

DOI: 10.13382/j.jemi.B2205453

融合 A* 蚁群和动态窗口法的机器人路径规划*

赵倩楠^{1,2} 黄宜庆^{1,2}

(1. 安徽工程大学电气工程学院 芜湖 241000; 2. 高端装备先进感知与智能控制教育部重点实验室 芜湖 241000)

摘 要: 针对蚁群算法在全局路径规划时无目的搜索、收敛慢和规划的路径不平滑等问题,本文提出了一种融合 A* 蚁群和动态窗口法(dynamic window algorithm, DWA)的平滑路径规划方法。首先,对于传统蚁群算法,利用改进 A* 算法非均匀分配初始信息素,解决算法初期搜索无目的问题;给出算法自定义的移动步长和搜索方式,提高路径寻优效率;修改转移概率函数中的启发函数值并增加障碍物影响因子,在避免死锁现象的同时加快收敛速度;采用二次路径优化策略,使得路径更短更平滑;其次在动态窗口法的评价函数中引入动态避障评价子函数,提高路径的安全性。仿真实验结果表明,改进 A* 蚁群算法较传统蚁群算法可减少 8.75% 的路径长度和 59% 的转折点数,融合优化动态窗口法后,移动机器人既能保证在静态环境下规划出全局最优的路径,又能实现动态环境下的路径规划,有效躲避环境中出现的动态障碍物。

关键词: 蚁群算法;动态窗口算法;动态障碍物;路径规划;移动机器人

中图分类号: TP242.6 **文献标识码:** A **国家标准学科分类代码:** 510.80

Robot path planning based on A* ant colony and dynamic window algorithm

Zhao Qiannan^{1,2} Huang Yiqing^{1,2}

(1. College of Electrical Engineering, Anhui University of Engineering, Wuhu 241000, China;
2. Provincial Key Laboratory of Detection Technology and Energy Saving Devices, Wuhu 241000, China)

Abstract: Aiming at the problems of aimless search, slow convergence and unsmooth path planning of ant colony algorithm in global path planning, this paper proposes a smooth path planning method that combines A* ant colony and dynamic window algorithm. First, for the traditional ant colony algorithm, the improved A* algorithm is used to distribute initial pheromones unevenly to solve the aimless problem of initial search of the algorithm. The self-defined moving step size and searching method are given to improve the efficiency of path optimization. The heuristic function value in the transition probability function is modified and the obstacle influence factor is added to avoid deadlock and speed up the convergence. The secondary path optimization strategy is adopted to make the path shorter and smoother. Secondly, the dynamic obstacle avoidance evaluation sub function is introduced into the evaluation function of the dynamic window method to improve the path safety. The simulation results show that the improved A* ant colony algorithm can reduce the path length by 8.75% and the turning points by 59% compared with the traditional ant colony algorithm. After the dynamic window method is integrated and optimized, the mobile robot not only ensures the global optimal path planning in the static environment, but also realizes the path planning in the dynamic environment, effectively avoids dynamic obstacles in the environment.

Keywords: ant colony algorithm; dynamic window algorithm; dynamic obstacle; path planning; mobile robot

0 引 言

目前,移动机器人路径规划技术在自主导航领域占

有重要地位,已成为国内外学者研究的热点之一^[1-2],其任务是在避开障碍物的同时找到一条满足要求的安全最优路径^[3]。根据环境内的障碍物是否移动,将其分为静态路径规划^[4-6]和动态路径规划两类^[7-8],但实际生活中,

移动机器人大多都处于动态工作环境下,因此研究机器人如何在具有动态障碍物的环境下找到最优路径更具有实际意义。

目前不少学者已经提出了多种规划方法,其中,蚁群算法因并行性、易与其他算法融合等优点被普遍使用,但是算法本身存在速度慢、易堕入局部解等问题。文献[9]引入闭环反馈思想,利用处理后的反馈信息动态调节蚁群参数,从而规划出一条光滑路径。文献[10]提出蚁群-聚类自适应动态路径规划方法,通过平滑机制进行动态路径优化。文献[11]利用改进蚁群算法进行全局规划以及结合局部避碰策略进行局部规划,但与机器人的实际运动轨迹不符。文献[12]提出 DWA 算法规划出平滑路径且具有避障能力,但该算法容易转圈陷入死锁。文献[13]提出一种改进 DWA 算法,优化速度空间的同时提高了避障效率。文献[14]在动态窗口法中引入一种预测发生碰撞的模型,有效减少了在动态环境下的碰撞次数。但随着路径环境复杂化以及对规划路径的高精度要求,规划算法也在不断的更新提高,通常使用全局规划与局部避障相结合的融合算法。文献[15]利用 *Manhattan* 和 *Euclidean* 距离构造 A* 算法新的启发函数并剔除其多余路径点,再结合动态窗口法进行实时规划,使路径更加平滑。文献[16]使用双向信息素不均匀分配的蚁群算法和设置动态初始航向角的 DWA 算法相结合,实现在动态环境中的全局最优路径实时规划。

虽然上述算法一定程度上提高了路径规划效率和质量,但没有完全解决蚁群算法规划的路径折点多、不平滑及容易陷入死锁等问题。因此本文提出一种新的规划方法,首先,对蚁群算法提出改进:利用 A* 算法差异化蚁群的初始信息素,减少无效搜索;采用变步长搜索方式扩大蚂蚁搜索范围,提高搜索速度并缩短路径长度;在启发函数中引入放大系数 C 和 A* 算法的预估代价值,在增加多样性的同时增加目标点的向导作用,与此同时,结合障碍物影响因子避免路径陷入死锁;采用二次路径规划策略删除冗余节点。其次,在 DWA 的评价函数中引入动态避障评价子函数,提高实时避障能力。通过融合两种算法的优点,最终实现在不同环境下都可以及时避开障碍物找出全局最优路径。

1 基本算法

1.1 蚁群算法状态转移概率

蚂蚁会在经过的路径上释放信息素,蚁群算法就是将这一行为抽象化后得到的一种搜索算法。蚂蚁 $k(k = 1, 2, \dots, m)$ 在 t 时刻从位置 i 选择下一节点 j 的概率如式(1)所示:

$$p_{ij}^k(t) =$$

$$\begin{cases} \frac{[\tau_{ij}(t)]^\alpha \times [\eta_{ij}(t)]^\beta}{\sum_{s \in allowed_k} [\tau_{is}(t)]^\alpha \times [\eta_{is}(t)]^\beta}, & j \in allowed_k \\ 0, & j \notin allowed_k \end{cases} \quad (1)$$

式中: $\tau_{ij}(t)$ 为 t 时刻位置 i 与位置 j 路径上的信息素; $\eta_{ij}(t)$ 为启发函数, $\eta_{ij}(t) = 1/d_{ij}$, d_{ij} 为位置 i 与位置 j 之间的欧氏距离; α 为信息启发因子; β 为期望启发因子; $j \in allowed_k$ 为待选节点集合。

1.2 蚁群算法信息素更新

算法运行过程中,需要对所有完成路径规划的蚂蚁,不断地进行路径上信息素浓度的更新。更新方法如下:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (2)$$

式中: ρ 为挥发系数 ($0 < \rho < 1$); $\tau_{ij}(t)$ 为栅格 i 与栅格 j 之间的信息素浓度; m 为蚂蚁数目;且有:

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{l_k}, & \text{蚂蚁 } k \text{ 经过了路径 } (i, j) \\ 0, & \text{未经过} \end{cases} \quad (3)$$

式中: Q 为信息素强度; l_k 为路径长度。

1.3 环境模型建立

本文采用如图 1 所示栅格法进行对全局模型建立,对环境中的障碍物以机器人尺寸半径向外膨胀一定距离,栅格化后不满一个栅格的也按照一个栅格处理。由于环境中会出现动态和静态障碍物两种情况,动态障碍物采用图中标注的小方块表示,运动方向不定,静态障碍物采用黑色不通行栅格表示,其他为可通行空闲栅格。建立二维直角坐标系,对 N^2 ($N = 1, 2, \dots$) 个栅格依次编号为 $1, 2, \dots, N^2$,可用式(4)表示 (x_i, y_i) 与第 i 个栅格间的关系。

$$\begin{cases} x_i = a \times [\text{bmod}(i, N) - 0.5] \\ y_i = a \times [N + 0.5 - \text{ceil}(i/N)] \end{cases} \quad (4)$$

式中: a 为栅格边长, i 代表栅格序号, $\text{ceil}()$ 是向上取整函数; $\text{bmod}()$ 是求余函数。

2 改进 A* 蚁群算法

2.1 改进 A* 算法初始化信息素

前几代蚂蚁在路径搜索时,由于各栅格间的初始信息素均匀分布为一个常值 A ,会导致沿着信息素浓度高进行寻优的蚂蚁陷入困境。因此增加改进 A* 算法得到路径上的信息素含量,从而差异化蚁群信息素的初始含量,加快初期收敛速度且提高算法的实时性。

A* 算法是运用启发函数 $f(n) = g(n) + h(n)$ 来高效快速寻找路径的搜索方法,鉴于启发性原则, $h(n)$ 对算法能否规划出最优路径有很大的影响,当 $h(n)$ 刚好等于

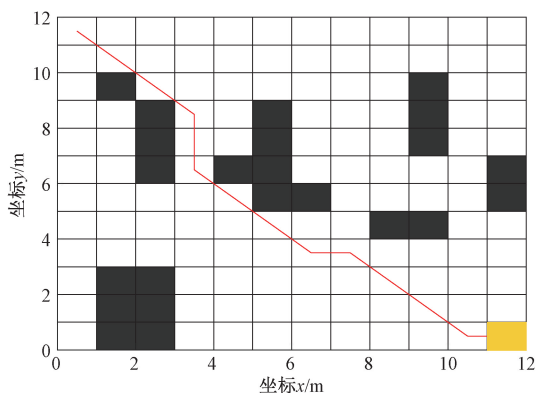


图 1 传统算法规划路径

Fig. 1 Traditional algorithm planning path

栅格 n 到终点的实际代价值时,可以达到最高搜索效率,这种理想状态很难实现,但在此基础上为不同环境下的启发代价设置一定的权重也显得很有必要,设置新的 A^* 算法启发函数为:

$$f(n) = g(n) + wh(n) \quad (5)$$

式中:通常取 $w \geq 1$,实际应用中可以灵活调动 w 的值,提高 A^* 算法的运行效率。

本文将按照式(6)增加 A^* 算法得到路径 R 上的信息素浓度,剩余路径上的浓度仍保持常值 A 。

$$\tau(R) = NA \quad (6)$$

式中: N 为大于 0 的常数。

2.2 变步长搜索方式

由于栅格地图中的蚂蚁只能够在其周围的相邻 4 个或者 8 个栅格选择下一步路径节点,这样不但收敛速度慢、出现多余路径尖角,而且可能会导致算法根本无法寻找到最优路径。变步长移动策略不再限制蚂蚁每次只移动一个栅格,多步长的搜索可以帮助蚂蚁提前发现目标点,也有利于更短路径的形成。但考虑到设置较大的步长,会对障碍物的轮廓刻画不准确,机器人在离障碍物较近时,也可能出现规划不出路径的情况,甚至还会带来安全隐患。因此本文提出如图 2 所示即活动范围为 2 的示意图,机器人 R 处于正中间的栅格中,在避开障碍物后,圆圈为机器人可直接到达点,鉴于路径优化原则,在选择下一步落脚点时一般采用由远及近的优先顺序。这样移动机器人可移动方向增多,会使路径长度更短更平滑。

2.3 改进转移概率公式

1) 启发函数的改进

传统蚁群算法中, $\eta_{ij}(t)$ 是栅格 i 和 j 之间距离的倒数,当 d_{ij} 越小,则栅格 j 被选择的概率越大,致使蚂蚁选择当前最短的路径放弃全局最优路径,且容易陷入死锁。因此,本文将预估代价值 d_{je} 加入到蚁群算法的启发函数

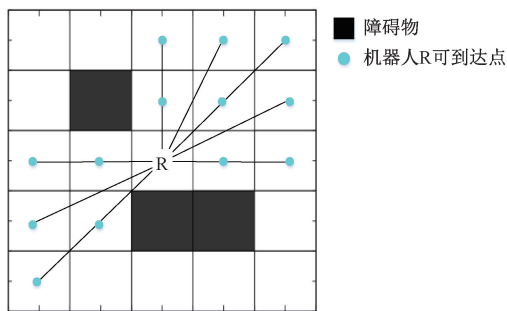


图 2 蚁群算法变步长搜索方式

Fig. 2 Variable step search method of ant colony algorithm

中,在蚁群算法中增加目标点的向导作用,以提高算法收敛速度和获得更优解。与此同时,又考虑到将下一待选栅格 j 与目标点 e 引入到启发函数中,在使得路径更有方向性的同时也会使得算法搜索空间变小。为了丰富算法解的多样性,在规划的初始阶段不引入目标点信息,利用传统蚁群启发函数,对整体的启发信息影响不大,因此本文在初始阶段引入启发函数放大系数 C ,从而更好地启发蚂蚁向短距离节点移动。这样既能保证算法解的多样性,又能快速找到目标点。故本文将启发函数改为:

$$\eta_{ij}^*(t) = \begin{cases} \frac{C}{d_{ij}}, & \text{其他} \\ (1 - \zeta)d_{ij} + \zeta d_{je}, & d_{si} \geq d_{ie} \end{cases} \quad (7)$$

式中: d_{ij} 表示栅格 i 到 j 的距离; d_{je} 表示下一待选节点到目标节点 e 的距离; d_{si} 表示起点到栅格 i 的距离; d_{ie} 表示栅格 i 到目标点 e 的距离; C 为放大系数; ζ 为常数。

2) 障碍物影响因子

蚁群在初期阶段或复杂环境下搜索时,大量的交叉路径和算法禁忌表的限制,极大可能使蚂蚁堕入死锁,如图 3 所示,蚂蚁移动到栅格 j 时,此时蚂蚁无路可走,陷入死锁,文献[17]提出蚂蚁回退策略,但频频回退会增加算法的搜索时间,使效率降低。因此本文引入待选栅格周围障碍物的信息影响因子,障碍物影响因子为:

$$Noif_j(t) = \begin{cases} 0, & \text{领域内连续 6 个栅格为障碍栅格} \\ \frac{1}{\sum_{obs \in jn} d_{j,obs}}, & \text{其他情况} \end{cases} \quad (8)$$

式中: $\sum_{obs \in jn} d_{j,obs}$ 表示下一待选栅格 j 到其周围邻域障碍物的距离之和;图 3 中待选栅格 j 的领域内有连续的 6 个障碍栅格,那么 $Noif_j(t)$ 为 0,若下一待选栅格为 $j \sim$ 时, $j \sim$ 的领域栅格内有 2 个障碍物,那么 $Noif_j(t)$ 表示栅格 $j \sim$ 到 2 个障碍物距离之和的倒数。

综上所述,在基本蚁群算法的基础上改进启发函数和添加障碍物影响因子,改进后的状态转移概率公式为:

$$p_{ij}^k(t) =$$

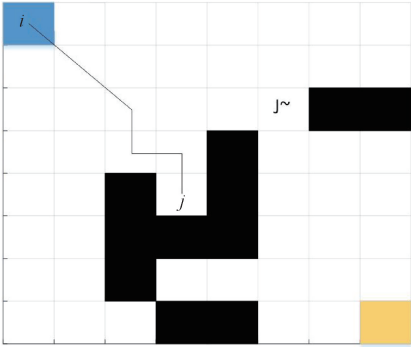


图3 死锁现象

Fig. 3 Deadlock phenomenon

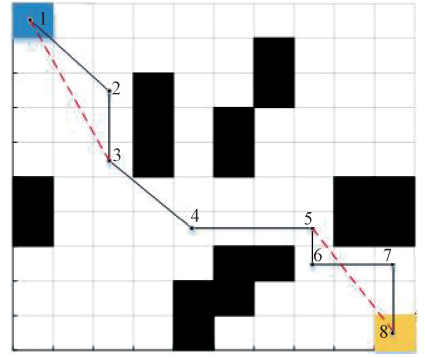


图4 结果对比

Fig. 4 Comparison of result

$$\begin{cases} \frac{[\tau_{ij}(t)]^\alpha \times [\eta_{ij}^*(t)]^\beta [Noif_j(t)]^\gamma}{\sum_{s \in allowed_k} [\tau_{is}(t)]^\alpha \times [\eta_{is}^*(t)]^\beta [Noif_s(t)]^\gamma}, j \in allowed_k \\ 0, j \notin allowed_k \end{cases} \quad (9)$$

式中: $\eta_{ij}^*(t)$ 为改进启发函数值; $Noif_j(t)$ 为下一待选栅格障碍物影响因子; γ 为障碍物影响因子权值。

2.4 二次路径规划

针对在栅格环境下,规划出来的路径存在转弯次数多、路径不平滑等问题,提出路径二次规划策略,可以对没必要的中间节点进行删除,从而缩短路径长度,提高工作效率。

假设改进蚁群算法得到的一条路径为 P , 将路径上的拐点依次记为 $p_1, p_2, p_3, \dots, p_n$, 如图4所示,黑色线段代表路径 P , 此图中 $n = 8$ 。然后从 p_1 , 依次连接 $p_2, p_3, \dots, p_m$, 判断线段 $p_1 p_m$ 是否是安全无碰路径, 若不是, 则 p_{m-1} 为路径关键节点, 若是, 则剔除 $p_2, \dots, p_{m-1}$, 接着从 p_m 继续连接路径上的节点进行判断, 直到判断到终点 p_n , 依次连接路径上的关键节点。如图4虚线所示, 删除不必要的节点后剩余 p_1, p_3, p_4, p_5, p_8 这5个关键节点。通过对比可以看出, 二次规划后的路径长度从 15.656 缩短到 13.905, 优化了 11.18%, 拐点从 8 个减少到 5 个, 优化了 37.5%, 综上, 经过二次规划的路径转弯次数少且平滑, 有效减少了路径长度。

3 算法融合

3.1 改进动态窗口法

动态窗口法^[18-19]是在速度二维空间中采样多组速度对, 然后模拟出机器人在这些速度对下的轨迹。再评价出最优的速度对完成路径规划。

1) 运动学模型

要想进行运动轨迹模拟, 就要知道运动模型, 故机器

人在 Δt 内做均匀运动的运动模型为:

$$\begin{cases} x = x + v_x \Delta t \cos(\theta_t) - v_y \Delta t \sin(\theta_t) \\ y = y + v_x \Delta t \sin(\theta_t) + v_y \Delta t \cos(\theta_t) \\ \theta(t) = \theta(t) + w_t \Delta t \end{cases} \quad (10)$$

2) 机器人速度模型

在速度空间中存在多组速度对, 但是受下述三者速度的约束, 取其交集将速度限制在一定的范围内。

(1) 机器人本身最大、最小速度

$$v_m = \{(v, w) \mid v \in [v_{\min}, v_{\max}], w \in [w_{\min}, w_{\max}]\} \quad (11)$$

(2) 电机加减速约束

$$v_d = \left\{ \begin{aligned} &(v, w) \mid v \in [v_c - a_{\min} \Delta t, v_c + a_{\max} \Delta t] \\ &w \in [w_c - a_{\min} \Delta t, w_c + a_{\max} \Delta t] \end{aligned} \right\} \quad (12)$$

式中: v_d 表示机器人能够达到的速度空间, $a_{\min}$ 表示最小加速度, $a_{\max}$ 表示最大加速度。

(3) 障碍物约束

$$v_a = \{(v, w) \mid v \leq \sqrt{2 \text{dist}(v, w)} v, w \leq \sqrt{2 \text{dist}(v, w)} w\} \quad (13)$$

式中: $\text{dist}(v, w)$ 是速度 (v, w) 对应的轨迹与最近障碍物的距离。

3) 改进评价函数

评价函数设计原则是让机器人及时避障同时找出全局最优路径。但是当环境中存在动态障碍物时, DWA 算法无法及时避障, 且易陷入局部最优, 因此本文在原有基础上引入动态避障的评价函数 $dym \sim \text{dist}(v, w)$, 使得机器人在感知到障碍物的速度与距离时综合判断下一时刻的速度组。经改进后的融合算法评价函数为:

$$\begin{cases} G(v, w) = \lambda \times \text{head}(v, w) + \psi \times \text{dist}(v, w) + \\ \omega \times \text{vel}(v, w) + \sigma \times dym \sim \text{dist}(v, w) \\ dym \sim \text{dist}(v, w) = a \times |v_0 - v_t| + \\ b \times \sqrt{(x_s - x_g)^2 + (y_s - y_g)^2} \end{cases} \quad (14)$$

式中: $head(v,w)$ 为方向角评价子函数; $dist(v,w)$ 为距离评价子函数; $vel(v,w)$ 为当前速度大小评价子函数; $dym \sim dist(v,w)$ 为动态避障评价函数; $\chi, \psi, \omega, \sigma$ 为各评价子函数加权系数; $|v_0 - v_t|$ 为模拟速度与当前速度差值; $\sqrt{(x_s - x_g)^2 + (y_s - y_g)^2}$ 为当前位置与障碍物中心的距离。

3.2 融合算法规划路径

融合算法路径规划的基本流程伪代码如下所示,通过充分利用两种算法的优势,从而实现安全避障的同时规划出最优路径。

步骤 1) 建立环境栅格地图模型, 设定初始信息素 $\tau_{ij}(0) = \tau_0$, 蚂蚁初始数量为 m , 初始时刻 $t = 0$, 迭代次数 $N_c = 0$, 初始化 α, β 与最大迭代次数 N_{max} 等参数, 设置蚂蚁编号 $k(k = 1, 2, \cdots, m)$;

步骤 2) 选取蚂蚁 k 的起始点 g_{begin} , 并添加至禁忌表中, 初始化当前坐标点 (x_c, y_c) , 并对其当前速度进行采样;

步骤 3) 使用 A* 蚁群算法在可行域 Z 中规划出一条全局最优路径 R_{best} , 确定 DWA 算法的子目标点 (x_{target}, y_{target}) 和起始点 $g_{dwa, target}$;

步骤 4) 基于运动模型对步骤 2) 蚂蚁的坐标点与采样速度生成模拟轨迹 $R_{simulation}$;

步骤 5) 对于步骤 3) 生成的路径与目标点、起始点, 以及步骤 4) 生成的模拟轨迹, 基于改进评价函数 $G(v, w)$ 选择其最优轨迹对应的速度 v 和角速度 w ;

步骤 6) 实验验证沿最优轨迹机器人 k 移动至节点 j ;

- 步骤 7) 判断机器人 k 所在节点 j 是否为目标节点;
 - 情况 1) 若节点 j 是目标节点, 跳至步骤 8);
 - 情况 2) 若未达到目标节点, 更换下一个转折点作为当前目标点, 并返回步骤 3);
- 步骤 8) 判断当前位置是否为全局端点;
 - 情况 1) 若未达到全局端点, 返回步骤 3);
 - 情况 2) 若达到全局端点, 结束循环。

4 仿真实验及结果分析

4.1 改进 A* 蚁群算法实验分析

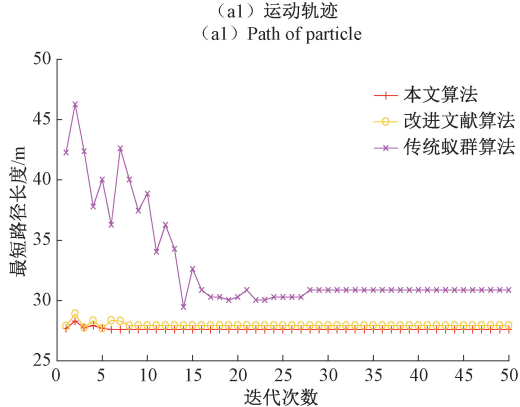
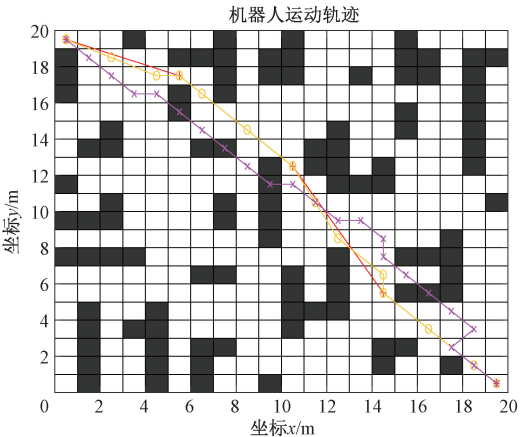
为检验 A* 蚁群算法的寻优性能, 对传统蚁群算法、文献[20]改进算法、本文算法进行仿真对比, 本文算法使用 MATLAB R2018a 编程, 实验参数如表 1 所示。

图 5 是在不同规模栅格下对 3 种算法寻优结果的对比图, 传统蚁群算法规划的路径用带×线段标记, 文献[20]改进算法规划的路径用带圆圈的线段标记, 本文算法规划路径用带十字的线段标记。表 2 是运行 20 次后

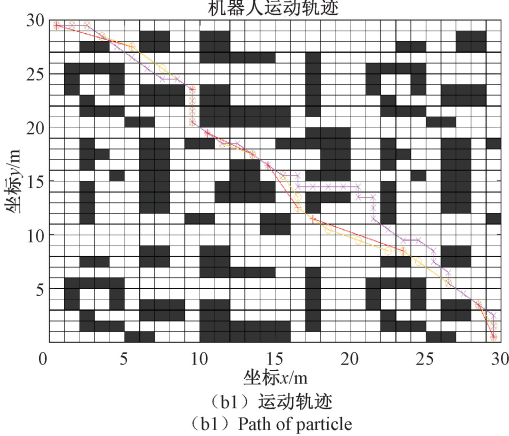
所得出的 3 种算法最优结果对比。

表 1 仿真实验参数

参数	数值
M : 蚂蚁数目	50
N : 迭代次数	50
α : 信息素因子	1
β : 启发函数因子	5
ρ : 为信息初始挥发因子	0.5
Q : 信息素初始值	1



(a) 20×20环境下轨迹迭代图
(a) Trajectory iteration diagram in 20×20 environment



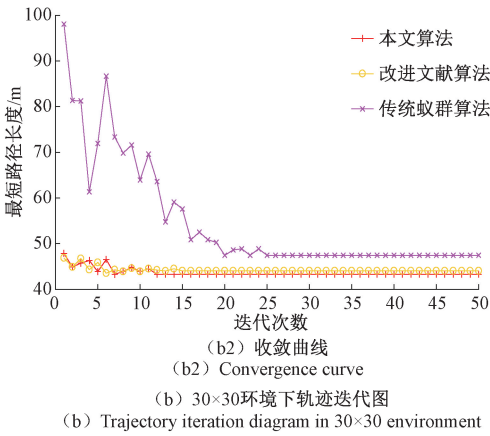


图5 移动机器人路径寻优结果

Fig.5 Mobile robot path optimization results

表2 3种算法结果对比

算法	20×20			30×30		
	路径	迭代	转折	路径	迭代	转折
	长度	次数	点数	长度	次数	点数
传统蚁群算法	29.455	28	10	47.455	25	22
文献[20]算法	27.914	8	6	44.043	15	15
本文算法	27.589	6	3	43.303	12	9

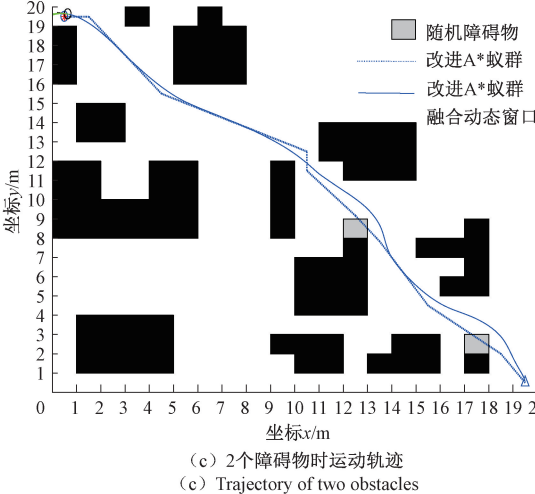
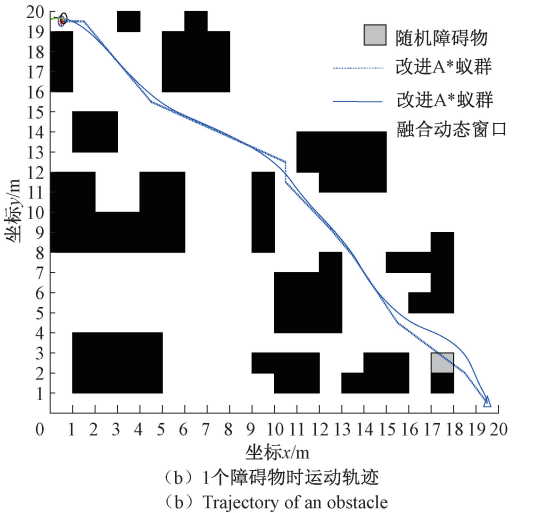
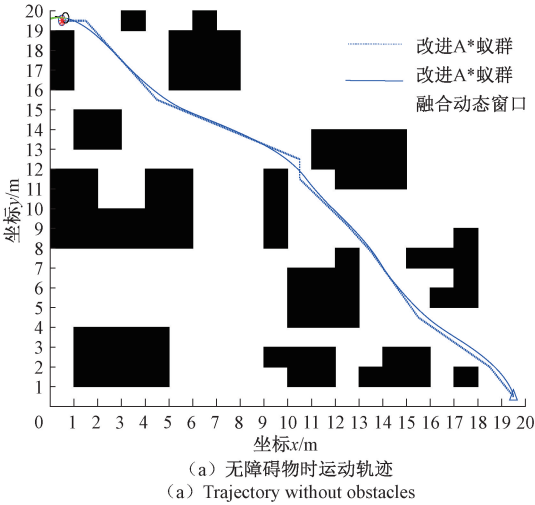
不难看出,两种环境下本文算法规划的路径平滑且规划路线略短。在20×20栅格环境下,本文算法和文献[20]改进算法及传统蚁群算法得到的最短路径长度分别为27.589、27.914、29.455,对应的迭代次数分别为3、6、10。在30×30栅格环境下,传统蚁群算法和文献[20]改进算法得到的最短路径长度分别为47.455、44.043,对应的迭代次数分别为25、15,而本文算法收敛到43.303仅仅需要迭代12次,本文算法比传统蚁群算法得到的路径长度缩短了8.75%、转折点数减少了59%。两种栅格环境下的路径规划仿真结果都说明了本文采用改进A*蚁群算法在确保求得最短路径较优的前提下,也大大提高了收敛速度。

4.2 融合算法的随机避障分析

设置4组仿真实验对融合算法进行验证,在环境相同的地图中采用改进A*蚁群算法、融合算法进行路径规划,设置三角形和圆形分别为起点栅格和终点栅格,在栅格地图中添加随机障碍物,评价函数各参数为: $\sigma = 0.1$, $\chi = 0.25$, $\psi = 0.2$, $\omega = 0.3$,仿真结果如图6所示。可以得出,不出现随机障碍物时两种算法都可以求出最优路径,但当环境中出现随机障碍物时,改进蚁群算法无法躲避障碍物,而融合算法可以实现障碍物的实时避障。

4.3 多动态障碍物环境中的避障实验

为了进一步验证融合算法在动态环境中避障的准确性,对机器人在20×20的栅格中进行实验,实验环境中设置3个动态障碍物和1个随机未知障碍物,其中白色圆点表示移动机器人,速度为1 m/s,光标为机器人探测器



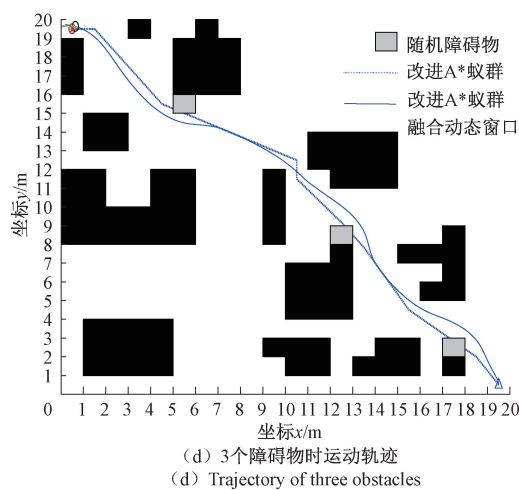


图 6 随机障碍路径规划轨迹图

Fig. 6 Random obstacle path planning trajectory

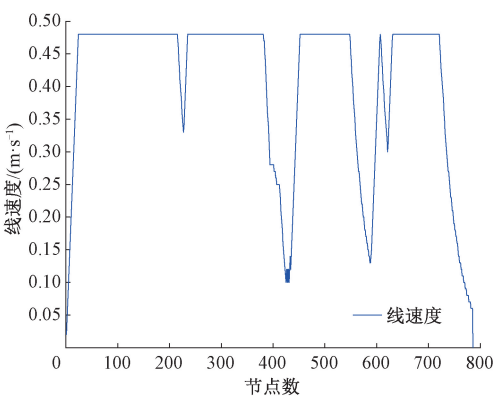
方向,矩形表示动态障碍物,该障碍物从起始位置按照规定的速度 0.6 m/s 匀速前往终点位置并停在原地,动态障碍物的起始坐标设置如表 3 所示,机器人的起点在 $[0.5, 14.5]$,终点在 $ANG19.5, 0.5]$,从图 7 可以看到,机器人从全局路径起始点出发,分别经历了与障碍物无碰撞、侧面碰撞、正面碰撞后,同样能准确快速地采取相应的避障策略找出平滑而安全的规划路径。图中虚线为改进 A* 蚁群算法规划的最优路径,实线为融合算法规划的实时路径,点状线为动态障碍物的移动轨迹。机器人控制参数反馈如图 7 所示。

表 3 动态障碍物参数

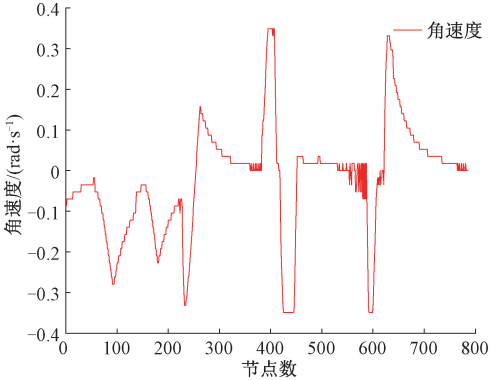
Table 3 Dynamic obstacle parameters

动态障碍物	起始坐标	终点坐标
D1	$[3.5, 13.5]$	$[3.5, 19.5]$
D2	$[14.5, 15.5]$	$[4.5, 15.5]$
D3	$[14.5, 0.5]$	$[6.5, 18.5]$
D4	$[15.5, 3.5]$	—

从图 8(a)可以看出,机器人在遇到动态障碍物前是按照改进 A* 蚁群算法规划出来的路线前进的,当检测到动态障碍物 D2 时,经计算机器人与 D2 的相对运动状态后预测会发生侧碰,于是机器人启动融合算法进行局部避障,成功规避了动态障碍物,之后机器人重新规划路线,朝着改进 A* 蚁群算法规划的最优路径靠近,如图 8(b2)所示。在图 8(a3)位置检测到动态障碍物 D3,并预测到与其发生正面碰撞,随即采用避障策略避开 D3,图 8(b3)为完成动态障碍物 D3 的避障。考虑到使障碍物停止不动对机器人造成干扰的作用,设置未知障碍物 D4,当检测到未知障碍物 D4 时,运用融合算法进行局部避障并实时规划靠近最优路径,避障成功结果如图 8(b4)所示。



(a) 线速度输出结果
(a) Linear speed output result



(b) 角速度输出结果
(b) Angular velocity output result

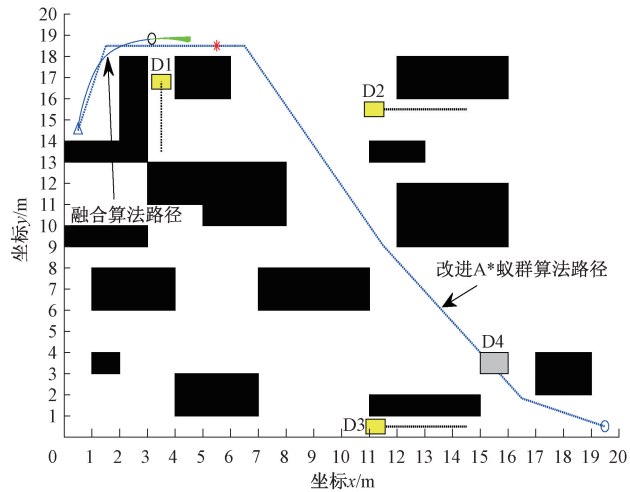
图 7 多动态障碍物机器人输出结果

Fig. 7 Output results of multi dynamic obstacle robot

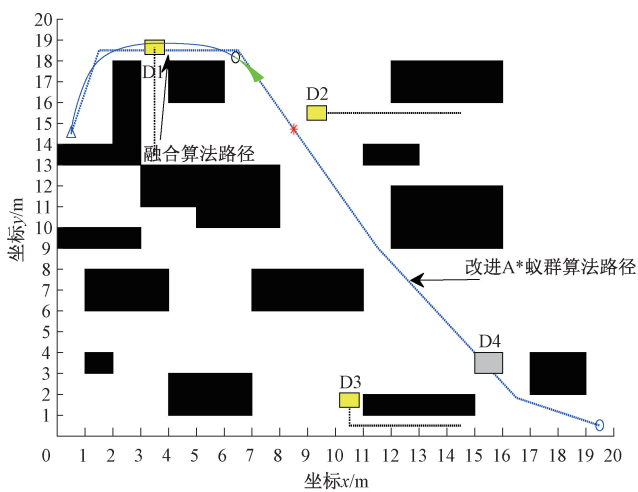
4. 4 实际环境规划实验

为了验证算法在实际环境中的适应性,采用图 9(a)实验平台 ROS 系统中 Gazebo 与 Rviz 工具,根据 Gmapping-SLAM 算法与激光雷达传感器获得的数据信息来完成对图 9(b)当前实验环境地图的构建,构建出如图 10(a)所示场地地图,确定机器人起始点后,调用改进 A* 蚁群算法,实现静态实验环境下机器人从起点到终点规划的全局路径,机器人按照规划好的路线向目标点前进,同时启动 Rviz 节点,利用 Rviz 工具进行可视化,观测到如图 10(b)所示规划的路径。

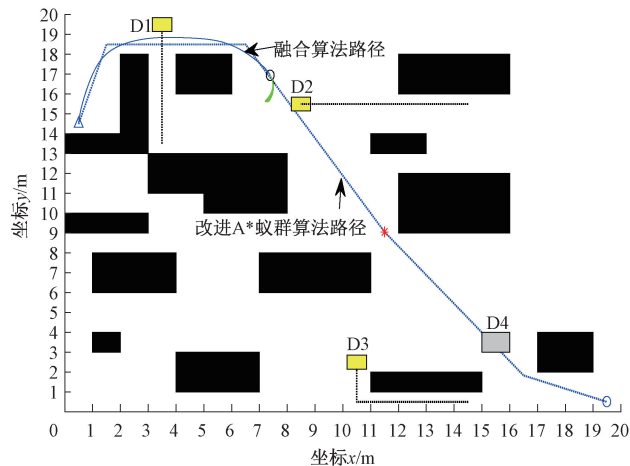
为进一步验证机器人在行进过程中是否能准确避开动态障碍物,在实验中选取一个自主搭建的机器人作为动态障碍物,如图 11 所示。在机器人规划好全局路径移动时,启动障碍物机器人,移动至机器人规划好的全局路径上,当动态障碍物进入到机器人雷达扫描的范围时,机器人会立即做出反应,运用融合算法进行局部避障,以此来规避动态障碍物,避障完成后,朝着改进 A* 蚁群算法规划的最优路径靠近。通过图 11 动态障碍物测试结果表明,本文融合算法可以使机器人在保证路径平滑的同时安全躲避地图中的动态障碍物。



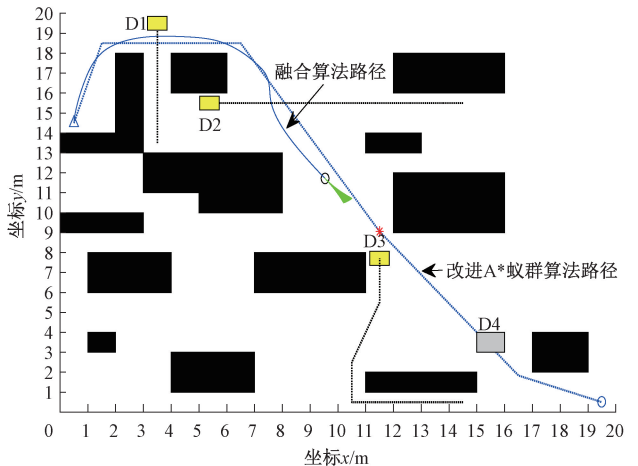
(a1) 正在避开D1
(a1) Avoiding D1



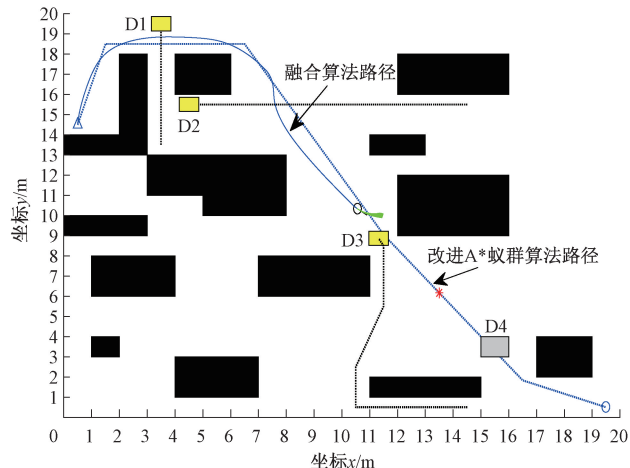
(b1) 完成D1的避障
(b1) Complete obstacle avoidance of D1



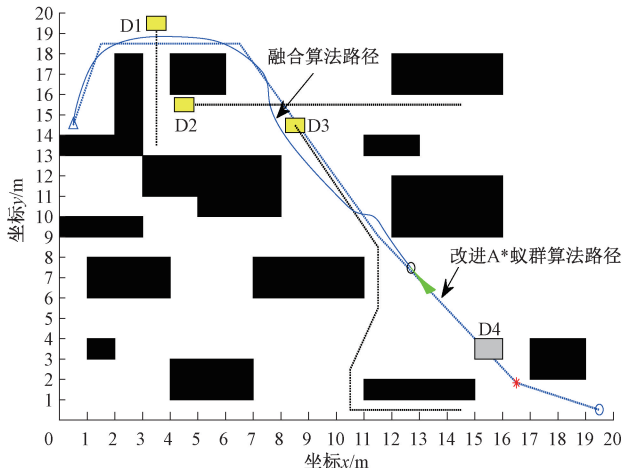
(a2) 正在避开D2
(a2) Avoiding D2



(b2) 完成D2的避障
(b2) Complete obstacle avoidance of D2



(a3) 正在避开D3
(a3) Avoiding D3



(b3) 完成D3的避障
(b3) Complete obstacle avoidance of D3

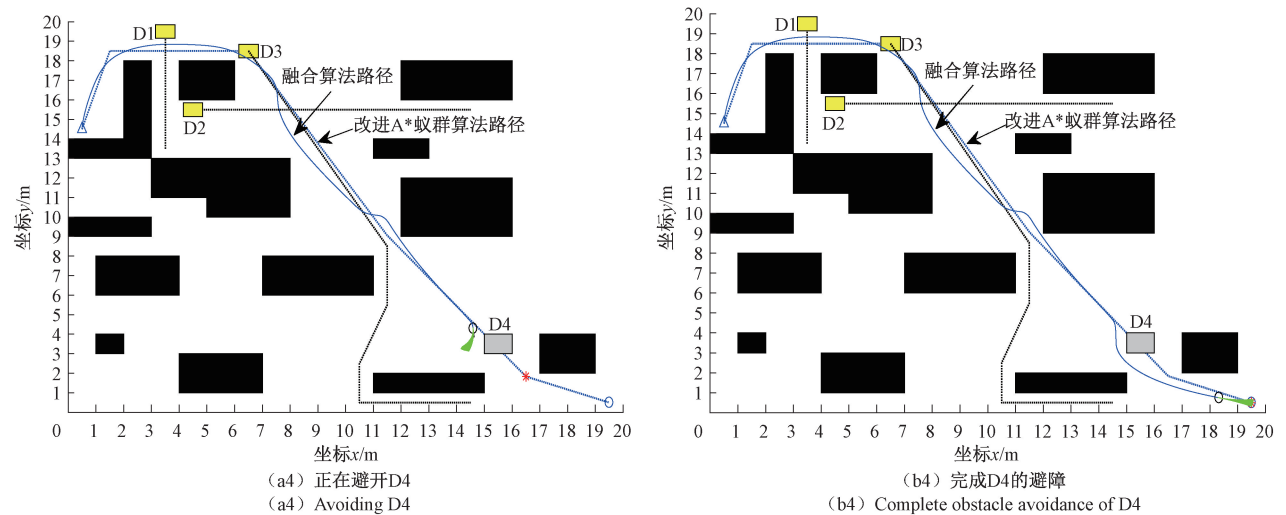


图 8 动态躲避障碍物过程
Fig. 8 Dynamic obstacle avoidance process

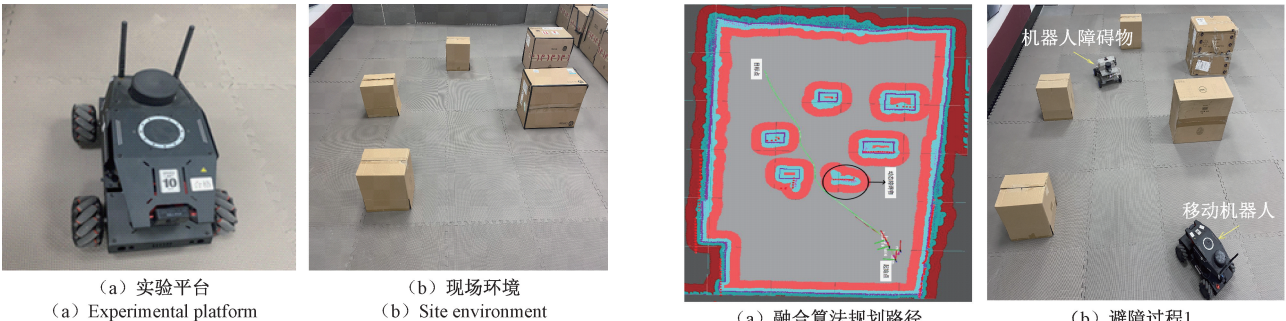


图 9 实验环境
Fig. 9 Experimental environment

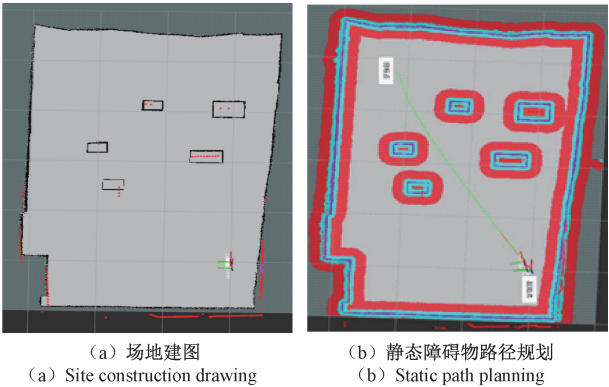


图 10 实验平台和场地建图
Fig. 10 Construction drawing of experimental platform and site

5 结 论

针对蚁群算法路径规划时效率低、规划的路径不平

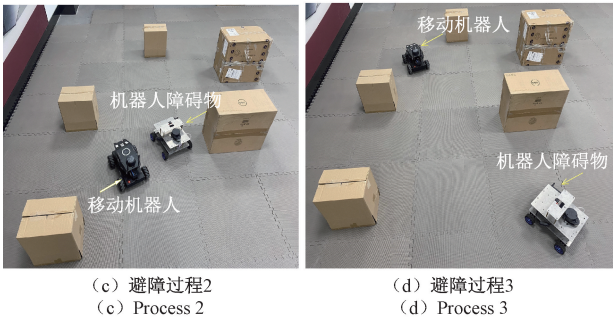


图 11 躲避动态障碍物测试
Fig. 11 Avoiding dynamic obstacles test

滑且无法实时避障以及 DWA 算法没有全局路径的指导易出现转圈现象等问题,本文对其不足进行了改进。首先,利用改进 A* 非均匀分配初始信息素,解决前期规划效率低的问题;其次,采用变步长搜索方式扩大蚂蚁搜索范围,提高搜索效率并缩短路径长度;再次,修改转移概率函数中的启发函数值并增加障碍物影响因子,解决死锁问题和加快收敛速度;然后,提出二次路径规划策略,对没必要的冗余节点进行删除;最后,通过改进 DWA 评

价函数使其有更好的局部避障功能。通过不同环境下的仿真实验表明了本文所提算法不仅能在静态障碍物下可行且在动态障碍物环境下也能及时避障规划出全局最优路径。

为了验证仿真实验结果的正确性,在实际环境中进行了实验,进一步验证了在真实动态环境下本文融合算法具有动态避障功能且运动轨迹贴合全局最优路径。

参考文献

- [1] 林韩熙,向丹,欧阳剑,等. 移动机器人路径规划算法的研究综述[J]. 计算机工程与应用,2021,57(18): 38-48.
- LIN H X, XIANG D, OUYANG J, et al. Overview of research on path planning algorithms for mobile robots [J]. Computer Engineering and Application, 2021, 57 (18): 38-48.
- [2] 张松灿,普杰信,司彦娜,等. 蚁群算法在移动机器人路径规划中的应用综述[J]. 计算机工程与应用,2020,56(8):10-19.
- ZHANG S C, PU J X, SI Y N, et al. Application of ant colony algorithm in mobile robot path planning [J]. Computer Engineering and Application, 2020, 56 (8): 10-19.
- [3] DAI X L, LONG S, ZHANG Z W, et al. Mobile robot path planning based on ant colony algorithm with A* heuristic method. [J]. Frontiers in Neurorobotics,2019, 13:15.
- [4] 迟旭,李花,费继友. 基于改进A*算法与动态窗口法融合的机器人随机避障方法研究[J]. 仪器仪表学报,2021,42(3):132-140.
- CHI X, LI H, FEI J Y. Research on robot random obstacle avoidance method based on the fusion of improved A* algorithm and dynamic window method [J]. Chinese Journal of Scientific Instrument, 2021, 42 (3): 132-140.
- [5] 江明,王飞,葛愿,等. 基于改进蚁群算法的移动机器人路径规划研究[J]. 仪器仪表学报,2019,40(2):113-121.
- JIANG M, WANG F, GE Y, et al. Research on mobile robot path planning based on improved ant colony algorithm [J]. Chinese Journal of Scientific Instrument, 2019,40 (2): 113-121.
- [6] 陈颖杰,高茂庭. 基于信息素初始分配和动态更新的蚁群算法[J]. 计算机工程与应用,2022,58(2):95-101.
- CHEN Y J, GAO M T. Ant colony algorithm based on pheromone initial allocation and dynamic update [J]. Computer Engineering and Applications, 2022, 58 (2): 95-101.
- [7] 槐创锋,郭龙,贾雪艳,等. 改进A*算法与动态窗口法的机器人动态路径规划[J]. 计算机工程与应用,2021,57(8):244-248.
- HUAI CH F, GUO L, JIA X Y, et al. Robot dynamic path planning based on improved A* algorithm and dynamic window method [J]. Computer Engineering and Application, 2021, 57 (8): 244-248.
- [8] 殷绍伟,彭力,戴菲菲. 融合改进A*蚁群和滚动窗口法的平滑路径规划[J]. 计算机科学与探索,2021,15(10):1969-1979.
- YIN SH W, PENG L, DAI F F. Smooth path planning combining improved A* ant colony and rolling window method [J]. Computer Science and Exploration, 2021, 15 (10): 1969-1979.
- [9] 黄辰,费继友,刘洋,等. 基于动态反馈A*蚁群算法的平滑路径规划方法[J]. 农业机械学报,2017,48(4):34-40,102.
- HUANG CH, FEI J Y, LIU Y, et al. Smooth path planning method based on dynamic feedback A* ant colony algorithm [J]. Journal of Agricultural Machinery, 2017,48 (4): 34-40, 102.
- [10] 刘新宇,谭力铭,杨春曦,等. 未知环境下的蚁群-聚类自适应动态路径规划[J]. 计算机科学与探索,2019,13(5):846-857.
- LIU X Y, TAN L M, YANG CH X, et al. Ant colony clustering adaptive dynamic path planning in unknown environment [J]. Computer Science and Exploration, 2019,13 (5): 846-857.
- [11] 任红格,胡鸿长,史涛. 基于改进蚁群算法的移动机器人动态路径规划[J]. 现代电子技术,2021,44(20):182-186.
- REN H G, HU H CH, SHI T. Improved ant colony algorithm for mobile robot path planning [J]. Modern Electronic Technology, 2021,44 (20): 182-186.
- [12] 彭育强,黄泽龙,李少伟. 基于动态窗口法的移动机器人自动避障导航研究[J]. 自动化仪表,2020,41(10):26-29,33.
- PENG Y Q, HUANG Z L, LI SH W. Research on automatic obstacle avoidance navigation of mobile robot based on dynamic window method [J]. Automation Instrument, 2020,41 (10): 26-29, 33.
- [13] 常新新,胡为,姬书得,等. 基于改进动态窗口法的移动机器人避障研究[J]. 组合机床与自动化加工技术,2021(7):33-36,39.
- CHANG X X, HU W, JI SH D, et al. Research on obstacle avoidance of mobile robot based on improved dynamic window method [J]. Modular Machine Tools and

- Automatic Processing Technology, 2021 (7): 33-36, 39.
- [14] MISSURA M, BENNEWITZ M. Predictive collision avoidance for the dynamic window approach [C]. International Conference on Robotics and Automation (ICRA), 2019: 8620-8626.
- [15] 程传奇, 郝向阳, 李建胜, 等. 融合改进 A* 算法和动态窗口法的全局动态路径规划[J]. 西安交通大学学报, 2017, 51(11): 137-143.
- CHENG CH Q, HAO X Y, LI J SH, et al. Global dynamic path planning combining improved A* algorithm and dynamic window method [J]. Journal of Xi'an Jiaotong University, 2017, 51 (11): 137-143.
- [16] 杨周, 刘海滨. 基于改进蚁群与动态窗口法的 AGV 动态路径规划[J]. 计算机工程与应用, 2022, 58(6): 287-295.
- YANG ZH, LIU H B. AGV dynamic path planning based on improved ant colony and dynamic window method [J]. Computer Engineering and Application, 2022, 58 (6): 287-295.
- [17] 马向华, 张谦. 改进蚁群算法在机器人路径规划上的研究[J]. 计算机工程与应用, 2021, 57(5): 210-215.
- MA X H, ZHANG Q. Research on improved ant colony algorithm in robot path planning [J]. Computer Engineering and Application, 2021, 57 (5): 210-215.
- [18] 刘建娟, 薛礼敏, 张会娟, 等. 融合改进 A* 与 DWA 算法的机器人动态路径规划[J]. 计算机工程与应用, 2021, 57(15): 73-81.
- LIU J J, XUE L Q, ZHANG H J, et al. Robot dynamic path planning based on improved A* and DWA algorithm [J]. Computer Engineering and Application, 2021, 57 (15): 73-81.
- [19] 王彬, 聂建军, 李海洋, 等. 优化 A* 与动态窗口法的移动机器人路径规划[J/OL]. 计算机集成制造系统: 1-17 [2022-12-01].
- WANG B, NIE J J, LI H Y, et al. Mobile robot path planning based on optimization A* and dynamic window method [J/OL]. Computer Integrated Manufacturing System: 1-17 [2022-12-01].
- [20] 刘双双, 黄宜庆. 多策略蚁群算法在机器人路径规划中的应用[J]. 计算机工程与应用, 2022, 58(6): 278-286.
- LIU SH SH, HUANG Y Q. Application of multi strategy ant colony algorithm in robot path planning [J]. Computer Engineering and Application, 2022, 58 (6): 278-286.

作者简介



赵倩楠, 现为安徽工程大学研究生, 主要研究方向为移动机器人路径规划、编队控制等。

E-mail: 1347082086@qq.com

Zhao Qiannan is now a M. Sc. candidate at Anhui Polytechnic University. Her main research interests include mobile robot path planning and formation control.



黄宜庆 (通信作者), 现为安徽工程大学教授, 主要研究方向为机器人与飞行器控制、移动机器人与路径规划。

E-mail: yiqhuang@ahpu.edu.cn

Huang Yiqing (Corresponding author) is now a professor of Anhui Polytechnic University. His main research interests include robot and aircraft control, mobile robot and path planning.