# ESO-MAPF: Bridging Discrete Planning and Continuous Execution in Multi-Agent Pathfinding

**Ján Chudý and Pavel Surynek**
Czech Technical University in Prague
Faculty of Information Technology
Thaákurova 9, 160 00 Praha 6, Czechia
{chudyja1, pavel.surynek}@fit.cvut.cz

**Abstract**

We present ESO-MAPF, a research and educational platform for experimenting with multi-agent path finding (MAPF). ESO-MAPF focuses on demonstrating the planning-acting chain in the MAPF domain. MAPF is the task of finding collision-free paths for agents from their starting positions to given individual goals. The standard MAPF uses the abstraction where agents move in an undirected graph via traversing its edges in discrete steps. The discrete abstraction simplifies the planning phase; however, resulting discrete plans often need to be executed in the real continuous environment. ESO-MAPF shows how to bridge discrete planning and the acting phase in which the resulting plans are executed on physical robots. We simulate centralized plans on a group of OZOBOT Evo robots using their reflex functionalities and outputs on the surface of the screen that serves as the environment. Various problems arising along the planning-acting chain are illustrated to emphasize the educational point of view.

## Introduction

Multi-agent path finding (MAPF) (Silver 2005; Ryan 2008; Standley 2010) is an abstraction for many real-life problems where agents, both autonomous or passive, need to be moved from their starting positions to given individual goal positions (see (Felner et al. 2017) for a survey). The environment in MAPF is modeled as an undirected graph where vertices represent positions and edges define the topology [1].

The standard variant of MAPF assumes that each agent starts in a given starting vertex and its task is to reach unique individual goal vertex. Such formalization encompasses many real-life navigation tasks (Cáp et al. 2013; Ma et al. 2017). Despite recent considerable progress in MAPF planning algorithms both optimal and sub-optimal with respect to various objectives (Surynek 2019a; Gange, Harabor, and Stuckey 2019; Li et al. 2020), little attention is devoted to the acting and robotic aspect of the problem though exceptions exist (Yakovlev, Andreychuk, and Vorobyev 2019).

We address the MAPF problem in a broad sense in this work. We take into account the entire intelligent multi-agent system construction from the **thinking** level represented by AI planning to the **acting** level (Ghallab, Nau, and Traverso 2016) represented by the execution of the plan originating at the thinking level by a group of physical robots - OZOBOTs Evo in our case (Evollve 2020).

There is still an open question for the MAPF research community regarding how the planning-acting chain should be balanced. We contribute by a specific case in which we are trying to keep the **planning (thinking)** phase simpler by adopting relatively strong abstraction where agents (representing robots) are discrete items placed in vertices of the graph and moved between discrete timesteps while the **acting phase** is more complex and converts the discrete plan to a continuous stream of reflex commands (represented as an animated curve on the surface of the screen).

On the other hand, there are planners for MAPF that consider continuous space and time and produce continuous plans (Andreychuk et al. 2019; Surynek 2020). Using such a planner could lead to a simpler acting phase, but a concrete study has not yet been done. Our observation is that the bottleneck in terms of the performance is rather on the planner's side, which justifies the usage of the discrete MAPF.

### Contribution

We introduce ESO-MAPF (ESO = Environment Surface Outputs), a research and educational platform for experimenting with the **planning-acting chain** in the MAPF domain. Unlike previous attempts to deploy MAPF plans on a group of robots that compile discrete plans directly into the sequence of commands for robots (Barták et al. 2018), the ESO-MAPF platform relies on the reflex-based behavior of OZOBOT Evo robots that by default follow the line drawn on the surface. ESO-MAPF uses the surface of the screen (any standard type of a flat display can be used) as the environment for agents and draws curves to control robots in real-time. This approach is less prone to errors in the execution, as shown in our related study (Chudý, Popov, and Surynek 2020) and is more flexible and modular.

## Background

Agents in *multi-agent path finding* (MAPF) are placed in vertices of an undirected graph so that there is at most one agent per vertex. Formally, $s : A \mapsto V$ is a *configuration* of agents in vertices of the graph. A configuration can be



[1] An almost identical setup is used in *graph pebbling* (Kornhauser, Wilensky, and Rand 2009; Parberry 2015; Ratner and Warmuth 1990) where however the focus is instead on computational complexity issues and theory.

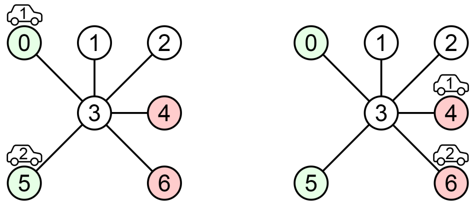


Figure 1: A MAPF instance with two agents.

transformed instantaneously to the next one by *valid* movements of agents; the next configuration corresponds to the next discrete *time step*. An agent can move into another vertex across an edge provided the target vertex is vacant or being simultaneously vacated by another agent. A *collision* occurs if two agents appear in the same vertex (*vertex collision*) or if two agents simultaneously cross the same edge in opposite directions (*edge collision*) (Sharon et al. 2015). The formal definition of MAPF follows, and an illustration is shown in Figure 1.

**Definition 1** Multi-agent path finding *(MAPF) is a 4-tuple* $(G = (V, E), A, s_0, g)$ *where* $G = (V, E)$ *is an undirected graph,* $A = \{a_1, a_2, ..., a_k\}$ *with* $k \in \mathbb{N}$ *is a set of agents where* $k \leq |V|$, $s : A \mapsto V$ *represents agents' starting vertices, and* $g : A \mapsto V$ *assigns each agent a goal vertex.*

We often aim to optimize various cumulative objectives in MAPF, the commonly used *sum-of-costs* (SoC) assigns each move (edge traversal) and wait for action a unit cost. SoC accumulates these costs for all $k$ agents. Obtaining SoC optimal solution in MAPF is an NP-hard problem (Ratner and Warmuth 1990).

## ESO-MAPF: a Planning-Acting Platform

The ESO-MAPF platform consists of two physical components - a group of robots and a flat-screen representing the environment. The current version supports environments defined by 4-connected grids with obstacles (see Figure 2).

Operationally, ESO-MAPF supports experiments on the following three stages of the planning-acting chain:

1. discrete **planning** - represented by SoC-optimal SAT-based planner (Surynek 2019b)
2. **conversion** of the **discrete plan** into a **continuous schedule** - discrete actions are substituted by sequences of primitive actions in the context-sensitive way resulting in a continuous schedule
3. **physical acting** and **sensing** - represented by reflex-based execution of the continuous plan by a group of differential drive OZOBOT Evo robots

The environment is displayed on the screen so that the surface is oriented horizontally, and robots can move on top of the screen's surface. The execution phase consists of animating path segments for individual robots that follow them using their predefined reflex-based behavior.

The substitution of edge traversal with primitive actions that are represented by path segments to be animated must take into account the context of the plan (that is, previous and following actions). For example, visiting three consecutive vertices once represents a curved turn, in the other case, it may be a straight line. A wait action (the robot is parked for a specific time) followed by the reverse of the previous action must be substituted by rotation of the robot etc.

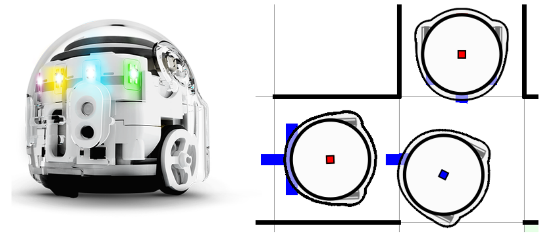

Figure 2: OZOBOT Evo robot (left) and a group of robots following animated curves in a grid maze - top view (right).

To mitigate desynchronization among robots and make the execution less prone to failure, colored path segments that indicate the robot's target speed are used to keep the robot where it is supposed to be. For example, if the robot goes faster than expected, it is slowed down and vice versa. This is done using a built-in sensor array of OZOBOTs.

### ESO-MAPF as an Education Platform

As an education platform, ESO-MAPF can illustrate to students all phases of the planning-acting chain. Moreover, problems at each stage can be demonstrated and ways how to mitigate them. At the planning stage, one can show an instance that cannot be solved optimally in a reasonable time (despite all simplifications and discretization, optimal MAPF remains an NP-hard problem). The conversion phase should use suitable substitution with primitive actions; not all substitutions work well in practice (such as displaying full path). Finally, the acting and sensing phase is prone to various failures, from losing the animated curve to a physical collision between robots.

### ESO-MAPF as a Research Platform

ESO-MAPF can support scalability tests in the acting phase in terms of the number of robots. The development of planning models that are less prone to various failures is important for practice and cannot be fully simulated in computers. Deploying the entire planning-acting chain provides a basis for discovering the failures of a priori unknown types.

## Conclusion

The ESO-MAPF[2] platform represents an affordable planning-acting chain deployment for the MAPF domain. All components (robots and the screen) are standard commercial products, but their combination with the software part provides a powerful tool capable of demonstrating important issues of planning and acting in MAPF[3]. For future work, we plan to re-balance the complexity of the planning stage, where a continuous planner will be used.

[2]This work has been supported by GAČR - the Czech Science Foundation, under the grant number 19-17966S.

[3]Video available on https://tinyurl.com/y2zk9tus.

# References


Andreychuk, A.; Yakovlev, K. S.; Atzmon, D.; and Stern, R. 2019. Multi-Agent Pathfinding with Continuous Time. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, 39–45. ijcai.org.

Barták, R.; Svancara, J.; Skopková, V.; and Nohejl, D. 2018. Multi-agent Path Finding on Real Robots: First Experience with Ozobots. In *Advances in Artificial Intelligence - IBERAMIA 2018 - 16th Ibero-American Conference on AI, Trujillo, Peru, November 13-16, 2018, Proceedings*, volume 11238 of *Lecture Notes in Computer Science*, 290–301. Springer.

Cáp, M.; Novák, P.; Vokrínek, J.; and Pechoucek, M. 2013. Multi-agent RRT: sampling-based cooperative pathfinding. In Gini, M. L.; Shehory, O.; Ito, T.; and Jonker, C. M., eds., *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS 2013*, 1263–1264. IFAAMAS.

Chudý, J.; Popov, N.; and Surynek, P. 2020. Emulating Centralized Control in Multi-Agent Pathfinding Using Decentralized Swarm of Reflex-Based Robots. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, SMC 2020*. IEEE Computer Society, in press.

Evollve, I. 2020. Ozobot. URL https://ozobot.com/.

Felner, A.; Stern, R.; Shimony, S. E.; Boyarski, E.; Goldenberg, M.; Sharon, G.; Sturtevant, N. R.; Wagner, G.; and Surynek, P. 2017. Search-Based Optimal Solvers for the Multi-Agent Pathfinding Problem: Summary and Challenges. In *Proceedings of the Tenth International Symposium on Combinatorial Search, SOCS 2017*, 29–37. AAAI Press.

Gange, G.; Harabor, D.; and Stuckey, P. J. 2019. Lazy CBS: Implicit Conflict-Based Search Using Lazy Clause Generation. In *Proceedings of the Twenty-Ninth International Conference on Automated Planning and Scheduling, ICAPS 2018, Berkeley, CA, USA, July 11-15, 2019*, 155–162. AAAI Press.

Ghallab, M.; Nau, D. S.; and Traverso, P. 2016. *Automated Planning and Acting*. Cambridge University Press. ISBN 978-1-107-03727-4.

Kornhauser, D.; Wilensky, U.; and Rand, W. 2009. Design Guidelines for Agent Based Model Visualization. *J. Artif. Soc. Soc. Simul.* 12(2).

Li, J.; Tinka, A.; Kiesel, S.; Durham, J. W.; Kumar, T. K. S.; and Koenig, S. 2020. Lifelong Multi-Agent Path Finding in Large-Scale Warehouses. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS '20, Auckland, New Zealand, May 9-13, 2020*, 1898–1900. International Foundation for Autonomous Agents and Multiagent Systems.

Ma, H.; Koenig, S.; Ayanian, N.; Cohen, L.; Hönig, W.; Kumar, T. K. S.; Uras, T.; Xu, H.; Tovey, C. A.; and Sharon, G. 2017. Overview: Generalizations of Multi-Agent Path Finding to Real-World Scenarios. *CoRR* abs/1702.05515.

Parberry, I. 2015. Solving the ($n^2$ - 1)-Puzzle with 8/3 $n^3$ Expected Moves. *Algorithms* 8(3): 459–465.

Ratner, D.; and Warmuth, M. K. 1990. NxN Puzzle and Related Relocation Problem. *J. Symb. Comput.* 10(2): 111–138.

Ryan, M. R. K. 2008. Exploiting Subgraph Structure in Multi-Robot Path Planning. *J. Artif. Intell. Res.* 31: 497–542.

Sharon, G.; Stern, R.; Felner, A.; and Sturtevant, N. R. 2015. Conflict-based search for optimal multi-agent pathfinding. *Artif. Intell.* 219: 40–66.

Silver, D. 2005. Cooperative Pathfinding. In *Proceedings of the First Artificial Intelligence and Interactive Digital Entertainment Conference*, 117–122. AAAI Press.

Standley, T. S. 2010. Finding Optimal Solutions to Cooperative Pathfinding Problems. In Fox, M.; and Poole, D., eds., *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010*. AAAI Press.

Surynek, P. 2019a. Lazy Compilation of Variants of Multi-robot Path Planning with Satisfiability Modulo Theory (SMT) Approach. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2019, Macau, SAR, China, November 3-8, 2019*, 3282–3287. IEEE.

Surynek, P. 2019b. Unifying Search-based and Compilation-based Approaches to Multi-agent Path Finding through Satisfiability Modulo Theories. In Kraus, S., ed., *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019*, 1177–1183. ijcai.org.

Surynek, P. 2020. On Satisfisfiability Modulo Theories in Continuous Multi-Agent Path Finding: Compilation-based and Search-based Approaches Compared. In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, ICAART 2020, Volume 2, Valletta, Malta, February 22-24, 2020*, 182–193. SCITEPRESS.

Yakovlev, K. S.; Andreychuk, A.; and Vorobyev, V. 2019. Prioritized Multi-agent Path Finding for Differential Drive Robots. In *2019 European Conference on Mobile Robots, ECMR 2019, Prague, Czech Republic, September 4-6, 2019*, 1–6. IEEE.