Reading and Implementation Task

Autonomous Ground Vehicle Research Group

2024

1 Research Paper

For this task, you are supposed to read and implement a part of the research paper titled, "Conflict-based search for optimal multi-agent pathfinding" [1]. This research paper introduces a novel framework that deals with finding a suitable set of optimal paths for each agent a_i belonging to a set of given agents $A = \{a_i | a_i \in A\}$. Note that each agent a_i has information about its starting vertex v_i and final vertex v_f, where v_i and v_f belong to a set of vertices V present in the graph $G = (V, E)$. Further details are described in the following sections.

2 Background

A huge part of the research work that happens in autonomous driving involves path planning and controls. In fact, our research group - Autonomous Ground Vehicle: Autonomy and Intelligence, has its own Planning and Controls division in the software team. With the several successful implementations of single-agent path planning algorithms taken into consideration, a good portion of the research is now also focused on multi-agent path planning, which brings in several complications like - collisions between agents, huge memory constraints, communication delays between agents interacting with the environment and several more.

This task requires you to implement one such algorithm that employs a two-level structure, with the low-level algorithm performing single-agent path planning given certain constraints and the high-level planning that involves a branch-and-bound-like approach (you need not delve deeper into branch-and-bound algorithms if you do not already have an idea) that keeps adding constraints to nodes in a tree. For a deeper explanation, you are, of course, advised to go through the paper.

You are advised to refer to the following sections to better understand the necessary preliminaries.

2.1 Pathfinding with A*

At its core, a pathfinding method searches a graph by starting at one vertex and exploring adjacent nodes until the destination node is reached, generally to find the cheapest route. Although graph searching methods such as a breadth-first search would find a route if given enough time, other methods, which "explore" the graph, would tend to reach the destination sooner [3].

The paper you are tasked to read and implement uses a pathfinding algorithm - A*. A* is an informed search algorithm, or a best-first search, meaning that it is formulated in terms of weighted graphs: starting from a specific starting node of a graph, it aims to find a path to the given goal node having the smallest cost (least distance travelled, shortest time, etc.). It maintains a tree of paths originating at the start node

and extends those paths one edge at a time until the goal node is reached [2]. A particular method that A* employs is the heuristic search. Again, you are advised to review the referenced sources to better understand or explore and get familiar with them.

2.2 Multi-agent pathfinding

In the multi-agent pathfinding problem (MAPF), we are given a set of agents, each with respective start and goal positions. The task is to find paths for all agents while avoiding collisions [1]. The paper for implementation assumes simultaneous steps taken by all the agents at each given time step. That is, assume that the agents cannot control their speeds or wait for other agents to cross by. At each time-step t, all the agents must move simultaneously.

3 Deliverables

3.1 Understanding of the paper

Your **understanding of the core ideas** conveyed in the paper will be tested through an interview after the task round. The paper in itself is long - 27 pages, including the references. However, with careful inspection, you will be able to get an idea of which parts of it would be necessary for a thorough reading.

3.2 Implementation

You must make a clean and readable implementation of the **conflict-based search algorithm** presented in the paper. You need not implement any of the variations of the same which is listed in the paper. Your algorithm implementation must be tested and shown to be working on a **custom grid-world environment**.

Divide the implementation into these broad sub-tasks:

Task 1 - Single Agent A*: Implement the A* algorithm for a single agent

Task 2 - High level of CBS: Implement the high-level CBS - constraint tree algorithm

Task 3 - Integration with multiple agents: Integrate the high level of CBS with the low level of CBS to complete the algorithm with testing on two or more agents

3.3 Added Novelty (Optional)

If you come up with added novelties to the given algorithm or workarounds that give better results in particular use cases, you will be awarded more points for this task.

References

[1] Guni Sharon, Roni Stern, Ariel Felner, and Nathan R. Sturtevant. Conflict-based search for optimal multi-agent pathfinding. *Artificial Intelligence*, 219:40–66, 2015.

[2] Wikipedia contributors. A* search algorithm — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=A*_search_algorithm&oldid=1211575887`, 2024. [Online; accessed 10-March-2024].

[3] Wikipedia contributors. Pathfinding — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Pathfinding&oldid=1210740392`, 2024.