

DOI:10.19651/j.cnki.emt.2415299

改进 A* 算法的移动机器人全局路径规划*

熊勇刚 李波 姚焱 付茂林 李城炫

(湖南工业大学机械工程学院 株洲 412007)

摘要: 针对 A* 算法在移动机器人路径规划存在搜索效率低, 路径斜穿障碍物顶点, 路径拐弯多等问题。提出一种改进的 A* 算法, 首先在 A* 算法的邻域扩展中采用避免斜穿障碍物顶点的策略; 再引入障碍物因素对评价函数进行指数加权, 减少不必要的搜索, 提高 A* 算法的效率和灵活性, 使算法偏向于选择障碍物较少的路径; 最后使用三次优化折线的策略, 加入障碍物安全距离, 减少路径上的冗余节点和拐弯。使用 MATLAB 进行实验仿真, 结果表明, 在 20 m×20 m、40 m×40 m、60 m×60 m 栅格地图环境下, 改进 A* 算法较传统 A* 算法, 搜索时间分别减少 70.12%、84.31%、91.44%, 扩展节点分别减少 53.77%、71.20%、74.30%, 路径累计拐弯角度分别减少 70.48%、76.31%、82.18%, 改进 A* 算法能够有效的提高移动机器人路径规划的效率, 路径更为平滑和安全, 且在复杂环境中优势更为明显。

关键词: A* 算法; 评价函数; 领域扩展; 安全距离; 路径规划

中图分类号: TP18 **文献标识码:** A **国家标准学科分类代码:** 520.2099

Global path planning of mobile robot with improved A* algorithm

Xiong Yonggang Li Bo Yao Tao Fu Maolin Li Chengxuan

(College of Mechanical Engineering, Hunan University of technology, Zhuzhou 412007, China)

Abstract: An improved A* algorithm is proposed to address the issues of low search efficiency, path diagonally crossing obstacle vertices, and excessive turns in mobile robot path planning. Firstly, a strategy is introduced to avoid diagonally crossing obstacle vertices during the neighborhood expansion in the A* algorithm. Secondly, an exponential weight is applied to the evaluation function based on obstacle factors to reduce unnecessary search and improve the efficiency and adaptability of the A* algorithm, favoring paths with fewer obstacles. Finally, a three-phase optimization strategy is employed, considering the obstacle safety distance, to minimize redundant nodes and turns in the path. MATLAB simulations are conducted in grid maps of sizes 20×20 m, 40×40 m, and 60×60 m. The results demonstrate that the improved A* algorithm significantly reduces search time by 70.12%, 84.31%, and 91.44%, respectively, and reduces the number of expanded nodes by 53.77%, 71.20%, and 74.30%, respectively. Moreover, the accumulated turning angles in the path are reduced by 70.48%, 76.31%, and 82.18%, respectively. The improved A* algorithm effectively enhances the efficiency of mobile robot path planning, resulting in smoother and safer paths, especially in complex environments.

Keywords: A* algorithm; evaluation function; field expansion; safe distance; path planning

0 引言

移动机器人广泛应用于工业、医疗、服务、军事等领域。路径规划是移动机器人独立完成给定任务的一个关键环节^[1]。根据是否预知环境, 将路径规划分为全局路径规划和局部路径规划^[2-3], 全局路径规划需要完全已知的工作环境, 如 Dijkstra^[4]、A*^[5-6]、D* 算法等^[7]; 局部路径规划依靠传感器、摄像头实时获取地图信息并规划路径, 如动态窗口

法(dynamic window approaches, DWA)^[8]、快速搜索模型树(rapidly-exploring random trees, RRT)、人工势场法(artificial potential field, APF)、粒子群算法^[9]、蚁群算法等^[10], A* 算法与这些算法相比具有原理简单、效率高, 能找到最优解、适用性广等优点^[11]。为了进一步提高 A* 的性能, 许多学者对传统的 A* 算法进行改进, 改进的方面集中在 A* 算法的搜索策略、领域扩展、评价函数、路径平滑

收稿日期: 2024-01-08

* 基金项目: 湖南省自然科学基金(2022JJ50078)、教育部创新基金(2021JQR026)项目资助

处理等。Rongshen 等^[12]提出将 8 邻域搜索改为带有方向的 5 邻域搜索,减少了不必要方向的搜索,但没有考虑到搜索会陷入死点的问题。张庆等^[13]采用选取跳点搜索策略而舍去大量不必要的邻节点,大大提高搜索效率,但不能保证能找到最优解。唐嘉宁等^[14]提出双向搜索的策略,交替搜索路径,加快搜索效率,但需要维护两个搜索队列,增加了搜索的复杂性和难度。Karlijn 等^[15]在评价函数中加入转弯成本的因子。Rustu 等^[16]使用一个策略池,多方案的选择一个更好的搜索策略,极大的改善了搜索的路径。汪首坤等^[17]提出变步长分段搜索法,解决了搜索死循环、搜索量大等问题,但步长选择困难且无法适用复杂环境。袁千贺^[18]通过量化环境信息对评价函数进行加权加快算法的搜索效率,但权重调节不够灵活。段会龙等^[19]提出一种基于 Bresenham 直线算法思想,删除路径上的中间节点,使路径更为平滑。以上算法运用各种算法提高了搜索效率,但很少考虑环境中障碍物对路径的影响,且规划的路径安全性和平滑性不高。

本文提出了一种改进 A* 算法的全局路径规划的新方法。根据障碍物在 A* 算法扩展邻域中的位置,将会产生路线斜穿障碍物顶点的节点舍去,解决了路径斜穿障碍物的问题;对传统 A* 算法的评价函数进行改进,引入障碍物指数因子进行动态加权,提高算法的搜索效率,并考虑障碍物对路径的影响,鼓励路线避开密集障碍物区域;最后引入障碍物安全距离,对规划后的路径进行三次折线优化。将改进后的 A* 算法在多种地图上进行路径规划仿真,对比传统 A* 算法和其他改进的 A* 算法,验证本文算法的优越性。

1 传统 A* 算法

1.1 地图建模

环境地图直接影响移动机器人的路径规划的结果,常用的地图建模方法有栅格地图,图结构地图,拓扑地图,点云地图。因为 A* 需要对地图进行遍历和搜索,栅格地图将环境划分为离散的网格,在 A* 算法的搜索过程中,可以通过简单的相邻关系和网格索引进行快速地计算节点之间的代价和路径,并找到最短路径,所以本文采用栅格地图。如图 1 所示,通过在网格中标记障碍物和无障碍空间来表示地图;黑色代表障碍物,白色为可通行的自由区域。

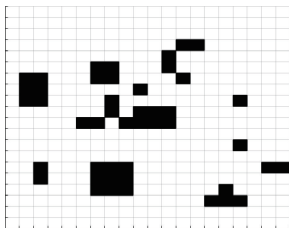


图 1 栅格地图

1.2 A* 算法

A* 算法是一种启发式搜索算法,通过评价函数来确定

搜索的方向,评价函数由当前节点到起点的实际距离代价和当前节点到终点的预估距离代价组成;在搜索过程,从起始点开始向四周扩展相邻节点,根据评价函数 $F(n)$ 来计算每个拓展节点的代价,选择评价价值最小的节点作为下一个需要拓展的节点,直到搜索到终点就完成路径的搜索。

传统 A* 算法的评价函数为:

$$F(n) = G(n) + H(n) \quad (1)$$

$G(n)$ 为实际代价, $H(n)$ 为预估代价,两者的权重系数相同。预估代价的计算方法常用的有欧式距离、曼哈顿距离、切比雪夫距离、对角线距离。本文选取欧式距离作为 $H(n)$ 代价函数,其计算公式如下:

$$H(n) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2)$$

式中: $(x_1, y_1), (x_2, y_2)$ 为当前扩展节点的坐标和目标节点的坐标。

2 A* 算法的改进

2.1 拓展领域的改进

传统的 A* 算法向四周扩展 8 个节点,如图 2(a) 所示。当搜索到障碍物则直接跳过该节点的计算,其他方向正常通行,这样会出现路线斜穿障碍物顶点的情况,导致通行有碰撞障碍物的风险。本文提出舍弃产生斜穿障碍物顶点的搜索,来避免碰撞。首先确定当前节点的垂直方向或者水平方向是否有相邻的障碍物,如果有就舍弃障碍物左右相邻或者上下相邻的节点的搜索,分别为图 2(b) 和 (c) 所示,具体的对应关系如表 1 所示

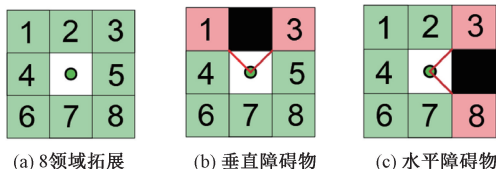


图 2 领域障碍物位置图

表 1 扩展字节点选择

障碍物节点	保留节点	舍去节点
2	4、5、6、7、8	1、2、3
4	2、3、5、7、8	1、4、6
5	1、2、4、6、7	3、5、8
7	1、2、3、4、5	6、7、8

2.2 评价函数的优化

传统的 A* 算法通过向周围方向的随机遍历大量的节点来搜索路径,会来回搜索许多不必要的节点,导致计算量十分大,搜索缓慢。A* 算法的核心为评价函数,而影响搜索速度和精度最主要的因素是启发函数中当前点到终点的估计值 $H(n)$ 和实际需要路径值 $H(n)^*$ 的大小关系;如果 $H(n) > H(n)^*$, 则搜索速度很快,但不能保证找到最短路径;当 $H(n) = H(n)^*$, 这种搜索最有效,搜索速度较快,也

能保证找到一条最短路径;当 $H(n) < H(n)^*$, 搜索速度变慢, 保证能找到一条最短的路径, 即在一定的范围内, 预估代价值 $H(n)$ 越大, 搜索速度越快; 预估代价值越靠近实际需要路径值 $H(n)^*$, 搜索越精确, 所以选择一个合适大小的预估代价值是 A* 算法的关键。又因为传统 A* 算法只考虑了距离成本, 没有考虑到障碍物对路径的影响, 能搜索到最短路径而不是较短且安全的路线。

综上所述, 本文提出引入障碍物指数因子对启发函数的预估代价值进行动态加权, 当前节点到目标点区域内障碍物较多时, 预估代价值增大, 搜索速度变快, 算法偏向于搜索障碍物较少的路径。当区域障碍物较少时, 算法会偏向于搜索最优路径而减慢搜索速度。这样算法在提高搜索效率的同时, 鼓励算法更偏向于经过障碍物较少的路径, 增加路径的安全性。改进后的启发函数如下:

$$F(n) = G(n) + e^{P(o)} \cdot H(n) \quad (3)$$

式中: $P(o)$ 为当前点到终点区域障碍物所占面积的比。

$$P(o) = \frac{n \cdot S}{dx \cdot dy} \quad (4)$$

式中: dx 为当前扩展节点到目标点的垂直位移, dy 为当前扩展节点到目标点的水平位移, n 为 dx 和 dy 组成的长方形区域内障碍物网格的数量, S 为单个障碍物的面积。

2.3 改善规划的路径

传统的 A* 算法规划的路径冗余节点多, 拐点多。应用于现实移动机器人工作场景, 反复的转向会增加路径的复杂度, 降低通行效率, 在速度比较大时连续转弯还会引起机器人的侧翻, 为了满足实际应用要求, 对机器人的路径轨迹进行优化处理, 去除多余节点, 减少拐弯次数, 从而提高机器人运动的效率、平稳性和安全性, 本文采用三次折线优化路径, 优化过程如图 3 所示, 具体优化步骤如下:

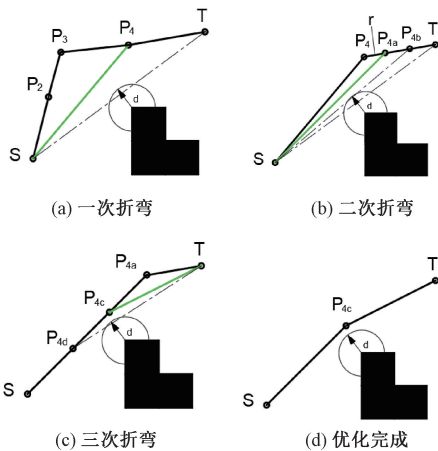


图 3 优化路径过程

1) 第 1 次优化, 设立安全距离 d , 提取关键节点, 去除冗余节点。如图 3(a) 所示, S 为起点, T 为终点, P 为中间节点, 障碍物为黑色方块, 设置机器人行驶路径与障碍物顶点的距离为 d , 低于这个距离将会有碰撞的风险。将

起点作为当前节点, 连接不相邻的两个节点, 如果连接的路径到障碍物顶点的距离大于安全距离, 则中间节点为冗余节点, 如 P_2, P_3 为冗余节点; 如果距离小于安全距离时, 将中间节点标记为关键点 (P_4), 并将该点作为当前节点, 重复上面的操作, 直到连接到终点。连接起点、关键点、终点得到优化后的路线如图优化后的路线为 $S-P_4-T$

2) 第 2 次优化, 如图 3(b) 所示, 以第 1 次优化的路径为基础, 从起点开始, 将起点作为当前节点, 在每两个拐点之间取步长为 r 的分节点, 如 P_{4a}, P_{4b} , 按顺序连接当前节点和分节点, 判断是否有碰撞障碍物的风险, 如果有, 则舍弃该节点。如果没有, 则保留该节点作为新路径的节点, 如 P_{4a} 。将该点作为当前节点, 将下两个拐点进行分节点处理, 再连接当前节点和分节点, 如此往复, 直到到达终点, 则新的路径为 $S-P_{4a}-T$ 。

3) 第 3 次优化, 以第 2 次优化的路径为基础, 从终点开始, 将终点前的两个节点进行分节点处理, 逆序连接分节点, 按照第 2 次优化的方法, 逆序优化路径, 如图 3 所示, 优化后的路径为 $T-P_{4c}-S$ 则改进后的算法流程图如图 4 所示, 黄色框为本文改进的部分。

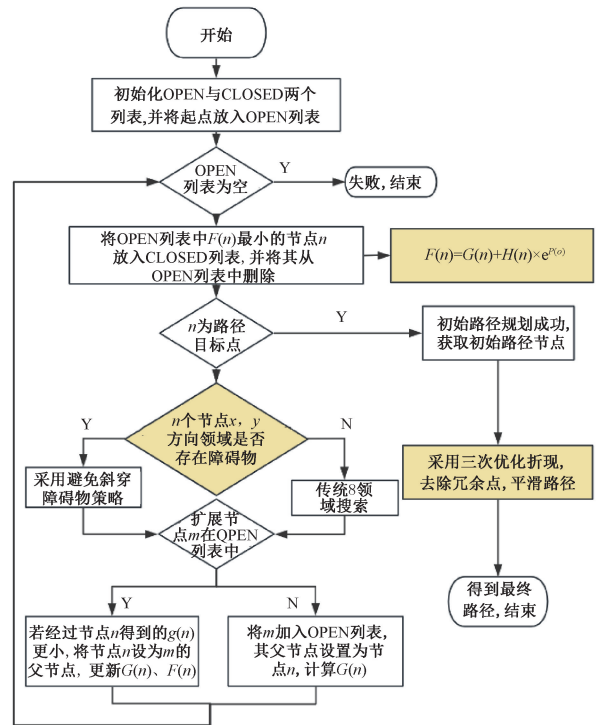


图 4 改进算法的流程图

3 仿真分析

为了验证本文算法的有效性和性能, 采用仿真软件 Matlab 2022a, 在 win10 操作系统, i5-8400 处理器和 16 g 内存环境下进行。

在不同环境地图下进行仿真, 对比传统 A* 算法、改进 A* 算法、其他改进算法所规划路径, 在路径长短, 搜索时

间、扩展节点数、转弯角度等方面的差别,并且考虑机器人实际的工作环境与运动规律,来分析改进 A* 算法优越性和还存在问题。

为了验证去除产生路径斜穿障碍物扩展节点的有效性,构建一个 20 m×20 m 的地图,每个栅格为边长为 1 m 的正方形,白色为可通行区域,黑色为障碍物,起始点为蓝色三角形,目标点为蓝色圆圈;起点位置坐标为(10.5, 16.5),终点为(1.5, 1.5);用传统 A* 算法和采用避免斜穿障碍物顶点策略的 A* 算法进行仿真对比,如图 5 所示,改进后的路线能避免穿过障碍物顶点,从而不会引起机器人横穿时引发的碰撞,增加了路径的安全性。

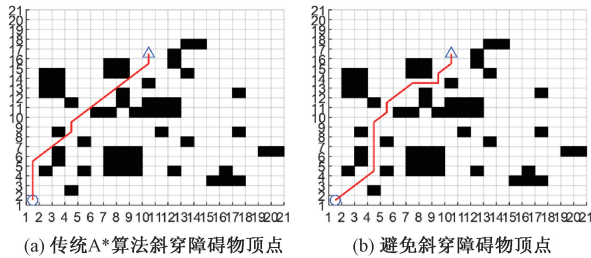


图 5 解决斜穿障碍物的问题

为了验证改进评价函数的有效性,起始坐标设为(17.5, 18.5),终点坐标为(1.5, 1.5),将传统评价函数的 A* 算法和改进评价函数的 A* 算规划的路径进行仿真对比,如图 6 所示。

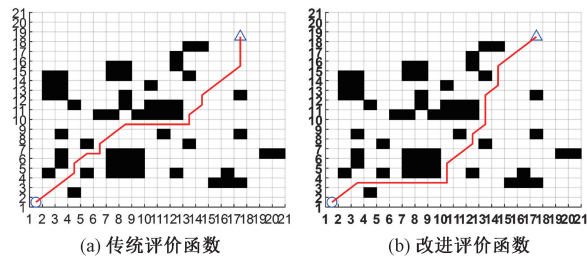


图 6 评价函数改进前后路径

从图 6 可知,传统评价函数规划的路线附近障碍物多,且多次穿过密集障碍物区域,而改进评价函数的 A* 算法规划的路径倾向于绕开密集障碍物,选择一条碰撞风险较低的路线,大大增强了路径的安全性。表 2 为改进前后的仿真结果,搜索时间为 20 组重复实验所得的平均值。由表 2 所示,改进评价函数的 A* 算法搜索时间减少 70.12%,拓展节点数减少了 53.77%,大大减少了计算量,提高了算法的搜索效率,并且路径长度只增加了 2.15%,没有对路径长度产生很大的影响。

表 2 仿真对比结果

A* 算法	路径长度/m	搜索时间/s	扩展节点
传统评价函数	27.142 1	0.010 157	212
改进评价函数	27.727 9	0.003 034	98

为了验证三次优化曲线的有效性,设置障碍物安全距离 $d=0.3\text{ m}$,分节点步长 $r=1\text{ m}$;将改进 A* 算法进行三次路径优化,对比原始路径和 3 次优化后的路径,如图 7 所示,可以明显的发现第 1 次优化减少了大量的冗余路径节点和拐弯数,第 2 次第 3 次进一步优化路径,路线得到了充分的优化。

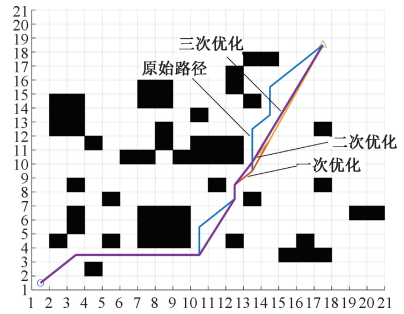


图 7 优化路线图

为了验证在复杂环境环境地图下,改进 A* 算法的有效性和优越性,建立 20 m×20 m、40 m×40 m、60 m×60 m 大小的栅格地图,设置起点分别为(17.5, 18.5)、(37.5, 40.5)、(59.5, 59.5)。终点都为(1.5, 1.5),对比传统 A* 算法、文献[18]算法、本文改进 A* 算法在不同地图下的仿真实验及数据,仿真路线图如图 8、9、10 所示。相应的数据为表 3 和图 11 所示。

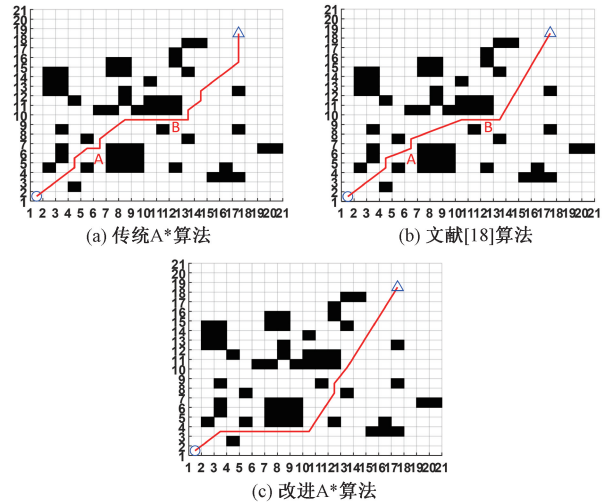


图 8 20×20 地图仿真

文献[18]算法通过当前点到终点的障碍物栅格数与起点到终点障碍物栅格数的比值对评价函数的预估代价进行动态加权;由表 3 数据得到,对比传统的 A* 算法,在路径长度,路径拐弯角度,搜索效率都有很大的提升。但算法的动态权重的调节不够灵活和准确,算法也没有鼓励绕过密集障碍物区域,相比文献[18]算法,本文算法进一步减少了不必要的扩展节点、搜索时间。虽然路径节点略有增加,但路径累计拐弯角度减少较为明显,路径更为平滑,更适合机器

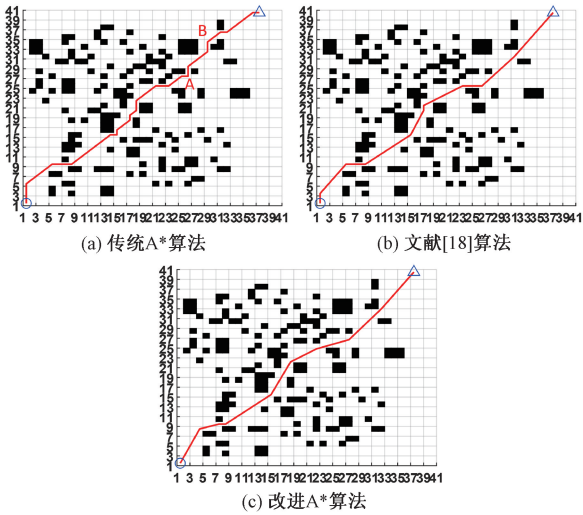


图9 40×40 地图仿真

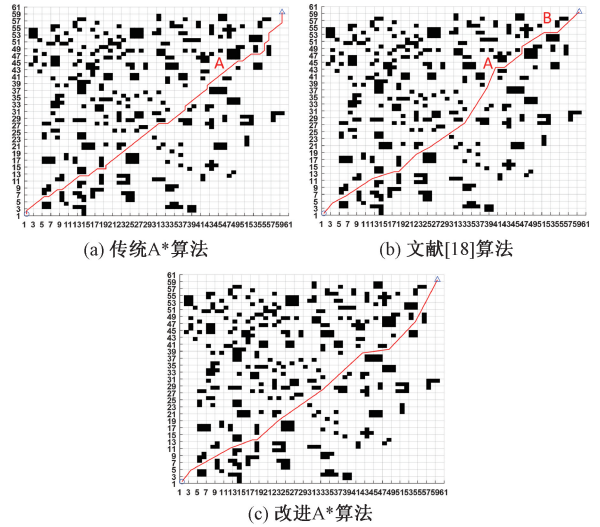


图10 40×40 地图仿真

表3 仿真实验结果

算法	地图大小/m ²	路径长度/m	搜索时间/s	扩展节点	路径节点	路径累计角度/(°)
传统A*算法	20×20	27.142 1	0.010 157	212	24	585
文献[18]算法		25.799 7	0.004 223	117	8	327.907 4
改进A*算法		26.489 8	0.003 034	98	11	172.685 3
传统A*算法	40×40	59.183 8	0.068 212	757	49	900
文献[18]算法		56.653 8	0.038 157	431	11	326.309 9
改进A*算法		55.509 5	0.010 700	218	16	213.711 3
传统A*算法	60×60	88.468 0	0.171 249	1 206	70	1 170
文献[18]算法		87.420 5	0.111 982	795	17	477.165 7
改进A*算法		85.198 1	0.014 651	310	28	220.007 6

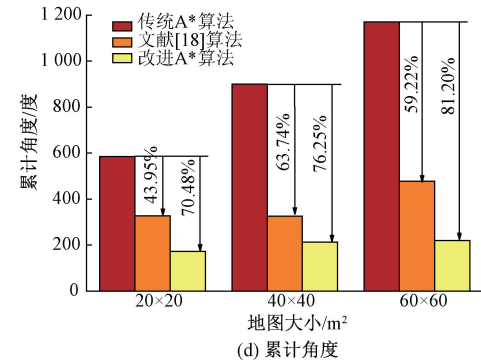
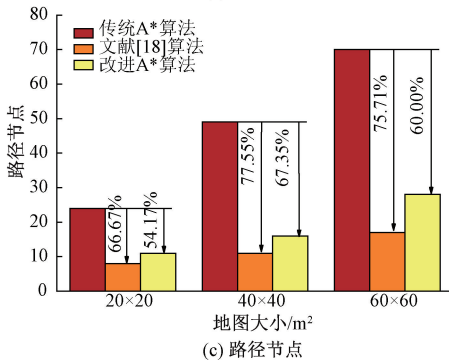
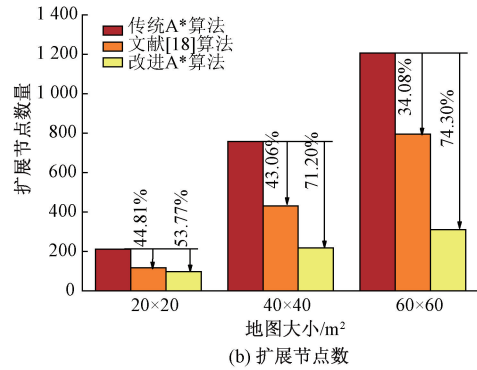
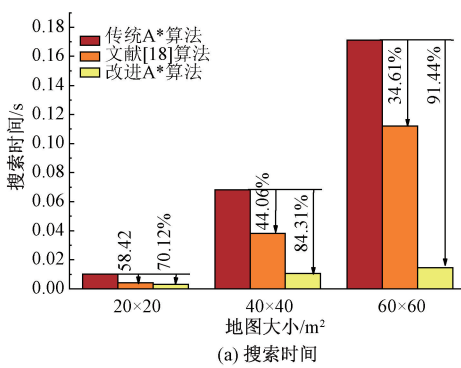


图11 算法性能图

人的实际运动规律,且从图 7~9 中标记的 A、B 位置可以明显的发现,本文算法规划的路径能一定程度上避开密集障碍物区域和狭窄通道,安全性更高。

由表 3 可以得到,对比传统 A* 算法,在相同障碍物环境下,改进 A* 算法引入障碍物动态权重因子,减少了许多不必要的扩展,意味着计算量减少,搜索时间减少,搜索效率提高;由于障碍物因子对预估代价的加权,使算法更偏向选择远离密集障碍物的路径。经过三次优化路径后,使路径更为平滑和安全。

由表 3 和图 10 所示,在环境尺寸为 20×20 时,改进的 A* 算法在搜索时间、扩展节点数、路径节点、路径累计角度有很明显的减少,分别减少了 70.12%、53.77%、54.17%、70.48%。路径长度也减少了 2.4%。在 40×40 的环境尺寸下,搜索时间、扩展节点数、路径节点、路径累计角度分别减少了 84.31%、71.20%、67.35%、76.25%。路径长度减少 6.2%。在 60×60 的环境尺寸下,搜索时间、扩展节点数、路径节点数、路径累计角度分别减少了 91.44%、74.30%、60.00%、81.20%。路径长度减少了 3.7%。综上所述,改进后的 A* 算法在不影响路径长度的基础上,提高了 A* 算法的搜索效率,减少搜索时间,并且增加了路径的平滑度和安全性,随着地图尺寸的增大和环境复杂度的增加,改进 A* 算法规划的路径的优越性更为明显。

4 结 论

为了解决传统 A* 算法计算量大,计算时间长,冗余节点多,路径拐弯多等问题,本文对传统的 A* 算法进行了改进,通过对搜索领域的取舍,避免了路径斜穿障碍物顶点;在评价函数的预估代价中引入障碍物信息,实现评价函数的自适应调整;提出一种三次优化路径方案,减少路径长度和折弯角度,提高了路径的安全性;通过实验证明,改进 A* 算法的搜索效率得到提高,并且规划的路径更平滑,安全性更高。针对移动机器人的路径规划问题,未来的研究方向将会在以下展开:融合其他路径规划算法来进一步提高算法性能和解决动态避障的问题。实现复杂环境中多机器人协同,多目标任务的路径规划。将算法运用到真实的移动机器人工作场景,进一步验证其有效性。

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.
- [2] 王梓强, 胡晓光, 李晓筱, 等. 移动机器人全局路径规划算法综述[J]. 计算机科学, 2021, 48(10): 19-29.
- [3] 皇甫淑云, 唐守锋, 童紫原, 等. 自主移动机器人路径规划方法研究综述[J]. 软件导刊, 2018, 17(10): 1-5.
- [4] OKAN B, JINNIE S. Learning analytics and computerized formative assessments: An application

- of dijkstra's shortest path algorithm for personalized test scheduling[J]. Mathematics, 2022, 10(13): 2230.
- [5] THEVIYANTHAN K. Computing the runs that should be scored every over when chasing a target in limited-overs cricket using the A* algorithm [J]. Applied Artificial Intelligence, 2021, 35(15): 87-101.
- [6] ZHANG L, ZHANG Y, ZENG M, et al. Robot navigation based on improved A* algorithm in dynamic environment [J]. Assembly Automation, 2021, 41(4): 419-430.
- [7] 霍凤财, 迟金, 黄梓健, 等. 移动机器人路径规划算法综述[J]. 吉林大学学报(信息科学版), 2018, 36(6): 639-47.
- [8] YOUSEF K, MASUD M, EDUARDO V, et al. An online unsupervised dynamic window method to track repeating patterns from sensor data [J]. IEEE transactions on cybernetics, 2020.
- [9] 梁旭, 刘才慧. 基于混合粒子群算法的在线检测路径规划[J]. 国外电子测量技术, 2015, 34(12): 30-4.
- [10] RICARDO S J, J. P C, ALFONSO G. Path planning for autonomous mobile robots: A review[J]. Sensors, 2021, 21(23): 7898-.
- [11] DAN X, HANXI L, JIAN O, et al. Combined improved A* and greedy algorithm for path planning of multi-objective mobile robot[J]. Scientific Reports, 2022, 12(1): 13273-.
- [12] RONGSHEN L, ZHIYONG W, XIANGUI L, et al. Fusion algorithm of the improved A* algorithm and segmented bézier curves for the path planning of mobile robots[J]. Sustainability, 2023, 15(3): 2483.
- [13] 张庆, 刘旭, 彭力, 等. 融合 JPS 和改进 A* 算法的移动机器人路径规划[J]. 计算机科学与探索, 2021, 15(11): 2233-40.
- [14] 唐嘉宁, 彭志祥, 李孟霜, 等. 基于改进 A* 算法的无人路径规划研究[J]. 电子测量技术, 2023, 46(8): 99-104.
- [15] KARLIJN F, JOOST V E. Efficient path planning for automated guided vehicles using A* (Astar) algorithm incorporating turning costs in search heuristic [J]. International Journal of Production Research, 2023, 61(3): 707-725.
- [16] RUSTU A, YUSUF Y M. Multi-strategy and self-adaptive differential sine-cosine algorithm for multi-robot path planning [J]. Expert Systems With Applications, 2023, 232.
- [17] 汪首坤, 邸智, 王军政, 等. 基于 A* 改进算法的机械臂避障路径规划[J]. 北京理工大学学报, 2011, 31(11): 1302-6.
- [18] 袁千贺, 魏国亮, 田昕, 等. 改进 A* 和 DWA 融合的移动机器人导航算法[J]. 小型微型计算机系统, 2023, 44(2): 334-9.
- [19] 段会龙, 武炎明, 刘金国. 改进型 A* 算法的可重构机器人路径规划研究[J]. 电子测量技术, 2023, 46(6): 44-50.

作者简介

熊勇刚, 博士, 教授, 研究生导师, 主要研究方向为车辆工程和智能机器人等。

E-mail: xygyxj@163.com

李波, 硕士, 主要研究方向为无人车的路径规划。

E-mail: 1658239394@qq.com