

文章编号: 2095-2163(2022)12-0055-07

中图分类号: TP242

文献标志码: A

建筑机器人融合改进的 A* 与 TEB 算法的运动规划研究

徐定明, 李子信, 张怡俊

(上海工程技术大学 机械与汽车工程学院, 上海 201620)

摘要: 针对建筑环境移动操作机器人运动规划问题, 提出改进的 A* 算法融合 TEB (Timed Elastic Band) 算法进行机器人的运动控制, 完成建筑机器人近距离停靠在平行于打孔位置墙面的任务。从仿真以及实验可以看出, 相比于传统 A* 算法, 改进 A* 算法时效更高, 路径更平滑, 便于建筑机器人生成较好的全局路径轨迹。融合 TEB 算法实时求解以一定的频率进行局部路径规划, 机器人运动过程中激光雷达负责定位以及获得障碍物信息, 高效完成运动规划任务。并提出了进一步改进措施, 可以优化建图定位参数, 也可继续改进全局与局部路径规划算法, 补偿机器人运动误差, 以达到更好的导航效果。

关键词: 改进 A* 算法; TEB 算法; 建筑机器人; 运动规划

Research on motion planning of construction robots based on improved A* and TEB algorithm

XU Dingming, LI Zixin, ZHANG Yijun

(School of Mechanical and Automotive Engineering, Shanghai University of Engineering Science, Shanghai 201620, China)

[Abstract] Aiming at the motion planning problem of mobile operation robots in building environment, an improved A* algorithm combined with the Timed Elastic Band (TEB) algorithm is proposed to control the robot's motion, and the construction robot could dock in a close distance on the wall parallel to the drilling position. It can be verified from the simulation and experiments that compared with the traditional A* algorithm, the improved A* algorithm has higher time-effectiveness and smoother path, which is convenient for the construction robot to generate better global path trajectory. TEB algorithm is integrated to solve the local path planning at a certain frequency in real time. Lidar is responsible for locating and obtaining obstacle information during the robot movement, thus completing the motion planning task efficiently. Furtherly, improvement measures are proposed, which can optimize the mapping location parameters, and improve the global and local path planning algorithms to compensate the robot motion errors, so as to achieve better navigation effect.

[Key words] improved A* algorithm; TEB algorithm; construction robots; motion planning

0 引言

目前, 建筑业正积极推行智能化、对象集约化等新型建造方式, 智能建造环境下移动作业机器人就是该领域中较好的解决方案之一, 涉及到避障、路径规划、定位、导航、传感和通信等关键技术。其中, 运动控制和路径规划问题被认为是复杂的任务^[1]。

Zhu 等人^[2]提出了基于深度学习的方法和导航框架。张贺等人^[3]提出一种基于速度斥力场改进人工势场方法的全向移动机器人自主动态避障方法, 并验证了自主动态避障算法的有效性和实用性。胡杰等人^[4]改进了传统人工势场法在路径规划中存在的缺陷, 将引力函数分段, 避免引力过大而碰到障碍物的情况。Kashyap 等人^[5]概述并实现一种新型的仿人机器人混合控制器来绘制最优路径, 采用人工势场和飞蛾火焰优化混合方法对仿人机器人进行导航。刘子毅等人^[6]提出了一种应用 BIM

(Building Information Model) 信息与激光雷达获取的现实环境信息进行导航并保证路径最优的方法。Agarwal 等人^[7]基于黏菌的改进群智能算法在 Matlab 2020a 验证出在生成最佳无碰撞路径上花费的时间和精力更少。李承远^[8]对 TEB 算法进行改进, 提出了减少优化变量的数量, 使用参数三次样条插值的方法将路径点插值成一条平滑的路径曲线的方法, 有效地降低了动态环境下 TEB 算法的计算成本, 解决了算法实时性能较低的问题。Mugarza 等人^[9]提出了一种基于改进 D* 的交通控制器, 用于动态确定 AGV 的无碰撞轨迹。劳彩莲等人^[10]根据温室环境下移动机器人作业的实时路径规划要求, 提出一种基于改进 A* 算法与动态窗口法相结合的温室机器人路径规划算法, 跟踪误差保持在 0.22 m 以内、定位误差不大于 0.28 m, 能够满足实际需求。柴红杰等人^[11]在障碍物边缘放置虚拟障碍物形成缓冲区来改进 A* 算法, 减少机器人碰撞几率, 提高

作者简介: 徐定明(1997-), 男, 硕士研究生, 主要研究方向: 移动操作机器人。

收稿日期: 2022-02-28

路径规划效率和机器人稳定性。Qi 等人^[12]提出的 MOD-RRT* 可以生成更高质量的初始路径。Lyu 等人^[13]提出了一种新的基于图的机器人路径优化算法,其中关键是动态分配权值的 Floyd 算法,并证明该方法的有效性和鲁棒性。

综上,本文通过减少扩建节点方向改变和次数,以及根据障碍物占比率和复杂程度优化启发式函数。改进的 A* 算法融合 TEB 算法进行建筑机器人的运动控制,高效完成建筑机器人近距离停靠在平行于打孔位置的墙面的任务,验证可行性。

1 传统 A* 算法与 TEB 算法

1.1 传统 A* 算法

传统 A* 算法流程如图 1 所示。图 1 中,栅格化环境地图的节点信息全部加入到 *queue* 队列中,初始化所有的节点信息。分别设置起始点到当前点的真实代价值 $G(n)$ 和当前点到目标点的估算代价值 $H(n)$,随后进行小顶堆排序,得到总代价值 $F(n)$ 最小节点坐标的索引。判断当前索引值是否等于目标坐标点的索引值,如果相等,到达终点,可以结束迭代求解。如果不等,继续求解当前节点相邻 4 个方向节点的 $F(n)$ 值,更新容器的节点代价值,排序找到最小值循环求解,直至到达目标点为止。在此基础上根据已经扩展好的节点值,从目标点反向依次找到 $F(n)$ 最小的值,也就找到了一条新路径。

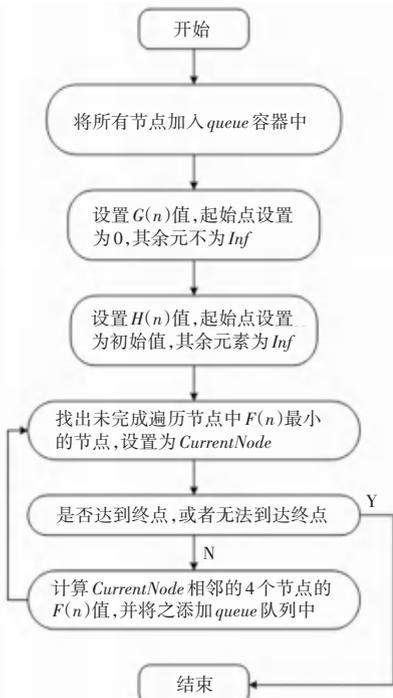


图 1 A* 算法流程图

Fig. 1 Flowchart of A* algorithm

1.2 局部路径规划 TEB 算法

A* 算法生成全局路径,TEB 路径规划期接收全局路径,同时激光雷达负责定位以及获得障碍物信息,以一定的频率进行局部路径规划,计算控制指令并下发给自动驾驶机器人下位机控制器。

1.2.1 机器人运动模型

图 2 为机器人运动学模型。由图 2 可知,6 个轮中 2 个驱动轮分别位于两侧中心处, (X_w, Y_w, Z_w) 为全局坐标系, (X_r, Y_r, Z_r) 为机器人基坐标系,坐标中心点在两驱动轮连线的中心轴线中点处,机器人前轮虚拟转向角等于中心转向角 φ , $(\dot{x}(t), \dot{y}(t), \dot{\varphi}(t))^T$ 为机器人姿态变化值,可得其求解方式如下所示:

$$\begin{pmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\varphi}(t) \end{pmatrix} = \begin{pmatrix} \cos(\varphi(t)) & 0 \\ \sin(\varphi(t)) & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \dot{w}(t) \\ \dot{\omega}(t) \end{pmatrix} \quad (1)$$

$$\begin{pmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\varphi}(t) \end{pmatrix} = \begin{pmatrix} r & r \\ -\frac{r}{L} & \frac{r}{L} \end{pmatrix} \begin{pmatrix} \dot{\omega}_L(t) \\ \dot{\omega}_R(t) \end{pmatrix} \quad (2)$$

从上式得出,给定机器人输入 $u = (v(t), \omega(t))^T$,即可得到机器人姿态变化以及求解出左右驱动轮编码器的控制指令 $\omega_L(t)$ 和 $\omega_R(t)$,从而驱动小车到达目标位置。

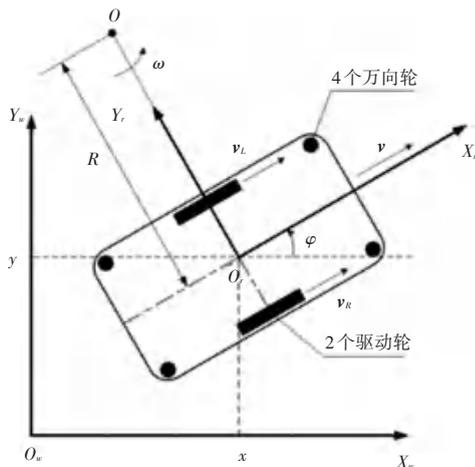


图 2 机器人运动学模型

Fig. 2 Kinematic model of the robot

1.2.2 TEB 轨迹模型

TEB 模型示意如图 3 所示。由图 3 可知,世界坐标系下每个机器人的位姿可表示为 $S_i = [x_i, y_i, \varphi_i]$,一系列的位姿点组成了弹性带 C ,并可由下式进行数学描述:

$$C = \{S_i\}, i = 0, 1, \dots, n, n \in N \quad (3)$$

位姿 S_i 到 S_{i+1} 的时间为 ΔT_i , 其数学表述为:

$$\Delta \tau = \{\Delta T_i\}, i = 0, 1, \dots, n - 1 \quad (4)$$

机器人从位姿 S_i 经过时间 ΔT_i 运动到 S_{i+1} 便组成了时间弹性带, 详见图 3。

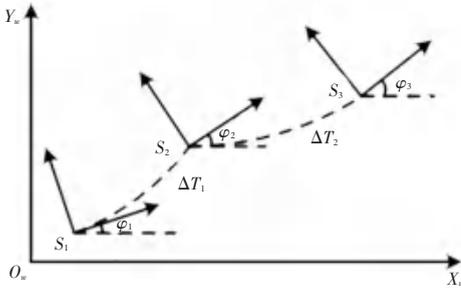


图 3 TEB 模型示意图

Fig. 3 Schematic diagram of TEB model

为了优化运动轨迹, 假设轨迹信息 $T: = (C, \Delta \tau)$, 对轨迹加一定约束, 由两大部分组成。第一部分包括速度、角速度、线加速度、角加速度的约束以及自身运动学的限制; 第二部分则是与障碍物的距离以及与全局路径的偏离程度。 S_i 处的速度约束求解方程分别见式(1):

$$v_i \approx \frac{1}{\Delta T_i} \left\| \begin{matrix} x_{i+1} - x_i \\ y_{i+1} - y_i \end{matrix} \right\| \quad (5)$$

$$\omega_i \approx \frac{\varphi_{i+1} - \varphi_i}{\Delta T_i} \quad (6)$$

加速度相应的约束是由 2 个连续速度的 3 个位姿求解而来, 可表示为:

$$a_i \approx \frac{2 \times (v_{i+1} - v_i)}{\Delta T_{i+1} + \Delta T_i} \quad (7)$$

$$\alpha_i \approx \frac{2 \times (w_{i+1} - w_i)}{\Delta T_{i+1} + \Delta T_i} \quad (8)$$

非完整约束的差分驱动下相邻的 2 个机器人位姿可以看成弧线运动, 如图 4 所示, S_i 与 S_{i+1} 在曲率恒定的圆弧上。

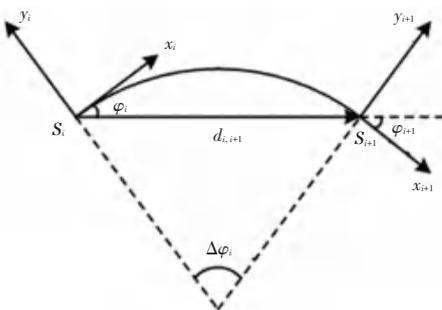


图 4 差分驱动求解图

Fig. 4 Differential drive solution diagram

图 4 中, $[x_i, y_i, \varphi_i]$ 、 $[x_{i+1}, y_{i+1}, \varphi_{i+1}]$ 是位姿; $d_{i,i+1}$ 是圆的弦长; φ_i 与 φ_{i+1} 分别是全局坐标系下的转角。由此推出运动学约束的代价函数为:

$$f_{nh}(S_i, S_{i+1}) = \left\| \begin{matrix} \frac{e}{c} \cos \varphi_i \ddot{\varphi} & \frac{e}{c} \cos \varphi_{i+1} \ddot{\varphi} \\ \frac{e}{c} \sin \varphi_i \ddot{\varphi} & \frac{e}{c} \sin \varphi_{i+1} \ddot{\varphi} \\ 0 & 0 \end{matrix} \right\| \times d_{i,i+1} \quad (9)$$

枫智机器人的转弯半径约束在最小半径之上, 此处需用到的公式为:

$$r_i = \left| \frac{v_i}{\omega_i} \right| \geq \frac{\|d_{i,i+1}\|_2}{\left| 2 \sin \frac{\Delta \varphi_i}{2} \right|} = r_{min} \quad (10)$$

此外, 还有障碍物代价函数、时间最优的目标函数等在机器人行走过程中起约束作用, 不再赘述。

1.2.3 算法求解

算法求解流程如图 5 所示, 对全局路径 $\{Z_j\}$ 做离散化处理, 成为带有时间信息的位姿序列 $T(C, \tau)$, 每次迭代删除求解过的位姿和插入新的状态点。建立起 $T(C, \tau)$ 与目标函数以及约束函数之间的关系, 求解最优轨迹 T^* , 通过最小化一个非线性最小二乘代价函数求得:

$$T^* \approx \arg \min_{T \in \{S_1, S_n\}} \sum_k \sigma_k f_k(T) \quad k \in \{T, P\} \quad (11)$$

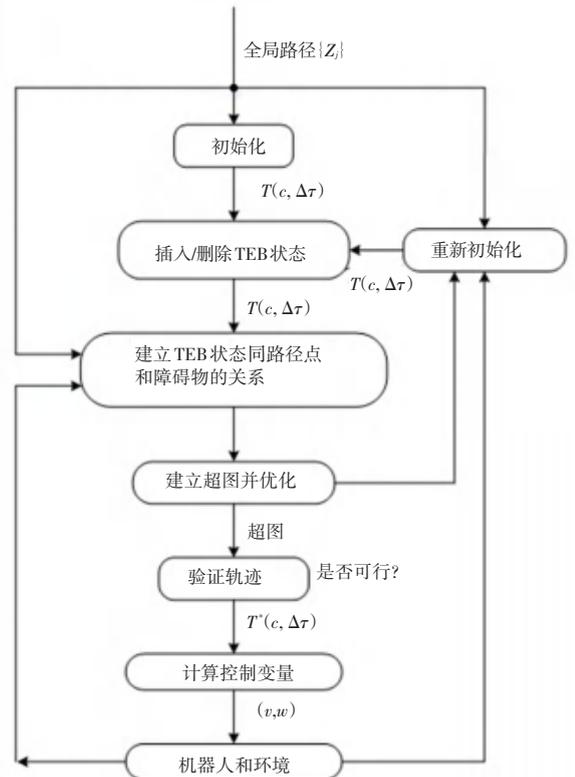


图 5 TEB 求解流程图

Fig. 5 Flowchart of TEB solution

其中, f_p 表示速度、加速度等惩罚函数; f_T 表示障碍物、路径等目标函数; σ_k 表示配置的权重; $T/\{S_1, S_n\}$ 表示起始和终止位姿 S_1 与 S_n 不受优化。TEB 优化问题可以转化成超图问题。求解出来的结果 T^* , 可以计算输入的控制变量 $u = (v(t), \omega(t))^T$, 相应地求解出左右驱动轮编码器的控制指令 $\omega_L(t)$ 和 $\omega_R(t)$, 从而驱动小车到达目标位置。

2 A* 算法改进与 TEB 算法融合

2.1 改进 A* 算法

2.1.1 A* 算法改进策略

首先,从流程图可以看出,在循环找到目标点的过程中,寻找扩展节点是找到 $F(n)$ 最小的扩展方向,扩展方向的频繁改变会导致生成的路径曲折,机器人运行卡顿,对工业应用机器人效率影响十分严重。由于是搜索 *queue* 容器里所有节点最小值,搜索过程中可能有不同的路径方向,所以只针对同一路径条件下扩展方向进行研究。这里对每个转向节点进行判断,如果最小节点方向与原路径方向下一节点代价值相差不大,则认定没有必要进行转向。

其次,为了扩大工业上移动机器人的运行效率,配合上文对启发式函数 $H(n)$ 的计算方法进行改进。3种启发式函数的计算方式,研究后可知,曼哈顿距离最大,欧几里得距离次之,切比雪夫距离最小。 $H(n)$ 值越大,则搜索路径的目的性越强,但环境地图障碍物太多会导致搜索路径频频碰壁,效率反而不高,因此计算合适的启发式函数值是必要的。

为了合理优化 $H(n)$, 更好地描述障碍物状态,引入障碍物占比系数 K 和离散率系数 D , 分别表示障碍物栅格数占环境地图总栅格数比重和障碍物分散程度。2个系数可以很好地表征环境地图障碍物多少以及复杂程度等信息。相应数学公式具体如下:

$$K = \frac{N}{(|x_s - x_t| + 1) \times (|y_s - y_t| + 1)} \quad (12)$$

其中, (x_s, y_s) 是起始点; (x_t, y_t) 是目标点; N 表示障碍物栅格数。 $H(n)$ 选用曼哈顿距离,评价函数 $F(n)$ 相应发生变化。此处用到的数学公式可写为:

$$H(n) = |x_s - x_t| + |y_s - y_t| \quad (13)$$

$$F(n) = G(n) + (1 - \ln K) \times D \times H(n) \quad (14)$$

由式(13)、式(14)可知,障碍物越多、越分散,则 D 的值越小,其栅格数 N 也越大, K 值越大, $1 -$

$\ln K$ 变小,将会扩展更多搜索方向,适应复杂环境。

2.1.2 仿真验证对比分析

对机器人的路径规划算法进行仿真,验证算法的有效性。根据国家标准中机器人移动能力算法测评要求建立仿真环境场景。仿真试验场景如图6所示。地面为水平地面,包含4面墙,墙内面积为 $2\,500\text{ m}^2$,墙高度为 $2\,500\text{ mm}$,包含天花板,天花板高度为 $2\,500\text{ mm}$,颜色为白色,包含4根圆柱和 A、B、C、D、E、F、G、H 共8个物体,4根圆柱的位置固定, A-H 坐标非固定,这8个物体相对均匀地分布于4面墙内,且彼此之间并不接触。圆柱及其它物体规格见表1。

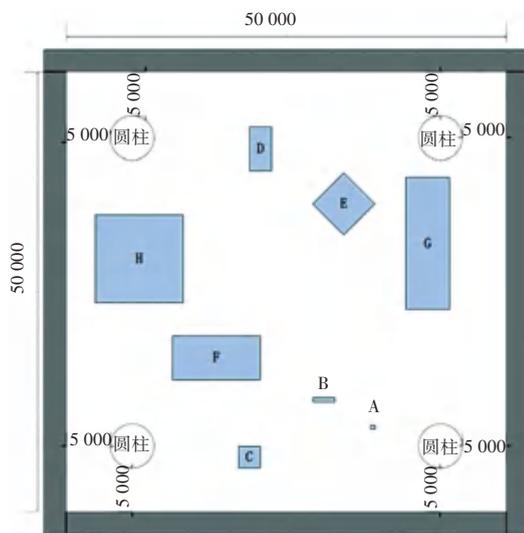


图6 仿真场景模拟图

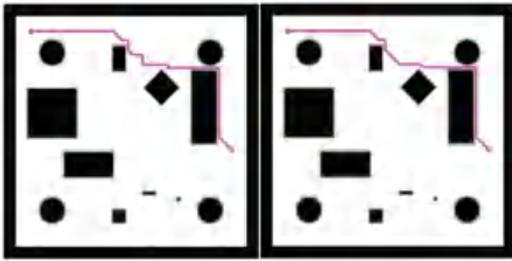
Fig. 6 Simulation scene simulation diagram

表1 仿真场景物体尺寸表

Tab. 1 Objects size table of simulation scene

物体	数量	尺寸	颜色
A	1	长 500 mm 宽 500 mm 高 1 000 mm	白
B	1	长 2 500 mm 宽 500 mm 高 1 000 mm	白
C	1	长 2 500 mm 宽 2 500 mm 高 1 000 mm	白
D	1	长 5 000 mm 宽 2 500 mm 高 1 000 mm	白
E	1	长 5 000 mm 宽 2 500 mm 高 1 000 mm	白
F	1	长 10 000 mm 宽 5 000 mm 高 1 000 mm	白
G	1	长 15 000 mm 宽 5 000 mm 高 1 000 mm	白
H	1	长 10 000 mm 宽 10 000 mm 高 1 000 mm	白
圆柱	4	直径 5 000 mm 高 2 500 mm	白

栅格化环境地图如图 7 所示。在图 7 中, 模拟实际导航, 机器人长 1 000 mm, 且可将之看成机器人中心点的移动, 障碍物需膨胀 500 mm 长度, 膨胀部分称为代价地图。仿真计算结果见表 2。从仿真结果来看, 改进 A* 算法相对于传统 A* 算法在计算时间和拐点数上都有较大的改善。从表 2 中的实验仿真数据可以看出, 改进 A* 算法有较好的实时性, 路径平滑性更好, 便于机器人较好地沿着路径轨迹移动到目标点。需通过实验做进一步验证。



(a) 传统 A* 算法仿真图 (b) 改进 A* 算法仿真图

图 7 仿真结果图

Fig. 7 Simulation results diagram

表 2 仿真计算结果统计表

Tab. 2 Statistical table of simulation calculation results

算法	拐点	迭代次数	路径长度
传统 A*	11	3 994	50.399 8
改进 A*	9	3 303	41.326 4

2.2 TEB 参数优化

为了机器人导航避障效果更好, 进行参数的一系列优化。研究可知, 机器人加载了一系列的参数文件, 其中 `teb_local_planner_params.yaml` 等参数文件要着重考虑。并对机器人运行轨迹、机器人本体参数、打孔位置约束以及障碍物约束等参数进行修正。移动操作建筑机器人部分运动参数设置如下: 最大线速度为 0.5 m/s, 最大角速度为 0.6 rad/s, 最大线加速度为 0.5 m/s², 最大角加速度为 0.5 rad/s², 转角容许误差为 0.05 m, 与障碍物间的最小距离为 0.25 m, 障碍物膨胀半径为 0.5 m。

2.3 融合策略

机器人采用精度较高的北阳 UTS-10LX 激光雷达, 使用 `cartographer` 算法功能包进行环境栅格建图, 并通过雷达准确获取周围环境的深度信息, 定位到自己的位置, 结合时间弹性带算法进行实时的局部路径规划, 达到避障的目的, 获取机器人运动的最佳路径。将改进的 A* 算法与 TEB 相融合, 以保证

动态工作路径的实时性和时效性。

根据已知的当前实验室环境建立地图, 进行全局路径规划, 从规划的全局路径中截取当前位置的一段, 将之离散化为带有时间信息的点, 通过自身运动以及与障碍物代价函数的约束, 求解到时间最优的位姿序列轨迹, 并根据当前计算的速度以及角速度值, 下发指令到 AGV 左右驱动轮的编码器, 驱动小车高效到达目标位置。融合算法流程如图 8 所示。

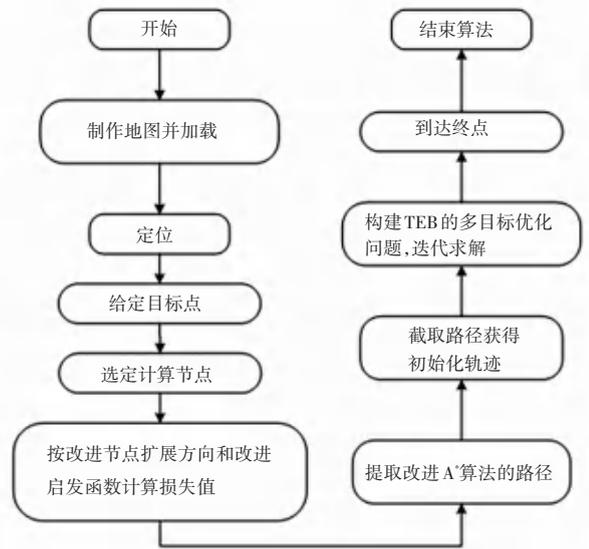


图 8 融合算法流程图

Fig. 8 Flow chart of the fusion algorithm

3 机器人运动规划实验验证与分析

3.1 实验准备

在机器人上进行实验验证, 使用平台自带的里程计以及北阳 UTS-10LX 激光雷达感知周围环境。且小车配有处理器为 i7 八代的工控机作为主机, 个人笔记本作为从机, 从机配有 i7 十代的虚拟机, 主机和从机虚拟机都安装有 `ubuntu18.04` 的操作系统和 `ROS(melodic)`。在实验室内网下, 主机、从机以及从机虚拟机可通过 IP 互相通信, 从机和从机虚拟机可实现对主机的远程控制。

在所搭建的实验场景中, 选用 `cartographer` 算法功能包进行环境地图的建模, 运行程序节点如图 9 所示。由图 9 可知, 启动 `teleop_twist_joy` 为手柄控制节点以及 `cmdvel2can` 为速度控制等节点, 建立环境地图。运行包内 `local.launch`, 通过对子图的扫描匹配来实现定位, 做好算法验证前的建图和定位准备。环境建图如图 10 所示。

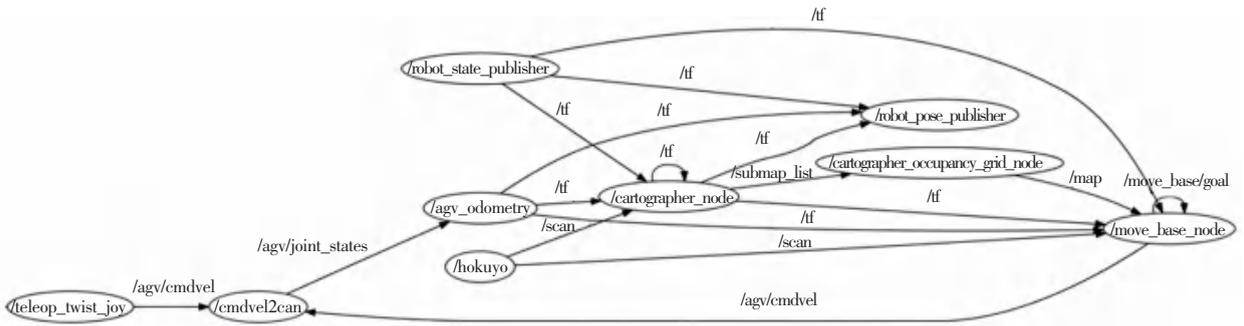


图9 融合算法程序运行节点图

Fig. 9 Running nodes graph of the fusion algorithm program



图10 cartographer 环境建图

Fig. 10 cartographer environment construction

3.2 实验验证

本文的融合算法适用于大范围非结构化建筑环境,选择实验环境对算法可行性进行验证。利用移动操作机器人分别对传统 A* 算法和改进 A* 算法进行实验。

3.2.1 传统 A* 算法实验验证

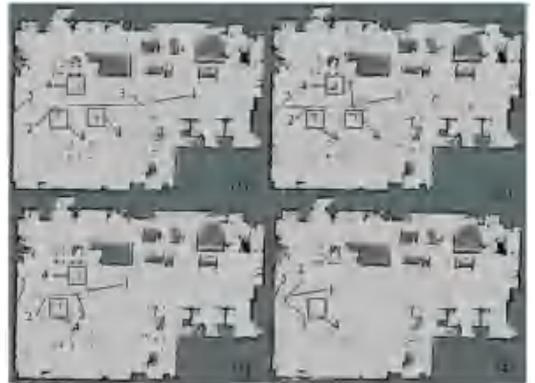
现场实验验证图如图 11 所示。图 11 中,白色点云是小车定位轮廓与环境地图匹配点,1 指的黑色圆角矩形为移动机器人本体,2 指的黑色线条为改进 A* 算法生成的全局路径,3 指的红色线条为实时截取全局路径生成的局部路径。融合算法的实验步骤描述如前文融合算法内容所述,加载使用调参 cartographer 建好的栅格地图,给定目标点位的坐标和方向,机器人生成全局路径,TEB 算法实时截取全局路径生成局部路径,放置的桌子为 4,可认为是建筑环境承重水泥柱,5 为打孔位置,实际验证机器人绕过建筑环境水泥柱并近距离停靠在平行于墙面的打孔位置,观察并记录移动机器人到达后的位姿数据信息以及导航避障情况。



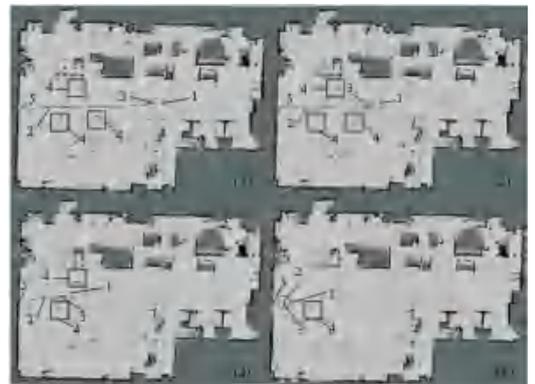
图11 现场实验验证图

Fig. 11 Field experiment verification diagram

给定打孔位点,机器人运动规划过程如图 12 所示。图 12 中,1 表示机器人本体,2 表示全局路径,3 表示局部路径,4 表示水泥柱,5 表示墙面打孔位置。其中,采用柜子模拟建筑环境中的水泥承重柱。经过 10 组实验发现,实时生成的全局路径局部有许多小齿状拐点,影响机器人的运行效率。



(a) 传统 A* 运动规划图



(b) 改进 A* 运动规划图

图12 建筑机器人运动规划验证实验图

Fig. 12 Experimental diagram of construction robots motion planning verification

3.2.2 改进 A* 算法实验验证

同理,消除实验数据的偶然性,共做 10 组实验。改进 A* 算法生成的全局路径较传统 A* 算法来说,减少了许多拐点,路径更为平滑。到达的目标点和

所需要的方向满足建筑机器人位置和方向的要求,验证该改进算法策略的可行性。

4 结束语

(1) 提出改进的 A* 算法融合 TEB 算法进行建筑机器人的运动控制,并进行了实验验证。高效完成建筑机器人近距离停靠在平行于打孔位置墙面的任务,验证了建筑环境下该策略的有效性。其中,提出改进的 A* 算法,比传统 A* 算法效率更高,路径平滑性更好,便于建筑机器人生成较好的全局路径轨迹。

(2) 系统分析建筑机器人导航避障,提出进一步改进策略:可以通过优化建图定位参数、比如环境子地图匹配频率等,减少小车定位变动次数,以减少小车导航避障过程中方向摆动,提高运动效率;也可通过研发算法继续改进全局与局部路径规划算法,补偿机器人运动误差,以达到更好的导航效果。

参考文献

- [1] GUL F, MIR I, ABUALIGAH L, et al. A Consolidated Review of Path Planning and Optimization Techniques: Technical Perspectives and Future Directions [J]. *Electronics*, 2021, 10(18): 2250.
- [2] ZHU Kai, ZHANG Tao. Deep reinforcement learning based mobile robot navigation: A review [J]. *Tsinghua Science and Technology*, 2021, 26(5): 674-691.

- [3] 张贺, 缪存孝, 唐友军, 等. 移动机器人自主动态避障方法研究 [J]. *北京航空航天大学学报*, 2022, 48(06): 1013-1021.
- [4] 胡杰, 张华, 傅海涛, 等. 改进人工势场法在移动机器人路径规划中的应用 [J]. *机床与液压*, 2021, 49(03): 6-10.
- [5] KASHYAP A K, PARHI D R. Obstacle avoidance and path planning of humanoid robot using fuzzy logic controller aided owl search algorithm in complicated workspaces [J]. *Industrial Robot: The International Journal of Robotics Research and Application*, 2021.
- [6] 刘子毅, 李铁军, 孙晨昭, 等. 基于 BIM 的建筑机器人自主导航策略优化研究 [J]. *计算机工程与应用*, 2022, 58(15): 302-308.
- [7] AGARWAL D, BHARTI P S. Implementing modified swarm intelligence algorithm based on Slime moulds for path planning and obstacle avoidance problem in mobile robots [J]. *Applied Soft Computing*, 2021, 107: 107372.
- [8] 李承远. 基于改进 TEB 算法的自主导航控制 [D]. 杭州: 杭州电子科技大学, 2020.
- [9] MUGARZA I, MUGARZA J C. A coloured Petri net-and D* Lite-based traffic controller for automated guided vehicles [J]. *Electronics*, 2021, 10(18): 2235.
- [10] 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(01): 14-22.
- [11] 柴红杰, 李建军, 姚明. 改进的 A* 算法移动机器人路径规划 [J]. *电子器件*, 2021, 44(02): 362-367.
- [12] QI J, YANG H, SUN H. MOD-RRT*: A sampling-based algorithm for robot path planning in dynamic environment [J]. *IEEE Transactions on Industrial Electronics*, 2020, 68(8): 7244-7251.
- [13] LYU D, CHEN Ziwei, CAI Zesu, et al. Robot path planning by leveraging the graph-encoded Floyd algorithm [J]. *Future Generation Computer Systems*, 2021, 122: 204-208.

(上接第 54 页)

参考文献

- [1] 陈义君, 雷永江. 森林火灾的危害和防火措施 [J]. *现代化农业*, 2022(02): 38-39.
- [2] 陈健. 全球多地集中爆发森林大火 [J]. *生态经济*, 2021, 37(10): 1-4.
- [3] 刘树东, 姚文渤, 张艳. 雾天条件下基于机器视觉的森林火灾监测 [J]. *计算机工程与科学*, 2020, 42(07): 1253-1261.
- [4] 富雅捷, 张宏立. 基于迁移学习的卷积神经网络森林火灾检测方法 [J]. *激光与光电子学进展*, 2020, 57(04): 128-134.
- [5] 傅天驹, 郑嫦娥, 田野, 等. 复杂背景下基于深度卷积神经网络的森林火灾识别 [J]. *计算机与现代化*, 2016(03): 52-57.
- [6] 杨剑, 刘方涛, 张涛, 等. 基于改进型残差网络烟雾图像识别 [J]. *科学技术与工程*, 2019, 19(32): 236-243.
- [7] MUHAMMA D K, KHAN S, PALADE V, et al. Edge intelligence-assisted smoke detection in foggy surveillance environments [J]. *IEEE Transactions on Industrial Informatics*, 2020, 16(2): 1067-1075.
- [8] KHAN S, MUHAMMAD K, MUMTAZ S, et al. Energy-efficient deep CNN for smoke detection in foggy IoT environment [J]. *IEEE Internet of Things Journal*, 2019, 6(6): 9237-9245.
- [9] ÖZBEK M E, YILDIZ U E. Smoke detection from foggy environment based on color spaces [J]. *International Journal of Applied Mathematics*

Electronics and Computers, 2021, 9(3): 72-78.

- [10] MA Ningning, ZHANG Xiangyu, ZHENG Haitao, et al. ShuffleNet V2: Practical guidelines for efficient CNN architecture design [C]// *European Conference on Computer Vision*. Cham: Springer, 2018: 1-19.
- [11] HENDRYCKS D, GIMPEL K. Gaussian error linear units (GELUs) [J]. *arXiv preprint arXiv:1606.08415*, 2016.
- [12] LIN M, CHEN Q, YAN S. Network in network [J]. *arXiv preprint arXiv:1312.4400*, 2013.
- [13] BASHA S H S, DUBEY S R, PULABAIGARI V, et al. Impact of fully connected layers on performance of convolutional neural networks for image classification [J]. *Neurocomputing*, 2020, 378: 112-119.
- [14] WANG Qilong, WU Banggu, ZHU Pengfei, et al. ECA-Net: Efficient channel attention for deep convolutional neural networks [C]// *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020: 11534-11542.
- [15] SUN Guolei, CHOLAKKAL H, KHAN S, et al. Fine-grained recognition: Accounting for subtle differences between similar classes [C]// *Proceedings of the Aaai Conference on Artificial Intelligence*. Honolulu, Hawaii, USA: AAAI, 2019, 34(07): 12047-12054.