

DOI:10.3969/j.issn.1003-5060.2023.11.002

基于无人叉车行驶时间优化的改进 A* 算法

童亚男, 肖本贤

(合肥工业大学 电气与自动化工程学院, 安徽 合肥 230009)

摘要:传统 A* 算法规划所得路径拐点个数多、转弯角度大,导致无人叉车在行驶过程中需要频繁进行制动、转向等操作,增加了行驶时间和危险系数。为解决上述问题,文章以无人叉车为研究对象,采用 8 向栅格地图对环境建模,提出一种基于行驶时间优化的 A* 算法。通过引入转弯行驶时间及转弯代价补偿参数优化实际代价函数,以减少路径中的拐点;向启发函数引入父节点启发函数,同时对启发函数的权重进行自适应调整,使搜索方向在更快地向目标节点靠近的同时保证路径最优。仿真实验结果表明:改进 A* 算法有效且实用;相较于传统 A* 算法,改进 A* 算法在 30×30 栅格地图中拐点个数减少 40.00%,转弯总角度减少 60.00%;在 50×50 栅格地图中拐点个数减少 47.10%,转弯总角度减少 65.90%。

关键词:路径规划;A* 算法;无人叉车;栅格地图;路径优化

中图分类号:TP242.6

文献标志码:A

文章编号:1003-5060(2023)11-1447-08

Improved A* algorithm based on the optimization of driving time of unmanned forklifts

TONG Yanan, XIAO Benxian

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

Abstract:The traditional A* algorithm plans a path with many inflection points and large turning angles, which leads to frequent braking and steering operations of unmanned forklifts during the driving process, increasing the driving time and danger factor. To solve the above problems, this paper takes unmanned forklift as the research object, adopts eight-way grid map to model the environment, and proposes an A* algorithm based on driving time optimization. The actual cost function is optimized by introducing a turning driving time function and turning cost compensation parameter to reduce the inflection points; the parent node heuristic function is introduced to the heuristic function, while the weights of the heuristic function are adaptively adjusted to make the search direction approach the target node faster while ensuring the optimal path. The effectiveness and practicability of the improved A* algorithm are verified by the simulation test. Comparing the simulation results of the improved A* algorithm and the traditional A* algorithm, the number of inflection points is reduced by 40.00% and the total turning angle is reduced by 60.00% in the 30×30 grid map; the number of inflection points is reduced by 47.10% and the total turning angle is reduced by 65.90% in the 50×50 grid map.

Key words:path planning; A* algorithm; unmanned forklift; grid map; path optimization

当下世界范围内蓬勃发展的电子商务带来的
日益增长的仓储物流需求与手动操纵工业车辆的

熟练技术人员的不足之间的矛盾日益凸显,人工
手动操作显然不足以应付物流高峰期的生产和交

收稿日期:2022-04-02;修回日期:2022-04-21

基金项目:国家重点研发计划资助项目(2016YFF0102200)

作者简介:童亚男(1996—),女,湖南岳阳人,合肥工业大学硕士生;

肖本贤(1964—),男,安徽无为,博士,合肥工业大学教授,硕士生导师。

付需求。为保持高质量的生产和供应水平^[1-3],越来越多的仓库系统选择采用各种机器人和无人驾驶工业运输和搬运车辆来替代传统劳动力完成一系列不需要专业知识的工作^[4-6]。因此,移动机器人和无人驾驶车辆的路径规划算法成为研究热点,如神经网络^[7-8]、Dijkstra 算法、蚁群算法^[9]、人工势场法、RRT* 算法^[10-12]、A* 算法^[13-16]等智能路径规划算法。

神经网络算法学习时间长,规划效率低。Dijkstra 算法搜索为盲搜索,可以得到全局最优路径,但运行效率低下。人工势场法计算量小,但存在局部极小陷阱。RRT* 算法搜索速度快,但具有很强的随机性,导致收敛速度慢、生成路径不平滑。

A* 算法是一种启发式搜索算法,近年来被广泛应用于路径规划领域。由于该算法具有最优性、完备性和高效性,其系列改进算法一直是路径规划领域的研究热点。启发函数能引导搜索向最优方向进行,在静态网络中搜索路径有效而直接,能搜索到最优路径;但也存在不足,得到的路径转折点多,存在大量冗余节点,不利于无人驾驶车辆进行实际跟踪^[17-19];文献[20]提出一种可搜索无线邻域的改进 A* 算法为无人驾驶车辆寻找可行路径,对栅格进行线性插值,提高搜索效率,但在复杂地图上优势不明显;文献[21]提出以矩阵单元进行拓展,提高了算法效率,优化了路径质量,但在低复杂度且不开阔环境中,性能优化不明显;文献[22]通过拓展可搜索邻域来增加路径搜索方向,规划所得路径比传统 A* 算法的更短,转弯次数更少,但搜索时间大大增加,在复杂精细地图上实用性不高。

本文提出一种基于行驶时间优化的改进 A* 算法。以无人叉车为研究对象,从以下 3 个方面进行改进:

1) 对实际代价函数进行优化。引入转弯代价补偿参数用以表示转弯额外行驶时间,将转弯操作导致的额外操作时间纳入评估函数,使搜索过程趋向于航向不变的节点。

2) 对启发函数优化。引入父节点启发函数值,同时对启发函数权重进行自适应调整,确保搜索过程快速靠近目标节点且路径最优。

3) 对路径规划的初步结果进行路径剪枝,平滑路径降低路径转弯总角度,使路径适于无人叉车的实际跟踪。

针对以上改进,本文设计了不同环境地图进行仿真实验,证明所提出改进算法的有效性和实用性。

1 环境建模

无人叉车的出现提高了工作效率,降低了人力和设备成本。无人叉车的工作效率提升不仅意味着提升叉车行驶速度,还意味着缩短工作循环内的工作时间。叉车在弯道行驶时,需要进行额外减速、制动和重新加速,这不仅增加了工作循环内的工作时间,还会影响无人叉车的整体性能,带来侧翻风险,导致更高的车辆维护成本^[4]。

栅格法是目前最常用的环境建模方法之一。将地图切割成大小相同相互连接的栅格,用黑色栅格代表障碍物所在区域,即不可通行,空白栅格表示该区域可供穿越行驶。在常见的 8 向栅格图中,叉车可在相邻可通行栅格间横向、纵向或斜向 45° 行驶;16 向 24 邻域栅格图进一步增加了搜索方向,拓展邻域的增加使得拓展节点时搜索量成倍增加。文献[22]证明机械地增加搜索邻域来扩大 A* 算法的搜索方向在一定程度上改善了路径的平滑度,但算法搜索时间和存储空间显著增加。出于搜索效率考量,本文改进算法基于 8 向栅格地图进行环境建模。

在结构化地图中,叉车在通道中行驶,通道可分为单行道、双行道和多行道,在通道两侧有货架、堆叠货物等多种类型的障碍物。因为在非结构化环境,如小型非结构化仓库中障碍物分布不规则、通道宽度不确定的情况下,叉车行驶可能产生更多转弯,转弯角度更大,所以有更大的路径优化需求。因此,本文的改进 A* 算法在进行有效性验证时建立了多种栅格地图对叉车工作环境进行模拟。

2 传统 A* 算法

A* 算法通过设计带约束性的启发式函数,在状态空间内对当前节点的每一个待扩展节点的成本进行评估,并且设计与问题特征相关的启发函数对性能进行调整,通过比较每个扩展节点的成本,选择最理想的节点去扩展,直到发现目标节点。在执行路径规划时,OPEN list 用于保存搜索过程中遇到的扩展节点,将这些节点按评估函数成本 $f(n)$ 值的大小排序;CLOSE list 用于保存 OPEN list 中评估函数 $f(n)$ 值最小的可扩展节点。

A* 算法的评估函数 $f(n)$ 计算公式为:

$$f(n) = g(n) + h(n) \quad (1)$$

该函数计算由两部分组成,即从初始节点 S 到候选节点 n 的路径的实际距离成本 $g(n)$ 和从候选节点 n 到目标节点 G 的最优路径的评估成本 $h(n)$ 。

实际距离成本 $g(n)$ 使用欧几里德距离累加表示,两点间欧几里德计算公式如下:

$$d_{i,j} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \quad (2)$$

则相邻节点 n 的实际代价函数 $g(n)$ 值和启发函数 $h(n)$ 值可用下列公式进行计算:

$$g(n) = \sum_{i=1}^n d_{i,i-1} \quad (3)$$

$$h(n) = d_{n,G} = \sqrt{(x_G - x_n)^2 + (y_G - y_n)^2} \quad (4)$$

其中: x_G, y_G 为目标节点 G 的横坐标和纵坐标; x_n, y_n 为候选节点 n 的横坐标和纵坐标。

传统 A* 算法路径规划过程如图 1 所示。

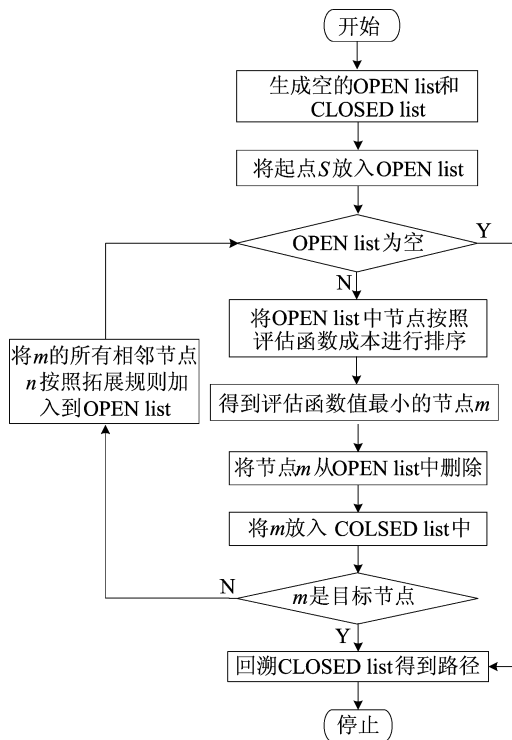


图 1 传统 A* 算法流程

为提高搜索效率、缩短运算时间,本文将栅格图中每个栅格大小设置为 1×10 ,当叉车在栅格间横向或竖向移动时,其运动距离记为 1,斜向移动时,运动距离记为 $\sqrt{2}$, $h(n)$ 的计算公式如下:

$$h(n) = 1.4 \min(\Delta x, \Delta y) + |\Delta x - \Delta y| \quad (5)$$

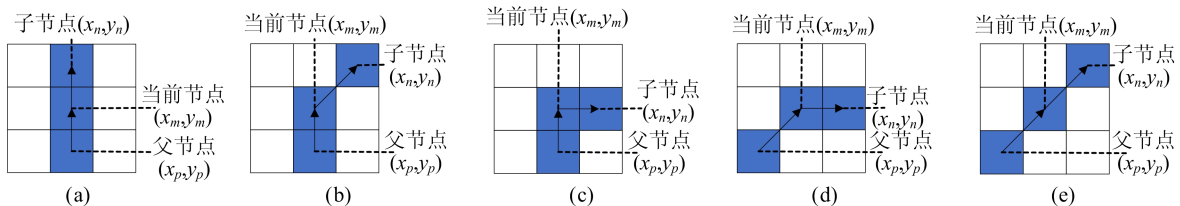


图 2 节点与行驶方向

其中: $\Delta x = |x_n - x_{n+1}|$; $\Delta y = |y_n - y_{n+1}|$; x_{n+1}, y_{n+1} 分别为节点 $n+1$ 的横、纵坐标。

传统 A* 算法的评价函数综合了 Dijkstra 算法和最佳优先搜索 (best-first-search, BFS) 算法。将 A* 算法评价函数 $f(n)$ 一般化,用下列公式表示:

$$f(n) = \alpha g(n) + \beta h(n) \quad (6)$$

当系数 $\alpha=1, \beta=0$ 时,算法退化为 Dijkstra 算法,倾向于广度优先搜索,即搜索偏向接近起点的节点,搜索范围大、时间长,但可求得两点之间的最短路径;当系数 $\alpha=0, \beta=1$ 时,算法搜索启发性更强,更倾向于搜索离目标节点更近的节点,此时 A* 算法可视为广度优先搜索 (breadth first search, BFS) 算法,但规划所得路径不一定是最优路径,因此评估函数 $f(n)$ 中各项权重需要结合实际需要合理分配以得到更优、更平滑的路径。

3 改进 A* 算法

本节对传统 A* 算法进行改进,目标是取得静态环境下无人叉车行驶时间最短的路径。一方面向实际代价估计函数 $g(n)$ 中引入转弯代价补偿参数构造转弯行驶时间函数,以此减少生成路径中的转弯次数;另一方面同时向原有的启发函数中引入父节点启发函数 $h(p)$,并对自身权重进行自适应调整;最后对规划所得初始路径进行剪枝,得到最优路径。

3.1 转弯代价补偿

叉车行驶过程中存在 5 种不同的直行与转弯情形,如图 2 所示。父节点 p 、当前节点 m 和拓展节点 n 的坐标位置可以用于判断前进方向是否发生改变:

$$\lambda_1 = (x_m - x_p) - (x_n - x_m) \quad (7)$$

$$\lambda_2 = (y_m - y_p) - (y_n - y_m) \quad (8)$$

其中: (x_m, y_m) 为当前节点 m 的坐标; (x_p, y_p) 为父节点 p 的坐标; (x_n, y_n) 为拓展节点 n 的坐标; λ_1 为横向转向因子; λ_2 为纵向转向因子;当 $\lambda_1 = \lambda_2$ 时,叉车行驶方向不变,当 $\lambda_1 \neq \lambda_2$ 时,航向改变,需转弯。

实际行驶过程中,同样距离行驶在转弯路段所需时间远超过直行路段,而传统 A* 算法评估函数中 $g(n)$ 和 $h(n)$ 仅表示实际已规划的路径长度和候选节点到目标节点的最短距离,未考虑转弯操作过程中由减速制动、重新加速等操作带来的额外时间成本,规划所得路径转弯次数多,转弯角度大,在无人叉车的实际工作中并不适用。

为解决上述问题,本文将行驶时间细分为直行行驶时间成本和由当前节点 m 拓展到候选节点 n 因转弯增加的行驶时间成本 $g_t(n)$ 。直行行驶时间由实际代价函数 $g(n)$ 表示;转弯行驶时间 $g_t(n)$ 需要引入转弯代价补偿参数 ω 进行计算,由转弯代价补偿参数 ω 与拓展栅格的实际距离 d 加权表示,即

$$g_t(n) = \omega(\lambda_1 - \lambda_2)d_{m,n} \quad (9)$$

改进后实际代价函数 $g'(n)$ 由两部分组成,即固有的直线行驶成本 $g(n)$ 和额外的转弯时间成本 $g_t(n)$,其计算公式为:

$$g'(n) = g(n) + g_t(n) = \sum_{i=1}^n [1 + \omega(\lambda_1 - \lambda_2)]d_{i-1,i} \quad (10)$$

3.2 启发函数优化

A* 算法的启发函数能使算法的搜索方向趋向于更有目的地接近端点。为约束算法搜索方向,减少往返搜索次数,使其快速靠近目标节点,本文在启发函数中引入了父节点的启发估计代价 $h(p)$,改进后的启发函数计算公式为:

$$h'(n) = h(n) + h(p) \quad (11)$$

考虑到本文启发函数 $h(p)$ 使用欧几里德距离对两点间距离进行估计,此距离为两点之间最短距离,而无人叉车的实际工作环境中存在货架、货物、建筑等障碍物,使得两点间实际路径距离要远大于启发函数所估计的最短距离。在传统 A* 算法中,启发函数权重为 1,在小范围内有多个评价函数值相等的候选节点,需要进行多次比较,搜索效率低。针对上述问题,本文通过比较拓展节点到目标节点及起始节点的距离远近,对启发函数的权重 β 进行自适应调整,以提高搜索效率,同时避免路径规划结果陷入局部最优,启发函数权重 β 的计算公式如下:

$$\beta = 1 + h(n)/D \quad (12)$$

其中, D 为起始节点到目标节点的欧几里德距离,是一个常量。 D 的计算公式如下:

$$D = d_{G,S} = \sqrt{(x_S - x_G)^2 + (y_S - y_G)^2} \quad (13)$$

其中: (x_G, y_G) 为起始节点坐标; (x_S, y_S) 为目标

节点坐标。当候选节点离目标节点较远时, $h(n)$ 较大,启发函数权重 β 取值较大,路径搜索方向快速接近目标节点;当候选节点逐渐靠近目标节点时, $h(n)$ 减小,启发函数权重 β 也随之减小,此时算法搜索范围扩大以确保目标点可达,避免陷入局部最优解。

经改进后评价函数 $f(n)$ 的计算公式为:

$$f(n) = g'(n) + \beta h'(n) = \sum_{i=1}^n [1 + \omega(\lambda_1 - \lambda_2)]d_{i-1,i} + \left[1 + \frac{h(n)}{D}\right][h(n) + h(p)] \quad (14)$$

3.3 路径剪枝

对改进后的 A* 算法所得路径采取全局路径剪枝,以剔除冗余拐点,进一步平滑路径,减小小路径拐弯总角度。

得到初始路径后,从目标节点 G 开始,直接将点 G 与倒数第 2 个拐点 N_2 相连接,若两点连线不与障碍物相交,则与倒数第 3 个拐点 N_3 连接,依此类推,直到连线遇到障碍物。假设目标节点 G 与倒数第 i 个拐点 N_i 的连线与障碍物相交,即 $L(G, N_i) = +\infty$,则依次连接目标节点 G 与倒数第 i 个拐点 N_i 和倒数第 $i-1$ 个拐点 N_{i-1} 之间(包括拐点 N_{i-1})的所有节点,直到节点 X 与目标节点 G 的连线是一条不与障碍物相交的直线,保留节点 X 与节点 G ,舍弃节点 X 到节点 G 间其余节点,以节点 X 为起点,继续向前回溯,直到回溯到起始节点 S ;最后依次连接所有剩余关键节点,得到剪枝后的优化路径。

路径剪枝过程演示如图 3 所示。

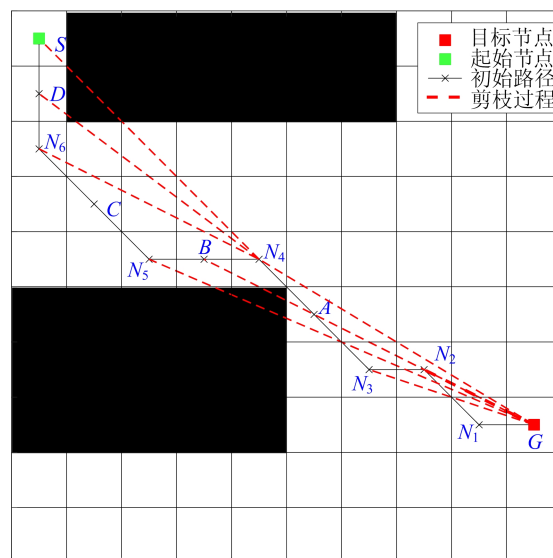


图 3 路径剪枝过程

其中:红、绿色节点分别表示路径目标节点和起始节点;黑色实线为初始路径;红色虚线为剪枝过程路径。

首先将目标节点 G 与倒数第 2 个拐点 N_2 相连,路径与障碍物不交叉;继续依次连接倒数第 3 个拐点 N_3 、倒数第 4 个拐点 N_4 连接,其路径与障碍物均不交叉;直到 G 与倒数第 5 个拐点 N_5 相连,路径与障碍物交叉,即 $L(G, N_5) = +\infty$; N_4 、 N_5 存在中间节点 B ,连接 G 、 B ,其路径与障碍物相交 $L(G, B) = +\infty$,则该段路径剪枝结果为:

$$\text{Path}(G, N_4) = G \rightarrow N_4 \quad (15)$$

以 N_4 为下一次剪枝的起点,将其与其后倒数第 2 个拐点即 N_6 连接,路径与障碍物不交叉;继续向前回溯,到达起始节点 S ,连接 S 、 N_4 , $L(S, N_4) = +\infty$;连接两拐点的中间节点 D ,有 $L(D, N_4) = +\infty$,则该段路径剪枝结果为:

$$\text{Path}(N_4, S) = N_4 \rightarrow N_6 \rightarrow S \quad (16)$$

剪枝完成后最优路径完整表示如下:

$$\text{Path}(G, S) = G \rightarrow N_4 \rightarrow N_6 \rightarrow S \quad (17)$$

剪枝前、后路径对比如图 4 所示。其中,红色路线为路径剪枝得到的最优路径。

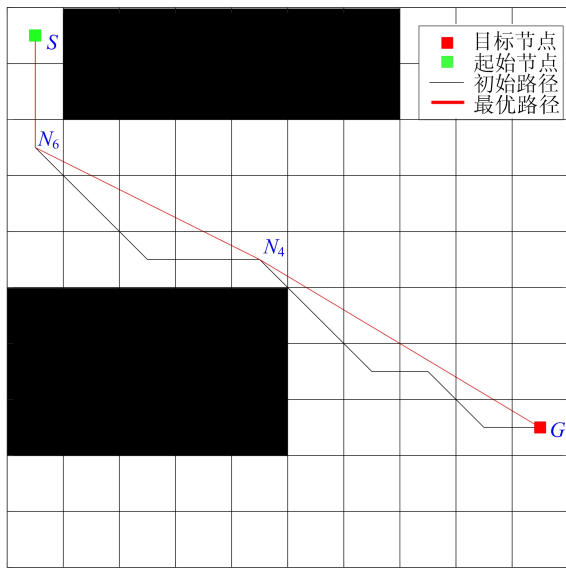


图 4 剪枝前、后路径对比

传统 A* 算法和本文改进 A* 算法的最优路径对比数据见表 1 所列。

表 1 最优路径对比

算法	规划时间/s	路径长度	拐点总数	拐弯总角度/(°)
A* 算法	0.241	130.00	6	270.0
本文算法	0.223	102.96	1	54.5

由表 1 可知,相较于传统 A* 算法,本文改进 A* 算法所得到的最优路径长度减少 20.80%,拐点总数减少 83.33%,转弯总角度减少 215.5°,优化效果显著。

文献[23]提出的改进 A* 算法只保留起始点和拐点进行剪枝操作,共线节点在路径优化的第 1 步就被忽略,剪枝过程粗放,导致所得最终路径并非最优,在复杂地图上优化效果欠佳。本文所提出的改进 A* 算法优先判断拐点间的连线是否与障碍物相交,无需逐个节点搜索,缩短了计算时间;同时保留了与障碍物相交和不相交的相邻连线间的共线节点用于精确搜索,得到最优路径。

4 实验分析

4.1 参数 ω 的选取对结果的影响

为研究不同的转弯代价补偿参数 ω 对 A* 算法所得路径长度和拐弯节点总数的影响,在 30×30 的栅格地图中,随机生成 50 组起始节点和目标节点,在转弯代价补偿参数 ω 取不同值时,使用本文改进 A* 算法进行路径规划实验。实验过程中,参数 ω 的变化范围控制在 $[0, 1.2]$ 之间,取值间隔设置为 0.1,记录改进 A* 算法关键性能参数,性能表现变化趋势如图 5 所示。

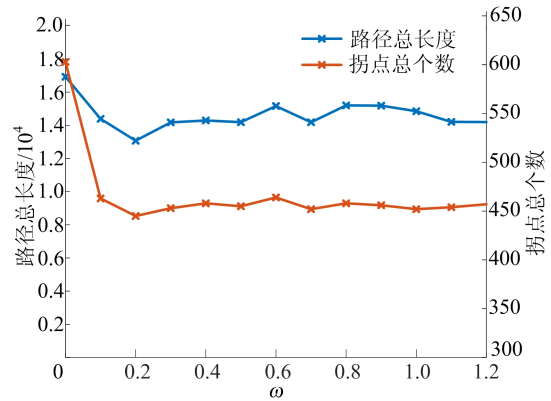


图 5 ω 不同取值时算法性能

由图 5 可知: $\omega=0$ 时为传统 A* 算法,与本文改进 A* 算法相比,其规划结果拐点数量最多,路径长度最长;在进行实际代价估计函数和启发函数的优化以及对路径剪枝后,本文改进 A* 算法规划所得路径整体的拐点个数和路径总长度大大减小;在转弯代价补偿参数 $\omega=0.2$ 时,转弯次数显著减少,且路径长度最短,本文改进的 A* 算法比传统 A* 算法转弯节点个数减少了 29.23%,路径总长度减少了 20.63%,效果最为理想。

因此本文改进 A* 算法转弯代价补偿参数 ω 的最佳取值为 0.2, 下文所有仿真对比实验皆基于此条件进行。

4.2 仿真结果

为了评估本文提出的基于无人叉车行驶时间优化的改进 A* 算法的性能, 在 2.3 GHz 处理器、16 GiB 运行内存的 Windows 11 平台上进行仿真实验。

本文分别模拟结构化和非结构化环境构造栅格地图, 使用传统 A* 算法、蚁群(ant colony optimization, ACO)算法、Dijkstra 算法和本文改进 A* 算法进行路径规划, 分析并对比 4 种算法的性能。

4.2.1 结构化环境地图

本文建立了叉车结构化工作环境栅格地图, 在该地图上设立相同的起始节点, 使用 4 种算法进行路径规划, 仿真实验结果如图 6 所示。其中: 绿色栅格为起始节点; 红色栅格为目标节点; 黑色栅格为障碍物; 白色栅格为可通行区域; 红色轨迹为算法规划所得路径。

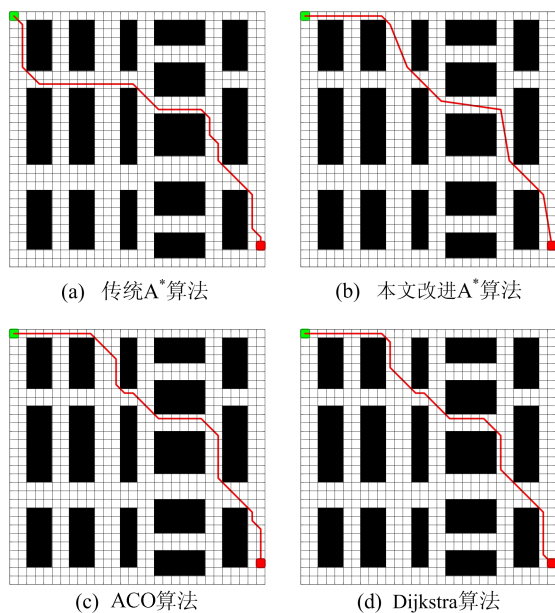


图 6 工作实景栅格地图仿真结果

工厂地图上各算法的仿真性能指标结果对比见表 2 所列。由表 2 可知: 在搜索时间上, 本文改进 A* 算法比传统 A* 算法减少了 28.10%, 比 Dijkstra 算法减少了 34.80%, 实时性得到明显提升; 本文改进 A* 算法所得路径长度为 463.49, 比传统 A* 算法缩短了 4.20%。

决定路径平滑程度的关键数据为拐点个数及

拐弯总角度。拐点个数越少, 拐弯总角度越小, 路径就越平滑, 单个工作循环内需要进行的降速制动等操作频率就越低, 行驶时间就越短。从表 2 还可以看出: 本文改进 A* 算法比传统 A* 算法拐点个数减少了 46.20%, 比 Dijkstra 算法减少了 41.70%, 比 ACO 算法减少了 36.40%; 拐点个数的减少, 使得路径拐弯总角度明显减少, 这一指标本文改进 A* 算法比传统 A* 算法减少 53.60%, 比 Dijkstra 算法减少 50.10%, 比 ACO 算法减少 45.30%。

综上分析可知, 本文改进 A* 算法的效率更高, 路径长度更短, 且转弯角度更小, 路径更平滑。

表 2 4 种算法性能指标对比

算法	规划时间/s	路径长度	拐点总数	拐弯总角度/(°)
本文算法	0.289	463.49	7	271.75
A* 算法	0.402	483.85	13	585.00
ACO 算法	9.993	477.99	12	540.00
Dijkstra 算法	0.443	472.13	11	495.00

4.2.1 非结构化环境地图

为了验证本文改进 A* 算法的普适性及在复杂环境下的有效性, 分别构建 30×30 和 50×50 的非结构化地图用于模拟更为复杂的工作环境。各算法仿真结果如图 7、图 8 所示。

2 种非结构化地图中各算法性能参数及本文改进 A* 算法、传统 A* 算法、ACO 算法和 Dijkstra 算法的性能对比见表 3、表 4 所列。

本文改进 A* 算法旨在降低无人叉车单个工作循环内的行驶时间。从图 7、图 8 可以看出, 3 种改进算法相较于传统 A* 算法路径规划结果都更平滑, 路径长度更短。但对比 2 种尺寸地图下 3 种改进算法的仿真结果可知: 随着地图复杂程度增加, ACO 算法、Dijkstra 算法减少拐点数量、平滑路径转弯角度的效果变差, 在 50×50 环境地图中仍有较多冗余拐点保留; 而本文改进 A* 算法在 50×50 地图中路径剪枝效果仍然稳定, 使得拐点个数和转弯角度大大减小, 大幅减少因转弯而增加的操作时间。

由表 4 可知: 本文改进 A* 算法在 50×50 的地图中比传统 A* 算法拐点数量减少 47.10%, 拐弯总角度减少 65.90%, 路径总长度减少 2.24%, 路径平滑程度有很大提升; 比 ACO 算法、Dijkstra 改进算法表现更稳定, 实时性更强, 优化效果更好。

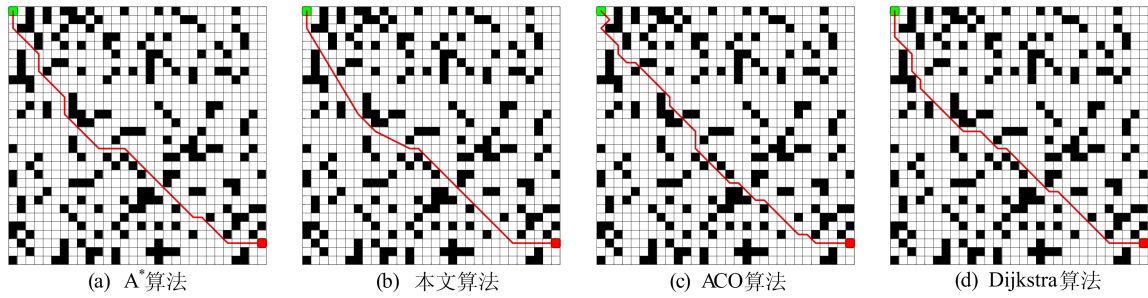


图 7 30×30 非结构化环境地图仿真结果

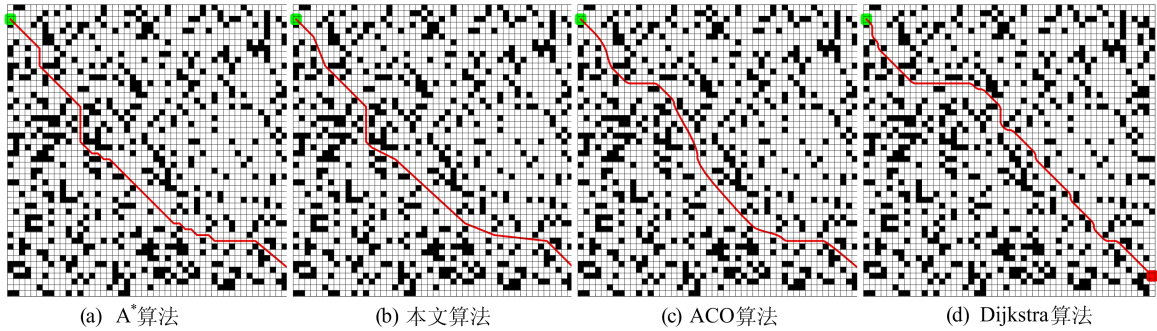


图 8 50×50 非结构化环境地图仿真结果

表 3 30×30 地图各算法性能参数

算法	路径长度	拐点总数	拐弯总角度/(°)	减少率/%		
				路径长度	拐点总数	拐弯总角度
本文算法	425.19	6	180.00	0	0	0
A* 算法	436.98	10	450.00	2.70	40.00	60.00
ACO 算法	440.32	17	855.00	4.51	64.71	78.95
Dijkstra 算法	439.53	12	251.08	2.70	50.00	66.67

表 4 50×50 地图各算法性能参数

算法	路径长度	拐点总数	拐弯总角度/(°)	减少率/%		
				路径长度	拐点总数	拐弯总角度
本文算法	706.18	9	245.61	0	0	0
A* 算法	722.37	17	720.00	2.24	47.10	65.90
ACO 算法	727.40	22	990.00	2.92	59.09	75.19
Dijkstra 算法	724.98	20	900.00	2.59	55.00	72.71

5 结 论

针对传统 A* 算法路径规划结果实际应用于工业无人叉车上不足,本文提出了一种改进的 A* 算法对无人叉车行驶时间进行优化。该算法通过引入转弯代价补偿参数设计转弯行驶时间函数和启发函数的优化,在保证路径最优的情况下减少路径规划中拐点个数和转弯总角度,引入自适应权重缩短无人叉车从起始位置到目标位置所需的时间,确保目标可达。仿真实验结果验证了该算法的有效性和可靠性。

本文提出的改进 A* 算法适用于已知环境的路径规划,规划所得路径拐点少,转弯角度小,更

适用于无人叉车的实际作业,能大大减少无人叉车工作中转向、刹车次数,提高车身稳定性及工作效率。未来的工作需要考虑单个无人叉车载重不同、重心位置改变时所能跟踪的最大转角并进行路径规划,进一步考虑在多个无人叉车协同工作的动态环境中得到路径短、拐点少的平滑路径,以适应更加多变的工作场景。

[参 考 文 献]

[1] DRAGANJAC I, MIKLIC D, KOVACIC Z, et al. Decentralized control of multi-AGV systems in autonomous warehousing applications[J]. IEEE Transactions on Automation Science and Engineering, 2016, 13(4):1433-1447.

(下转第 1478 页)

- 情感分析[J]. 计算机应用, 2018, 38(11): 3075-3080.
- [12] 马学森, 陈树友, 许向东, 等. 基于神经网络与马尔可夫组合模型的视频流行度预测算法[J]. 电信科学, 2021, 37(8): 18-26.
- [13] SALINAS D, FLUNKERT V, GASTHAUS J, et al. DeepAR: probabilistic forecasting with autoregressive recurrent networks[J]. International Journal of Forecasting, 2020, 36(3): 1181-1191.
- [14] 武维, 李泽平, 杨华蔚, 等. 融合内容特征和时序信息的深度注意力视频流行度预测模型[J]. 计算机应用, 2021, 41(7): 1878-1884.
- [15] LIAO D L, XU J, LI G F, et al. Popularity prediction on online articles with deep fusion of temporal process and content features[C]//Proceedings of the 33rd AAAI Conference on Artificial Intelligence. Palo Alto, CA: AAAI Press, 2019: 200-207.
- [16] SHIH S Y, SUN F K, LEE H Y. Temporal pattern attention for multivariate time series forecasting[J]. Machine Learning, 2019, 108(8/9): 1421-1441.
- [17] WEI X L, LEI B B, OUYANG H B, et al. Stock index prices prediction via temporal pattern attention and long-short-term memory[J]. Advances in Multimedia, 2020(4): 8831893.
- [18] 王竹荣, 薛伟, 牛亚邦, 等. 基于注意力机制的泊位占有率预测模型研究[J]. 通信学报, 2020, 41(12): 182-192.

(责任编辑 李 凯)

(上接第 1453 页)

- [2] FREGO M, BEVILACQUA P, DIVAN S, et al. Minimum time-minimum jerk optimal traffic management for AGVs[J]. IEEE Robotics and Automation Letters, 2020, 5(4): 5307-5314.
- [3] 倪郁东, 李媛媛, 沈吟东, 等. 基于改进人工蜂群算法的移动机器人路径规划研究[J]. 合肥工业大学学报(自然科学版), 2019, 42(11): 1569-1575.
- [4] LEMERLE P, HÖPPNER O, REBELLE J. Dynamic stability of forklift trucks in cornering situations: parametrical analysis using a driving simulator[J]. Vehicle System Dynamics, 2011, 49(10): 1673-1693.
- [5] 张艳, 张明路, 蒋志宏, 等. 动态环境下移动机器人路径规划的研究[J]. 合肥工业大学学报(自然科学版), 2020, 43(10): 1297-1306.
- [6] 陈虹, 郭露露, 宫洵, 等. 智能时代的汽车控制[J]. 自动化学报, 2020, 46(7): 1313-1332.
- [7] SUN H, ZHANG W, YU R, et al. Motion planning for mobile robots-focusing on deep reinforcement learning: a systematic review[J]. IEEE Access, 2021, 9: 69061-69081.
- [8] 陈秋莲, 郑以君, 蒋环宇, 等. 基于神经网络改进粒子群算法的动态路径规划[J]. 华中科技大学学报(自然科学版), 2021, 49(2): 51-55.
- [9] 王庆, 徐海明, 吕品, 等. 基于改进蚁群算法的多旋翼无人机航迹规划研究[J]. 合肥工业大学学报(自然科学版), 2021, 44(9): 1172-1178.
- [10] 杜明博, 梅涛, 陈佳佳, 等. 复杂环境下基于 RRT 的智能车辆运动规划算法[J]. 机器人, 2015, 37(4): 443-450.
- [11] 刘成菊, 韩俊强, 安康. 基于改进 RRT 算法的 RoboCup 机器人动态路径规划[J]. 机器人, 2017, 39(1): 8-15.
- [12] BAZZI D, PRIORA G, ZANCHETTIN A M, et al. RRT* and goal-driven variable admittance control for obstacle avoidance in manual guidance applications[J]. IEEE Robotics and Automation Letters, 2022, 7(2): 1920-1927.
- [13] ZHAO J, ZHANG Y, MA Z, et al. Improvement and verification of A-star algorithm for AGV path planning[J]. Computer Engineering and Applications, 2018, 54(21): 217-223.
- [14] XIN W, WANLIN L, CHAO F, et al. Path planning research based on an improved A* algorithm for mobile robot[J]. IOP Conference Series: Materials Science and Engineering, 2019, 569(5): 52044.
- [15] 劳彩莲, 李鹏, 冯宇. 基于改进 A* 与 DWA 算法融合的温室机器人路径规划[J]. 农业机械学报, 2021, 52(1): 14-22.
- [16] 宗成星, 陆亮, 雷新宇, 等. 一种基于 A* 算法的空间多自由度机械臂路径规划方法[J]. 合肥工业大学学报(自然科学版), 2017, 40(2): 164-168.
- [17] 吕恩利, 林伟加, 刘妍华, 等. 基于 B 样条曲线的智能叉车托盘拾取路径规划研究[J]. 农业机械学报, 2019, 50(5): 394-402.
- [18] LIN R, HUANG H, LI M. An automated guided logistics robot for pallet transportation[J]. Assembly Automation, 2020, 41(1): 45-54.
- [19] HUANG Y, DING H, ZHANG Y, et al. A motion planning and tracking framework for autonomous vehicles based on artificial potential field elaborated resistance network approach[J]. IEEE Transactions on Industrial Electronics, 2020, 67(2): 1376-1386.
- [20] 辛煜, 梁华为, 杜明博, 等. 一种可搜索无限个邻域的改进 A* 算法[J]. 机器人, 2014, 36(5): 627-633.
- [21] 李冲, 张安, 毕文豪. 单边矩形扩展 A* 算法[J]. 机器人, 2017, 39(1): 46-56.
- [22] 崔宝侠, 王森弛, 段勇. 基于可搜索 24 邻域的 A* 算法路径规划[J]. 沈阳工业大学学报, 2018, 40(2): 180-184.
- [23] 王维, 裴东, 冯璋. 改进 A* 算法的移动机器人最短路径规划[J]. 计算机应用, 2018, 38(5): 1523-1526.

(责任编辑 胡亚敏)