



Priority inheritance with backtracking for iterative multi-agent path finding

Keisuke Okumura, Manao Machida, Xavier Défago, Yasumasa Tamura,
Priority inheritance with backtracking for iterative multi-agent path finding,
Artificial Intelligence,
Volume 310,
2022

汇报人姓名：殷明睿

汇报日期：2024.09.24

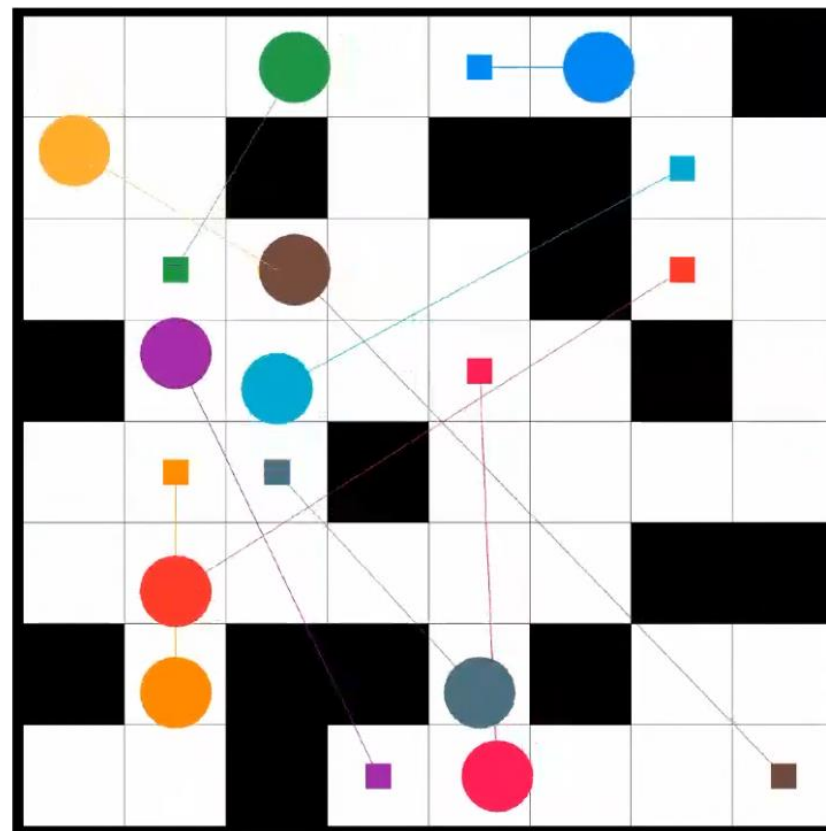
MAPF: Multi-Agent Path Finding



MAPF: Multi-Agent Path Finding

- Solution: 无碰撞的解决方案
- Cost: 总行程时间、距离、完成时间等

NP-hard Problem



MAPF: Multi-Agent Path Finding

- 潜在的常见问题:
- 多个代理的无碰撞寻路:
 - 快速、实时
 - 可扩展
 - 更少的冗余动作（最优性）

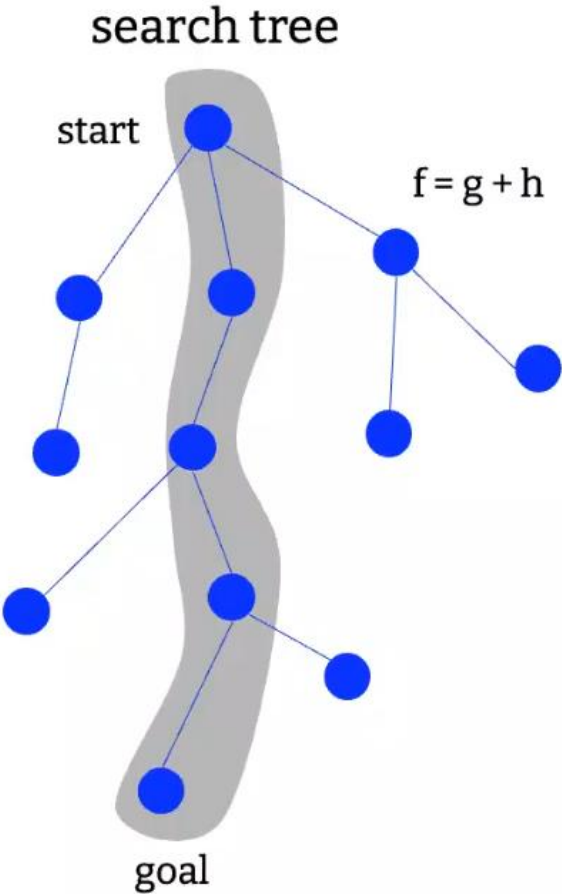


Centralization
理论保证（完整性最优性）

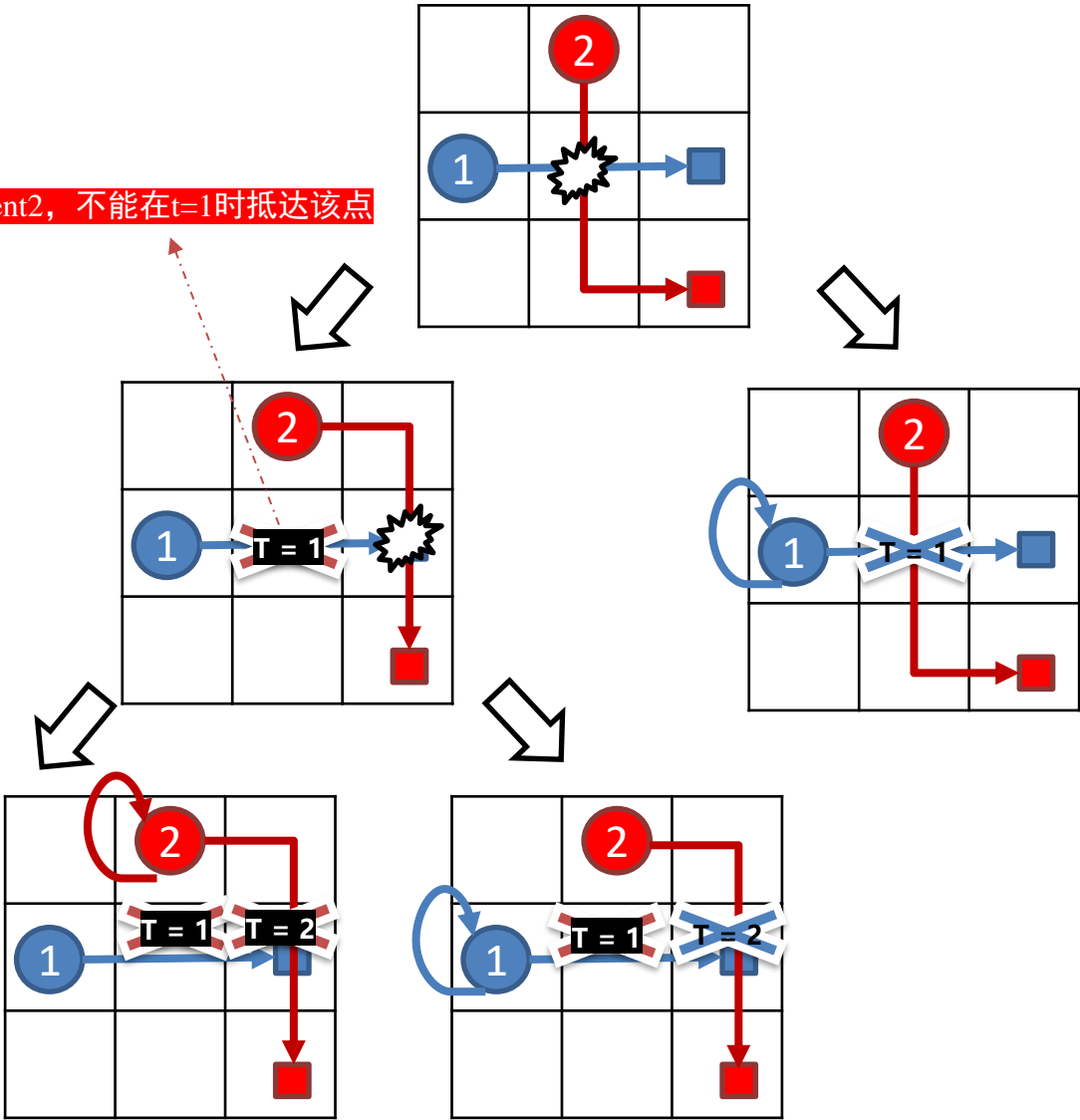
Decentralization
快速、可扩展

-> 具有良好理论保证的可扩展集中式寻路算法

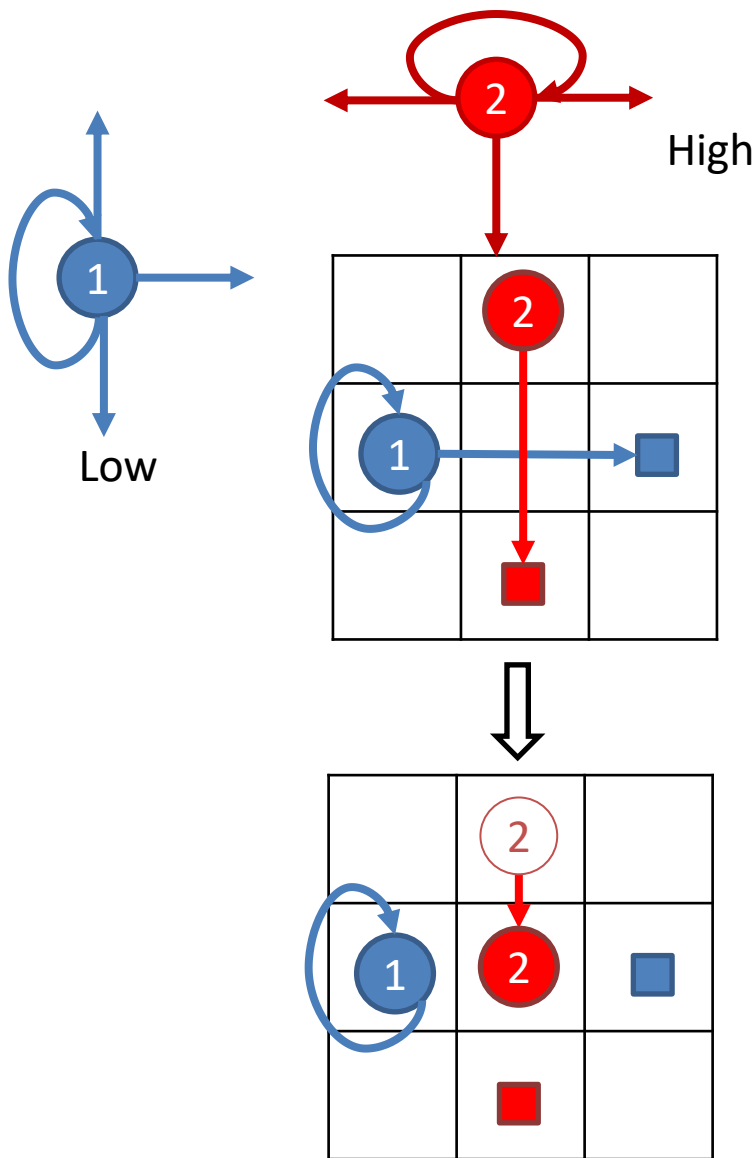
A* 和 CBS



对于agent2, 不能在t=1时抵达该点

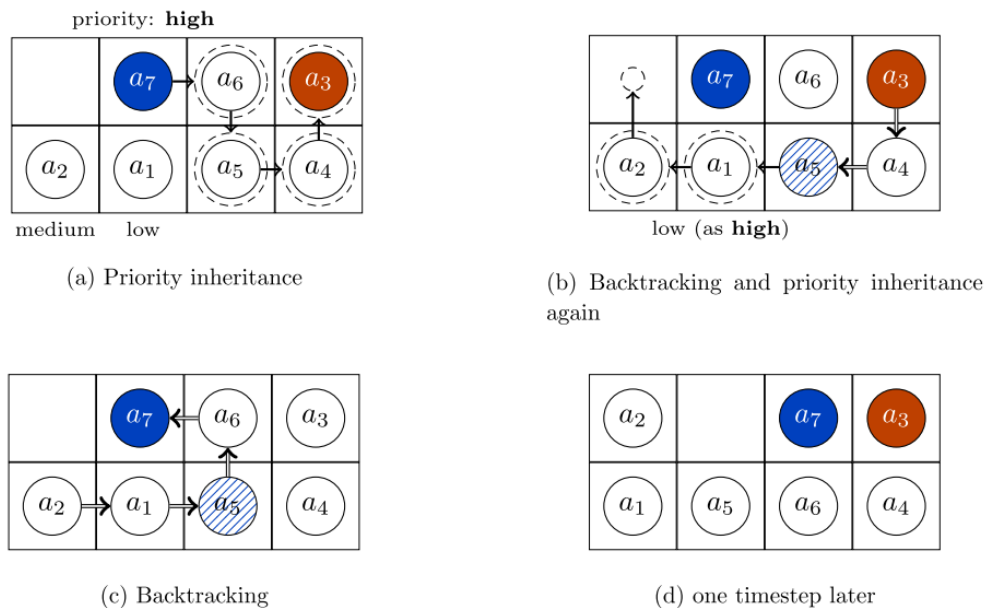
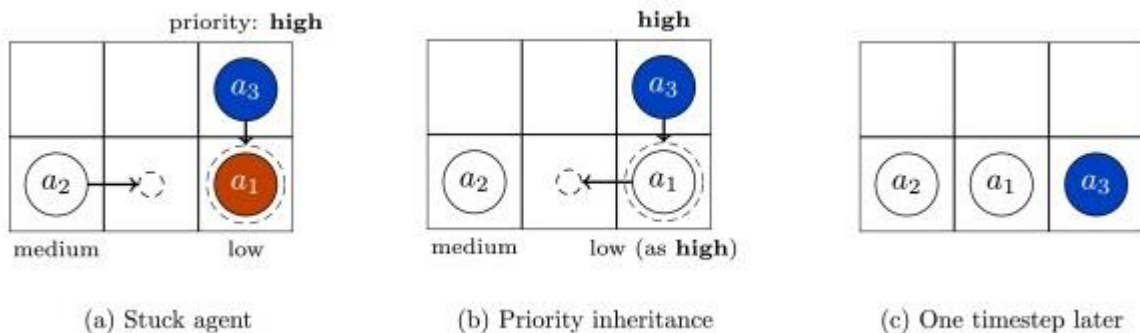


PIBT: 带回溯的优先级继承:



- 初始化:
 - 计算最小步数 T_{\min}
 - 设定初始优先级
- 运行PIBT调动agent直到 T_{\min}
- PIBT:
 - 定义候选节点C作为agent的下一个位置
 - 根据每个节点到目标的位置对C进行排序, 过滤掉优先级更高的节点请求的位置
 - 对于C中的每个节点v, 检查智能体是否能在下一个时间步移动到v, 并且移动/不动该智能体

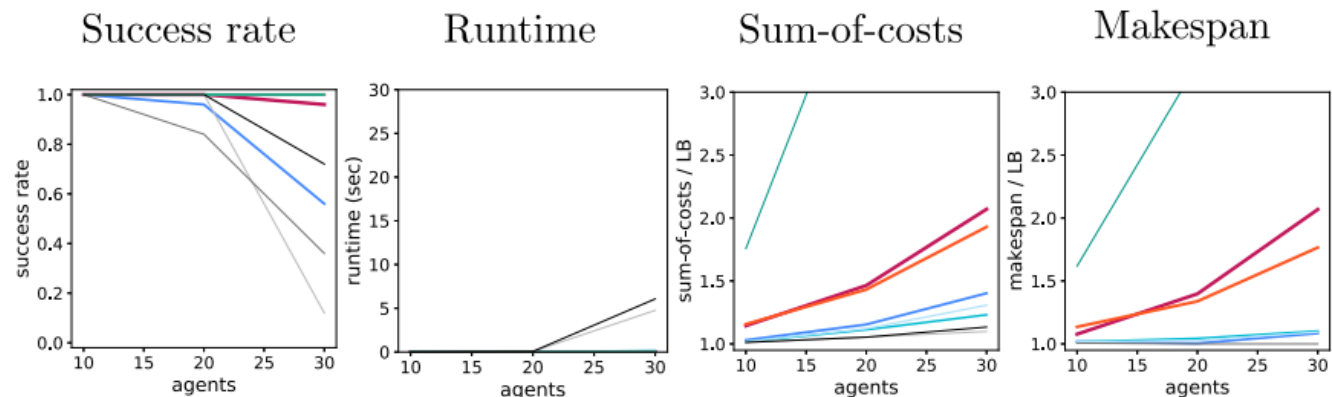
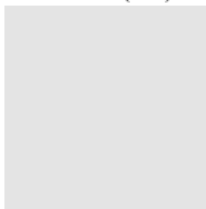
当低优先级代理 X 阻碍高优先级代理 Y 的移动时，代理 X 暂时继承代理 Y 的较高优先级。



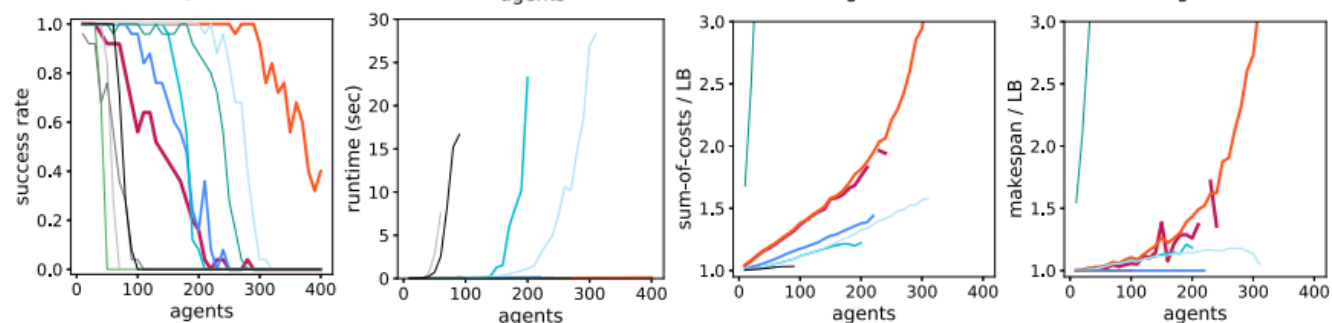
双箭头表示回溯的流程，由于 a_3 卡住了 (a)，回溯无效返回到 a_4 ，接着返回到 a_5 ， a_5 对 a_1 执行其他优先级继承 (b)， a_1 、 a_5 、 a_6 、 a_7 等待回溯结果 (c)，然后开始移动 (d)

Performance of PIBT

empty-8-8
8 × 8 (64)



random-32-32-20
32 × 32 (819)



empty-48-48
48 × 48 (2,304)

