

激光 SLAM 理论与实践 - 作业 5

peng00bo00

January 2, 2021

1. 使用高斯牛顿方法进行匹配的代码可参见./gaussian_newton_ws/src/gaussian_newton_scanmatcher/src 路径下 gaussian_newton_methodp.cpp 文件。匹配后的轨迹可参见 Fig.1，其中红色曲线为里程计轨迹，绿色曲线为高斯牛顿匹配后的轨迹。

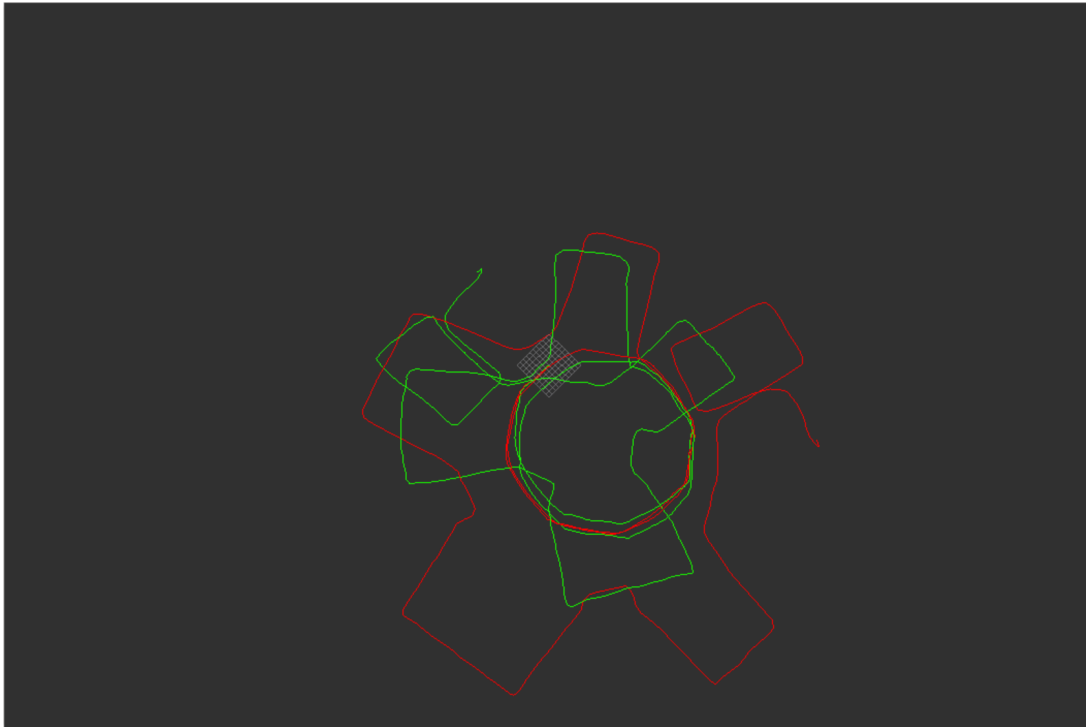


Figure 1: 高斯牛顿匹配

2. 要提高匹配精度可以考虑以下改进：

- (a) 使用更高级的优化算法如 LM 算法等提高每次迭代的精度；
- (b) 计算势场时首先对点云进行过滤，去掉噪声和离群点；
- (c) 对势场进行插值时使用精度更高的插值算法如三次插值等；

3. 1) NDT 算法的优化函数 (score) 为所有点在对应 cell 内的概率之和:

$$\text{score}(p) = \sum_i \exp \left\{ \frac{-(x'_i - q_i)^T \Sigma_i^{-1} (x'_i - q_i)}{2} \right\} \quad (1)$$

其中, p 为待估计参数 $p = (t_x, t_y, \phi)$; x'_i 为原始点 x_i 经过变换后的位置; q_i, Σ_i 为 x'_i 对应 cell 的正态分布均值与方差。

2) NDT 算法优化过程如下:

- (a) 根据上一次的扫描结果初始化 cell 并计算相应的均值和协方差;
- (b) 初始化待估计参数 p ;
- (c) 根据当前的参数 p 将当前扫描结果变换到不同的 cell 中;
- (d) 计算当前变换下对应的优化函数 score;
- (e) 利用高斯牛顿法优化 score 函数并得到新的参数 p ;
- (f) 重复以上步骤直至收敛。

4. 使用深度优先算法 (DFS) 来实现分支定界的步骤如下：

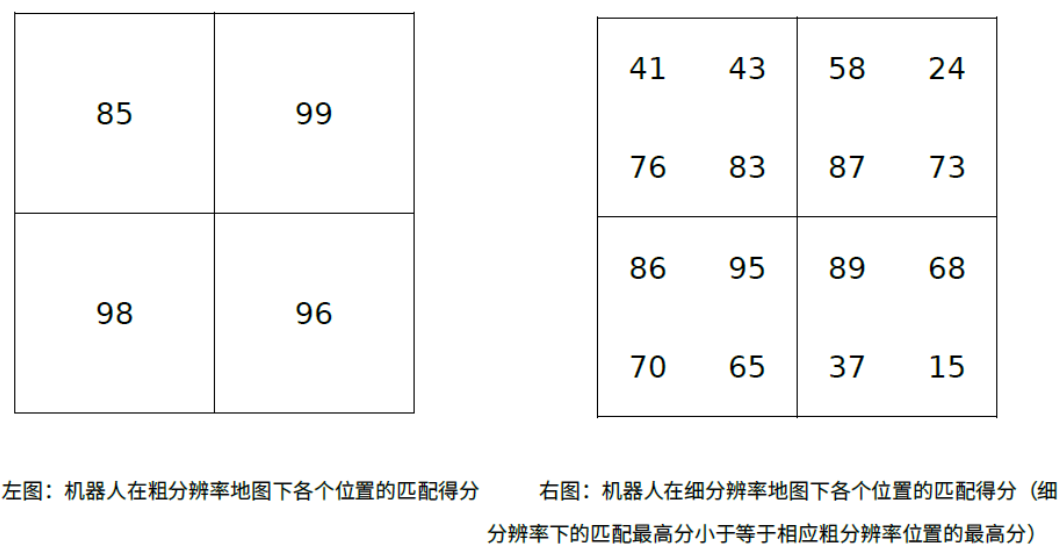


Figure 2: 多分辨率地图

- (a) 初始化栈，将粗分辨率地图的 4 个节点推入栈中并排序，此时栈中节点得分为 [85, 96, 98, 99]；
- (b) 从栈中弹出最后一个节点（得分为 99），由于它不是叶节点对其进行分支，得到 4 个子节点得分分别为 [58, 24, 87, 73]；
- (c) 将子节点全部推入栈中并排序，此时栈中节点得分为 [24, 58, 73, 85, 87, 96, 98]；
- (d) 从栈中弹出最后一个节点（得分为 98），由于它不是叶节点对其进行分支，得到 4 个子节点得分分别为 [86, 95, 70, 65]；
- (e) 将子节点全部推入栈中并排序，此时栈中节点得分为 [24, 58, 65, 70, 73, 85, 86, 87, 95, 96]；
- (f) 从栈中弹出最后一个节点（得分为 96），由于它不是叶节点对其进行分支，得到 4 个子节点得分分别为 [89, 68, 37, 15]；
- (g) 将子节点全部推入栈中并排序，此时栈中节点得分为 [15, 24, 37, 58, 65, 68, 70, 73, 85, 86, 87, 89, 95]；
- (h) 从栈中弹出最后一个节点（得分为 95），由于它是叶节点搜索结束，该节点对应位置即为机器人位置，最后匹配得分为 95；

在深度优先算法中需要不断对栈进行排序，因此可以考虑使用最大堆来代替栈以提高算法运行效率。