

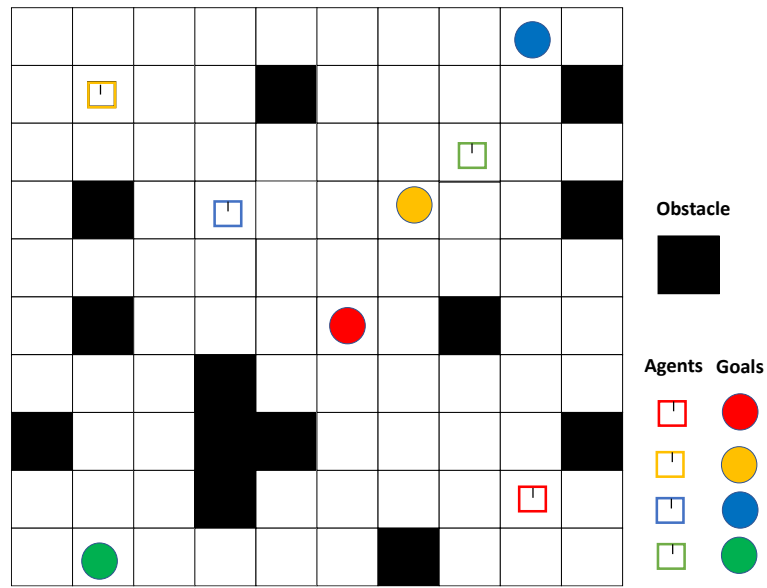
# Multi-Agent Pathfinding for Deadlock Handling on Rotational Movements

Frodo Kin Sun Chan, Yan Nei Law and Edmond Shiao Bun Lai\*

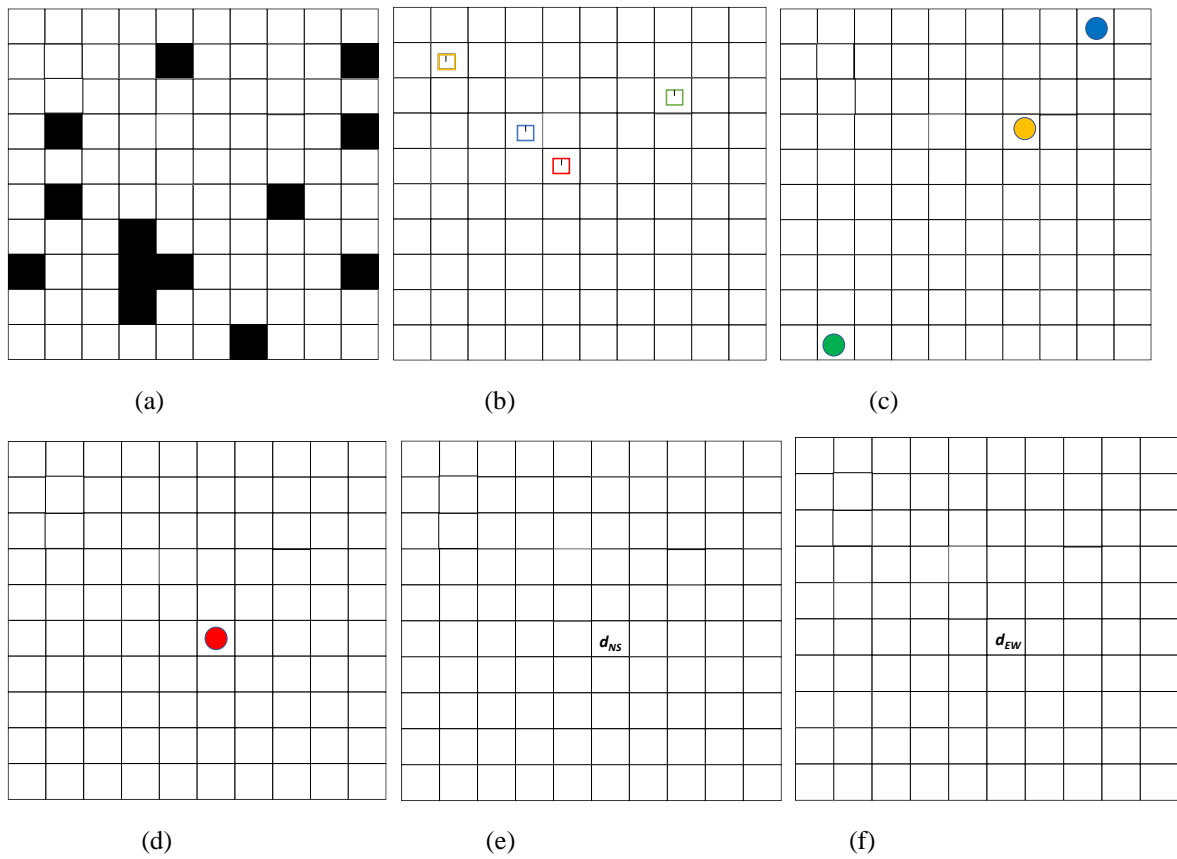
---

\*All authors are with the Hong Kong Industrial Artificial Intelligence and Robotics Centre, Hong Kong.  
{frodochan, ivylaw}@hkflair.org, edmondlai@hkpc.org

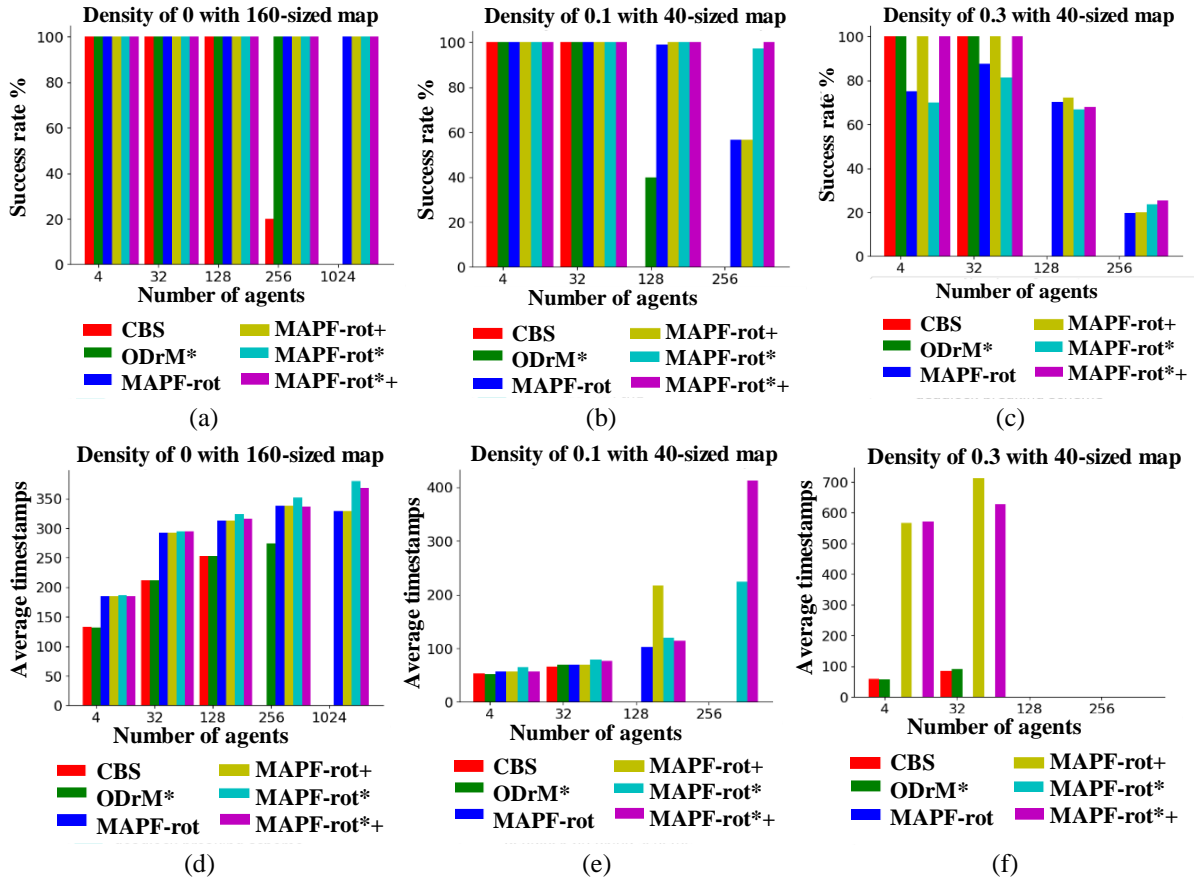
## Supplementary information



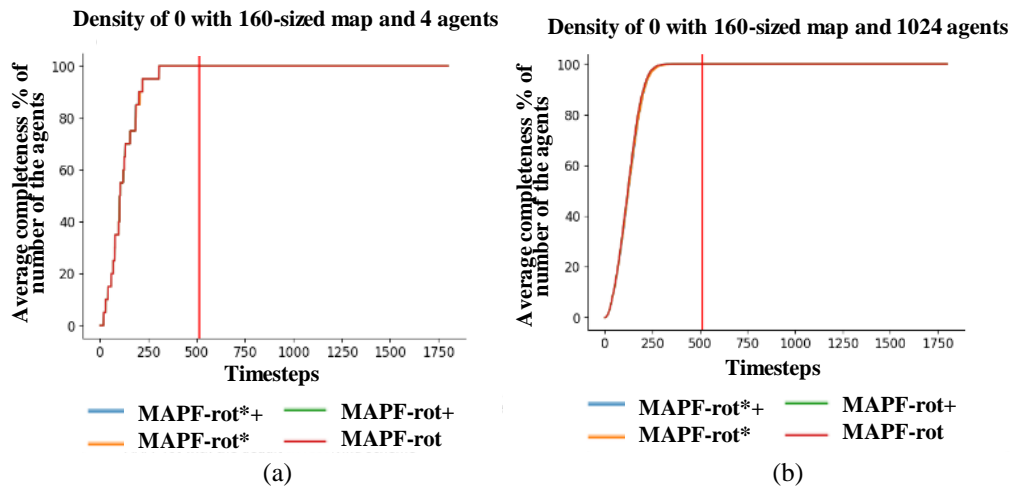
**Fig. S1.** A layout of the simulation environment.



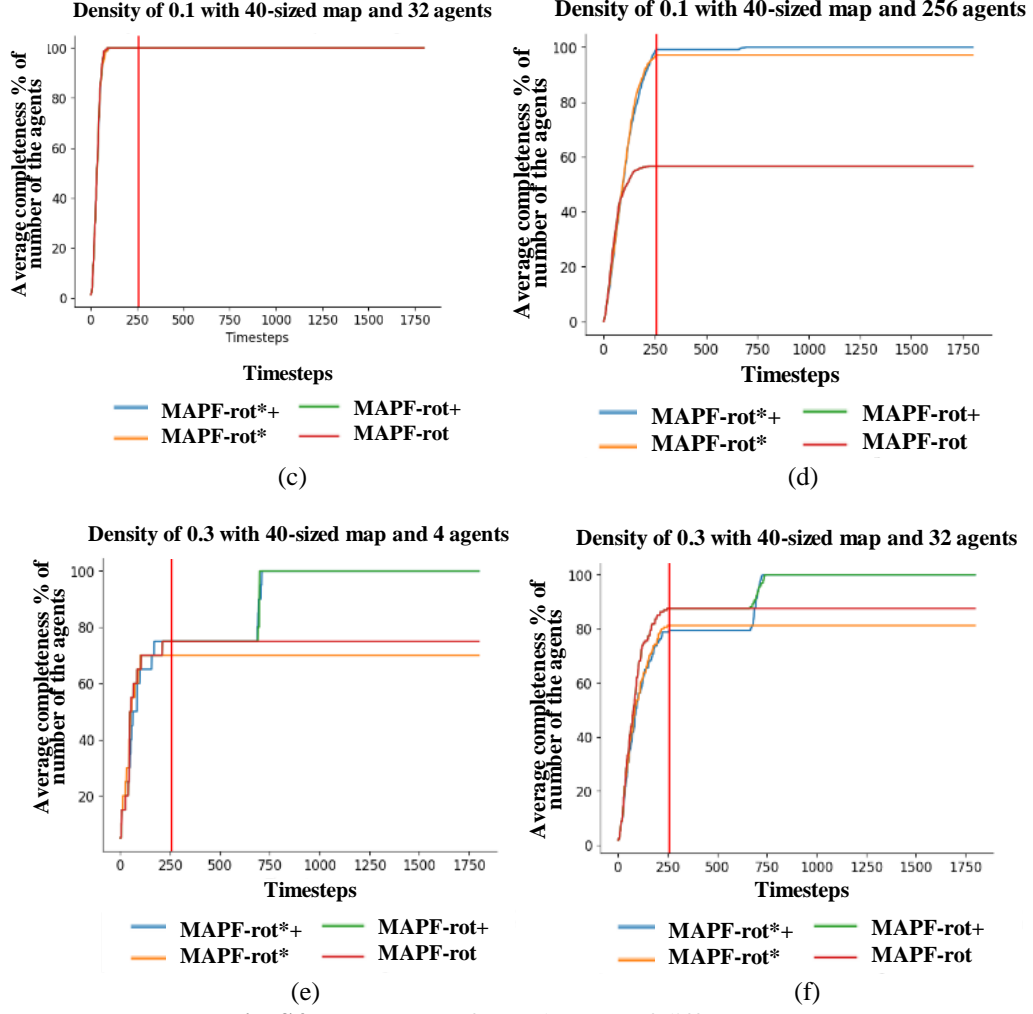
**Fig. S2.** Observation Space: (a) obstacles (b) agents' position (c) neighbors' goals (d) agent's goal (e) vertical map (f) horizontal map



**Fig. S3.** Success rate % and average timestamps<sup>2</sup> on (a, d) density of 0 with 160-sized map, (b, e) density of 0.1 with 40-sized map, (c, f) density of 0.3 with 40-sized map



<sup>2</sup> The results are shown as empty when more than half of the trials are not completed. MAPF-rot+ means MAPF-rot with deadlock resolving scheme, MAPF-rot\* means MAPF-rot with deadlock breaking scheme, MAPF-rot\*+ means MAPF-rot with deadlock breaking and resolving scheme.

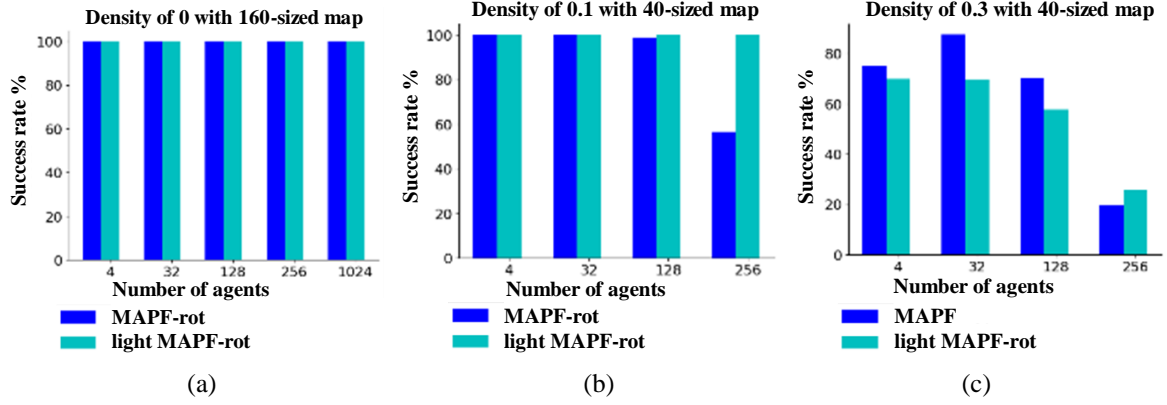


**Fig. S4.** Percentages of completeness of different cases

In **Fig. S4(a-b)**, three algorithms perform quite similar because the deadlock seldom happens for situation with large map and no obstacle. In **Fig. S4(c-f)**, three algorithms performs differently when the number of agents increases with high density of obstacle. In **Fig. S4(d)**, MAPF-rot only finishes 56.64%, but MAPF-rot+ and MAPF-rot\*+ can have 97.27% and 100.00%. In **Fig. S4(f)**, MAPF-rot only finishes 28.13%, but MAPF-rot+ and MAPF-rot\*+ can have 82.50% and 91.25%.

Supp. C

The complete version of Fig. 5



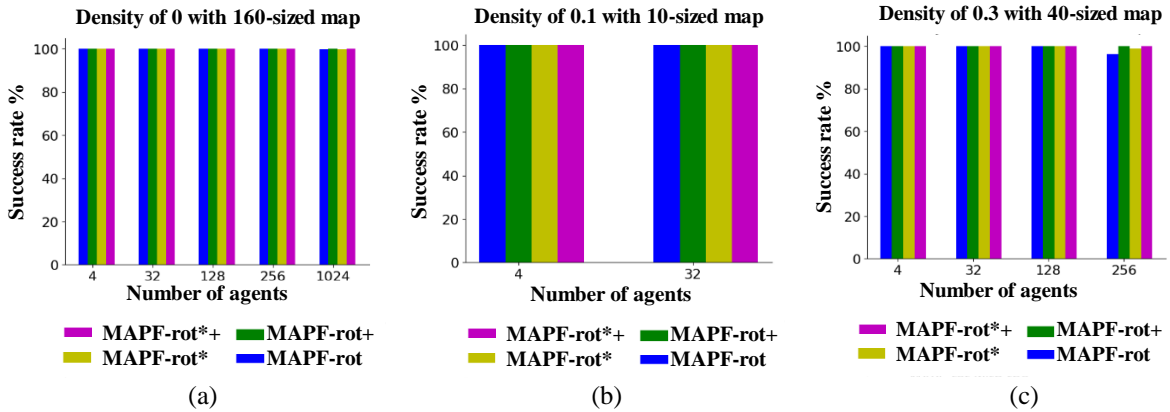
**Fig. S5.** The success rate of different settings including (a) density of 0 with 160-sized map, (b) density of 0.1 with 40-sized map, (c) density of 0.3 with 40-sized map.

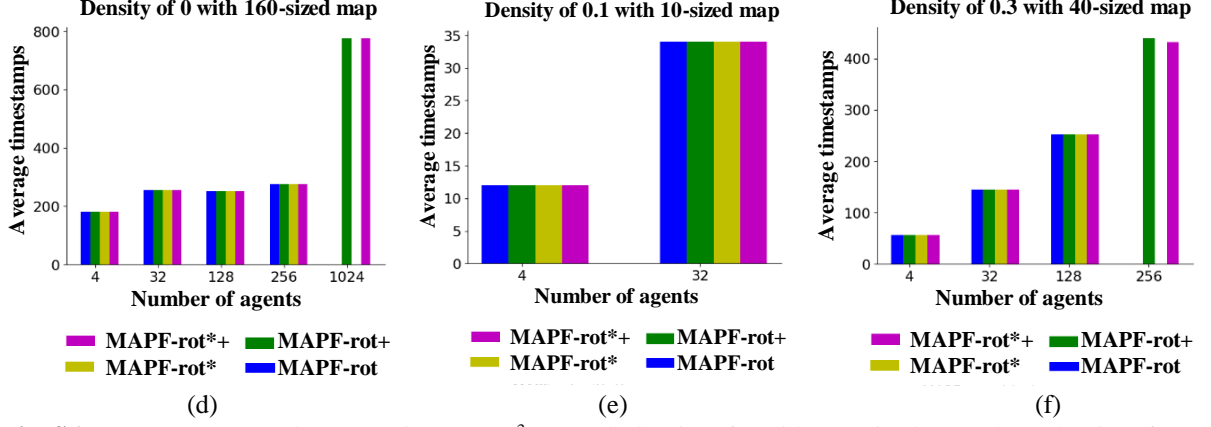
Supp. D

The feasibility of proposed schemes

To demonstrate the feasibility of the proposed schemes, the environment and PRIMAL model with the movement on 4-connected path and 8-connected path, are evaluated with the proposed schemes. In the movement on 8-connected path, the actions of an agent not only include idle and moving in cardinal directions, which are the movement on 4-connected path, but they also include moving to the ordinal directions, i.e. diagonal move. Thus, the diagonal move has to be enabled for 8-connected path in the environment settings.

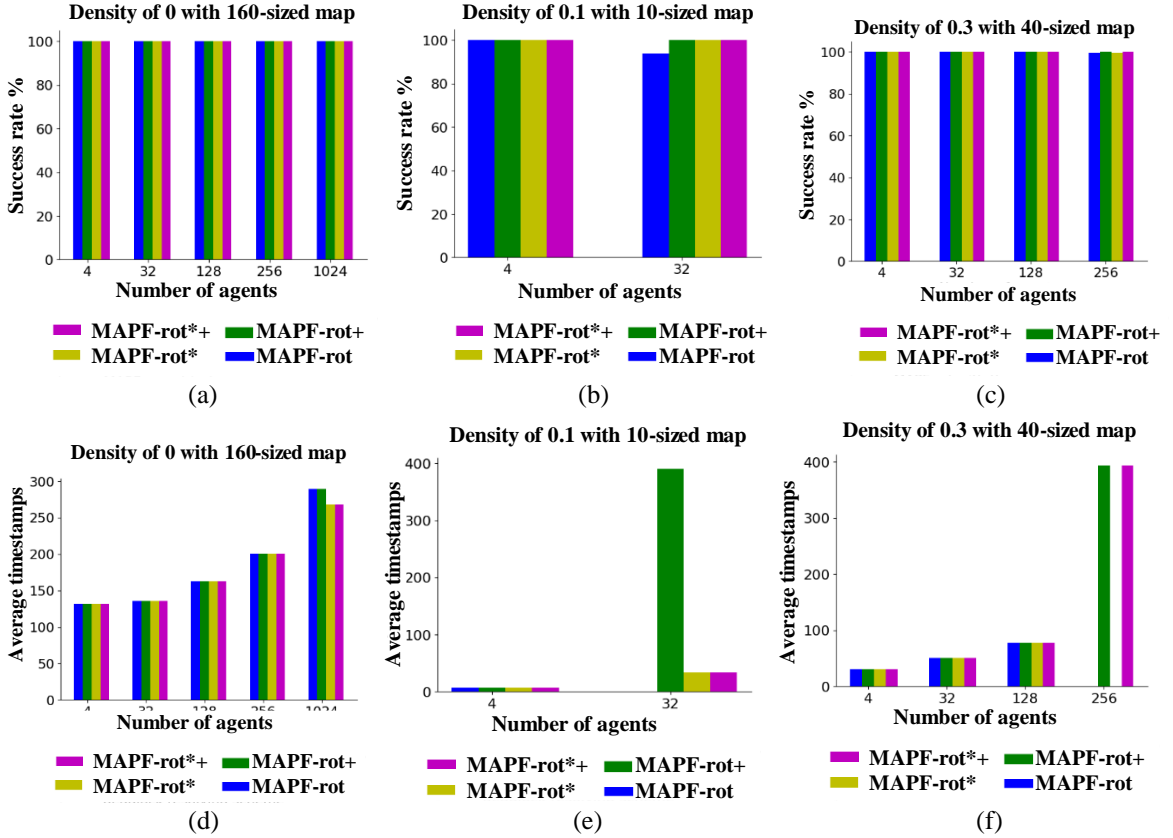
In the experiment,  $N$  is set as 20 empirically in the deadlock breaking scheme because its large value is better for the case of few occurrences of deadlock. Besides, the rollback sequence of deadlock resolving scheme is also implemented with the actions for both connected paths (e.g. moving left becomes moving right and moving top-left becomes moving bottom-right). The experiment is carried out with one trial in the dataset of Section 4.1 and the evaluation method in Section 4.2 (including the success rate and average timestamps) is also used for measurement in this section. The experimental result is shown in **Fig. S6**. Most of the result can reach 100% except for the results with 1024 agents. In these cases including the 4-connected path with 1024 agent of **Fig. S6** (a), the 8-connected path with 1024 agent of **Fig. S7**(b), and both connected paths with 1024 agent of **Fig. S6**(c) and **Fig. S7**(c), the deadlock resolving scheme is useful to improve the success rate. Although their average timestamps can exceed the timeout in **Fig. S6**(d-f) and **Fig. S7**(d-f), this scheme is beneficial for the agents to resolve the deadlock and complete the goals. Besides, the deadlock breaking scheme is also advantageous for enhancing the success rate in the cases of 8-connected path with 32 agents of **Fig. S7**(b) and 4-connected path with 256 agents of **Fig. S6**(c). Furthermore, it also reduces the average timestamps in the cases of 8-connected path with 1024 agents of **Fig. S7**(d), 8-connected path with 32 agents of **Fig. S7**(e), and 4-connected path with 256 agents of **Fig. S6**(f). Thus, both schemes are useful to apply in different configurations for boosting the results on MAPF.





**Fig. S6.** Success rate % and average timestamps<sup>3</sup> on (a, d) density of 0 with 160-sized map, (b, e) density of 0.1 with 10-sized map, (c, f) density of 0.3 with 40-sized map for 4-connected path

Besides, the inference times of the four methods are based on the same model inferencing from MAPF-rot. Thus, they have the same inference time {0.01s, 0.03s, 0.13s, 0.25s, 1.02s} for different number of agents {4, 32, 128, 256, 1024}.



**Fig. S7.** Success rate % and average timestamps<sup>4</sup> on (a, d) density of 0 with 160-sized map, (b, e) density of 0.1 with 10-sized map, (c, f) density of 0.3 with 40-sized map for 8-connected path

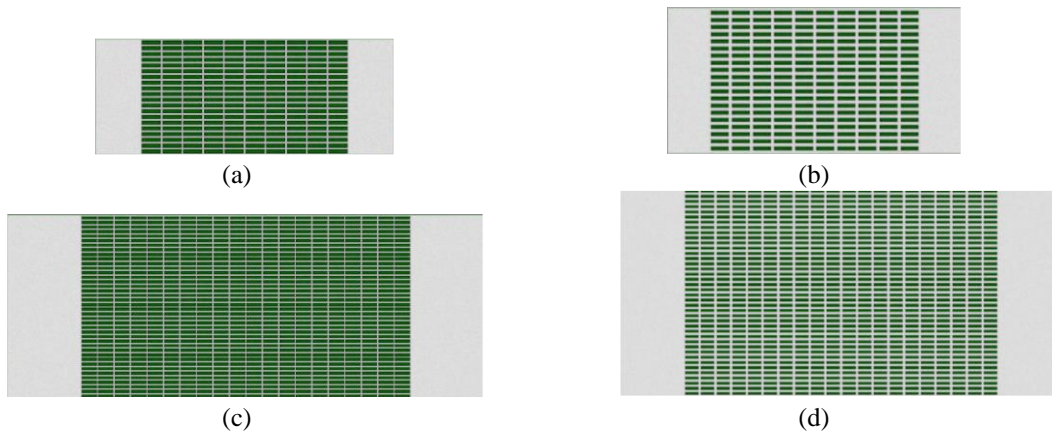
<sup>3</sup> The results are shown as empty when more than half of the trials are not completed. MAPF-rot+ means MAPF-rot with deadlock resolving scheme, MAPF-rot\* means MAPF-rot with deadlock breaking scheme, MAPF-rot\*+ means MAPF-rot with deadlock breaking and resolving scheme.

<sup>4</sup> The results are shown as empty when more than half of the trials are not completed. MAPF-rot+ means MAPF-rot with deadlock resolving scheme, MAPF-rot\* means MAPF-rot with deadlock breaking scheme, MAPF-rot\*+ means MAPF-rot with deadlock breaking and resolving scheme.

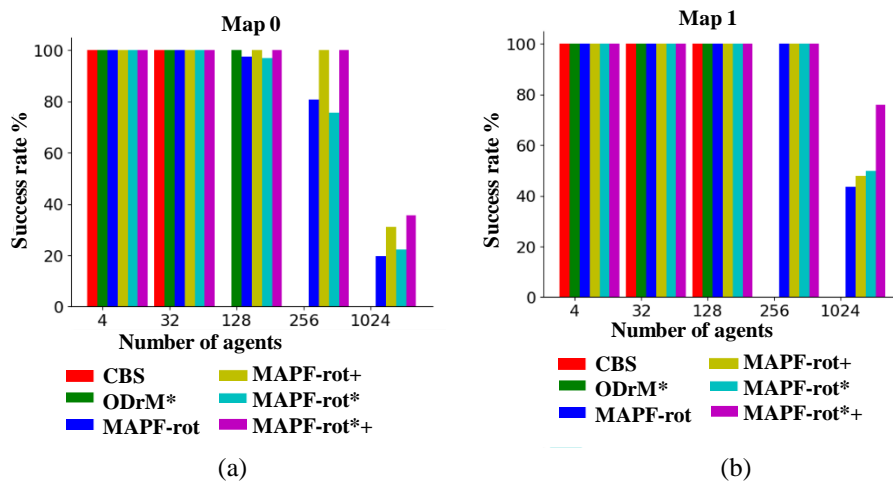
## Supp. E The proposed schemes under the warehouse settings

In order to evaluate the performance of the proposed algorithm in the warehouses, four maps shown in **Fig. S8** are used for experiments and they are obtained from MAPF benchmark website (Ref: <https://movingai.com/benchmarks/mapf/index.html>). The four maps simulate the real scenario of warehouses and they are challenging because warehouse aisles are very narrow for AGV movement. Besides, the experiment is performed on a laptop with Intel i7-10850H CPU, 32GB RAM, and GTX 1650 Ti with Max-Q design.

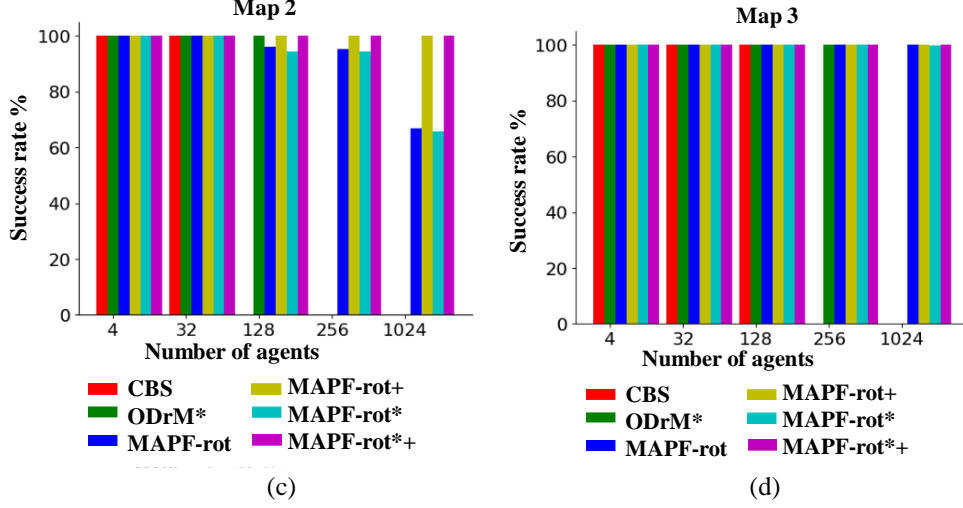
The testing dataset is based on different maps {0, 1, 2, 3} and different numbers of agents {4, 32, 128, 256, 1024}. The testing settings and methods are the same as the settings and methods in Session 4.2. For each setting, 1 trial is performed. The success rates of different methods are shown in **Fig. S9**. The complete results and some demo videos are posted online<sup>5</sup>. In **Fig. S9** and **Fig. S10**, the success rates and timestamps of different methods are quite similar when few agents (4 and 32) are tested. When the number of agents increases, MAPF-rot with deadlock breaking and resolving scheme shows better performance than others. Besides, **Fig. S9** shows that MAPF-rot has an improvement of the success rate with the existence of deadlock resolving scheme. Thus, this experiment further confirms the feasibility of the proposed algorithm in the warehouse settings.



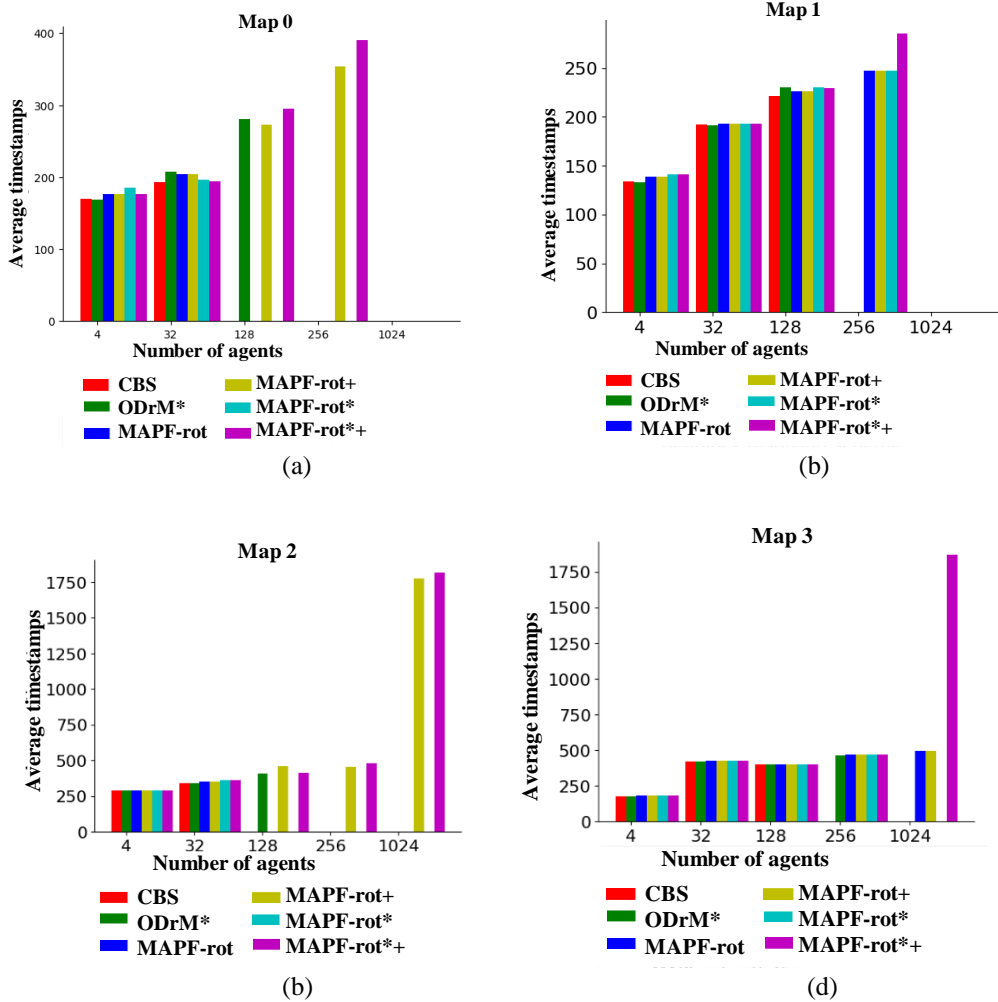
**Fig. S8.** The layouts of (a) Map 0 with size 161x63, (b) Map 1 with size 170x84, (c) Map 2 with size 321x123, (d) Map 3 with size 340x164



<sup>5</sup> GitHub project page : <https://github.com/F0048/MAPF>



**Fig. S9.** Success rate of different methods. Success rate % on (a) Map 0 with size 161x63, (b) Map 1 with size 170x84, (c) Map 2 with size 321x123, (d) Map 3 with size 340x164



**Fig. S10.** Timestamps of different methods. Average timestamps<sup>6</sup> on (a) Map 0 with size 161x63, (b) Map 1 with size 170x84, (c) Map 2 with size 321x123, (d) Map 3 with size 340x164

<sup>6</sup> The results are shown as empty when more than half of the trials are not completed. MAPF-rot+ means MAPF-rot with deadlock resolving scheme, MAPF-rot\* means MAPF-rot with deadlock breaking scheme, MAPF-rot\*+ means MAPF-rot with deadlock breaking and resolving scheme.