

# 基于稀疏位姿优化的激光 SLAM 算法研究

申东, 徐雨航\*, 李强, 邸敬, 黄霞

兰州交通大学电子与信息工程学院, 甘肃 兰州 730070

**摘要** 基于图优化的同步定位与建图(SLAM)算法的后端优化部分一般采用直接非线性优化方法。但是直接非线性优化方法的计算时间与图大小的立方成比例增长,优化大型位姿图成为移动机器人一个比较大的瓶颈问题。因此在基于图优化的框架下,采用基于稀疏位姿优化的 SLAM 算法,通过直接线性稀疏矩阵求解方法来高效计算约束图的大型稀疏矩阵,并与生成树初始化方式进行配合处理和优化。同时在自主搭建的移动机器人平台上进行实验,并在室内不同环境下对基于稀疏位姿优化的 SLAM 算法与 Gmapping、Hector 进行对比分析。结果表明,所提算法不仅在建图精度上有着明显的优势,而且内存占用也更小。

**关键词** 遥感; 激光雷达; 同步定位与建图; 稀疏位姿优化; 楚列斯基分解

中图分类号 TP242.6

文献标志码 A

doi: 10.3788/LOP202158.2428003

## Research on Laser SLAM Algorithm Based on Sparse Pose Optimization

Shen Dong, Xu Yuhang\*, Li Qiang, Di Jing, Huang Xia

School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, Gansu 730070, China

**Abstract** The back-end optimization part of the simultaneous localization and mapping (SLAM) algorithm based on graph optimization generally uses a direct nonlinear optimization method. However, the calculation time of the direct nonlinear optimization method increases proportionally with the cube of the graph size, and optimizing large-scale pose graphs has become a crucial bottleneck for mobile robots. Therefore, under the framework of graph optimization, the SLAM algorithm based on sparse pose optimization is used in this work to efficiently calculate the large sparse matrix of the constraint graph through the direct linear sparse matrix solving method. Additionally, it is processed and optimized by using the spanning-tree initialization method. At the same time, experiments are performed on an autonomously built mobile robot platform and the SLAM algorithm based on sparse pose optimization is compared with Gmapping and Hector algorithms in different indoor environments. Results show that the proposed algorithm is superior in mapping accuracy and has a lower CPU load.

**Key words** remote sensing; laser radar; simultaneous localization and mapping; sparse pose optimization; Cholesky decomposition

**OCIS codes** 280.3640; 150.5758; 150.1135

## 1 引言

同步定位与地图构建(SLAM)是机器人实现自主导航的关键。近年来,对 SLAM 的研究取得了很

大的进展。根据所采用的数学优化框架,激光 SLAM 可分为基于滤波和基于图形的优化激光 SLAM<sup>[1]</sup>。目前激光 SLAM 算法中基于滤波的算法采用 Markov 假设,仅考虑相邻数据之间的转换,

收稿日期: 2020-12-04; 修回日期: 2021-01-20; 录用日期: 2021-02-17

基金项目: 国家自然科学基金(61741113)、甘肃省科技计划(17JR5RA097)、甘肃省高等学校创新能力提升项目(2019B-052)

通信作者: \*1254871265@qq.com

随着地图逐渐增大,地图创建的累计误差会不断加大,影响建图精度<sup>[2]</sup>。因此将图论的思想融入到 SLAM 中的方法在提出后便受到越来越多的研究人员关注。

在激光 SLAM 领域, Lu 等<sup>[3]</sup>最早于 1997 年提出了基于图优化的 SLAM 方法,他们通过迭代最近点(ICP)算法扫描匹配确定扫描之间的成对匹配点,然后通过迭代线性化优化位姿图。Olson 等<sup>[4]</sup>则采用随机梯度下降法来优化位姿图。这种方法的优点是易于实现且能够减小对初始值的估计误差。后来, Grisetti 等<sup>[5]</sup>应用基于树的参数化扩展了这种方法,从而大大提高了收敛速度。这些方法的主要问题是它们假定位姿图中的误差是均匀的,因此很难应用于某些约束条件未指定的图形<sup>[6]</sup>。考虑到系统的稀疏性, Dellaert 等<sup>[7]</sup>提出了一种使用稀疏直接线性解算器的  $\sqrt{\text{SAM}}$  算法。Kaess 等<sup>[8]</sup>随后提出了一种关于  $\sqrt{\text{SAM}}$  的改进算法,称为 iSAM,它对与非线性最小二乘问题相关的线性矩阵进行增量更新。然而直接非线性优化的计算时间与图大小的立方成比例增长,在优化大型位姿图时存在很大的问题<sup>[9-11]</sup>。

在基于图优化的激光 SLAM 算法基础上,本文采用稀疏位姿优化方法,从约束图中高效计算稀疏矩阵,并用直接稀疏线性方法求解;在 Levenberg-Marquardt(LM)优化过程中使用楚列斯基分解实现稀疏矩阵  $\mathbf{H}$  下的状态增量  $\Delta x$  的高效快速求解。所提算法考虑约束包含的协方差信息,在每一次迭代中均会对当前位姿的所有约束进行线性化,提高了精度且具有非常快的收敛速度。

## 2 基于稀疏位姿优化的激光 SLAM 算法

### 2.1 基于图优化的 SLAM 框架

基于滤波的 SLAM 算法思路是用前一个时刻的值来估计下一个时刻<sup>[12]</sup>。而优化则是把所有状态看成变量,把运动方程和观测方程看成变量间的约束,构造误差函数,然后最小化这个误差<sup>[13]</sup>。

位姿图由一系列非线性约束连接机器人位姿点构成,这些约束来自于位姿点周围共同的特征观测<sup>[14]</sup>。在基于图优化的 SLAM 中,机器人的位姿由图中的节点建模,并用它们在环境中的位置来标记。里程计测量的观测结果产生的位姿之间的空间约束被编码在节点之间的边上<sup>[15]</sup>,更详细地说,基

于图优化的 SLAM 算法根据初始传感器测量值构建图形。图中的每个节点代表一个机器人的位姿和获得的测量值。两个节点之间的边表示与机器人位姿相关的空间约束。约束在于两个位姿之间相对变换的概率分布。一旦构建了图形,就需要找到最能满足约束的机器人位姿配置。因此,在基于图形的 SLAM 中,问题被分解为两个任务:1)从原始测量值构建图形也即图形构建;2)在给定图形边缘的情况下确定位姿的最可能配置也即图形优化<sup>[16]</sup>。图形构建通常被称为前端,主要基于传感器数据确认约束关系;而图形优化则被称为后端,通过获取的约束关系去校正机器人的位姿。在图优化 SLAM 中,一般将前端和后端分开处理,如图 1 所示。这样要求是因为前端需要一个半优化状态的地图去限制潜在在约束关系的搜索范围。当后端的估计越准,前端生成的约束关系越鲁棒且越快。因此,优化问题的精度和性能对整个建图系统具有决定性的影响。

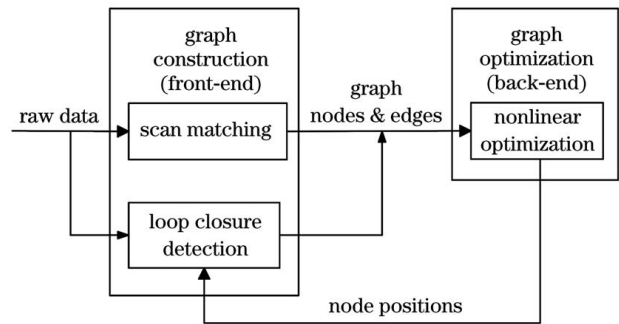


图 1 图优化 SLAM 框架图

Fig. 1 Graph-based SLAM framework

### 2.2 改进 LM 迭代后端优化方法

为了对位姿和约束进行优化,使用 LM 算法作为框架,在 2D 建图中对于稀疏矩阵,采用特殊方法来提高效率,这类类似于计算机视觉中通过 LM 有效处理相机位姿和特征的稀疏优化方法。系统的变量是机器人全局位姿  $\mathbf{c}$  的集合,由位移和方位角组成:

$$\mathbf{c}_i = [t_i, \theta_i] = [x_i, y_i, \theta_i]^T, \quad (1)$$

式中:  $t_i$  为机器人的位置信息;  $\theta_i$  为机器人在地图中的航向角;  $(x_i, y_i)$  为机器人在地图中的具体位置。约束是从一个节点  $\mathbf{c}_i$  到另一个节点  $\mathbf{c}_j$  间的测量值。在节点  $\mathbf{c}_i$  和  $\mathbf{c}_j$  之间的测量偏差为  $\bar{\mathbf{z}}_{ij}$ , 其精度矩阵为  $\mathbf{\Lambda}_{ij}$  (协方差的逆)。真实情况下,任意的节点  $\mathbf{c}_i$  和  $\mathbf{c}_j$  之间的偏差为

$$h(\mathbf{c}_i, \mathbf{c}_j) \equiv \begin{cases} \mathbf{R}_i^T(t_j - t_i) \\ \theta_j - \theta_i \end{cases}, \quad (2)$$

式中:  $\mathbf{R}_i$  是关于  $\theta_i$  的  $2 \times 2$  旋转矩阵;  $h(\mathbf{c}_i, \mathbf{c}_j)$  也被



一列都有一个元素,最后一个是非零元素的总数, col\_ptr 指向每一列的起始位置。最后,由 row\_ind 给出列中每个元素对应的行索引。

CCS 能够高效地存储大型稀疏矩阵,但是因为每个非零元素插入某个列时都会引起该列所有序列的移动,所以频繁插入时会导致复杂度较高。

最有效的方法是按列的顺序创建稀疏矩阵的,这也需要循环遍历约束  $\|c\|$  次。而在此只需要遍历约束一次,并将每个  $3 \times 3$  的矩阵块  $J_i^T \Lambda_{ij} J_i$  存储至一个特殊的指向性块数据结构中,该数据结构与 CCS 格式基本一致。创建稀疏矩阵的过程如图 3 所示。

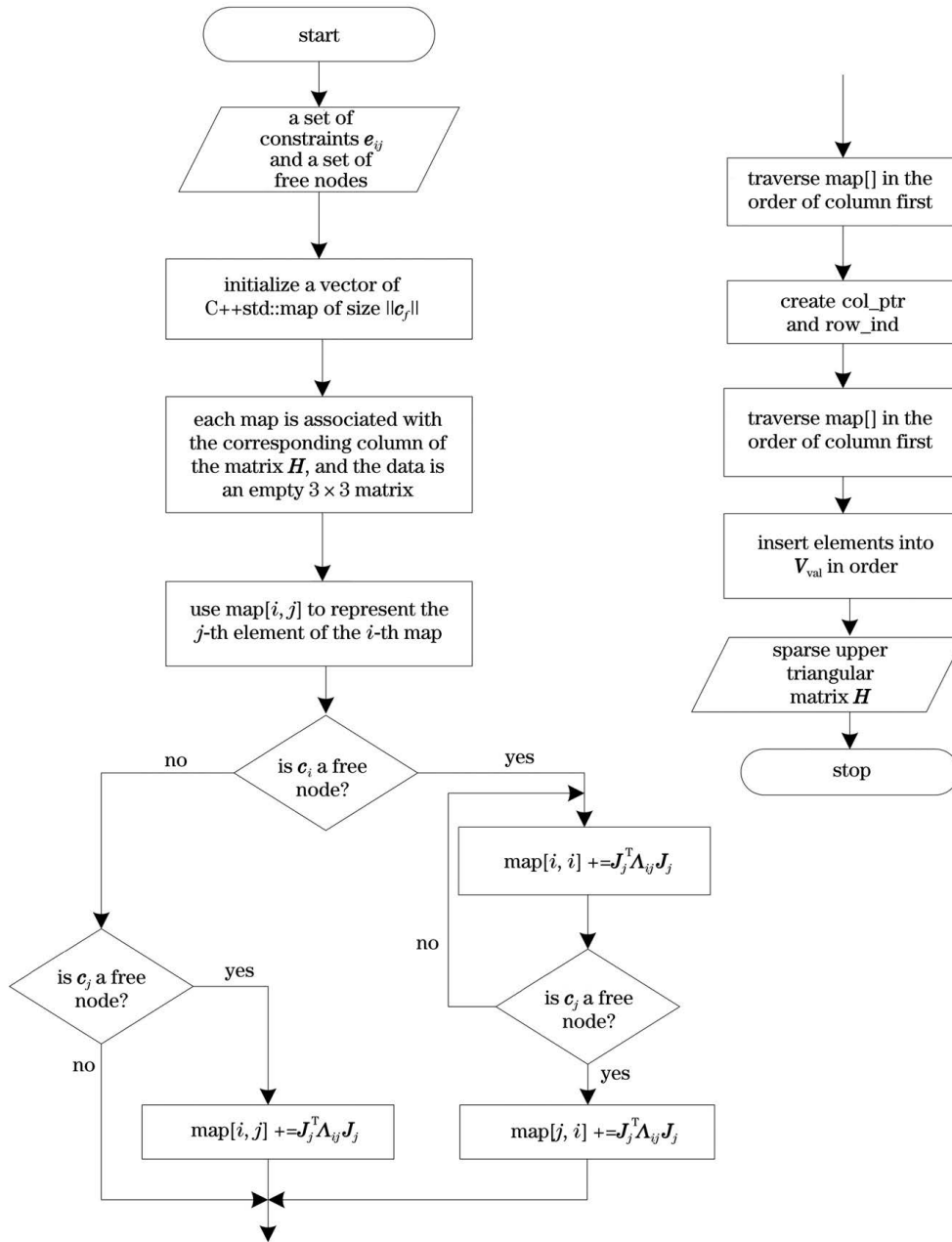


图 3 稀疏矩阵的建立流程图  
Fig. 3 Flow chart of sparse matrix establishment

LM 系统算法过程如图 4 所示。运行单次迭代允许 LM 进行增量操作,以便在迭代之间添加更多节点。该算法是可持续的,因为在迭代之间保存了  $\lambda$ ,因此在下一次迭代时可以根据其结果调整  $\lambda$ 。增加几个节点和测量值不会对系统产生太大的变化,

因此  $\lambda$  值能够体现梯度下降法和牛顿-欧拉法的状态信息。当出现回环时,系统可能无法找到比较好的最小值,此时会在下几次迭代时提高  $\lambda$ ,选用系统传向较好的搜索路径。系统启动时, $\lambda$  值设置为  $10^{-4}$ 。如果一次更新之后比上一次的误差更小,则



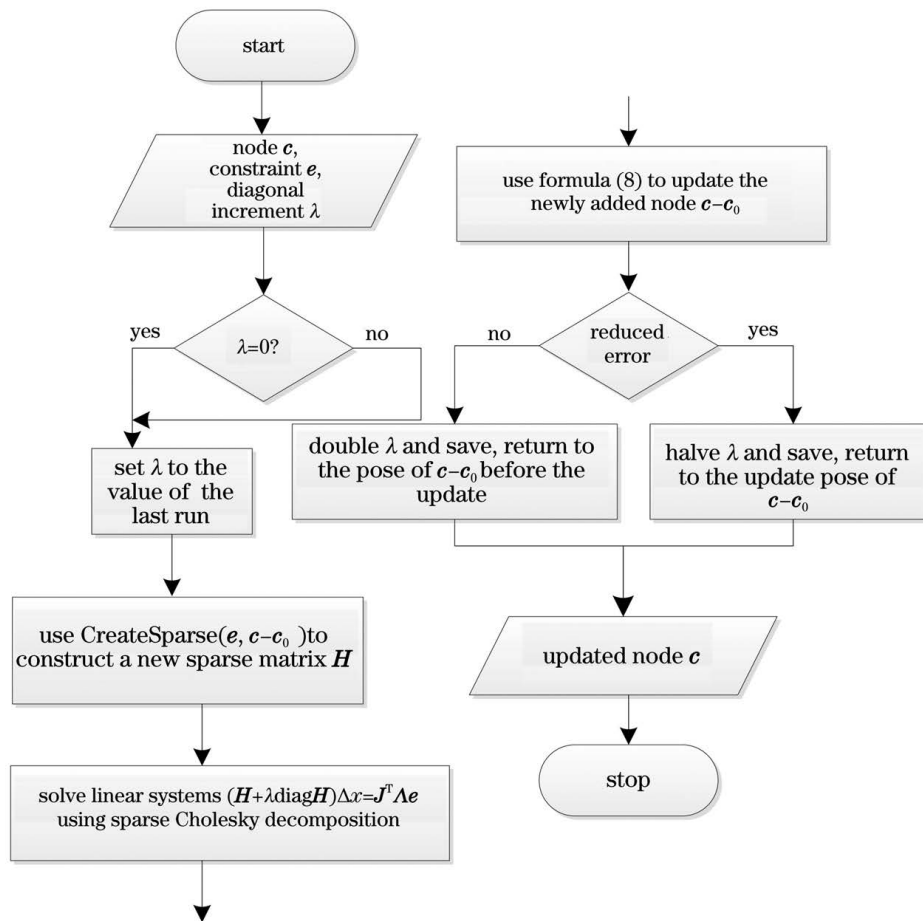


图 4 LM 系统算法

Fig. 4 Continuable LM algorithm

$\lambda$  减半;如果误差相同或者更大,则  $\lambda$  加倍,这在增量式优化的过程中非常有效。只要在增加节点时误差减小,那么  $\lambda$  同样减小,系统保持在牛顿-欧拉法的区域里。当增加一个连接约束,导致系统出现无法矫正的干扰时,则  $\lambda$  增大,系统退回到更加鲁棒的梯度下降法。

另外,对于在离线优化时初始化参数不好可能导致的错误,可以在参数初始化时与生成树初始化方式进行配合处理。优化使用广度优先访问在图上构生成树,其中树根是图中的第一个节点。根据生成树的深度优先访问来初始化节点的位置,子对象的位置被设置为根据连接约束转换的父对象的位置。该初始化方法在某些情况下比起常用的基于里程计的方法表现效果更好。

### 3 实验结果及分析

#### 3.1 硬件系统架构

实验平台采用自主搭建的四轮移动机器人。该平台主要由 RPLIDAR A2 激光雷达、NVIDIA

Jetson TX2、STM32F1 驱动板、GY-85 九轴陀螺仪、大负载带编码器电机和编程操作平台组成。整体系统硬件框图如图 5 所示。

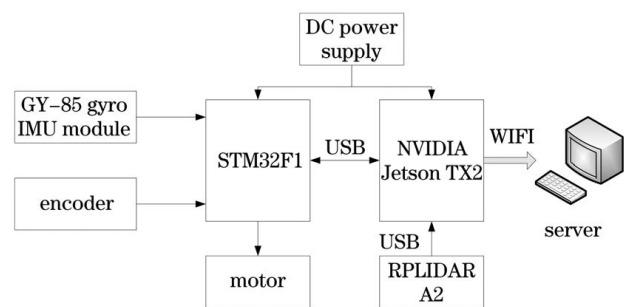


图 5 系统的整体硬件结构图

Fig. 5 Overall hardware structure diagram of the system

主控部分采用 NVIDIA Jetson TX2。机器人的底层驱动是 STM32F1 驱动板和大功率带编码器电机。激光数据由 RPLIDAR A2 激光雷达采集,其具有 8000 次/s 的高速测量频率,可以在二维平面 18 m 的半径范围内进行 360°全方位的激光测距扫描,并产生所在空间的点云地图信息。

360 线 AB 编码器用于测量机器人左右轮的运动速度, GY-85 模块用于测量机器人在三维空间的姿态、方向和加速度。通过 STM32F1 进行 PID 控制, 机器人不仅可以精确地移动, 而且主控制器还输出机器人左右轮的速度、角度、里程等信息来进行定位。服务器使用虚拟机, 在惠普笔记本电脑上安装 Ubuntu 16.04 操作系统和机器人操作系

统(ROS)。ROS 分布式处理框架用于将数据无线传输到服务器, 生成机器人周围环境地图。移动机器人由电压为 12 V, 电流为 8400 mAh 的可充电锂电池供电。图 6 为移动机器人主要部件及实物图, 基于系统的整体硬件框图搭建了一个四轮移动机器人, 可以搭载多种传感器且运行平稳。

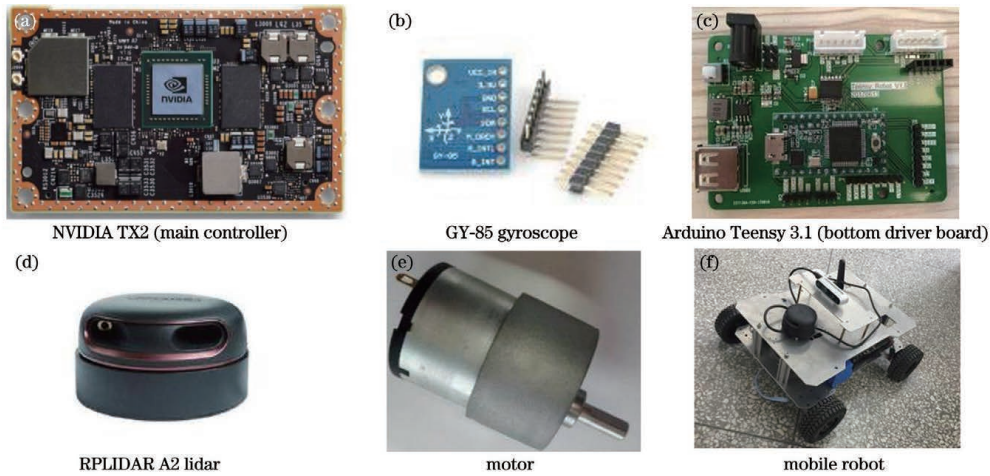


图 6 移动机器人主要部件及实物图

Fig. 6 The main parts and physical maps of the mobile robot

### 3.2 分析与讨论

Gmapping 是一种采用 RBPF 算法的基于滤波的 SLAM 算法, 在 2D SLAM 中应用广泛。而 Hector 是一种使用高斯-牛顿法的基于优化的 SLAM 算法(解最小二乘问题), 不需要里程计, 但对激光雷达要求较高。同时对 Gmapping、Hector 以及基于稀疏位姿优化的 SLAM 算法进行实验。为了充分地体现算法的建图能力, 分别在室内长廊型环境和多障碍物的复杂实验室环境下进行实验。

在实验过程中, 统一将移动机器人的线速度设置为 0.5 m/s, 角速度设置为 0.3 rad/s, 同时采用 Rviz 进行地图可视化。Rviz 是机器人操作系统中非常强大的三维可视化工具, 可以在模型世界中显示传感器数据。保证 PC 端和机器人在同一局域网下, 通过无线网络连接, 在 PC 端服务器打开两个终端, 一个用于启动相应的节点, 另一个作为控制端向机器人发送命令。完成以上操作, 建图结果如图 7 所示。

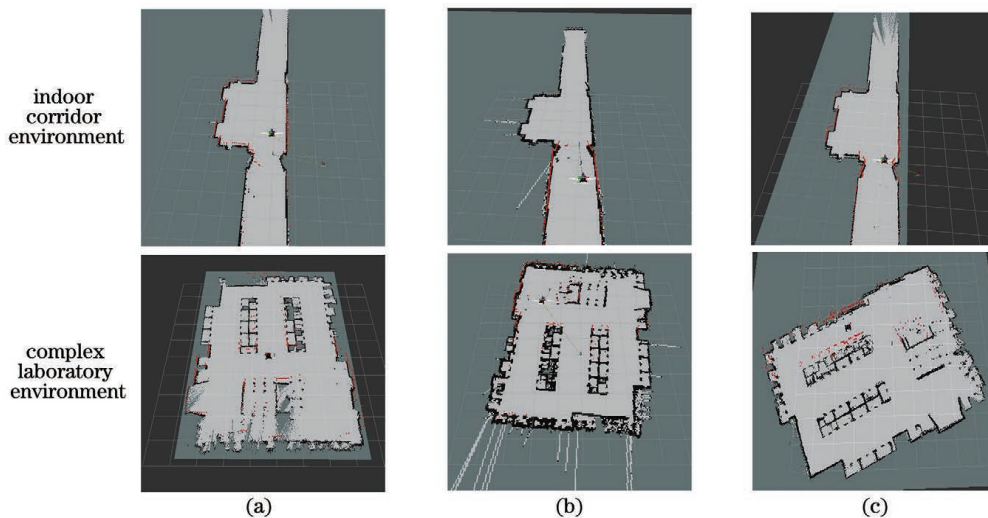


图 7 三种算法构建地图实验结果。(a)Gmapping;(b)Hector;(c)所提算法

Fig. 7 Experimental results of the three algorithms for constructing map. (a) Gmapping; (b) Hector; (c) proposed algorithm

根据图 7 的实验结果,可以看出基于稀疏位姿优化的 SLAM 算法的建图精度优于 Gmapping 和 Hector 算法。尤其在障碍物多的复杂实验室环境中,Gmapping、Hector 算法所建地图毛刺较多,且存

在重影的现象,而基于稀疏位姿优化的 SLAM 算法所建地图较为规整,可以较好地反映出真实环境。为了进一步体现算法的性能,选取机器人路径上的十个特征点进行误差分析,量化结果如表 1 和表 2 所示。

表 1 三种算法在长廊型环境下的误差分析

Table 1 Error analysis of the three algorithms in the corridor environment

Parameter	Gmapping	Hector	Proposed algorithm
Max error /m	0.0790	0.09300	0.0770
Min error /m	0.0250	0.03200	0.0280
Mean square error(MSE) /m <sup>2</sup>	0.0023	0.01870	0.0023
Root mean square error(RMSE) /m	0.0482	0.13697	0.0481
Mean absolute error(MAE) /m	0.0443	0.05140	0.0452

表 2 三种算法在复杂实验室环境下的误差分析

Table 2 Error analysis of the three algorithms in complex laboratory environment

Parameter	Gmapping	Hector	Proposed algorithm
Max error /m	0.0310	0.0520	0.0250
Min error /m	0.0170	0.0240	0.0130
MSE /m <sup>2</sup>	0.0005	0.0011	0.0003
RMSE /m	0.0231	0.0343	0.0194
MAE /m	0.0227	0.0329	0.0190

根据表 1 和表 2 中的数据可以看出:在简单的长廊型环境下,基于稀疏位姿优化的 SLAM 算法建图误差与 Gmapping 基本相同,比 Hector 算法低;在复杂的实验室环境中,所提算法建图误差更明显低于其他两种算法。按照绝对误差值绘制的误差曲线如图 8 和图 9 所示,可以更加直观地反映出所提算法建图时具有良好的鲁棒性。

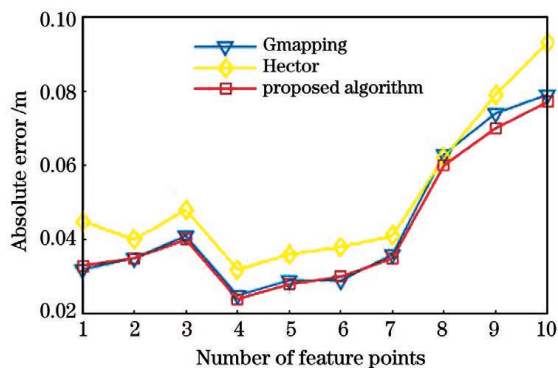


图 8 三种算法在长廊型环境下的误差曲线

Fig. 8 Error curves of the three algorithms in corridor

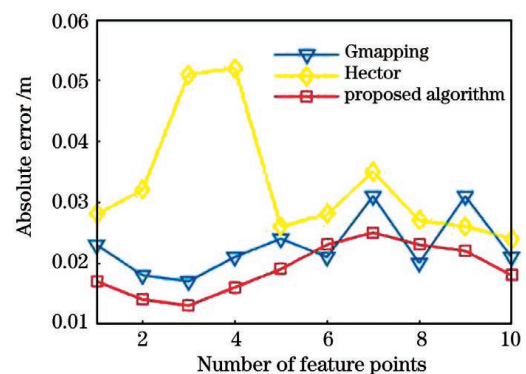


图 9 三种算法在复杂实验室环境下的误差曲线

Fig. 9 Error curves of the three algorithms in complex laboratory

所提算法因为只需要很少的迭代步骤,因此收敛速度很快,在一定程度上提高了计算速度,减少了内存消耗。在统一将机器人线速度设置为 0.5 m/s,角速度设置为 0.3 rad/s 的情况下,通过对比表 3 与表 4 中三种算法的 CPU 占用率以及运行时间,所提算法的计算性能更好,内存占用更小。

表 3 三种算法的 CPU 占用率  
Table 3 CPU load of the three algorithms

Parameter	Gmapping	Hector	Proposed algorithm
Mean CPU load /%	7.0738	6.1213	5.3897
Median CPU load /%	5.6100	5.9310	5.3010
Standard deviation of CPU load /%	4.4312	1.987	1.310

表 4 三种算法的运行时间

Table 4 Running time of the three algorithms unit: s

Environment	Gmapping	Hector	Proposed algorithm
Corridor	25.81	28.64	24.40
Laboratory	105.92	120.48	98.04

## 4 结 论

在基于图优化的框架下,采用了一种基于稀疏位姿优化的 SLAM 算法来有效处理 2D 位姿图的优化问题。该算法通过采用高效的线性矩阵构建和稀疏非迭代乔里斯基分解,可以有效表示和求解大型稀疏位姿图。线性系统在求解过程中能够有效减小内存消耗,因此能够显著提升计算性能。实验结果表明,与 2D SLAM 中常用的 Gmapping 和 Hector 算法相比,基于稀疏位姿优化的 SLAM 算法的性能更加优越,不仅在建图精度上有着明显的优势,鲁棒性更好,且能很好地适应于复杂的环境。在后续的工作中,将在现有的实验平台基础上加入深度相机,实现激光 SLAM 与视觉 SLAM 的融合。

## 参 考 文 献

- [1] Wang Z L, Zhao J, Cai H G. A survey of front-end method for graph-based SLAM under large-scale environment [J]. Journal of Harbin Institute of Technology, 2015, 47(1): 75-85.  
王忠立, 赵杰, 蔡鹤皋. 大规模环境下基于图优化 SLAM 的图构建方法 [J]. 哈尔滨工业大学学报, 2015, 47(1): 75-85.
- [2] Sang D. Research on AGV indoor autonomous navigation algorithm based on graph optimized laser SLAM [D]. Jinan: Shandong University, 2019.  
桑迪. 基于图优化激光 SLAM 的 AGV 室内自主导航算法研究 [D]. 济南: 山东大学, 2019.
- [3] Lu F, Milios E. Globally consistent range scan alignment for environment mapping [J]. Autonomous Robots, 1997, 4(4): 333-349.
- [4] Olson E, Leonard J, Teller S. Fast iterative alignment of pose graphs with poor initial estimates [C]//Proceedings 2006 IEEE International Conference on Robotics and Automation, May 15-19, 2006, Orlando, FL, USA. New York: IEEE Press, 2006: 2262-2269.
- [5] Grisetti G, Stachniss C, Burgard W. Nonlinear constraint network optimization for efficient map learning [J]. IEEE Transactions on Intelligent Transportation Systems, 2009, 10(3): 428-439.
- [6] Montemerlo M, Thrun S, Koller D, et al. FastSLAM: a factored solution to the simultaneous localization and mapping problem [C]//Proceedings of the AAAI National Conference on Artificial Intelligence, February 2-9, 2002, New York. California: AAAI Press, 2002: 593-598.
- [7] Dellaert F, Kaess M. Square-root SAM: simultaneous localization and mapping via square root information smoothing [J]. The International Journal of Robotics Research, 2006, 25(12): 1181-1203.
- [8] Kaess M, Ranganathan A, Dellaert F. iSAM: incremental smoothing and mapping [J]. IEEE Transactions on Robotics, 2008, 24(6): 1365-1378.
- [9] Yang K. Research on 2D SLAM method in dynamic scene [D]. Beijing: Beijing Jiaotong University, 2018.  
杨阔. 动态场景下的 2D SLAM 方法研究 [D]. 北京: 北京交通大学, 2018.
- [10] Zhang Y, Zheng X F, Luo Y, et al. SLAM algorithm with Gaussian distributed resampling Rao-Blackwellized particle filter [J]. Control and Decision, 2016, 31(12): 2299-2304.  
张毅, 郑潇峰, 罗元, 等. 基于高斯分布重采样的 Rao-Blackwellized 粒子滤波 SLAM 算法 [J]. 控制与决策, 2016, 31(12): 2299-2304.
- [11] Montemerlo M, Thrun S, Koller D, et al. Fast SLAM 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably intensive [C]//Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI), August 9, 2003, Menlo Park, CA, USA. Menlo Park: AAAI, 2003: 1151-1156.
- [12] Yin Z, Lü Y J, Lü J N. An optimized SLAM substation complex environment map construction [J]. Journal of Shandong University of Science and



- Technology (Natural Science), 2020, 39(2): 126-132.
- 尹政, 吕英俊, 吕嘉男. 一种优化的 SLAM 变电站复杂环境地图构建[J]. 山东科技大学学报(自然科学版), 2020, 39(2): 126-132.
- [13] Liu D, Duan J M, Meng X Y. A SLAM system of intelligent vehicle based on laser radar [J]. Laser Journal, 2018, 39(7): 76-82.
- 刘丹, 段建民, 孟晓燕. 基于激光雷达的智能车 SLAM 系统[J]. 激光杂志, 2018, 39(7): 76-82.
- [14] Segura M J, Auat Cheein F A, Toibero J M, et al. Ultra wide-band localization and SLAM: a comparative study for mobile robot navigation [J]. Sensors, 2011, 11(2): 2035-2055.
- [15] Grisetti G, Stachniss C, Burgard W. Improved techniques for grid mapping with Rao-Blackwellized particle filters [J]. IEEE Transactions on Robotics, 2007, 23(1): 34-46.
- [16] Wang Y R, Deng G Q, Liu Y, et al. Optimal design of RBPF-SLAM system based on LIDAR sensor [J]. Transducer and Microsystem Technologies, 2017, 36(9): 77-80.
- 王依人, 邓国庆, 刘勇, 等. 基于激光雷达传感器的 RBPF-SLAM 系统优化设计 [J]. 传感器与微系统, 2017, 36(9): 77-80.
- [17] Yan L, Dai J C, Tan J X, et al. Global fine registration of point cloud in LiDAR SLAM based on pose graph [J]. Acta Geodaetica et Cartographica Sinica, 2019, 48(3): 313-321.
- 闫利, 戴集成, 谭骏祥, 等. SLAM 激光点云整体精配准位姿图技术 [J]. 测绘学报, 2019, 48(3): 313-321.