Formatting Instructions for CoRL 2020

Anonymous Author(s)
Affiliation
Address
email

Abstract: ABSTRACT PLACEHOLDER

Keywords: Sampling-based path planning, PLACEHOLDER FOR KEYWORD

1 Introduction

Path planning is a search problem where the goal is to find a sequence of actions that will lead to an obstacle-free optimal path from a start state to a target state [1]. Grid-based search algorithms have been proposed in the past [2] [3] that guarantee finding optimal paths that exists. However, these techniques do not scale well to higher dimensional problems as they require discretization of the state space. On the other hand, sampling-based path planning algorithms [4] [5] can solve high-dimensional path planning problems more efficiently, but often compute sub-optimal paths. Many attempts have been made to make improvements to sampling-based methods, using techniques such as heuristic guided searches [6], or the reduction of the sampling space [7]. The goal of this project is to implement three probabilistic sampling-based path planning algorithms, and compare their performances through simulations.

2 Problem

Path planning in continuous space is a difficult problem. On one hand, there are graph based approaches such as Dijkstra's algorithm, A*, etc. These approaches find the optimal solution, if one exists. In the case of A*, it uses a heuristic to find the optimal solution in fewer iterations. However, graph based approaches require discretization of continuous space. This is a problem for two reasons: First, a increasing the number of dimensions causes an explosion in the number of nodes. Second, increasing the resolution with finer discretization also causes a growth in the number of nodes. Having a high number of nodes creates a problem because the computational time of graph based approaches increases rapidly with the number of nodes.

On the other hand, there are sampling based approaches. They sample states from the continuous space, and create a tree, known as a Random Tree (RT). In Rapidly-exploring RT (RRT) algorithms, each new sampled state is added to the closest point on current tree. A more detailed description of the various sampling based approaches is presented in 3. However, while sampling rather than discretizing makes them more scalable, they do not have the ordered nature of graph based approaches for a more efficient search process.

PLACEHOLDER FOR PROBLEM - DISCUSS MOTIVATION FOR USING ASYMPTOTICALLY OPTIMAL SAMPLING BASED PATH PLANNING (asymptotic optimality, probabilistic completeness)

3 Related Work

FOR EACH ALGORITHM, LIST TECHNIQUES/ALGORITHMS ITS BUILDING ON OR IMPROVING UPON AND GIVE BRIEF A EXPLANATION

RRT - Create a tree of connected nodes by sampling a state from the continuous space and extending closest state on the existing tree towards the newly sampled state RRT* - Before each iteration of RRT, optimize the existing tree - More optimal path - Longer iterations Informed RRT* - Once a sub-optimal solution is found, reduces the continuous space from which states are being sampled to a space (an ellipse) that can improve the current solution, allowing for faster convergence to the optimal FMT* - Marching method to process single set of samples Randomized A* (RA*) and Sampling based A* (SBA*) - Sample near heuristically selected vertices of the graph - Biases growth of the tree but requires methods avoid local minima - Good for spaces with no obstacles, but not good for spaces with a lot of obstacles - They address these problems but with trade-offs - SBA* trade-off doesn't seem bad, may be worth comparing? NRRT* - RRT*, A*, Informed RRT*, A*-RRT*, Thtee*-RRT* (mention limitations due to scalability)

4 Proposed Method

- BRIEF DESCRIPTION OF EACH OF THE ALGORITHMS BEING IMPLEMENTED (INCLUDE ANY TECHNICAL/THEORETICAL DETAILS TO HIGHLIGHT CORE IDEAS, DO NOT GO INTO DETAILS)

In this project, we propose a comparison study of three asymptotically-optimal sampling-based path planning algorithms: Fast-marching trees (FMT*), batch informed trees (BIT*), and neural rapidly random-exploring trees (NRRT*) [8]. The following section will briefly describe each of these algorithms.

4.1 Fast-Marching Trees

Sampling based algorithms can be divided into two categories: Single query (RRT* []) and Multiple query. FMT* combines the

4.2 Bath Informed Trees

Uses batches of samples to perform an ordered search Sample randomly but use heuristic to determine which sample from the batch to process next Allows converging asymptotically towards the global optimum with anytime resolution (quickly find sub-optimal solution, then optimize it) Incorporate the new samples into the existing search Focus on the subproblem that contain a better solution

4.3 Neural RRT*

Neural RRT* is a non-uniform sampling-based path planning algorithm that aims to take advantage of the optimal path costs of the A* algorithm as well as the asymptotic optimality of the RRT* algorithm. Instead of sampling from the state space using just a uniform distribution, a non-uniform distribution learned by a convolutional neural network (CNN) trained on optimal paths generated from the A* algorithm is used to achieve non-uniform sampling. Unlike many of the RRT variant algorithms [CITATIONS PLACEHOLDER], the NRRT* algorithm does not depend on any human-designed heuristics to improve its performance. It also maintains the probabilistic completeness and the asymptotic optimality of the RRT* algorithm. Through simulations, it has been shown to perform superior compared to traditional algorithms such as RRT* and informed RRT* (IRRT*) [CITAION PLACEHOLDER].

5 Proposed Evaluation

We will implement the individual algorithms in python as per the discussions and directives mentioned in the individual papers ([9] FMT, [8] NRRT*,\cite{BIT* paper cite} BIT*). In order to make direct comparisons between optimal paths returned by the different algorithms, we will run them on a common set of cluttered environments (these would include bug traps and a set of 2D maze). We will measure 2 metrics as part of running the simulation: Optimal Path Cost ($\mathcal{J}$) and Execution TIme ($\mathcal{T}$). ET is calculated based on the convergence criteria for each algorithm. We evaluate the cost ($\mathcal{J}$) as follows: For a grid having N nodes in the free space $\mathcal{X}_{free}$, let $\mathcal{P} \subset \mathcal{X}_{free}$ be the set of nodes along the optimal path as computed by the path planning algorithm $\{\mathcal{X}_{free}, x_{init}, \mathcal{X}_{goal}\}$; Then the cost $\mathcal{J}$ is defined as,

$$\mathcal{J} = \sum_{i=1}^{N} \mathbb{I}[x_i \in \mathcal{P}], \text{ where } x_i \text{ are the nodes in the free space.} \tag{1}$$

Since, NRRT* does not involve setting the sample size / iteration count, we will do an additional comparison of the variation of cost ($\mathcal{J}$) and ET ($\mathcal{T}$) with sample count for FMT* and BIT*. Comparison will be done for 10 different sample counts in the range $[1000, 10000]$. Finally, we will also provide a qualitative comparison of the optimal paths resulting from each algorithm for the same set of environments highlighting their behaviour around corners and near obstacles.

6 Citations

Citations can be made using either \citep{} or \citet{}, depending from the appropriateness. To avoid the citation moving to the next line, it is often a good practice to replace the space before with a tilde (˜) character. Example 1: "CoRL is the best conference ever [**?**]." Example 2: "**?**] proved, both theoretically and numerically, that CoRL is the best conference ever."

References

[1] H. Liu. Chapter 1 - introduction. In H. Liu, editor, *Robot Systems for Rail Transit Applications*, pages 1–36. Elsevier, 2020. ISBN 978-0-12-822968-2. doi:https://doi.org/10.1016/B978-0-12-822968-2.00001-2. URL `https://www.sciencedirect.com/science/article/pii/B9780128229682000012`.

[2] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968. doi:10.1109/TSSC.1968.300136.

[3] A. Stentz. Optimal and efficient path planning for partially-known environments. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 3310–3317 vol.4, 1994. doi:10.1109/ROBOT.1994.351061.

[4] S. M. LaValle. Rapidly-exploring random trees : a new tool for path planning. *The annual research report*, 1998.

[5] L. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, 1996. doi:10.1109/70.508439.

[6] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot. Informed rrt: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2997–3004, 2014. doi:10.1109/IROS.2014.6942976.

[7] M. Brunner, B. Brüggemann, and D. Schulz. Hierarchical rough terrain motion planning using an optimal sampling-based method. In *2013 IEEE International Conference on Robotics and Automation*, pages 5539–5544, 2013. doi:10.1109/ICRA.2013.6631372.

[8] J. Wang, W. Chi, C. Li, C. Wang, and M. Q.-H. Meng. Neural rrt*: Learning-based optimal path planning. *IEEE Transactions on Automation Science and Engineering*, 17(4):1748–1758, 2020. doi:10.1109/TASE.2020.2976560.

[9] L. Janson and M. Pavone. Fast marching trees: a fast marching sampling-based method for optimal motion planning in many dimensions - extended version. *CoRR*, abs/1306.3532, 2013.