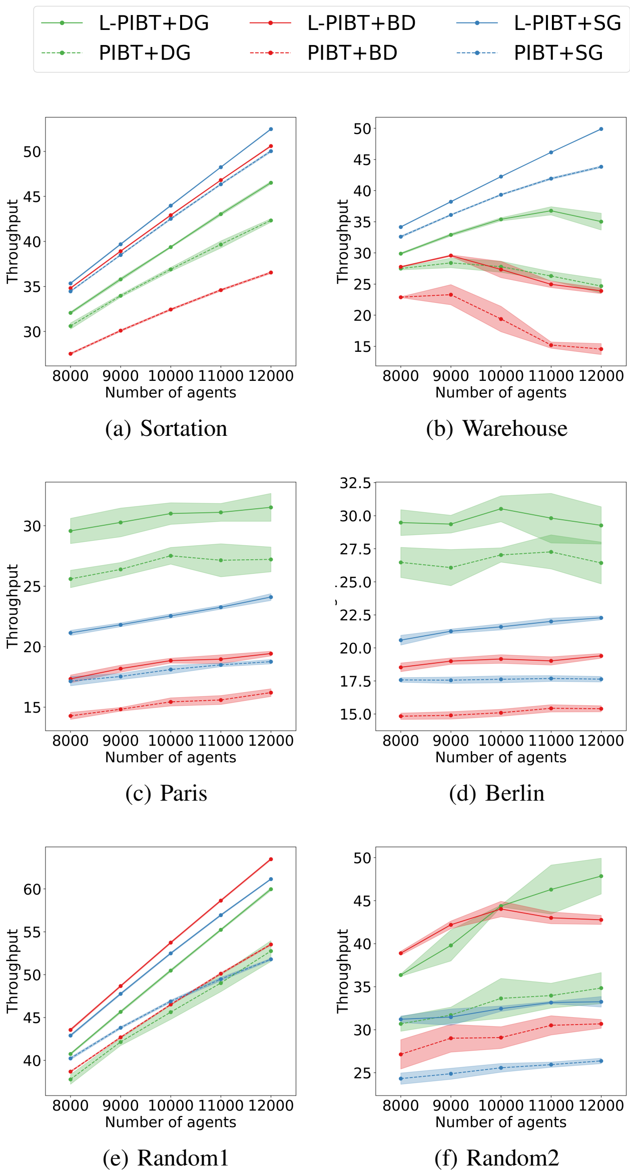


(a) Sortation (b) Warehouse

(c) Paris (d) Berlin

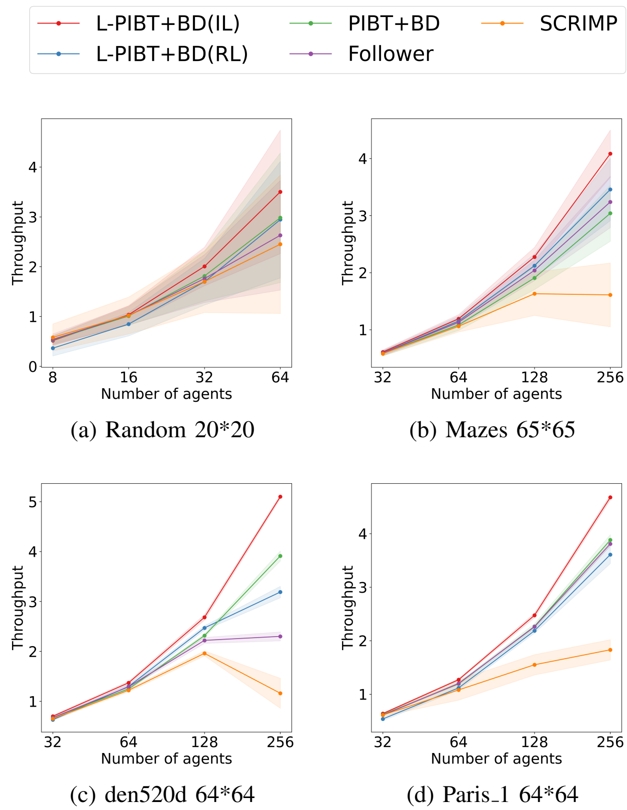
(e) Random1 (f) Random2

Fig. 6: Evaluation with different numbers of agents.

APPENDIX

A. Evaluation with Different Numbers of Agents

In this section, we compare Learnable PIBT and PIBT with different global guidance and different numbers of agents. The conclusions are similar to the ones in section V. With the same global guidance, Learnable PIBT consistently outperforms PIBT, proving the effect of learning. Also, different global guidance excels in different scenarios. An interesting observation is that in map Random2, Backward Dijstraj (BD) performs better with $< 10,000$ agents, and Dynamic Guidance (DG) performs better with $> 10,000$ agents. The reason may be that with more agents, there is potentially more congestion, and DG addresses congestion better than BD.



(a) Random 20*20 (b) Mazes 65*65

(c) den520d 64*64 (d) Paris_1 64*64

Fig. 7: Evaluation on Learn-to-Follow Benchmark.

B. Evaluation on Learn-to-Follow Benchmark

This section compares different decentralized methods with the Learn-to-Follow Benchmark [12] to validate the superiority of our SILLM (Learnable PIBT). Specifically, we compare Learnable PIBT trained with imitation learning and with reinforcement learning, Follower [12], SCRIMP [10], and PIBT [5]. For a simple comparison, we only use Backward Dijstraj as global guidance. All the training settings are the same as the ones in the Follower paper [12]. Notably, Learnable PIBT and Follower are trained on 40 Mazes maps and tested on 10 test maps and other maps. Our Learnable PIBT trained with imitation learning performs the best consistently across 4 different kinds of maps. Notably, Follower actually only outperforms PIBT in Mazes maps but may fail to outperform PIBT in other maps, which means the generalization ability of Follower still needs improvement. In contrast, our Learnable PIBT is much more generalizable.

C. Real-World Mini Example

Since during the planning process, our algorithm assumes the position of all agents to be perfectly known at all times, we use ground truth positions for our virtual robots and use external localization (here, the Optitrack Motion Capture System) to obtain accurate position information for our real robots. However, the planned path may not be executed accurately due to disturbances and control inaccuracies. To eliminate these errors, we implement an Action Dependency Graph (ADG) [32], [33].

The video demo is available in the supplementary material. From our experiment with 10 real agents, we observe that agents can reach their goals quickly without collisions, and errors are eliminated by the ADG, demonstrating the potential of using our method in the real world. In our experiment with 100 virtual agents, we compare PIBT with Learnable PIBT. We can observe that Learnable PIBT outperforms PIBT with 50% more throughput.

D. Computation Resources

Our models are trained on servers with 72 vCPU AMD EPYC 9754 128-Core Processor, 4 RTX 4090D (24GB), and 240 GB memory. Training on each map takes ≤ 12 hours.