



基于图优化(Graph-based) 激光SLAM方法

主讲人 曾书格

越凡创新技术负责人
电子科技大学硕士





Graph-based SLAM



1. Pose Graph的构建



2. 回环检测方法



3. 非线性最小二乘原理



4. 非线性最小二乘在SLAM中的应用



Graph-based SLAM



1. Pose Graph的构建



2. 回环检测方法



3. 非线性最小二乘原理



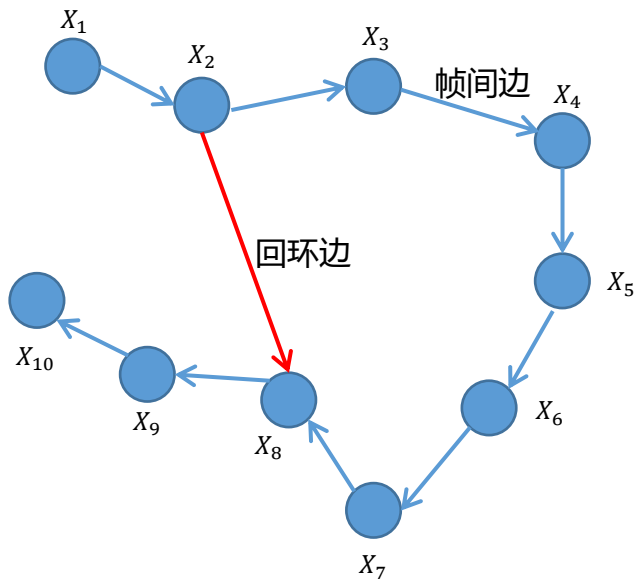
4. 非线性最小二乘在SLAM中的应用



Pose Graph的构建



Pose Graph的概念



Pose Graph示意图

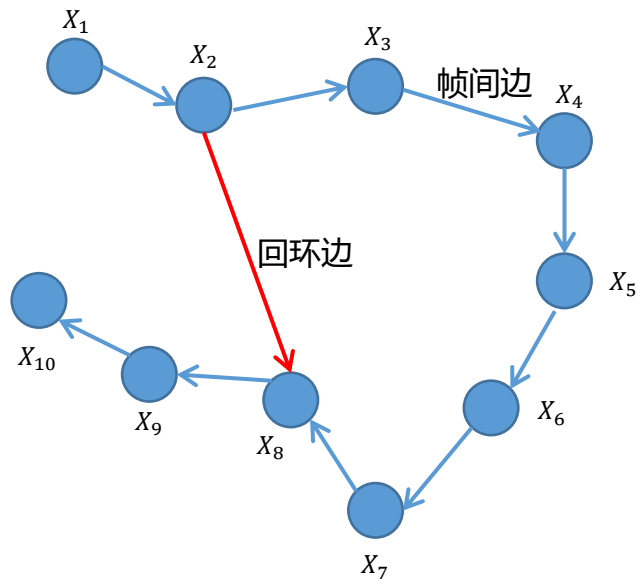
- 用一个图(Graph)来表示SLAM问题
- 图中的节点来表示机器人的位姿(x, y, yaw)
- 两个节点之间的边表示两个位姿的空间约束(相对位姿关系以及对应方差)
- Graph-based SLAM: 构建图, 并且找到一个最优的配置(各节点的位姿), 让预测与观测的误差最小



Pose Graph的构建



Pose Graph的概念



Pose Graph示意图

- 一旦形成回环即可进行优化消除误差
- 里程积分的相对位姿视为预测值
- 回环计算的相对位姿视为观测值
- Graph-based SLAM: 构建图并调整各节点的位姿, 让预测与观测的误差最小

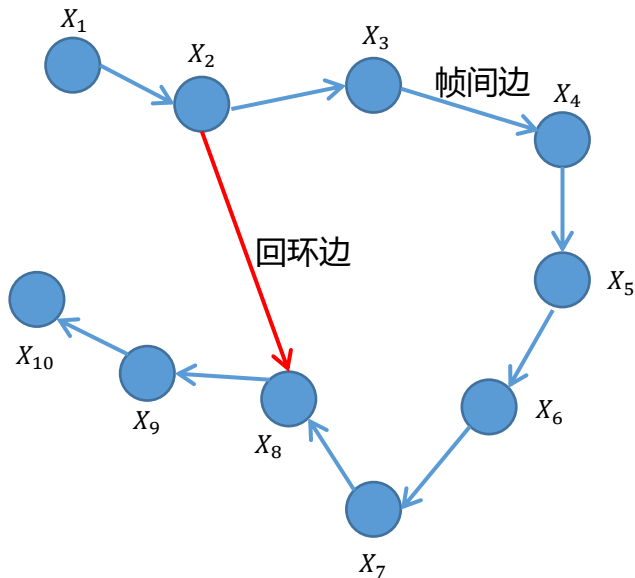
非线性最小二乘优化



Pose Graph的构建

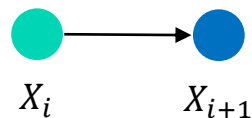


图的构建---帧间边



Pose Graph示意图

- 里程计测量：



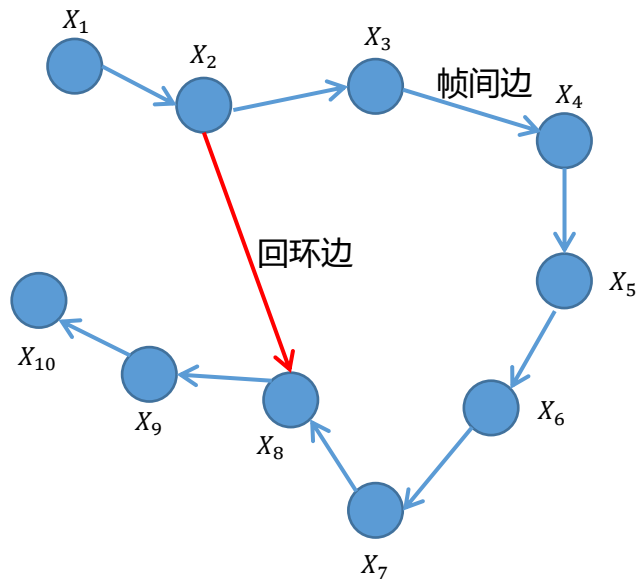
- 相邻节点之间的相对位姿关系，可以由里程计、IMU、帧间匹配计算得到



Pose Graph的构建

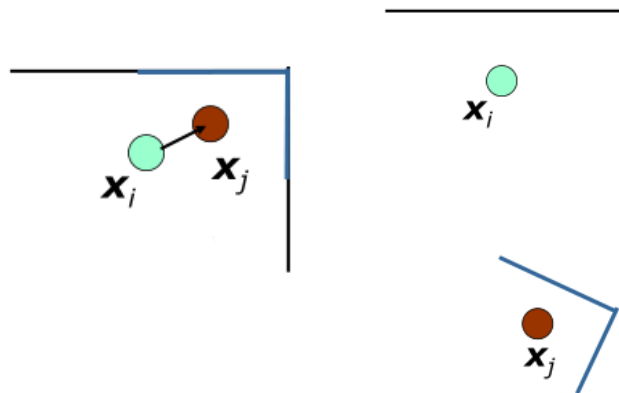


图的构建---回环边



Pose Graph示意图

- 通过回环检测得到:



- 节点 i 和节点 j 在空间上相邻(观测到同样的数据), 但是时间上不相邻
- 用帧间匹配算法计算一个相对位姿



Graph-based SLAM



1. Pose Graph的构建



2. 回环检测方法



3. 非线性最小二乘原理



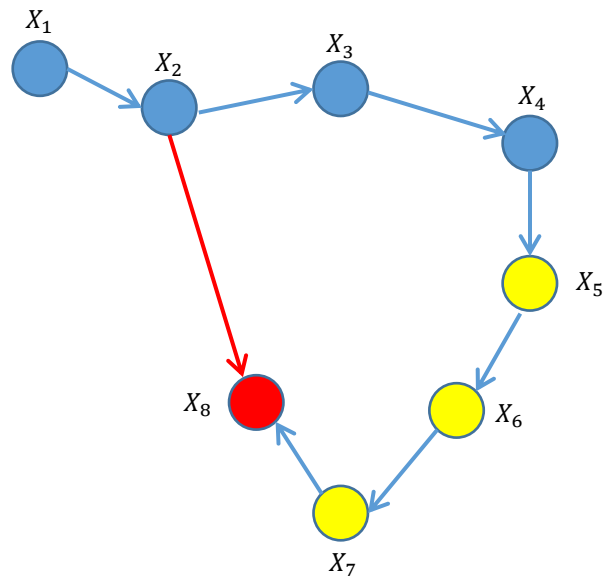
4. 非线性最小二乘在SLAM中的应用



回环检测



一个简单的回环检测方法



回环检测示意图

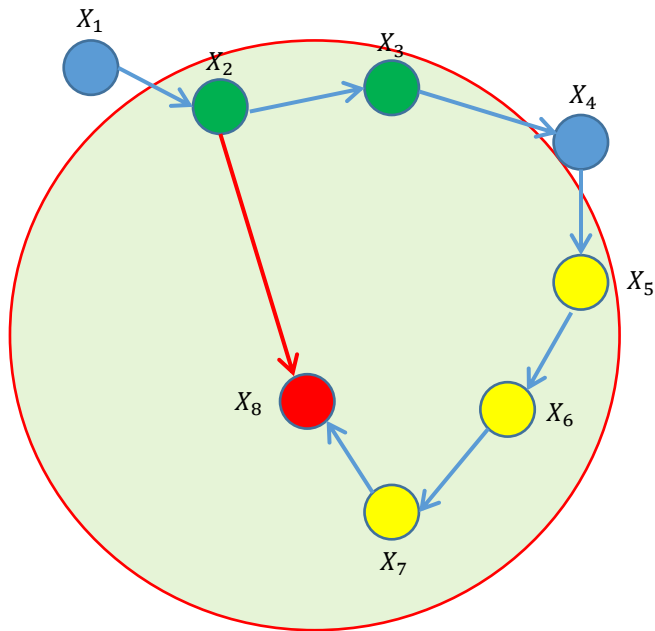
- 把节点分为active 和 inactive两部分
- 跟当前节点(红色节点)时间相近的节点称为active node(黄色节点), 其他的称为inactive node
- 找到当前节点周围一定范围内所有inactive节点, 作为回环候选帧
- 当前节点和回环候选帧进行匹配, 根据得分判断是否形成回环



回环检测



一个简单的回环检测方法



回环检测示意图

- 把节点分为active 和 inactive两部分
- 跟当前节点(红色节点)时间相近的节点称为active node(黄色节点), 其他的称为inactive node
- 找到当前节点周围一定范围内所有inactive节点, 作为回环候选帧(绿色节点)
- 当前节点和回环候选帧进行匹配, 根据得分判断是否形成回环



Graph-based SLAM



1. Pose Graph的构建



2. 回环检测方法



3. 非线性最小二乘原理



4. 非线性最小二乘在SLAM中的应用



非线性最小二乘(Non-Linear Least Square)



解决的问题

给定一个系统，其状态方程为： $f(x) = z$

x 表示系统的状态向量—即需要估计的值；

z 表示系统的观测值，可以通过传感器进行直接观测；

$f(x)$ 表示一个非线性的映射函数，状态向量 x 可以通过非线性函数 $f(x)$ 映射得到 z 。

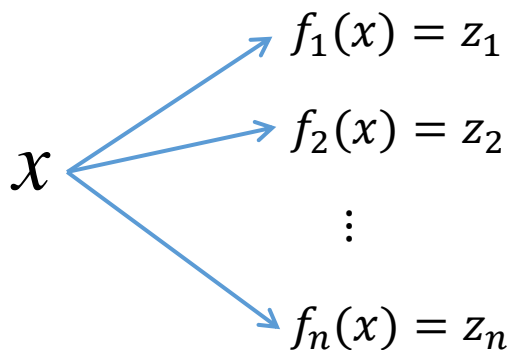
- 给定该系统的 n 个混有噪声的观测值 (z_1, \dots, z_n) ，估计状态向量 x ，使得其经过 $f(x)$ 映射之后的预测值和观测值的误差最小
- 和线性最小二乘基本相同，只是状态方程 $f(x)$ 是一个非线性函数



非线性最小二乘(Non-Linear Least Square)



示意图



z'_1

z'_2

z'_n

- x 表示机器人的位置
- $f(x)$ 为观测模型，节点之间相对位姿计算函数
- z 为帧间匹配或者回环检测计算出来的相对位姿
- 找到最优的 x ，让预测和观测的误差最小

状态向量

观测模型--预测值

测量值



非线性最小二乘(Non-Linear Least Square)



误差函数

目标为最小化预测和观测的差，因此误差即为预测和观测的差：

$$e_i(x) = f_i(x) - z'_i$$

假设误差服从高斯分布，即 $e_i(x) \sim N(\mathbf{0}, \Omega_i)$ ， Ω_i 为对应的信息矩阵。我们定义误差的联合概率分布为：

$$G(e_i(x)) = \prod_i \frac{1}{(2\pi)^{D/2} |\Omega_i|^{1/2}} \exp[-\frac{1}{2} e_i(x)^T \Omega_i e_i(x)]$$

最终目标是使得误差尽可能趋近于 $\mathbf{0}$ (均值)，等价于每个高斯分布取得最大值，因此误差的联合概率分布 $G(e_i(x))$ 取得最大值。

$$\ln G(e_i(x)) = \sum \frac{1}{(2\pi)^{D/2} |\Omega_i|^{1/2}} - \frac{1}{2} \sum e_i(x)^T \Omega_i e_i(x)$$

令非线性最小二乘的目标函数为：

$$\min F(x) = \min \sum e_i(x)^T \Omega_i e_i(x)$$



非线性最小二乘(Non-Linear Least Square)



解决的问题

目标函数：

$$\min_x F(x)$$

直接想法：求 $F(x)$ 关于变量 x 的导数，令其等于0，求解方程即可。

对于凸函数来说，上述想法是可行的，但对于非凸函数，通常采用基于梯度的优化方法。

- $F(x)$ 为关于 x 的非线性方程，能否把其化为关于 x 的线性方程。

线性化：泰勒展开



非线性最小二乘(Non-Linear Least Square)



线性化

$$F(x) = \sum e_i(x)^T \Omega_i e_i(x)$$

误差函数 $e_i(x)$ 是非线性函数，因此 $F(x)$ 是关于 x 的非线性函数。对误差函数 $e_i(x)$ 进行线性化得：

$$e_i(x + \Delta x) = e_i(x) + J_i(x)\Delta x$$

其中， J 为映射函数 $F(x)$ 对状态向量 x 的导数，称之为Jacobian矩阵。

$$J_i(x) = \left(\frac{\partial f_i(x)}{\partial x_1}, \frac{\partial f_i(x)}{\partial x_2}, \dots, \frac{\partial f_i(x)}{\partial x_n} \right)$$

因此，函数 $F(x)$ 的可化解为：

$$F(x + \Delta x) = \sum e_i^T(x + \Delta x) \Omega_i e_i(x + \Delta x)$$



非线性最小二乘(Non-Linear Least Square)



线性化

$$\begin{aligned} F(x + \Delta x) &= \sum e_i^T (x + \Delta x) \Omega_i e_i (x + \Delta x) \\ &= \sum (e_i(x) + J_i \Delta x)^T \Omega_i (e_i(x) + J_i \Delta x) \\ &= \sum e_i^T \Omega_i e_i + e_i^T \Omega_i J_i \Delta x + \Delta x^T J_i^T \Omega_i e_i + \Delta x^T J_i^T \Omega_i J_i \Delta x \\ &= \sum e_i^T \Omega_i e_i + 2 \underline{e_i^T \Omega_i J_i} \Delta x + \Delta x^T \underline{J_i^T \Omega_i J_i} \Delta x \\ &= \sum c_i + \sum (2b_i^T \Delta x + \Delta x^T H_i \Delta x) \end{aligned}$$

其中, $b_i^T = e_i^T \Omega_i J_i$

$$H_i = J_i^T \Omega_i J_i$$



非线性最小二乘(Non-Linear Least Square)



线性化

$$\begin{aligned} F(x + \Delta x) &= \sum c_i + \sum (2b_i^T \Delta x + \Delta x^T H_i \Delta x) \\ &= \sum c_i + \sum 2b_i^T \Delta x + \Delta x^T \sum H_i \Delta x \\ &= \sum c_i + 2b^T \Delta x + \Delta x^T H \Delta x \end{aligned}$$

$F(x + \Delta x)$ 为关于变量 Δx 的二次函数，令其关于 Δx 的导数等于0，可求解得到 $F(x + \Delta x)$ 的极值，即

$$\frac{\partial F(x + \Delta x)}{\partial \Delta x} = 2b + 2H\Delta x = 0$$

$$H\Delta x = -b$$

$$\Delta x^* = -H^{-1}b$$

- 令 $x = x + \Delta x^*$ ，然后不断迭代，直至收敛即可。



非线性最小二乘(Non-Linear Least Square)



流程

- 1.线性化误差函数:

$$e_i(x + \Delta x) = e_i(x) + J_i \Delta x$$

- 2.构建线性系统:

$$b^T = \sum e_i^T \Omega_i J_i \quad H = \sum J_i^T \Omega_i J_i \quad H \Delta x = b$$

- 3.求解线性系统:

$$\Delta x^* = -H^{-1}b$$

- 4.更新解, 并不断迭代直至收敛:

$$x = x + \Delta x^*$$



Graph-based SLAM



1. Pose Graph的构建



2. 回环检测方法



3. 非线性最小二乘原理



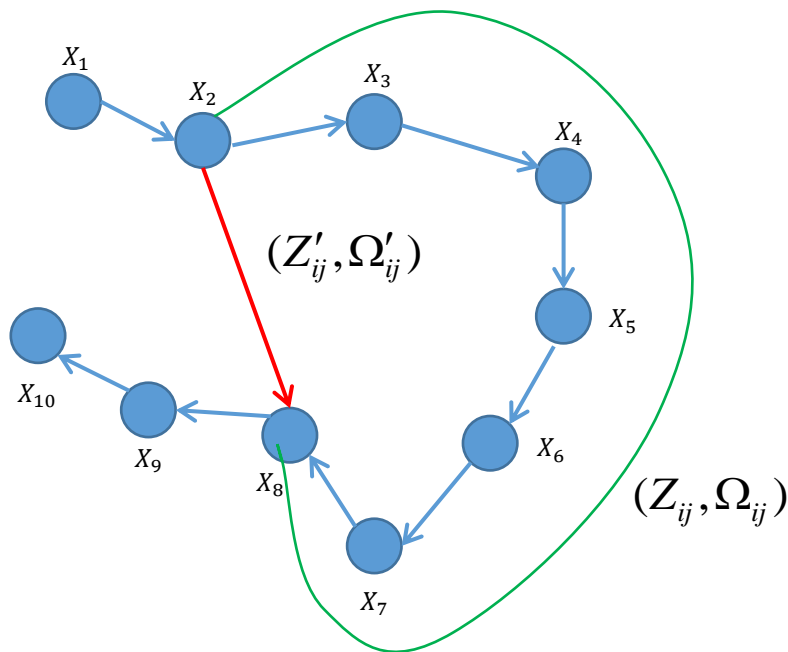
4. 非线性最小二乘在SLAM中的应用



NL Least Square在SLAM中的应用



误差函数



- 观测值为匹配计算得到的节点 i 和节点 j 的相对位姿：

$$z'_{ij} = (t_{ij}, \theta_{ij})$$
$$Z'_{ij} = V2T(z'_{ij})$$

- 预测值为里程积分得到的当前节点 i 和节点 j 的相对位姿：

$$Z_{ij} = f(x_i, x_j) = X_i^{-1} X_j$$
$$X_i = V2T(x_i)$$
$$X_j = V2T(x_j)$$

- 误差函数的定义：

$$e_{ij}(x) = T2V(Z'_{ij}{}^{-1} Z_{ij})$$



NL Least Square在SLAM中的应用



误差函数--预测值

- 已知:

$$X_i = \begin{bmatrix} R_i & t_i \\ 0 & 1 \end{bmatrix} \Rightarrow X_i^{-1} = \begin{bmatrix} R_i^T & -R_i^T t_i \\ 0 & 1 \end{bmatrix} \quad X_j = \begin{bmatrix} R_j & t_j \\ 0 & 1 \end{bmatrix}$$

- 预测值:

$$Z_{ij} = X_i^{-1} X_j = \begin{bmatrix} R_i^T & -R_i^T t_i \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_j & t_j \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_i^T R_j & R_i^T (t_j - t_i) \\ 0 & 1 \end{bmatrix}$$

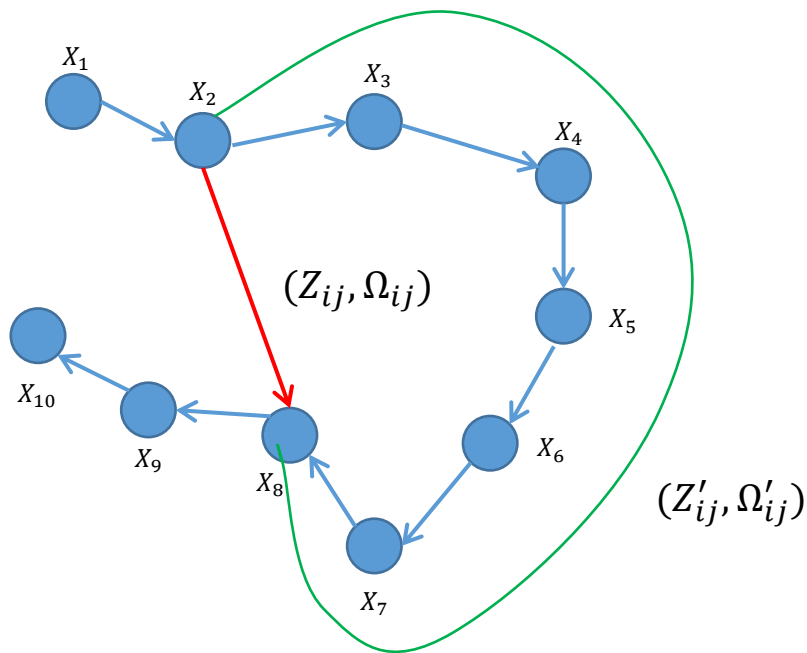
$$z_{ij} = T2V(Z_{ij}) = \begin{pmatrix} R_i^T (t_j - t_i) \\ \theta_j - \theta_i \end{pmatrix}$$



NL Least Square在SLAM中的应用



误差函数



- 误差函数的矩阵形式:

$$e_{ij}(x) = \begin{Bmatrix} R_{ij}^T (R_i^T (t_j - t_i) - t_{ij}) \\ \theta_j - \theta_i - \theta_{ij} \end{Bmatrix}$$

- 对应的Jacobian矩阵:

$$\frac{\partial e_{ij}(x)}{\partial x_i} = \begin{bmatrix} -R_{ij}^T R_i^T & R_{ij}^T \frac{\partial R_i^T}{\partial \theta} (t_j - t_i) \\ 0 & -1 \end{bmatrix}$$

$$\frac{\partial e_{ij}(x)}{\partial x_j} = \begin{bmatrix} R_{ij}^T R_i^T & 0 \\ 0 & 1 \end{bmatrix}$$



NL Least Square在SLAM中的应用



误差函数的线性化

- 误差函数：

$$e_{ij}(x + \Delta x) = e_{ij}(x) + J_{ij}\Delta x$$

$$J_{ij} = \frac{\partial e_{ij}(x)}{\partial x}$$

- 因为误差函数只跟 x_i 和 x_j 有关，因此具有下列的性质：

$$\begin{aligned} \frac{\partial e_{ij}(\mathbf{x})}{\partial \mathbf{x}} &= \left(0 \dots \frac{\partial e_{ij}(\mathbf{x}_i)}{\partial \mathbf{x}_i} \dots \frac{\partial e_{ij}(\mathbf{x}_j)}{\partial \mathbf{x}_j} \dots 0 \right) \\ \mathbf{J}_{ij} &= \left(0 \dots \mathbf{A}_{ij} \dots \mathbf{B}_{ij} \dots 0 \right) \end{aligned}$$



NL Least Square在SLAM中的应用



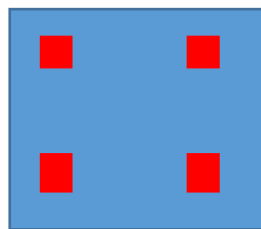
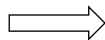
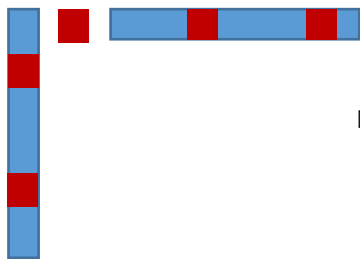
误差函数的线性化

- Jacobian矩阵的形式:

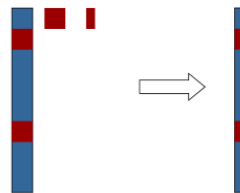
$$\mathbf{J}_{ij} = \begin{pmatrix} \mathbf{0} \dots \mathbf{0} & \underbrace{\frac{\partial \mathbf{e}(\mathbf{x}_i)}{\partial \mathbf{x}_i}}_{\mathbf{A}_{ij}} & \mathbf{0} \dots \mathbf{0} & \underbrace{\frac{\partial \mathbf{e}(\mathbf{x}_j)}{\partial \mathbf{x}_j}}_{\mathbf{B}_{ij}} & \mathbf{0} \dots \mathbf{0} \end{pmatrix}$$

- Jacobian是一个稀疏的向量，因此其会导致 H 矩阵的稀疏性:

$$H_{ij} = \mathbf{J}_{ij}^T \Omega_{ij} \mathbf{J}_{ij}$$



$$\mathbf{b}_{ij} = \mathbf{J}_{ij}^T \Omega_{ij} \mathbf{e}_{ij}$$





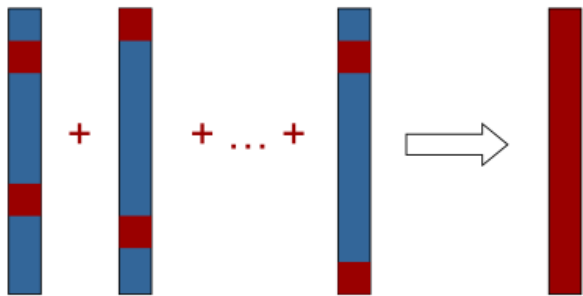
NL Least Square在SLAM中的应用



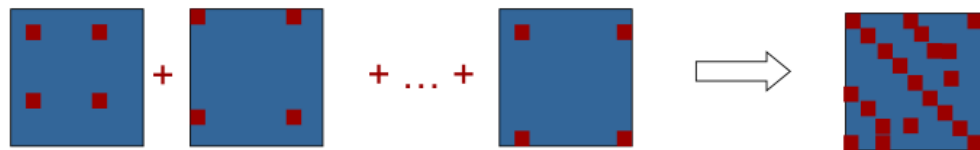
误差函数的线性化

- H 和 b 的最终形式:

$$b = \sum_{ij} b_{ij}$$



$$H = \sum_{ij} H_{ij}$$



- H 矩阵为稀疏矩阵, 可以利用此特征进行快速求解。



NL Least Square在SLAM中的应用



固定坐标系

- 观测值观测到的两个位姿之间的相对位姿
- 满足相对位姿约束的解有无穷多组
- 为了让解唯一，必须加入一个约束条件让某一个位姿固定，一般选择第一个位姿，即：

$$\Delta x_1 = 0$$

等价于：

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \Delta x_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$



NL Least Square在SLAM中的应用



固定坐标系

- 加入的约束为：

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \Delta x_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

- 求解的线性系统为：

$$H\Delta x = -b$$

- 因此等价于：

$$H_{11} += \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



NL Least Square在SLAM中的应用



构建线性系统

- 已知误差项和Jacobian矩阵 A_{ij} 和 B_{ij}
- 向量 b 的更新为:

$$b_i^T += e_{ij}^T \Omega_{ij} A_{ij}$$

$$b_j^T += e_{ij}^T \Omega_{ij} B_{ij}$$

- 矩阵 H 的更新为:

$$H_{ii} += A_{ij}^T \Omega_{ij} A_{ij}$$

$$H_{ij} += A_{ij}^T \Omega_{ij} B_{ij}$$

$$H_{ji} += B_{ij}^T \Omega_{ij} A_{ij}$$

$$H_{jj} += B_{ij}^T \Omega_{ij} B_{ij}$$



NL Least Square在SLAM中的应用



求解

- 已知矩阵 H 和向量 b

- 求解线性方程组：

$$\Delta x = -H^{-1}b$$

- 不断进行迭代，直至收敛：

$$x = x + \Delta x$$

optimize(x):

```
while (!converged)
    (H, b) = buildLinearSystem(x)
     $\Delta \mathbf{x}$  = solveSparse(H $\Delta \mathbf{x}$  = -b)
    x = x +  $\Delta \mathbf{x}$ 
end
return x
```



Cartographer代码讲解



参考资料

[1]G2O:A General Framework for Graph Optimization

[2]A tutorial graph-based slam

[3]Efficient Sparse pose adjustment for 2D mapping

[4] Real-Time Loop Closure in 2D LIDAR SLAM



作业



详细见作业说明



结语

感谢聆听！

Thanks for Listening