

控制与决策

Control and Decision

基于时空冲突和非完整约束下的多AMR调度算法研究

申子祥, 孙训红, 从永正, 余兰林

引用本文:

申子祥, 孙训红, 从永正, 等. 基于时空冲突和非完整约束下的多AMR调度算法研究[J]. *控制与决策*, 2025, 40(3): 981–986.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2024.0380>

您可能感兴趣的其他文章

Articles you may be interested in

[参数不确定离散时间系统的有限时间输出反馈预见控制器设计](#)

Design of finite-time output feedback preview controller for discrete-time systems with parameter uncertainty
控制与决策. 2021, 36(9): 2074–2084 <https://doi.org/10.13195/j.kzyjc.2019.1584>

[基于移动传感器/执行器网络的时滞分布参数系统镇定控制](#)

Stabilization control for a class of distributed parameter systems with time-delay based on mobile sensor and actuator networks
控制与决策. 2021, 36(8): 1955–1962 <https://doi.org/10.13195/j.kzyjc.2019.1309>

[多航天器系统分布式固定时间输出反馈姿态协同跟踪控制](#)

Distributed fixed-time output feedback attitude coordination tracking control for multiple rigid spacecraft
控制与决策. 2021, 36(5): 1049–1058 <https://doi.org/10.13195/j.kzyjc.2019.0968>

[天临空协同对地观测任务规划模型与并行竞争模因算法](#)

Planning model and parallel competing memetic algorithm for space-near space-air based cooperative earth observation missions
控制与决策. 2021, 36(3): 523–533 <https://doi.org/10.13195/j.kzyjc.2020.0732>

[有向切换拓扑条件下多航天器分组姿态协同控制](#)

Group attitude coordinated control of multi-spacecraft with directed switching topologies
控制与决策. 2021, 36(10): 2389–2398 <https://doi.org/10.13195/j.kzyjc.2020.0311>

基于时空冲突和非完整约束下的多 AMR 调度算法研究

申子祥, 孙训红, 从永正, 余兰林[†]

(合肥工业大学 电气与自动化工程学院, 合肥 230009)

摘要: 针对多自主移动机器人 (AMR) 调度系统中的决策规划问题, 提出一种分层策略的基于时空冲突和非完整约束下的多 AMR 调度算法. 首先, 提出加权时空混合 A* 算法作为算法的底层, 进行单 AMR 路径规划, 以确保规划的路径满足阿克曼类型 AMR 的运动学模型; 然后, 引入结合分离轴定理的冲突检测树作为算法的上层, 对底层算法规划出的路径进行精确地时空冲突检测, 检测出的时空冲突被用作底层算法的时空约束; 最后, 再次调用底层算法, 考虑时空约束进行路径规划, 以实现高效的 AMR 冲突消解. 仿真结果表明, 所提出算法能够为多 AMR 调度系统高效地规划出满足非完整约束的无冲突路径.

关键词: 多 AMR 系统; 非完整约束; 时空冲突检测; 加权时空混合 A* 算法; 冲突消解; 路径规划

中图分类号: TP18 文献标志码: A

DOI: 10.13195/j.kzyjc.2024.0380

引用格式: 申子祥, 孙训红, 从永正, 等. 基于时空冲突和非完整约束下的多 AMR 调度算法研究 [J]. 控制与决策, 2025, 40(3): 981-986.

Research on multi-AMR scheduling algorithm based on spatiotemporal conflicts and nonholonomic constraints

SHEN Zi-xiang, SUN Xun-hong, CONG Yong-zheng, YU Lan-lin[†]

(School of Electrical Engineering and Automation, Hefei University of Technology, Hefei 230009, China)

Abstract: In order to address the decision planning problem in multi-AMR scheduling systems, this paper proposes a hierarchical strategy-based multi-AMR scheduling algorithm under spatiotemporal conflicts and nonholonomic constraints. Firstly, this paper proposes the weighted spatiotemporal hybrid A* algorithm as the lower-level of this algorithm to conduct single AMR path planning, ensuring that the planned paths conform to the kinematic model of the Ackermann-type AMR. Then, this paper proposes a conflict detection tree combined with the separating axis theorem as the upper-level algorithm to conduct precise spatiotemporal conflict detection on the paths planned by the lower-level of this algorithm. The detected spatiotemporal conflicts are utilized as the spatiotemporal constraints for the lower-level of this algorithm. Finally, the lower-level of this algorithm is utilized once again, taking into account the spatiotemporal constraints for path planning, to achieve efficient AMR conflict resolution. Simulation results demonstrate that the proposed approach efficiently generates conflict-free paths satisfying nonholonomic constraints for multi-AMR scheduling systems.

Keywords: multi-AMR system; nonholonomic constraints; spatiotemporal conflict detection; weighted spatiotemporal hybrid A* algorithm; conflict resolution; path planning

0 引言

近年来, 随着科技的快速发展, 越来越多的自主移动机器人 (AMR) 被应用于工业生产和仓储物流系统中^[1]. 目前, 在多 AMR 规划领域所面临的难点在于, 在解决多个 AMR 间的路径交叉和冲突的同时确保系统求解的高效性以及所规划的路径能够符合实际 AMR 的运动约束^[2].

在单个 AMR 领域, 研究人员已提出了多种不同

的路径规划算法, 如 Dijkstra 算法、贪婪最佳优先搜索算法、A* 算法等多种规划算法^[3-7]. 多 AMR 系统涉及多个机器人在共享环境中协同工作, 具有更高的复杂性和挑战性. 多 AMR 路径规划算法包含基于优先级的算法^[8-11]、基于人工智能的算法^[12-15]和基于冲突搜索的算法^[16-17]等多类算法. 在基于优先级的算法中, Cooperative A* 算法使用 A* 算法来计算每个 AMR 的最佳路径, 并通过交换信息协调移动避免

收稿日期: 2024-04-07; 录用日期: 2024-08-04.

基金项目: 国家自然科学基金项目 (62003276); 中央高校基本科研业务费专项资金项目 (JZ2022HG TB0299).

[†]通信作者. E-mail: lanlinyu@hfut.edu.cn.

碰撞, 扩展出效率更高的 HCA*算法和 WHCA*算法^[8-10]。但是, 这几种算法均需要同时维护所有 AMR 的通信。PBS 算法按照优先级进行规划: 低优先级的 AMR 在高优先级 AMR 规划的结果上进行修正, 以避免碰撞^[11]; 在基于人工智能的算法中: PRIMAL 算法采用分布式架构, 进行多 AMR 的路径规划, 并对其改进, 提出了 PRIMAL2 算法, 将其扩展为终身规划^[12-13]; MADDPG 算法在连续空间中对其他 AMR 行为进行预测, 实现了多 AMR 间的协同决策^[14]; MAPPO 算法通过限制策略更新的大小来确保训练的稳定性, 使得算法更易收敛^[15]。但是, 这些算法需要大量的样本, 且可解释性较差。在基于冲突搜索的算法中, CBS 算法是一种分层策略的多 AMR 路径规划算法, 算法底层对每个 AMR 进行独立的路径规划, 上层通过检测底层路径间的冲突为底层算法划分时空冲突约束, 然后再次调用底层进行规划, 从而解决多 AMR 中的冲突问题^[16]。

本文对 CBS 算法进行改进, 提出基于时空冲突和非完整约束下的多 AMR 调度算法。主要内容如下: 1) 将 CBS 算法上层的时空冲突检测树与分离轴定理相结合, 以实现精准的时空冲突检测; 2) 将 CBS 算法底层的时空 A*算法改进为时空混合 A*算法, 采用阿克曼类型 AMR 的运动学模型进行节点扩展, 使得算法规划的路径符合实际工程需要; 3) 将底层算法改进为加权时空混合 A*算法, 以静态地图下的 A*算法规划的路径成本作为启发函数的启发值, 并进行加权, 提升算法的规划效率。

1 问题描述

传统 CBS 算法采用时空 A*算法作为底层算法。时空 A*算法以栅格地图为基础进行路径规划, 其规划出的路径在转弯时为直角, 而阿克曼类型 AMR 在转弯时的路径为圆弧, 导致阿克曼类型 AMR 无法按照时空 A*算法规划的路径运动。因此, 本文以多 AMR 调度系统为研究对象, 目的是考虑 AMR 非完整约束为阿克曼模型的情况下, 为所有 AMR 规划出无冲突的可行路径。

1.1 阿克曼转向 AMR 数学模型

本文的研究对象为阿克曼转向 AMR, 具体运动方式如图 1 所示。其中: 阿克曼转向 AMR 的状态由 (x, y, θ) 表示; AMR 后轴的中心位置为机体坐标系的原点, 其在惯性坐标系 (XOY) 下的坐标为 (x', y') ; θ 为 AMR 的机体坐标系与惯性坐标系的夹角, 定义为偏航角; AMR 的绕动转向中心为 O' 。

AMR 的角速度 ω 通过 AMR 的前轮转角 Φ 和线

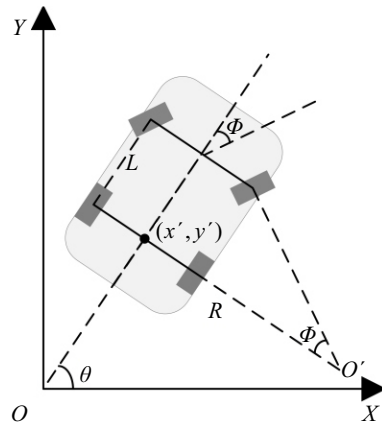


图1 基于阿克曼转向几何的模型示意图

速度 v 进行控制, 其计算公式为

$$\omega = \frac{v}{R} = \frac{v \tan(\Phi)}{L}. \quad (1)$$

其中: R 为 AMR 后轴中心到绕动转向中心 O' 的距离, L 为 AMR 前轴到 AMR 后轴的距离。因此, 在离散时间下, 阿克曼转向 AMR 在惯性坐标系下的运动学模型为

$$\theta_t = \theta_{t-1} + \frac{v \tan(\Phi_{t-1})}{L}, \quad (2)$$

$$x_t = x'_{t-1} + v \cos(\theta_{t-1}) + L \cos(\theta_{t-1})/2, \quad (3)$$

$$y_t = y'_{t-1} + v \sin(\theta_{t-1}) + L \sin(\theta_{t-1})/2, \quad (4)$$

这里 (x, y) 为 AMR 中心在惯性坐标系下的位置。

1.2 问题分析

考虑 N 个 AMR, 在包含 M 个静态障碍物 O 的封闭仓储空间 C 内运动。定义如下: AMR 的位置坐标为 AMR 中心在惯性坐标系下的位置, 表示为 (x, y) ; 偏航角为 AMR 自身坐标系与惯性坐标系的夹角, 表示为 θ 。因此, 第 i 个 AMR 的起点坐标为 $S_i = (x_{s_i}, y_{s_i}, \theta_{s_i})$, 终点坐标为 $G_i = (x_{g_i}, y_{g_i}, \theta_{g_i})$, 在离散时间下, 其路径表示为 $P_i = [p_i^1, p_i^2, \dots, p_i^t]$ 。其中: p_i^t 为第 t 个时刻 AMR _{i} 的位置坐标和偏航角, $\text{Space}(p_i^t)$ 为 AMR _{i} 在该时刻所采用的包围盒所占据的空间, $\text{Space}(O)$ 为所有静态障碍物本体所占据的空间。

若规划出的路径 P 满足以下条件, 则表明该路径是无冲突的且 AMR 可执行的路径:

- 1) 规划出的路径需要满足 AMR 的非完整约束;
- 2) AMR 在规划出的路径运行时不会与任何静态障碍物发生碰撞, 即 $\text{Space}(p_i^t) \cap \text{Space}(O) = \emptyset$;
- 3) 任意两个 AMR 在规划出的路径上运行时不会发生碰撞, 即 $\text{Space}(p_i^t) \cap \text{Space}(p_j^t) = \emptyset (\forall i \neq j)$ 。

考虑问题的合理性, 做出如下假设。

- 1) AMR 的起始位置与终止位置不同, 且不会与静态障碍物发生碰撞, 即 $S_i \neq G_i, \text{Space}(S_i) \cap \text{Space}(O)$

$= \emptyset, \text{Space}(G_i) \cap \text{Space}(O) = \emptyset;$

2) 任意两个 AMR 的起始位置与终止位置不会发生碰撞, 即 $\text{Space}(S_i) \cap \text{Space}(S_j) = \emptyset, \text{Space}(G_i) \cap \text{Space}(G_j) = \emptyset (\forall i \neq j).$

2 基于时空冲突和非完整约束下的多 AMR 调度算法

本文对基于冲突的搜索 (CBS) 算法进行改进, 底层采用加权时空混合 A* 算法, 在基于阿克曼运动学模型的非完整约束下为单个 AMR 规划路径. 当所有 AMR 均规划出自己的路径后, 在上层采用结合分离轴定理的冲突检测树, 对所有 AMR 的路径进行时空冲突检测. 当上层算法判断底层算法规划出的路径无时空冲突时, 表示规划结束. 此时, 算法为所有 AMR 规划出无冲突的满足非完整约束的可行路径.

2.1 时空冲突检测树

CBS 算法的冲突检测和时空约束均是根据栅格的位置进行的, 但是, 本文研究的 AMR 根据非完整约束进行节点扩展, 无法使用栅格进行冲突检测和时空约束. 因此, 本文使用包围盒表示 AMR 所占据的空间进行冲突检测和时空约束.

在冲突检测时, 若在相同时间 t 下存在两个 AMR_{*i*} 和 AMR_{*j*}, 其包围盒所占空间 $\text{Space}(p_i^t)$ 与 $\text{Space}(p_j^t)$ 存在交集, 则发生冲突. 此时分别对两个 AMR 增加时空约束, 第 1 组为 AMR_{*i*} 增加时空约束 $\{p_j, t\}$, 表示在第 t 个时刻 AMR_{*i*} 的包围盒所占空间 $\text{Space}(p_i^t)$ 不允许与空间 $\text{Space}(p_j^t)$ 存在交集, 而 AMR_{*j*} 不加约束; 第 2 组为 AMR_{*j*} 增加时空约束 $\{p_i, t\}$, 而 AMR_{*i*} 不加约束.

一般的 AMR 冲突检测采用最小包围圆作为包围盒, 如图 2 所示. 但是, 此时 AMR 的两侧存在大量冗余空间, 使得原本不会发生冲突的 AMR 仍然检测出冲突. OBB 包围盒即矩形包围盒, 而实际 AMR 的形状近似矩形. 因此, 本文选用 OBB 包围盒来近似表示 AMR 占据的空间.

设每个 OBB 包围盒的顶点坐标分别为 $D_1 = (-rl, rw), D_2 = (rl, rw), D_3 = (rl, -rw), D_4 =$

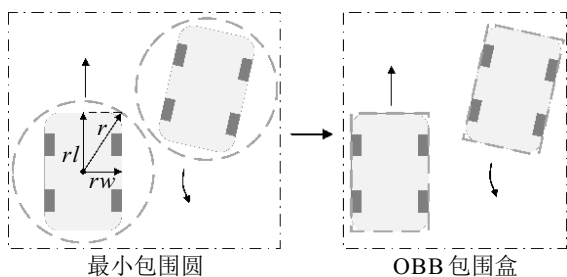


图2 两种包围盒下 AMR 的冲突情况

$(-rl, -rw),$ 其每个顶点的实际坐标的计算公式为

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix}. \quad (5)$$

其中: (x, y) 为 OBB 包围盒的中心点坐标, 即 AMR 的位置; θ 为 OBB 包围盒的旋转角度, 即 AMR 的偏航角. 然后, 计算由相邻顶点 $D_m = (D_{mx}, D_{my})$ 和 $D_n = (D_{nx}, D_{ny})$ 组成向量 $\vec{D}_{mn} = D_m - D_n (mn \in \{12, 23, 34, 41\})$ 的法向量 $\vec{F}_{mn} = (F_{mnx}, F_{mny})$, 即

$$\begin{bmatrix} F_{mnx} \\ F_{mny} \end{bmatrix} = \frac{1}{\|\vec{D}_{mn}\|} \begin{bmatrix} D_{ny} - D_{my} \\ D_{mx} - D_{nx} \end{bmatrix}. \quad (6)$$

对于 AMR_{*i*} 和 AMR_{*j*}, 包围盒分别为 OBB_{*i*} 和 OBB_{*j*}, 包围盒的顶点 D 向所有法向量 \vec{F}_{imn} 和 \vec{F}_{jmn} 中的任意法向量 \vec{F} 进行投影, 结果 Pr 为

$$\text{Pr} = D_x F_x + D_y F_y. \quad (7)$$

取 OBB_{*i*} 包围盒和 OBB_{*j*} 包围盒的最小和最大投影结果作为投影区间 Intv_i 和 Intv_j , 若存在 $\text{Intv}_i \cap \text{Intv}_j = \emptyset$, 则表明两个包围盒不冲突; 否则, 表明两个包围盒冲突.

2.2 加权时空混合 A* 算法

本文采用加权时空混合 A* 算法作为底层单个 AMR 的路径规划算法, 以实现冲突消解的同时满足 AMR 的非完整约束. 其中 AMR 的非完整约束方式为阿克曼转向几何方式, 如图 1 所示.

基于上述非完整约束, AMR 通过选取不同的前轮转角和线速度进行空间节点扩展, 如图 3 所示.

由于采用非完整约束的方式进行空间节点扩展, AMR 前轮转角与线速度不同, 产生的空间扩展节点也不同, 且在多 AMR 调度系统中, 需要将其他 AMR

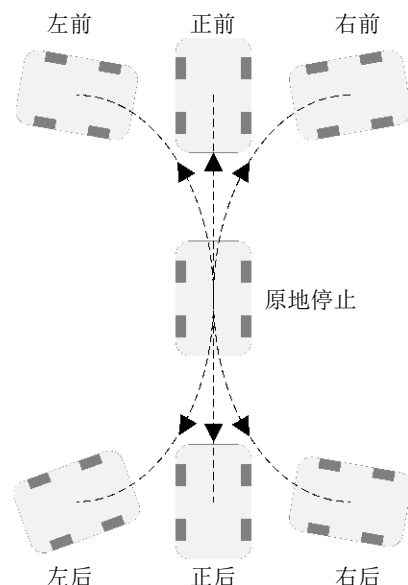


图3 非完整约束下的空间节点扩展

视为时空约束,同时考虑空间维度和时间维度进行求解.因此,相较于传统的A*算法,所提出算法的扩展节点数量更多,规划速度更慢.

鉴于此,本文进一步将底层单AMR规划算法改进为加权时空混合A*算法,以静态地图下的A*算法规划的路径成本作为启发函数的启发值,并引入权重参数,增加启发式估计代价的权重,使得算法更加倾向于沿着启发式估计代价的方向进行搜索.此时的节点信息为 $(p, t, cost, dir, angle)$.其中: p 为AMR中心在全局坐标系下的位置和偏航角, t 为在当前节点AMR已经运动的时间, $cost$ 为当前节点的代价值.为了使得规划的路径符合实际场景需求,在代价函数 $cost$ 中加入变向惩罚和倒车惩罚.这里: dir 为当前节点下AMR的运动方向, $dir = 1$ 表示前进, $dir = -1$ 表示后退, $dir = 0$ 表示原地停止; $angle$ 为AMR的前轮转角.算法在节点 n 处的代价函数 $cost(n)$ 的计算公式如下所示:

$$cost(n) = g(n) + \lambda h(n). \quad (8)$$

其中: $h(n)$ 为在该节点的启发式估计代价; λ 为启发式估计代价加权常数; $g(n)$ 为当前节点的实际代价,其计算公式为

$$g(n) = \alpha g_1 + \beta g_2 + \gamma g_3 + g_4; \quad (9)$$

$$\begin{cases} \alpha = 1, & dir \neq -1; \\ \alpha > 1, & dir = -1. \end{cases} \quad (10)$$

这里: g_1 为每走一步的代价, α 为倒车惩罚系数, g_2 为AMR前轮转角 $angle$ 的大小, β 为转向惩罚系数, g_3 为前轮转角变化值, γ 为变向惩罚系数, g_4 为之前的代价和.加权时空混合A*算法流程如图4所示.

2.3 算法复杂度分析

针对所提出基于加权时空混合A*的冲突搜索算法(WSTH-CBS),当上层共存在 c 次冲突时,上层的时间复杂度为 $O(2^c)$.由于算法底层的加权混合A*算法的启发函数采用加权方式,当 $\lambda > 1$ 时,算法更偏向目标点方向进行扩展,底层加权混合A*算法的时间复杂度为 $O(b^{d/\lambda})$.此时,WSTH_CBS算法的实际时间复杂度为 $O(2^c \cdot b^{d/\lambda})$.

3 仿真实验分析

分别在空白地图场景和随机障碍场景进行测试,每种场景中AMR的数量分别为5辆、10辆、15辆.场景地图的大小分别为 $50\text{ m} \times 50\text{ m}$ 、 $80\text{ m} \times 80\text{ m}$ 、 $100\text{ m} \times 100\text{ m}$.随机障碍场景中障碍物的个数分别为10、20、40.测试所提出基于时空混合A*的冲突搜索算法(STH-CBS)和基于加权时空混合A*的冲

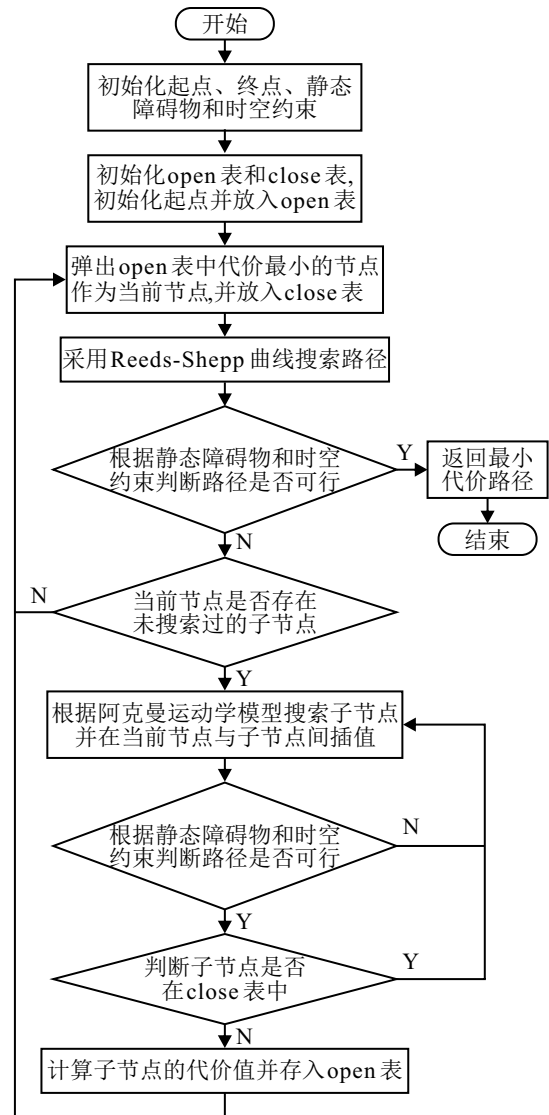


图4 加权时空混合A*算法流程

突搜索算法(WSTH-CBS)相较于基于优先级的搜索算法(PBS)在规划效率上的提升,并对每种场景地图进行5组测试取平均值表示算法的实际性能.

表1为算法规化时间对比.由表1可知,STH-CBS算法与WSTH-CBS算法的规划效率高于PBS算法.而且当规划AMR数量变多和场景变复杂时,PBS算法后规划的AMR的时空约束过多,会导致可行路径过少难以搜索,从而规划效率相较于另外两种算法变得更低.

为了测试所提出算法在仓储环境下的性能,再次进行仿真实验,场景地图的大小为 $50\text{ m} \times 50\text{ m}$,且再增加3种基于贝塞尔曲线和改进安全区间的路径规划算法作为对比算法,分别为Bezier Anytime SIPP(BA-SIPP)算法、Bezier Weighted SIPP(BW-SIPP)算法和Bezier Focal SIPP(BF-SIPP)算法.

表2为规模仓储环境下的不同算法规划结果对比.由表2可知,前3种方法的规划时间较短,但是

表1 算法规划时间对比

场景类型	地图大小	规划算法	规划时间/s
空白地图	50 m × 50 m	PBS	4.814
		STH-CBS	2.730
		WSTH-CBS	2.451
空白地图	80 m × 80 m	PBS	21.675
		STH-CBS	20.655
		WSTH-CBS	18.972
空白地图	100 m × 100 m	PBS	110.666
		STH-CBS	55.915
		WSTH-CBS	52.088
随机障碍	50 m × 50 m	PBS	5.393
		STH-CBS	4.571
		WSTH-CBS	4.340
随机障碍	80 m × 80 m	PBS	56.487
		STH-CBS	31.448
		WSTH-CBS	22.964
随机障碍	100 m × 100 m	PBS	232.450
		STH-CBS	82.548
		WSTH-CBS	62.908

表2 50 m × 50 m 规模仓储环境下不同算法规划结果对比

规划算法	平均规划时间/s	平均路径长度/m	平均成功率/%
BA-SIPP	5.12	40.432	80
BW-SIPP	4.262	42.246	80
BF-SIPP	4.844	41.432	84
STH-CBS	11.722	38.848	100
WSTH-CBS	7.064	40.352	100

规划的路径长度较长, 且由于进行曲线平滑导致规划路径与平滑后路径存在偏差, 从而降低了规划成功率. 而所提出算法由于基于 AMR 运动学模型进行规划, 算法规划成功率始终为 100%.

为了测试所提出 STH-CBS 算法和 WSTH-CBS 算法的性能随着 AMR 数量逐渐增多的变化情况, 再次在仓储环境下进行仿真实验, 场景地图的大小为 50 m × 50 m. 算法对比结果如图 5 所示. 其中: 横轴为 AMR 数量, 左侧纵轴为算法规划时间, 右侧纵轴为算法规划的平均路径长度, 带有方块和圆点的线段分别为 STH-CBS 算法和 WSTH-CBS 算法的规划时间, 带有五角星和菱形的线段分别为 STH-CBS 算法和 WSTH-CBS 算法规划的平均路径长度.

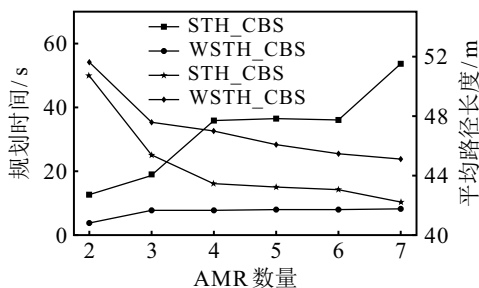


图5 仓储 50 m × 50 m 场景算法对比

AMR 数量每次增加 1 台, 逐渐增加至 7 台. 6 次仿真实验 STH-CBS 算法的平均规划时间为 32.29 s, 平均路径长度为 44.67 m; WSTH-CBS 算法的平均规划时间为 7.28 s, 平均路径长度为 47.13 m. WSTH-CBS 算法相较于 STH-CBS 算法的平均路径长度增长 5.51%, 平均规划时间减少 77.45%.

采用所提出 WSTH-CBS 算法, 在仓储场景中进行规划, 其 AMR 的时空轨迹如图 6 所示, 图 7 为截取图 6 中不同时刻的 AMR 位置. 其中: 黑色部分为仓储货架, 虚线矩形为 AMR 的起点位置, 连接的虚线为该 AMR 的运动轨迹, 填充的矩形为 AMR 在当前时刻的位置. 图 7(a) 为 AMR 运动 10 s 的位置和轨迹, 此时上方 AMR 向右转弯, 下方 AMR 向左转弯, 成功实现避碰; 图 7(b) 为 AMR 最终位置和运动

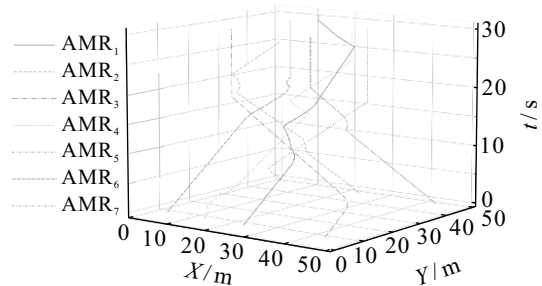
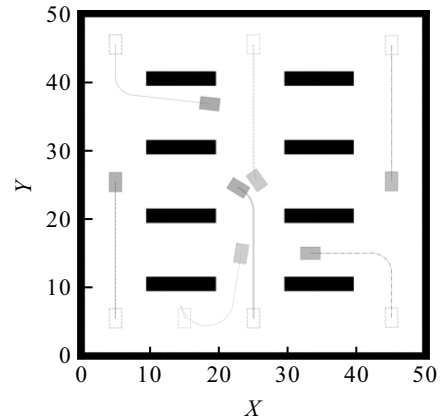
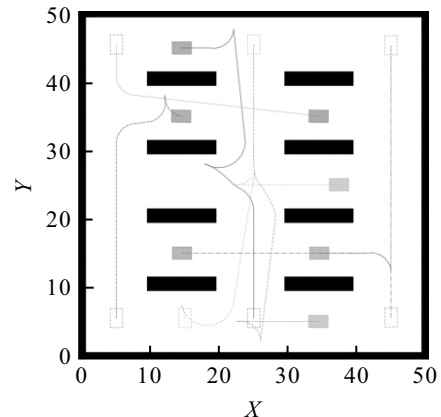


图6 仓储 50 m × 50 m 场景 AMR 时空轨迹



(a) AMR 运动 10 s 的轨迹



(b) AMR 最终的运动轨迹

图7 仓储 50 m × 50 m 场景 AMR 不同时刻位置

轨迹.

4 结论

本文针对多 AMR 规划问题,对 CBS 算法进行了改进,提出了一种分层策略的基于时空冲突和非完整约束下的多 AMR 调度算法.通过结合分离轴定理的冲突检测树作为算法的上层,进行了精确地时空冲突检测.采用加权时空混合 A*算法作为算法的底层,进行了单 AMR 路径规划.仿真结果表明所提出算法能够成功为多 AMR 调度系统高效地规划出满足非完整约束的无冲突路径.

参考文献 (References)

- [1] 姚宇,莫鹏里,商攀,等. B2B 配送中考虑时间窗一致性的车辆路径优化模型与算法[J]. 控制与决策, 2024, 39(1): 244-252.
(Yao Y, Mo P L, Shang P, et al. Model and algorithm for vehicle routing problem considering time window consistency in B2B distribution[J]. Control and Decision, 2024, 39(1): 244-252.)
- [2] 纪苗苗,吴志彬. 考虑工人路径的多智能体强化学习空间众包任务分配方法[J]. 控制与决策, 2024, 39(1): 319-326.
(Ji M M, Wu Z B. A multi-agent reinforcement learning algorithm for spatial crowdsourcing task assignments considering workers' path[J]. Control and Decision, 2024, 39(1): 319-326.)
- [3] Wang J Y, Li Y H, Li R X, et al. Trajectory planning for UAV navigation in dynamic environments with matrix alignment Dijkstra[J]. Soft Computing, 2022, 26(22): 12599-12610.
- [4] Jesuthas N J A, Somaskandan S. Path-finding and planning in a 3D environment an analysis using bidirectional versions of Dijkstra's, weighted A*, and greedy best first search algorithms[C]. Proceedings of the 2nd Asian Conference on Innovation in Technology. Ravet, 2022: 1-8.
- [5] Liu S, Liu S H, Xiao H P. Improved gray wolf optimization algorithm integrating A* algorithm for path planning of mobile charging robots[J]. Robotica, 2024, 42(2): 536-559.
- [6] 周熙栋,张辉,陈波. 非结构化场景下基于改进 JPS 算法的机器人路径规划[J]. 控制与决策, 2024, 39(2): 474-482.
(Zhou X D, Zhang H, Chen B. Mobile robot path planning based on improved JPS algorithm in unstructured scenarios[J]. Control and Decision, 2024, 39(2): 474-482.)
- [7] Dolgov D, Thrun S, Montemerlo M, et al. Path planning for autonomous vehicles in unknown semi-structured environments[J]. The International Journal of Robotics Research, 2010, 29(5): 485-501.
- [8] Liu Y M, Chen M X, Huang H J. Multi-agent pathfinding based on improved cooperative A* in kiva system[C]. Proceedings of the 5th International Conference on Control, Automation and Robotics. Beijing, 2019: 633-638.
- [9] Bnaya Z, Felner A. Conflict-oriented windowed hierarchical cooperative A*[C]. IEEE International Conference on Robotics and Automation. Hong Kong, 2014: 3743-3748.
- [10] Silver D. Cooperative pathfinding[J]. Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 2021, 1(1): 117-122.
- [11] Ma H, Harabor D, Stuckey P J, et al. Searching with consistent prioritization for multi-agent path finding[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33(1): 7643-7650.
- [12] Sartoretti G, Kerr J, Shi Y F, et al. PRIMAL: Pathfinding via reinforcement and imitation multi-agent learning[J]. IEEE Robotics and Automation Letters, 2019, 4(3): 2378-2385.
- [13] Damani M, Luo Z Y, Wenzel E, et al. PRIMAL₂: Pathfinding via reinforcement and imitation multi-agent learning-lifelong[J]. IEEE Robotics and Automation Letters, 2021, 6(2): 2666-2673.
- [14] Lowe R, Wu Y, Tamar A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[J/OL]. 2017, arXiv: 1706.02275.
- [15] Zhan G, Zhang X M, Li Z C, et al. Multiple-UAV reinforcement learning algorithm based on improved PPO in ray framework[J]. Drones, 2022, 6(7): 166.
- [16] Sharon G, Stern R, Felner A, et al. Conflict-based search for optimal multi-agent pathfinding[J]. Artificial Intelligence, 2015, 219: 40-66.

作者简介

申子祥 (2000-), 男, 硕士生, 主要研究方向为多智能体深度强化学习、路径规划, E-mail: zixiang.shen@mail.hfut.edu.cn;

孙训红 (1998-), 男, 博士生, 主要研究方向为有限时间多智能体协同控制, E-mail: xunhong.sun@mail.hfut.edu.cn;

从永正 (1997-), 男, 博士生, 主要研究方向为有限时间多智能体协同控制, E-mail: yongz.cong@mail.hfut.edu.cn;

余兰林 (1992-), 女, 副研究员, 博士, 主要研究方向为复杂网络系统模型分析及优化协同控制、多智能体深度强化学习, E-mail: lanliny@hfut.edu.cn.