

Conflict-Based Search for Multi-Agent Path Finding

Woonsang Kang

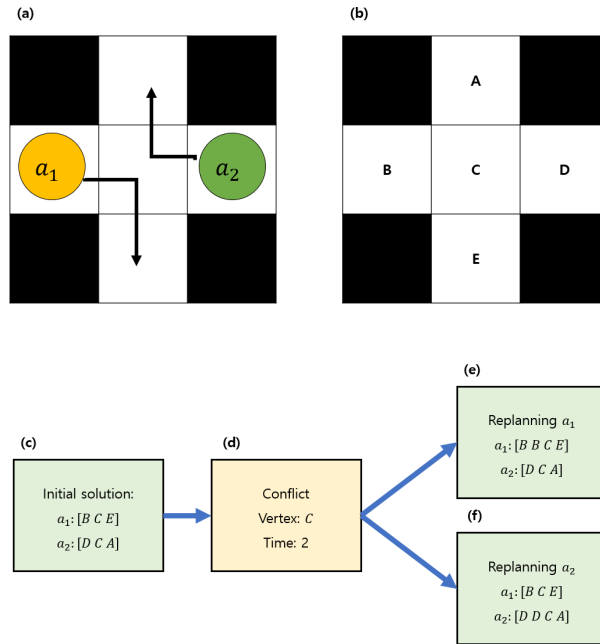
November 28, 2023

1 Introduction

Multi-agent path finding (MAPF) is a pivotal domain in the realm of robotics and artificial intelligence, focusing on the coordination of multiple autonomous agents as they navigate within a shared environment. Each agent must traverse from its starting position to a pre-defined goal, ensuring that paths do not result in any form of collision or congestion. The purpose of MAPF is not merely to avoid obstacles and conflicts. It centers on optimizing various parameters, such as minimizing total travel distance, reducing energy consumption, or balancing the workload among agents. With the rise of collaborative robotic systems in various sectors - from warehouses teeming with automated guided vehicles to urban skies dotted with delivery drones, and even hospitals with robot assistants - the necessity of MAPF becomes abundantly clear. Effective solutions in this domain not only amplify the efficiency and throughput of operations but are also paramount in ensuring safety, preventing costly damages, and fostering harmonious coexistence of multiple robots in intertwined trajectories. As our reliance on autonomous systems grows, so does the imperative for sophisticated MAPF strategies that can cater to the intricacies of diverse real-world scenarios.

2 Conflict-Based Search

Conflict-Based Search (CBS) is a two-level search algorithm used for MAPF. At the high-level, it builds a *constraint tree* (CT) where each node represents a set of constraints imposed on agents. When a conflict arises between two agents, CBS branches and creates a new (high-level) node in the CT with added constraints to resolve the conflict. At the low level, CBS uses a standard path finding algorithm, like A*, to find individual paths for agents that respect the constraints from the CT. CBS continues to iterate between the high and low levels until it either finds a solution with paths for all agents that don't conflict with each other, or determines that no such paths exist.



3 Implementation Details

You can try out the demo at the link below.

https://github.com/monni1729/HYU_mapf_fall2023