Proceedings of the Fourteenth International Symposium on Combinatorial Search (SoCS 2021)

From Classical to Colored Multi-Agent Path Finding

Roman Barták, Marika Ivanová, Jiří Švancara
Charles University, Faculty of Mathematics and Physics
Prague, Czech Republic
bartak@ktiml.mff.cuni.cz, ivanova@ktiml.mff.cuni.cz, jiri.svancara@mff.cuni.cz

Abstract

Multi-Agent Path Finding (MAPF) deals with the problem of finding collision-free paths for a set of agents moving in a shared environment. Colored MAPF generalizes MAPF by defining groups of agents that share a set of destination locations. In the paper, we evaluate several approaches to optimally solve colored MAPF problem. We also investigate methods for obtaining lower bounds on optimal solution based on several relaxation techniques.

Introduction

Multi-Agent Path Finding (MAPF) is the problem of finding collision-free paths for a set of agents moving in a shared environment. In a classical formulation, each agent has its own destination. However, there are applications where the agents are split into groups of agents that share their goal locations. Such generalisation of MAPF is called a *Colored MAPF*. Colored MAPF problems can be found in computer games, transportation problems (Ma and Koenig 2016), or computer art using mobile robots and drones (Barták and Mestek 2021).

An extreme version of Colored MAPF, where all agents are in one group, can be solved in polynomial time, but finding a makespan-optimal solution to Colored MAPF with at least two groups is NP-hard (Surynek 2010).

Opposite to classical MAPF, there was not much attention paid to Colored MAPF. There is only a single work (Ma and Koenig 2016) that proposed a solving technique for Colored MAPF – Conflict-Based Min-Cost-Flow (CBM).

Solution Methods

The *CBM* algorithm is based on Conflict-Based Search (CBS). On the high-level, a search is performed over conflicts among the groups, while on the low-level each group of agents is navigated by a polynomial time algorithm that is compliant with the restrictions of the search. If there is a collision between two groups, new restrictions are applied and the search continues until a collision-free solution is found.

The first two models we propose are based on reduction to SAT. We define variables $At(v, a, i)$ – agent a is in vertex v at time step i. Constraints over these variables are introduced to ensure the correct movement of the agents. This model is a direct alteration of a SAT-based model used to solve classical MAPF problem (Surynek 2014). The only difference is that each agent is allowed to arrive into any of the specified goals. We refer to this model as *SAT-basic*.

Another approach is to realize that all of the agents in a group are interchangeable – when two agents from a single group swap their positions, it is equivalent to both of them waiting in their current locations. We do not need to define the variables $At(v, a, t)$ for each agent a but rather for each group c, $At(v, c, t)$. This approach saves many variables entering the SAT solver and may prove to be more efficient. We refer to this model as *SAT-grouped*.

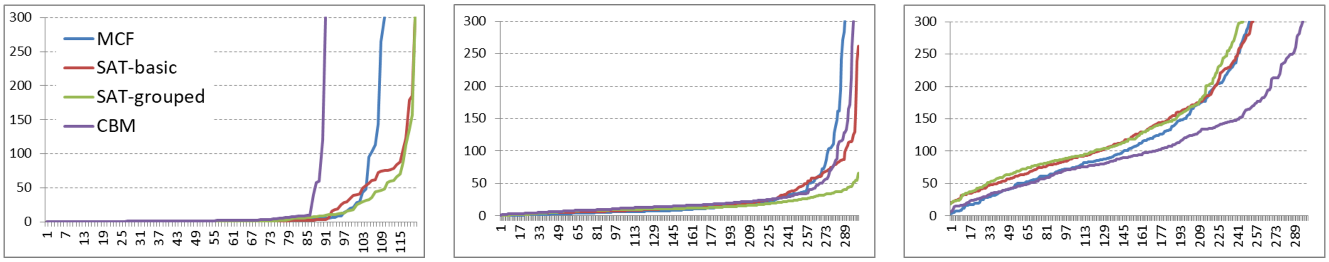
Colored MAPF has been shown to be equivalent to the multi-commodity network flow problem (MCF) in (Ma and Koenig 2016), where the authors use a directed layered graph constructed in a way that allows formulating the problem using standard MCF constraints. We utilize a simpler layered graph that requires several more constraints in addition to the standard MCF formulation because the solution time of this model is advantageous according to our preliminary experiments. Groups of agents in Colored MAPF represent different commodities flowing through the extended spatial-temporal layered graph with unit edge capacities.

Lower Bounds

All of the reduction-based models find the makespan-optimal solution by iteratively increasing the makespan until a feasible solution is found. Therefore, it is important to find a tight lower bound. We present three possible strategies:

1) Simple strategy. A lower bound for classic MAPF is obtained by computing shortest path $\ell(s, g)$ between start and goal location of each agent. In Colored MAPF, this approach is generalized by taking the minimum of the shortest paths between start location and any of the possible goal locations. This is a lower bound for a single agent. The global lower bound is the maximum of these numbers.

2) Degree strategy. A drawback of the simple strategy is that the lower bound can be based on two agents from the same group aiming for the same goal. This can be strengthened by imposing that each agent is sent to a different goal.



(a) (b) (c)

Figure 1: Figures 1a – 1c show number of instances (x-axis) solved in a given time limit (y-axis) by each of the studied solvers. In order they show the results for maps of size 8×8, 16×16 and 32×32.

For each group c, we build a complete weighted bipartite graph B_c with partitions corresponding to start and goal locations, in which edge costs are defined by $\ell(s, g)$. We then iteratively remove edges in the order of decreasing costs as long as none of the nodes in B_c is isolated. This ensures that each vertex is reachable since all of them have to be used in the solution. The cost of the last removed edge is the desired lower bound.

3) Matching strategy. The lower bound obtained by Degree strategy can be improved by checking whether B_c still contains a complete matching. The reasoning behind the correctness is similar to the Degree strategy – all of the agents have to reach some goal, therefore each start has to be matched to some goal. Removing the longest edges (which means ignoring the longest path) provides a lower bound for makespan as opposed to finding a matching with the smallest cost which would provide a lower bound for sum of costs (another cost function often used in classical MAPF).

Observation 1. *Let* Γ *be an instance of Colored MAPF and* $\lambda_X(\Gamma)$ *be the lower bound obtained by method* X*. Then,*

$$\lambda_{sim}(\Gamma) \leq \lambda_{deg}(\Gamma) \leq \lambda_{match}(\Gamma).$$

Empirical Evaluation

The experiments are performed on grid maps with no obstacles (*empty*) and with 20% of random vertices impassable (*random*) of sizes 8×8, 16×16 and 32×32, with the number of groups $k = 5$ and $k = 10$. In every instance, the groups' sizes are the same $|A_1| = \cdots = |A_k|$, and the total number of agents increases from k to 100 (40 for map sizes 8×8) with the increment of k (i.e. $|A| = k, 2k, \ldots, 100$). A time limit of 5 minutes is imposed on runtime of each instance.

First, we compare the three strategies for obtaining lower bounds. Simple, Degree and Matching strategies yield optimal makespan estimate in 200, 362, and 654 out of 720 cases respectively. Fig. 2 shows their effect on the efficiency of SAT models (negligible runtime to compute lower bounds is included). The clear winner is the Matching strategy and will be used in the rest of the experiments.

The growth of runtime of each algorithm is depicted in Fig. 1a-1c. Both MCF and SAT models fall behind CBM on the largest maps (Fig. 1c). In the medium sized maps

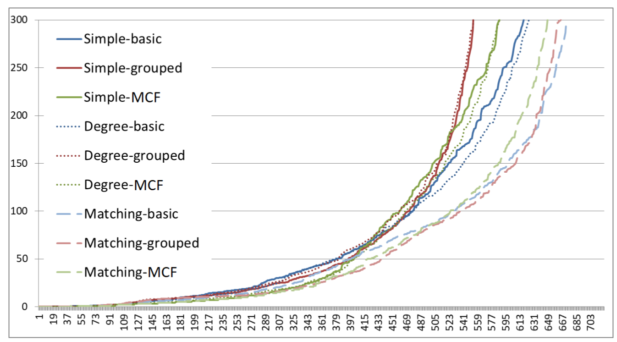


Figure 2: The impact of different lower bounding methods on runtime for each proposed model. The graph shows number of instances (x-axis) solved in a given time limit (y-axis).

(Fig. 1b), the difference is much less obvious with SAT-grouped being the best. In the smallest maps, CBM is outperformed by all other methods (Fig. 1a). These results comply with the observation made for the classical MAPF problem that the CBS-based algorithm performs well on sparse instances, and reduction-based algorithms perform well on smaller, more dense instances (Švancara and Barták 2019). Surprisingly, SAT-grouped is not always better than SAT-basic, especially on the largest maps.

In some cases, we observe that adding more agents to the instance can make it easier to solve. This is counterintuitive since in classical MAPF adding more agents makes the problem harder. In Colored MAPF, additional agents are assigned to some group and with them, new goal locations are added. These goal locations may be used by other agents as well which may yield a shorter and easier to compute plan.

The detailed description of proposed models and wider empirical evaluation is available at the full version of the paper (Barták, Ivanová, and Švancara 2021).

Acknowledgements

This research is supported by the the Czech-Israeli Cooperative Scientific Research Project LTAIZ19014 and by the project 19-02183S of the Czech Science Foundation.

References

Barták, R.; Ivanová, M.; and Švancara, J. 2021. Colored Multi-Agent Path Finding: Solving Approaches. *The International FLAIRS Conference Proceedings* 34. doi:10.32473/flairs.v34i1.128535. URL https://journals.flvc.org/FLAIRS/article/view/128535.

Barták, R.; and Mestek, J. 2021. OzoMorph: Demonstrating Colored Multi-Agent Path Finding on Real Robots. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*. AAAI Press.

Ma, H.; and Koenig, S. 2016. Optimal Target Assignment and Path Finding for Teams of Agents. In *Proceedings of AAMAS 2016*, 1144–1152.

Surynek, P. 2010. An Optimization Variant of Multi-Robot Path Planning Is Intractable. In *Proceedings of AAAI 2010*.

Surynek, P. 2014. Compact Representations of Cooperative Path-Finding as SAT Based on Matchings in Bipartite Graphs. In *Proceedings of ICTAI 2014*, 875–882.

Švancara, J.; and Barták, R. 2019. Combining Strengths of Optimal Multi-Agent Path Finding Algorithms. In *Proceedings of ICAART 2019*, 226–231.