

面向受限工业环境的移动检测机器人 全局路径规划方法

吴应¹, 武越¹, 张凯^{2,3}, 孙晓界⁴, 陈思远¹

(1. 西南交通大学 电气工程学院, 成都 610031; 2. 四川大学 空天科学与工程学院, 成都 610065;
3. 机器人卫星实验室, 成都 610065; 4. 河南科技大学 信息工程学院, 河南 洛阳 471023)

摘要: 针对当前全局路径规划方法在受限工业环境中所规划的全局路径存在安全性不足、曲率不连续等问题, 提出一种融合改进 Dijkstra 算法和二次规划的移动检测机器人全局路径规划方法。首先改进 Dijkstra 算法, 在原始评价函数的基础上增加障碍物代价项, 使规划路径远离障碍物, 提出自适应评价函数, 平衡路径长度与路径安全性, 使用改进的 Dijkstra 算法生成从起始位置到目标位置且远离障碍物的引导路径; 然后改进二次规划方法, 对引导路径进行平滑处理, 改进约束条件以保证平滑后路径的安全。仿真实验结果验证了本文算法的有效性; 与 ROS 内置的 Dijkstra 算法及 A* 算法相比, 本文算法能够提升全局路径的平滑性与安全性; 与其他融合算法相比, 本文算法能够在不降低全局路径平滑性的同时提升其安全性。

关键词: 受限工业环境; 移动检测机器人; 全局路径规划; Dijkstra 算法; 二次规划

中图分类号: TP242

文献标志码: A DOI:10.3969/j.issn.1003-1251.2025.03.007

Global Path Planning Method for Mobile Detection Robots in Constrained Industrial Environments

WU Ying¹, WU Yue¹, ZHANG Kai^{2,3}, SUN Xiaojie⁴, CHEN Siyuan¹

(1. Southwest Jiaotong University, Chengdu 610031, China; 2. Sichuan University, Chengdu 610065, China;
3. Robotic Satellite Laboratory, Chengdu 610065, China; 4. Henan University of Science and Technology, Luoyang 471023, China)

Abstract: To address the issues of insecurity and curvature discontinuity of the current global path planning in restricted industrial environments, a global path planning method for mobile detection robots on the basis of improved Dijkstra algorithm and quadratic programming is proposed. Firstly, an improved Dijkstra algorithm is proposed, which adds an obstacle cost term to the original evaluation function to make the planned path avoid obstacles, and introduces an adaptive evaluation function to keep a balance between the path length and safety. This improved Dijkstra algorithm results in a guiding path from the starting position to the target position while keeping a safe distance from obstacles. Then, an improved quadratic programming method is used to smooth the guiding path, with enhanced constraints to ensure the safety of the smoothed path. Simulation results demonstrate the effectiveness of the proposed method. Compared with the built-in Dijkstra and A* algorithms in ROS, the proposed algorithm improves both the smoothness and safety of the global path. Additionally, compared with other fusion algorithms, the proposed method enhances the safety of the global

path without compromising its smoothness.

Key words: restricted industrial environments; mobile detection robots; global path planning; Dijkstra algorithm; quadratic programming

近年来,在工业领域,特别是轨道交通行业,为节省人力并提高检测精度,使用自主移动机器人取代人工检查已成为一种趋势。与工作于宽敞空间的机器人^[1-2]不同,移动检测机器人通常工作于空间狭窄的受限工业环境中,且配备机械臂、三维摄像机等大型设备,质量和尺寸均较大^[3-4]。现有的全局路径规划方法在受限工业环境中所规划的全局路径存在贴近障碍物、路径曲折等问题,难以满足导航需求。如何实现移动检测机器人在受限工业环境中的自主安全导航,生成远离障碍物且平滑的全局路径成为当前研究中面临的挑战。

在受限工业环境中,很难直接生成一个同时满足安全性和平滑性要求的全局路径。比较有效的解决方法^[5-6]是:首先生成初始引导路径,然后通过后端优化等对初始引导路径进行平滑处理。

移动机器人的初始引导路径生成主要分为图搜索和空间采样两类方法。图搜索的经典算法包括 Dijkstra^[7]、A*^[8-9]与 D*^[10]等。Dijkstra 算法规划出的路径具有完备性,但其路径规划时间较长,规划出的路径靠近障碍物且比较曲折;A* 算法在 Dijkstra 算法的基础上增加了启发项,减少了路径规划时间,但仍未解决规划路径靠近障碍物且曲折的问题;D* 算法是一种增量式路径搜索算法,可在环境状态变化时快速更新路径,但该算法的效率和性能在很大程度上依赖于所选择的启发式函数和路径更新策略。尽管上述经典图搜索方法在大部分场景中能够规划出安全的路径,但其依赖于较大膨胀半径的设置,而在以狭窄空间为代表的受限工业环境中,设置较大的膨胀半径会导致无法规划出路径,设置较小的膨胀半径则会导致规划的路径不具备安全性。基于空间采样的经典算法主要是快速探索随机树(RRT)算法^[11-12],该类算法存在随机性强、导向性差、规划路径时间长等不足,适用于解决高维、多约束的路径优化问题,在二维狭窄空间难以发挥其优势。

在得到初始引导路径后,通常使用拟合、插值、优化等后端处理方法对其进行平滑处理,以满足路径的平滑性要求。采用多项式插值实现路径平滑是经典的基于插值的平滑方案^[13],平滑后的路径曲线经过所有路径关键点,但在狭窄空间内,

使用高阶多项式进行插值可能会引起过拟合现象,即路径可能会在局部波动较大,导致运动不稳定或规划出不符合实际情况的路径;采用贝塞尔曲线实现路径平滑是经典的基于拟合的平滑方案^[14],其通过调整控制点的位置实现局部曲线调整,在狭窄空间内适应路径变化的能力较强,但贝塞尔曲线的变化主要依赖于控制点的位置,可能导致局部路径优化但非全局最优,不宜用于需要考虑全局路径优化的情况;梯度下降法^[15]与二次规划^[16]等是经典的基于优化的平滑方案,采用梯度下降的平滑方法时,如初值设置不当,可能出现梯度爆炸与梯度消失,导致最终路径的平滑性不满足要求,基于二次规划的平滑方法既能保证全局最优,也不会出现梯度爆炸与梯度消失等问题,能够在受限环境中生成曲率连续的平滑路径,但保证优化后路径的安全性是尚需解决的问题。

为解决上述问题,本文提出一种面向受限工业环境的移动检测机器人全局路径规划方法,将全局路径规划过程分为引导路径生成与路径平滑处理两部分。首先对 Dijkstra 算法进行改进,在原始评价函数的基础上增加障碍物代价项,使规划路径远离障碍物,进一步提出自适应评价函数,以平衡路径长度与路径安全性,使用改进后的 Dijkstra 算法生成从起始位置到目标位置且远离障碍物的引导路径;然后改进二次规划方法,对引导路径进行平滑处理,改进二次规划约束条件,以保证平滑后路径的安全性。

1 改进 Dijkstra 算法

传统 Dijkstra 算法在路径规划中的评价函数仅考虑起始节点与当前节点之间的实际代价,通常采用起始节点与当前节点之间的距离进行计算,其数学描述为

$$F(i) = G(i) \quad (1)$$

式中: $F(i)$ 表示从初始节点经由节点*i*到目标节点的估计代价; $G(i)$ 表示从初始节点到节点*i*的实际代价。

传统 Dijkstra 算法通过设置障碍物的膨胀半径保证路径的安全,但在受限工业环境中,即使设置了适中的膨胀半径,也可能导致规划出的路径

离障碍物较远时,其可变化范围增大,当路径点距离障碍物较近时,其可变化范围减小。路径平滑效果如图1所示,图中圆点为初始引导路径点,方框为路径点的可变化范围。路径点在方框内变化可保证优化后的路径安全,优化后得到的平滑且安全的全局路径如图中曲线示意。

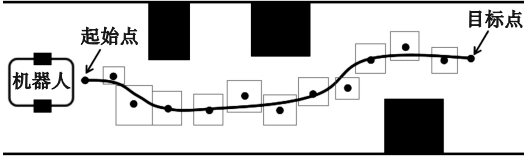


图1 路径平滑效果图

Fig.1 Smoothed path visualization

为生成上述平滑且安全的规划路径,在式(12)的基础上增加路径点的变化范围约束,即

$$\begin{cases} \min X^T(w_1A_1^T A_1 + w_2A_2^T A_2 + w_3I)X + w_3f^T X \\ \text{s. t. } D_{\min_obs} - D_{\text{safety}} \leq X - X_r \leq D_{\min_obs} + D_{\text{safety}} \\ x_1 = x_{1r}, y_1 = y_{1r} \\ x_n = x_{nr}, y_n = y_{nr} \end{cases} \quad (13)$$

其中

$$X_r = [x_{1r}, y_{1r}, x_{2r}, y_{2r}, \dots, x_{nr}, y_{nr}]^T \quad (14)$$

$$D_{\min_obs} = [d_{\min_obs}(1), \dots, d_{\min_obs}(n)]^T \quad (15)$$

$$D_{\text{safety}} = [d_{\text{safety}}, d_{\text{safety}}, \dots, d_{\text{safety}}]^T \quad (16)$$

3 算法融合

本文将改进的Dijkstra算法与二次规划方法相融合,生成从起始位置到目标位置的平滑且安全的全局路径。改进Dijkstra算法:在原始评价函数的基础上增加障碍物代价项,使规划路径远离障碍物;提出自适应评价函数,以平衡路径长度与路径安全性。改进二次规划方法:对二次规划方法中的约束条件进行优化,确保在对引导路径进行平滑处理的过程中,路径的安全性得到有效保障。

本文算法的具体流程如图2所示。图2中:开放列表和关闭列表是用于管理节点状态的数据结构,前者存储当前待处理的节点,后者记录已处理的节点;邻居节点是与当前节点相邻的上下左右四个节点,算法会评估这些节点以决定下一步的搜索方向;父节点表示某个节点的前驱节点,通过追踪父节点可以重建从起始节点到目标节点的完整路径;最小代价策略则指导算法在选择节点时优先考虑代价最低的节点,以确保搜索时尽量

沿着最优路径前进。通过上述设计,算法能够高效地执行路径搜索,最终找到从起始节点到目标节点的最优路径。

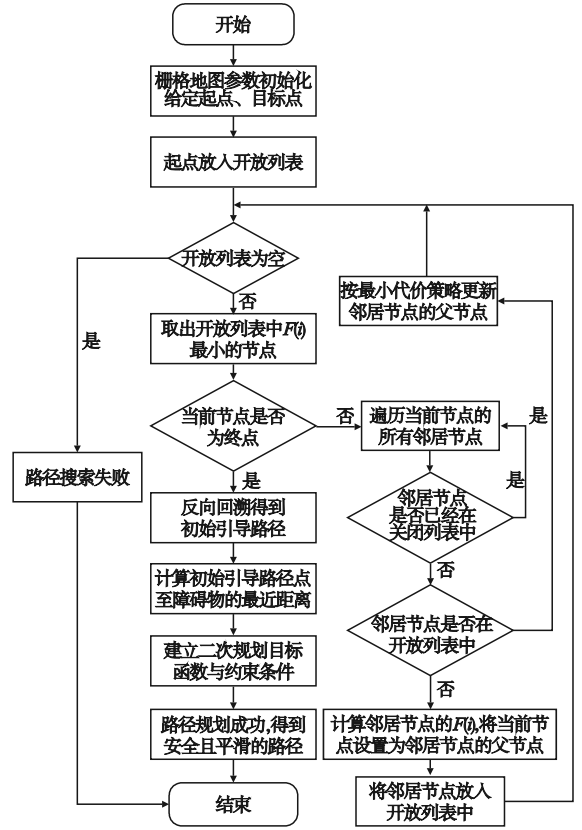


图2 算法流程图

Fig.2 Algorithm flowchart

4 仿真实验与结果分析

通过设计仿真实验验证本文算法的有效性和实用性。实验计算平台中央处理器为Intel(R) Core(TM) i5-7300HQ CPU @ 2.50 GHz,软件环境为Ubuntu20.04,并基于ROS系统进行开发。

测试地图为包含直角转弯、狭窄通道及复杂障碍物群等场景的受限工业环境。地图分辨率为0.02 m,机器人长为0.6 m、宽为0.5 m。采用ROS内置的Dijkstra算法、ROS内置的A*算法、文献[16]中基于Dijkstra算法与二次规划方法的融合算法及本文算法进行对比实验。

实验参数设置:依据机器人的几何尺寸与膨胀半径的含义,ROS内置的Dijkstra算法和ROS内置的A*算法中膨胀半径设置为0.33 m;文献[16]中融合算法参数设置详见原文献;本文改进Dijkstra算法参数设置为 $w_g(i) = 1$ 、 $w_h(i) = 4$ 、 $a = 2$ 、 $d_{\text{safety}} = 0.35$ m、 $p = 0.2$ 、 $q = 0.2$ 、 $c = 3$,平滑

算法参数设置为 $w_1 = 1 \times 10^4$ 、 $w_2 = 0$ 、 $w_3 = 0$ 、 $d_{\text{safety}} = 0.33 \text{ m}$ 。

4.1 ROS 内置 Dijkstra 算法及 A* 算法实验结果

采用 ROS 内置 Dijkstra 算法与 A* 算法进行路径规划时,需设置膨胀半径,但在受限环境下,如膨胀半径较大,会因空间过于狭窄而无法规划出路径,如膨胀半径较小,会因规划出的路径紧贴障碍物而导致机器人沿该路径行驶时发生碰撞。设置适中的膨胀半径,分别采用 ROS 内置 Dijkstra 算法及 A* 算法进行全局路径规划,实验结果如图 3 和图 4 所示。可见,即使设置了适中的膨胀半径,Dijkstra 算法与 A* 算法规划出的全局路径仍非常曲折,且靠近障碍物。路径曲折会导致机器人行驶出现摆动,路径贴近障碍物会导致机器人行驶的安全性降低。

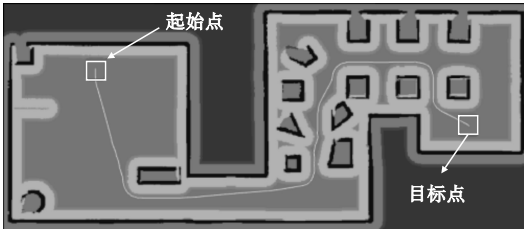


图 3 ROS 内置 Dijkstra 算法的实验结果
Fig. 3 Experimental results of built-in Dijkstra algorithm in ROS

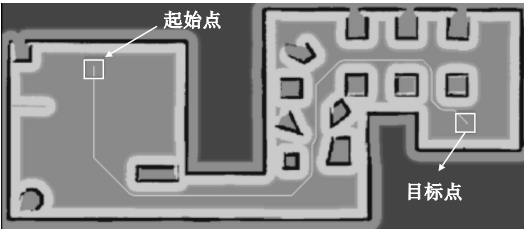


图 4 ROS 内置 A* 算法的实验结果
Fig. 4 Experimental results of built-in A* algorithm in ROS

4.2 文献[16]算法的实验结果

采用文献[16]中提出的算法进行全局路径规划,实验结果如图 5 所示。可见,该算法不能保证在狭窄空间中生成平滑且安全的全局路径。由于二次规划平滑过程中未考虑路径安全的约束条件,亦即平滑后路径点至最近障碍物的距离未施加最小值约束,因而导致机器人行驶至某些路径点时发生碰撞。尽管文中提出可通过梯度下降法对碰撞路段进行调整,但该方法需要设置适当的初值,否则会出现梯度消失或梯度下降的问题。

本文实验中出现了梯度消失现象,无法对碰撞路径进行调整,导致整个全局路径不满足安全性。

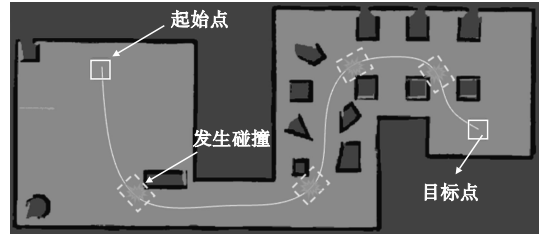


图 5 文献[16]算法的实验结果
Fig. 5 Experimental results of the algorithm in literature [16]

4.3 本文算法的实验结果

采用本文算法进行全局路径规划,实验结果如图 6 所示。由图 6 可知,本文算法无需设置膨胀半径即能生成远离障碍物且平滑的全局路径,满足移动巡检机器人对全局路径平滑性与安全性的要求。改进的 Dijkstra 算法能够在复杂的工业环境中生成兼顾路径长度与路径安全的全局路径;改进的二次规划方法对引导路径进行平滑处理的效果良好,且二次规划问题的约束条件保证了平滑后路径的安全。

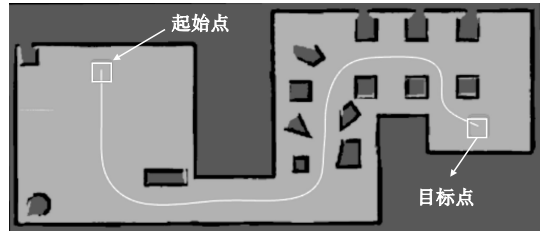


图 6 本文算法的实验结果
Fig. 6 Experimental results of the proposed algorithm

4.4 对比实验分析

从路径曲率、路径点至最近障碍物的距离等角度对实验结果进行分析,路径曲率的连续性反映路径的平滑性,路径点至最近障碍物的距离反映路径的安全性。为保证实验结果准确可靠,对四种不同算法各重复实验 5 次,实验结果显示所有实验均规划成功,将每种算法的 5 次实验数据取平均值作为最终结果,得到全局路径规划的路径曲率与路径点至最近障碍物的距离如图 7 和图 8 所示,实验数据对比如表 1 所示。

由表 1 可见:与 ROS 内置 Dijkstra 算法及 A* 算法相比,本文算法能够极大降低全局路径的平均曲率,提高全局路径中路径点至最近障碍物的平均距离,进而提升全局路径的平滑性与安全性,且几乎不会增加路径的长度;与文献[16]中的融

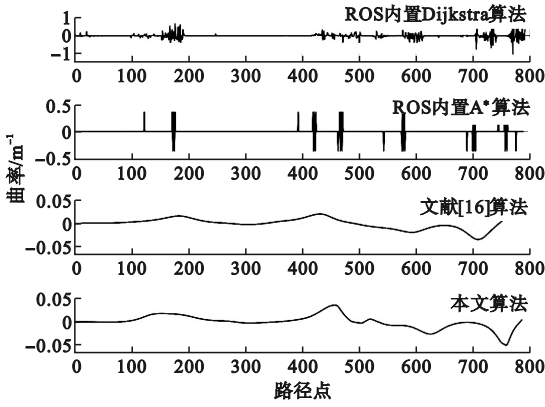


图7 路径曲率

Fig.7 Path curvature

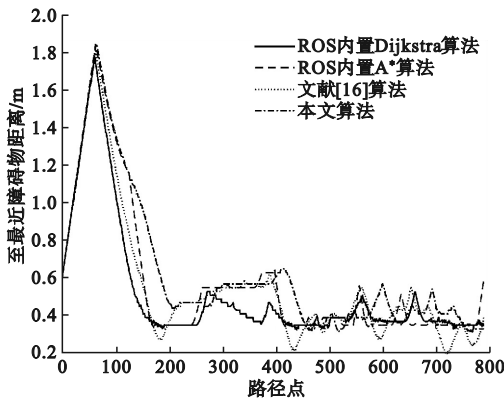


图8 路径点至最近障碍物距离

Fig.8 Distance from path point to the nearest obstacle

表1 实验数据对比

Table 1 Comparison of experimental results

算法	平均曲率/ m^{-1}	至最近障碍物的平均距离/m	路径长度/m	求解时间/s
Dijkstra 算法	0.048	0.53	16.72	0.03
A* 算法	0.017	0.59	17.25	0.02
文献[16]算法	0.010	0.56	16.42	0.09
本文算法	0.011	0.63	17.15	0.08

合算法相比,本文算法在几乎不降低全局路径平均曲率的同时提高了全局路径中路径点至最近障碍物的平均距离,从而在保证路径平滑性的同时提高了安全性。尽管本文算法的规划时间是 ROS 内置 Dijkstra 算法或 A* 算法的 3~4 倍,但完全能够满足受限工业环境下低速巡检机器人的全局路径实时更新频率要求。

5 结论

提出了一种面向受限工业环境的移动检测机

器人全局路径规划方法。改进了 Dijkstra 算法,在原始评价函数的基础上增加了障碍物代价项,使得所规划路径远离障碍物,进一步提出自适应评价函数,平衡了路径长度与路径安全性;改进了二次规划方法,对引导路径进行了平滑处理,改进约束条件,保证了平滑后路径的安全。仿真实验结果验证了本文算法的有效性。

本文提出的算法在求解时间方面略显不足,未来的工作中将重点解决该问题,以扩大算法的使用场景。

参考文献 (References):

- [1] HU K, GU C J, CHEN J M. LTrack: a LoRa-based indoor tracking system for mobile robots[J]. IEEE Transactions on Vehicular Technology, 2022, 71(4): 4264-4276.
- [2] DAY A, KARAMOZAS I. A study in zucker: insights on interactions between humans and small service robots[J]. IEEE Robotics and Automation Letters, 2024, 9(3): 2471-2478.
- [3] NAAZARE M, ROSAS F G, SCHULZ D. Online next-best-view planner for 3D-exploration and inspection with a mobile manipulator robot[J]. IEEE Robotics and Automation Letters, 2022, 7(2): 3779-3786.
- [4] JANG K, KIM S, PARK J. Motion planning of mobile manipulator for navigation including door traversal[J]. IEEE Robotics and Automation Letters, 2023, 8(7): 4147-4154.
- [5] JIAN Z Q, ZHANG S Y, CHEN S T, et al. A global-local coupling two-stage path planning method for mobile robots[J]. IEEE Robotics and Automation Letters, 2021, 6(3): 5349-5356.
- [6] LI X H, SUN Z P, CAO D P, et al. Real-time trajectory planning for autonomous urban driving: framework, algorithms, and verifications[J]. IEEE/ASME Transactions on Mechatronics, 2016, 21(2): 740-753.
- [7] BUZACHIS A, CELESTI A, GALLETTA A, et al. Evaluating an application aware distributed Dijkstra shortest path algorithm in hybrid cloud/edge environments[J]. IEEE Transactions on Sustainable Computing, 2022, 7(2): 289-298.
- [8] GAN X L, HUO Z H, LI W. DP-A*: for path planning of UGV and contactless delivery[J]. IEEE Transactions on Intelligent Transportation Systems, 2023, 25(1): 907-919.
- [9] LIU X A, MA D L, YANG M Q, et al. Modified block A* path-planning method for hybrid-driven underwater gliders[J]. IEEE Journal of Oceanic Engineering, 2022, 47(1): 20-31.
- [10] REN Z Q, RATHINAM S, LIKHACHEV M, et al. Multi-objective path-based D* lite[J]. IEEE Robotics and Automation Letters, 2022, 7(2): 3318-3325.
- [11] 李玮炜, 张文波, 张林丛. 基于采样空间约束的改进 RRT 算法[J]. 沈阳理工大学学报, 2023, 42(1): 13-18, 27. LI W W, ZHANG W B, ZHANG L C. Improvement of RRT algorithm based on sampling space constraint[J]. Journal of Shenyang Ligong University, 2023, 42(1): 13-18, 27. (in Chinese)
- [12] CHEN L, SHAN Y X, TIAN W, et al. A fast and efficient double-tree RRT*-like sampling-based planner applying on mobile robotic systems[J]. IEEE/ASME Transactions on Mechatronics, 2018, 23(6): 2568-2578.

(下转第 59 页)