A Hierarchical Approach to Multi-Agent Path Finding

Han Zhang, Mingze Yao, Ziang Liu, Jiaoyang Li, Lucas Terr, Shao-Hung Chan, T. K. Satish Kumar and Sven Koenig
{zhan645, mingzeya, ziangliu, jiaoyanl, terr, shaohung}@usc.edu, tkskwork@gmail.com, skoenig@usc.edu

USC University of Southern California

Abstract

Solving Multi-Agent Path Finding (MAPF) instances optimally is NP-hard, and existing optimal and bounded suboptimal MAPF solvers thus usually do not scale to large MAPF instances. Greedy MAPF solvers scale to large MAPF instances, but their solution qualities are often bad. In this paper, we therefore propose a novel MAPF solver, Hierarchical Multi-Agent Path Planner (HMAPP), which creates a spatial hierarchy by partitioning the environment into multiple regions and decomposes a MAPF instance into smaller MAPF sub-instances for each region. For each sub-instance, it uses a bounded-suboptimal MAPF solver to solve it with good solution quality. Our experimental results show that HMAPP is able to solve as large MAPF instances as greedy MAPF solvers while achieving better solution qualities on various maps.

Background

Multi-Agent Path Finding (MAPF)

MAPF is the planning problem of finding a set of collision-free paths for a team of agents on a given graph. Each agent is required to move from a given start location to a given goal location.

Existing Work on Solving MAPF

Solving the MAPF problem optimally is known to be NP-hard. Thus, existing optimal and bounded-suboptimal MAPF solvers usually do not scale to large MAPF instances. Greedy MAPF solvers are able to scale to large MAPF instances, but their solution qualities are often bad.

Spatial hierarchies have not yet been used extensively for MAPF. The Spatially Distributed Multi-Agent Planner (SDP) [3] partitions a map into high-contention and low-contention regions and uses different MAPF solvers for regions of different types. Different from HMAPP, SDP does not partition the map if no high-contention regions are found. Also different from HMAPP, SDP can only solve MAPF instances in which none of the start and goal locations are inside high-contention regions.

HMAPP

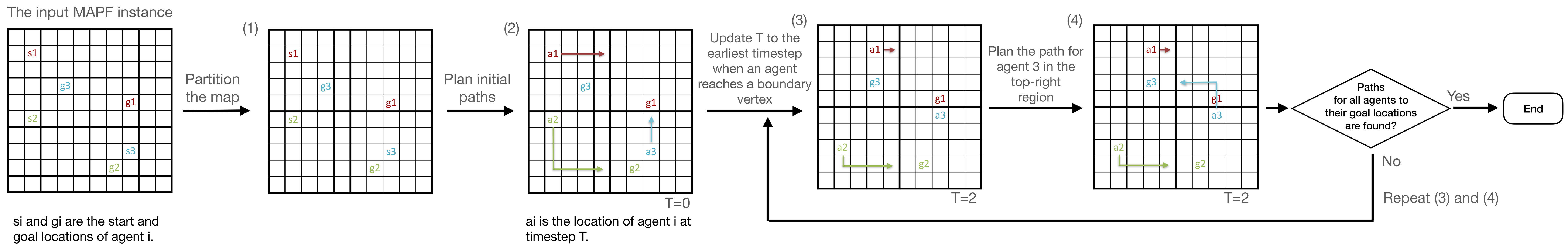

Figure 1: illustrates the operation of HMAPP. Given a MAPF instance, HMAPP first partitions the map into regions (Step (1)). A high-level planner generates the sequence of regions that an agent should visit to reach its goal location. Each regional planner initially plans a set of collision-free sub-paths for all agents in its region (Step(2)). When an agent reaches the boundary location to its next region, the regional planner of the next region replans a set of collision-free sub-paths for this agent (Steps (3) and (4)). HMAPP repeats Steps (3) and (4) until all agents have reached their goal locations. HMAPP is not a complete MAPF solver.

Experimental Results

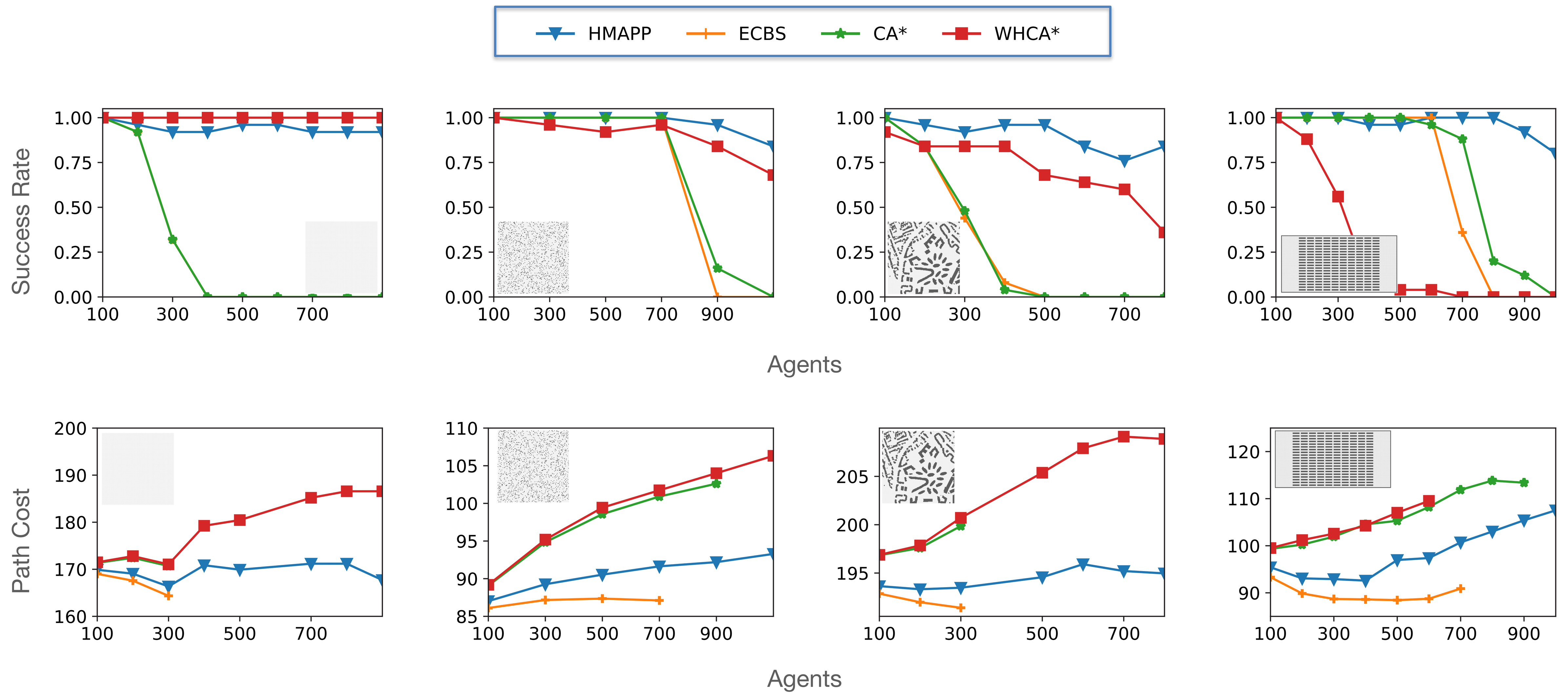

Figure 2: shows the success rate (= percentage of solved instances within 5 minutes) and average path cost.

We compared HMAPP with CA* [2], WHCA* [2] and ECBS [1] on different grids. We evaluated all MAPF solvers on four grids: (a) the 256×256 empty grid, (b) a 128×128 grid with 10% randomly blocked vertices, (c) Paris_1_256 and (d) warehouse-10-20-10-2-2. Our experimental results show that HMAPP is able to solve as large MAPF instances as greedy MAPF solvers while achieving better solution qualities.

Future Work

HMAPP is a general algorithmic framework that can use different methods for partitioning the map, planning the sequence of regions that an agent needs to visit and planning paths for agents inside each region. The current version of HMAPP still uses naive approaches for map partitioning and high-level planning Future work includes: (1) enhancing HMAPP with theoretical guarantees; (2) studying how the map partitioning affects runtime and solution quality of HMAPP; and (3) taking potential congestion into account for high-level planning.

Acknowledgments

The research at the University of Southern California was supported by the National Science Foundation (NSF) under grant numbers 1409987, 1724392, 1817189, 1837779 and 1935712, as well as a gift from Amazon.

[1] Max Barer, Guni Sharon, Roni Stern, and Ariel Felner. Suboptimal variants of the conflict-based search algorithm for the multi-agent pathfinding problem. In *International Symposium on Combinatorial Search (SoCS)*, pages 19–27, 2014.
[2] David Silver. Cooperative pathfinding. In *AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, pages 117–122, 2005.
[3] Christopher Makoto Wilt and Adi Botea. Spatially distributed multiagent path planning. In *International Conference on Automated Planning and Scheduling (ICAPS)*, page 332–340, 2014.