移动机器人导航规划的双向平滑A-star算法

岳高峰^{1,2}, 张萌^{1,2*}, 沈超^{1,2}, 管晓宏^{1,2}

1. 西安交通大学网络空间安全学院, 西安 710049;

2. 西安交通大学智能网络与网络安全教育部重点实验室, 西安 710049

* E-mail: mengzhang2009@xjtu.edu.cn

收稿日期: 2020-05-06; 接受日期: 2020-06-29; 网络版发表日期: 2021-02-23

国家重点研发计划(编号: 2019YFB1312002)、国家自然科学基金(批准号: 61903292)、陕西省自然科学基金(编号: 2019JQ-084)和中央高校基本科研业务费(编号: xzy012019040)资助项目

摘要 针对主流移动机器人导航中蒙特卡罗定位(MCL)算法误差大、传统A-star规划路径实时性差且所得路线为折线的问题, 提出了一种基于双向平滑A-star算法的路径规划导航策略. 首先提出多传感器融合蒙特卡罗定位的位姿估计定位算法; 其次通过优化A-star算法的代价函数, 使其整体搜索方向变为双向从而提高算法实时性; 最后采取Bézier曲线对所规划路径进行平滑优化, 解决A-star算法规划路径折线多、转弯角度过小而无法满足不同物理约束的问题. 仿真实验结果展示, 相比传统A-star轨迹规划算法, 本文算法在路径长度和时间方面分别减少12.1%, 37.2%, 平均安全距离和路径平滑度分别提升35.2%, 69.9%; 同时在保证的定位精度下所测试的实际实验证明了本文导航规划算法的正确性和可行性.

关键词 自主定位导航, MS-MCL定位, A-star算法, 双向搜索, Bézier曲线

1 引言

随着人们生活水平的提高, 移动机器人产业得到大力发展, 自主定位与导航作为其智能化水平的重要体现^[1], 得到国内外学者的广泛研究, 同时定位导航技术的快速发展也影响和改变着人们的生活方式^[2].

Leonard于1987年最早归纳出移动机器人导航的三个基本问题: 起点-路径-终点^[3], 大量学者相继由此展开研究. 在未知或已知的环境中, 移动机器人导航的关键在于其自身定位的精度, 室外定位可以广泛采用GPS和INS融合的方法, 定位精度和性能都可以满足基本应用, 但是在室内却显得无能为力. 室内定位方法有

航迹推算定位、地图匹配定位以及基于信标的定位. 基于航迹的推算是利用编码器、陀螺仪、加速度计等惯性量测设备, 根据状态控制方程进行反积分的过程, 在短时间、误差容忍度大的情况下可以使用^[4]. 常用方法有卡尔曼滤波法和互补滤波^[5], 鉴于该方法的自主性和抗干扰能力, 在移动机器人中得到广泛使用. 基于地图匹配的定位方法多采用传感器进行建图的方式完成^[6], 通过机载超声波雷达等设备进行全方面精细的环境建图, 栅格地图创建简单且能够较好地保留全局信息, 1986年IEEE会议提出的同步定位与建图(simultaneous localization and mapping, SLAM)方法成为至今使用最多的自主导航方式^[7]; Su等人^[8]于

引用格式: 岳高峰, 张萌, 沈超, 等. 移动机器人导航规划的双向平滑A-star算法. 中国科学: 技术科学, 2021, 51: 459–468

Yue G F, Zhang M, Shen C, et al. Bi-directional smooth A-star algorithm for navigation planning of mobile robots (in Chinese). Sci Sin Tech, 2021, 51: 459–468, doi: 10.1360/SST-2020-0186

2012年提出基于曲率特征的地图匹配定位方法,通过参考轨迹的尺度不变积分特征消除了传感器航迹推算的误差,从数字环境中选择匹配的候选路段,根据曲率和航向角的变化确定相关匹配点;基于信标的定位系统依赖于一系列环境中已知特征的信标,可以提供快速、稳定、精确的绝对位置信息,但安装和维护信标花费很高,典型系统有Ward等人于1997年提出的Bats系统^[9]. 移动机器人的自主导航方法较多,常用的全局路径规划有Dijkstra算法、A-star算法、蚁群算法等^[10]. 单纯的Dijkstra算法^[11]并不考虑目标的方向问题,而是通过最短路径实现导航,该方法在地图环境较小时可以使用,随着地图的增大计算量也会急剧增长. 基于启发函数提出的A-star算法解决了这个问题,也是目前最常用的机器人轨迹规划算法;2017年,Chauri等人^[12]设计了松弛A-star算法,其增大了约束线性复杂度,与A-star算法相比,执行时间大大减少,松弛A-star之所以成为较好的路径规划器,主要是由于它在所有度量之间提供了良好的权衡,并证明其算法可比传统A-star执行时间得到有效缩减.

本文研究分析了目前已有的自主定位导航策略, 针对导航中机器人存在机械误差而导致定位不准确, 提出了一种MS-MCL(Multi-sensor Monte Carlo localization)定位算法; 对于A-star轨迹规划算法路径为折线而无法较好应用的问题, 设计了基于A-star的双向平滑路径策略, 改进代价函数并加入Bézier曲线保证导航路径的实时性和平滑性。

2 建图与定位

2.1 环境建图与MCL定位

针对室内环境及导航精度要求高等要求, 选择采用栅格地图法^[13]来对其进行建图描述。移动机器人主要依靠实时里程计信息和激光雷达数据, 实现对低特征环境的状态感知, 并输出基于网格的2D地图。

系统采样编码器和激光雷达原始数据以四元素法输入,在笛卡尔坐标系下可定量建立机器人运动学模型,关联机器人坐标系和地图坐标系后即可实现对地图及障碍的描述,多次地图更新后就完成了针对环境的2D真实重构,实现流程如图1所示。

考虑到轮式移动机器人的运动状态具有高度非线性, 实时位姿描述又是典型非高斯动态系统^[14]的状态

图 1 环境建图流程图

Figure 1 Flow chart of environmental mapping.

递推估计问题,选择引入蒙特卡罗定位(MCL)算法。蒙特卡罗定位算法是一种基于粒子滤波的定位算法^[15],通过在空间内均匀分布大量粒子并结合状态转移方程来得到模拟运动状态,用真实状态粒子作为代价标准从而筛选权重较大的预测值,并进行重采样带入状态转移方程得到新预测粒子,迭代操作即可实现对移动机器人的实时定位。算法实现流程如图2所示。

定义激光雷达采样数据集为 $Z^n(n=1, \dots, n)$, 编码器采样数据集为 $u^n(n=1, \dots, n)$, 机器人位姿用 $x_n=[x, y, \theta]^T$ 表示。基于编码器数据和当前位姿参数预估下次可能状态的概率密度函数如式(1)所示:

$$p(x_{n+1}|Z^n) = \int p(x_{n+1}|x_n, u_n)p(x_n|Z^n)dx_n. \quad (1)$$

结合激光雷达数据和地图信息后进行相关验证并得到下次位姿估计值, 其概率密度函数如式(2)所示:

$$p(x_{n+1}|Z^{n+1}) = \frac{p(z_{n+1}|x_{n+1})p(x_{n+1}|Z^n)}{p(z_{n+1}|Z^n)}. \quad (2)$$

2.2 MS-MCL优化定位

传统MCL定位算法主要依据激光雷达点云数据和编码器数据来做预测实现定位,但其存在数据源容易产生误差而导致定位失败的问题。本文设计了多传

图 2 (网络版彩图) MCL 算法流程图

Figure 2 (Color online) Flow chart of MCL algorithm.

传感器优化的蒙特卡洛算法(multi-sensor Monte Carlo algorithm, MS-MCL), 将IMU(inertial measurement unit)数据经惯性导航解算得到的机器人加速度、角加速度等参数通过卡尔曼滤波融合于机器人运动模型, 改进编码器因机械误差而产生定位精度不足的问题。

在实际工程中, 用编码器原始数据转换处理可以得到机器人绝对运动参数即位置和姿态, 经坐标转换建立地图-机器人状态数据的关联, 完成地图坐标系下移动机器人位姿数据的实时更新, 同时依据激光雷达测距原理^[16,17]确定障碍物位置。定义机器人坐标系为 xoy 、地图坐标系 XOY , 机器人平面几何中心 O 距障碍物 A 的两次采样结果分别为 L_1, L_2 , 同时令机器人坐标系 X 轴、 OA 连线与地标系 x 轴夹角分别为 θ, φ , 第二次采样测得夹角为 φ' , 本文抽象移动机器人平面运动模型如图3所示。

设定雷达采样时刻 t_0, t_1, t_2 , 机器人沿 x 轴移动距离 L_3 , 可得机器人和障碍物的距离 L_1, L_2 , 相对障碍物位姿角度变化:

$$\varphi' - \varphi = \arccos \frac{(t_1 - t_0)^2 + (t_2 - t_1)^2 - \left[\frac{v_x}{c} \frac{v_y}{c} \right] \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} (t_2 - t_0)^2}{2(t_1 - t_0)(t_2 - t_1)} \quad (3)$$

同时得到机器人坐标系同地图坐标系之间转换

图3 (网络版彩图)机器人运动模型图

Figure 3 (Color online) Flow chart of MCL algorithm.

关系:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (4)$$

通过卡尔曼滤波将机器人运动数学模型和编码器参数做相关融合, 惯性导航解算出校正矩阵, 其数学表达式如式(5)所示, 如此便在上一次定位结果的基础上对本次位置参数进行了补偿。

$$\begin{cases} \hat{x}_{k-} = \mathbf{A}\hat{x}_{k-1} + \mathbf{B}U_{k-1}, \\ \hat{x} = \hat{x}_{k-} + \frac{(\mathbf{A}P_{k-1}\mathbf{A}^T + \mathbf{Q})\mathbf{H}^T}{\mathbf{H}(\mathbf{A}P_{k-1}\mathbf{A}^T + \mathbf{Q})\mathbf{H}^T + \mathbf{R}}(Z_k - \mathbf{H}\hat{x}_{k-}), \end{cases} \quad (5)$$

式中, $\hat{x}_k$: k 时刻后验状态估计值, 即最优估计, P_{k-1} : $k-1$ 时刻后验估计协方差, $\mathbf{A}$: 系统状态转移矩阵, $\mathbf{Q}$: 过程激励噪声协方差, $\mathbf{R}$: 测量噪声协方差, $\mathbf{B}$: 将输入转换为状态的矩阵, $Z_k - \mathbf{H}\hat{x}_{k-}$: 实际观测和预测观测的残差, $\hat{x}_{k-}$: k 时刻先验状态估计值, U_{k-1} : $k-1$ 时刻系统输入, $\mathbf{H}$: 状态到测量的转换矩阵。

3 导航算法

3.1 A-star算法

由于A-star搜索算法易于计算和实现, 因此得到了广泛的应用。A-star算法作为一种启发式图搜索算法, 不需要访问所有节点, 对环境反应迅速, 搜索示意图如图4所示。

算法从理论上保证了搜索节点扩展后全局最优解的收敛性和估值函数的变化^[17]。在A-star方法中, 搜索区域被转换成一个二维数组, 网络的中心被定义为节点(n)。数组的每个元素都是网格地图的一个正方形。二维方格定义为可通过节点(open表)和不可通过节点(close表), 代价函数的一般形式见式(6):

$$f(n) = g(n) + h(n), \quad (6)$$

式中, $f(n)$ 是从初始点到当前节点 n 到目标点的代价函数, $g(n)$ 是状态空间中初始节点到节点 n 的实际代价, $h(n)$ 是从节点 n 到目的节点的最优路径的估计代价。

$f(n)$ 评估open表中每个节点的重要性并确定节点的顺序。open表用于记录已计算但未展开的节点, close表用于存储已展开的节点。在搜索过程的所有步骤中, 首先需要找到open表中的节点来计算代价函数, 然后

图4 (网络版彩图)A-star 算法搜索示意图

Figure 4 (Color online) Search diagram of A-star algorithm.

添加到close表,直到open表为空,算法实际应用过程如图5所示。

传统A-star算法的优点是规划阶段引入了启发信息作为向目标点移动的决策辅助,从而无需再遍历整张地图,有效降低了计算复杂度,节省了规划时间^[18,19]。传统A-star算法存在四方向和八方向等搜索方式,该算法的路径搜索结果如图6所示。

从理论上分析, A-star算法随着搜索空间的扩大和节点数目的增加,搜索时间也会增加。A-star算法的应用会受到室内环境图大小的限制,同时也存在因客观因素导致的不合理性,如不考虑移动小车自身体积大小及物理学状态等多约束条件。

3.2 改进A-star算法

A-star算法可以较优地规划路径,传统A-star算法规划的路线是基于网格的折线并且影响实时性^[20,21],这给移动机器人实时跟踪路径带来了严重困扰。

文中我们提出双向A-star改进算法,使用双向迭代方式进行前进搜索,将起始点和目标点重映像:起始点

即目标点、目标点即起始点,且在路径搜寻的过程中实时迭代起始点和目标点,路径每个节点将是起始点和目标点的二重身份。我们对评价函数进行改写,如式(7)所示:

$$f(\cdot) = (1 - \lambda)f_{1(n)} + \lambda f_{2(n)} \quad (7)$$

$$f_{1(n)} = g_{i(n)} + h_{i(n)} \quad (8)$$

$$\lambda = \frac{D[(x_{2(s)}, y_{2(s)}), (x_{(g)}, y_{(g)})]}{D[(x_{(s)}, y_{(s)}), (x_{(g)}, y_{(g)})]}, \quad (9)$$

式中 λ 为代价函数相应权值, D 为欧几里得距离。改进的代价函数可以动态调整起始点和目标点双向搜索的速度,使其进行分布式并行处理,算法迭代方法如式(10)和(11)所示,改进方法示意图如图7所示。

$$(x_s, y_s) \xrightarrow{f(\cdot)} (x_g, y_g), \quad (10)$$

$$\begin{cases} (x_1, y_1) \xleftarrow{f(\cdot)} (x_s, y_s), \\ \dots \\ (x_n, y_n) \xleftarrow{f(\cdot)} (x_g, y_g). \end{cases} \quad (11)$$

实际应用中移动机器人并不完全适合A-star所规划的折线路径,机器人行进过程中需要满足一些物理特性约束。本文采用贝塞尔曲线(Bézier curve)进行约束优化求解,贝塞尔曲线是由法国工程师Pierre Bézier于1962年提出的^[22],其可以满足车体运动学约束及障碍物的定点约束,是一种较为理想的路径优化方法。

n 阶贝塞尔曲线简化了轨迹规划的求解,其控制方程如式(12)所示。

$$P(t) = \sum_{i=0}^n \binom{n}{i} P_i (1-t)^{n-i} t^i, \quad (12)$$

式中 P 代表定点, n 为节点个数, $t \in [0, 1]$, n 阶贝塞尔曲线其实是 $n-1$ 阶的等效插值^[23,24]。本文以三阶贝塞尔

图5 (网络版彩图)算法应用流程

Figure 5 (Color online) Algorithm application process.

图 6 (网络版彩图)四方向(a)和八方向(b)搜索结果

Figure 6 (Color online) Four directions (a) and eight directions (b) search results.

图 7 改进算法示意图

Figure 7 Schematic diagram of improved algorithm.

曲线为例, 曲线在起始点的曲率可由式(13)计算:

$$\kappa(0) = \frac{3}{4} \times \frac{|(P_1 - P_0) \times (P_2 - P_0)|}{|P_1 - P_0|^3}. \quad (13)$$

在A-star规划的路径中计算其相邻3节点的相交角度并进行相应曲线优化, 使其绕过障碍物点并缩短距离, 贝塞尔曲线在某节点的曲率关系可表达为

$$\kappa(\tau) = \frac{x'(\tau)y''(\tau) - y'(\tau)x''(\tau)}{(x'^2(\tau) + y'^2(\tau))^{\frac{3}{2}}}. \quad (14)$$

在本文中, 节点的控制选择为障碍物的存在点即 $[P_1, P_2]$, 当 $\tau=0.25$ 时轨迹优化原理图如图8所示.

本文对标准A-star算法引入双向搜索方式并改进其代价函数, 通过Bézier曲线优化其轨迹, 改进算法的流程如图9所示.

4 实验与分析

4.1 实验设计

本文设计了两组测试实验, 一组为计算机仿真模

图 8 (网络版彩图)贝塞尔曲线轨迹优化示意图

Figure 8 (Color online) Bézier curve path optimization diagram.

图9 (网络版彩图)本文算法流程图

Figure 9 (Color online) Algorithm flow chart of this paper.

拟实验, 可证明本文算法的正确性, 系统软件环境为 Windows 7专业版和MATLAB 2016b; 第二组为实际测试实验, 主要为了验证本文算法的可行性, 所测试机器人是基于ROS的高性价比移动机器人——Turtlebot, 该机器人广泛用于算法研究, 实物图如图10所示。

4.2 结果与分析

本文第一组为仿真实验, 主要对传统A-star算法和改进的A-star算法进行性能对比, 实验中黄色点为始末点, 深蓝色为可行空间, 浅蓝色为障碍物存在点, 红色为传统A-star所得结果, 黄色路径为本文算法所得, 实验结果如图11所示。

对上述结果进行分析, 从图11(a)中可以看到, 尽管传统A-star算法选择的是八方向搜索方式, 其路径有所缩短, 但结果却不完全满足实际机器人运动要求且不是最优的。图11(b)中本文算法的结果(黄色)明显在路

图10 Turtlebot 机器人

Figure 10 Turtlebot robot.

径长度方面少于传统算法(红色), 本文改进算法路径为平滑路径且转弯次数较少, 更加满足实际移动机器人的物理特性。

图 11 (网络版彩图)算法仿真结果. (a) 传统 A-star 算法; (b) 改进的 A-star 算法
Figure 11 (Color online) Algorithm simulation results. (a) Traditional A-star algorithm; (b) improved A-star algorithm.

本文对算法耗时进行定量测试, 随着网格数量增加, 算法实时性变化曲线如图12所示.

绿色曲线为本文算法耗时描述, 可以看出随着网格数目的增加, 两种算法耗时曲线开始分离, 证明本文算法随着地图环境的增大, 实时性占明显优势. 为了验证本文算法的可扩展性, 图13为3种不同环境栅格数量下障碍物随机分布的情况, 图中黑色为障碍点和边界点, 红色为close列表节点, 绿色为open列表节点, 红色折线路径为传统A-star算法所得, 蓝色平滑路径为本文改进的A-star算法所得, 明显看到本文改进算法比传统A-star算法在路径方面更短, 更加满足移动机器人的运动学约束.

针对仿真实验, 本文除了采用传统的距离、运行时间作为评价标准, 还提出平均安全距离(average safe distance, ASD)和路径平滑度(path smoothness, PS)等算

图 12 (网络版彩图)算法运行时间曲线图
Figure 12 (Color online) Algorithm running time curve.

法评价指标. 平均安全距离是指每个节点沿搜索方向连接最近障碍物的欧式距离, 平均安全距离计算方法

图 13 (网络版彩图)改进算法仿真结果

Figure 13 (Color online) Simulation results of improved algorithm.

如式(15)所示; 路径平滑度是指在路径可行条件下, 导航路径经过的节点数与始终点直线所经过节点的大小比例关系, 计算方法如式(16)所示:

$$\bar{S} = \frac{\sum_{i=1}^n \sum_{j=1}^8 \|P_i(x,y) - O_j(x,y)\|}{n}, \quad (15)$$

$$\bar{P} = \frac{P_n - L_n}{L_n} \times 100\%, \quad (16)$$

式中 n 为节点数, P_n 为算法规划后的路径所经节点数, L_n 为始终点直线所经过节点数, 性能比较结果如表1所示.

4.3 实际实验

对于改进定位导航策略进行实际测试. 图14为改进MS-MCL定位算法与MCL定位算法性能对比图. 水平线是机器人静止状态的定位, 中间段是运动过程中机器人从 X , Y 两个维度描述的实时定位, MS-MCL相比MCL算法在性能上有较好的改善, 说明其在定位中积累误差得到有效削减, 有利于我们进一步进行了导航算法的测试研究.

在本文定位算法的基础上我们进行移动机器人实

表 1 算法性能比较表

Table 1 Algorithm performance comparison table

算法	路径长度	时间 (s)	ASD	PS
A-star 算法	27.21	0.293	0.583	14.6
改进 A-star 算法	23.91	0.184	0.788	24.8
性能提升(%)	12.1	37.2	35.2	69.9

图 14 (网络版彩图)MS-MCL 和 MCL 性能对比

Figure 14 (Color online) Performance comparison between MS-MCL and MCL.

际导航算法测试, 图15为网格建立和路径规划的结果图, 图16为实际移动机器人运行序列图, 实际实验表明本文的路径规划导航算法可以完成既定任务, 满足实际导航的要求, 本文导航策略具有实际应用的可行性和正确性.

5 结论

针对移动机器人定位中存在目标误差和多约束条件下A-star导航算法缺乏可行性的问题, 提出了基于Bézier曲线优化的双向A-star算法. 首先, 设计了MS-MCL定位算法, 有效地提高了定位精度, 方便了后续导航; 其次, 改进了A-star算法的启发式功能, 使机器人能够进行双向迭代搜索; 最后, 利用Bézier曲线使全局路径更加平滑, 解决了由于网格划分导致算法实用

图 15 (网络版彩图)网格建立和路径规划

Figure 15 (Color online) Grid establishment and path planning.

性较差的问题。

本文设计了大量的仿真实验和实际实验。与传统的A-star路径规划算法相比,改进算法的路径长度和运行时间分别减少了12.1%和37.2%。除此之外,平均安全距离提升35.2%,路径平滑度提升了69.9%。同时,

图 16 (网络版彩图)机器人实际运行图

Figure 16 (Color online) Actual operation diagram of robot.

实际实验证明,改进后的定位策略能够适应里程计的不精确性,并在保证定位精度的前提下,验证了该导航规划算法的可行性和优越性。

参考文献

- dos Santos D H, Goncalves L M G. A gain-scheduling control strategy and short-term path optimization with genetic algorithm for autonomous navigation of a sailboat robot. *Int J Adv Robotic Syst*, 2019, 16: 172988141882183
- Yang J, Yang Y K, Li J S, et al. A novel satellite-equipped receiver for autonomous monitoring of GNSS navigation signal quality. *Sci China Tech Sci*, 2016, 59: 1137–1146
- Rachid L, Amine S. SLAM algorithm implementation in a UAV, based on a heterogeneous system: A survey. In: The 4th World Conference on Complex Systems (WCCS). Ouarzazate, 2019
- Tateno S, Cho Y, Li D, et al. Improvement of pedestrian dead reckoning by heading correction based on optimal access points selection method. Kanazawa: SICE Annual Conference, 2017, 2017: 321–326
- Liu X C, Wang H, Fu D, et al. An area-based position and attitude estimation for unmanned aerial vehicle navigation. *Sci China Tech Sci*, 2015, 58: 916–926
- Chen H, Li G, Cai J. Spectral-energy efficiency tradeoff in full-duplex two-way relay networks. *IEEE Syst J*, 2015, 6: 1–10
- Doer C, Scholz G, Trommer G F. Indoor laser-based SLAM for micro aerial vehicles. *Gyroscopy Navig*, 2017, 8: 181–189
- Su K, Deng Z, Huang Z. A novel localization approach for autonomous vehicles based on map matching with curvature features. *Robot*, 2013, 34: 440
- Rashidzadeh R, Li S. A hybrid indoor location positioning system. *IET Wireless Sensor Sys*, 2018, 9: 328–343
- Nelabhotla A, Dinamarca C. Bioelectrochemical CO₂ reduction to methane: MES integration in biogas production processes. *Appl Sci*, 2019, 9: 1056–1057
- Ciesielski K C, Falcão A X, Miranda P A V. Path-value functions for which Dijkstra's algorithm returns optimal mapping. *J Math Imag Vis*, 2018, 60: 1025–1036
- Chaari I, Koubaa A, Bennaceur H, et al. Design and performance analysis of global path planning techniques for autonomous mobile robots in grid environments. *Int J Adv Robot Syst*, 2017, 14: 172988141666366

- 13 Ryu K, Dantanarayana L, Furukawa T, et al. Grid-based scan-to-map matching for accurate 2D map building. *Adv Robot*, 2016, 30: 431–448
- 14 Tronarp F, Garcia-Fernandez A F, Sarkka S. Iterative filtering and smoothing in nonlinear and non-Gaussian systems using conditional moments. *IEEE Signal Proc Lett*, 2018, 25: 408–412
- 15 Wang D, Xu C, Yuan P, et al. A revised Monte Carlo method for target location with UAV. *J Intell Robot Syst*, 2020, 97: 373–386
- 16 Yu Y, Liu B, Chen Z. Improving the performance of pseudo-random single-photon counting ranging lidar. *Sensors*, 2019, 19: 3620
- 17 Liang Q, Zhang H, Sun G, et al. Water regions extraction from LiDAR data in coast. *Remote Sens Technol Appl*, 2018, 33: 136–142
- 18 Duchoň F, Babinec A, Kajan M, et al. Path planning with modified a star algorithm for a mobile robot. *Procedia Eng*, 2014, 96: 59–69
- 19 Bakdi A, Hentout A, Boutami H, et al. Optimal path planning and execution for mobile robots using genetic algorithm and adaptive fuzzy-logic control. *Robot Autonom Syst*, 2017, 89: 95–109
- 20 Li S A, Chou L H, Chang T H, et al. Obstacle avoidance of mobile robot based on hyperOmni Vision. *Sens Mater*, 2019, 31: 1021
- 21 Yang D, Xu B, Rao K, et al. Passive infrared (PIR)-based indoor position tracking for smart homes using accessibility maps and A-star algorithm. *Sensors*, 2018, 18: 332
- 22 Sun L Y, Zhu C G. Bézier surfaces with special coordinate curve nets. *J Math Res Appl*, 2019, 39: 700–708
- 23 Tharwat A, Elhoseny M, Hassanien A E, et al. Intelligent Bézier curve-based path planning model using chaotic particle swarm optimization algorithm. *Cluster Comput*, 2019, 22: 4745–4766
- 24 Elhoseny M, Shehab A, Yuan X. Optimizing robot path in dynamic environments using genetic algorithm and bezier curve. *J Intel Fuzzy Sys*, 2017, 33: 2305–2316

Bi-directional smooth A-star algorithm for navigation planning of mobile robots

YUE GaoFeng^{1,2}, ZHANG Meng^{1,2}, SHEN Chao^{1,2} & GUAN XiaoHong^{1,2}

¹ School of Cyber Science and Engineering, Xi'an Jiaotong University, Xi'an 710049, China;

² Ministry of Education Key Lab For Intelligent Network and Network Security, Xi'an Jiaotong University, Xi'an 710049, China

Aiming at solving the problems in existing mobile robot navigation, such as lower-precision localization of the MCL algorithm, poor real-time ability of the traditional A-star algorithm and excessive polylines of the planned path, this paper proposes a novel A-star-algorithm-based mobile smooth navigation strategy. Firstly, the multi-sensor Monte Carlo localization (MS-MCL) algorithm is proposed for the pose estimation. Secondly, by improving the A-star algorithm to the bi-directional pattern and using the cost function iteratively, the algorithm has better real-time performance. Finally, Bézier curve is used to optimize the planned path, which solves the problem that the algorithm cannot be applied in practice because of excessive polylines and tiny turning angles. In comparison with the traditional A-star trajectory planning algorithm, simulation experiments fully proved that the algorithm designed in this paper reduces the path length and running time by 12.1% and 37.2%, respectively, and the average safety distance and the path smoothness are increased by 35.2% and 69.9%, respectively. Meanwhile, the results of the practical experiment verify the correctness and feasibility of this navigation planning algorithm under the guaranteed localization accuracy.

autonomous localization and navigation, MS-MCL localization, A-star algorithm, bi-directional search, Bézier curve

doi: [10.1360/SST-2020-0186](https://doi.org/10.1360/SST-2020-0186)