

[文章编号] 1003-4684(2024)02-0005-05

移动机器人 2D 激光 SLAM 算法仿真与实现

马志艳^{1,2}, 邵长松¹

(1 湖北工业大学农机工程研究设计院, 湖北 武汉 430068;

2 湖北省农机装备智能化工程技术研究中心, 湖北 武汉 430068)

[摘 要] 随着无人驾驶技术的迅速发展,同步定位与建图技术因其精度高、稳定性好的优点备受人们关注。基于激光雷达传感器,选择主流的 Gmapping 和 Cartographer 算法,搭建实验环境,对两种算法进行对比仿真与实验建图,并对两种算法的建图效果进行深度的分析。基于滤波器的 Gmapping 算法计算量小,依赖于里程计信息,适用于小尺度、低特征环境中;基于图优化的 Cartographer 算法累计误差低,精度高,适用于精度和稳定性要求较高的场合。

[关键词] 移动机器人; 2D 激光同步定位与建图; Gmapping 算法; Cartographer 算法; 仿真

[中图分类号] TP242 **[文献标识码]** A

同步定位与建图(simultaneous localization and mapping, SLAM)是指机器人在未知环境运行的过程中,通过传感器获取周围环境数据信息,实现同时定位与地图构建^[1-2]。目前,SLAM 技术的应用已在多个行业初具规模,在机器人、VR/AR、无人机、无人驾驶等领域也有所涉及。根据使用的环境,感知传感器不同逐渐形成视觉 SLAM 和激光 SLAM 两种。其中,视觉 SLAM 可面对高动态环境,有着超强的场景辨识能力,但是受光线影响较大^[2-5];激光 SLAM 能有效克服环境光的影响,可在黑暗、无纹理环境中工作,建图精度较高,稳定性好^[4-9]。本文通过实验来对比分析激光 SLAM 中 Gmapping 和 Cartographer 不同原理算法的定位、建图精度和稳定性、建图耗时以及内存与 CPU 的使用率等。

1 2D 激光 SLAM 算法建图原理

1.1 Gmapping 算法

Gmapping 算法是基于滤波器原理的激光 SLAM 中应用最广泛、效果最好的。它依赖于里程计信息,通过控制数据和观测数据来联合求解位姿和实时建图^[10],使用 RBPF 方法,定位部分与粒子滤波大致相同,完成状态预测、测量、更新、重采样等流程(图 1)。

在 slam 中,机器人在陌生环境获得一张环境地图,同时确定自己在地图中的位置,并对机器人位姿

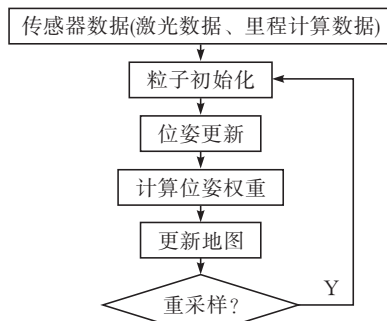


图 1 Gmapping 算法流程

和地图这两个状态变量进行估计。为了降低 SLAM 算法中的复杂性,利用条件贝叶斯法则,把 SLAM 分解为定位和构建地图两步,先定位后建图。计算公式如下:

$$p(x_{1:t}, m | z_{1:t}, u_{1:t-1}) = p(m | x_{1:t}, z_{1:t}) p(x_{1:t} | z_{1:t}, u_{1:t-1}) \quad (1)$$

RBPF 方法使用粒子滤波器进行后验估计,容易产生粒子退化现象,可能丢失重要粒子。Gmapping_slam 针对上述问题,改进提议分布和选择性重采样。融合里程计信息和激光观测模型作为提议分布,降低了粒子耗散问题,提高了建图的精度。提议分布:

$$p(x_t | m_{t-1}^{(i)}, x_{t-1}^{(i)}, z_t, u_{t-1}) = \frac{p(z_t | m_{t-1}^{(i)}, x_t) p(x_t | x_{t-1}^{(i)}, u_{t-1})}{p(z_t | x_{t-1}^{(i)}, u_{t-1})} \quad (2)$$

Gmapping 算法中,选择性重采样可以降低采

[收稿日期] 2022-07-15

[基金项目] 国家重点研发计划基金资助项目(2018YFD0701002-03)

[第一作者] 马志艳(1976-),男,湖北武汉人,工学博士,湖北工业大学副教授,研究方向为计算机视觉与 SLAM 技术。

[通信作者] 邵长松(1996-),男,山东临沂人,湖北工业大学硕士研究生,研究方向为移动机器人的 SLAM 与导航。

样次数,有利于减少粒子退化,是根据所设定的阈值和粒子的权重变化来进行判定的。当粒子的权重大于阈值就进行重采样^[11],权重

$$w_t^{(i)} = w_{t-1}^{(i)} p(z_t | m_{t-1}^{(i)}, x_{t-1}^{(i)}, u_{t-1}) \quad (3)$$

式(1)(2)(3)中: x 表示机器人的位姿状态, z 表示传感器观测的特征, m 表示由 z 观测到的环境地图, u 表示输入控制,下标表示各个状态的时刻。

1.2 Cartographer 算法

Cartographer 算法是由谷歌公司开源的一个实时室内建图算法,生成的栅格地图分辨率可达 $5 \text{ cm}^{[12]}$ 。Cartographer 算法是基于图优化的 SLAM,分为前端和后端(图 2)。前端将实时的激光数据扫描匹配,在得到最佳位姿后,插入到子图中;后端通过闭环检测加约束进行优化,消除前端扫描匹配中位姿估计所产生的累计误差^[13-14]。

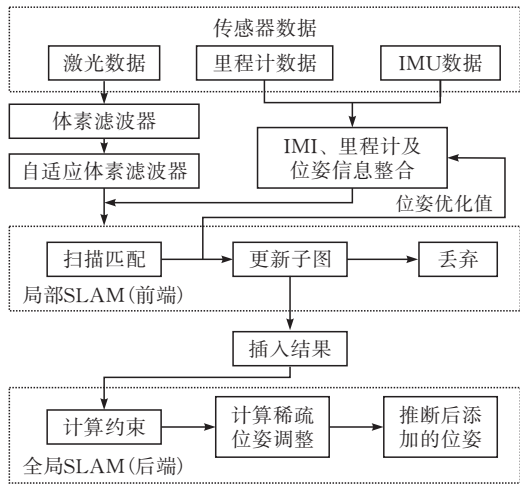


图 2 Cartographer 算法框架

Cartographer 算法的整个系统分为局部优化和全局优化两部分。局部优化就是将激光扫描帧与子图匹配的过程,即将扫描帧插入到子图之前,需要通过 ceres 求解器对扫描帧位姿和子图进行优化,将此问题转化为求解非线性最小二乘问题。其公式如下:

$$\operatorname{argmin}_{\xi} \sum_{k=1} (1 - M_{\text{smooth}}(\mathbf{T}_{\xi} h_k))^2 \quad (4)$$

式中: \mathbf{T}_{ξ} 表示相对转移矩阵; h_k 表示雷达帧中的数据点; M_{smooth} 双立方插值函数。

全局优化在后端进行,通过闭环检测来实现。激光扫描帧仅与近期的子地图进行匹配。而环境地图由一系列子图构成,局部扫描匹配就会逐渐累积误差,长期建图累积的误差需要通过全局优化来进行消除。通过稀疏位姿调整法来优化扫描帧和子图的位姿。激光扫描帧插入到子图的位姿会被储存在内存中,当子图稳定后,所有的激光扫描帧和子图都会被进行闭环检测。

2 SLAM 实验平台构建

试验选择机器人移动底盘 Husky,搭载 TX2 工控机安装 Ubuntu 和 ROS 操作系统,配置 VLP-16 激光雷达和 UM7 惯性单元,另加轮式里程计,同时远程 PC 配置 Ubuntu 18.04 和 melodic ROS 系统。路由器组建局域网,Husky 工控机和远程 PC 通过无线与路由器连接,然后分别配置 TX2 工控机和远程 PC 的 bashrc 文件和 hosts 文件,通过 IP 地址将工控机和 PC 连接,实现远程控制,完成机器人在未知环境中的定位与 SLAM 建图。

2.1 软件平台

操作系统选择 Ubuntu 18.04,安装 melodic 版本的 ROS 作为软件平台。ROS 是一套编写机器人软件的通用框架,支持多种编程语言、代码复用,同时拥有开源、免费等众多优点,集成了大量的工具、库、协议,极大简化机器人平台下的任务创建与行为控制。ROS 集成了 URDF、Gazebo、Rviz 等丰富的工具包,为仿真与实验提供了良好的可视化工具与人机交互界面。

2.2 硬件平台

搭建如图 3 所示的实验平台。传感器选择多线程激光雷达 Velodyne VLP-16,实时扫描当前环境信息;中层控制配置了 NVIDIA Jetson TX2,接收和处理实时数据信息;惯性导航使用 UM7;底盘配置了 Husky A200,控制精度高,应用场景丰富,具有强大的载荷能力与动力系统,速度可达 1 m/s 。



图 3 实验平台

激光雷达 VLP-16 可实现实时收发数据,有效测量范围为 100 m 。其耗电低,设计轻巧(只有 830 g),占用空间少,方便安装以及双重回传功能。VLP-16 支持 16 个通道, 360° 水平视场角, $\pm 15^\circ$ 可调范围的 30° 垂直视场角,以及每秒高达 30 万个点数据输出,多用于无人驾驶、三维建模、机器人导航避障等。

VLP-16 传感器使用飞行时间(ToF)方法。探测器对阵列在其固定外壳内快速旋转,扫描周围环境,通过光线发射和反射时间差或相位差,来计算测量环境的深度信息:

$$D = C(t_1 - t_0) / 2$$

(5)

式中: C 为光速, $t_1 - t_0$ 为光线发射到接收的时间差, D 为目标距离。由于 $t_1 - t_0$ 时间差的数非常小,误差较大,一般采用基于相位差的方法:

$$\Delta t = \frac{\Delta \varphi}{2\pi f_m}$$

(6)

$$D = \frac{C\Delta \varphi}{4\pi f_m}$$

(7)

式中: Δt 为光线发射到接收的时间差, $\Delta \varphi$ 为发射到接收的相位差, f_m 为频率。

3 仿真及实地建图实验

3.1 仿真建图

利用 ROS 中的仿真工具 Gazebo,搭建一个工厂的简单环境模型(图 4)。模型由四周矮墙组成,中间放置多个圆形、方形、长方形等障碍物。



图 4 工厂环境仿真模型

笔记本型号为华硕 VM590,其处理器为 AMD A10-7400P 四核,内存 DDR3L 1600 MHz 12 GB,显卡 AMD Radeon R6 Graphics。安装 Ubuntu 18.04,配置 melodic ROS 操作系统,在搭建好的仿真环境中依次启动所有节点,分别运行 Gmapping 和 Cartographer 建图方案;选择底盘机器人模型 Husky A200,利用模拟的激光雷达 VLP-16 获取周围环境信息,通过 rviz 界面实时观察建图过程(图 5、图 6)。

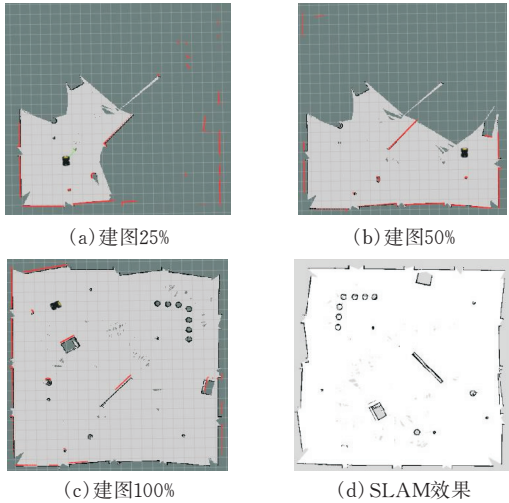


图 5 Gmapping_slam 图

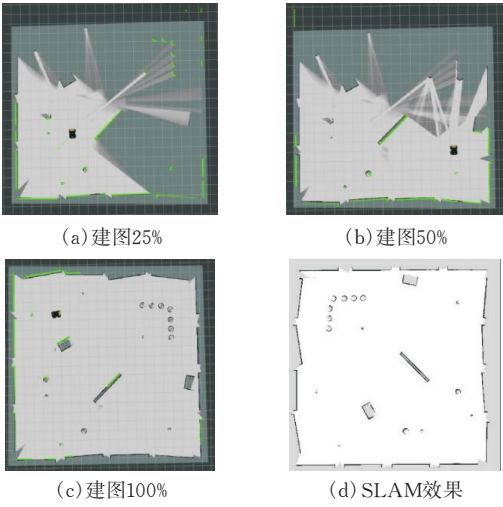


图 6 Cartographer_slam 图

3.2 实地建图

在保证远程 PC 和机器人在同一局域网条件下,通过设置 ROS_MASTER_URI 和 ROS_IP 的 IP 地址,远程连接 PC 与移动机器人控制底盘,PC 端发送指令控制。

机器人在测试环境中正常运行,实验环境选择实验楼大厅、走廊及单个实验室房间内部(图 7)。在测试环境中选择走廊的长度、宽度,实验室的长宽,空调的长度等 8 个测试点进行编号,使用米尺等工具分别测量并记录结构长度,通过 RVIZ 选取相同位置的图测值进行对比,以此评价分析两种算法的建图精度。



图 7 实地建图环境

按照实验方案,远程 PC 与移动机器人远程连接之后,移动机器人端运行传感器和底盘控制程序,PC 端运行 SLAM 与导航算法。机器人端在 NVIDIA Jetson TX2 工控机上启动底盘控制节点;PC 端分别启动 Gmapping 和 Cartographer 的 launch 文件,通过键盘控制节点完成 SLAM 建图。以 Gmapping 建图为例(Cartographer 算法类似),远程 PC 启动 rviz 可视化工具,可以看到 husky 模型和激光雷达扫描出来的边界;启动 gmapping 算法,在 Rviz 可视化器中,选择“navigation”组中的可视化器;启动控制节点,通过远程 PC 上的按键来控制小车运动,实现 slam 建图。建图结束,输入地图保存指令将地图以 Gmapping_slam 为文件名保存

在主目录下。

图 8 为 Gmapping 算法和 Cartographer 算法 SLAM 的最终效果图。

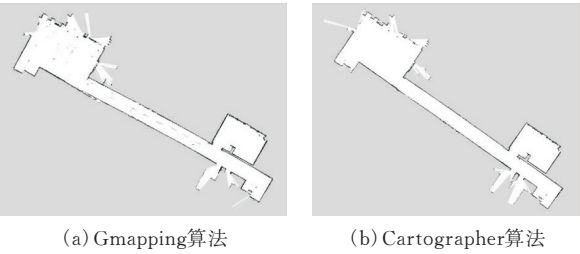


图 8 SLAM 图

4 结果分析与讨论

Gmapping 是基于滤波原理的算法，核心思想是概率问题，对当前位姿进行估计，但没有充分利用历史数据信息，在定位与建图中存在线性化及更新效率等问题，不能适用于满足大尺度的环境。Cartographer 是基于图优化的算法，核心思想是实时构建位姿，充分利用传感器所有历史观测数据，且分为前端和后端，增加了回环检测：前端负责扫描匹配、特征提取和数据关联；后端负责位姿图优化；回环检测环节可以修正累计误差。Cartographer 可适用于室外和室内的大尺度环境中，但在大尺度环境中后端优化时计算量偏大。

从仿真建图效果来看 Cartographer 算法明显优于 Gmapping 算法。图 5d 为 Gmapping_slam 效果图，其构图结构整体比较粗糙，出现了结构错乱的情况，边缘处呈现锯齿状，不够平滑、方正；图 6d 为 Cartographer_slam 效果图，其构图结构清晰方正，无错乱，边界线条平滑，更符合真实的环境情况。

从实地建图(图 8)来看，在测试场景中选择 8 处特征点进行测量，分别进行实地测量和通过 RVIZ 自带的 measure 工具进行图测量，然后通过实测值与图测值分别计算出绝对误差和相对误差的绝对值。其具体数据如表 1、表 2 所示。

表 3 Gmapping 和 Cartographer 算法实验的结果

算法	原理	地图结构 清晰度	地图边 缘形状	建图 时间	内存、CPU 占用率	适合场景
Gmapping	滤波器	结构粗糙清晰度低	呈锯齿状不够平滑	快	低	小尺度、低特征
Cartographer	图优化	结构清晰	线条平滑	慢	高	精度、稳定性要求高

总体来说，Gmapping 算法计算量小，建图用时短，内存和 CPU 占用率低，但非常依赖于里程计信息，构建的地图结构略微粗糙，适用于小尺度、低特征的快速建图。Cartographer 计算量大、建图用时较长，对内存和 CPU 要求较高，但在建图效果与精

表 1 Gmapping SLAM 算法测量数据

测量点	实测值/cm	图测值/cm	绝对误差/cm	相对误差的绝对值/%
1	645	649.4	4.4	0.682
2	197	201.1	3.1	1.574
3	445	428.9	—16.1	3.618
4	3930	3986.6	56.6	1.440
5	960	945.5	—14.5	1.510
6	660	697.3	37.3	5.651
7	95	97.6	2.6	2.747
8	60	58.9	—1.1	1.833

表 2 Cartographer SLAM 算法测量数据

测量点	实测值/cm	图测值/cm	绝对误差/cm	相对误差的绝对值/%
1	645	646.3	2.3	0.357
2	197	198.2	1.2	0.609
3	445	446.9	1.9	0.427
4	3930	3955.8	23.8	0.656
5	960	955.7	—4.3	0.447
6	660	669.1	9.1	1.379
7	95	95.6	0.6	0.667
8	60	58.7	—0.3	0.500

两种算法的相对误差绝对值对比如图 10 所示。Gmapping 算法误差相对较大，折线图上下跳动明显，稳定性较低；Cartographer 算法误差较小，精度高，折线图无大幅度跳动，稳定性较好。

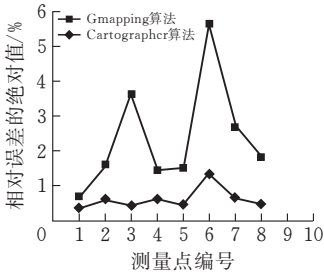


图 10 相对误差的绝对值对比

本文针对 2D 激光 SLAM 中的 Gmapping 和 Cartographer 建图算法进行了仿真和实时地图构建，并对结果进行了分析与讨论。两种算法实验结果如表 3 所示。

度方面，建图效果更全面，构建的地图结构清晰，边缘处平滑，误差低，精度高，远优于 Gmapping，这是因为 Cartographer 基于图优化原理，增加了回环检测，修正了大部分累加误差，适合于精度和稳定性要求较高的应用场合。

5 结束语

SLAM 技术与移动机器人的正常工作密不可分,是移动机器人实现自主定位、环境感知、实时建图、路径规划、自主导航的重要技术支撑。本文基于 ROS 操作系统,讨论了两种不同工作原理的 2D 激光 SLAM 算法,并进行了仿真、实地测试。总体而言,基于图优化的 Cartographer 算法在建图精度、稳定性效果上表现更优;Gmapping 算法的易用性更好,可为室内 SLAM 建图提供一定的借鉴。

[参 考 文 献]

[1] 胡春旭. ROS 机器人开发实践[M]. 北京: 机械工业出版社, 2018: 497-501.

[2] 田野,陈宏巍,王法胜,等.室内移动机器人的 SLAM 算法综述[J].计算机科学,2021,48(09):223-234.

[3] 孙宗涛,朱永强.2D 激光 slam 算法在室内建图对比研究[J].内燃机与配件,2020(01):15-16.

[4] 沈斯杰,田昕,魏国亮,等.基于 2D 激光雷达的 SLAM 算法研究综述[J].计算机技术与发展,2022,32(01):13-18.

[5] 郭金辉,陈秀万,王媛.视觉惯性 SLAM 研究进展[J].火力与指挥控制,2021,46(01):1-8.

[6] WU J,HAO X,ZHENG J. Automatic background filtering and lane identification with roadside LiDAR data [C].2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), 2017:1-6.

[7] GRISETTI G, STACHNISS C, BURGARD W. Improved techniques for grid mapping with rao-blackwellized particle filters[J]. IEEE Transactions on Robotics, 2007, 23(01): 34-46.

[8] 李辉. 基于激光雷达的 2D-SLAM 的研究[D].杭州:浙江工业大学,2017.

[9] HESS W, KOHLER D, RAPP H, et al. Real-time loop closure in 2D LIDAR SLAM[C]. 2016 IEEE International Conference on Robotics and Automation (ICRA), 2016:1271-1278.

[10] 陈智君,郝奇,伍永健,等.基于二元正态分布匹配和非线性优化的激光 SLAM 研究[J].组合机床与自动化加工技术,2021(09):19-23.

[11] 刘媛媛,公建宁,张萌,等.基于单舵轮 AGV 的 Gmapping SLAM 导航算法研究[J].制造业自动化,2020,42(02):128-130.

[12] LEE Y C, YU W, LIM J H, et al. Sonar grid map based localization for autonomous mobile robots[C]. IEEE/ASME International Conference on Mechnronic & Embedded Systems & Applications. IEEE, 2008: 558-563.

[13] LU F, MILIOS E. Globally consistent range scan alignment for environ-globally consistent range scan alignment for environment mapping[J]. Autonomous Robots, 1997, 4(04):333-349.

[14] 张亮,刘智宇,曹晶瑛,等.扫地机器人增强位姿融合的 Cartographer 算法及系统实现[J].软件学报,2020,31(09):2678- 2690.

Simulation and Implementation of 2D Laser Slam
Algorithm for Mobile Robots

MA Zhiyan^{1,2}, SHAO Changsong¹

(1 Agricultural Machinery Engineering Research and Design Institute of
Hubei Univ. of Tech., Wuhan 430068,China ;

2 Hubei Province Agricultural Machinery Equipment Intelligent Engineering
Technology Research Center , Wuhan 430068, China)

Abstract: With the rapid development of unmanned driving technology, the SLAM method has attracted much attention due to its high accuracy and good stability. Based on the lidar sensor, the mainstream Gmapping and Cartographer algorithms are selected; the experimental environment is built; the two algorithms are compared, simulated and experimentally constructed. Finally, the in-depth analysis and discussion of the mapping effects of the two algorithms are carried out. The filter-based Gmapping algorithm has a small amount of calculation and relies on odometer information, and is suitable for small-scale and low feature environments; the graph-based Cartographer algorithm has low cumulative error and high accuracy, and is suitable for applications with high accuracy and stability requirements.

Keywords: mobile robot;2D laser SLAM; Gmapping algorithm; Cartographer algorithm; simulation