Local Path Planning Algorithm with Risk Areas for Mobile Robots*

OU Xianhua, YOU Zhongnan, HE Xiongxiong*
(College of Information Engineering, Zhejiang University of Technology, Hangzhou Zhejiang 310023, China)

Abstract: In order to meet the obstacle avoidance needs of mobile robots in dynamic environments, a local path planning algorithm that integrates risk areas is proposed. This algorithm uses the obstacle speed detected by two-dimensional LiDAR to expand the risk areas and combines the risk areas with the timed elastic band (TEB) algorithm to achieve the dynamic obstacle avoidance. The specific process of the algorithm is as follows. Firstly, the LiDAR point cloud data obtained by the 2d LiDAR is preprocessed and clustered to obtain obstacle point cloud clusters. Then, the data association matrix is calculated and point cloud cluster matching is performed. At the same time, the adaptive Kalman filter is used to track the obstacle point cloud cluster to obtain the optimal estimation of the obstacle motion information. Finally, the risk areas is expanded using the obstacle's motion information, and these risk areas are added as a new constraint to the TEB pose sequence, and then the g2o optimization algorithm is used to obtain a local motion trajectory that takes the obstacle's motion information into account. Experiments in simulations and mobile robot platform prove that the proposed method can improve the obstacle avoidance efficiency of robots when facing dynamic obstacles.

Key words: 2D LiDAR; dynamic obstacle avoidance; risk areas; TEB algorithm; adaptive Kalman filter
EEACC: 6140 **doi**: 10.3969/j.issn.1004-1699.2026.09.009

融合风险区域的移动机器人局部路径规划算法*

欧县华, 游仲南, 何熊熊*
(浙江工业大学信息工程学院, 浙江 杭州 310023)

摘 要: 为了满足移动机器人在动态环境下的避障需要, 提出了一种融合风险区域的局部路径规划算法, 该算法利用二维激光雷达检测到的障碍物速度来拓展风险区域, 并将风险区域与时间弹性带算法(Timed Elastic Band, TEB)算法相结合以达到动态避障的效果。算法具体流程如下, 首先, 对二维激光雷达获得的激光点云数据进行预处理和聚类操作得到障碍物点云簇。然后, 计算数据关联矩阵并进行点云簇匹配, 同时利用自适应卡尔曼滤波对障碍物点云簇进行跟踪, 得到障碍物运动信息的最优估计。最后, 使用障碍物的运动信息对风险区域进行拓展, 并将此风险区域作为一种新约束加入到 TEB 位姿序列中, 再使用 g2o 优化算法得到一条考虑障碍物运动信息的机器人局部运动轨迹。在仿真环境和移动机器人实验平台中的实验表明, 所提出的方法能提高机器人在面对动态障碍物时的避障效率。

关键词: 二维激光雷达; 动态避障; 风险区域; TEB 算法; 自适应卡尔曼滤波
中图分类号: TN911.7; TP212.6 **文献标识码**: A **文章编号**: 1004-1699(2026)09-1987-09

随着科技的进步, 在室内环境中, 移动机器人可以替代人工进行繁重的巡逻、运输等任务, 从而解放人力并提高工作效率, 如送餐机器人、无人叉车、运输机器人等。跟随全局路径、规避障碍物并安全到达目标点是移动机器人自主导航所需要面临的挑战之一, 特别是在存在动态障碍物的环境下。在动态环境下, 机器人需要处理大量的环境信息, 做出最优的控制决策。在移动机器人进行导航的过程中, 全局路径规划算法[1-3]负责计算出全局最优路径, 局部路径规划算法在收到全局路径之后, 需要结合摄像头、激光雷达或者超声波等传感器信息, 计算出一个最优的局部路径策略, 使移动机器人在最大程度跟随全局路径的同时能精准避开障碍物并尽快到达目标点。

障碍物检测在移动机器人的自主导航中同样发挥着关键作用, 当前大部分多目标检测算法是基于图像的[4-6], 其通常需要较大的计算量, 且受光照影响较大。虽然图像能给予更多信息, 但是仅对于机器人避障来说, 优先考虑的是障碍物运动信息检测而不是障碍物识别, 因此很多信息无用, 而且图像处理会占用大量的计算资源, 造成算法的实时性变差。为了避免此问题, 本文采用二维激光雷达来检测障碍物的运动信息[7-9]。

局部路径规划算法有动态窗口法(Dynamic Window Approach, DWA)、TEB 等。DWA[10-12]是通

项目来源: 国家自然科学基金项目(62303418 和 62233016); 浙江省自然科学基金项目(LQ22F030016)
收稿日期: 2024-04-15 **修改日期**: 2024-09-25

过在速度空间中采样多组速度并模拟出这些速度在一定时间内形成的运动轨迹,再通过评价函数对这些运动轨迹进行评分,最终挑选出最优的轨迹及其对应的线速度和角速度。该算法具有计算复杂度低、实时性强的优点,但容易陷入局部最优,且只适用于差速运动模型的机器人。TEB[13-17]算法通过在起始点和目标点中插入 N 个时间间隔为 T 的机器人位姿,采用图优化算法求出最优的局部位姿序列并下发运动指令使机器人移动。以上提到的两种经典算法基本可以实现移动机器人的局部路径规划,但都没有考虑到障碍物的运动信息,它们都默认把障碍物当成是静态的,这会导致规划出的局部路径只是当前时刻最优,而这条路径却没有考虑到障碍物未来可能到达的位置,这将造成移动机器人在动态环境下的避障效果变差。

动态环境中的移动机器人避障已取得一系列研究成果。Guo 等[18]提出了一种适用与两轮差速机器人的动态避障算法,它采用双目相机检测动态障碍物,同时使用拓展卡尔曼滤波估计动态障碍物的运动状态,并通过障碍物的运动状态来拓展动态风险区域以实现动态避障,然而该方法只适用于两轮差速运动模型的机器人。文献[19]在全局路径上设立虚拟目标(Virtual Target,VT),机器人通过在动态环境中实时追踪 VT 来实现在动态环境的安全平稳运行,此文章还提出了一种全局路径重新规划方案以应对密集环境中出现的不可跟踪区域的情况。Dong 等[20]提出了一种将二维激光雷达和 TEB 算法进行结合的算法,仿真和实验证明了系统在实时检测和障碍物跟踪方面的能力,且具备一定的鲁棒性。移动机器人的最终动态避障效果不仅取决于避障算法[21],还依赖于其他系统,比如定位系统[22]、控制系统[23]和路径跟踪算法[24]等,但是该文只针对局部路径规划算法和障碍物运动信息检测算法做出融合改进。

综上所述,上面提到的一些针对机器人的动态避障算法都有考虑到障碍物的运动信息,但是却存在一些局限性,为了提高移动机器人在进行局部路径规划的时候针对动态障碍物的规避效率,本文提出了一种融合风险区域和 TEB 算法的局部路径规划算法,该算法利用二维激光雷达检测动态障碍物运动信息并拓展风险区域,最终规划出一条融合障碍物运动信息的无碰撞轨迹。算法具体流程如下,首先,通过基于条件密度的聚类算法(Conditional Density-Based Spatial Clustering of Applications with Noise,CDBSCAN)和自适应卡尔曼滤波[25]的二维点云多目标跟踪算法获取障碍物的运动信息集合。然后,结合障碍物运动信息和 TEB 位姿序列对风险区域进行拓展,并将这些区域加入到 TEB 位姿序列中,通过图优化算法得到一条考虑障碍物运动信息的 TEB 位姿序列。最后,通过一系列仿真和实际环境下的实验,验证了该方法可以有效地实现移动机器人的动态避障。

1 障碍物运动信息检测

1.1 点云聚类

二维激光雷达扫描得到的原始点云坐标由极径和极角组成,其坐标原点是雷达,表达式为(θ,r),这种形式的数据还无法满足我们对静态障碍物过滤和动态障碍物运动信息预测的需求,我们需要将其转化为地图坐标系下的坐标(x,y)再进行处理。点云坐标由(θ,r)到(x,y)的坐标转换公式如下:

$$\begin{bmatrix}x\\y\end{bmatrix}=\begin{bmatrix}\cos\theta_r & -\sin\theta_r\\ \sin\theta_r & \cos\theta_r\end{bmatrix}\times\begin{bmatrix}\cos\theta\\ \sin\theta\end{bmatrix}\times r+\begin{bmatrix}x_r\\ y_r\end{bmatrix} \tag{1}$$

式中:(x_r,y_r,θ_r)为机器人在地图坐标系下的位姿,x_r,y_r为机器人的横坐标和纵坐标,θ_r为机器人的朝向,x,y是转换之后全局点云的坐标,θ和r为极坐标系下点云点的极角和极径。

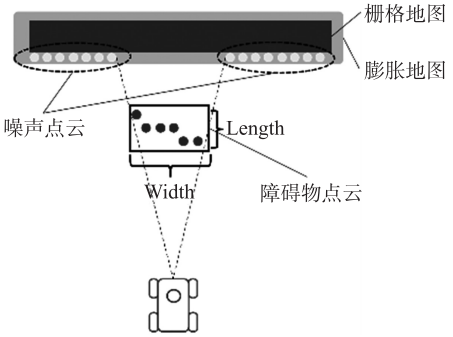


图 1 CDBSCAN 聚类得到的聚类结果

得到全局点云集合后,还需要对点云进行聚类,本文利用 CDBSCAN 算法对该点云集合进行聚类(图 1),利用已知膨胀栅格地图作为掩膜,过滤掉背景点云并聚类得到 K 个障碍物点云簇,原 DBSCAN 有两个超参数 ε 和 MinPts,其中 ε 是邻域范围,MinPts 表示该邻域范围内包含点云的最小数量,这两个超参数的设置将取决于移动机器人的宽度、行人的步长以及激光雷达的分辨率。CDBSCAN 在原算法的基础上添加了一个超参数,即膨胀之后的背景栅格地图,每当聚类算法搜索到一个点云的时候,除了判断该点云 ε 邻域内的点云数量是否小于 MinPts,还会通过这个背景栅格地图来判断该点是否是噪声点。点云经过聚类之后会得到 K 个点云簇,此外还需要计算每个点云簇在 x 方向上和 y 方向上的长度,方便后续对关联矩阵中交并比的计算。

1.2　数据关联矩阵

多障碍物的目标跟踪中数据关联矩阵的计算至关重要,因为需要保证当前帧的点云簇集合与上一帧的点云簇集合一一对应,下面将介绍本文数据关联矩阵的计算方式。首先,由点云聚类得到的障碍物点云中心集的表达式如下:

$$\boldsymbol{P}^{\text{pre}} \triangleq \{(px_i^{\text{pre}}, py_i^{\text{pre}}), i \in [1, M]\} \tag{2}$$

$$\boldsymbol{P}^{\text{now}} \triangleq \{(px_j^{\text{now}}, py_j^{\text{now}}), j \in [1, K]\} \tag{3}$$

式中:$\boldsymbol{P}^{\text{pre}}$是前一帧障碍物点云簇集合,$\boldsymbol{P}^{\text{now}}$是当前帧点云簇集合,$M$ 和 K 为其点云簇集合的总数。数据关联矩阵由距离差 $\mathbf{mat}_{\text{dist}}$和交并比 $\mathbf{mat}_{\text{IoU}}$两个矩阵加权组成,具体展开式如下:

$$\mathbf{mat} = a \times \mathbf{mat}_{\text{dist}} + b \times \mathbf{mat}_{\text{IoU}} \tag{4}$$

式中:

$$\mathbf{mat}_{\text{dist}} = \begin{bmatrix} d(p_1^{\text{pre}}, p_1^{\text{now}})^{-1} & \cdots & d(p_1^{\text{pre}}, p_K^{\text{now}})^{-1} \\ \cdots & \ddots & \cdots \\ d(p_M^{\text{pre}}, p_1^{\text{now}})^{-1} & \cdots & d(p_M^{\text{pre}}, p_K^{\text{now}})^{-1} \end{bmatrix} \tag{5}$$

$$\mathbf{mat}_{\text{IoU}} = \begin{bmatrix} \text{IoU}(p_1^{\text{pre}}, p_1^{\text{now}}) & \cdots & \text{IoU}(p_1^{\text{pre}}, p_K^{\text{now}}) \\ \cdots & \ddots & \cdots \\ \text{IoU}(p_M^{\text{pre}}, p_1^{\text{now}}) & \cdots & \text{IoU}(p_M^{\text{pre}}, p_K^{\text{now}}) \end{bmatrix} \tag{6}$$

$\mathbf{mat}_{\text{dist}}$矩阵中数据的具体展开如下,即当前帧和前一帧中每两个点云簇中心的距离为:

$$d(p_i^{\text{pre}}, p_j^{\text{now}}) = \sqrt{(px_i^{\text{pre}} - px_j^{\text{now}})^2 + (py_i^{\text{pre}} - py_j^{\text{now}})^2} \tag{7}$$

而 $\mathbf{mat}_{\text{IoU}}$矩阵中数据的具体含义就是当前某个点云簇和前一帧某个点云簇之间的交并比,交并比就是两个点云簇的外包矩形区域的交集比并集,最终得到一个(0,1)之间的数值。

此外,为了消除不同特征之间的量纲影响,提高匹配精度,需对 $\mathbf{mat}_{\text{dist}}$和 $\mathbf{mat}_{\text{IoU}}$这两个矩阵进行归一化处理:将原始数据线性地映射到[0,1]区间上,映射函数为:

$$x' = (x - \min)/(\max - \min) \tag{8}$$

式中:x 是原始数据,x'为归一化之后的数据,min 是矩阵中的最小值,max 为矩阵中的最大值。

1.3　自适应卡尔曼滤波跟踪

通过数据关联矩阵可以得到帧间障碍物点云簇的配对关系,然后,通过卡尔曼滤波来跟踪障碍物以得到其运动信息。但是,由于激光雷达点云获取的障碍物信息噪声分布不易建模,同时障碍物的过程噪声也不易观测,所以这里使用自适应卡尔曼滤波对障碍物进行跟踪。在障碍物跟踪过程中,算法中的过程噪声和观测噪声的协方差矩阵将不断更新,以达到最佳跟踪效果。障碍物的状态方程为:

$$\boldsymbol{X}_t = \boldsymbol{A}\boldsymbol{X}_{t-1} + \boldsymbol{W}_{t-1} \tag{9}$$

式中:$\boldsymbol{X}_t = [x, y, v^x, v^y, a^x, a^y]^{\text{T}}$ 是一个 6 维的状态向量,包括了 x 和 y 方向上的位置、速度和加速度。$\boldsymbol{X}_{t-1}$为上一时刻的状态,$\boldsymbol{W}_{t-1}$是过程噪声,其概率分布未知,$\boldsymbol{A}$ 为状态转移矩阵,具体展开表达式如下:

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & \Delta t & 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & 1 & 0 & \Delta t & 0 & \frac{\Delta t^2}{2} \\ 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

矩阵中 Δt 为观测间隔,即帧间的时间间隔,该时间可能会不断改变,所以状态转移矩阵也会不断更新。动态障碍物的观测方程为:

$$\boldsymbol{Z}_t = \boldsymbol{H}\boldsymbol{X}_t + \boldsymbol{V}_t \tag{11}$$

式中:$\boldsymbol{Z}_t = [x, y, v^x, v^y]^{\text{T}}$ 为观测量,包括坐标以及障碍物在 x 方向和 y 方向上的速度。观测量来自帧间点云聚类,其中点云聚类将得到障碍物位置,参考微分的思想即可获得 v^x 和 v^y,只要将采样间隔 Δt 设置得够小,就可以将两个采样间隔中的障碍物看为匀速运动,即 $v_t^x = (x_t - x_{t-1})/\Delta t$,$v_t^y$ 同理,$\boldsymbol{V}_t$ 是观测噪声。观测矩阵 $\boldsymbol{H}$ 为 4×4 的单位矩阵。

障碍物运动状态的后验估计 $\hat{X}_t$ 通过卡尔曼增益 K_t 计算,公式如下:

$$\hat{\boldsymbol{X}}_t = \hat{\boldsymbol{X}}_t^- + \boldsymbol{K}_t(\boldsymbol{Z}_t - \boldsymbol{H}\hat{\boldsymbol{X}}_t^-) \tag{12}$$

式中:$\hat{\boldsymbol{X}}_t$ 则为障碍物状态的最优估计。

观测噪声和过程噪声的协方差矩阵更新公式如下:

$$\boldsymbol{R}_t = \boldsymbol{R}_{t-1} \times \max\left(1, \frac{|\boldsymbol{P}_t - \boldsymbol{P}_{t-1}|}{R_{t-1}}\right) \tag{13}$$

$$\boldsymbol{Q}_t = \boldsymbol{Q}_{t-1} \times \max\left(1, \frac{|\boldsymbol{P}_t^- - \boldsymbol{P}_{t-1}^-|}{\boldsymbol{Q}_{t-1}}\right) \tag{14}$$

式中:$\boldsymbol{P}_t$ 和 $\boldsymbol{P}_t^-$ 为后验与先验估计的协方差矩阵。

最终,在这一过程中不断更新得到的 $\hat{\boldsymbol{X}}_t$ 就是预测到的障碍物运动信息,下面将介绍如何将运动信息与 TEB 算法联系起来以实现动态避障。

2　改进 TEB 算法

TEB 算法是一种经典的局部路径规划算法,它在 EB 算法的基础上,添加了各方面的约束,如时间约束、机器人运动学约束等。它把理想路径想象成一根橡皮筋,把所有的外部约束当成使橡皮筋发生形变的外力作用。原 TEB 算法在 ROS 中的应用是

通过局部代价地图来获取障碍物的位置信息,因此算法获取的障碍物信息不包括障碍物的运动信息,这会造成算法在有动态障碍物的环境下运行效果不佳,主要体现在可能来不及规避障碍物和避障路线不是最优。因此,为了优化算法在动态环境下的运行效果,本文在原 TEB 算法的基础上引入了风险区域来优化规划出来的路径。本文在原来约束函数的基础上引入一个新的约束函数:

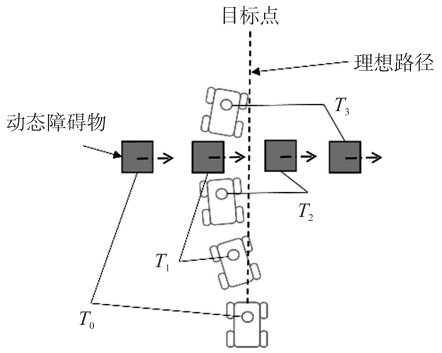
$$f_{\text{risk}}=e(-d_{\min,j},-r_{\min},\varepsilon,S,n) \tag{15}$$

式中:$d_{\min,j}$是位姿点与危险区域的最小距离,$r_{\min}$是机器人离风险区域的最小距离,e 函数具体展开式如下:

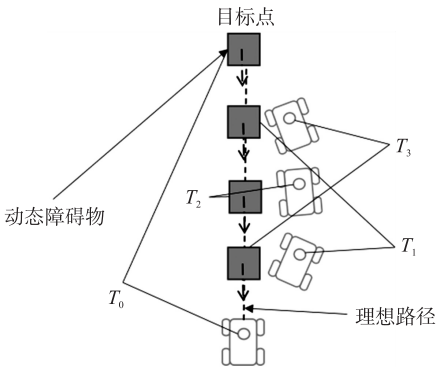
$$e(x,x_r,\varepsilon,S,n)\cong\begin{cases}\left[\dfrac{x-(x_r-\varepsilon)}{S}\right]^n & x>x_r-\varepsilon\\ 0 & \text{其他}\end{cases} \tag{16}$$

式中:x_r 为极限值,S、n 和 ε 影响近似的准确度,分别为缩放、多项式阶数以及近似值附近的一个小位移。

改进 TEB 算法在面对不同方向障碍物的避障示意如图 2 所示,图中小车代表同一移动机器人在不同时刻的位置,灰色矩形是不同时刻的动态障碍物,虚线箭头代表障碍物的速度方向,机器人的目标在图上方,黑色虚线为理想运动轨迹,$T_0\sim T_3$ 表示 0 时刻到 3 时刻。由图 2(a)可以看出,在检测到障碍物是从左往右运动时,改进 TEB 算法就规划出避障路径向左避障,同样从如图 2(b)的障碍物的避障示意图可以看出,在检测到障碍物的运动方向是朝向机器人从上往下运动的时候,算法就做出决策进行提前避障,而不是等进入到障碍物影响区域才避障。



(a) 改进TEB算法在面对侧面障碍物的避障流程



(b) 改进TEB算法在面对前方障碍物的避障流程

图 2 改进 TEB 算法的理想动态避障效果

图 3 为所提出风险区域的示意图,图中实线方形区域是当前时刻检测到的障碍物位置信息,它有 x 方向和 y 方向上的分速度,虚线区域是第 k 个 TEB 位姿点对应的障碍物位置,灰色长条矩形就是风险区域,它是由所有 TEB 位姿对应的未来时刻障碍物位置信息组成。第 j 个障碍物的位置信息 $P_j=[x_j,y_j]^{\mathrm{T}}$ 如下所示:

$$P_j=P_0+\sum_{i=0}^{j}\left[\mathrm{V}+\Delta t_i^{i+1}+\frac{\mathrm{A}\times(\Delta t_i^{i+1})^2}{2}\right],\quad j\in[1,k] \tag{17}$$

式中:$\mathrm{V}=[v_x,v_y]^{\mathrm{T}}$ $\mathrm{A}=[a_x,a_y]^{\mathrm{T}}$ 是当前时刻所检测到的障碍物速度和加速度,Δt_i^{i+1} 是第 $i+1$ 个与第 i 个 TEB 位姿的时间间隔。

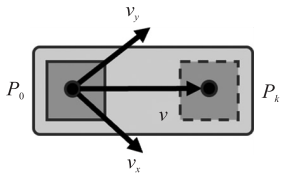
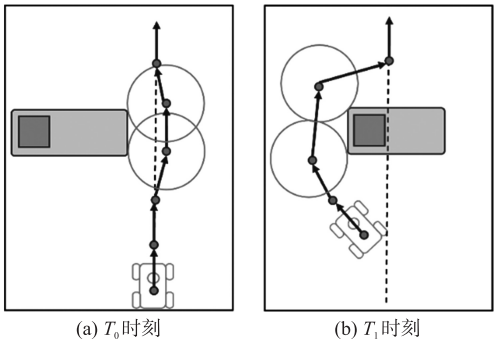
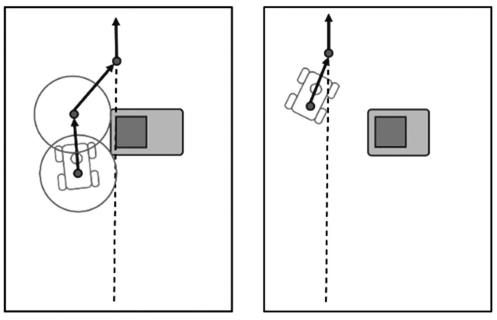


图 3 风险区域示意图

图 4 所示为图 2(a)中改进 TEB 算法实现动态避障的详细过程,图中将每个时刻的风险区域范围和 TEB 位姿点的具体状态进行了可视化。机器人的理想路径是黑色虚线,深灰色矩形是障碍物,浅灰色区域是障碍物拓展的风险区域,实线箭头代表 TEB 位姿序列,箭头指向为速度的方向,圆圈是位姿点与障



(a) T_0时刻　(b) T_1时刻



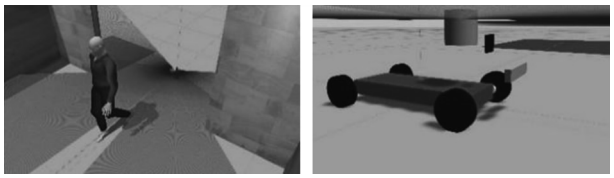
(c) T_2时刻　(d) T_3时刻

图 4 改进 TEB 算法的理想避障效果流程图

碍物的最近距离范围,如果位姿点与障碍物距离小于这个范围,TEB 位姿序列则会受到影响并发生变动。从图中可知,在 T_0 时刻,由于剩余的 TEB 位姿点最多,障碍物危险区域被拓展得最长,这时虽然障碍物没有进入理想路径影响范围,但是其所拓展的风险区域已经到达这个范围,所以 TEB 位姿序列会被向右拉伸。在 T_1 时刻,障碍物进入理想路径的范围内,但由于其风险区域朝右拓展,所以在 g2o 优化算法[26]的作用下,TEB 位姿序列被向左拉升,机器人在算法的控制下采取向左躲避障碍物的操作。在 T_2 与 T_3 时刻障碍物逐渐远离理想轨迹并不会影响到 TEB 位姿序列,机器人逐渐回归到理想路径。因此,改进 TEB 算法可以将经典 TEB 算法与障碍物运动信息结合起来,规划出一条融合障碍物风险区域的局部路径,提高移动机器人在面对动态障碍物的避障效率。

3　仿真和实验

为了验证改进 TEB 算法在动态环境下运行的鲁棒性和有效性,将搭建仿真和实验平台进行验证。仿真环境为运行内存为 16 GB 的 64 位 Ubuntu18.04 操作系统,平台为 ROS 和 Gazebo,仿真机器人采用阿克曼运动模型的竞速小车,具体的仿真环境如图 5 所示,图 5(a)所示为仿真中作为动态障碍物的仿真行人,图 5(b)所示为仿真所用的阿克曼机器人,图中深灰色区域为激光雷达可以扫描到的区域。首先,在没有动态障碍物的静态环境进行建图。然后,添加相对应的行人作为动态障碍物进行实验并录制 rosbag,然后,对 rosbag 的数据进行提取和分析。我们总共搭建了三个仿真环境和一个实验平台,在这四个环境下的实验将验证改进 TEB 算法在面对不同数量和速度的障碍物的避障效果,并且,对于所有的仿真场景,我们都把改进 TEB 算法与原 TEB 算法的运行效果进行了比较,在最后一个仿真中,本文还将改进 TEB 与基于 MPC 以及 State Lattice 的局部路径规划算法进行了比较。



(a) 仿真行人　　(b) 仿真小车

图 5　仿真环境

3.1　仿真实验 1

第一组仿真实验用于验证改进 TEB 算法在面对前方动态障碍物时的避障效果,并且将其实验结果与原 TEB 算法进行比较,过程如图 6 所示,小车的起点为(4,0),终点设置为(12,0),行人的起点为(11,0),终点为(6,0),当小车启动的时候,行人同时运动,在三次仿真中行人分别以 0.5 m/s、0.8 m/s、1.0 m/s 的速度朝小车的方向运动。

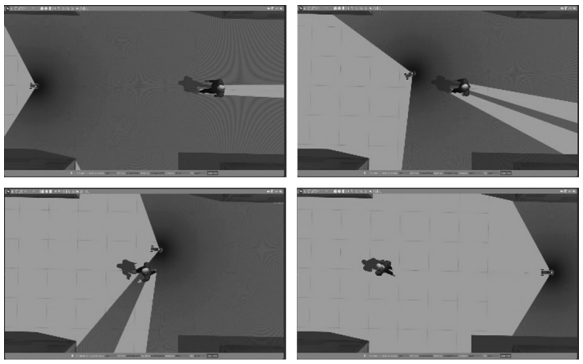
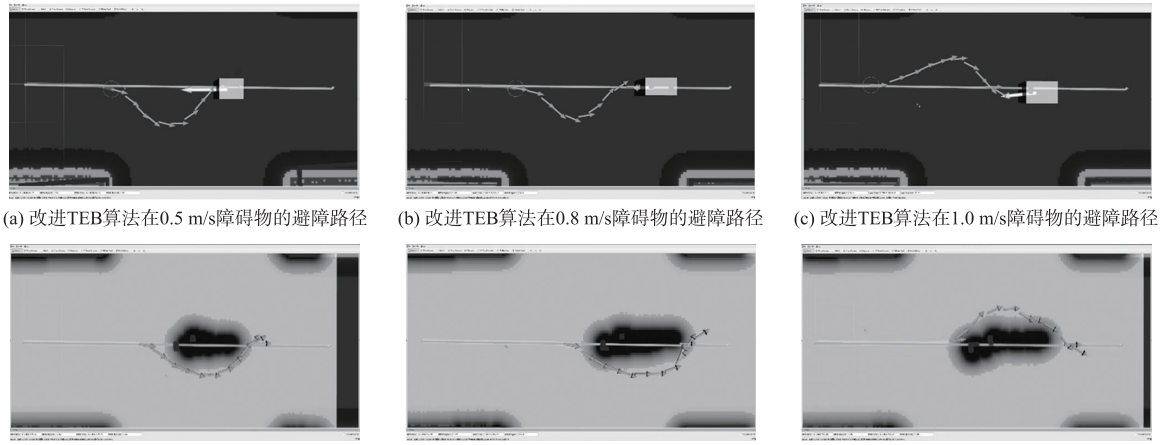


图 6　第一组仿真实验环境流程图

实验的详细过程如图 7 所示,图 7(a)与图 7(d)、图 7(b)与图 7(e)、图 7(c)与图 7(f)分别代表机器人在面对行人速度为 0.5 m/s、0.8 m/s 和 1.0 m/s 下开始进行避障时的 RViz 界面图,其中图 7(a)、图 7(b)、



(a) 改进TEB算法在0.5 m/s障碍物的避障路径　(b) 改进TEB算法在0.8 m/s障碍物的避障路径　(c) 改进TEB算法在1.0 m/s障碍物的避障路径

(d) 原TEB算法在0.5 m/s障碍物下的避障路径　(e) 原TEB算法在0.8 m/s障碍物下的避障路径　(f) 原TEB算法在1.0 m/s障碍物下的避障路径

图 7　第一组仿真的实验过程图

图 7(c)是改进 TEB 算法在开始避障时刻的 RViz 界面图,图 7(d)、图 7(e)、图 7(f)是原 TEB 算法开始避障时的 RViz 界面图,中间实线是全局路径,粗实线是机器人当前的运动轨迹,细线箭头表示 TEB 位姿序列,改进 TEB 算法中的白色半透明立方体框是检测到的障碍物框,白色箭头是障碍物的速度方向和大小,在原 TEB 算法效果图中,即图 7(d)、图 7(e)、图 7(f)中黑灰色部分。由图 7(a)、图 7(b)、图 7(c)可知,改进 TEB 算法能在离行人还有一定距离的时候就做出避障行为。

若把机器人进行避障行为时机器人与障碍物的距离称之为反应距离,改进算法和原算法的反应距离与行人速度的关系如表 1 所示。从表中可知,在改进 TEB 算法的作用下,反应距离与行人距离呈正相关,而在原 TEB 算法中,反应距离甚至还随着行人的速度的增大而缩小,在这样的情况下如果遇到较快速度的障碍物,移动机器人会来不及避障,给机器人的局部路径规划造成安全隐患,相反,如果能根据障碍物的速度控制避障行为的提前距离,就能在保证机器人在有动态障碍物环境下的导航安全。该实验结果很好地验证了所提出改进 TEB 算法在面对迎面而来障碍物时的避障安全性和高效性。

表 1 机器人反应距离和行人速度的关系表

行人速度/(m/s)	原 TEB/m	改进 TEB/m
0.5	1.582	2.920
0.8	1.239	3.529
1.0	1.055	4.250

3.2 仿真实验 2

第二组仿真实验用于验证改进 TEB 算法在面对侧方动态障碍物的避障效果与原 TEB 算法相比的优势之处,仿真环境和流程如图 8 所示。

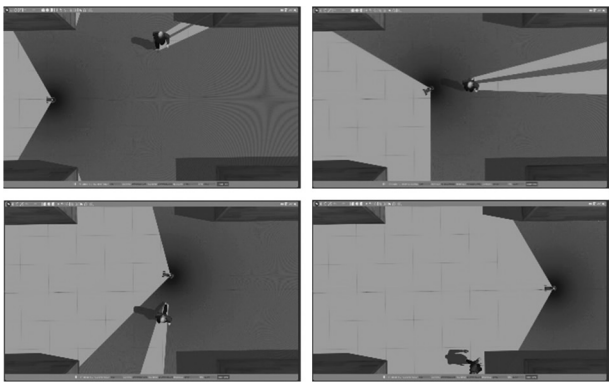
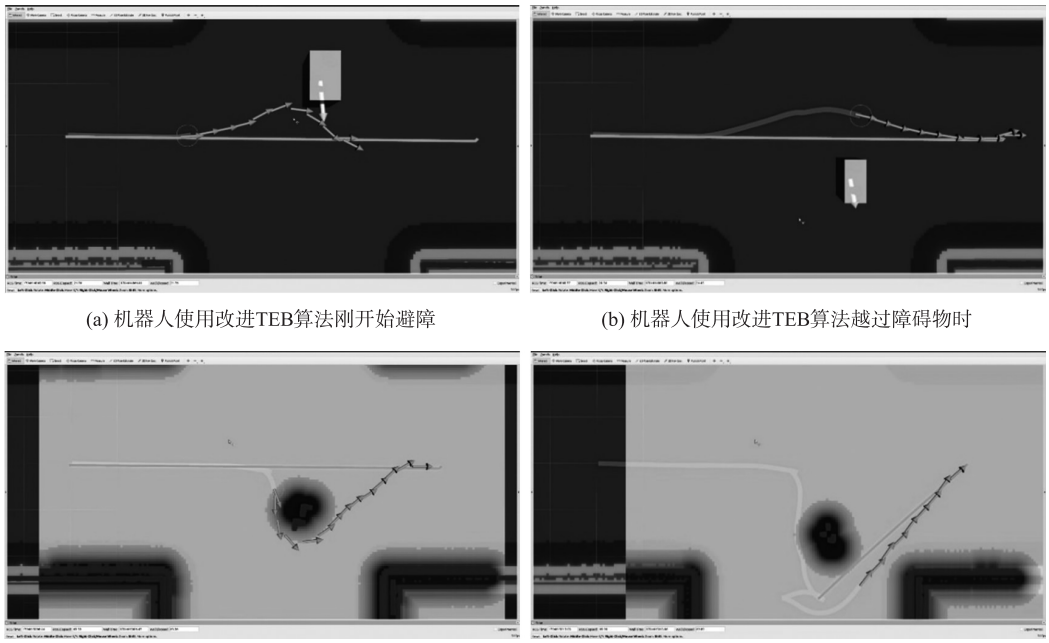


图 8 机器人躲避侧面行人效果图

图 8 展示了改进 TEB 算法在 Gazebo 仿真环境下的避障流程,这组实验中机器人的起点和终点保持不变,而行人的运动轨迹与第一组不同,它将以 0.8 m/s 的速度在(9,2.5)和(9,-2.5)之间做往复运动,当行人从(9,2.5)开始移动的时候机器人同时启动,观察机器人的避障行为。实验具体过程的 RViz 界面如图 9 所示,图 9(a)、图 9(b)、图 9(c)、图 9(d)分别代表机器人在改进 TEB 算法和原 TEB 算法的作用下对侧方障碍物时的避障轨迹。由图 9(c)与图 9(d)中的机器人轨迹可以看出,在原 TEB 算法的作用下,机器人会出现在避障时跟随动态障碍物一起运动的情况,这将导致机器人避障效率的下降。而改进



(a) 机器人使用改进TEB算法刚开始避障　(b) 机器人使用改进TEB算法越过障碍物时

(c) 机器人使用原TEB算法刚开始避障的时刻　(d) 机器人使用原TEB算法越过障碍物的时刻

图 9 第二组仿真实验的结果图

TEB 算法就没有这种问题,它可以检测出障碍物的运动方向和速度,在图 9(a)中的 TEB 位姿序列可以看到,虽然当前时刻行人位置不在小车的理想运动轨迹上,但是通过拓展的风险区域新约束,改进 TEB 算法就采取了提前左转避障的操作。此外由图 9(b)与图 9(d)中粗实线所示的运动轨迹也可以明显看出,改进 TEB 算法的避障路径相对于原算法也减短很多。

3.3　仿真实验 3

第三组仿真实验用于验证改进 TEB 算法在多障碍物复杂环境下的局部路径规划效果与其他局部路径规划算法相比是否具有优势之处,其中包括原 TEB 算法、基于 MPC 的规划器和基于 State Lattice 的局部路径规划器,仿真环境如图 10 所示,其中行人的数量增加为四个,且在不同的轨迹上进行往复运动,机器人的最大运行速度为 1.2 m/s,机器人的目标是从坐标(0,0)安全地到达目标点(11,0),对于这四种算法,我们都进行了 50 次实验并处理得到了实验结果数据,其中包括四种算法的成功率、总距离和总时间,通过表 2 呈现,由表 2 可知,改进 TEB 算法在保证较高成功率的前提下,还提高了机器人的平均速度,此仿真实验的结果也验证了改进 TEB 算法在多动态障碍物的环境下相对其他先进算法具有更强的高效性和鲁棒性。

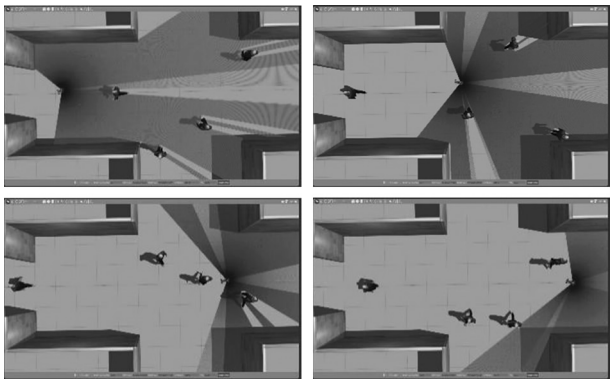


图 10　机器人在复杂环境下的导航避障效果图

表 2　两种方法的成功率、距离和时间

算法	成功率	距离/m	时间/s
原 TEB	70%	15.61	18.81
MPC	80%	14.54	16.64
State lattice	90%	15.33	17.58
改进 TEB	95%	12.03	14.82

3.4　移动机器人平台实验

本文使用 Jetson Nano、镭神 M10P 二维激光雷达、STM32 等硬件设备搭建了机器人移动平台,对所提出的基于改进 TEB 算法的动态避障算法的避障效果进行验证,图 11 展示了真实环境实验的机器人。移动机器人平台实验将验证改进 TEB 算法对于迎面而来的行人是否会采取提前避障的操作。真实实验环境和实验结果如图 12 所示,给定机器人的起点为(-1,0),终点为(5,0),当机器人启动时一个人作为动态障碍物缓慢地面向机器人运动。

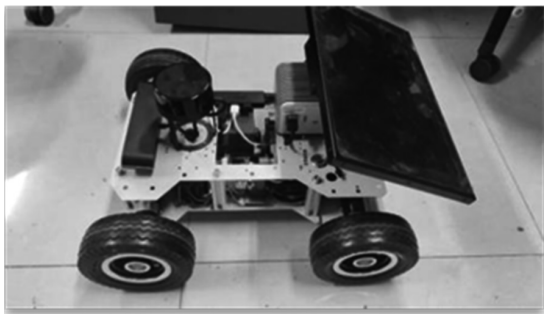
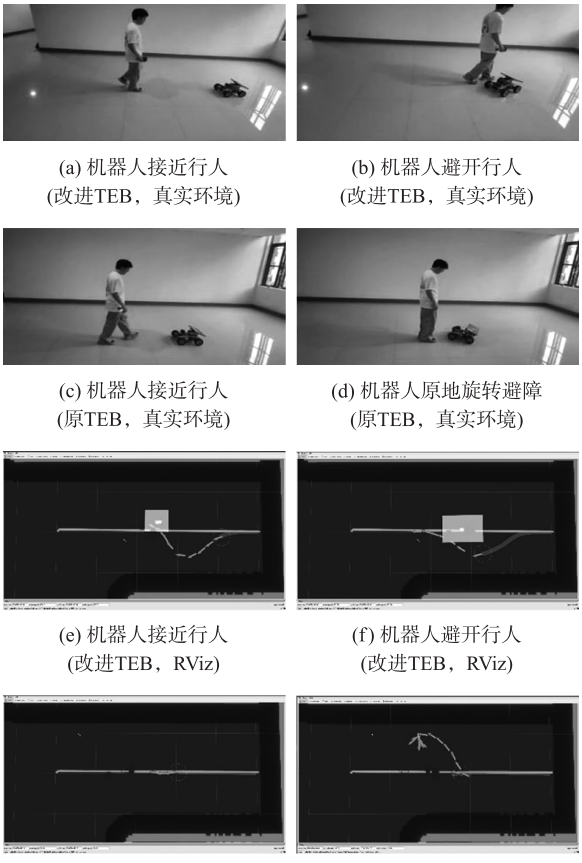


图 11　移动机器人平台



(a) 机器人接近行人 (改进TEB，真实环境)　(b) 机器人避开行人 (改进TEB，真实环境)

(c) 机器人接近行人 (原TEB，真实环境)　(d) 机器人原地旋转避障 (原TEB，真实环境)

(e) 机器人接近行人 (改进TEB，RViz)　(f) 机器人避开行人 (改进TEB，RViz)

(g) 机器人接近行人 (原TEB，RViz)　(h) 机器人原地旋转避障 (原TEB，RViz)

图 12　移动机器人平台实验和对应的 RViz 界面图

图 12 展示了改进 TEB 算法和原 TEB 算法在移动机器人平台的运行图片和对应的 RViz 截图,图 12(a)、图 12(b)、图 12(e)、图 12(f)和图 12(c)、图 12(d)、图 12(g)、图 12(h)分别是机器人在改进 TEB 算法和原 TEB 算法下面对迎面而来行人的避障过程图,其中图 12(a)、图 12(b)、图 12(c)、图 12(d)

是真实环境下的照片,图 12(e)、图 12(f)、图 12(g)、图 12(h)是对应时刻的 RViz 界面,在 RViz 界面中,中间白色实线是理想全局路径,灰色粗实线为机器人运动轨迹,带箭头线则是 TEB 位姿序列,图 12(e)、图 12(f)中的半透明矩形则是算法检测到的障碍物外包矩形框。

通过图 12(c)、图 12(d)、图 12(g)、图 12(h)可以看出,在原 TEB 算法中,只有当行人与机器人的距离足够近,机器人才能进行避障操作,这会导致机器人原地转向,如果这时行人没有注意到移动机器人而保持直行,会与机器人发生碰撞,存在安全隐患。而由图 12(a)、图 12(b)、图 12(e)、图 12(f)明显看到,在改进 TEB 算法中,机器人在面对迎面而来的行人的时候会做出提前避障的操作,虽然图 7(a)中行人与机器人的距离比图 7(c)中行人与机器人的距离更大,但是在改进 TEB 算法的作用下,机器人就已经开始避障,即使这时行人保持自己的运动轨迹,机器人规划的避障路线也能与行人错开,具较高的安全性。移动机器人平台实验验证了所提出算法在面对动态障碍物能够进行提前避障,并且安全地规避障碍物。

仿真和移动机器人的实验结果表明,融合 TEB 算法和风险区域可以使移动机器人在遇到动态障碍物时提前避障,且提前避障的距离与障碍物的运动速度有关,避障成功率更高,此外,改进 TEB 还结合了基于 CDBSCAN 聚类和自适应卡尔曼滤波的多目标跟踪算法,提高了检测到的障碍物信息的准确度。实验结果证明在存在动态障碍物的环境中,改进 TEB 算法与原 TEB 算法相比,缩短了避障时间,提高了避障的效率和安全性。

4 结论

本文针对移动机器人在室内动态环境下的避障问题,提出了一种融合动态障碍物风险区域的改进 TEB 算法。改进的 TEB 算法在原来约束函数的基础上引入了风险区域新约束,设计了一种考虑动态障碍物风险区域的改进 TEB 算法,通过处理二维激光雷达点云信息来获取障碍物运动信息,并且拓展障碍物风险区域,优化了机器人在动态环境下的避障路径。为了验证改进 TEB 算法在动态环境下避障的有效性和鲁棒性,本文在 Gazebo 中针对室内动态避障问题进行了仿真实验,且在三种环境下的仿真结果都表明了改进 TEB 算法在动态环境下相对于原 TEB 算法的有效性,它可以根据障碍物的运动速度进行提前避障,减少避障时间、缩短避障路径、提高避障效率。此外,本文还完成了 ROS 小车在真实环境下的动态避障实验,实验结果表明本文提出的改进 TEB 算法在移动机器人平台同样具有预测障碍物速度和动态避障的能力,进一步验证了改进 TEB 算法的可行性。未来,我们将针对障碍物运动状态的不确定进行进一步避障算法研究。

参考文献:

[1] 王梓强,胡晓光,李晓筱,等. 移动机器人全局路径规划算法综述[J]. 计算机科学,2021,48(10):19-29.

[2] 吴涛,谢志军,陈科伟. 改进 D * lite 和时间弹性带法的移动机器人路径规划[J]. 传感技术学报,2022,35(4):486-494.

[3] Zhang L, Zhang Y, Li Y. Mobile Robot Path Planning Based on Improved Localized Particle Swarm Optimization [J]. IEEE Sensors Journal,2021,21(5):6962-6972.

[4] Sun T, Pan W, Wang Y, et al. Region of Interest Constrained Negative Obstacle Detection and Tracking with a Stereo Camera[J]. IEEE Sensors Journal,2022,22(4):3616-3625.

[5] 邹珺淏,任酉贵,冷芳玲,等. LW-YOLOv7SAR:轻量 SAR 图像目标检测方法[J]. 小型微型计算机系统,2025,46(1):143-150.

[6] Ren W, Wang X, Tian J, et al. Tracking-by-Counting: Using Network Flows on Crowd Density Maps for Tracking Multiple Targets[J]. IEEE Transactions on Image Processing,2021,30:1439-1452.

[7] 陈建,于殿泓,惠鑫鑫,等. 基于二维激光雷达多车道车辆检测与分车算法的研究[J]. 传感技术学报,2022,35(11):1513-1520.

[8] Duong H T, Suh Y S. Human Gait Tracking for Normal People and Walker Users Using a 2D LiDAR[J]. IEEE Sensors Journal,2020,20(11):6191-6199.

[9] 韩建栋,范学媛. Myvoxel R-CNN:基于体素的三维点云目标检测模型[J]. 小型微型计算机系统,2024,45(8):1908-1913.

[10] 王洪斌,尹鹏衡,郑维,等. 基于改进的 A * 算法与动态窗口法的移动机器人路径规划[J]. 机器人,2020,42(3):346-353.

[11] 邵思扬,吴旭航,林远山,等. 基于改进 DWA 和人工势场法的编队控制研究[J]. 电子器件,2024,47(3):711-717.

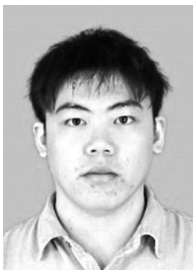
[12] Fox D, Burgard W, Thrun S. The Dynamic Window Approach to Collision Avoidance[J]. IEEE Robotics & Automation Magazine,1997,4(1):23-23.

[13] Roesmann C, Feiten W, Woesch T, et al. Trajectory Modification Considering Dynamic Constraints of Autonomous Robots [C]//7th German Conference on Robotics, Munich, Germany,2012:74-79.

[14] Rösmann C, Feiten W, Wösch T, et al. Efficient Trajectory Optimization Using a Sparse Model[C]//2013 European Conference on Mobile Robots(ECMR), Barcelona, Spain,

2013:138-143.
[15] Rösmann C,Hoffmann F,Bertram T. Planning of Multiple Robot Trajectories in Distinctive Topologies[C]//2015 European Conference on Mobile Robots(ECMR),Lodon,UK,IEEE 2015:1-6.
[16] Rösmann C,Hoffmann F,Bertram T. Kinodynamic Trajectory Optimization and Control for Car-like Robots[C]//2017 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS),Vancouver,BC,Canada,2017:5681-5686.
[17] Rösmann C,Oeljeklaus M,Hoffmann F,et al. Online Trajectory Prediction and Planning for Social Robot Navigation[C]//2017 IEEE International Conference on Advanced Intelligent Mechatronics(AIM),Munich,Germany,2017:1255-1260.
[18] Guo B,Guo N,Cen Z. Obstacle Avoidance with Dynamic Avoidance Risk Region for Mobile Robots in Dynamic Environments[J]. IEEE Robotics and Automation Letters,2022,7(3):5850-5857.
[19] Wang C,Chen X,Li C,et al. Chase and Track:Toward Safe and Smooth Trajectory Planning for Robotic Navigation in Dynamic Environments[J]. IEEE Transactions on Industrial Electronics,2023,70(1):602-613.
[20] Dong H Y. Weng C,Guo C,et al. Real-Time Avoidance Strategy of Dynamic Obstacles via Half Model-Free Detection and Tracking With 2D LiDAR for Mobile Robots[J]. IEEE/ASME Transactions on Mechatronics,2021,26(4):2215-2225.
[21] 潘富强,曾成,马国红,等. 一种融合改进 A * 算法与改进动态窗口法的 AGV 路径规划[J]. 传感技术学报,2023,36(1):68-77.
[22] 程捷,郭振武,周坤,等. 基于改进相关性扫描匹配的室内大场景下移动机器人的全局定位方法[J]. 机器人,2023,45(5):581-590.
[23] Liu C J,Liu Y. Robot Planning and Control Method Based on Improved Time Elastic Band Algorithm[C]//2023 4th International Conference on Computer Engineering and Application(ICCEA),Hangzhou,China,2023:911-915.
[24] Sun C,Zhang X,Zhou Q,et al. A Model Predictive Controller with Switched Tracking Error for Autonomous Vehicle Path Tracking[J]. IEEE Access,2019,7:53103-53114.
[25] Welch G,Bishop G. An Introduction to the Kalman Filter[R]. Chapel Hill,NC,USA:University of North Carolina at Chapel Hill,1995.
[26] Kümmerle R,Grisetti G,Strasdat H,et al. G2o:A General Framework for Graph Optimization[C]//2011 IEEE International Conference on Robotics and Automation,Shanghai,China,IEEE,2011:3607-3613.

欧县华(1990—),男,博士,讲师,主要研究方向:SLAM、数据驱动控制,xhou@zjut.edu.cn;



游仲南(1999—),男,硕士研究生,主要研究方向:移动机器人路径规划和避障,youayou121@163.com;

何熊熊(1965—),男,博士,教授,主要研究方向:学习控制、信号处理,数据驱动控制,hxx@zjut.edu.cn。