非线性最小二乘问题的求解：solver

2020年1月11日　　21:21

高斯牛顿求解

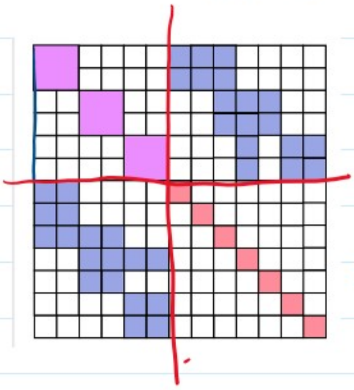
$$\underbrace{J^T\Sigma^{-1}J}_{H \text{ or } \Lambda}\,\delta\xi = \underbrace{-J^T\Sigma^{-1}}_{b} r$$

$\Rightarrow H\delta x = -b$

$\delta x = -H^{-1}b$　但是 通常H有7个自由度的不可观 → 不满秩 → 不可逆.

即使假设可逆 → 计算量 H^{-1} 过大.

解决方法 → 舒尔补. 利用H稀疏性求解.



⇒ 可以将其写为：

$$\begin{bmatrix} H_{pp} & H_{pl} \\ H_{lp} & H_{ll} \end{bmatrix}\begin{bmatrix} \Delta x_p^* \\ \Delta x_l^* \end{bmatrix} = \begin{bmatrix} -b_p \\ -b_l \end{bmatrix}$$

$$\xRightarrow{\text{舒尔补}} \begin{bmatrix} H_{pp}' & 0 \\ x & x \end{bmatrix}\begin{bmatrix} \Delta x_p^* \\ \Delta x_l^* \end{bmatrix} = \begin{bmatrix} -b_p' \\ -b_l' \end{bmatrix}$$

$\Rightarrow H_{pp}'\Delta x_p^* = -b_p'$

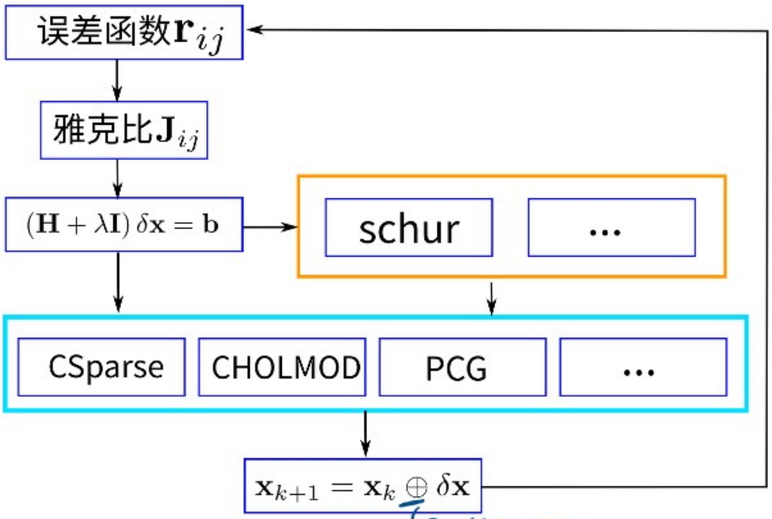
$\Rightarrow \Delta x_p^*$ ✓ yes.

从数学上来：$(H_{pp} - H_{pl}H_{ll}^{-1}H_{pl}^T)\Delta x_p^* = -b_p + H_{pl}H_{ll}^{-1}b_l$

出 Δx_p^*

$\underline{H_{ll}}\,\Delta x_l^* = -b_l - H_{pl}^T\Delta x_p^*$

整个流程可表达为：



广义的加法

$R = R\exp(w^\wedge)$

一些 Tipps: → 几个有用的建议

信息矩阵 H 不满秩，如何操作？

① 使用LM算法，加阻尼因子使得系统满秩，可求解，但是得到的结果可能会发生一定的变化

第一开始时间时阻尼因子不会太大

但这个轨迹整体会有一个漂移.

② 添加先验约束，增加系统的可观性.

比如假定第一个相机的Pose为固定；然后 { ① 固定 Landmark，确定尺度 ② 固定2个相机.

比如 g2o tutorial 中第一个pose的信息矩阵加上单位阵 $H[11] += I$，但[illegible].

$$\begin{bmatrix} H_{11}+I & & \\ & \ddots & \\ & & \ddots \end{bmatrix} x = b$$

去掉后

$$\Rightarrow I \cdot \delta x_{11} = 0$$

代码中如何实现 fix？

① 添加超强先验，使得对应的信息矩阵巨大. $\Rightarrow \delta x = H^{-1} b$.

② 设定对应雅克比矩阵为0，意味着残差等于0（$H = J^T \Sigma^{-1} J$. $b = J^T \Sigma^{-1}$）

$$\Rightarrow \begin{bmatrix} 0 & \cdots \\ & H_{22} & \cdots \\ & & \Sigma_{jj} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ b_2 \\ b_3 \end{bmatrix}$$

$+\lambda I$

这后再加上阻尼因子之后:

左右同加上零化之后:

$(\lambda I).G\delta x_i = 0.$

g2o:

顶点: vertex: (landmark, 相机 pose)

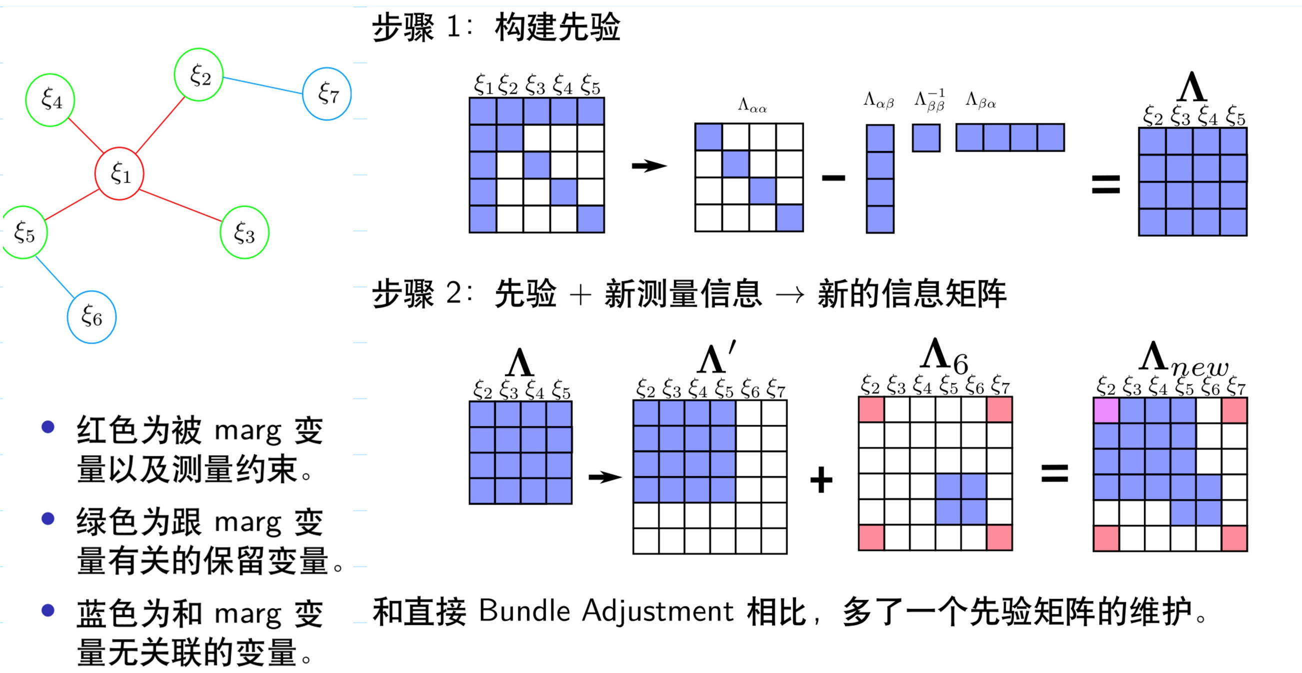
id. 维度. 矩阵. index

定义 变量加法.

边: edge.

残差计算. 雅克比矩阵计算.

求解器: solver.



如何更新先验残差?

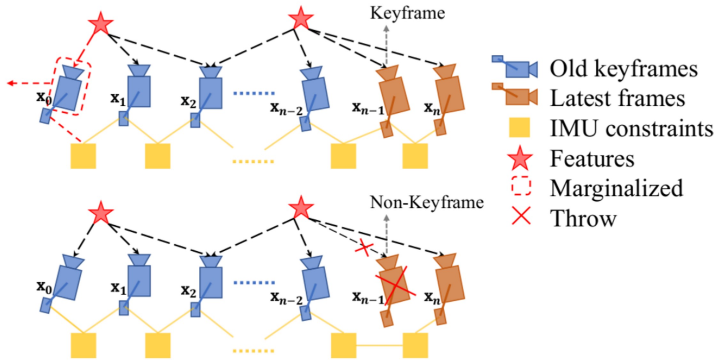
→ 先验信息矩阵固定不变，但随着迭代的推进，变量被不断优化，先验残差要跟随优化.

→ $b_p' = b_p + \frac{\partial b_p}{\partial x_p} \cdot \delta x_p$

$= b_p + \frac{\partial (J^T \Sigma^{-1} r)}{\partial x_p} \delta x_p$

近似 J: $= b_p + J^T \Sigma^{-1} J \, \delta x_p = b_p + \Lambda_p \delta x_p$

VINS里的滑动窗口策略:



。当滑动窗口中第二新的图像帧为关键帧，则 marg 最老的帧，以及上面的路标 点。

• 当滑动窗口中第二新的图像帧不是关键帧，则丢弃这一帧上的视觉测量信息， IMU 预积分传给下一帧.