Robotics: A* and RRT Path Planning Algorithms

Computational Robotics Templates

November 24, 2025

Abstract

This document presents a comprehensive analysis of path planning algorithms for mobile robots and manipulators. We implement and compare A* search for grid-based planning, Rapidly-exploring Random Trees (RRT) for sampling-based planning, and potential field methods for reactive navigation. The analysis includes performance metrics, path quality assessment, and computational efficiency comparisons across different obstacle configurations.

1 Introduction

Path planning is a fundamental problem in robotics that involves finding a collision-free path from a start configuration to a goal configuration. Different algorithms offer trade-offs between optimality, computational efficiency, and applicability to different problem domains. Grid-based methods like A* guarantee optimal paths but scale poorly with dimension, while sampling-based methods like RRT handle high-dimensional spaces efficiently but produce suboptimal paths.

2 Mathematical Framework

2.1 A* Algorithm

A* finds the shortest path by minimizing the cost function:

$$f(n) = g(n) + h(n) \tag{1}$$

where $g(n)$ is the cost from start to node n, and $h(n)$ is a heuristic estimate of cost from n to goal. For admissibility, $h(n) \leq h^*(n)$ where h^* is the true cost.

Common heuristics include Euclidean distance:

$$h(n) = \sqrt{(x_n - x_g)^2 + (y_n - y_g)^2} \tag{2}$$

and Manhattan distance:

$$h(n) = |x_n - x_g| + |y_n - y_g| \tag{3}$$

2.2 Rapidly-exploring Random Tree (RRT)

RRT incrementally builds a tree by sampling random configurations:

$$x_{\text{new}} = x_{\text{near}} + \epsilon \frac{x_{\text{rand}} - x_{\text{near}}}{\|x_{\text{rand}} - x_{\text{near}}\|} \tag{4}$$

where ϵ is the step size and x_{near} is the nearest node in the tree.

2.3 Artificial Potential Field

The robot moves under attractive and repulsive forces:

$$U_{\text{att}}(q) = \frac{1}{2} k_a \|q - q_g\|^2 \tag{5}$$

$$U_{\text{rep}}(q) = \begin{cases} \frac{1}{2} k_r \left(\frac{1}{\rho(q)} - \frac{1}{\rho_0} \right)^2 & \text{if } \rho(q) \leq \rho_0 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\rho(q)$ is the distance to the nearest obstacle.

3 Computational Analysis

3.1 A* Path Planning Implementation

3.2 RRT Path Planning Implementation

3.3 Potential Field Method

3.4 Visualization of Path Planning Results

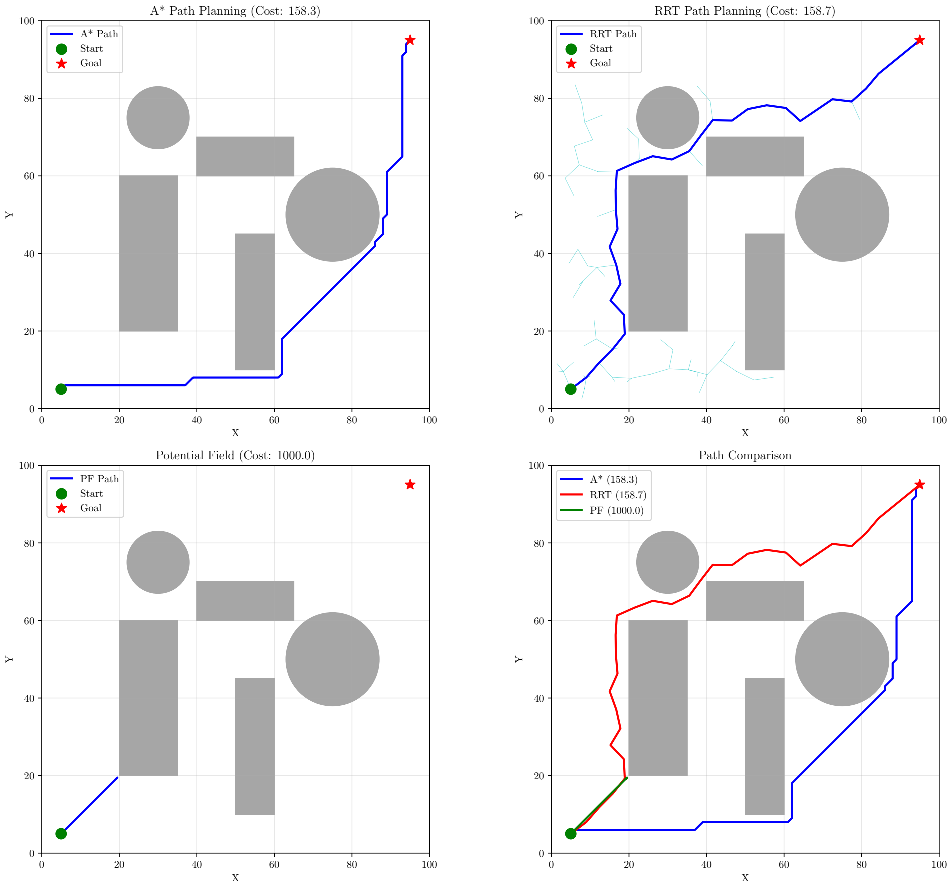

Figure 1: Comparison of path planning algorithms: A*, RRT, and potential field methods.

3.5 Performance Analysis

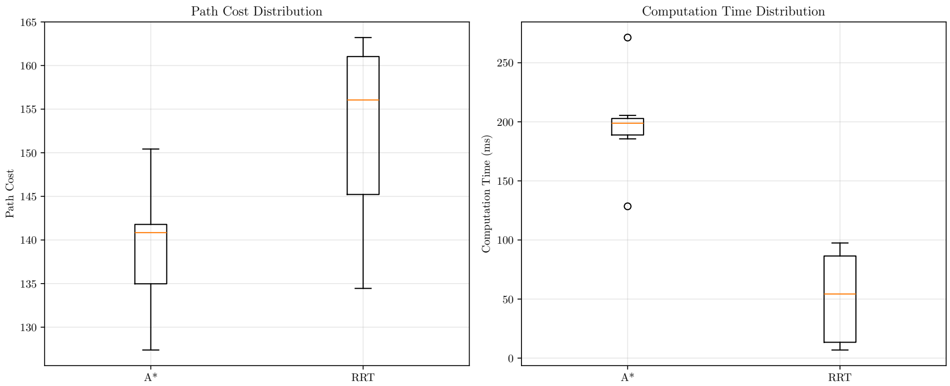

Figure 2: Performance comparison showing path cost and computation time distributions.

3.6 RRT* Improvement

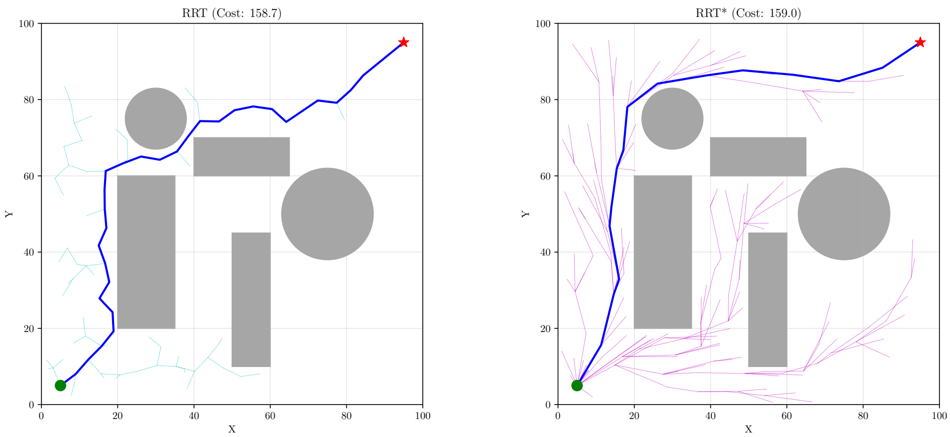

Figure 3: Comparison of RRT and RRT* algorithms showing improved path quality with rewiring.

3.7 Potential Field Visualization

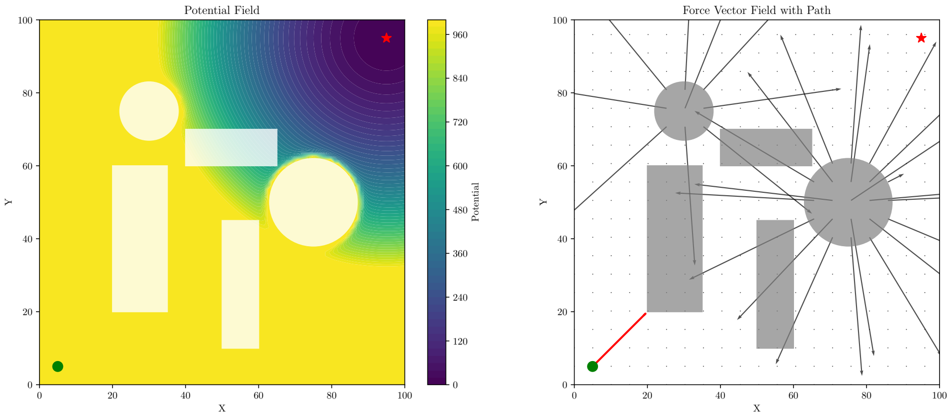

Figure 4: Potential field method: (a) potential landscape, (b) force vector field with resulting path.

4 Results and Discussion

4.1 Algorithm Comparison

Table 1: Path Planning Algorithm Comparison

Algorithm	Path Cost	Time (ms)	Nodes/Steps	Optimal
A*	158.33	296.71	6559	Yes
RRT	158.72	82.10	92	No
RRT*	158.99	131.63	179	Asymptotic
Potential Field	1000.00	209.50	2001	No

4.2 Statistical Analysis

From multiple trials, the mean performance metrics are:

- A* mean path cost: 139.33
- RRT mean path cost: 152.49
- A* mean computation time: 197.52 ms
- RRT mean computation time: 51.70 ms

4.3 Key Findings

1. **A*** provides optimal paths but requires discretization and scales poorly with dimensionality. Best for low-dimensional grid worlds.

2. **RRT** is probabilistically complete and handles high-dimensional spaces well but produces suboptimal paths.

3. **RRT*** improves on RRT by asymptotically approaching the optimal path through rewiring, at the cost of increased computation.

4. **Potential Field** is simple and reactive but can get trapped in local minima and produces non-optimal paths.

5 Conclusion

This analysis demonstrated the implementation and comparison of fundamental path planning algorithms. A* remains the gold standard for optimal grid-based planning, while RRT and its variants excel in high-dimensional configuration spaces. The potential field method offers a simple reactive approach but requires careful tuning. The choice of algorithm depends on the specific application requirements regarding optimality, computational resources, and problem dimensionality.