

控制与决策

Control and Decision

基于路标点追踪的差速驱动机器人运动规划

刘延彬, 姜媛媛

引用本文:

刘延彬, 姜媛媛. 基于路标点追踪的差速驱动机器人运动规划[J]. *控制与决策*, 2023, 38(9): 2529–2536.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.2025>

您可能感兴趣的其他文章

Articles you may be interested in

[一种基于免疫机理的确定性移动机器人路径规划算法](#)

A path planning algorithm of deterministic mobile robot based on immune mechanism

控制与决策. 2021, 36(10): 2418–2426 <https://doi.org/10.13195/j.kzyjc.2020.0059>

[移动机器人运动规划中的深度强化学习方法](#)

Deep reinforcement learning for motion planning of mobile robots

控制与决策. 2021, 36(6): 1281–1292 <https://doi.org/10.13195/j.kzyjc.2020.0470>

[机器人信息增益RRT环境探索算法](#)

Robot RRT based on information gain for environment exploration

控制与决策. 2021, 36(11): 2683–2689 <https://doi.org/10.13195/j.kzyjc.2020.1007>

[基于 \$\text{pm}3\sigma\$ 正态概率区间分族遗传蚁群算法的移动机器人路径规划](#)

Path planning of mobile robot based on $\text{pm}3\sigma$ normal probability interval population division using genetic ant-colony algorithm

控制与决策. 2021, 36(12): 2861–2870 <https://doi.org/10.13195/j.kzyjc.2020.0745>

[凸优化与A*算法结合的路径避障算法](#)

Convex optimization and A-star algorithm combined path planning and obstacle avoidance algorithm

控制与决策. 2020, 35(12): 2907–2914 <https://doi.org/10.13195/j.kzyjc.2019.0351>

基于路标点追踪的差速驱动机器人运动规划

刘延彬^{1†}, 姜媛媛²

(1. 安徽理工大学 力学与光电物理学院, 安徽 淮南 232001;

2. 安徽理工大学 电气与信息工程学院, 安徽 淮南 232001)

摘要: 首先, 针对三角网格路径规划方法采用D-P算法提取路标点时, 其最大阈值不易确定等问题, 提出基于碰撞检测的抽取路标点方法. 同时采用Pure Pursuit算法跟踪路标点, 对差速驱动机器人进行运动规划; 然后, 通过实验对比分析表明, 在提取路标点时, 与D-P算法相比, 碰撞检测方法更加优越; 最后, 通过差速驱动机器人的运动规划实验表明: Pure Pursuit算法追踪路标点方法规划的运动轨迹为一条光滑曲线, 能够有效地避开地图上的障碍物; 机器人角速度和线速度均为光滑函数, 变化平缓, 在路标点附近出现较小波动, 波动范围均在允许值内; 运动规划时间为0.049 s, 完全能够满足实际需求. 研究表明, 基于路标点追踪的移动机器人运动规划是一种简单有效的运动规划方法.

关键词: 运动规划; 三角网格; 碰撞检测; 路标点; 路标点追踪

中图分类号: TP391

文献标志码: A

DOI: 10.13195/j.kzyjc.2021.2025

引用格式: 刘延彬, 姜媛媛. 基于路标点追踪的差速驱动机器人运动规划[J]. 控制与决策, 2023, 38(9): 2529-2536.

Motion planning of differential driven robot based on tracking

LIU Yan-bin^{1†}, JIANG Yuan-yuan²

(1. School of Mechanics and Optoelectronic Physics, Anhui University of Science and Technology, Huainan 232001, China; 2. College of Electrical and Information Engineering, Anhui University of Science and Technology, Huainan 232001, China)

Abstract: When using the triangular mesh method in path planning, aiming at the problem that the maximum threshold of the D-P algorithm for extracting waypoints is not easy to determine, this paper proposes a method for extracting waypoints based on collision detection. The Pure Pursuit algorithm is used to track the way points and plan the motion of the differential driven robot. The experimental results show that the collision detection method is better than the D-P algorithm in extracting waypoints. Finally, the motion planning experiment of the differential driven robot shows that the motion trajectory planned by the Pure Pursuit algorithm tracking waypoints is a smooth curve, which can effectively avoid obstacles on the map. The angular velocity and linear velocity of the robot are smooth functions with gentle changes. There are small fluctuations near the waypoints, and the fluctuation range is within the allowable value. The motion planning time is 0.049 s, which can fully meet the actual needs. The results show that the motion planning of mobile robot based on road marking tracking is a simple and effective motion planning method.

Keywords: motion planning; triangular mesh; collision detection; waypoints; tracking way points

0 引言

运动规划与路径规划是不同的两个概念. 路径规划是单纯几何问题, 是指在机器人的位形空间中找到一条连接初始点与终点的无碰撞连通路, 只考虑位形空间的障碍物约束, 不考虑运动学和动力学约束^[1]. 对于一些特殊情况, 路径规划得到的参考

路径在实际应用中可能根本无法实现^[2]. 运动规划是在约束构型空间中规划出一条具有控制序列的运动轨迹, 使得机器人能够完成预定运动. 运动规划输出速度或力等直接实施的控制信号, 不但需要满足几何约束, 而且还需要满足运动学或动力学约束^[1]. 运动规划是自主移动机器人领域的重要研究方向^[2], 其可

收稿日期: 2021-11-21; 录用日期: 2022-04-27.

基金项目: 安徽省重点研究与开发计划项目(202104g01020012); 安徽理工大学环境友好材料与职业健康研究院研发专项基金项目(ALW2020YF18).

责任编辑: 谢晖.

[†]通讯作者. E-mail: liuyanbin1974@126.com.

应用于各种实际工程问题,具有重要的理论和现实意义. 运动规划可分解为3个过程^[3]:

1) 参考路径规划,为运动规划提供参考路径;

2) 参考路径跟踪,获得规划的运动轨迹、行走速度、转角等运动学或动力学控制信号;

3) 碰撞检测,在位形空间中检测规划运动轨迹是否与障碍物发生干涉,得到无碰撞规划运动轨迹.

路径规划的目标是识别从初始位置到最终位置的无碰撞连通路. 经过多年发展,路径规划技术研究已经成熟. 目前,主要的路径规划算法可分为5种^[1,4]: 1) 精确路线图方法^[5-8]; 2) 单元分解^[9-10]; 3) 人工势场法^[11-12]; 4) 智能算法^[13-18]; 5) 基于随机采样的路径规划^[19-22]. 基于随机采样的路径规划方法主要有两种, RRT和PRM方法,文献[1]表明,它们在高维空间路径规划中具有显著优势,因而得到了广泛关注. 基于随机采样方法的路径规划计算速度较慢,路径冗余节点较多,不具有可重复性. 对于复杂地图(如具有狭长通道的地图),随机采样方法可能得不到有效的路径. 为解决随机采样算法的不可重复性等缺点,文献[23]提出了TMM方法. TMM方法采用三角网格对地图剖分,进行确定性采样,搜索得到参考路径;基于D-P算法缩减参考路径冗余节点,提取路标点;采用三次样条函数对路标点进行拟合给出光滑的规划路径. TMM方法对地图具有很强的适应性,是一种优秀的路径规划算法. 文献[23]表明, D-P算法在提取路标点时,很难给出最优距离阈值,且距离阈值选取不合理,规划路径会与障碍物产生干涉.

针对D-P算法的缺点,本文提出基于碰撞检测抽取路标点的方法. 同时采用Pure Pursuit算法跟踪路标点,对差速驱动机器人进行运动规划. 通过实验对比分析验证了碰撞检测更加优越. 最后,通过复杂地图的差速驱动机器人运动规划问题验证所提出算法的可靠性.

1 基于三角网格地图的路标点提取

对连续曲面进行网格剖分的基本单元形状可取为三角形、四边形、曲边形等,由于三角形网格对地图的适应性较好,且TMM方法是一种优秀的路径规划算法^[23],可很好地求解复杂地图中的机器人路径规划问题,本文选用三角形网格.

1.1 三角网格地图建模和表示方法

三角形网格剖分如图1所示. 三角网格地图可用节点矩阵和单元连接矩阵表示,设剖分后的可行域有 m 个节点和 k 个单元,且第 i 个节点 P_i 的坐标为 (x_i, y_i, z_i) ,则节点矩阵可表示为

$$P = \begin{bmatrix} x_1 & x_2 & \dots & x_m \\ y_1 & y_2 & \dots & y_m \\ z_1 & z_2 & \dots & z_m \end{bmatrix}^T, \quad (1)$$

单元连接矩阵可表示为

$$C = \begin{bmatrix} p_1^{(1)} & p_2^{(1)} & \dots & p_k^{(1)} \\ p_1^{(2)} & p_2^{(2)} & \dots & p_k^{(2)} \\ p_1^{(3)} & p_2^{(3)} & \dots & p_k^{(3)} \end{bmatrix}^T. \quad (2)$$

其中: $p_j^{(i)}$ 为第 j 个单元的第 i 个节点编号;此外,连接矩阵还表示单元中节点的连接关系,即在第 j 个单元中, $p_j^{(1)}$ 与 $p_j^{(2)}$ 连接, $p_j^{(2)}$ 与 $p_j^{(3)}$ 连接, $p_j^{(3)}$ 与 $p_j^{(1)}$ 连接. 以图1为例,第 j 个单元包含节点3、4和6,其中:节点3与6连接,节点6与4连接,节点4与3连接.

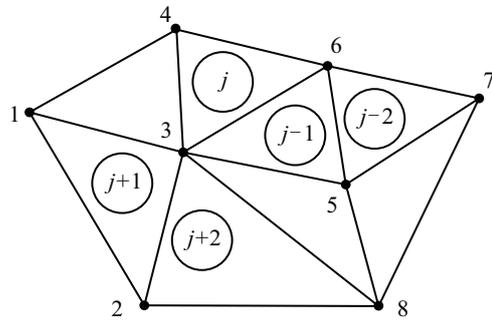


图1 三角剖分网格

基于三角网格地图建立赋权有向循环图,需要对各单元进行组装,组装过程是将单元矩阵元素按照相关节点的编号放置于整体节点关联矩阵中. 由于只有在同一单元中相关联的节点才会在整体关联矩阵相应的行和列中出现,则在整体矩阵中,必然会存在大量的0矩阵,整体矩阵为稀疏矩阵. 由单元连接矩阵,可构造节点关联矩阵 T 和 S ,有

$$T = [t_1 \ t_2 \ \dots \ t_n], \quad (3)$$

$$S = [s_1 \ s_2 \ \dots \ s_n], \quad (4)$$

其中 T 和 S 中的元素均为网格节点编号. T 中的第 i 个元素 t_i 与 S 中的第 i 个元素 s_i 是相互关联的,表示节点 t_i 与节点 s_i 连接. 由于节点间的距离不同,设关联矩阵 T 与 S 的距离矩阵 D 为

$$D = [d_1 \ d_2 \ \dots \ d_n]. \quad (5)$$

1.2 参考路径提取

基于三角网格地图的路径规划是在给定的初始位姿点 B 和目标点 E 的条件下,在三角网格地图中搜索出一条连接 B 与 E 的可行路径. 首先,在三角网格地图中求取 B 与 E 距离最近的节点 \bar{B} 和 \bar{E} ,在节点矩阵中,分别采用 B 和 E 代替 \bar{B} 和 \bar{E} ;然后,采用启发式搜索算法从赋权无向连通图 G 中搜索出一条连接点

B 与 E 的可行路径. 启发式算法包括广度优先搜索算法、深度优先算法、 A^* 搜索算法以及Dijkstra搜索算法等, 本文采用Dijkstra搜索算法. 搜索的路径依赖于网格节点, 网格大小不同, 网格数量亦不同, 且网格节点数目有限, 不能完全覆盖整个自由空间, 因此得到的最终路径并不是最优路径, 而只是相对三角网格地图的最优路径.

1.3 基于碰撞检测的路标点提取

在运动规划过程中, 节点过多会带来控制器的震荡. 基于三角网格地图的路径节点过多, 机器人在运动过程中需要频繁地改变速度和角速度来调整运动姿态. 且拐点过多使得行走路径过长, 机器人行驶时间加长. 本文采用碰撞检测方法剔除参考路径中的冗余节点, 提取若干节点作为路标点. 碰撞检测原理如图2所示. 若要检测 A 、 B 两点间的连线是否与障碍物发生碰撞, 则需对直线 AB 进行等距采样, 逐个检测采样点 P_1, P_2, \dots, P_n 是否处于障碍物的位置.

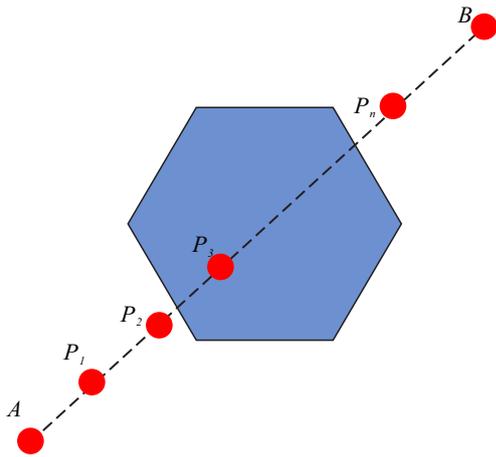


图2 碰撞检测

基于碰撞检测提取路标点原理如图3所示. 假设参考路径的节点为 Q_1, Q_2, Q_3, Q_4 和 Q_5 , 提取路标点的步骤如下.

step 1: 检测节点 Q_1 与 Q_3 的连线是否与路障发生碰撞, 若无碰撞, 则舍弃节点 Q_2 , 保留 Q_3 作为路标点.

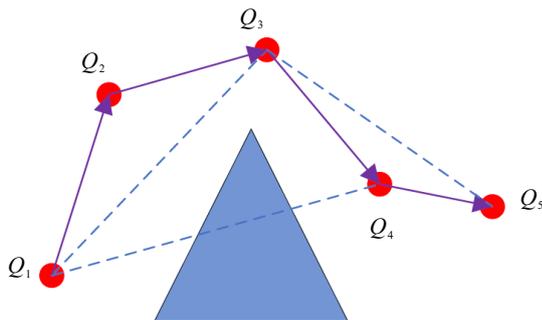


图3 碰撞检测提取路标点

step 2: 检测节点 Q_3 与 Q_4 的连线是否与路障发生碰撞, 若无碰撞, 则保留节点 Q_4 作为路标点.

step 3: 检测节点 Q_3 与 Q_5 的连线是否与路障发生碰撞, 若无碰撞, 则删除节点 Q_4 , 保留节点 Q_5 作为路标点.

step 4: 给出路标点序列 Q_1, Q_3 和 Q_5 .

2 运动规划

运动规划是自主移动机器人关键技术之一. 在满足几何约束、运动学或动力学约束的前提下, 运动规划需要在机器人的约束构型空间中规划出一条具有控制序列的运动轨迹. 本文针对四轮差速驱动机器人, 采用Pure Pursuit算法跟踪路标点的方法完成运动规划.

2.1 四轮差速驱动机器人运动方程

差速驱动机器人运动学如图4所示, 输入(控制)变量为右轮的速度 $v_R(t)$ 和左轮的速度 $v_L(t)$, r 为车轮半径, L 为左右车轮间的距离, 点ICR为机器人行驶的速度瞬心, $R(t)$ 为机器人行驶轨迹的瞬时半径, 则机器人转动角速度 $\omega(t)$ 为

$$\omega(t) = \frac{v_L(t)}{R(t) - \frac{L}{2}} = \frac{v_R(t)}{R(t) + \frac{L}{2}} \quad (6)$$

由式(6), 得到

$$\omega(t) = \frac{v_R(t) - v_L(t)}{L} \quad (7)$$

瞬时转弯半径为

$$R(t) = \frac{L v_R(t) + v_L(t)}{2 v_R(t) - v_L(t)} \quad (8)$$

且

$$v(t) = \omega(t)R(t) = \frac{v_R(t) + v_L(t)}{2} \quad (9)$$

则其运动学随体坐标系方程为

$$\begin{bmatrix} \dot{x}_M \\ \dot{y}_M \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} v_{xM}(t) \\ v_{yM}(t) \\ \omega(t) \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ 0 & 0 \\ -\frac{r}{L} & \frac{r}{L} \end{bmatrix} \begin{bmatrix} \omega_L(t) \\ \omega_R(t) \end{bmatrix} \quad (10)$$

运动学全局坐标系方程为

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos(\varphi(t)) & 0 \\ \sin(\varphi(t)) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix} \quad (11)$$

采用左右轮的线速度表示的运动学全局坐标系方程为

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos(\varphi(t)) & 0 \\ \sin(\varphi(t)) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{v_R(t) + v_L(t)}{2} \\ \frac{v_R(t) - v_L(t)}{L} \end{bmatrix} \quad (12)$$

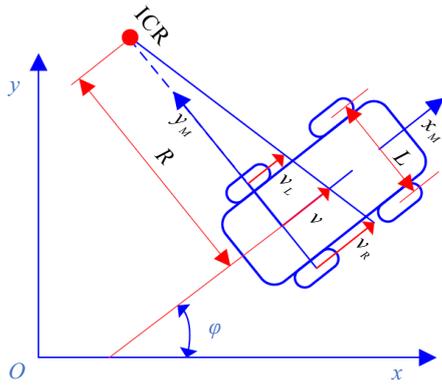


图4 差速驱动机器人运动学

2.2 Pure Pursuit 路标点跟踪

路标点跟踪是指在机器人行驶过程中,通过比较机器人本体当前位置与预期路标点间的偏差,对机器人进行转向控制,从而使得机器人沿着期望路标点行驶. Pure Pursuit 方法是基于几何方法的路标点跟踪算法. 由于基于几何方法的控制算法简单且直接,不用考虑机器人的动力学模型,需要控制的参数少,适用范围比较广泛^[24-27].

如图5所示, Pure Pursuit 路标点跟踪算法以移动机器人后轴中心点为切点,移动机器人的纵向本体为切线,通过控制移动机器人的转角,使得移动机器人可沿着一条经过路标点的圆弧行驶. 路标点跟踪需要控制移动机器人的后轴中心点经过要追踪的点,由正弦定理可得

$$\frac{l_d(t)}{\sin(2\alpha(t))} = \frac{R(t)}{\sin\left(\frac{\pi}{2} - \alpha(t)\right)}, \quad (13)$$

$$\frac{l_d(t)}{2 \sin(\alpha(t)) \cos(\alpha(t))} = \frac{R(t)}{\cos(\alpha(t))}. \quad (14)$$

由式(14),得到

$$\frac{l_d(t)}{\sin(\alpha(t))} = 2R(t). \quad (15)$$

将式(8)代入(15),得到

$$\frac{v_R(t) + v_L(t)}{v_R(t) - v_L(t)} = \frac{l_d(t)}{L \sin(\alpha(t))}, \quad (16)$$

因此

$$\frac{v_R(t) + v_L(t)}{v_R(t) - v_L(t)} = \frac{l_d(t)}{L \sin(\alpha(t))}. \quad (17)$$

由式(7)和(17)可得到纯追踪控制算法的控制量为

$$\begin{aligned} v_R(t) &= v(t) \left(1 + \frac{L \sin(\alpha(t))}{l_d(t)}\right), \\ v_L(t) &= v(t) \left(1 - \frac{L \sin(\alpha(t))}{l_d(t)}\right). \end{aligned} \quad (18)$$

若 t 时刻自主移动机器人与目标路标点的夹角为 $\alpha(t)$, 距离目标路标点的前视距离为 $l_d(t)$, 且自主移动机器人轴距 L 固定, 则利用式(18)可得到左右轮的

转速 $v_R(t)$ 和 $v_L(t)$.

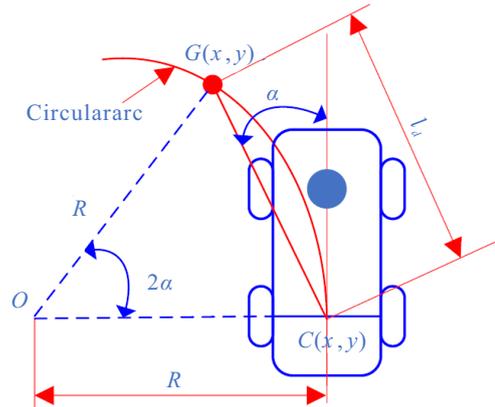


图5 轨迹跟踪

设自主移动机器人当前姿态与目标路标点 in 横向上的误差为 e_l , 则夹角正弦为

$$\sin(\alpha) = \frac{e_l(t)}{l_d(t)}, \quad (19)$$

圆弧的弧度可写为

$$\kappa = \frac{2}{l_d^2(t)} e_l(t). \quad (20)$$

由式(20)可知, Pure Pursuit 追踪控制器是横向转角的 P 控制器, P 控制器受前视距离的影响很大. 在不同的速度下需要选择不同的前视距离. 本文将前视距离表示为车辆纵向速度的线性函数^[27], 即

$$l_d(t) = kv(t), \quad (21)$$

因此, 左右轮的速度变为

$$\begin{aligned} v_R(t) &= v(t) \left(1 + \frac{L \sin(\alpha)}{kv(t)}\right), \\ v_L(t) &= v(t) \left(1 - \frac{L \sin(\alpha)}{kv(t)}\right). \end{aligned} \quad (22)$$

由式(22)可知, Pure Pursuit 纯追踪控制器的调整变为系数 k 的调整, 通常需要使用最大、最小前视距离约束前视距离. 越大的前视距离意味着轨迹追踪曲线越平滑; 越小的前视距离使得轨迹追踪越加精确, 但是会带来控制器的震荡.

3 仿真实验

实验环境为 Windows 10 系统, Matlab 2021 a, 英特尔奔腾处理器, CPU 3.60 GHz (8CPUs), 内存 16 GB. 机器人轮距 0.4 m, 在路径规划过程中被视为质点. 基于机器人实际尺寸大小, 对障碍物进行了膨胀处理, 防止机器人在运行中与障碍物发生碰撞.

3.1 碰撞检测算法与 D-P 算法对比

对于 D-P 算法, 无法给出最优最大距离 ϵ , 只能多次尝试, 选取相对最优值, 且最大距离 ϵ 影响 D-P 算法提取路标点的效果, 故本文分别与不同的最大距离 ϵ

进行对比. 原始地图和网格剖分如图6所示, 机器人的起始点为[1, 0], 终点为[12, 2]. 当 $\epsilon = 0.6$ 时, 对比如图7所示; 当 $\epsilon = 0.7$ 时, 对比如图8所示; 当 $\epsilon = 0.8$ 时, 对比如图9所示.

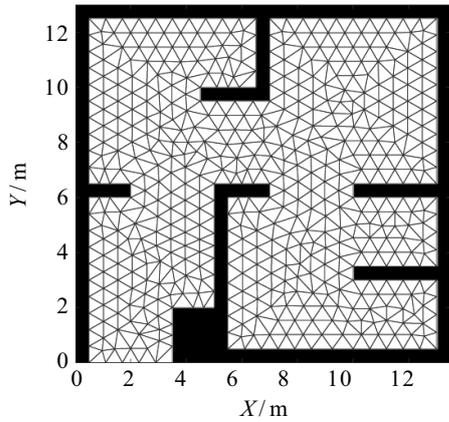


图6 对比地图与网格剖分

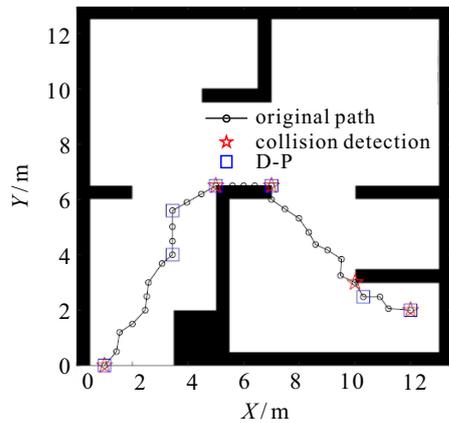


图7 $\epsilon = 0.6$ 时, 路标点提取对比

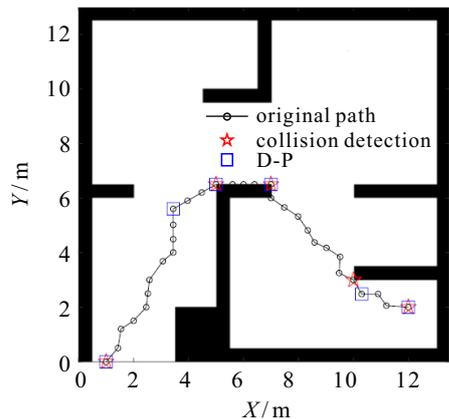


图8 $\epsilon = 0.7$ 时, 路标点提取对比

由图7可见: 两种算法均能够大幅缩减参考路径的冗余节点, 且能够保留关键特征; 碰撞检测方法提取的路标点为5个; D-P算法提取的路标点为7个, 仍有2个冗余点. 图8中: D-P算法提取的路标点为6个, 仍有1个冗余点. 图9中: D-P算法提取的路标点为5个, 但是D-P算法没有检测出障碍物的特征, 图9中的AB连线会与路障发生干涉, 即路标点提取失败. 由

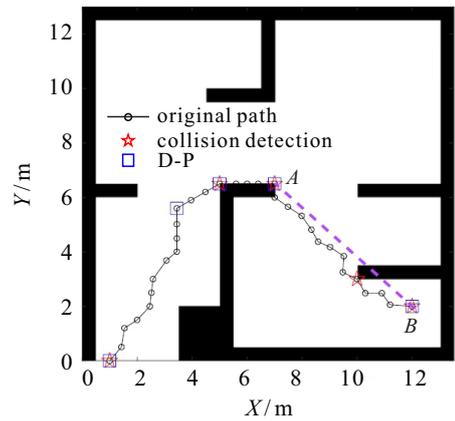


图9 $\epsilon = 0.8$ 时, 路标点提取对比

图7~图9可见, 随着最大距离 ϵ 的增加, D-P算法提取的路标点个数逐渐减少, 但是会丢失参考路径的部分重要信息, 直至无法满足机器人运动规划需求. 对于复杂地图, 更难给出最优的 ϵ 值, 只能多次尝试, 给出相对较好的 ϵ 值, 而碰撞检测算法不受参数的影响. 由以上分析可知, 碰撞检测明显优于D-P算法.

3.2 移动机器人运动规划

本节通过较复杂的室内环境地图测试所提出算法的有效性. 测试地图为 $25\text{ m} \times 20\text{ m}$, 地图和网格剖分如图10所示. 实验起始点坐标为[1, 0], 终点坐标为[18, 18]. 由图10可见, 三角网格分布均匀, 尺寸相差不大, 网格质量很好. Pure Pursuit算法的参数设置如下: 速度增益 $K_P = 0.5$, 采样时间 $T_P = 0.5\text{ s}$, 中心最大线速度 $V_M = 1\text{ m/s}$, 最大角速度 $\omega_M = 1\text{ rad/s}$.

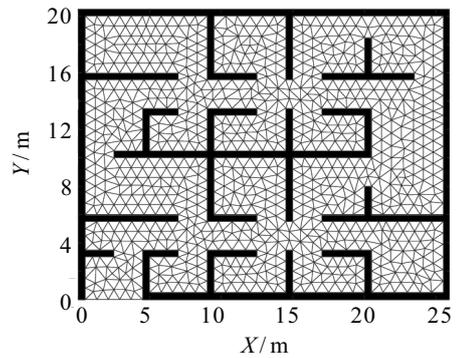


图10 复杂地图和网格剖分

3.2.1 基于碰撞检测算法的运动规划

基于碰撞检测提取的路标点如图11所示, 程序运行时间为 0.049 s . 由图11可见, 基于三角网格方法规划的参考路径能够有效避开障碍物, 找到起始点与终点的参考路径, 但是冗余节点较多, 且参考路径转折次数多, 不适合移动机器人行走. 图11中: 碰撞检测算法有效地剔除了参考路径中的冗余节点, 提取出7个参考路径节点作为机器人运动规划的路标点.

Pure Pursuit算法追踪路标点的运动轨迹如图

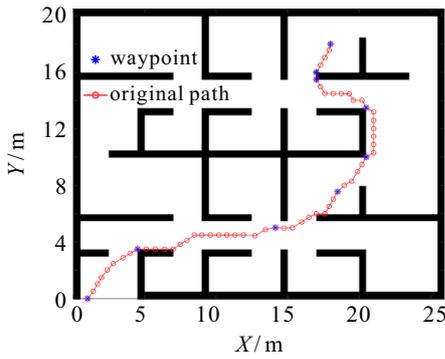


图 11 参考路径和碰撞检测路标点

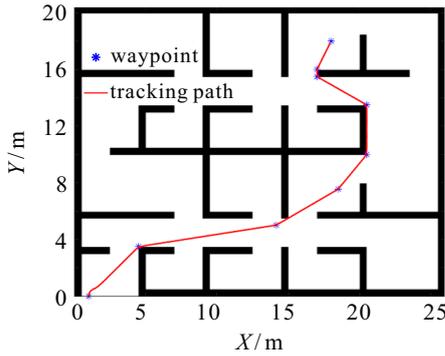


图 12 基于碰撞检测的运动规划轨迹

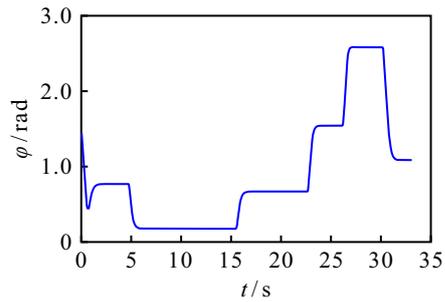
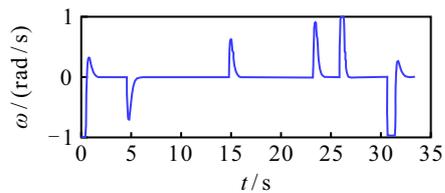
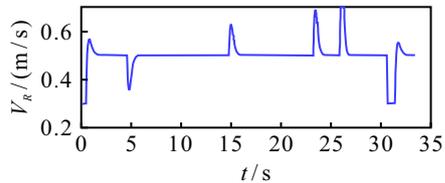


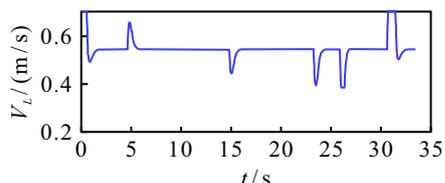
图 13 基于碰撞检测的航向角随时间变化曲线



(a) 角速度



(b) 右轮线速度



(c) 左轮线速度

图 14 基于碰撞检测的角速度、线速度随时间变化曲线

12所示,航向角变化曲线如图13所示,角速度和线速度变化曲线如图14所示.由图12~图14可见,Pure Pursuit算法得到的运动轨迹为光滑曲线(一阶导数连续),有效地避开了地图上的障碍物.由图13和图14可见,航向角亦为光滑连续曲线,在路标点位置处过度平缓,没有出现突变.由图14可见,角速度 ω 、右轮线速度 V_R 和左轮线速度 V_L 均为连续函数,变化相对平缓,只是在路标点附近出现较小的波动,波动范围均在允许值内,完全可满足机器人运行要求.由以上分析可见,基于路标点追踪的方法能够很好地完成复杂地图的机器人运动规划.

3.2.2 基于D-P算法的运动规划

基于D-P算法提取的路标点如图15所示,程序运行时间为0.052 s.由图15可见,D-P算法提取的路标点为24个,远多于碰撞检测算法提取的路标点个数.

Pure Pursuit算法追踪路标点的运动轨迹如图16所示,航向角变化曲线如图17所示,角速度和线速度

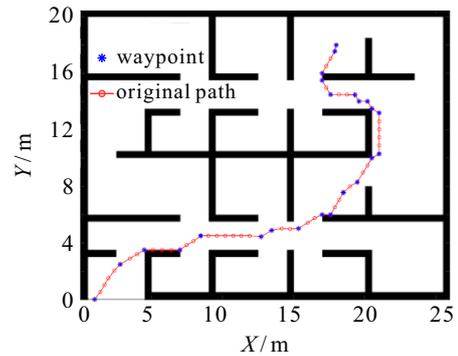


图 15 参考路径和D-P算法路标点

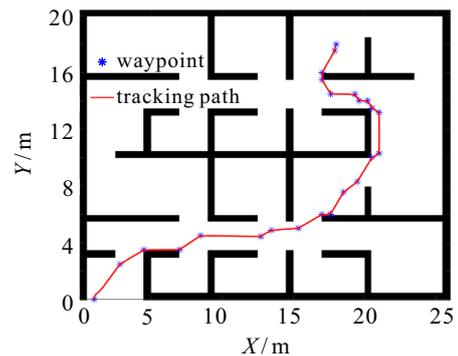


图 16 基于D-P算法的运动规划轨迹

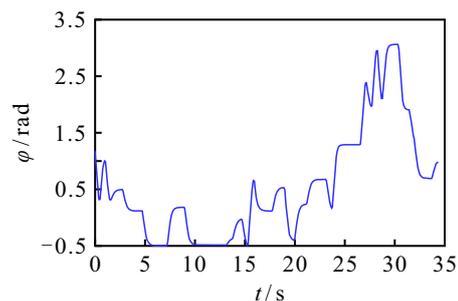


图 17 基于D-P算法的航向角随时间变化曲线

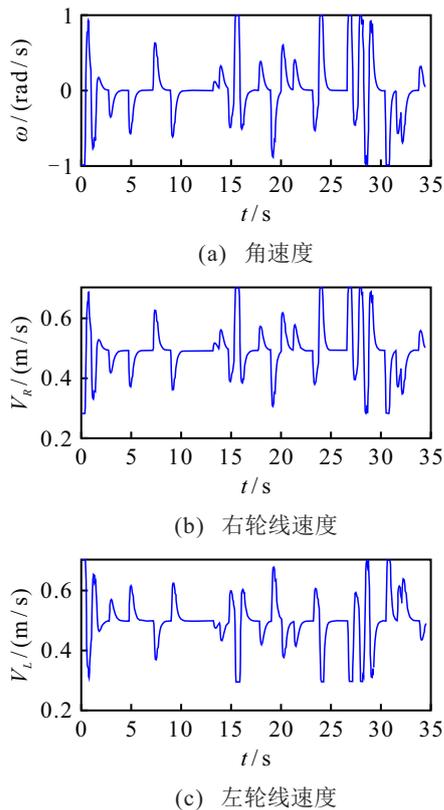


图 18 基于D-P算法的角速度、线速度随时间变化曲线

变化曲线如图18所示. 由图16可见, Pure Pursuit算法可对数量较多的路标点进行追踪, 能够得到光滑运动轨迹. 与图12对比, 基于D-P算法规划的运动轨迹拐点较多, 路径较长.

对比图13与图17可见, 图17中的航向角的变化频率和波动范围较大. 对比图14与图18可见, 图18中的角速度、线速度变化频繁, 机器人需要不断地调整左右轮转速. 通过对比可知, 在运动轨迹规划时, 基于碰撞检测算法的路标点追踪明显优于基于D-P算法的路标点追踪.

4 结论

本文提出了基于路标点跟踪的移动机器人运动规划方法. 采用碰撞检测方法改进了TMM路径规划中的路标点提取效果. 针对四轮差速驱动机器人, 采用Pure Pursuit算法跟踪路标点的方法完成运动规划. 研究表明:

1) 在提取路标点时, 碰撞检测方法明显优于D-P算法.

2) Pure Pursuit算法追踪路标点方法规划的运动轨迹是一条数学意义的光滑曲线, 能够有效地避开地图上的障碍物.

3) 机器人角速度 ω 、线速度 V_R 以及 V_L 均为光滑函数, 变化相对平缓, 只是在路标点附近出现较小的

波动, 波动范围均在允许值内, 完全可满足机器人控制要求.

4) 基于路标点追踪的运动规划程序运行速度快, 整体运行时间为0.049s, 完全能够满足工程需要.

5) 基于路标点追踪的差速移动机器人运动规划是一种简单且有效的方案.

参考文献(References)

- [1] Elbanhawi M, Simic M. Sampling-based robot motion planning: A review[J]. IEEE Access, 2014, 2: 56-77.
- [2] Tzafestas S G. Mobile robot control and navigation: A global overview[J]. Journal of Intelligent & Robotic Systems, 2018, 91(1): 35-58.
- [3] Lashkari N, Biglarbegian M, Yang S X. Development of novel motion planning and controls for a series of physically connected robots in dynamic environments[J]. Journal of Intelligent & Robotic Systems, 2019, 95(2): 291-310.
- [4] Ge S, Cui Y. Dynamic motion planning for mobile robots using potential field method[J]. Autonomous Robots, 2002, 13: 207-222.
- [5] Maekawa T, Noda T, Tamura S, et al. Curvature continuous path generation for autonomous vehicle using B-spline curves[J]. Computer-Aided Design, 2010, 42(4): 350-359.
- [6] Bhattacharya P, Gavrilova M L. Roadmap-based path planning-using the voronoi diagram for a clearance-based shortest path[J]. IEEE Robotics & Automation Magazine, 2008, 15(2): 58-66.
- [7] Jan G E, Sun C C, Tsai W C, et al. An $O(n \log n)$ shortest path algorithm based on delaunay triangulation[J]. IEEE/ASME Transactions on Mechatronics, 2014, 19(2): 660-666.
- [8] Elbanhawi M, Simic M, Jazar R. Autonomous robots path planning: An adaptive roadmap approach[J]. Applied Mechanics and Materials, 2013, (373/374/375): 246-254.
- [9] Cowlagi R V, Tsiotras P. Multiresolution motion planning for autonomous agents via wavelet-based cell decompositions[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2012, 42(5): 1455-1469.
- [10] Pun-Cheng L S C, Tang M Y F, Cheung I K L. Exact cell decomposition on base map features for optimal path finding[J]. International Journal of Geographical Information Science, 2007, 21(2): 175-185.
- [11] Rasekhipour Y, Khajepour A, Chen S K, et al. A potential field-based model predictive path-planning controller for autonomous road vehicles[J]. IEEE Transactions on Intelligent Transportation Systems, 2017, 18(5):

- 1255-1267.
- [12] García M A P, Montiel O, Castillo O, et al. Optimal path planning for autonomous mobile robot navigation using ant colony optimization and a fuzzy cost function evaluation[J]. *Analysis and Design of Intelligent Systems Using Soft Computing Techniques*, 2007, 41: 790-798.
- [13] Zou Q, Cong M, Liu D, et al. Robotic path planning based on episodic-cognitive map[J]. *International Journal of Control, Automation and Systems*, 2019, 17(5): 1304-1313.
- [14] Zhang J H, Zhang Y, Zhou Y. Path planning of mobile robot based on hybrid multi-objective bare bones particle swarm optimization with differential evolution[J]. *IEEE Access*, 2018, 6: 44542-44555.
- [15] Tsai C C, Huang H C, Chan C K. Parallel elite genetic algorithm and its application to global path planning for autonomous robot navigation[J]. *IEEE Transactions on Industrial Electronics*, 2011, 58(10): 4813-4821.
- [16] Kumar S. Generalized sampling-based feedback motion planners[M]. Texas College Station: Texas A & M University, 2011: 22-34.
- [17] 孙辉辉, 胡春鹤, 张军国. 移动机器人运动规划中的深度强化学习方法[J]. *控制与决策*, 2021, 36(6): 1281-1292.
(Sun H H, Hu C H, Zhang J G. Deep reinforcement learning for motion planning of mobile robots[J]. *Control and Decision*, 2021, 36(6): 1281-1292.)
- [18] 范俊岩, 褚雨, 岳迪, 等. 一种基于免疫机理的确定性移动机器人路径规划算法[J]. *控制与决策*, 2021, 36(10): 2418-2426.
(Fan J Y, Chu Y, Yue D, et al. A path planning algorithm of deterministic mobile robot based on immune mechanism[J]. *Control and Decision*, 2021, 36(10): 2418-2426.)
- [19] Marble J D, Bekris K E. Asymptotically near-optimal planning with probabilistic roadmap spanners[J]. *IEEE Transactions on Robotics*, 2013, 29(2): 432-444.
- [20] Kim J, Woo S H. Reference test maps for path planning algorithm test[J]. *International Journal of Control, Automation and Systems*, 2018, 16(1): 397-401.
- [21] 郑维, 张涛, 王洪斌, 等. 分级随机采样弱随机RRT算法及在移动机器人运动规划中的应用[J]. *计量学报*, 2021, 42(9): 1172-1181.
(Zheng W, Zhang T, Wang H B, et al. Hierarchical random sampling weak random RRT algorithm and application for motion planning of mobile robot[J]. *Acta Metrologica Sinica*, 2021, 42(9): 1172-1181.)
- [22] 谭建豪, 潘豹, 王耀南, 等. 基于改进RRT*FN算法的机器人路径规划[J]. *控制与决策*, 2021, 36(8): 1834-1840.
(Tan J H, Pan B, Wang Y N, et al. Robot path planning based on improved RRT*FN algorithm[J]. *Control and Decision*, 2021, 36(8): 1834-1840.)
- [23] Liu Y B, Jiang Y Y. Robotic path planning based on a triangular mesh map[J]. *International Journal of Control, Automation and Systems*, 2020, 18(10): 2658-2666.
- [24] Ahn J, Shin S, Kim M, et al. Accurate path tracking by adjusting look-ahead point in pure pursuit method[J]. *International Journal of Automotive Technology*, 2021, 22(1): 119-129.
- [25] Yu L L, Yan X X, Kuang Z X, et al. Driverless bus path tracking based on fuzzy pure pursuit control with a front axle reference[J]. *Applied Sciences*, 2019, 10(1): 230.
- [26] Wang L H, Liu M J. Path tracking control for autonomous harvesting robots based on improved double arc path planning algorithm[J]. *Journal of Intelligent & Robotic Systems*, 2020, 100(3/4): 899-909.
- [27] Wu X, Angeles J, Zou T, et al. Receding-horizon vision guidance with smooth trajectory blending in the field of view of mobile robots[J]. *Applied Sciences*, 2020, 10(2): 676-684.

作者简介

刘延彬(1974—), 男, 副教授, 博士, 从事机器人自主导航、环境感知等研究, E-mail: liuyanbin1974@126.com;
姜媛媛(1982—), 女, 教授, 博士, 从事故障诊断、环境感知等研究, E-mail: jyyl672@sina.com.