

基于最优控制策略的复杂环境移动机器人轨迹规划

张泮虹 倪涛 赵亚辉 翟海阳 赵泽仁
(燕山大学车辆与能源学院, 秦皇岛 066000)

摘要: 复杂环境下移动机器人轨迹规划由于障碍物放置杂乱且无规律,常常面临避障失败的问题。本文将机器人的轨迹规划归结为优化问题,提出了一种基于优化策略的轨迹规划方法。该方法包括3部分:首先,对优化问题的约束建模,包括机器人的运动学模型、变量极值约束、障碍物避碰模型;然后,建立优化求解策略,通过决策变量区间均分、内置插值点和基于拉格朗日多项式的变量描述方式进行离散化,针对离散化导致的约束失效对变量进行等距时间离散并建立惩罚函数,从而实现有效避障;最后,基于随机分形搜索算法对上述优化模型进行求解。仿真结果表明,本文所述方法可以有效解决移动机器人在复杂环境下的障碍物避碰问题。

关键词: 移动机器人; 复杂环境; 轨迹规划; 障碍物避碰; 优化模型

中图分类号: TG156 文献标识码: A 文章编号: 1000-1298(2022)07-0414-08 OSID: 

Optimal Control-based Trajectory Planning Method of Mobile Robot in Complex Environment

ZHANG Panhong NI Tao ZHAO Yahui ZHAI Haiyang ZHAO Zeren
(School of Vehicle and Energy, Yanshan University, Qinhuangdao 066000, China)

Abstract: As autonomous mobile machines become more intelligent, the trajectory planning of mobile robots in complex environments often faces the problem of obstacle avoidance failure due to the cluttered and irregular placement of obstacles. Trajectory planning of robot was reduced to optimization problem, that was, the objective function was defined, and then the constraint conditions were set according to the actual planning requirements of the mobile robot, and the appropriate solver was selected. Firstly, the constraint modeling of optimization, including the robot's kinematics, geometric model, variable extremum constraint and obstacle collision avoidance model were considered. Then an optimization strategy was established to discretize variables by means of interval equipartition of variables, built-in interpolation points and variable description method based on Lagrange polynomial. And for the constraint failure caused by discretization, the variables were discretized by equidistance time and the penalty function was established, so as to realize effective obstacle avoidance. Finally, the stochastic fractal search algorithm was used to solve the above optimization problem. The simulation results showed that the method described can effectively solve the obstacle avoidance problem of mobile robots in complex environments. Also both satisfied the constraints of maximum speed and maximum steering angle. Compared with the existing classical algorithm in the simulation, the experimental results showed that the algorithm described had good robustness in the narrow environment with many obstacles.

Key words: mobile robot; complex environment; trajectory planning; collision avoidance; optimization model

0 引言

自主移动机器人主要包括前端感知、中间规划、终端控制3部分^[1-3]。规划层主要是为移动机器人

计算出一条由起点到终点的无障碍的可行路径^[4],是影响移动机器人移动行为质量的最直接因素。

目前,路径规划方法主要包括图搜索、势场、随机采样、曲线插值、机器学习以及优化方法^[5-6]。

图搜索的方法是对环境进行网格化建模,然后在图中搜索符合任务要求的路径。在对建模要求精度不高的条件下,基于 A* 的图搜索^[7-8]能够快速生成质量较高的移动路径,对于建模精度要求较高的运动规划任务需要对图搜索获得的粗糙路径进行精细化处理之后才能用于底层的控制环节^[9]。

势场方法^[10]是对环境中的障碍物和非障碍物建模为斥力场和引力场,通过对势场函数进行改进^[11-12]可适应不同的应用环境。基于势场的算法是^[13-15]通过修改传统人工势场法斥力场函数获得平顺且安全的无碰撞路径,但是易陷入局部最优^[16]。

随机采样的方法是在环境中随机生成一系列采样点,然后筛选出满足任务要求的序列作为路径。相比于图搜索,随机采样可以处理较高维度的运动规划问题,具有概率完备性。主要包括概率路标算法 (PRM)^[17] 和快速搜索随机树算法 (RRT)^[18],其中 RRT 由于其规划出的路径可行性强而应用广泛。尽管随机采样的方法具有概率完备性,但是无法处理复杂的约束条件,其生成运动状态具有盲目性,导致最终解的不稳定,且解通常不是最优的^[19]。

曲线插值^[20]的方法是通过预设的路径点拟合生成连续性、平滑性较好的路径。常见的曲线插值方法包括杜宾曲线法、Reeds - Sheep 曲线法、多项式样条曲线拟合法,等等。该方法生成的路径一般具有可跟踪的性质,因此通常是配合其他规划方法对轨迹进行平滑处理或生成初始路径,比如王明强等^[21]用三次样条曲线获取道路基准线。

机器学习的方法是以作业需求、机器人的始末状态、环境设置等信息作为输入量,通过增强学习模型^[22]、卷积神经网络模型^[23]、隐式马尔可夫模型^[24]、高斯混合模型^[25]等进行学习,输出移动机器人的行驶路径/轨迹,然而机器学习需要大量的训练样本,并需要提前进行离线学习,不适合解决未知环境下实时性的路径规划问题^[26]。

基于优化的方法是在动态系统模型的基础上加入约束条件以及任务需求,构成了最优控制问题,以最优控制问题的形式描述移动机器人的规划任务,具有直观、准确、客观的特点,这是前述几种方法所不具备的。目前计算最优控制方法已应用于车辆泊车^[27]、避障^[28]等结构化环境的驾驶情景中,基于最优控制形式建立的模型中一般包含复杂约束条件,一些研究将这些复杂的约束条件进行简化并利用现代智能优化算法 (如粒子群^[29]、随机分形搜索^[30]

等)进行求解。从目前的研究来看,基于最优控制思想去解决轨迹生成问题主要有以下难点:用于描述车辆运动学特性的方程是非凸的;用于限制车辆和障碍物碰撞的几何约束是高度非凸且不可微;避碰约束的维度和障碍物的密度是密切相关的,障碍物较多环境下会导致规划时间急剧增加;规划过程会产生大量的局部最优。因此,在复杂环境下的轨迹规划仍是一个需要研究的问题。

本文在上述研究的基础上,提出一种基于最优控制的复杂环境下移动机器人的轨迹规划方法,首先针对优化求解问题建立运动学模型、几何模型、变量极值点约束以及避障模型等约束,通过对控制变量进行拉格朗日离散化,针对离散化阶段的约束失效进行等距时间离散并引入惩罚函数,最后通过随机分形搜索算法对所构建的优化问题进行求解。

1 优化约束模型

1.1 移动机器人的数学模型

由于机器人在复杂路面上的行驶速度较小,轮胎的侧滑影响较小,因此移动机器人的运动学模型是基于自行车模型构建的。如图 1 所示,其数学表达式为

$$\begin{cases} \frac{dx(t)}{dt} = v(t) \cos\theta(t) \\ \frac{dy(t)}{dt} = v(t) \sin\theta(t) \\ \frac{d\theta(t)}{dt} = \frac{v(t) \tan\phi(t)}{L} \end{cases} \quad (t \in [0, t_f]) \quad (1)$$

式中 t ——时间

t_f ——移动机器人整个运动过程总耗时

$(x(t), y(t))$ ——移动机器人后轮轴线中点坐标,表示机器人实时位置

$v(t)$ ——机器人后轮 (点 P) 实时线速度

$\theta(t)$ ——机器人实时方向角

$\phi(t)$ ——机器人前轮实时转向角

L ——机器人前后轮之间的纵向距离

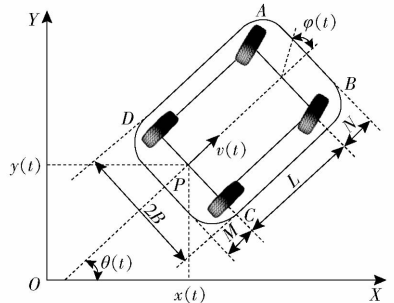


图 1 移动机器人运动学模型

Fig. 1 Kinematic model of mobile robot

此外,为了保证后续规划路径在规避障碍物时更加有效,本文在运动学建模时加入了 M 、 N 两个量,分别表示移动机器人后悬长度和前悬长度,更加符合实际情况。

从模型的数学表达式可以看出,如果机器人的实时速度 $v(t)$ 和方向角 $\theta(t)$ 已知,那么其位置 $P(x(t),y(t))$ 和前轮的实时转向角 $\phi(t)$ 均可以通过数学积分来推导得出。因此将 $\mathbf{u}=(v(t),\theta(t))$ 作为控制向量, $\mathbf{x}=(x(t),y(t),\phi(t))$ 作为状态向量。

此外,由图 1 所示的几何关系,可以得出移动机器人的几何模型,即 4 个边角点的坐标,以便于后续避障模型的搭建。

$$\begin{cases} x_A=x(t)+(L+N)\cos\theta(t)-B\sin\theta(t) \\ y_A=y(t)+(L+N)\sin\theta(t)+B\cos\theta(t) \\ x_B=x(t)+(L+N)\cos\theta(t)+B\sin\theta(t) \\ y_B=y(t)+(L+N)\sin\theta(t)-B\cos\theta(t) \\ x_C=x(t)-M\cos\theta(t)+B\sin\theta(t) \\ y_C=y(t)-M\sin\theta(t)-B\cos\theta(t) \\ x_D=x(t)-M\cos\theta(t)-B\sin\theta(t) \\ y_D=y(t)-M\sin\theta(t)+B\cos\theta(t) \end{cases} \quad (2)$$

式中 B ——前后轮之间的横向距离的一半

在移动机器人的路径规划中,最重要的原则是保证机器人避开环境中的所有障碍物;其次,也将移动机器人本身硬件所具有的最大速度和转向角加入优化条件中;此外,对移动机器人终点状态也做了相应的约束。

1.2 边界约束

在移动机器人的整个运动过程中,其线性速度不宜过高,转向角也存在极大值,即

$$\begin{cases} |v(t)|\leq v_{\max} \\ |\phi(t)|\leq \phi_{\max} \end{cases} \quad (3)$$

1.3 环境约束

环境约束主要包括移动机器人的始末状态约束以及保证机器人在移动过程中不与环境中的障碍物发生碰撞。本文将障碍物建模为矩形,通过不规则的放置来使环境的复杂程度加大。

假设 $Q(X,Y)$ 表示移动机器人的某一边角点,如图 2 所示。 A 、 B 、 C 、 D 分别表示矩形障碍物的 4 个边角点,设矩形的几何尺寸为 $m\times n$,其中, $\mathbf{l}_{AQ}=(X-m,Y-n)$, $\mathbf{l}_{AD}=(-m,0)$, $\mathbf{l}_{AB}=(0,-n)$ 。

若点 Q 在障碍物矩形区域内,即说明 $0<X<n$ 和 $0<Y<m$ 。由此可得到关系式

$$\begin{cases} 0<\mathbf{l}_{AQ}\mathbf{l}_{AB}<\mathbf{l}_{AB}\mathbf{l}_{AB} \\ 0<\mathbf{l}_{AQ}\mathbf{l}_{AD}<\mathbf{l}_{AD}\mathbf{l}_{AD} \end{cases} \quad (4)$$

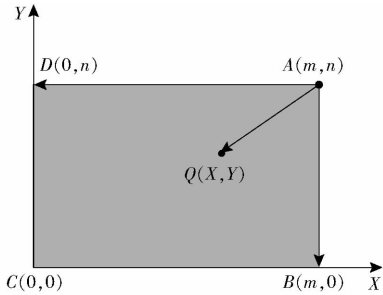


图 2 避障模型构建

Fig. 2 Obstacle avoidance model construction

本文所述算法对移动机器人的终点状态做了相应的约束,在实际应用中,根据移动机器人不同的任务需求其终点状态也可以做相应的改变。本文定义机器人的终点是以 $v(t_f)=0$ 的状态进入到预先定义好的矩形框区域内。由于凸集特性,本文所定义的终端条件等效于移动机器人的 4 个角点位于终端矩形框内部的约束。移动机器人终端约束类似于矩形障碍物约束,此处不再赘述。

2 优化求解策略

本文将路径规划问题归结为一个最优控制问题,主要包括离散化和目标函数设计两部分。

2.1 离散化阶段

将 $[0,t_f]$ 区间平均分成 N_f 份,每一个子区间为 $[t_{i-1},t_i](i=1,2,\cdots,N_f)$ 。如图 3 所示,在每个子区间内设置 $K+1$ 个插值点 $\{z_{i0},z_{i1},\cdots,z_{iK}\}$,并通过拉格朗日多项式来描述控制变量,以 $v(t)$ 为例,即

$$v(t_{i-1}+(t_i-t_{i-1})\tau)=\sum_{j=0}^K\left(z_{ij}\prod_{k=0,k\neq j}^K\frac{\tau-\tau_k}{\tau_j-\tau_k}\right) \quad (5)$$

其中, $\tau\in[0,1]$, τ_i 表示高斯点,当 K 确定时可以离线计算。因此当插值点足够多时,就能很好地表示控制变量 $v(t)$ 和 $\phi(t)$ 。此外,为了保证子区间之间变量的连续性,需要保证前后区间端点处的插值点相等,即

$$\sum_{j=0}^K\left(z_{(i-1)j}\prod_{k=0,k\neq j}^K\frac{1-\tau_k}{\tau_j-\tau_k}\right)=\sum_{j=0}^K\left(z_{ij}\prod_{k=0,k\neq j}^K\frac{0-\tau_k}{\tau_j-\tau_k}\right) \quad (6)$$

即 $z_{iK}=z_{(i+1)0}$ (7)

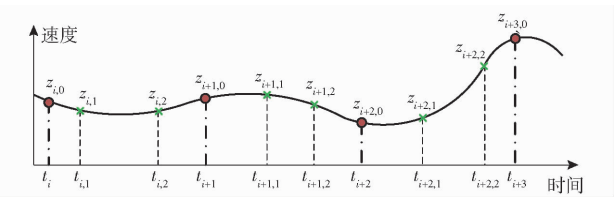


图 3 拉格朗日离散化

Fig. 3 Lagrange discretization

在用插值点描述控制变量之后, 移动机器人的状态变量就可以通过数值积分的方式进行计算, 如图 4 所示, 在数值积分之前, 把前述的分段拉格朗日多项式等距时间均分成 N_{sp} (足够大) 份, 即 $v(t)$ 可以表示为 $\{v_0, v_1, \dots, v_{N_{sp}-1}\}$ 。

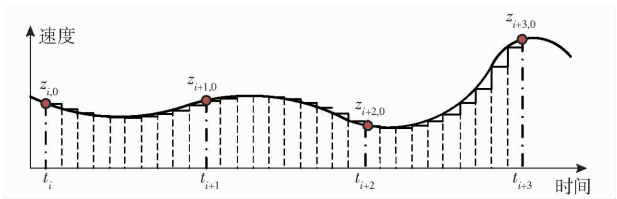


图 4 等距时间离散化
Fig. 4 Isometric time discretization

2.2 目标函数设计

上一阶段仅仅是将连续变量通过拉格朗日插值的形式进行离散化, 而第 1 节所述的约束条件也仅仅是针对插值点, 但是插值点之间的时刻并没有被约束, 这将引起如下问题: 如图 5 所示, 即便插值点 1、2、3 均满足控制变量约束条件, 但由于控制变量的连续性, 也仍然会存在某些不符合约束条件的时刻。此外, 如图 6 所示, 尽管在离散化情况下机器人的运动轨迹满足了障碍物约束, 但在连续时间内, 图示箭头所指机器人运动的下一时刻将会和障碍物发生碰撞, 从而导致任务失败。

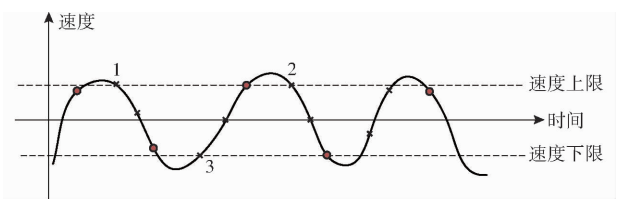


图 5 连续变量越界
Fig. 5 Continuous variable out of bounds

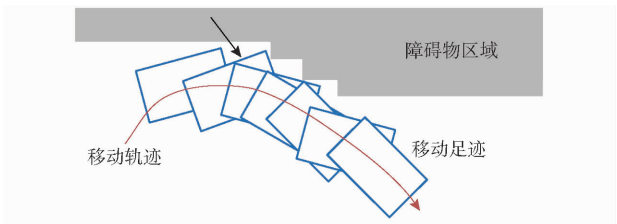


图 6 连续变量避障失效示意图
Fig. 6 Continuous variable obstacle avoidance failure

针对如图 5 所示的问题, 在数值积分之前, 将重新均分的各个离散化点, 即集合 $\{v_0, v_1, \dots, v_{N_{sp}-1}\}$ 中的每个元素都进行边界约束检查, 若 v_i 超出边界线, 则令 $v_i = v_{\max}$ 或者令 $v_i = v_{\min}$, 将有效减少控制变量超出边界的情况。

针对图 6 所示的问题, 本文在目标函数设计中加入了惩罚函数, 对于机器人运动过程中避障, 基于式(4), 得到障碍物的惩罚值

$$Q_c = \min((l_{AB}l_{AB} - l_{AQ}l_{AB}), l_{AQ}l_{AB}, (l_{AD}l_{AD} - l_{AQ}l_{AD}), l_{AQ}l_{AD}) \tag{8}$$

式(8)保证了点 Q 是以最小的代价位于障碍物外的, 对于移动机器人终端约束也可以用类似的思想, 以保证最终机器人位于终点所定义的矩形框内。

因此, 最终惩罚项

$$Q = \begin{cases} 0 & (Q_c + Q_t = 0) \\ N + Q_c + Q_t & (Q_c + Q_t > 0) \end{cases} \tag{9}$$

为了将求解轨迹的可行解与不可行解区分开, 本文引入一个足够大的定值 N 来保证可行解的惩罚值绝对小于不可行解的惩罚值。显然, 若路径能保证无障碍和满足最终状态, 那么其惩罚值为 0。

3 优化求解器

根据前文所述, 将离散化的插值点和移动机器人的运动时间 t_f 作为决策变量。然后, 通过积分得到状态变量, 检查其边界约束, 并将障碍物惩罚函数加入优化准则中。考虑到元启发算法具有全局优化的能力, 本文采用了随机分形搜索算法^[31]作为整个优化过程的求解器。

分形指的是具有自相似性的现象、图像或者物理过程等, 随机分形表现在结构或复杂度上具有统计意义上的自相似性, 一般是通过莱维飞行、高斯游走等随机规则来产生。随机分形主要包括扩散和更新两个过程, 扩散过程是基于粒子的当前位置, 因此有利于寻找全局最优解。

首先是种群的初始化, 在这个过程中, 基于约束条件随机初始化种群中的每一个个体, 第 j 个个体的初始化方程为

$$P_j = L_B + \varepsilon(U_B - L_B) \tag{10}$$

其中, L_B 和 U_B 分别是求解问题向量的上下边界, 本文指的是控制变量转向角和速度的上下限值, ε 是在区间 $[0, 1]$ 上服从均匀分布的随机数。

初始化种群之后, 计算个体的适应度函数以获得最佳个体 B_p , 然后所有个体都围绕当前的位置游走以搜索环境空间, 随机搜索算法采用高斯游走作为唯一的游走方式, 即

$$\begin{cases} G_1 = \text{Gaussian}(\mu_{Bp}, \sigma) + (\varepsilon B_p - \varepsilon' P_i) \\ G_2 = \text{Gaussian}(\mu_p, \sigma) \end{cases} \tag{11}$$

式中 P_i ——第 i 个个体的位置
 $\varepsilon, \varepsilon'$ ——区间 $[0, 1]$ 上服从均匀分布随机数
 μ_{Bp}, μ_p ——高斯函数均值
 σ ——高斯参数标准差

$\mu_{BP} = |B_p|, \mu_p = |P_i|$, 高斯参数中的标准差为

$$\sigma = \left| \frac{\lg g}{g} (P_i - B_p) \right| \tag{12}$$

第 1 次更新过程, 首先根据个体的适应度进行排序, 然后赋予每个个体性能级别, 个体适应度为

$$P_{a_i} = \frac{\text{rank}(P_i)}{N} \tag{13}$$

式中 N ——种群的个体数量

$\text{rank}(P_i)$ ——个体 P_i 在种群中的排序
之后判断是否满足 $P_{a_i} < \varepsilon$, 若满足则更新个体 P_i 的第 j 个分量, 否则保持不变, 更新公式为

$$P'_i(j) = P_r(j) - \varepsilon(P_i(j) - P_i(j)) \tag{14}$$

式中 P'_i ——个体 P_i 更新后的位置

P_r, P_i ——种群中随机选择的个体

第 2 次更新过程主要是根据个体的分量进行, 对于更新后的 P'_i , 判断是否满足 $P'_{a_i} < \varepsilon$, 若满足则更新 P'_i 的当前位置, 否则保持不变, 更新公式为

$$P''_i = \begin{cases} P'_i - \hat{\varepsilon}(P'_i - B_p) & (\varepsilon' \leq 0.5) \\ P'_i + \hat{\varepsilon}(P'_i - P_r) & (\varepsilon' > 0.5) \end{cases} \tag{15}$$

式中 P'_r, P'_i ——第 1 次更新后种群中选择的个体

$\hat{\varepsilon}$ ——高斯正态分布生成的随机数
若 P''_i 对应的适应度优于 P'_i 对应的适应度, 则用 P''_i 更新替换掉 P'_i 。

4 仿真实验

4.1 实验设置

仿真实验在 Matlab 2019b 中展开, 计算机环境是 Windows10, 处理器 Intel (R) Core (TM) i7 - 10870H CPU @ 2.20 GHz 2.21 GHz, RAM 16 GB, 本文所述算法中提到的一些参数设置如表 1 所示。为了验证本文所述算法的有效性, 鉴于现在大多数的规划算法均是机器人建模为质点, 与真实情况相差较大, 在本仿真实验中, 引入了“足迹”的概念, 即表示各个时刻移动机器人在地面上的投影, 以足迹和障碍物不发生交集来作为避障有效的判定, 如 1.3 节所述。将建好的模型经拉格朗日离散化处理, 然后代入 SFS 算法中进行迭代求解, 最终得到最优解。

4.2 障碍物较少场景验证

为验证本文所述算法的有效性, 场景 1 随意设置 10 个矩形障碍物, 用户可根据环境或者任务不同随意指定起点, 本文取 $x_1(0) = -40 \text{ m}, y_1(0) = 15 \text{ m}$, 终点设置在以 $(0, 0)$ 为中心, 长为 2.5 m、宽为 1.5 m 的终端盒子内。初始方位角为 0° , 初始转向角为 0° , 初始速度为 0。从图 7a 可以看出, 所规划的轨迹可以实现准确避障, 即说明优化模型中的避

表 1 基于优化算法的参数

Tab. 1 Parameter based on optimization algorithm	
参数	数值
前悬长度 N/m	0.25
后悬长度 M/m	0.25
前后轮纵向距离 L/m	1.50
前后轮横向距离一半 B/m	2.00
线速度最大值 $v_{\max}/(\text{m} \cdot \text{s}^{-1})$	3.0
前轮转角最大值 $\phi_{\max}/\text{rad}$	0.714
拉格朗日离散区间数 N_{fp}	5
插值点个数 K	3
高斯插值点 $[\tau_0, \tau_1, \tau_2, \tau_3]$	$[0, 0.16, 0.65, 1]$
等距时间离散区间数 N_{sp}	1 000
SFS 算法中种群数 S_N	20
SFS 算法中最大迭代数 M_{CN}	10 000
SFS 算法中最大数 M_{DN}	1
违反约束惩罚值 N	1 000

障模型是有效的, 此外终端约束也满足, 图 7b 中的速度最大值控制在 3.0 m/s , 前轮转角控制在 0.714 rad 内, 即边界约束有效, 同时可以看出移动机器人的足迹和障碍物矩形的交集为空, 因此针对离散化导致的避障失效问题也得到了有效解决。

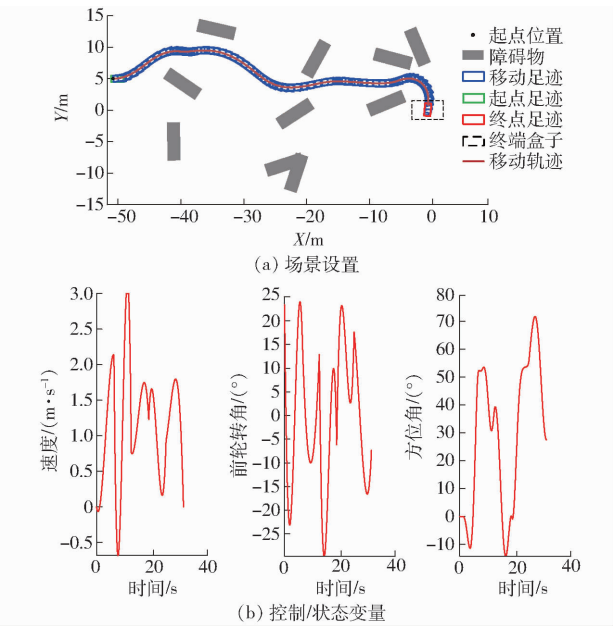


图 7 障碍物较少场景

Fig. 7 Scenarios with fewer obstacles

4.3 狭窄区域场景验证

场景 2 设置了 10 个障碍物并设置了一个狭窄区域, 起点设置为 $x_2(0) = -50 \text{ m}, y_2(0) = 5 \text{ m}$, 终点设置在以 $(0, 0)$ 为中心, 长为 2.5 m、宽为 1.5 m 的终端盒子内。初始方位角为 0° , 初始转向角为 0° , 初始速度为 0。从图 8a 可以看出, 所规划的轨迹可以实现准确避障, 即说明优化模型中的避障模型是有效的, 此外终端约束也严格满足, 图 8b 中的

速度最大值控制在 3.0 m/s, 前轮转角控制在 0.714 rad 内, 即边界约束有效, 同时可以看出移动机器人的足迹和障碍物矩形的交集为空, 因此针对离散化导致的避障失效问题也得到了有效的解决, 因此本文所述方法在狭窄区域内仍具有一定的稳定性。

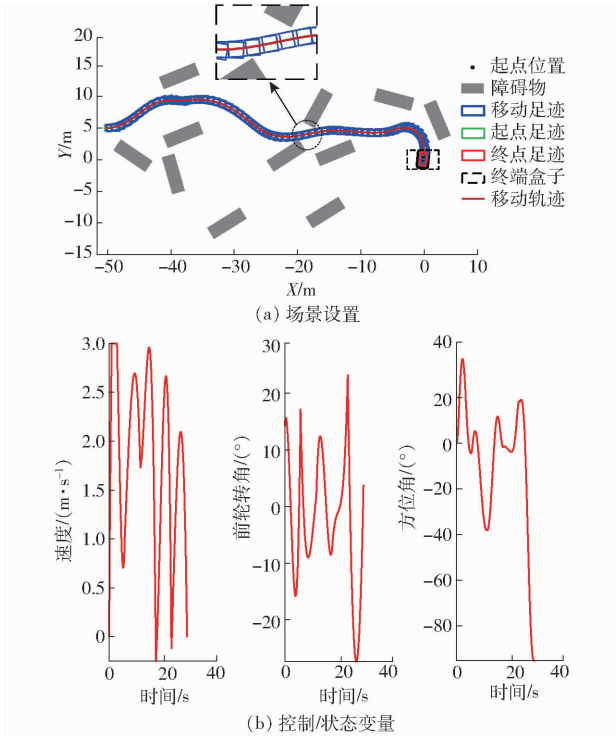


图 8 狭窄区域场景

Fig. 8 Narrow area scenario

4.4 复杂场景验证

场景 3 设置了 30 个障碍物矩形区域, 起点设置为 $x_3(0) = -40\text{ m}$, $y_3(0) = 0\text{ m}$, 终点设置在以 (0,0) 为中心, 长为 2.5 m、宽为 1.5 m 的终端盒子内。初始方位角为 0°, 初始转向角为 0°, 初始速度为 0。从图 9a 可以看出, 所规划的轨迹可以实现准确避障, 即说明优化模型中的避障模型是有效的, 此外严格满足终端约束, 图 9b 中的速度最大值控制在 3.0 m/s, 前轮转角控制在 0.714 rad 内, 即边界约束有效, 同时可以看出移动机器人的足迹和障碍物矩形的交集为空, 因此针对离散化导致的避障失效问题也得到了有效的解决, 在复杂环境下进一步证明了本文所述方法具有较强的鲁棒性。

4.5 仿真实验对比分析

采用 Matlab 随机生成了 10 个场景, 由于混合 A* 算法考虑了机器人的运动学模型, 而且算法中也加入了非线性优化和非参数插值, 因此将本文所述优化算法与混合 A* 算法作比较, 结果如表 2 所示, 成功率是指移动机器人足迹不与矩形障碍物产生交集, 如图 10 虚线圆处所示, 计算第 4 障碍物和第 6

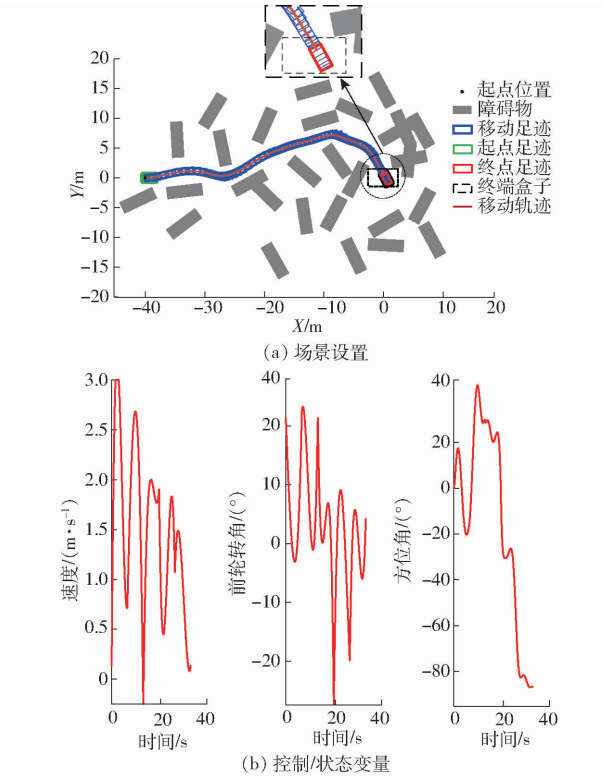


图 9 障碍物较多场景

Fig. 9 Scenarios with many obstacles

障碍物与混合 A* 算法所规划轨迹的最短距离, 分别为 0.312 m 和 0.965 m, 而本文设定的移动机器人的宽为 2 m, 长为 4 m, 也就说明移动机器人在实际移动中会与障碍物发生碰撞, 导致避障失效, 即混合 A* 算法在狭窄区域内并不有效, 同时, 基于优化算法所得到的轨迹较短, 时间也相对较小, 效率更高。这里的轨迹长度是对求得的离散轨迹点求和, 即

$$L = \sum_{i=2}^N \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (16)$$

表 2 本文算法与混合 A* 算法性能对比

Tab. 2 Performance comparison between proposed algorithm and hybrid A* algorithm			
算法	轨迹长度/m	CPU 运行时间/s	成功率/%
基于优化算法	56.027 9	8.9	100
混合 A* 算法	66.824 6	17.6	80

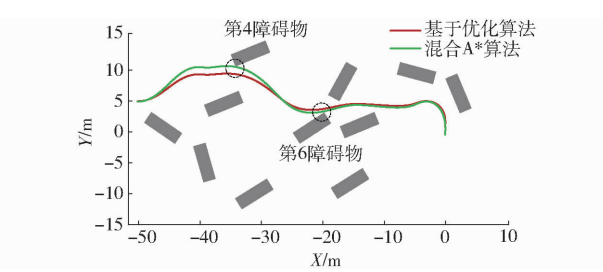


图 10 本文算法与混合 A* 算法运行轨迹

Fig. 10 Proposed algorithm and hybrid A* algorithm running track

此外,与现有文献中的基于优化的规划算法比较,用于验证本文所提算法的求解质量。选取文献[32]中提到的优化算法作为基准,用于验证其余算法的最优性损失,假设算法 X 以代价值 c_X 取得最优解,文献[33]算法以 c_{baseline} 为代价值取得最优解,那么算法 X 的最优性损失计算式为^[34]

$$r_X = \frac{c_X - c_{\text{baseline}}}{c_{\text{baseline}}} \times 100\%$$

(17)

很明显最优性损失越小,说明求解质量越高。通过计算,本文所述算法最优性损失为 19.72%,文献[32]算法为 45.1%,文献[33]算法为 34.61%,因此本文所述算法在轨迹规划的求解上质量更高,输出的轨迹具有更好的性能。

此外,验证 SFS 算法中迭代次数对于求解结果的影响。选择场景 1,对不同迭代次数下求得的目标函数值作比较,如表 3 所示,从表 3 可以看出,迭代次数在大于 500 以后,目标函数值的降低率趋缓,因此迭代次数并不是越大越好,选择合适的迭代次数会增加求解效率。

表 3 不同迭代次数 SFS 算法的目标值

Tab.3 Target values of different iterations of SFS algorithm

最大迭代次数	最优目标函数值
100	3 217.77
200	1 961.13
300	1 959.91
400	1 295.81
500	1 121.40
600	1 023.82
700	1 022.44
800	1 011.10
900	1 010.30
1 000	1 009.50

5 结论

- (1)将轨迹规划转化为最优控制问题,通过设置控制变量极值、障碍物避碰、起点终点状态等优化限制条件,采用随机分形搜索算法的元启发策略进行求解,得到了移动机器人精确的移动轨迹,本文所述方法在复杂、狭窄区域均具有较好的表现。
- (2)通过等距时间离散,解决了由离散化导致的离散点之间约束失效的问题,Matlab 仿真实验显示移动机器人的线速度不超过最大速度 3.0 m/s,转向角不超过最大转角 0.714 rad,均满足变量的边界约束。

参 考 文 献

[1] 李柏,张友民,邵之江. 自动驾驶车辆运动规划方法综述[J]. 控制与信息技术,2018(6):1-6.
LI Bai, ZHANG Youmin, SHAO Zhijiang. A review of motion planning methods for autonomous vehicles[J]. Control and Information Technology, 2018(6):1-6. (in Chinese)

[2] 孙振平. 自动驾驶汽车智能控制系统[D]. 长沙:国防科学技术大学,2004.
SUN Zhenping. Autonomous driving vehicle intelligent control system [D]. Changsha: National University of Defense Technology,2004. (in Chinese)

[3] GONZALEZ D, PEREZ J, MILANES V. A review of motion planning techniques for automated vehicles[J]. IEEE Transactions on Intelligent Transportation Systems, 2016, 17(4):1135-1145.

[4] BRIAN P, MICHAL C, ZHENG Y S, et al. A survey of motion planning and control techniques for self-driving urban vehicles [J]. IEEE Transactions on Intelligent Vehicles, 2016, 1(1):33-55.

[5] 冯来春. 基于引导域的参数化 RRT 无人驾驶车辆运动规划算法研究[D]. 合肥:中国科学技术大学,2017.
FENG Laichun. Research on parameterized RRT motion planning algorithm for unmanned vehicle based on guidance domain [D]. Hefei: University of Science and Technology of China, 2017. (in Chinese)

[6] 周俊,何永强. 农业机械导航路径规划研究进展[J]. 农业机械学报,2021,52(9):1-14.
ZHOU Jun, HE Yongqiang. Research progress on navigation path planning of agricultural machinery[J]. Transactions of the Chinese Society for Agricultural Machinery,2021,52(9):1-14. (in Chinese)

[7] PETER E H, NILS J N, BERTRAM R. A formal basis for the heuristic determination of minimum cost paths[J]. IEEE Transactions on Systems Science and Cybernetics, 1968, 4(2):100-107.

[8] DMITRI D, SEBASTIAN T, MICHEAL M, et al. Practical search techniques in path planning for autonomous driving[C]//4th International Conference on Autonomous Robots and Agents, 2009.

[9] 段书用,王启帆,韩旭,等. 具有确保安全距离的 A* 路径优化方法[J]. 机械工程学报, 2020,56(18):206-215.
DUAN Shuyong, WANG Qifan, HAN Xu,et al. Improved A-star algorithm for safety insured optimal path with smoothed corner turns[J]. Journal of Mechanical Engineering, 2020,56(18):206-215. (in Chinese)

[10] KHATIB O. Real-time obstacle avoidance system for manipulators and mobile robots[J]. The International Journal of Robotics Research, 1986, 5(1):90-98.

[11] 刘梓林,黎予生,郑玲. 基于复杂环境点云稀疏表示的无人驾驶汽车局部路径规划方法[J]. 机械工程学报, 2020, 56(2):164-173.
LIU Zilin, LI Yusheng, ZHENG Ling. Local path planning for autonomous vehicles based on sparse representation of point cloud in unstructured environments[J]. Journal of Mechanical Engineering, 2020,56(2):164-173. (in Chinese)

- [12] 张强, 陈兵奎, 刘小雍, 等. 基于改进势场蚁群算法的移动机器人最优路径规划[J]. 农业机械学报, 2019, 50(5): 24–32, 42.
ZHANG Qiang, CHEN Bingkui, LIU Xiaoyong, et al. Ant colony optimization with improved potential field heuristic for robot path planning[J]. Transactions of the Chinese Society for Agricultural Machinery, 2019, 50(5): 24–32, 42. (in Chinese)
- [13] RASEKHIPOUR Y A, KHAJEPOUR A, CHEN S, 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.
- [14] HUANG Z, CHU D, WU C, et al. Path planning and cooperative control for automated vehicle platoon using hybrid automata[J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 20(3): 959–974.
- [15] WANG H, HUANG Y, KHAJEPOUR A, et al. Crash mitigation in motion planning for autonomous vehicles[J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 20(9): 3313–3323.
- [16] 唐志荣, 冀杰, 吴明阳, 等. 基于改进人工势场法的车辆路径规划与跟踪[J]. 西南大学学报(自然科学版), 2018, 40(6): 174–182.
TANG Zhirong, JI Jie, WU Mingyang, et al. Vehicle path planning and tracking based on improved artificial potential field method[J]. Journal of Southwest University (Natural Science Edition), 2018, 40(6): 174–182. (in Chinese)
- [17] KAVRAKI L E, SVESTKA P, OVERMARS M H. Probabilistic roadmaps for path planning in high-dimensional configuration spaces[J]. IEEE Transactions on Robotics and Automation, 1996, 12(4): 566–580.
- [18] LA VALLE S M. Randomized kinodynamic planning[J]. International Journal of Robotics & Research, 1999, 15(5): 378–400.
- [19] 张卫波, 肖继亮. 改进 RRT 算法在复杂环境下智能车路径规划中的应用[J]. 中国公路学报, 2021, 34(3): 225–234.
ZHANG Weibo, XIAO Jiliang. Application of improved RRT algorithm in intelligent vehicle path planning under complex environment[J]. China Journal of Highway and Transport, 2021, 34(3): 225–234. (in Chinese)
- [20] MASONE C, MOHAMMADI M, GIORDANO P R, et al. Shared planning and control for mobile robots with integral haptic feedback[J]. International Journal of Robotics Research, 2018, 37(11): 1395–1420.
- [21] 王明强, 王震坡, 张雷. 基于碰撞风险评估的智能汽车局部路径规划方法研究[J]. 机械工程学报, 2021, 57(10): 29–41.
WANG Mingqiang, WANG Zhenpo, ZHANG Lei. Research on local path planning method of intelligent vehicle based on collision risk assessment[J]. Journal of Mechanical Engineering, 2021, 57(10): 29–41. (in Chinese)
- [22] 郑睿. 基于增强学习的无人车辆智能决策方法研究[D]. 长沙: 国防科学技术大学, 2013.
ZHENG Rui. Research on intelligent decision-making method of unmanned vehicle based on enhanced learning[D]. Changsha: National University of Defense Technology, 2013. (in Chinese)
- [23] ROBERT-ADRIAN R, KINGA B F, et al. Collision avoidance using deep learning-based monocular vision[J]. SN Computer Science, 2021, 2(5): 375.
- [24] LEFEVRE S, CARVALHO A, BORRELLI F. A learning-based framework for velocity control in autonomous driving[J]. IEEE Transactions on Automation Science and Engineering, 2016, 13(1): 32–42.
- [25] 黄如林. 无人驾驶汽车动态障碍物避撞关键技术研究[D]. 合肥: 中国科学技术大学, 2017.
HUANG Rulin. Research on key technologies of collision avoidance with dynamic obstacles in unmanned vehicles[D]. Hefei: University of Science and Technology of China, 2017. (in Chinese)
- [26] 李辉, 祁宇明. 一种复杂环境下基于深度强化学习的机器人路径规划方法[J]. 计算机应用研究, 2020, 37(增刊): 129–131.
LI Hui, QI Yuming. A robot path planning method based on deep reinforcement learning in complex environment[J]. Computer Application Research, 2020, 37(Supp.): 129–131. (in Chinese)
- [27] LI Bai, WANG Kexin, SHAO Zhijiang. Time-optimal maneuver planning in automatic parallel parking using a simultaneous dynamic optimization approach[J]. IEEE Transactions on Intelligent Transportation Systems, 2016, 17(11): 3263–3274.
- [28] GU T. Improved trajectory planning for on-road self-driving vehicles via combined graph search, optimization, and topology analysis[D]. Pittsburgh: Carnegie Mellon University, 2017.
- [29] MAC T T, COPOT C C, TRAN D T, et al. A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization[J]. Applied Soft Computing, 2017, 59(2): 68–76.
- [30] LI Bai, SHAO Zhijiang. Precise trajectory optimization for articulated wheeled vehicles in cluttered environments[J]. Advances in Engineering Software, 2016, 92: 40–47.
- [31] SALIMI H. Stochastic fractal search: a powerful metaheuristic algorithm[J]. Knowledge-Based Systems, 2015, 75: 1–18.
- [32] LI Bai, SHAO Zhijiang. A unified motion planning method for parking an autonomous vehicle in the presence of irregularly placed obstacles[J]. Knowledge-Based Systems, 2015, 86: 11–20.
- [33] JANG C, KIM C, LEE S, et al. Re-plannable automated parking system with a standalone around view monitor for narrow parking lots[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 21(2): 777–790.
- [34] ZIEGLER J, BENDER P, DANG T, et al. Trajectory planning for Bertha—a local, continuous method[C]//IEEE Intelligent Vehicles Symposium Proceedings, 2014.