

基于改进 A* 算法的移动机器人全局最优路径规划^①李晓露^{②*} 熊禾根^{③**} 陶 永^{***} 李公法^{**}

(* 武汉科技大学冶金装备及其控制教育部重点实验室 武汉 430081)

(** 武汉科技大学机械传动与制造工程湖北省重点实验室 武汉 430081)

(***) 北京航空航天大学机械工程及自动化学院 北京 100191)

摘要 在栅格地图环境下,传统 A* 算法搜索路径时选取的路径点受限于栅格中心,并且路径的转折角度固定为特定的离散值,因此存在长度非最优以及冗余转折较多的问题。为此,提出一种新启发搜索策略下的改进 A* 算法。在探索当前节点的每个邻域节点时,将邻域点父节点的选取范围扩大到从当前点至起始点的整个支路,采用邻域点与支路上的点直接相连的方式,找到所需真实代价 G 最小的安全路径,此时支路上对应的点便为该邻域点的父节点。实验结果表明,改进算法能够有效提高路径质量,即规划的路径长度更短、转折更少。与此同时,真实代价 G 和预估代价 F 之间的偏差降低,启发函数的启发能力增强,寻路效率也有所提高。

关键词 改进 A* 算法; 移动机器人; 路径规划; 路径质量; 寻路效率

0 引言

机器人被誉为制造业皇冠顶端的明珠,是衡量一个国家科技创新能力和制造水平的标尺,也推动着新一轮的产业革命。近年来,在国家的大力扶持下,机器人产业发展迅速,移动机器人也成为当前研究的热点,并应用到各种不同的领域^[1]。路径规划是移动机器人研究中的重点问题,是指机器人根据特定的需求或评价指标从当前工作环境中搜索出一条从起始点到目标点的最优无碰撞路径^[2-3]。基于栅格地图,常用的路径规划方法有遗传算法^[4-5]、蚁群算法^[6-9]、A* 算法^[10]、D* 算法^[11-12]等。

A* 算法是一种典型的启发式搜索算法,具有快速、高效地搜索最优路径的优点^[13],因此被广泛应用于移动机器人的路径规划问题中。目前,已经有众多学者针对不同需求对传统 A* 算法的路径长度、寻路效率和转折点数量等方面进行了改进。文

献[14]引入了多项式曲线,改善了传统 A* 算法路径的平滑性,但未考虑算法的效率。文献[15]提出一种新的单边矩形扩展 A* 算法,采用受迫扩展规则,适度改善了路径长度、转折点数量以及算法效率,但路径长度仍存在改进空间。文献[16]通过筛选跳点的方式减少了需要扩展的节点数量。文献[17]采用双向搜索的方式以及正反向搜索交替机制,都提升了寻路效率,但也存在转折点数量较多、路径长度并非最优的问题。

在栅格环境下,传统 A* 算法扩展节点时只选取栅格中心,路径点选择受限、转折方向被离散化导致路径长度不是最优,在此提出一种新的改进 A* 算法,在添加安全距离的前提下通过检查两点直连路径安全性的方式,扩大 OPEN 表中节点的父节点选取范围,降低转折的局限性。实验结果表明,相比传统 A* 算法,改进算法能明显提升路径规划问题的求解质量和求解效率,因而具有较重要的理论和

① 国家自然科学基金(51875422,51575407)和国家科技支撑计划(2015BAF01B03)资助项目。

② 男,1991年生,硕士生;研究方向:机器人路径规划;E-mail:lixiaoluqiuxue@126.com

③ 通信作者,E-mail: xionghen@126.com

(收稿日期:2020-03-20)

现实意义。

1 问题描述

移动机器人的路径规划一般分为两步,一是利用机器人配置的传感器将物理世界的信息转换成机器人可以识别的环境模型,二是引入相关的算法在环境模型中搜索并生成适合机器人行走的路径。

1.1 栅格法环境建模

栅格法是移动机器人路径规划最常用的环境建模方法,能简单而精确地表示各种不规则障碍物的信息,其基本思想是将机器人的工作空间划分为大小相同的栅格,并用数值矩阵来储存环境信息^[18]。常规栅格法将物理环境严格划分为自由区域和障碍物区域,因此数值矩阵可以简化为 0-1 矩阵。在栅格地图中,自由空间用白色栅格表示,矩阵中的值为 0;障碍物空间用黑色栅格表示,矩阵中的值为 1。

设将机器人工作空间划分为 $R \times C$ 个栅格,其中 R 为每行栅格数, C 为每列栅格数。用 M 表示数值矩阵, M_{ij} 为 0-1 矩阵中每个元素的值,用 Ω 表示工作空间所有的环境信息,则栅格地图的数学模型如式(1)所示。

$$\Omega = \sum_{\substack{i \in [1, R] \\ j \in [1, C]}} M_{ij}, \quad M_{ij} = \begin{cases} 0 & \text{无障碍} \\ 1 & \text{有障碍} \end{cases} \quad (1)$$

图 1 所示为栅格地图示例,对应的数值矩阵:

$$M = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

为方便进行路径规划,在建立环境模型时通常会根据机器人本体的最大半径对障碍物进行膨胀处理,之后便可以将机器人当作一个质点,而忽略机器人本体尺寸对路径安全性的影响。

本文讨论的机器人为全向移动机器人,环境中的障碍物均为静态障碍物并已进行膨胀处理。设定单个栅格的边长为 1,机器人质点与障碍物的安全距离 S 为栅格边长的一半,即规划的路径与所有障碍物的最小距离必须大于或等于 0.5。

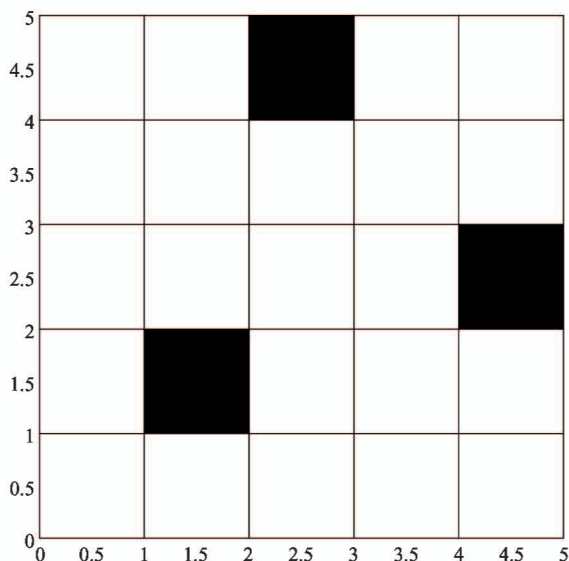


图 1 栅格地图示例

1.2 传统 A* 算法

A* 算法综合了 Dijkstra 算法和 BFS (Breadth-First-Search) 算法各自的特点,既能搜索到最短路径,也尽可能地保证搜索效率。传统 A* 算法的核心是其评价函数综合考虑了两个代价值,一个是起始点到当前节点的真实代价,另一个是当前节点到目标点的预估代价,可以引导搜索方向,减小搜索范围,提高搜索效率。

传统 A* 算法估价函数的形式一般如下:

$$f(n) = g(n) + h(n) \quad (2)$$

式中, $f(n)$ 为全局评价函数; $g(n)$ 为真实代价函数; $h(n)$ 为启发函数,评估当前节点 n 到目标节点的预估代价。

启发函数在估价函数中起关键性的作用,影响 A* 算法规划路径的长度和搜索路径的效率。当 $h(n)$ 小于 $g(n)$ 时,需搜索的节点数量多、效率低,但能够得到最短路径;当 $h(n)$ 大于 $g(n)$ 时,需搜索的节点数量少、效率高,但无法保证得到最短路径^[19]。因此,为保证最短路径并提高搜索效率,应该使预估代价小于并尽可能接近真实代价。

由于二维空间中代价值一般指两个节点之间的线段距离,因此启发函数一般采用欧氏距离,即:

$$h(n) = \sqrt{(x_n - x_g)^2 + (y_n - y_g)^2} \quad (3)$$

式中, (x_n, y_n) 为当前点 n 的坐标, (x_g, y_g) 为目标点坐标。

A* 算法的寻路过程可以简单概括为:从起始点开始探索,不断访问其邻域节点(通常为 8 邻域),将已访问但未探索的自由节点存入开启列表 OPEN,已探索的节点存入封闭列表 CLOSE,然后从 OPEN 表中选取评估代价最小的节点作为下一个探索的点,直到发现目标点为止,最后从目标点向起始点回溯得到最终路径,相应流程图如图 2 所示。

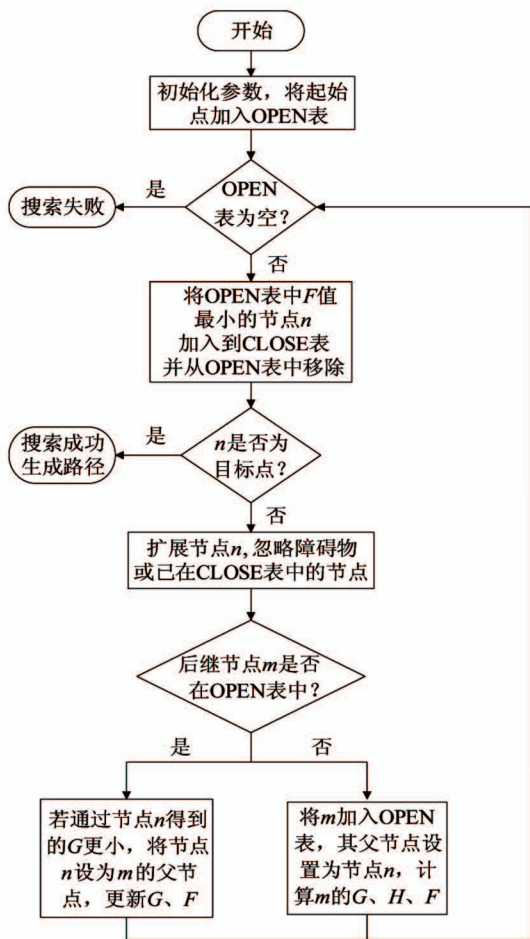


图 2 传统 A* 算法流程图

1.3 栅格环境下传统 A* 算法的缺陷与原因分析

基于栅格地图进行路径规划,传统 A* 算法存在的缺陷主要如下。

(1) 路径点数量较多且位置固定于栅格的中心,搜索路径的时候方向随机,导致最终路径的长度往往不是实际最优,冗余的转折也可能较多。

(2) 为保证能搜索到最短路径,预估代价一般使用两点的欧氏距离,而传统路径生成策略需要的真实代价与预估代价之间存在较大差距,导致需要搜索的范围增加、搜索效率降低。

在讨论传统 A* 算法搜索路径时,两个非相邻路径点之间的真实代价 G 是否合理,以间隔 1 个点的情况为例。3 个点分别为节点 n 、其父节点 $n-1$ 和其祖父节点 $n-2$, 节点 n 和节点 $n-2$ 之间传统路径的真实代价 G 的计算方式如式(4)所示。

$$G = \sqrt{(x_n - x_{n-1})^2 + (y_n - y_{n-1})^2} + \sqrt{(x_{n-1} - x_{n-2})^2 + (y_{n-1} - y_{n-2})^2} \quad (4)$$

式中, (x_n, y_n) 为节点 n 的坐标, (x_{n-1}, y_{n-1}) 为节点 $n-1$ 的坐标, (x_{n-2}, y_{n-2}) 为节点 $n-2$ 的坐标。

用 G' 来表示节点 n 和节点 $n-2$ 之间最优路径的真实代价,传统路径真实代价 G 的合理性判定转化为 G 和 G' 是否相等或者近似相等,此时需要分两种情况进行讨论。

第 1 种情况,当 3 个点共线时,如图 3 所示。

节点 n 与节点 $n-2$ 之间最优路径的真实代价 G' 应为两点的欧氏距离,如式(5)所示。

$$G' = \sqrt{(x_n - x_{n-2})^2 + (y_n - y_{n-2})^2} \quad (5)$$

可以看出,此时 $G = G'$,但父节点是多余的路径点,可以消除。

第 2 种情况,当 3 个点不共线时,此时也需要分两种情况进行讨论。

(1) 当节点 n 和节点 $n-2$ 的直连路径在安全距离 S 内存在障碍物,包括穿过障碍物的情况,如图 4 所示,实线为传统路径,虚线为最优路径。

为了让机器人和障碍物保持安全间距,传统路径比较理想,真实代价 G 十分接近最优路径的真实代价 G' ,即 $G \approx G'$ 。

(2) 当两点直连路径安全距离内无障碍物时,如图 5 所示。

可以看出,节点 n 与节点 $n-2$ 的直连路径应为最优路径(虚线所示),最优路径的真实代价 G' 为两点的欧氏距离,如式(5)所示。也就是说,此时传统路径(实线所示)的真实代价 G 明显大于 G' ,这导致传统 A* 算法规划路径的长度不是最优或出现冗余转折。与此同时,预估代价(欧氏距离)与 G 之间存在的较大差距,使得启发函数的启发效果变差,需要搜索的范围增加,搜索效率也因此降低。

上述原因使得传统 A* 算法规划的路径在现实环境下往往不利于机器人的工作,需要进一步优化。

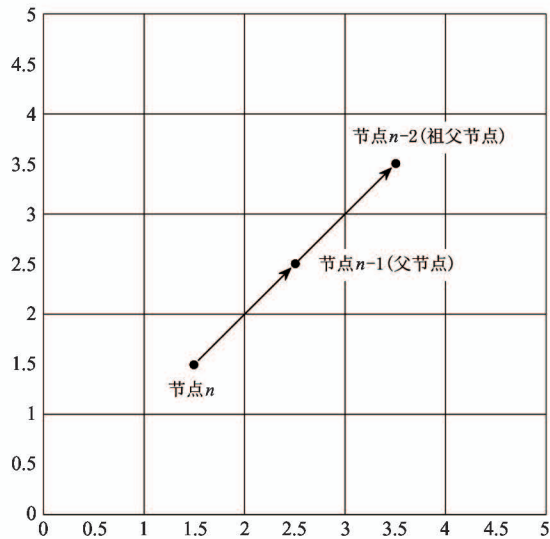
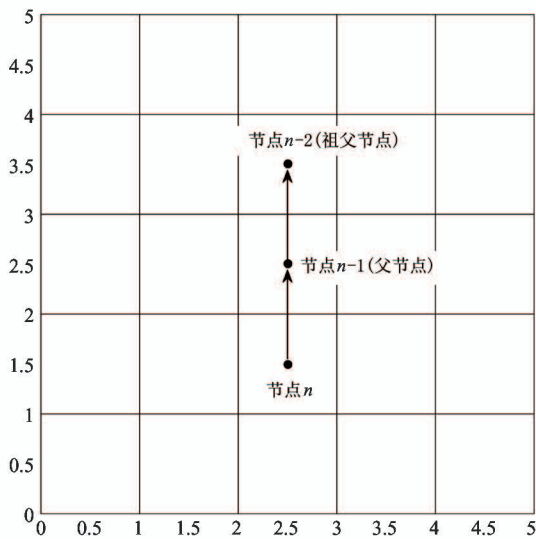


图3 三点共线情形示意图

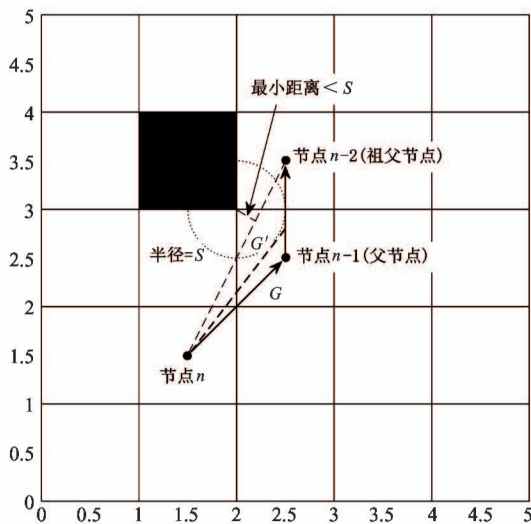


图4 直连路径安全距离内有障碍物

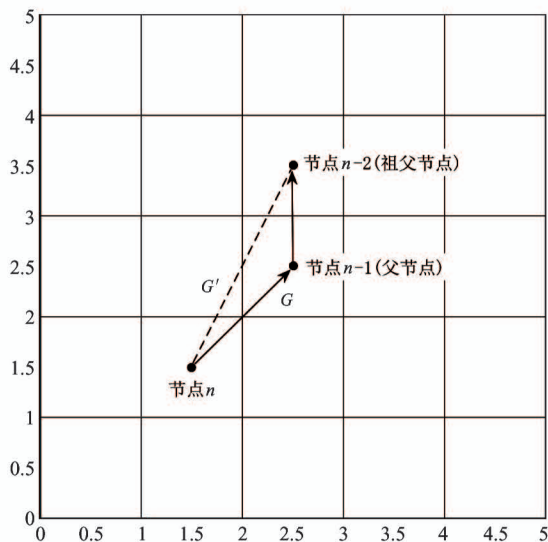


图5 直连路径安全距离内无障碍物

2 改进 A* 算法

针对传统 A* 算法存在的问题,提出改进方案,在扩展节点 m 的邻域节点 n 时,优化 n 的父节点选取策略,父节点的选取不再只考虑节点 m ,而是将范围扩大到 m 至起始点所在支路上的所有节点。

在改进方案中,会尝试将节点 n 与支路上的点直接相连,因此必须判断该直连路径的安全性,判断方法分为以下两步。

(1) 设支路上的点为 p , 建立以节点 n 与节点 p 为端点的线段所在直线的方程,如下式:

$$(y_n - y_p) \cdot x + (x_p - x_n) \cdot y + (x_n \cdot y_p - x_p \cdot y_n) = 0 \quad (6)$$

式中, (x_n, y_n) 为节点 n 的坐标, (x_p, y_p) 为节点 p 的坐标。

(2) 将节点 n 与节点 p 作为对角顶点形成的与坐标轴平行或垂直的矩形视为可能存在障碍物对该线段所表示的路径产生威胁的区域,针对区域内可能存在的障碍物,将其所在栅格的 4 个顶点分别代入式(6)的左侧,若计算得到的值均同号,说明该路径不接触该障碍物;接着,分别计算障碍物栅格 4 个顶点到直线的距离(任意点到直线的距离见式(7)),

若均大于安全值,说明该障碍物对该路径不存在威胁。若矩形区域内无障碍物或者所有障碍物都对该路径不存在威胁,说明该路径是安全的。

$$L = \frac{|(y_n - y_p) \cdot x_0 + (x_p - x_n) \cdot y_0 + (x_n \cdot y_p - x_p \cdot y_n)|}{\sqrt{(y_n - y_p)^2 + (x_p - x_n)^2}} \quad (7)$$

式中, (x_0, y_0) 为任意点的坐标, (x_n, y_n) 为节点 n

的坐标, (x_p, y_p) 为节点 p 的坐标。

2.1 扩展未发现自由点的改进方案

若节点 n 是未发现的自由点,将 n 加入到 OPEN 表,回溯 m 所在支路上的节点(包括 m),必然能找到点 $P1$,使得 n 能够沿直线安全到达 $P1$,且经由 $P1$ 到达起始点所需的真实代价最小,然后将 $P1$ 设置为节点 n 的父节点,并计算出 G 和 F 值。图6、图7为改进示例。

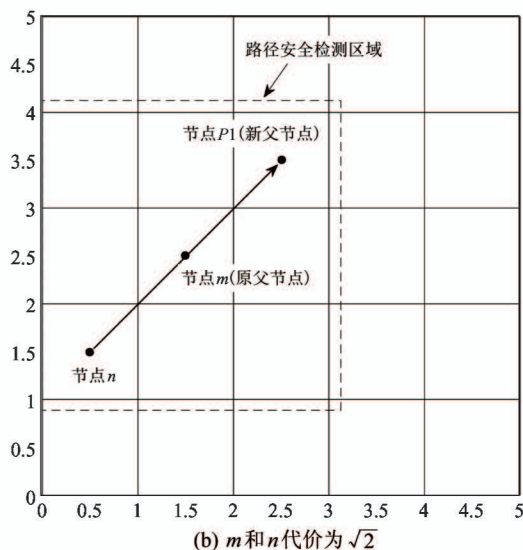
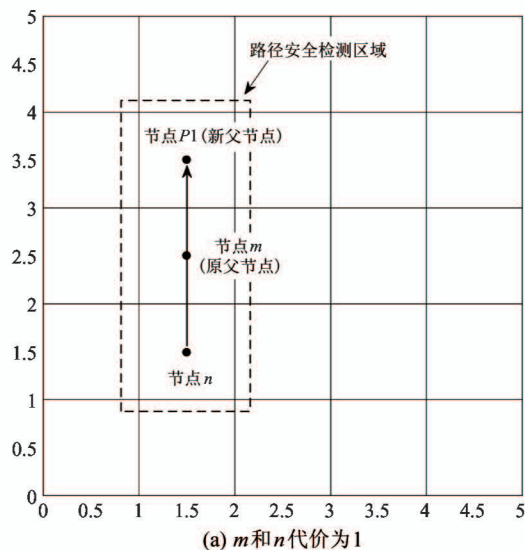


图6 共线时改进方案与传统方案对比

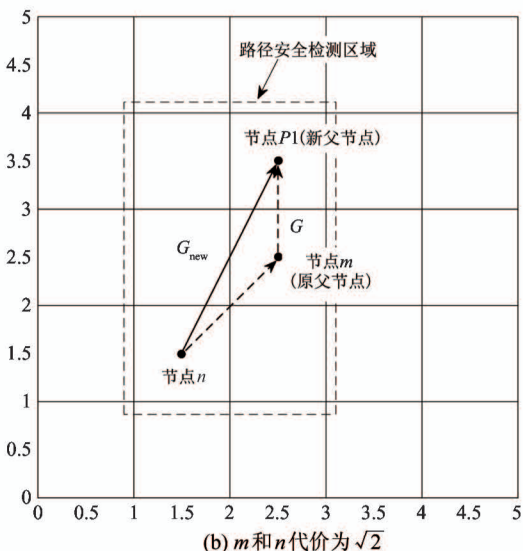
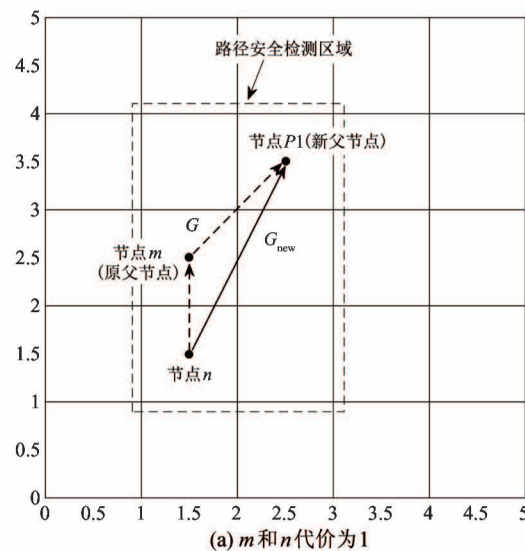


图7 不共线时改进方案与传统方案对比

可以发现,共线时,改进方案与原方案相比,路径代价相同,但舍去了冗余的中间路径点;不共线

时,新方案的路径(实线所示,代价为 G_{new})更短,同时也舍去了冗余的中间路径点。图8所示为扩展未

发现自由点改进方案流程图。

2.2 扩展已发现自由点的改进方案

若节点 n 已在 OPEN 表中,回溯 m 所在支路上的节点(包括 m),必然能找到点 $P2$, 使得 n 能够沿直线安全到达 $P2$, 并且经由 $P2$ 到达起始点所需的真实代价最小。将 n 通过 m 支路所能得到的最小代价记为 G_{new} , 若 G_{new} 小于原来的 G 值,则将 $P2$ 设置为节点 n 新的父节点,并更新 G 和 F 值。图 9 所示为扩展已发现自由点的改进方案的示例。

可以看出,相比传统方案,改进方案规划路径的长度更短,转折次数也更少,图 10 所示为改进方案的流程图。

上述改进方案可以保证已生成的路径为实际的

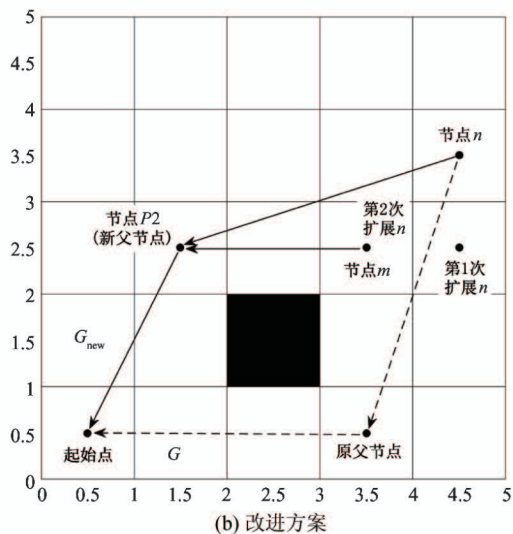
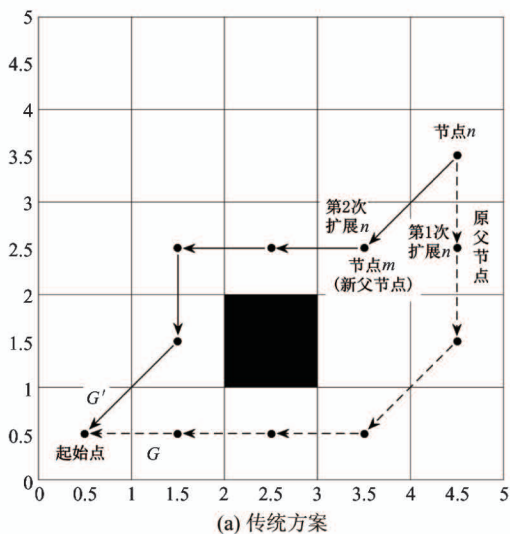


图 9 改进方案与传统方案对比

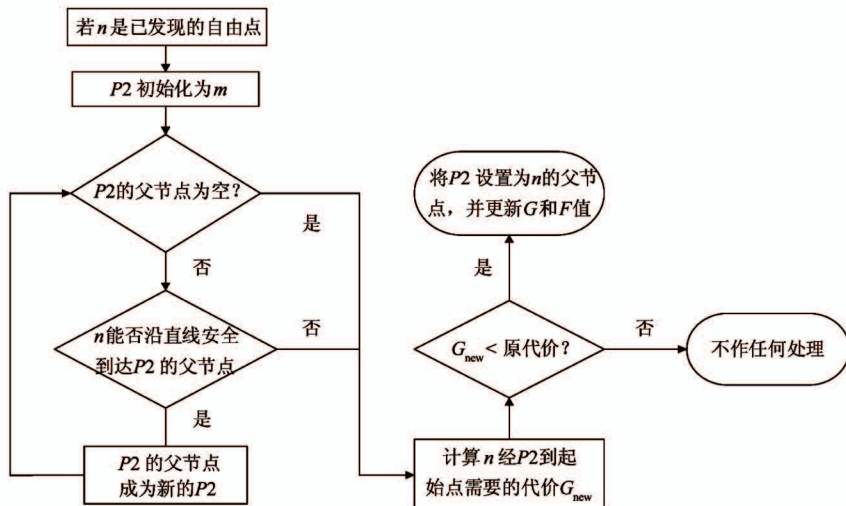


图 10 扩展已发现自由点改进方案流程图

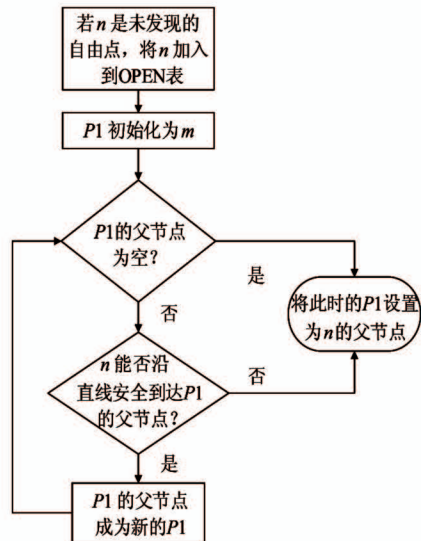


图 8 扩展未发现自由点改进方案流程图

最短路径,并舍去大量的中间路径点,降低预估代价与真实代价之间的偏差,减少搜索范围。因此,改进算

法能够有效优化路径的长度、平滑性以及寻路效率。

改进 A* 算法的总流程图如图 11 所示。

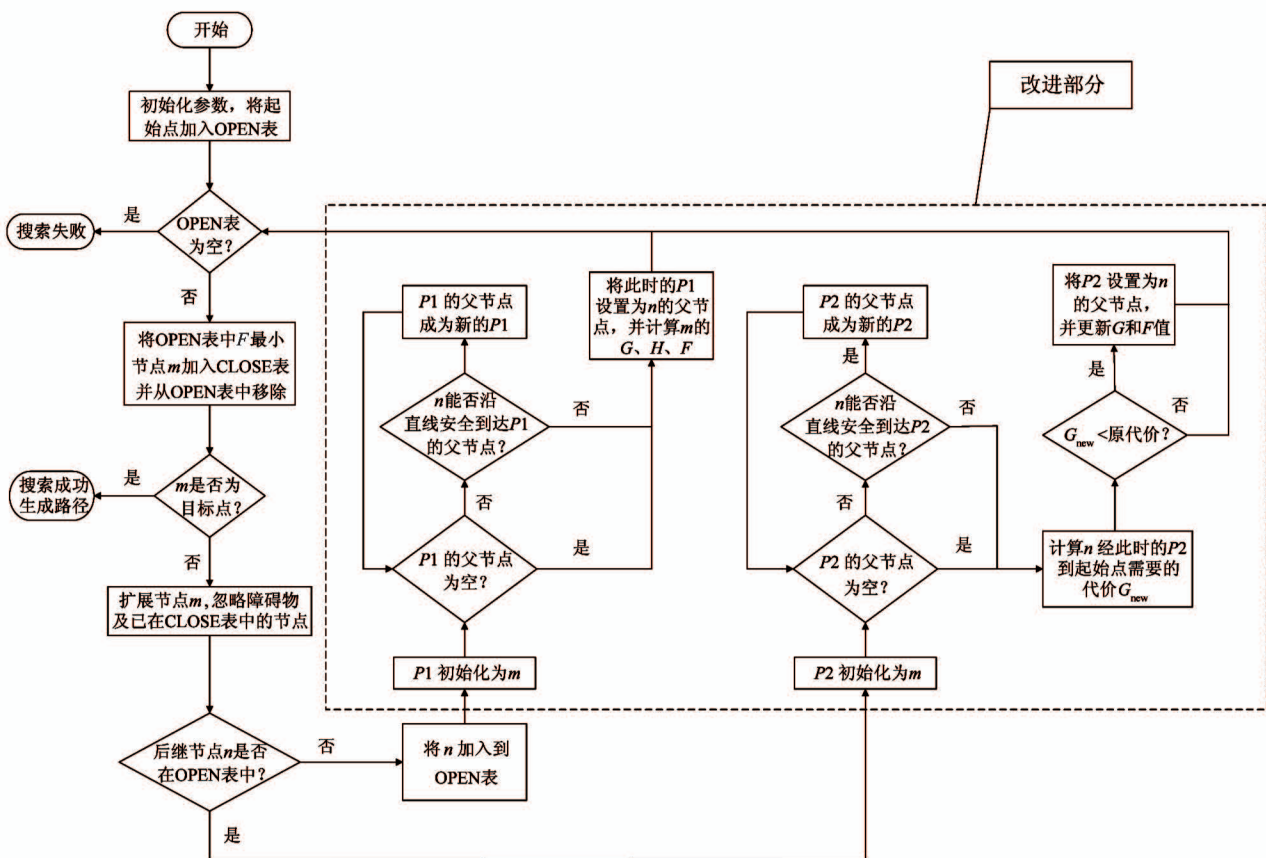


图 11 改进 A* 算法流程图

3 实验及结果

为了验证所提改进算法的可行性和有效性,使用 C++ 编程语言实现了传统 A* 算法和所提改进算法并进行路径规划实验,所用栅格地图的规模为 40 × 40,相关实验数据用 Matlab 软件进行可视化,最后对两种算法规划路径的质量和效率进行定性和定量分析。

3.1 集中障碍物环境

传统 A* 算法和改进 A* 算法规划的路径如图 12 所示,灰色区域为搜索的范围。可以看出,采用传统 A* 算法所规划的路径,每个路径点均处于栅格中心,路径较长且有不少转折点,搜索范围较大;改进算法规划的路径,所有路径点均是转折点,两个相邻转折点之间的路径不再严格经过栅格中

心,路径长度、转折点数量和搜索范围都有所减少。

传统 A* 算法和改进 A* 算法规划路径的长度、路径点数量、转折点数量、搜索范围(节点数)及搜索时间等数据如表 1 所示。

根据表 1 数据,相比传统 A* 算法,改进 A* 算法所规划的路径长度减少约 5.5%,路径点数量减少约 95.6%,转折点数量减少约 83.3%,搜索范围减少约 34.8%,搜索时间减少约 66.0%。

3.2 随机障碍物环境

传统 A* 算法和改进 A* 算法规划的路径如图 13 所示,相关数据如表 2 所示。根据表 2 数据,相比传统 A* 算法,改进 A* 算法所规划的路径长度减少约 4.3%,路径点数量减少约 61.8%,转折点数量减少约 34.4%,搜索范围减少约 17.7%,搜索时间减少约 31.8%。

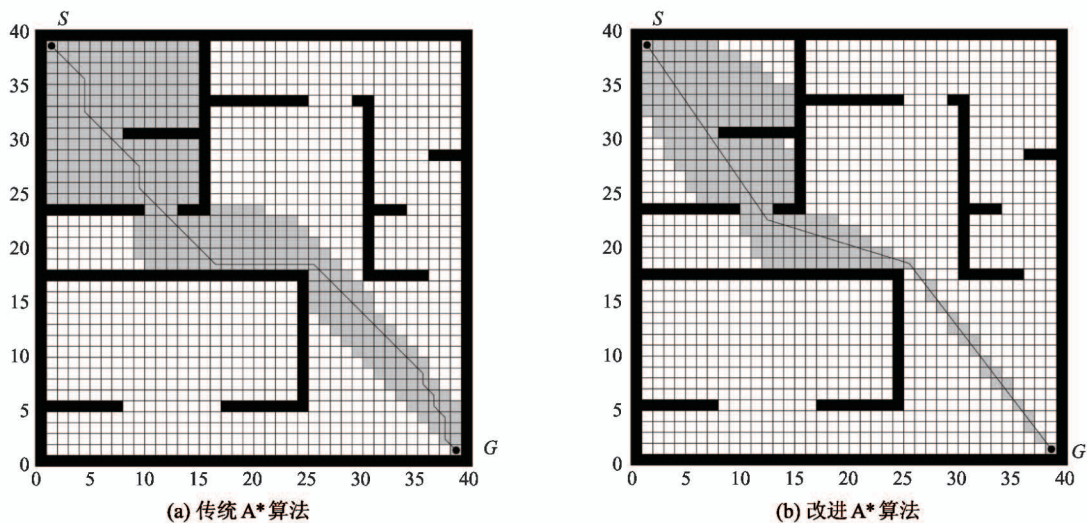


图 12 集中障碍物环境下传统 A* 算法和改进 A* 算法路径对比

表 1 集中障碍物环境下的实验数据

| 算法 | 路径长度 | 路径点数量 | 转折点数量 | 搜索范围 | 搜索时间/s |
|----------|---------|-------|-------|------|--------|
| 传统 A* 算法 | 57.5980 | 45 | 12 | 376 | 0.103 |
| 改进 A* 算法 | 54.4189 | 2 | 2 | 245 | 0.035 |

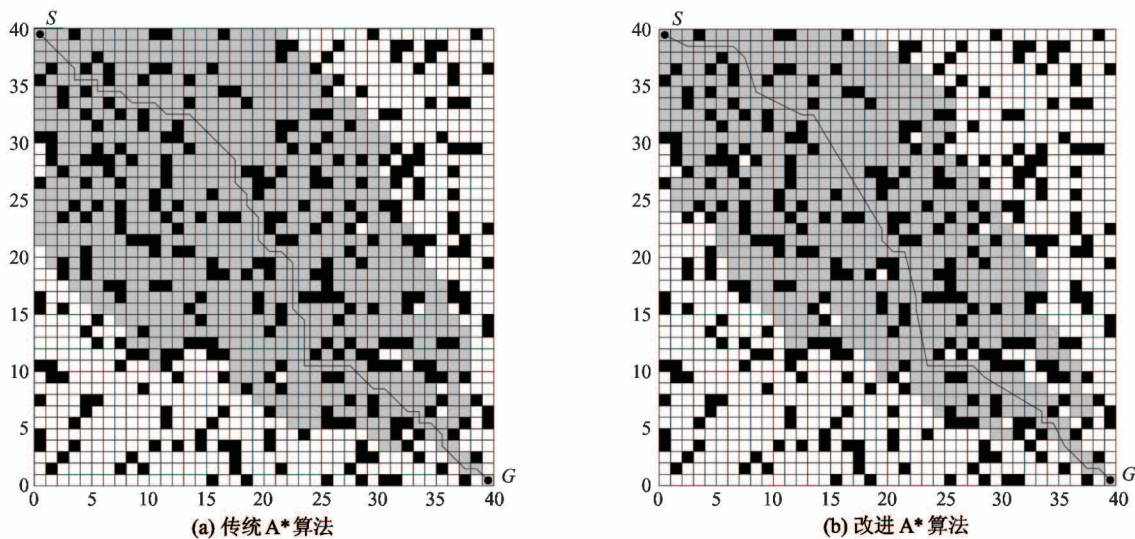


图 13 随机障碍物环境下传统 A* 算法和改进 A* 算法路径对比

表 2 随机障碍物环境下的实验数据

| 算法 | 路径长度 | 路径点数量 | 转折点数量 | 搜索范围 | 搜索时间/s |
|----------|---------|-------|-------|------|--------|
| 传统 A* 算法 | 65.1127 | 55 | 32 | 774 | 0.132 |
| 改进 A* 算法 | 62.3068 | 21 | 21 | 637 | 0.090 |

4 结论

提出改进 A* 算法,采用搜索路径与优化原路径并行的策略,只要满足直连路径安全的条件,任意

非相邻点之间的路径便能够被优化,路径转折角度可以覆盖 $0 \sim 360^\circ$,同时消去大量冗余路径点,只剩下少数关键点,并缩短了路径长度。在提高寻路效率方面,不再从设计新启发函数的角度,而是优化已

生成路径的真实代价,使预估值更接近真实值,增强启发函数的启发能力,减少搜索范围,提高效率。

参考文献

[1] 徐岩,吴立勋,段星光. 轮腿式机器人多运动模式运动学分析与建模[J]. 科学技术与工程,2015,15(36): 67-72
 [2] 卜新苹,苏虎,邹伟,等. 基于复杂环境非均匀建模的蚁群路径规划[J]. 机器人,2016,38(3):276-284
 [3] 陈洋,赵新刚,韩建达. 移动机器人三维路径规划方法综述[J]. 机器人,2010,32(4):568-576
 [4] Lamini C, Benhlima S, Elbekri A. Genetic algorithm based approach for autonomous mobile robot path planning [J]. *Procedia Computer Science*, 2018, 127: 180-189
 [5] 魏彤,龙琛. 基于改进遗传算法的移动机器人路径规划[J]. 北京航空航天大学学报,2020,46(4):703-711
 [6] 代亚兰,熊禾根,陶永,等. hAPF-ACO: 广义障碍环境下的移动机器人路径规划算法[J]. 高技术通讯, 2018,28(1):67-77
 [7] Wang Z Q, Zhu X G, Han Q Y. Mobile robot path planning based on parameter optimization ant colony algorithm [J]. *Procedia Engineering*, 2011, 15: 2738-2741
 [8] 游晓明,刘升,吕金秋. 一种动态搜索策略的蚁群算法及其在机器人路径规划中的应用[J]. 控制与决策, 2017,32(3):552-556
 [9] 陈劲峰,黄卫华,王肖,等. 基于改进蚁群算法的移动机器人路径规划[J]. 高技术通讯,2020,30(3):291-297

[10] 董朝瑞,郭欣,李宁,等. 基于改进 A* 算法的多机器人动态路径规划[J]. 高技术通讯,2020,30(1):71-81
 [11] Saranya C, Unnikrishnan M, Ali S, et al. Terrain based D* algorithm for path planning [J]. *IFAC-Papers On-Line*, 2016, 49(1): 178-182
 [12] 刘军,冯硕,任建华. 移动机器人路径动态规划有向D* 算法[J]. 浙江大学学报(工学版),2020,54(2):1-10
 [13] Song R, Liu Y C, Bucknall R. Smoothed A* algorithm for practical unmanned surface vehicle path planning [J]. *Applied Ocean Research*, 2019, 83: 9-20
 [14] 单伟,孟正大. 基于改进 A* 算法的平滑路径设计[J]. 东南大学学报(自然科学版),2010,40(增1):155-161
 [15] Zhang A, Li C, Bi W H. Rectangle expansion A* path-finding for grid maps [J]. *Chinese Journal of Aeronautics*, 2016, 29(5): 1385-1396
 [16] 赵晓,王铮,黄程侃,等. 基于改进 A* 算法的移动机器人路径规划[J]. 机器人,2018,40(6):903-910
 [17] 吴鹏,桑成军,陆忠华,等. 基于改进 A* 算法的移动机器人路径规划研究[J]. 计算机工程与应用,2019,55(21):227-233
 [18] Grimmer A, Clemens J, Wille R. Formal methods for reasoning and uncertainty reduction in evidential grid maps [J]. *International Journal of Approximate Reasoning*, 2017, 87: 23-39
 [19] 陈若男,文聪聪,彭玲,等. 改进 A* 算法在机器人室内路径规划中的应用[J]. 计算机应用,2019,39(4): 1006-1011

Global optimal path planning for mobile robots based on improved A* algorithm

Li Xiaolu^{***}, Xiong Hegen^{**}, Tao Yong^{***}, Li Gongfa^{***}

(* The Key Laboratory of Metallurgical Equipment and Control of Education Ministry, Wuhan University of Science and Technology, Wuhan 430081)

(** The Key Laboratory of Metallurgical Transmission and Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan 430081)

(*** School of Mechanical Engineering and Automation, Beihang University, Beijing 100191)

Abstract

In a grid map environment, the path points selected by the traditional A* algorithm for searching a path are limited by the grid center, and the turning angle of the path is fixed to some specific discrete values, so there are problems of non-optimal length and many redundant transitions. In this paper, an improved A* algorithm under a new heuristic search strategy is proposed. When exploring each adjacent node of the current node, the selection range of the parent node of the adjacent point is expanded to the entire branch from the current point to the starting point, and adopt the way that the adjacent points are directly connected to the points on the branch road to find the safe path with the smallest real cost G . At this time, the corresponding point on the branch is the parent node of the adjacent point. Experimental results show that the improved algorithm can effectively improve the path quality, that is, the planned path length is shorter and the number of turns is less. At the same time, the deviation between the real cost G and the estimated cost F is reduced, the heuristic ability of the heuristic function is enhanced, and the pathfinding efficiency is also improved.

Key words: improved A* algorithm, mobile robot, path planning, path quality, pathfinding efficiency