Research of Cartographer Graph Optimization Algorithm Based on Indoor Mobile Robot

Jiali Xu¹, Di Wang², Maosheng Liao³, and Wenshuai Shen⁴

School of Mechatronics Engineering and Automation, Shanghai Key Laboratory of Intelligent Manufacturing and Robotics, Shanghai University, Shanghai, China

¹markmarry@shu.edu.cn Email: starryxjl@163.com

Abstract. In the recent years, mobile robot technology has received widespread attention from all over the world, and Simultaneous Localization and Mapping (SLAM) technology has always been a hot topic for researchers. In this paper, the Cartographer graph optimization SLAM algorithm is studied. Aiming at the problem that loopback detection may have wrong loopback points in the presence of noise, similar and uniform characteristics, a delay judgment module is added to filter the wrong loopback points. The Cartographer algorithm before and after optimization is deployed to the mobile robot platform based on ROS, and a 2D laser scanner is used for the map of indoor building. The experiment results prove that after the algorithm is improved, the overall relative error of the map is reduced by 1.148%, and the accuracy of mapping is improved.

1. Introduction

Mobile robot operation technology is at the forefront of current scientific research, symbolizing the direction of high-tech development. At present, more and more autonomous mobile robots have been successfully developed and widely used in the service industry, logistics industry, military and other fields. Especially for indoor mobile robots, their navigation and positioning have become research hotspots in industry and academia [1]. The working environment of indoor mobile robots is more closed and compact than that of outdoor, signals such as Global Positioning System (GPS) are difficult to penetrate buildings and cannot provide environmental information flexibly in unknown environments. The emergence of Simultaneous Localization and Mapping (SLAM) greatly solves the localization problem in the indoor environment. According to different sensors, SLAM is mainly divided into two categories: laser SLAM and visual SLAM. VSLAM has few applications in indoor localization and navigation due to its indirect information, large calculation and storage requirements, strong dependence on light, and heavy network burden. Laser SLAM started early, and has mature technical theory, high accuracy, less calculation, and is not affected by light. Therefore, laser SLAM is widely used in indoor mobile robot.

Robot Operation System (ROS) is the most versatile robot development platform, which can greatly improve the software reuse rate and reduce the development difficulty, and can be deployed on various hardware platforms. ROS integrates various SLAM function packages, such as Gmapping[2], Karto, Hector, Cartographer [3]. Gmapping is a filter-based SLAM technology, it is affected by errors accumulated over time and cannot guarantee stability and accuracy. Cartographer adopts graph-based optimization method, which can obtain relatively high-precision 2D maps in real time and can be applied to large-area map construction.
Nowadays, the mapping technologies used by mobile robots on the market are outdated mostly. Based on advanced theory, it is of more practical significance to deploy a mobile robot platform with advanced technology and realize the decoupling of software and hardware modules, so this paper will study the Cartographer SLAM algorithm based on laser graph optimization and propose an optimization method. Then the Cartographer algorithm before and after optimization is deployed to the ROS-based mobile robot platform, and this paper will take a description of the mobile platform both in hardware and software.

2. Algorithm description

2.1. Creation of local submap
Firstly, the scan data is converted to submap coordinate system by plane transformation.

\[
T_{\xi}h_k = \begin{bmatrix}
\cos \xi_\theta & -\sin \xi_\theta \\
\sin \xi_\theta & \cos \xi_\theta
\end{bmatrix} h_k + \begin{bmatrix}
\xi_x \\
\xi_y \\
\xi_z
\end{bmatrix}
\]

(1)

Where, \(\xi = (\xi_x, \xi_y, \xi_\theta)\) is the pose of the mobile robot in a 2D environment, \(\xi_x\) and \(\xi_y\) respectively represent the amount of translation in the x and y directions, and \(\xi_\theta\) represents the amount of rotation in the two-dimensional plane. The scan data of lidar is expressed as \(H = \{h_k\}_{k=1}^{\infty}, h_k \in \mathbb{R}^3\). \(T_{\xi}\) is the pose transformation matrix of the laser coordinate system relative to the submap coordinate system.

When new scan data is inserted into the submap grid, the probability value of the grid \(x\) is updated by equation (2).

\[
\begin{cases}
\text{odds}(p) = \frac{p}{1-p} \\
M_{new} = \text{clamp}(\text{odds}^{-1}(M_{old}(x)) \cdot \text{odds}(P_{hit}))
\end{cases}
\]

(2)

Before the new scan data is inserted into the submap, Cartographer uses the Ceres-Solver library [4] to perform nonlinear least squares matching (equation 3) on the scan data and the submap to obtain the pose \(\xi\) that maximizes the current point cloud map to the sub-map coordinate system. \(M_{\text{smooth}}\) is the bicubic interpolation function of \(\mathbb{R}^2 \rightarrow \mathbb{R}\).

\[
\arg\min_{\xi} \sum_{k=1}^{K} \left(1 - M_{\text{smooth}}(T_{\xi}h_k)\right)^2
\]

(3)

2.2. Loopback detection
Cartographer uses branch-and-bound scan matching (BBS) [5] method for loopback detection. By constructing a search window \(\mathcal{W}\), the current frame is compared optimally with the submap and the score is evaluated to determine whether the current data frame is a loopback point. In the equation 4, the \(M_{\text{nearest}}\) function is extended to the entire point cloud data plane \(\mathbb{R}^2\) by the \(M_{\text{smooth}}\) function. However, traversing the Search window \(\mathcal{W}\) iteration in this manner was less efficient, so Cartographer used BBS to accelerate the iteration rate.

\[
\xi^* = \arg\min_{\xi \in \mathcal{W}} \sum_{k=1}^{K} M_{\text{nearest}}(T_{\xi}h_k)
\]

(4)

2.3. Global optimization.
In the end, sparse pose adjustment (SPA) [6] is used for global optimization at the back end. It is also a nonlinear least squares optimization problem.
\[
\arg\min_{\Xi^m, \Xi^s} \frac{1}{2} \sum_{ij} \rho(E^2(\xi^m_i, \xi^s_j; \Sigma_{ij}, \xi_{ij}))
\]

(5)

Where, \(\Xi^m\) is the set of map poses, \(\Xi^s\) is the pose of the data key frame, \(\Sigma_{ij}\) is the covariance matrix between \(\xi^m_i\) and \(\xi^s_j\). \(\xi_{ij}\) is the coordinate transformation relationship of the data frame relative to the matched submap \(\xi^m_i\). \(\rho\) is a robust kernel function, which wraps the function \(E\). \(E\) represents the residual between the observed value and the predicted value.

2.4. Improvement of loopback detection

Cartographer uses loopback detection to reduce cumulative error [7]. While, when in the environmental characteristics are highly similar, loopback detection may have wrong loopback points, so a delay judgment module can be used to guarantee the precision of mapping.

Figure 1. Schematic diagram of delay judgment module.

As show in figure 1, a and b represent the corresponding points of the loop pose obtained by the window \(\mathcal{W}\). In the state represented by a and b, the observation data obtained by the robot is very similar. The relative pose \(T_1\) between a and b is given by the match and is represented by the constrained edge formed by a and b in the graph model. During the actual progress of the robot, when repeatedly passing through the same position, a number of pose points similar to a and b will be generated at discrete times. After adding the delay judgment module, \(T_1\) will not be added to the back-end optimization model immediately, but will wait for a period of time \(t\). If there is no corresponding loopback point in this interval, it means that the pose point detected by this loopback a and b have low credibility and can be discarded. When the corresponding loopback points c and d reappear within the interval, the relative poses of a, b, c, and d have the following relationship:

\[
T_1 \cdot T_2 \cdot T_3 \cdot T_4 = I
\]

(6)

Equation 6 is the relationship in an ideal situation. In fact, there are always differences in the observation data between the corresponding points. The value of \(T_1 \cdot T_2 \cdot T_3 \cdot T_4\) will not be strictly equal to the identity matrix \(I\), so a matrix norm threshold \(G\) can be introduced [8]. When the norm of the multiplication of the four matrices is greater than \(G\), the loop detection is considered incorrect; when the norm is less than \(G\), the detected loop is considered valid, and \(T_1, T_3\) is added to the graph optimization model. Then the schematic diagram of the improved cartographer SLAM is shown in figure 2 (a). Figure 2 (b) is a detailed flowchart of the delay judgement module.

Figure 2. (a) Improved cartographer schematic diagram. (b) Delay judgment module flow chart.
3. Experiments and tests
Experiments will be conducted in the small square room and the long gallery respectively in order to test the improved Cartographer algorithm, and the algorithm will be deployed on modern mobile platform Autolabor2.5.

3.1. Mobile Robot Platform
In order to ensure the stability of mobile robot, the four-wheel drive Autolabor2.5 will be selected as the mobile platform which is an open source robot based on ROS. As shown in figure 3 (a), The Raspberry Pi 3b (1.2 GHz Cortex-A53 64-bit CPU, 1 GB RAM, Raspbian OS) development board is used as the core processing unit, and the 4 USB interfaces provided by it can easily carry various sensor devices. The lidar is Rplidar A1, with a scanning range of 12 meters. The motion controller is the Arduino Mega 2560, equipped with PID algorithm combined with the Hall sensor on the motor, which can accurately control the tire rotation speed, and the Arduino will also provide the odometer information required by the ROS system. The software structure information is shown in figure 3 (b), using the distributed characteristics of ROS, running SLAM, path planning and control nodes on the core development board, and the map visualization nodes running on PC (Intel i5-8250U CPU, 8GB RAM). The optimization of the Cartographer algorithm is achieved by modifying the FastCorrelativeScanMatcher class in the Cartographer SLAM package.

![Figure 3. (a)Hardware structure. (b) Software structure.](image)

3.2. Experimental environment
The experimental environment is shown in figure 4. After measurement, the size of the square indoor room in figure 4 (a) is 6.5×6.5m. The maximum length of the long corridor in figure 4 (b) is 22m, and the maximum width is 5m.

![Figure 4. (a) The square room. (b) The long corridor.](image)
3.3. Experimental results
As shown in figure 5. (a) and (b) are mapping results in the square room scenes before and after optimization, (c) and (b) are mapping results in the long corridor before and after optimization. Due to the change of initial pose, the map is not aligned intuitively, and Corresponding points have been marked in this paper. In the small Square room scene, it can be seen from the corresponding three places A, B and C that the improved Cartographer has less glitches on the grid map construction and the surrounding noise points are more sparse. But on the whole, there is no obvious difference in the global consistency of the Cartographer algorithm before and after the improvement in the small room scene.

While in the long corridor scene, there are corners in C and D, the map constructed by Cartographer after the improvement is more closed than before the improvement obviously. Place F is a stainless steel railing. Place G is a hollow area. After the improvement, the map constructed by Cartographer has fewer wall burrs and the surface is flat and even. In the two places A and E, the map constructed by Cartographer before the improvement obviously shifted, and the edge of the tail of the promenade cannot be consistent. However, after the improvement, Cartographer obviously maintains the overall consistency of the appearance of the corridor.

![Figure 5.](image)

The length of position 1 and 2 in the small room scene and position 1, position 2 and position 3 in the long corridor scene were measured by laser rangefinder and compared with the raster map measured by ROS mapping tool. The results are shown in table 1.

| Scene  | Position Number | Actual value | Map measurement (after) | Relative error(%) (after) | Map measurement (before) | Relative error(%) (before) |
|--------|-----------------|--------------|-------------------------|--------------------------|--------------------------|--------------------------|
| Room   | 1               | 450.8        | 443.2                   | 1.686                    | 443.7                    | 1.575                    |
|        | 2               | 650.4        | 656.4                   | 0.923                    | 642.6                    | 1.199                    |
| Gallery| 1               | 485.8        | 481.2                   | 0.947                    | 479.3                    | 1.338                    |
|        | 2               | 1797.8       | 1775.1                  | 1.263                    | 1730.8                   | 3.727                    |
|        | 3               | 2147.7       | 2103.3                  | 2.067                    | 2044.9                   | 4.787                    |
| Average value | - | -     | -                       | 1.377                    | -                        | 2.525                    |

As can be seen from table 1. In the small indoor scene, the relative error at positions 1 and 2 after the improvement is almost same as before. While in the long corridor scene, the relative error is greatly reduced compared with before. The most striking comparison is at position 2 in gallery, where the relative error in the map measurement before improvement is 3.727%, three times the relative error after improvement of 1.263%. The overall average relative error in the two scenes is 1.377% after improvement and 2.525% before, which proves that the improved algorithm has reduced the overall relative error of the map by 1.148% indoors, and the accuracy of building maps has been improved especially in larger area environments.
4. Conclusions
This paper studies the Cartographer graph optimization algorithm. Aiming at the problem that loopback detection may have wrong loopback points in the presence of noise, similar and uniform characteristics, a delay judgment module is added to filter the wrong loopback points. The Cartographer algorithm before and after optimization is deployed to the mobile robot platform based on ROS. And this paper makes a description on hardware and software structure of the mobile platform. Experiments are conducted in the small square room and the long corridor. The visualization is implemented with Rviz. Experiments prove that the improved Cartographer algorithm has reduced the overall relative error of the map by 1.148% indoors, and the accuracy of building maps has been improved especially in larger area environment.

Acknowledgment
This work is supported by the IIOT Innovation and Development Special Foundation of Shanghai under Grant No.2017-GYHLW-01037.

References
[1] Gongxu, L. (2018) Review on the development of indoor navigation and positioning technology. Journal of Navigation and Positioning, 6(02): 7-14.
[2] Grisetti G, Stachniss C, Burgard W. (2005) Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters. In: IEEE International Conference on Robotics and Automation. Barcelona. pp.2432-2437.
[3] Hess W, Kohler D, Rapp H. (2016) Real-time loop closure in 2D LIDAR SLAM. In: IEEE International Conference on Robotics and Automation. Stockholm. pp.1271-1278.
[4] Agarwal S, Mierle K. (2012) Ceres solver. http://ceres-solver.org.
[5] Clausen J. (2003) Branch and bound algorithms-principles and examples. Computer Science.:1-30.
[6] Konolige K, Grisetti G, Kümmerle R, Burgard W, Limketkai B and Vincent R. (2010) Efficient Sparse Pose Adjustment for 2D mapping. In: IEEE/RSJ International Conference Intelligent Robots and Systems. Taipei. pp. 22-29.
[7] Yuanqiang Y. (2017) Research on intelligent wheelchair Interior design and autonomous navigation Technology. Harbin Institute of Technology.
[8] Hao J. (2019) SLAM and navigation robot designs based on the Cartographer algorithm. Shandong university.