Towards Streamlined Learning and Search for Multi-Agent Optimization

Thomy Phan
University of Bayreuth, Germany
thomy.phan@uni-bayreuth.de

Abstract

Many real-world problems can be modeled as cooperative multi-agent systems (MAS), such as fleet management, industrial operations, and communication networks, where multiple agents collaborate to optimize a shared objective. Optimizing cooperative MAS is difficult due to the combinatorial nature of joint actions and environmental factors. Thus, many practical multi-agent optimization approaches specialize in particular problem classes to exploit structural properties for effective and efficient optimization. Unfortunately, such specializations can lead to complex and inflexible methods that cannot be seamlessly combined or transferred to novel domains without substantial engineering effort. In this paper, we advocate an approach towards streamlined learning and search for multi-agent optimization. Focusing on multi-agent path finding as an exemplary problem, we propose to simplify two popular approaches to MAPF, namely multi-agent reinforcement learning and adaptive search. Through these simplifications, we aim to enable seamless combination and transferability of our methods without substantial engineering.

1 Introduction

Many real-world problems can be modeled as cooperative multi-agent systems (MAS), such as fleet management, industrial operations, and communication networks, where $N \geq 2$ agents collaborate to optimize a shared objective [Foerster *et al.*, 2018; Oliehoek and Amato, 2016].

Optimizing cooperative MAS is difficult due to the combinatorial nature of joint actions and environmental factors (e.g., obstacles and communication channels), in particular constraints (e.g., deadlines and collision avoidance) [Boutilier, 1996; Oliehoek and Amato, 2016; Sharon *et al.*, 2015]. Thus, for practical feasibility, many *multi-agent optimization* approaches use learning and/or search methods to find suboptimal solutions that satisfy all constraints [Fioretto *et al.*, 2018; Oliehoek and Amato, 2016; Stern *et al.*, 2019].

Decentralized POMDPs (Dec-POMDPs) represent a general framework for multi-agent optimization [Oliehoek and Amato, 2016]. However, Dec-POMDP solvers based on learning and/or search do not scale to large numbers of agents in practice due to the black box nature of Dec-POMDPs, which ignores important structural properties of the underlying problem, such as spatial relationships between agents [Fioretto *et al.*, 2018; Phan *et al.*, 2021]. Thus, many practical solvers focus on special cases to exploit such structures for better efficiency and scalability [Fioretto *et al.*, 2018].

Unfortunately, such specializations can lead to complex and inflexible learning or search approaches that cannot be seamlessly combined or transferred to novel domains without substantial engineering effort [Phan *et al.*, 2025; Phan *et al.*, 2026b]. We argue that simplifying common learning and search approaches could mitigate tight specialization, while improving the combination and transferability of multi-agent optimization methods [Phan *et al.*, 2024a; Cai *et al.*, 2025]. Furthermore, they can improve the overall usability of these methods without requiring substantial tuning or a deep understanding of hyperparameters.

In this paper, we advocate an approach towards streamlined learning and search for multi-agent optimization. Focusing on *multi-agent path finding (MAPF)* as an exemplary optimization problem, we propose to simplify two popular approaches to MAPF, namely multi-agent reinforcement learning and adaptive search. In the former, we will introduce methods to scale and simplify learning in MAPF with hierarchical value factorization and auto-curricula for structured exploration. In the latter, we will introduce methods to simplify search-based MAPF solvers with adaptive mechanisms, substantially improving their effectiveness and efficiency. Through these simplifications, we aim to enable seamless combination and transferability of our methods without substantial engineering [Cai *et al.*, 2025].

2 Background: Multi-Agent Path Finding

We focus on *multi-agent path finding (MAPF)* as an exemplary optimization problem. The goal in MAPF is to find collision-free paths for $N \geq 2$ agents with assigned start and goal locations [Stern *et al.*, 2019; Sharon *et al.*, 2015]. Solving MAPF is highly relevant to various applications, including warehouse operations [Li *et al.*, 2021d], smart manufacturing [Phan *et al.*, 2020], multi-robot systems [Zhang *et al.*, 2023], and search & rescue missions [Jennings *et al.*, 1997]. However, finding optimal solutions, i.e., feasible paths with

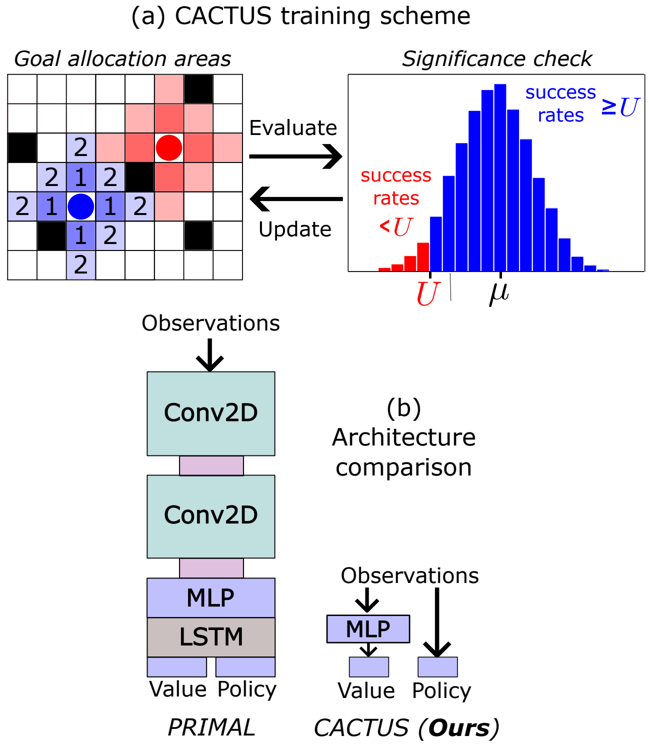

Figure 1: **(a)** Curriculum scheme of CACTUS [Phan *et al.*, 2026b]. The agents (colored circles) are trained and evaluated w.r.t. the shaded goal-allocation areas around the agents. When the average completion rate μ exceeds a curriculum threshold $U \in (0, 1)$ with a specified confidence level, allocation areas are expanded. The numbers in the blue cells indicate the shortest-path distance to the blue agent. **(b)** Architectural comparison between the engineering-focused PRIMAL [Sartoretti *et al.*, 2019] and the much simpler CACTUS [Phan *et al.*, 2026b].

minimal costs, i.e., agent travel times, is NP-hard and thus intractable for large-scale instances [Sharon *et al.*, 2015].

State-of-the-art MAPF solvers are *search-based*, using information about all agents to find *collision-free plans* [Sharon *et al.*, 2015]. The search algorithms can be *optimal or bounded suboptimal* (e.g., CBS, ECBS, and EECBS), with guarantees about the travel time (sub-)optimality, but exponential time scaling [Sharon *et al.*, 2015; Barer *et al.*, 2021; Li *et al.*, 2021c], or *unbounded suboptimal* (e.g., LaCAM, MAPF-LNS, MAPF-LNS2), with polynomial time scaling but arbitrarily high solution costs [Okumura, 2023; Li *et al.*, 2021a; Li *et al.*, 2022]. Search-based MAPF solvers commonly depend on handcrafted guidance [Okumura *et al.*, 2022; Li *et al.*, 2021a; Li *et al.*, 2022] for domain-specific aspects, e.g., 4-neighbor grid worlds [Li *et al.*, 2021b] and corridor maps [Cohen and Koenig, 2016], which do not generalize to other scenarios [Phan *et al.*, 2024c].

3 Multi-Agent Reinforcement Learning (RL)

Most search-based MAPF solvers are centralized due to their dependence on global information, i.e., the paths and constraints of all agents. This dependence requires reliable, centralized communication with substantial bandwidth for large-scale systems, which is a limiting factor in practice for scalability and robustness [Oliehoek and Amato, 2016]. Even fast unbounded suboptimal solvers like LaCAM and MAPF-LNS suffer from this limitation, which impedes their large-scale deployment beyond spatially confined applications such as warehouses [Moldagalieva *et al.*, 2026; Nagai and Okumura, 2026; Phan *et al.*, 2026b].

Multi-agent reinforcement learning (RL) has become popular for learning decentralized policies via trial-and-error from rewards, enabling scalable control under partial observability and limited communication [Foerster *et al.*, 2018]. We focus on *cooperative settings*, where all agents obtain a *team reward* $r_t \in \mathbb{R}$ for executing a joint action a_t in state s_t at time step t, without further indication of their individual contributions [Oliehoek and Amato, 2016]. The state of the art in cooperative multi-agent RL is based on the *centralized training for decentralized execution (CTDE)* paradigm, where all policies are trained in a controlled environment, such as a laboratory or simulator, to exploit global information for learning coordinated behavior. After training, the policies can be executed independently in a decentralized manner, requiring no (or just limited) communication [Foerster *et al.*, 2018].

3.1 Deep Multi-Agent RL

Deep multi-agent RL uses neural networks to learn policies for high-dimensional state and observation spaces [Lowe *et al.*, 2017; Foerster *et al.*, 2018; Rashid *et al.*, 2018]. *Multi-agent credit assignment* is a major challenge in cooperative settings due to the lack of explicit indication of individual agents' contributions to the team's success or failure [Foerster *et al.*, 2018; Rashid *et al.*, 2018]. *Value factorization* is a popular credit assignment approach, where a value function $Q_{tot}(s_t, a_t) = \mathbb{E}[\sum_t^T r_t]$, learned from the shared rewards r_t, is decomposed into *agent-wise utilities* $\langle Q_i \rangle_{1 \leq i \leq N}$ using some *factorization operator* $\Psi(\langle Q_i \rangle_{1 \leq i \leq N}) = Q_{tot}$. The factorization operator is often implemented as constrained neural network that enforces consistency between the optimization of Q_{tot} and the separate optimization of the individual utilities Q_i [Rashid *et al.*, 2018; Son *et al.*, 2019]. The resulting utilities can be directly used for decentralized execution [Rashid *et al.*, 2018] or to train policies via actor-critic techniques [Lowe *et al.*, 2017; Foerster *et al.*, 2018].

Despite value factorization representing a theoretically general method, its effectiveness has been demonstrated only in domains with a handful of agents [Rashid *et al.*, 2018; Phan *et al.*, 2023]. In large-scale domains with hundreds or thousands of agents, which are common in the MAPF literature [Okumura, 2023; Li *et al.*, 2021a; Li *et al.*, 2022], the factorization operator Ψ can become a bottleneck, where flat credit assignment schemes are insufficient for effective coordination, falling short compared to unbounded suboptimal search-based solvers like LaCAM and MAPF-LNS.

To this end, we devised the hierarchical factorization method VAST [Phan *et al.*, 2021], which uses a bi-level decomposition scheme, where the factorization operator Ψ decomposes the value function Q_{tot} into *group-wise utilities* $Q_k = \sum_i Q_{k,i}$, which are then further decomposed into agent-wise utilities $Q_{k,i}$ for the respective members i of group k. The agent-wise decomposition is done with a linear operator, enabling group variability across time steps to adapt to different situations. VAST can provably maintain the consistency between optimizing Q_{tot} and Q_i, regardless of the sub-team definition, supporting its flexibility and adaptability. The groups or sub-teams can be assigned using domain-specific methods, such as spatial clustering, or learned via meta-gradient methods to optimize effectiveness in an outer loop [Phan *et al.*, 2021]. We evaluated VAST in MAPF-related domains, such as a warehouse and a battle domain, demonstrating better scalability, sample efficiency, and effectiveness than state-of-the-art value factorization methods [Rashid *et al.*, 2018; Son *et al.*, 2019].

3.2 Curricula for Structured Exploration

The state of the art in multi-agent RL is primarily designed for Dec-POMDPs, which are more general but also more complex than MAPF, as they ignore structural properties, such as spatial relationships among agents. This lack of structural consideration severely limits sample efficiency with random exploration, requiring a considerable number of trials, which is computationally expensive [Phan *et al.*, 2026b].

To mitigate this exploration problem, most prior work relies on additional support from search-based solvers [Sartoretti *et al.*, 2019; Skrynnik *et al.*, 2024], e.g., for imitation learning of path finding or collision avoidance. However, due to the combinatorial space of MAPF w.r.t. the number of joint actions, potential paths, and environments, the learned models must have sufficient capacity to memorize all important aspects of MAPF. This has led to unreasonably large neural networks with complex architectures [Sartoretti *et al.*, 2019] (Figure 1b), compared with simpler architectures that are common in the multi-agent RL literature [Foerster *et al.*, 2018; Rashid *et al.*, 2018]. Another way to shape learning is to use handcrafted reward functions [Alkazzi and Okumura, 2024], which may work in specific scenarios but could lead to unintended side effects when transferred to another domain [Foerster *et al.*, 2018]. [Alkazzi and Okumura, 2024] and [Phan *et al.*, 2025] provide overviews of various aspects of machine learning that are often engineered substantially to achieve empirical success of multi-agent RL-based MAPF.

The extensive engineering confirms the difficulty of exploiting the structural properties of MAPF to improve exploration efficiency and learning of multi-agent RL. To this end, we devised *MAPF curricula for structured exploration* that enable efficient exploration and learning while simplifying the overall multi-agent RL approach to MAPF (Figure 1b).

CACTUS is the *first auto-curriculum approach to MAPF*. It uses a *reverse curriculum scheme* by controlling the goal allocation radius for all agents [Phan *et al.*, 2026b]. Starting with a radius of 1, CACTUS gradually increases the goal-allocation area around each agent by incrementing the radius (Figure 1a) if the success rate of all agents reaching their respective goals improves significantly based on a statistical test. CACTUS substantially simplifies multi-agent RL for MAPF w.r.t. the training scheme, the model architecture (Figure 1b), and the reward definition, which simply penalizes each time step for each agent that has not yet reached its goal. CACTUS significantly improves effectiveness and efficiency, requiring less than 5% of the training effort of prior engineering-focused works, like PRIMAL [Sartoretti *et al.*, 2019], in terms of data, compute, and parameters.

We conducted preliminary analyses, proving that CACTUS improves the sample efficiency over other multi-agent RL methods that rely on naive random exploration, and devised an extension to CACTUS, called C^{3}LARA, which incentivizes goal allocations at intersecting allocation areas for more robust collision avoidance, without requiring additional support of a search-based solver [Phan *et al.*, 2026b].

We also explored curricula based on environment generation using variational autoencoders and cross-entropy optimization [Phan *et al.*, 2025], as well as dynamic agent grouping based on the intersection of goal-allocation areas [Phan and Koenig, 2026], inspired by VAST.

Our work on curriculum learning demonstrates how simplification can advance the state of the art in RL-based MAPF in terms of effectiveness and efficiency, as well as usability due to fewer tunable hyperparameters. This insight is important for investigating further algorithmic aspects of multi-agent RL to provide a basis for transferring our methods to novel domains without substantial engineering.

4 Adaptive Search-Based Solvers

Search-based solvers are the state of the art in MAPF in terms of effectiveness [Sharon *et al.*, 2015; Barer *et al.*, 2021; Li *et al.*, 2021c] and efficiency [Okumura, 2023; Li *et al.*, 2021a; Li *et al.*, 2022]. These solvers are often enhanced with structural knowledge through handcrafted guidance, which can improve their effectiveness and efficiency for specific MAPF scenarios, such as 2D grid worlds [Stern *et al.*, 2019].

In practice, unbounded suboptimal solvers are preferred to meet real-time requirements and to enable fast replanning in the event of unexpected changes. More recently, these solvers have become popular to quickly provide "expert" data for supervised machine learning of policies or guidance/heuristic functions [Huang *et al.*, 2022; Veerapaneni *et al.*, 2025; Jiang *et al.*, 2025; Andreychuk *et al.*, 2025].

MAPF-LNS, based on *large neighborhood search (LNS)*, is one example of an unbounded suboptimal solver that can find solutions for instances with hundreds of agents [Li *et al.*, 2021a; Huang *et al.*, 2022]. Given an *initial solution* and a set of *destroy heuristics*, MAPF-LNS iteratively destroys and replans a fixed number of paths, replacing the current solution if the new plan has a lower cost. Since MAPF-LNS creates a collision-free plan at each iteration, it can be used in an *anytime manner* and terminated once a time budget runs out. The original MAPF-LNS from [Li *et al.*, 2021a] employs three handcrafted heuristics, focusing on the most-delayed agent, the most-occupied location, and a random neighborhood, respectively. An adaptive selection mechanism chooses one of these destroy heuristics at each iteration (Figure 2a).

In the following, we will focus on MAPF-LNS, since LNS is used to solve related NP-hard problems like vehicle routing, scheduling problems, and mixed-integer linear programming [Phan *et al.*, 2024a; Cai *et al.*, 2025]. Thus, by simplifying and improving MAPF-LNS, we hope to lay the foundation for seamless transfer of our methods to other problems.

Due to the handcrafted destroy heuristics, MAPF-LNS

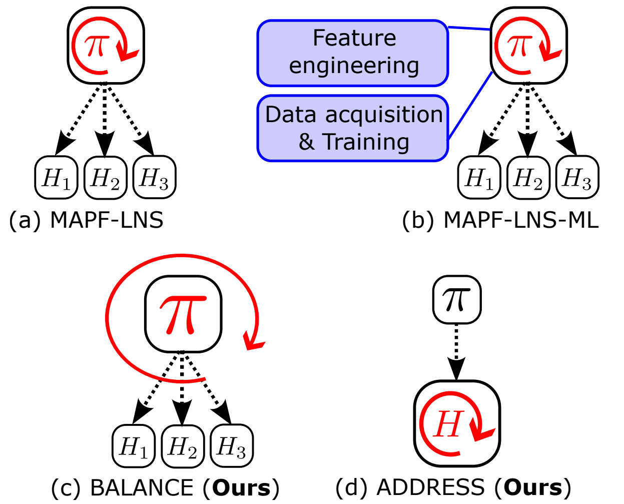

Figure 2: Schematic comparison of different MAPF-LNS schemes. **(a)** The original MAPF-LNS using three handcrafted destroy heuristics H_1, H_2, and H_3, and an adaptive selection method π. **(b)** MAPF-LNS-ML integrates machine-learning-based guidance, requiring additional feature engineering, data acquisition, and training of the guidance model. **(c)** BALANCE uses a bi-level multi-armed bandit π approach to automatically select hyperparameters and destroy heuristics without extensive prior tuning, training, or engineering effort. **(d)** ADDRESS uses a single adaptive destroy heuristic.

lacks generality, limiting its transferability to other domains. MAPF-LNS-ML is a *machine-learning-enhanced version* of MAPF-LNS, which selects neighborhoods via a scoring model, predicting their solution improvement potential [Huang *et al.*, 2022]. The scoring model is trained on data generated by the original MAPF-LNS from a selected set of instances. While machine learning seems appealing due to its generalizing capability, the overall approach is still handcrafted and labor-intensive due to feature engineering and manual selection of training instances, as well as expensive and inflexible due to hyperparameter tuning and offline training of the scoring model without further adaptation [Phan *et al.*, 2024a; Phan *et al.*, 2024c] (Figure 2b).

We devised an adaptive variant, called BALANCE, which uses a *bi-level multi-armed bandit* to optimize the destroy heuristic and neighborhood size selection (Figure 2c). The bandits are trained online to minimize the total costs of the current MAPF-LNS solution, without any feature engineering, hyperparameter tuning, or offline (pre-)training. BALANCE demonstrated delay improvements by at least 50% compared with MAPF-LNS and MAPF-LNS-ML across common benchmark instances [Stern *et al.*, 2019].

ADDRESS further simplifies BALANCE by employing a *single adaptive destroy heuristic* that depends on a seed agent selected via a multi-armed bandit [Phan *et al.*, 2024c]. A neighborhood is generated by randomly selecting agents in close proximity to the seed agent. Despite the substantial simplification, ADDRESS was shown to further reduce delays by at least 50% in large-scale instances with $N = 1000$ agents.

Similar to our work on curriculum learning, our adaptive search approaches also demonstrate that simplification can advance the state of the art in search-based MAPF in terms of effectiveness, efficiency, and usability. This is important for further investigating suboptimal MAPF solvers and for providing a basis for seamless combination and transferability of our methods without substantial engineering. Our adaptive search could also provide high-quality data to machine learning approaches, improving their effectiveness.

5 Conclusion and Future Work

We advocate an approach to streamlined learning and search for multi-agent optimization, focusing on MAPF as an exemplary problem. We proposed to simplify two popular approaches to MAPF, namely multi-agent RL and adaptive search. In the former, we introduced methods to scale and simplify learning in MAPF with hierarchical value factorization for scalable training and auto-curricula for efficient structured exploration. In the latter, we introduced methods to simplify search-based MAPF solvers with adaptive mechanisms, substantially improving their effectiveness and efficiency. Through these simplifications, we aim to enable seamless combination and transferability of our methods without substantial engineering, as demonstrated by the transfer of our adaptive solver BALANCE [Phan *et al.*, 2024a] to mixed-integer linear programming [Cai *et al.*, 2025].

In the future, we plan to pursue the following directions:

- **Combining learning and search**: So far, combining learning and search required substantial engineering effort, largely due to incompatibilities between the methods. Given our simplified machinery, we plan to identify adequate connecting points for more principled and straightforward combinations.
- **Addressing information asymmetry**: Centralized and decentralized optimization approaches operate with different information views. This information asymmetry can lead to emergent side-effects, which may occur in CTDE-based multi-agent RL and decentralized policy learning from centralized MAPF solvers. We plan to address this issue with recurrence-based multi-agent RL [Phan *et al.*, 2023] and counterfactual reasoning [Bareinboim *et al.*, 2015; Li *et al.*, 2025; Phan *et al.*, 2026a] for multi-agent RL and adaptive search-based solvers.
- **Decentralized learning and search**: We plan to investigate decentralizing our current CTDE schemes and adaptive search-based solvers. In multi-agent RL, we will consider peer-incentivization methods for decentralized learning of cooperative behavior [Phan *et al.*, 2024b; Altmann *et al.*, 2026]. For adaptive search, we will consider enhancing decentralized search algorithms with factorized utilities as heuristics [Phan *et al.*, 2018].
- **Robustness**: Most multi-agent optimization methods focus on idealized scenarios, in which agents and environments do not change adversarially. However, in real-world applications, failures, manipulations, and environmental changes are expected and thus must be considered. We plan to leverage adversarial and adaptive learning for robust decentralized policies [Phan *et al.*, 2020] and fast but effective replanning [Phan *et al.*, 2024c].

Acknowledgments

The research was supported by the National Science Foundation (NSF) under grant numbers 2112533 and 2321786. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the sponsoring organizations, agencies, or the U.S. government. Special thanks go to Sven Koenig (University of California, Irvine) for his substantial guidance and support.

References

[Alkazzi and Okumura, 2024] Jean-Marc Alkazzi and Keisuke Okumura. A Comprehensive Review on Leveraging Machine Learning for Multi-Agent Path Finding. *IEEE Access*, 12:57390–57409, 2024.

[Altmann *et al.*, 2026] Philipp Altmann, Maximilian Zorn, Sven Koenig, and Thomy Phan. Dynamic Reward Incentives for Emergent Cooperation under Changing Rewards. *Transactions on Machine Learning Research (TMLR)*, 2026. To appear.

[Andreychuk *et al.*, 2025] Anton Andreychuk, Konstantin Yakovlev, et al. MAPF-GPT: Imitation Learning for Multi-Agent Pathfinding at Scale. *AAAI Conference on Artificial Intelligence*, 2025.

[Bareinboim *et al.*, 2015] Elias Bareinboim, Andrew Forney, and Judea Pearl. Bandits with Unobserved Confounders: A Causal Approach. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28. Curran Associates, Inc., 2015.

[Barer *et al.*, 2021] Max Barer, Guni Sharon, Roni Stern, and Ariel Felner. Suboptimal Variants of the Conflict-Based Search Algorithm for the Multi-Agent Pathfinding Problem. *International Symposium on Combinatorial Search (SoCS)*, pages 19–27, 2021.

[Boutilier, 1996] Craig Boutilier. Planning, Learning and Coordination in Multiagent Decision Processes. In *6th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 195–210, 1996.

[Cai *et al.*, 2025] Junyang Cai, Serdar Kadioğlu, and Bistra Dilkina. Balans: Multi-Armed Bandits-based Adaptive Large Neighborhood Search for Mixed-Integer Programming Problems. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2566–2574, 2025.

[Cohen and Koenig, 2016] Liron Cohen and Sven Koenig. Bounded Suboptimal Multi-Agent Path Finding Using Highways. In *International Joint Conference on Artificial Intelligence*, pages 3978–3979, 2016.

[Fioretto *et al.*, 2018] Ferdinando Fioretto, Enrico Pontelli, and William Yeoh. Distributed Constraint Optimization Problems and Applications: A Survey. *Journal of Artificial Intelligence Research*, 61:623–698, 2018.

[Foerster *et al.*, 2018] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual Multi-Agent Policy Gradients. *AAAI Conference on Artificial Intelligence*, 32(1), 2018.

[Huang *et al.*, 2022] Taoan Huang, Jiaoyang Li, Sven Koenig, and Bistra Dilkina. Anytime Multi-Agent Path Finding via Machine Learning-Guided Large Neighborhood Search. In *36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9368–9376, 2022.

[Jennings *et al.*, 1997] James Jennings, Greg Whelan, and William Evans. Cooperative Search and Rescue with a Team of Mobile Robots. In *1997 8th International Conference on Advanced Robotics. Proceedings. ICAR'97*, pages 193–200, 1997.

[Jiang *et al.*, 2025] He Jiang, Yutong Wang, Rishi Veerapaneni, et al. Deploying Ten Thousand Robots: Scalable Imitation Learning for Lifelong Multi-Agent Path Finding. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7, 2025.

[Li *et al.*, 2021a] Jiaoyang Li, Zhe Chen, Daniel Harabor, Peter J. Stuckey, and Sven Koenig. Anytime Multi-Agent Path Finding via Large Neighborhood Search. In *International Joint Conference on Artificial Intelligence*, pages 4127–4135, 2021.

[Li *et al.*, 2021b] Jiaoyang Li, Daniel Harabor, Peter J. Stuckey, et al. Pairwise Symmetry Reasoning for Multi-Agent Path Finding Search. *Artificial Intelligence*, page 103574, 2021.

[Li *et al.*, 2021c] Jiaoyang Li, Wheeler Ruml, and Sven Koenig. EECBS: A Bounded-Suboptimal Search for Multi-Agent Path Finding. In *AAAI Conference on Artificial Intelligence*, volume 35, pages 12353–12362, 2021.

[Li *et al.*, 2021d] Jiaoyang Li, Andrew Tinka, Scott Kiesel, Joseph W Durham, TK Satish Kumar, and Sven Koenig. Lifelong Multi-Agent Path Finding in Large-Scale Warehouses. In *AAAI Conference on Artificial Intelligence*, volume 35, pages 11272–11281, 2021.

[Li *et al.*, 2022] Jiaoyang Li, Zhe Chen, Daniel Harabor, Peter J. Stuckey, and Sven Koenig. MAPF-LNS2: Fast Repairing for Multi-Agent Path Finding via Large Neighborhood Search. *AAAI Conference on Artificial Intelligence*, 36(9):10256–10265, Jun. 2022.

[Li *et al.*, 2025] Mingxuan Li, Junzhe Zhang, and Elias Bareinboim. Confounding Robust Deep Reinforcement Learning: A Causal Approach. In *Advances in Neural Information Processing Systems*, volume 38, pages 138292–138325. Curran Associates, Inc., 2025.

[Lowe *et al.*, 2017] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In *Advances in Neural Information Processing Systems*, pages 6379–6390, 2017.

[Moldagalieva *et al.*, 2026] Akmaral Moldagalieva, Keisuke Okumura, Amanda Prorok, et al. db-LaCAM: Fast and Scalable Multi-Robot Kinodynamic Motion Planning with Discontinuity-Bounded Search and Lightweight MAPF. In *International Conference on Automated Planning and Scheduling (ICAPS)*, volume 36, pages 738–746, 2026.

[Nagai and Okumura, 2026] Hiroki Nagai and Keisuke Okumura. From Gridworlds to Warehouses: Adapting Lightweight One-Shot Multi-Agent Pathfinding for AGVs. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 8 2026.

[Okumura *et al.*, 2022] Keisuke Okumura, Manao Machida, et al. Priority Inheritance with Backtracking for Iterative Multi-Agent Path Finding. *Artificial Intelligence*, page 103752, 2022.

[Okumura, 2023] Keisuke Okumura. LaCAM: Search-Based Algorithm for Quick Multi-Agent Pathfinding. *AAAI Conference on Artificial Intelligence*, pages 11655–11662, 2023.

[Oliehoek and Amato, 2016] Frans A Oliehoek and Christopher Amato. A Concise Introduction to Decentralized POMDPs. *Springer*, 2016.

[Phan and Koenig, 2026] Thomy Phan and Sven Koenig. Spatially Grouped Curriculum Learning for Multi-Agent Path Finding. In *AAAI Conference on Artificial Intelligence*, pages 29642–29650, 2026.

[Phan *et al.*, 2018] Thomy Phan, Lenz Belzner, Thomas Gabor, et al. Leveraging Statistical Multi-Agent Online Planning with Emergent Value Function Approximation. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 730–738, 2018.

[Phan *et al.*, 2020] Thomy Phan, Thomas Gabor, Andreas Sedlmeier, et al. Learning and Testing Resilience in Cooperative Multi-Agent Systems. In *19th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '20, page 1055–1063, Richland, SC, 2020.

[Phan *et al.*, 2021] Thomy Phan, Fabian Ritz, Lenz Belzner, Philipp Altmann, et al. VAST: Value Function Factorization with Variable Agent Sub-Teams. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 24018–24032, 2021.

[Phan *et al.*, 2023] Thomy Phan, Fabian Ritz, Philipp Altmann, Maximilian Zorn, Jonas Nüßlein, Michael Kölle, Thomas Gabor, and Claudia Linnhoff-Popien. Attention-Based Recurrence for Multi-Agent Reinforcement Learning under Stochastic Partial Observability. In *International Conference on Machine Learning (ICML)*, 2023.

[Phan *et al.*, 2024a] Thomy Phan, Taoan Huang, Bistra Dilkina, and Sven Koenig. Adaptive Anytime Multi-Agent Path Finding Using Bandit-Based Large Neighborhood Search. *AAAI Conference on Artificial Intelligence*, 2024.

[Phan *et al.*, 2024b] Thomy Phan, Felix Sommer, Fabian Ritz, Philipp Altmann, Jonas Nüßlein, et al. Emergent Cooperation from Mutual Acknowledgment Exchange in Multi-Agent Reinforcement Learning. *Autonomous Agents and Multi-Agent Systems*, 2024.

[Phan *et al.*, 2024c] Thomy Phan, Benran Zhang, Shao-Hung Chan, and Sven Koenig. Anytime Multi-Agent Path Finding with an Adaptive Delay-Based Heuristic. In *arXiv preprint arXiv:2408.02960*, 2024.

[Phan *et al.*, 2025] Thomy Phan, Timy Phan, and Sven Koenig. Generative Curricula for Multi-Agent Path Finding via Unsupervised and Reinforcement Learning. *Journal of Artificial Intelligence Research (JAIR)*, pages 2471–2534, 2025.

[Phan *et al.*, 2026a] Thomy Phan, Shao-Hung Chan, and Sven Koenig. Truncated Counterfactual Learning for Anytime Multi-Agent Path Finding. In *AAAI Conference on Artificial Intelligence*, 2026.

[Phan *et al.*, 2026b] Thomy Phan, Joseph Driscoll, Justin Romberg, and Sven Koenig. Confidence-Based Curricula for Multi-Agent Path Finding via Reinforcement Learning. *Autonomous Agents and Multi-Agent Systems*, 40(23), 2026.

[Rashid *et al.*, 2018] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder de Witt, et al. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning (ICML)*, pages 4295–4304, 2018.

[Sartoretti *et al.*, 2019] Guillaume Sartoretti, Justin Kerr, Yunfei Shi, Glenn Wagner, TK Satish Kumar, Sven Koenig, and Howie Choset. PRIMAL: Pathfinding via Reinforcement and Imitation Multi-Agent Learning. *IEEE Robotics and Automation Letters*, 4(3):2378–2385, 2019.

[Sharon *et al.*, 2015] Guni Sharon, Roni Stern, Ariel Felner, et al. Conflict-Based Search for Optimal Multi-Agent Pathfinding. *Artificial Intelligence*, pages 40–66, 2015.

[Skrynnik *et al.*, 2024] Alexey Skrynnik, Anton Andreychuk, Maria Nesterova, Konstantin Yakovlev, and Aleksandr Panov. Learn to Follow: Decentralized Lifelong Multi-Agent Pathfinding via Planning and Learning. In *AAAI Conference on Artificial Intelligence*, volume 38, pages 17541–17549, 2024.

[Son *et al.*, 2019] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, et al. QTRAN: Learning to Factorize with Transformation for Cooperative Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning (ICML)*, pages 5887–5896, 2019.

[Stern *et al.*, 2019] Roni Stern, Nathan Sturtevant, Ariel Felner, Sven Koenig, Hang Ma, Thayne Walker, Jiaoyang Li, Dor Atzmon, Liron Cohen, TK Kumar, Roman Barták, and Eli Boyarski. Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks. In *International Symposium on Combinatorial Search*, volume 10, pages 151–158, 2019.

[Veerapaneni *et al.*, 2025] Rishi Veerapaneni, Arthur Jakobsson, Kevin Ren, Samuel Kim, et al. Work Smarter Not Harder: Simple Imitation Learning with CS-PIBT Outperforms Large-Scale Imitation Learning for MAPF. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 10229–10236, 2025.

[Zhang *et al.*, 2023] Hejia Zhang, Shao-Hung Chan, Jie Zhong, Jiaoyang Li, et al. Multi-Robot Geometric Task-and-Motion Planning for Collaborative Manipulation Tasks. *Autonomous Robots*, 47(8):1537–1558, 2023.