving and static) that encounters. If there's an unexpected obstacle, the local planner adjusts the robot's path to avoid a collision.
- **Global Planner**:
  - it see the BIG PICTURE as it process of all entire routes from the current position to destination position , including the environment and all obstacles in the path.
  - It doesn't react to things happening in real time, like a person suddenly walking by. It assumes the environment is mostly static and doesn't change rapidly.
  - **Exapmle** : it is like the GPS .

both local and global planing important as the local interest in the handling the immediate actions and obsatcles avoidance and gloabl planning deals with general route and make entire big overview.

4 Challenges in Path Planning

1. **dynamic environment** : due to the dynamics of the environment and moving obstacles lead to the difficult for robot to adapt and need good algorithm to solve that but , is still challenge.
2. **High-Dimensional spaces::** there are a large number of variables and parameters that need to be considered to describe the robot's state accurately and the computational required to search and navigate through this space can grow exponentially.

5 Future Trends in Path Planning

1. **Machine Learning Integration**: Machine learning techniques, such as deep reinforcement learning and neural networks, are being integrated into path planning to improve decision-making in complex and dynamic environments and these methods can adapt the dynamic environment from learn exiting data.

6 Practical Applications of Path Planning

Self-driving cars rely on path planning to navigate roads safely.

7 Conclusion and Resources

The path planing is very important for robotics to deal with environment and save time and make efficient decisions in dynamics and open area.

7.1 The resources:

- Path Planning Algorithms For Robotic Systems: it's a website explain in summary Algorithms For the Path Planning in Robotic Systems.
- Wikipedia - Path finding: addition resources on path planning algorithms.
- algorithm and the future trend:website explain the path planing algorithm and the future trend withe machine learning