

[\[关闭\]](#)

@Xiaobuyi 2018-03-22 15:09 字数 26139 阅读 3210

VINS-Mono代码分析总结

SLAM

- - [VINS-Mono代码分析总结](#)
 - [参考文献](#)
 - [前言](#)
 - [1 预积分的推导](#)
 - [1.1 离散状态下预积分方程](#)
 - [1.2 离散状态下误差状态方程](#)
 - [2 前端KLT跟踪](#)
 - [3 系统初始化](#)
 - [3.1 相机与IMU之间的相对旋转](#)
 - [3.2 相机初始化](#)
 - [3.3 视觉与IMU对齐](#)
 - [3.3.1 陀螺仪Bias修正](#)
 - [3.3.2 初始化速度、重力向量 \$g\$ 和尺度因子](#)
 - [3.3.3 纠正重力向量](#)
 - [4 后端优化](#)
 - [4.1 IMU测量误差](#)
 - [4.2 相机测量误差](#)
 - [4.3 闭环修正与优化](#)
 - [4.4 系统边缘化](#)
 - [4.4.1 边缘化的定义和目的](#)
 - [4.4.2 两种边缘化措施](#)
 - [4.4.3 舒尔补边缘化优化状态量](#)
 - [4.4.4 关于优化一致性\(First Estimate Jacobin, FEJ算法\)](#)
 - [4.4.5 滑窗优化\(Slide window\)](#)
 - [4.5 系统关键帧选择](#)
 - [5 闭环校正](#)
 - [5.1 闭环检测](#)
 - [5.2 闭环校正](#)
 - [6 全局的位姿优化](#)
 - [7 关键变量含义和代码注释中的一些称呼](#)
 - [7.1 代码注释中的自定义名称](#)
 - [7.2 代码中的关键变量](#)
 - [8 遗留的问题](#)
 - [2.当视差角过小的时候，视觉测量的协方差会很大。](#)

参考文献

- [1] [VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator](#), Tong Qin, Peiliang Li, Zhenfei Yang, Shaojie Shen (technical report)
- [2] [Solà J. Quaternion kinematics for the error-state KF\[M\]// Surface and Interface Analysis. 2015.](#)
- [3] [Solà J, Yang Z, Shen S. Monocular Visual-Inertial State Estimation With Online Initialization and Camera-IMU Extrinsic Calibration\[J\]. IEEE Transactions on Automation Science & Engineering, 2017, 14\(1\):39-51.](#)
- [4] [Engel J, Schöps T, Cremers D. LSD-SLAM: Large-scale direct monocular SLAM\[C\]//European Conference on Computer Vision. Springer, Cham, 2014:834-849.](#)
- [5] [Shen S, Michael N, Kumar V. Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft MAVs\[C\]// IEEE International Conference on Robotics and Automation. IEEE, 2015:5303-5310.](#)
- [6] [Shen S, Mulgaonkar Y, Michael N, et al. Initialization-Free Monocular Visual-Inertial State Estimation with Application to Autonomous MAVs\[M\]// Experimental Robotics. Springer International Publishing, 2016.](#)
- [7] [Sibley G. A Sliding Window Filter for SLAM\[J\]. University of Southern California, 2006.](#)
- [8] [CamOdoCal: https://github.com/hengli/camodocal](#)
- [9] [Tightly-coupled Visual-Inertial Sensor Fusion based on IMU Pre-Integration](#)

前言

视觉与IMU融合的分类：

- 松耦合和紧耦合：按照是否把图像的Feature加入到状态向量区分，换句话说就是松耦合是在视觉和IMU各自求出的位姿的基础上做的耦合。
- 滤波法和优化法：

[参考](#)

Vins-Mono是视觉与IMU的融合中的经典之作，其定位精度可以媲美OKVIS，而且具有比OKVIS更加完善和鲁棒的初始化以及闭环检测过程。同时VINS-Mono也为该领域树立了一个信标吧，视觉SLAM的研究和应用会更新偏向于**单目+IMU**。因为在机器人的导航中，尤其是无人机的自主导航中，单目不具有RGBD相机(易受光照影响、获取的深度信息有限)以及双目相机(占用较大的空间。。。)的限制，能够适应室内、室外及不同光照的环境，具有较好的适应性。而且在增强和虚拟现实

中，更多的移动设备仅具有单个相机，所以单目+IMU也更符合实际情况。

那么为什么要进行视觉与IMU的融合呢，自己总结的主要有以下几点：

- 视觉与IMU的融合可以借助IMU较高的采样频率，进而提高系统的输出频率。
- 视觉与IMU的融合可以提高视觉的鲁棒性，如视觉SLAM因为某些运动或场景出现的错误结果。
- 视觉与IMU的融合可以有效的消除IMU的积分漂移。
- 视觉与IMU的融合能够校正IMU的Bias。
- 单目与IMU的融合可以有效解决单目尺度不可观测的问题。

上面总结了视觉与IMU融合的几个优点，以及与单目融合可以解决单目相机尺度不可测的问题。但是单目相机的尺度不可测也是具有优点的，参考[\[4\]](#)。单目尺度不确定的优点主要有两方面：单目尺度的不确定性，可以对不同的规模大小的环境空间之间进行游走切换，起到无缝连接的作用，比如从室内桌面上的环境和大规模的户外场景；诸如深度或立体视觉相机，这些具备深度信息的传感器，它们所提供的可靠深度信息范围，是有限制的，故而不像单目相机那样具有尺度灵活性的特点。

1 预积分的推导

1.1 离散状态下预积分方程

关于这部分的论文和代码中的推导，可以参考文献[\[2\]](#)中Appendx部分“A Runge-Kutta numerical integration methods”中的欧拉法和中值法。

$$w'_k = \frac{w_{k+1} + w_k}{2} - b_w \quad (1.1)$$

$$q_{i+1} = q_i \otimes \begin{bmatrix} 1 \\ 0.5w'_k \end{bmatrix} \quad (1.2)$$

$$a'_k = \frac{q_k(a_k + n_{a0} - b_{a_k}) + q_{k+1}(a_{k+1} + n_{a1} - b_{a_k})}{2} \quad (1.3)$$

$$\alpha_{i+1} = \delta\alpha_i + \beta_i t + 0.5a'_k \delta t^2 \quad (1.4)$$

$$\beta_{i+1} = \delta\beta_i + a'_k \delta t \quad (1.5)$$

1.2 离散状态下误差状态方程

论文中II.B部分的误差状态方程是连续时间域内，在实际代码中需要的是离散时间下的方程式，而且在前面的预积分方程中使用了中值法积分方式。所以在实际代码中和论文是不一致的。在推导误差状态方程式的最重要的部分是对 $\delta\theta_{k+1}$ 部分的推导。

由泰勒公式可得：

$$\delta\theta_{k+1} = \delta\theta_k + \delta\dot{\theta}_k \delta t \quad (1.6)$$

依据参考文献[2]中 "5.3.3 The error-state kinematics" 中公式(222c)及其推导过程有：

$$\delta\dot{\theta}_k = -[w_m - w_b]_{\times} \delta\theta_k - \delta w_b - w_n$$

对于中值法积分下的误差状态方程为：

$$\delta\dot{\theta}_k = -[\frac{w_{k+1} + w_k}{2} - b_{g_k}]_{\times} \delta\theta_k - \delta b_{g_k} + \frac{n_{w0} + n_{w1}}{2} \quad (1.7)$$

将式(1.7)带入式(1.6)可得：

$$\delta\theta_{k+1} = (I - [\frac{w_{k+1} + w_k}{2} - b_{g_k}]_{\times} \delta t) \delta\theta_k - \delta b_{g_k} \delta t + \frac{n_{w0} + n_{w1}}{2} \delta t \quad (1.8)$$

这部分也可以参考，文献[2]中 "7.2 System kinematics in discrete time" 小节。

接下来先推导 $\delta\beta_{k+1}$ 部分，再推导 $\delta\alpha_{k+1}$ 部分。 $\delta\beta_{k+1}$ 部分的推导也可以参考文献[2]中 "5.3.3 The error-state kinematics" 公式(222b)的推导。将式(1.5)展开得到：

$$\delta\beta_{i+1} = \delta\beta_i + \frac{q_k(a_k + n_{a0} - b_{a_k}) + q_{k+1}(a_{k+1} + n_{a1} - b_{a_k})}{2} \delta t$$

即，

$$\delta\beta_{i+1} = \delta\beta_i + \delta\dot{\beta}_i \delta t \quad (1.9)$$

文献[2]中，公式(222b)

$$\dot{\delta v} = -R[a_m - a_b]_{\times} \delta\theta - R\delta a_b + \delta g - Ra_n$$

对于中值法积分下的误差状态方程为：

$$\begin{aligned} \delta\dot{\beta}_i = & -\frac{1}{2} q_k[a_k - b_{a_k}]_{\times} \delta\theta - \frac{1}{2} q_{k+1}[a_{k+1} - b_{a_k}]_{\times} \delta\theta_{k+1} - \delta b_{g_k} \delta t + \frac{n_{w0} + n_{w1}}{2} \delta t) \delta\theta \\ & -\frac{1}{2} q_k \delta b_{a_k} - \frac{1}{2} q_{k+1} \delta b_{a_k} - \frac{1}{2} q_k n_{a0} - \frac{1}{2} q_k n_{a1} \end{aligned} \quad (1.10)$$

将式(1.8)带入式(1.10)可得

$$\begin{aligned} \delta\dot{\beta}_i = & -\frac{1}{2} q_k[a_k - b_{a_k}]_{\times} \delta\theta - \frac{1}{2} q_{k+1}[a_{k+1} - b_{a_k}]_{\times} ((I - [\frac{w_{k+1} + w_k}{2} - b_{g_k}]_{\times} \delta t) \delta\theta_k - \delta b_{g_k} \delta t + \frac{n_{w0} + n_{w1}}{2} \delta t) \\ & -\frac{1}{2} q_k \delta b_{a_k} - \frac{1}{2} q_{k+1} \delta b_{a_k} - \frac{1}{2} q_k n_{a0} - \frac{1}{2} q_k n_{a1} \end{aligned} \quad (1.11)$$

同理，可以计算出 $\delta\alpha_{k+1}$ ，可以写为：

$$\delta\alpha_{i+1} = \delta\alpha_i + \delta\dot{\alpha}_i \delta t \quad (1.12)$$

$$\begin{aligned} \delta\dot{\alpha}_i = & -\frac{1}{4} q_k[a_k - b_{a_k}]_{\times} \delta\theta \delta t - \frac{1}{4} q_{k+1}[a_{k+1} - b_{a_k}]_{\times} ((I - [\frac{w_{k+1} + w_k}{2} - b_{g_k}]_{\times} \delta t) \delta\theta_k - \delta b_{g_k} \delta t + \frac{n_{w0} + n_{w1}}{2} \delta t) \delta t \\ & -\frac{1}{4} q_k \delta b_{a_k} \delta t - \frac{1}{4} q_{k+1} \delta b_{a_k} \delta t - \frac{1}{4} q_k n_{a0} \delta t - \frac{1}{4} q_k n_{a1} \delta t \end{aligned} \quad (1.13)$$

最后是加速度计和陀螺仪bias的误差状态方程，

$$\delta b_{a_{k+1}} = \delta b_{a_k} + n_{ba} \delta t \quad (1.14)$$

$$\delta b_{w_{k+1}} = \delta b_{w_k} + n_{bg} \delta t \quad (1.15)$$

综合式(1.8)等误差状态方程，将其写为矩阵形式，

$$\begin{bmatrix} \delta \alpha_{k+1} \\ \delta \theta_{k+1} \\ \delta \beta_{k+1} \\ \delta b_{a_{k+1}} \\ \delta b_{g_{k+1}} \end{bmatrix} = \begin{bmatrix} I & f_{01} & \delta t & -\frac{1}{4}(q_k + q_{k+1})\delta t^2 & f_{04} \\ 0 & I - [\frac{w_{k+1} + w_k}{2} - b_{w_k}]_{\times} \delta t & 0 & 0 & -\delta t \\ 0 & f_{21} & I & -\frac{1}{2}(q_k + q_{k+1})\delta t & f_{24} \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} \delta \alpha_k \\ \delta \theta_k \\ \delta \beta_k \\ \delta b_{a_k} \\ \delta b_{g_k} \end{bmatrix} + \begin{bmatrix} \frac{1}{4} q_k \delta t^2 & v_{01} & \frac{1}{4} q_{k+1} \delta t^2 & v_{03} & 0 & 0 \\ 0 & \frac{1}{2} \delta t & 0 & \frac{1}{2} \delta t & 0 & 0 \\ \frac{1}{2} q_k \delta t & v_{21} & \frac{1}{2} q_{k+1} \delta t & v_{23} & 0 & 0 \\ 0 & 0 & 0 & 0 & \delta t & 0 \\ 0 & 0 & 0 & 0 & 0 & \delta t \end{bmatrix} \begin{bmatrix} n_{a0} \\ n_{w0} \\ n_{a1} \\ n_{w1} \\ n_{ba} \\ n_{bg} \end{bmatrix} \quad (1.16)$$

其中，

$$\begin{aligned} f_{01} &= -\frac{1}{4} q_k [a_k - b_{a_k}]_{\times} \delta t^2 - \frac{1}{4} q_{k+1} [a_{k+1} - b_{a_k}]_{\times} (I - [\frac{w_{k+1} + w_k}{2} - b_{g_k}]_{\times} \delta t) \delta t^2 \\ f_{21} &= -\frac{1}{2} q_k [a_k - b_{a_k}]_{\times} \delta t - \frac{1}{2} q_{k+1} [a_{k+1} - b_{a_k}]_{\times} (I - [\frac{w_{k+1} + w_k}{2} - b_{g_k}]_{\times} \delta t) \delta t \\ f_{04} &= \frac{1}{4} (-q_{k+1} [a_{k+1} - b_{a_k}]_{\times} \delta t^2) (-\delta t) \\ f_{24} &= \frac{1}{2} (-q_{k+1} [a_{k+1} - b_{a_k}]_{\times} \delta t) (-\delta t) \\ v_{01} &= \frac{1}{4} (-q_{k+1} [a_{k+1} - b_{a_k}]_{\times} \delta t^2) \frac{1}{2} \delta t \\ v_{03} &= \frac{1}{4} (-q_{k+1} [a_{k+1} - b_{a_k}]_{\times} \delta t^2) \frac{1}{2} \delta t \\ v_{21} &= \frac{1}{2} (-q_{k+1} [a_{k+1} - b_{a_k}]_{\times} \delta t^2) \frac{1}{2} \delta t \\ v_{23} &= \frac{1}{2} (-q_{k+1} [a_{k+1} - b_{a_k}]_{\times} \delta t^2) \frac{1}{2} \delta t \end{aligned}$$

将式(1.16)简写为，

$$\delta z_{k+1} = F \delta z_k + V Q$$

最后得到系统的雅克比矩阵 J_{k+1} 和协方差矩阵 P_{k+1} ，初始状态下的雅克比矩阵和协方差矩阵为单位阵和零矩阵，即

$$\begin{aligned} J_k &= I \\ P_k &= 0 \end{aligned}$$

$$J_{k+1} = F J_k \quad (1.17)$$

$$P_{k+1} = FP_k F^T + VQV_T \quad (1.18)$$

2 前端KLT跟踪

3 系统初始化

在提取的图像的Features和做完IMU的预积分之后，进入了系统的初始化环节，那么系统为什么要进行初始化呢，主要的目的有以下两个：

- 系统使用单目相机，如果没有一个良好的尺度估计，就无法对两个传感器做进一步的融合。这个时候需要恢复出尺度；
- 要对IMU进行初始化，IMU会受到bias的影响，所以要得到IMU的bias。

所以我们要从初始化中恢复出尺度、重力、速度以及IMU的bias，因为视觉(SFM)在初始化的过程中有着较好的表现，所以在初始化的过程中主要以SFM为主，然后将IMU的预积分结果与其对齐，即可得到较好的初始化结果。

系统的初始化主要包括三个环节：求取相机与IMU之间的相对旋转、相机初始化(局部滑窗内的SFM，包括没有尺度的BA)、IMU与视觉的对齐(IMU预积分中的 α 等和相机的translation)。

3.1 相机与IMU之间的相对旋转

这个地方相当于求取相机与IMU的一部分外参。相机与IMU之间的旋转标定非常重要，偏差1-2°系统的精度就会变的极低。这部分的内容参考文献[3]中V-A部分，这里做简单的总结。

设相机利用对极关系得到的旋转矩阵为 $R_{c_{k+1}}^{c_k}$ ，IMU经过预积分得到的旋转矩阵为 $R_{b_{k+1}}^{b_k}$ ，相机与IMU之间的相对旋转为 R_c^b ，则对于任一帧满足，

$$R_{b_{k+1}}^{b_k} R_c^b = R_c^{c_k} R_{c_{k+1}}^{c_k} \quad (3.1)$$

对式(3.1)可以做简单的证明，在其两边同乘 ${}^c x_{k+1}$ 得

$$\begin{aligned} R_{b_{k+1}}^{b_k} R_c^{b_k} {}^b x_{k+1} &= R_c^{c_k} R_{c_{k+1}}^{c_k} {}^c x_{k+1} \\ R_{b_{k+1}}^{b_k} {}^b x_{k+1} &= R_c^{b_k} {}^b x_k \\ {}^b x_k &= {}^b x_k \end{aligned}$$

将旋转矩阵写为四元数，则式(3.1)可以写为

$$q_{b_{k+1}}^{b_k} \otimes q_c^b = q_c^{c_k} \otimes q_{c_{k+1}}^{c_k}$$

将其写为左乘和右乘的形式，综合为

$$[Q_1(q_{b_{k+1}}^{b_k}) - Q_2(q_{c_{k+1}}^{c_k})]q_c^b = Q_{k+1}^k q_c^b = 0 \quad (3.2)$$

其中 $Q_1(q_{b_{k+1}}^{b_k})$ ， $Q_2(q_{c_{k+1}}^{c_k})$ 分别表示四元数的左乘和右乘形式，

$$\begin{aligned} Q_1(q) &= \begin{bmatrix} q_w I_3 + [q_{xyz}]_{\times} & q_{xyz} \\ -q_{xyz} & q_w \end{bmatrix} \\ Q_2(q) &= \begin{bmatrix} q_w I_3 - [q_{xyz}]_{\times} & q_{xyz} \\ -q_{xyz} & q_w \end{bmatrix} \end{aligned} \quad (3.3)$$

这个地方因为四元数的实部在前在后的表达不一样，而左右乘的形式也不一样，所以文献[2]和文献[3]存在区别。Vins-Mono使用的是Eigen实部在后的表达方式。

那么对于 n 对测量值，则有

$$\begin{bmatrix} w_1^0 Q_1^0 \\ w_2^1 Q_2^1 \\ \vdots \\ w_N^{N-1} Q_N^{N-1} \end{bmatrix} q_c^b = Q_N q_c^b = 0 \quad (3.4)$$

其中 w_N^{N-1} 为外点剔除权重，其与相对旋转求取得残差有关， N 为最新的视觉帧的index，其由最终的终止条件决定。残差可以写为，

$$r_{k+1}^k = \text{acos}((\text{tr}(\hat{R}_c^{b^{-1}} R_{b_{k+1}}^{b^{-1}} \hat{R}_c^b R_{c_{k+1}}^{c_k}) - 1)/2) \quad (3.5)$$

残差还是很好理解的，在具体的代码中可以计算公式(3.1)两边两个旋转的得角度差。在得到残差之后就可以进一步得到公式(3.4)中的权重，

$$w_{k+1}^k = \begin{cases} 1, & r_{k+1}^k < \text{threshold} \\ \frac{\text{threshold}}{r_{k+1}^k}, & \text{otherwise} \end{cases} \quad (3.6)$$

一般会将阈值 **threshold** 取做 5° 。至此，就可以通过求解方程(3.4)得到相对旋转，式(3.4)的解为 Q_N 的左奇异向量中最小奇异值对应的特征向量。

但是，在这里还要注意 **求解的终止条件(校准完成的终止条件)**。在足够多的旋转运动中，我们可以很好的估计出相对旋转 R_c^b ，这时 Q_N 对应一个准确解，且其零空间的秩为1。但是在校准的过程中，某些轴向上可能存在退化运动(如匀速运动)，这时 Q_N 的零空间的秩会大于1。判断条件就是 Q_N 的第二小的奇异值是否大于某个阈值，若大于则其零空间的秩为1，反之秩大于1，相对旋转 R_c^b 的精度不够，校准不成功。

3.2 相机初始化

这一阶段的思路就是单目相机的初始化过程，先求取本质矩阵求解位姿，进而三角化特征点，然后PnP求解位姿，不断重复的过程，直到恢复出滑窗内的Features和相机位姿，代码比较清晰。要注意的就是坐标系的和位姿的变换，容易混乱。如以下几个函数：

triangulateTwoFrames：输入是相机外参(世界到相机)，求解出的3D点是在世界坐标系下。

cv::solvePnP：该API输入是世界坐标系的点，求解出的是世界坐标系到相机坐标系的变换，所以一般需要将结果转置。

3.3 视觉与IMU对齐

视觉与IMU的对齐主要解决三个问题：

- (1) 修正陀螺仪的bias；
- (2) 初始化速度、重力向量 g 和尺度因子(Metric scale)；
- (3) 改进重力向量 g 的量值；

3.3.1 陀螺仪Bias修正

发现校正部分使用的都是一系列的约束条件，思路很重要啊。陀螺仪Bias校正的时候也是使用了一个简单的约束条件：

$$\min_{\delta b_w} \sum_{k \in B} \|q_{b_{k+1}}^{c_0^{-1}} \otimes q_{b_k}^{c_0} \otimes \gamma_{b_{k+1}}^{b_k}\|^2 \quad (3.7)$$

其中

$$\gamma_{b_{k+1}}^{b_k} \approx \hat{\gamma}_{b_{k+1}}^{b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} J_{b_w}^\gamma \delta b_w \end{bmatrix} \quad (3.8)$$

公式(3.7)的最小值为单位四元数 $[1, 0_v]^T$ ，所以将式(3.7)进一步写为，

$$\begin{aligned}
q_{b_{k+1}}^{c_0^{-1}} \otimes q_{b_k}^{c_0} \otimes \gamma_{b_{k+1}}^{b_k} &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
\hat{\gamma}_{b_{k+1}}^{b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} J_{b_w}^\gamma \delta b_w \end{bmatrix} &= q_{b_k}^{c_0^{-1}} \otimes q_{b_{k+1}}^{c_0} \\
\begin{bmatrix} 1 \\ \frac{1}{2} J_{b_w}^\gamma \delta b_w \end{bmatrix} &= \hat{\gamma}_{b_{k+1}}^{b_k^{-1}} \otimes q_{b_k}^{c_0^{-1}} \otimes q_{b_{k+1}}^{c_0}
\end{aligned} \tag{3.9}$$

只取式(3.9)式虚部，在进行最小二乘求解

$$J_{b_w}^{\gamma^T} J_{b_w}^\gamma \delta b_w = 2 * J_{b_w}^{\gamma^T} (\hat{\gamma}_{b_{k+1}}^{b_k^{-1}} \otimes q_{b_k}^{c_0^{-1}} \otimes q_{b_{k+1}}^{c_0}). \text{vec} \tag{3.10}$$

求解式(3.10)的最小二乘解，即可得到 δb_w ，注意这个地方得到的只是Bias的变化量，需要在滑窗内累加得到Bias的准确值。

3.3.2 初始化速度、重力向量 g 和尺度因子

在这个步骤中，要估计系统的速度、重力向量以及尺度因子。所以系统的状态量可以写为，

$$X_I = [v_{b_0}^{c_0}, v_{b_1}^{c_0}, \dots, g^{c_0}, s] \tag{3.11}$$

上面的状态量都是在 c_0 相机坐标系下。接着，有前面的由预积分部分，IMU的测量模型可知

$$\begin{aligned}
\alpha_{b_{k+1}}^{b_k} &= q_{c_0}^{b_k} (s(\bar{p}_{b_{k+1}}^{c_0} - \bar{p}_{b_k}^{c_0}) + \frac{1}{2} g^{c_0} \Delta t_k^2 - v_{b_k}^{c_0} \Delta t_k^2) \\
\beta_{b_{k+1}}^{b_k} &= q_{c_0}^{b_k} (v_{b_{k+1}}^{c_0} + g^{c_0} \Delta t_k - v_{b_k}^{c_0})
\end{aligned} \tag{3.12}$$

在3.1小节，我们已经得到了IMU相对于相机的旋转 q_b^c ，假设IMU到相机的平移量 p_b^c 那么可以很容易地将相机坐标系下的位姿转换到IMU坐标系下，

$$\begin{aligned}
q_{b_k}^{c_0} &= q_{c_k}^{c_0} \otimes q_b^c \\
s\bar{p}_{b_k}^{c_0} &= s\bar{p}_{c_k}^{c_0} + q_{c_k}^{c_0} p_b^c
\end{aligned} \tag{3.13}$$

综合式(3.12)和式(3.13)可得，

$$\begin{aligned}
\hat{z}_{b_{k+1}}^{b_k} &= \begin{bmatrix} \alpha_{b_{k+1}}^{b_k} - q_{c_{k+1}}^{c_0} p_b^c + q_{c_k}^{c_0} p_b^c \\ \beta_{b_{k+1}}^{b_k} \end{bmatrix} = H_{b_{k+1}}^{b_k} X_I + n_{b_{k+1}}^{b_k} \\
&\approx \begin{bmatrix} -q_{c_0}^{b_k} \Delta t_k & 0 & 1/2 q_{c_0}^{b_k} \Delta t_k^2 & q_{c_0}^{b_k} (\bar{p}_{b_{k+1}}^{c_0} - \bar{p}_{b_k}^{c_0}) \\ -q_{c_0}^{b_k} & q_{c_0}^{b_k} & q_{c_0}^{b_k} \Delta t_k & 0 \end{bmatrix} \begin{bmatrix} v_{b_k}^{c_0} \\ v_{b_{k+1}}^{c_0} \\ g^{c_0} \\ s \end{bmatrix}
\end{aligned} \tag{3.14}$$

这一步的推导也比较简单，以 H 矩阵的第一行，将式(3.13)的(2)式带入(3.12)的(1)式可得

$$\begin{aligned}
\alpha_{b_{k+1}}^{b_k} &= q_{c_0}^{b_k} (s(\bar{p}_{c_{k+1}}^{c_0} - \bar{p}_{c_k}^{c_0} + (q_{c_{k+1}}^{c_0} - q_{c_k}^{c_0}) p_b^c + \frac{1}{2} g^{c_0} \Delta t_k^2 - v_{b_k}^{c_0} \Delta t_k^2) \\
\alpha_{b_{k+1}}^{b_k} - q_{c_0}^{b_k} (q_{c_{k+1}}^{c_0} - q_{c_k}^{c_0}) p_b^c &= q_{c_0}^{b_k} (s(\bar{p}_{c_{k+1}}^{c_0} - \bar{p}_{c_k}^{c_0} p_b^c) + \frac{1}{2} g^{c_0} \Delta t_k^2 - v_{b_k}^{c_0} \Delta t_k^2)
\end{aligned}$$

但是不知道为什么论文，在做了近似之后漏了上式等式左边的 $q_{c_0}^{b_k}$ 。

最后求解最小二乘问题

$$\min_{\delta b_w} \sum_{k \in B} \|\hat{z}_{b_{k+1}}^{b_k} - H_{b_{k+1}}^{b_k} X_I\|^2$$

至此即可求解出所有状态量，但是对于重力向量 g^{c_0} 还要做进一步的纠正。在纠正 g^{c_0} 的过程中，会对速度也做进一步的优化。

3.3.3 纠正重力向量

这部分和上一小节的内容差不多，就是迭代不断更新重力向量的过程，要注意重力向量被表达为：

$$\hat{g} = g \cdot \hat{g} + \omega_1 b_1 + \omega_2 b_2$$

4 后端优化

后端优化是VINS-Mono中除了初始化之外，创新性最高的一块，也是真真的 **紧耦合** 部分，而初始化的过程事实上是一个 **松耦合**。因为初始化过程中的状态量并没有放在最底层融合，而是各自做了位姿的计算，但是在后端优化的过程中，所有优化量都是在一起的。

状态量

$$\begin{aligned} X &= [x_0, x_1, \dots, x_n, x_c^b, \lambda_0, \lambda_1, \dots, \lambda_m] \\ x_k &= [p_{b_k}^w, v_{b_k}^w, q_{b_k}^w, b_a, b_g], \quad k \in [0, n] \\ x_c^b &= [p_c^b, q_c^b] \end{aligned} \quad (4.1)$$

优化过程中的误差状态量

$$\begin{aligned} \delta X &= [\delta x_0, \delta x_1, \dots, \delta x_n, \delta x_c^b, \delta \lambda_0, \delta \lambda_1, \dots, \delta \lambda_m] \\ \delta x_k &= [\delta p_{b_k}^w, \delta v_{b_k}^w, \delta \theta_{b_k}^w, \delta b_a, \delta b_g], \quad k \in [0, n] \\ \delta x_c^b &= [\delta p_c^b, \delta q_c^b] \end{aligned}$$

进而得到系统优化的代价函数

$$\begin{aligned} \min_X \left\{ \|r_p - H_p X\|^2 + \sum_{k \in B} \|r_B(\hat{z}_{b_{k+1}}^{b_k}, X)\|_{P_{b_{k+1}}^{b_k}}^2 \right. \\ \left. + \sum_{(i,j) \in C} \|r_C(\hat{z}_i^{c_j}, X)\|_{P_i^{c_j}}^2 \right\} \end{aligned} \quad (4.2)$$

其中三个残差项依次是边缘化的先验信息、IMU测量残差以及视觉的观测残差。三种残差都是用马氏距离来表示的，这个在后面ceres的优化残差中要特别注意。

在优化过程中，每一次的高斯迭代，式(4.2)可以进一步被线性化为，

$$\begin{aligned} \left\{ \min_X \|r_p - H_p X\|^2 + \sum_{k \in B} \|r_B(\hat{z}_{b_{k+1}}^{b_k}, X) + H_{b_{k+1}}^{b_k} \delta X\|_{P_{b_{k+1}}^{b_k}}^2 \right. \\ \left. + \sum_{(i,j) \in C} \|r_C(\hat{z}_i^{c_j}, X) + H_i^{c_j} \delta X\|_{P_i^{c_j}}^2 \right\} \end{aligned} \quad (4.3)$$

其中 $H_{b_{k+1}}^{b_k}, H_i^{c_j}$ 为IMU测量和视觉测量方程的雅克比矩阵，在后面会有进一步的推导。

进一步将式(4.3)写作

$$\begin{aligned}
& (\Lambda_p + \sum H_{b_{k+1}}^{b_k^T} P^{-1} H_{b_{k+1}}^{b_k} + \sum H_l^{c_j^T} R^{-1} H_l^{c_j}) \delta x \\
& = (b_p + \sum H_{b_{k+1}}^{b_k^T} P^{-1} r_B + \sum H_l^{c_j^T} R^{-1} r_C)
\end{aligned}$$

最小化式(4.3)相当于求解线性方程

$$(\Lambda_p + \Lambda_B + \Lambda_C) \delta X = (b_p + b_B + b_C) \quad (4.4)$$

其中 $\Lambda_p, \Lambda_B, \Lambda_C$ 分别对应边缘先验、IMU和视觉测量的信息矩阵。这部分还需要进一步的理解。

4.1 IMU测量误差

这部分内容主要对应在后端的优化过程中IMU测量部分的残差以及在优化过程中的雅克比矩阵的求解。

首先推导IMU测量的残差部分，由文献[1]中body系下的预积分方程式(4)，可以得到IMU的测量模型式(13)

$$\begin{bmatrix} \hat{\alpha}_{b_{k+1}}^{b_k} \\ \hat{\gamma}_{b_{k+1}}^{b_k} \\ \hat{\beta}_{b_{k+1}}^{b_k} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} q_w^{b_k} (p_{b_{k+1}}^w - p_{b_k}^w + \frac{1}{2} g^w \Delta t^2 - v_{b_k}^w \Delta t) \\ p_{b_k}^{w^{-1}} \otimes q_{b_{k+1}}^w \\ q_w^{b_k} (v_{b_{k+1}}^w + g^w \Delta t - v_{b_k}^w) \\ b_{ab_{k+1}} - b_{ab_k} \\ b_{wb_{k+1}} - b_{wb_k} \end{bmatrix} \quad (4.5)$$

那么IMU测量的残差即可写为

$$r_B(\hat{z}_{b_{k+1}}^{b_k}, X) = \begin{bmatrix} \delta \alpha_{b_{k+1}}^{b_k} \\ \delta \theta_{b_{k+1}}^{b_k} \\ \delta \beta_{b_{k+1}}^{b_k} \\ \delta b_a \\ \delta b_g \end{bmatrix} = \begin{bmatrix} q_w^{b_k} (p_{b_{k+1}}^w - p_{b_k}^w + \frac{1}{2} g^w \Delta t^2 - v_{b_k}^w \Delta t) - \hat{\alpha}_{b_{k+1}}^{b_k} \\ 2[q_{b_{k+1}}^{w^{-1}} \otimes q_{b_k}^w \otimes \hat{\gamma}_{b_{k+1}}^{b_k}]_{xyz} \\ q_w^{b_k} (v_{b_{k+1}}^w + g^w \Delta t - v_{b_k}^w) - \hat{\beta}_{b_{k+1}}^{b_k} \\ b_{ab_{k+1}} - b_{ab_k} \\ b_{gb_{k+1}} - b_{gb_k} \end{bmatrix} \quad (4.6)$$

其中 $[\hat{\alpha}_{b_{k+1}}^{b_k}, \hat{\gamma}_{b_{k+1}}^{b_k}, \hat{\beta}_{b_{k+1}}^{b_k}]$ 来自于IMU预积分部分。式(4.4)和文献[1]中的式(22)有些不同，主要是第1项，在两个式子中是逆的关系，本文中的写法是为了和代码保持一致。本质上两者是没有区别的，因为残差很小的时候，第一项接近于单位四元数，所以取逆并没有什么影响，只是写残差的方式不一样。

在式(4.4)中，残差主要来自于两帧IMU的位姿、速度及Bias，即 $[p^w, q^w, v^w, b_a, b_w]$ ，或者说我们要利用IMU残差要优化的状态量也是这5个，如果是两帧IMU就是10个。在计算雅克比矩阵的时候，分为 $[p_{b_k}^w, q_{b_k}^w]$ ， $[v_{b_k}^w, b_{ab_k}, b_{wb_k}]$ ， $[p_{b_{k+1}}^w, q_{b_{k+1}}^w]$ ， $[v_{b_{k+1}}^w, b_{ab_{k+1}}, b_{wb_{k+1}}]$ 等四部分，分别表示为 $J[0], J[1], J[2], J[3]$ 。这部分内容的推导可以参考文献[2]中III.B式(14)部分。

这个地方很容易出错误，原因就是没有搞清楚求偏导的对象。整个优化过程中，IMU测量模型这一块，涉及到的状态量是 x_k ，但是参考文献[5]中IV.A部分，以及十四讲中10.2.2小节的目标函数可知，这个地方的雅克比矩阵是针对变化量 δx_k 的。所以，在后面求取四部分雅可比的时候，也不是对状态量求偏导，而是对误差状态量求偏导。

式(4.4)对 $[\delta p_{b_k}^w, \delta \theta_{b_k}^w]$ 求偏导得，

$$J[0] = \begin{bmatrix} -q_w^{b_k} & R_w^{b_k} [(p_{b_{k+1}}^w - p_{b_k}^w + \frac{1}{2} g^w \Delta t^2 - v_{b_k}^w \Delta t)]_{\times} \\ 0 & [q_{b_{k+1}}^{w^{-1}} q_{b_k}^w]_L [\hat{\gamma}_{b_{k+1}}^{b_k}]_R \\ 0 & R_w^{b_k} [(v_{b_{k+1}}^w + g^w \Delta t - v_{b_k}^w)]_{\times} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (4.7)$$

$J[0]$ 是 $15 * 7$ 的矩阵，其中 $R_w^{b_k}$ 是四元数对应的旋转矩阵，求偏导可以看做是对四元数或者旋转矩阵求偏导，求解过程可以参考

文献[2]。第3项只需右下角的3行3列。第1项可以参考文献[1]中4.3.4 “旋转向量的雅克比矩阵” 推导得到，第3项是可以参考文献[1]中公式(22)和文献[9]中3.2部分。

式(4.4)对 $[\delta v_{b_k}^w, \delta b_{ab_k}, \delta b_{wb_k}]$ 求偏导得，

$$J[1] = \begin{bmatrix} -q_w^{b_k} \Delta t & -J_{b_a}^\alpha & -J_{b_a}^\alpha \\ 0 & 0 & -[q_{b_{k+1}}^{w^{-1}} \otimes q_{b_k}^w \otimes \hat{\gamma}_{b_{k+1}}^{b_k}]_L J_{b_w}^\gamma \\ -q_w^{b_k} & -J_{b_a}^\beta & -J_{b_a}^\beta \\ 0 & -I & 0 \\ 0 & 0 & -I \end{bmatrix} \quad (4.8)$$

$J[1]$ 是 $15 * 9$ 的矩阵，第5项仍然取右下角的3行3列。

式(4.4)对 $[\delta p_{b_{k+1}}^w, \delta \theta_{b_{k+1}}^w]$ 求偏导得，

$$J[2] = \begin{bmatrix} -q_w^{b_k} & 0 \\ 0 & [\hat{\gamma}_{b_{k+1}}^{b_k^{-1}} \otimes q_w^{b_k} \otimes q_{b_{k+1}}^w]_L \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (4.9)$$

$J[2]$ 是 $15 * 9$ 的矩阵，第5项仍然取右下角的3行3列。

式(4.4)对 $[\delta v_{b_{k+1}}^w, \delta b_{ab_{k+1}}, \delta b_{wb_{k+1}}]$ 求偏导得，

$$J[3] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ q_w^{b_k} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \quad (4.10)$$

$J[3]$ 是 $15 * 9$ 的矩阵。

以上便是IMU测量模型的雅可比的矩阵的推导过程，雅克比矩阵主要还是在Ceres做优化的过程中，高斯迭代要使用到。

4.2 相机测量误差

相机测量误差万变不离其宗，还是要会到像素坐标差或者灰度差(光度误差)。Vins-Mono中的相机测量误差本质还是特征点的重投影误差，将特征点 P 从相机的 i 系转到相机的 j 系，即把相机的测量残差定义为，

$$r_C = (\hat{z}_l^{c_j}, X) = [b_1, b_2]^T \cdot (\bar{P}_l^{c_j} - \frac{P_l^{c_j}}{\|P_l^{c_j}\|}) \quad (4.11)$$

因为最终要将残差投影到切平面上， $[b_1, b_2]$ 是切平面上的一对正交基。反投影后的 $P_l^{c_j}$ 写为，

$$P_l^{c_j} = q_b^c(q_w^{b_j}(q_{b_i}^w(q_c^b \frac{\bar{P}_l^{c_i}}{\lambda_l} + p_c^b) + p_{b_i}^w - p_{b_j}^w) - p_c^b)$$

反投影之前的坐标 $\bar{P}_l^{c_j}, \bar{P}_l^{c_i}$ 写为，

$$\begin{aligned} \bar{P}_l^{c_j} &= \pi_c^{-1} \left(\begin{bmatrix} \hat{u}_l^{c_j} \\ \hat{v}_l^{c_j} \end{bmatrix} \right) \\ \bar{P}_l^{c_i} &= \pi_c^{-1} \left(\begin{bmatrix} \hat{u}_l^{c_i} \\ \hat{v}_l^{c_i} \end{bmatrix} \right) \end{aligned}$$

参与相机测量残差的状态量有, $[p_{b_i}^w, q_{b_i}^w]$, $[p_{b_j}^w, q_{b_j}^w]$, $[p_c^b, q_c^b]$ 以及逆深度 λ_l 。所以下面分别对这几个状态量对应的误差状态量求式(4.9)的偏导, 得到高斯迭代过程中的雅克比矩阵。

式(4.9)对 $[\delta p_{b_i}^w, \delta \theta_{b_i}^w]$ 求偏导, 得到 $3 * 6$ 的雅克比矩阵,

$$J[0] = \begin{bmatrix} q_b^c q_w^{b_j} & -q_b^c q_w^{b_j} q_{b_i}^w [q_c^b \frac{\bar{P}_l^c}{\lambda_l} + p_c^b]_{\times} \end{bmatrix} \quad (4.12)$$

式(4.9)对 $[\delta p_{b_j}^w, \delta \theta_{b_j}^w]$ 求偏导, 得到 $3 * 6$ 的雅克比矩阵,

$$J[1] = \begin{bmatrix} -q_b^c q_w^{b_j} & q_b^c q_w^{b_j} [q_{b_i}^w (q_c^b \frac{\bar{P}_l^c}{\lambda_l} + p_c^b) + p_{b_i}^w - p_{b_j}^w]_{\times} \end{bmatrix} \quad (4.13)$$

式(4.9)对 $[\delta p_c^b, \delta \theta_c^b]$ 求偏导, 得到 $3 * 6$ 的雅克比矩阵,

$$J[2] = \begin{bmatrix} q_b^c (q_w^{b_j} q_{b_i}^w - I_{3*3}) & -q_b^c q_w^{b_j} q_{b_i}^w q_c^b [\frac{\bar{P}_l^c}{\lambda_l}]_{\times} + [q_b^c (q_w^{b_j} (q_{b_i}^w p_c^b + p_{b_i}^w - p_{b_j}^w) - p_c^b)] \end{bmatrix} \quad (4.14)$$

式(4.9)对 $\delta \lambda_l$ 求偏导, 得到 $3 * 1$ 的雅克比矩阵,

$$J[3] = -q_b^c q_w^{b_j} q_{b_i}^w q_c^b \frac{\bar{P}_l^c}{\lambda_l^2} \quad (4.15)$$

以上就是相机测量误差以及误差方程的雅克比矩阵的求解过程。

在相机测量误差的文件projection_factor.cpp中的ProjectionFactor::check还没有理解

4.3 闭环修正与优化

这部分内容默认已经检测到闭环, 只涉及到后续的优化部分。

4.4 系统边缘化

4.4.1 边缘化的定义和目的

边缘化(marginalization)的过程就是将滑窗内的某些较旧或者不满足要求的视觉帧剔除的过程, 所以边缘化也被描述为将联合概率分布分解为边缘概率分布和条件概率分布的过程(说白了, 就是利用shur补减少优化参数的过程)。利用Sliding Window做优化的过程中, 边缘化的目的主要有两个:

- 滑窗内的pose和feature个数是有限的, 在系统优化的过程中, 势必要不断将一些pose和feature移除滑窗。
- 如果当前帧图像和上一帧添加到滑窗的图像帧视差很小, 则测量的协方差(重投影误差)会很大, 进而会恶化优化结果。LIFO导致了协方差的增大, 而恶化优化结果?
直接进行边缘化而不加入先验条件的后果:
- 无故地移除这些pose和feature会丢失帧间约束, 会降低了优化器的精度, 所以在移除pose和feature的时候需要将相关联的约束转变为一个先验的约束条件作为prior放到优化问题中
- 在边缘化的过程中, 不加先验的边缘化会导致系统尺度的缺失(参考[6]), 尤其是系统在进行退化运动时(如无人机的悬停和恒速运动)。一般来说只有两个轴向的加速度不为0的时候, 才能保证尺度可观, 而退化运动对于无人机或者机器人来说是不可避免的。所以在系统处于退化运动的时候, 要加入先验信息保证尺度的客观性。
以上就可以描述为边缘化的目的以及在边缘化中加入先验约束的原因。

4.4.2 两种边缘化措施

这两种边缘化的措施主要还是针对悬停和恒速运动等退化运动。下面就边缘化的过程做简要的总结。

设共有 $(B_0, B_1, \dots, B_N, B_{N+1}, \dots, B_{N++n})$ 个状态量，其中状态量 $X = [x_{B_0}^{B_0} \dots x_{B_{N-1}}^{B_0} | \lambda_l]$ 中的加速度计是经过充分激励的，那么状态量 $x_{B_N}^{B_0}$ 只有满足下面两个条件之一的时候才能被加入到滑窗内。

- (1) 两帧图像之间的时间差 Δt 超过阈值 δ 。
- (2) 排除旋转运动，两帧之间共同Features的视差超过阈值 ϵ 。

其中，条件(1)避免了两帧图像之间的IMU长时间积分，而出现漂移。条件(2)保证了系统的运动时，有足够视差的共视帧能够被加入到滑窗。

因为滑窗的大小是固定的，要加入新的Keyframe，就要从滑窗中剔除旧的Keyframe。在VINS-Mono中剔除旧帧有两种方式，剔除滑窗的首帧或者倒数第二帧(假设滑窗默认是由右向左滑的！)。所以关于剔除旧帧也有一定的剔除规则或者说是边缘化规则。

设定一个变量 $S = \text{float}/\text{fix}$ ， S 是由滑窗内的最新的两个Keyframe视差决定的，如果视差大于阈值 ϵ ，则边缘化最旧帧 $x_{B_0}^{B_0}$ ，如果视差小于 ϵ ，则边缘化倒数第二帧 $x_{B_{N-1}}^{B_0}$ 。具体可以参考文献[6]中 *Algorithm 1*。要注意边缘化的时候，不仅要移除相机位姿，被该相机首次观测到的Features也要移除。最终构建出的先验约束可以写为，

$$\Lambda_p^+ = \Lambda_p + \sum_{k \in I} H_{B_{k+1}}^{B_k^T} P_{B_{k+1}}^{B_k^{-1}} H_{B_{k+1}}^{B_k} + \sum_{l \in C} H_l^{B_j^T} P_l^{B_j^{-1}} H_l^{B_j} \quad (4.16)$$

其中 Λ_p 即为式(4.4)中的先验约束条件，矩阵 H, P 分别对应被移除的IMU和视觉状态量对应的雅克比矩阵和协方差矩阵。

由上面的剔除策略可知，当飞机处于悬停或者微小运动的时候，会一直边缘化新的视差较小的视觉帧。在保留了旧帧的同时，也保留了加速度信息，保证了尺度的可观性。但是，当系统处于速度较大的恒速运动时，加速度信息会伴随着旧帧移除，因此会发生尺度的漂移。(沈老师也在视频中提到，系统对恒速运动处理的还不是很好)。

图1.(a)所示为第一种边缘化方式，当状态量5和状态量4之间的视差过小的时，在下一状态量6添加的时候，会把状态量5给边缘化，包括特征点 f_1 也会被剔除。但是要注意的是这个时候状态量6也处于float状态，只有比较了状态量4和6的视差之后，6的状态才能确定。这种边缘化措施一般用在无人机处于悬停或者是微小运动的时候。

图1.(b)所示为第二种边缘化方式，如果状态量4和状态量5之间有足够大的视差则边缘化状态量0，接受状态量6，进一步判断状态量6的状态。

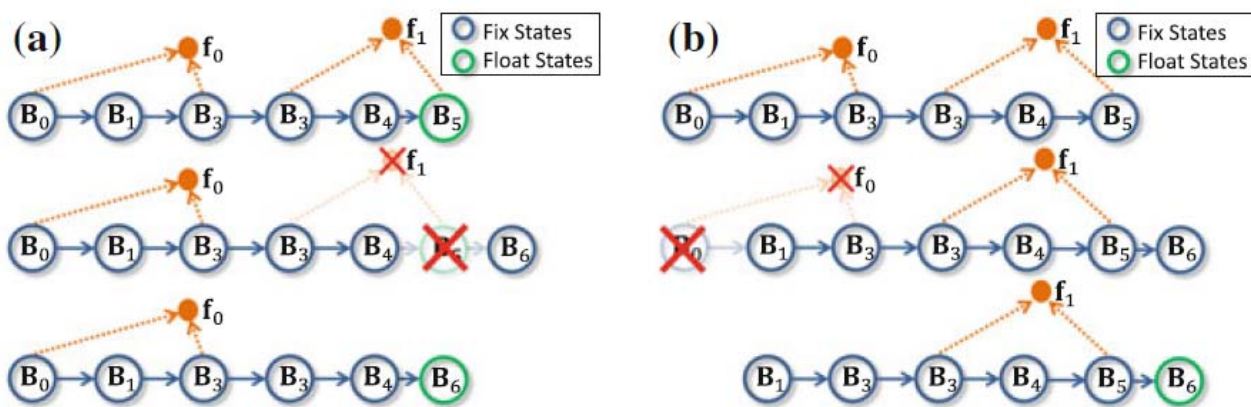


图1 两种边缘化方式示意图

4.4.3 舒尔补边缘化优化状态量

这一小节主要总结下面两个问题：舒尔补边缘化优化状态量和式(4.2)的非线性优化过程中的优化一致性(FEJ)。

设状态量 δx_m 和 δx_r 分别是需要被边缘化和保留的状态量，根据式(4.4)的线性方程及H矩阵的稀疏性和特性，将式(4.4)进一步写作：

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \delta x_m \\ \delta x_r \end{bmatrix} = \begin{bmatrix} b_m \\ b_r \end{bmatrix} \quad (4.17)$$

在式(4.17)两边同乘矩阵，得

$$\begin{bmatrix} I & 0 \\ -CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \delta x_m \\ \delta x_r \end{bmatrix} = \begin{bmatrix} I & 0 \\ -CA^{-1} & I \end{bmatrix} \begin{bmatrix} b_m \\ b_r \end{bmatrix}$$

可得

$$(-CA^{-1}B + D)\delta x_r = -CA^{-1}b_m + b_r \quad (4.18)$$

令,

$$H = (-CA^{-1}B + D), \quad b = -CA^{-1}b_m + b_r$$

在进行舒尔补过程中, 已经将 δx_m 给边缘掉, 得到了关于 δx_r 新的带有先验信息的关系式。然后将这部分带入到代价函数的边缘化残差中即可。也可以说, 由边缘化得到了条件概率 $p(\delta x_b | \delta x_m) \sim N(-CA^{-1}b_m, -CA^{-1}B + D)$ 。正如文献[7]中作者说的, "SLAM is tracking a normal distribution through a large state space"

4.4.4 关于优化一致性(First Estimate Jacobin, FEJ算法)

FEJ算法貌似是在边缘化过程中比较重要的内容。付兴银师兄在讲解OKVIS的[博客](#)中提到, 在Pose和Landmark分开边缘化的时候, 要使用FEJ算法, 为了只计算一次雅克比矩阵(还没弄懂)。博哥在毕设论文中提到, 在IMU和视觉测量残差分开边缘化的时候, 要保证优化一致性, 确保各部分在线性化的时候具有相同的线性化点。

个人理解, 这个地方应该有两层意思。回想高斯迭代过程中, 我们在求取雅克比矩阵的时候是针对 x_0 求取的, 在后面的迭代更新过程中, 雅克比矩阵是不变的(或者说是求取雅克比矩阵的点是不变的, 仍然在 x_0 处。只能说是还是在 x_0 附近计算雅克比, 而不是更新之后的 x), 也就是称作"fix the linearization point"。这样的话雅克比矩阵就是不变的了。但是仅能说, 在marg掉某些点之后, 和这些点相关联的点的雅克比矩阵是不变的。

4.4.5 滑窗优化(Slide window)

按理说是不应该把滑窗当做一小节来讲的, 边缘化、舒尔补都属于滑窗的范围, 但前面已经总结了。而且正如贺一加师兄在其[博客](#)中提到的, 滑窗的三大法宝"Marginalization", "Schur complement", "First estimate jacobian", 这些在前面也提到了, 所以这一小节基本没什么可以总结的地方。关于这部分内容, 可以参考这几篇博客, 讲的真是太好了。[okvis1](#), [okvis2](#), [知行合一1](#), [知行合一2](#)

4.5 系统关键帧选择

要注意的是在vins-mono中, 添加关键帧的时候是在添加完新的图像帧, 做完滑窗优化之后, 将滑窗内的倒数第二帧作为关键帧, 并加入到了关键帧序列当中, 具体[位置](#)。

Vins-mono在选择关键帧的共有两个条件:

- 根据平均视差(这里的视差包括了平移量和旋转量);
- 根据tracking的状态(以跟踪到的Features为主)。

对于视差的判断, 在特征处理主程序Estimator::processImage就有, 但是关于视差的判断, 并没有看到关于角度的补偿, 或者是在后面边缘化的具体过程中存在角度的补偿。关于视差判断的具体思路是:

建立包含滑窗内所有特征点集合的vectorfeature, 若满足以下条件就边缘化最旧帧, 否则边缘化倒数第二帧

- 滑窗内的关键帧个数小于3
- 如果当前帧的Features在滑窗内被观测到的个数小于20
- 计算滑窗内所有满足条件的Feature(至少有三帧观测到该Feature)的观测帧倒数第二帧和倒数第三帧的视差(归一化平面上的距离), 如果最终的平均距离大于10个像素点
- 满足上面条件的Feature的个数为0的时候

5 闭环校正

直接法不能像特征法那样直接将描述子利用DboW进行闭环检测, 所以在闭环的时候只能重新对Keyframe提取Brief等描述子建立Database进行闭环检测, 或者使用开源的闭环检测库[FabMap](#)来进行闭环检测。Vins-Mono选择了第一种闭环检测的方法。

5.1 闭环检测

Vins-Mono还是利用词袋的形式来做Keyframe Database的构建和查询。在建立闭环检测的数据库时, 关键帧的Features包括两部分: VIO部分的200个强角点和500 Fast角点。然后描述子仍然使用BRIEF(因为旋转可观, 匹配过程中对旋转有一定的适应性, 所以不用使用ORB)。

5.2 闭环校正

在闭环检测成功之后，会得到回环候选帧。所以要在已知位姿的回环候选帧和滑窗内的匹配帧做匹配，然后把回环帧加入到滑窗的优化当中，这时整个滑窗的状态量的维度是不发生变化的，因为回环帧的位姿是固定的。

6 全局的位姿优化

因为之前做的非线性优化本质只是在一个滑窗之内求解出了相机的位姿，而且在回环检测部分，利用固定位姿的回环帧只是纠正了滑窗内的相机位姿，并没有修正其他位姿(或者说没有将回环发现的误差分配到整个相机的轨迹上)，缺少全局的一致性，所以要进行一次全局的Pose Graph。**全局的Pose Graph较之滑窗有一定的迟滞性**，只有相机的Pose滑出滑窗的时候，Pose才会被加到全局的Pose Graph当中。

7 关键变量含义和代码注释中的一些称呼

因为VINS-Mono的代码写的有点乱，好多变量的命名过于随意，不能很好理解，而且为了注释方便自己也给某些变量或者环节做了命名。

7.1 代码注释中的自定义名称

- 闭环帧和匹配帧
闭环帧：在闭环匹配成功之后，Keyframe Database中的旧帧;匹配帧：当前滑窗之内，和闭环帧匹配的匹配帧。

7.2 代码中的关键变量

- [image\(estimator.cpp\)](#)
某一帧图像得到的某个特征点，某一帧图像得到的特征点， $\langle \text{Feature_id}, \langle \text{camera_id}, \text{Feature} \rangle \rangle$
- f_manager
滑窗的特征点管理器
- [keyframe_database\(estimator_node.cpp\)](#)
关键帧集合，每次在闭环检测线程中增加新元素，每个关键帧在滑窗内的导数第二个位置时被**添加**。
- [keyframe_buf\(estimator_node.cpp\)](#)
为闭环检测提供临时的Keyframe存储，**只包含滑窗内的倒数第二Keyframe**，当其包含的某帧进入闭环检测环节会被**弹出**。
- [retrive_data_vector](#)
存储闭环检测结果，包块闭环帧和匹配帧的位姿关系、匹配帧的ID、闭环帧的位姿、闭环帧中良好的匹配点以及匹配点良好匹配点的ID。

8 遗留的问题

1.为什么至少两个轴向上的加速度不为0的时候，尺度才是客观的。

在文献6的第3小节提到，在单目和视觉融合的时候，只有当至少两个轴向的加速度不为0的时候，尺度才是客观的。针对无人机来说，无人机的正常俯仰、横滚运动都会在两个轴向上有加速度。加速度仅存在于一个轴向的情况多属于平面运动的机器人。

2.当视差角过小的时候，视觉测量的协方差会很大。

•

内容目录

- ■ [VINS-Mono代码分析总结](#)
 - [参考文献](#)
 - [前言](#)
 - [1 预积分的推导](#)
 - [1.1 离散状态下预积分方程](#)
 - [1.2 离散状态下误差状态方程](#)
 - [2 前端KLT跟踪](#)
 - [3 系统初始化](#)
 - [3.1 相机与IMU之间的相对旋转](#)
 - [3.2 相机初始化](#)
 - [3.3 视觉与IMU对齐](#)

- [3.3.1 陀螺仪Bias修正](#)
 - [3.3.2 初始化速度、重力向量 \$g\$ 和尺度因子](#)
 - [3.3.3 纠正重力向量](#)
- [4 后端优化](#)
 - [4.1 IMU测量误差](#)
 - [4.2 相机测量误差](#)
 - [4.3 闭环修正与优化](#)
 - [4.4 系统边缘化](#)
 - [4.4.1 边缘化的定义和目的](#)
 - [4.4.2 两种边缘化措施](#)
 - [4.4.3 舒尔补边缘化优化状态量](#)
 - [4.4.4 关于优化一致性\(First Estimate Jacobin, FEJ算法\)](#)
 - [4.4.5 滑窗优化\(Slide window\)](#)
 - [4.5 系统关键帧选择](#)
- [5 闭环校正](#)
 - [5.1 闭环检测](#)
 - [5.2 闭环校正](#)
- [6 全局的位姿优化](#)
- [7 关键变量含义和代码注释中的一些称呼](#)
 - [7.1 代码注释中的自定义名称](#)
 - [7.2 代码中的关键变量](#)
- [8 遗留的问题](#)
 - [2.当视差角过小的时候,视觉测量的协方差会很大。](#)

•

•

- [Markdown语法 1](#)
- [CMD Markdown语法](#)
- [SLAM 2](#)
 - [深度滤波器](#)
 - [VINS-Mono代码分析总结](#)
- [其他 2](#)
 - [RARSAC算法](#)
 - [ROS控制移动底座](#)
- [数学基础 1](#)
 - [Ceres总结](#)
- 搜索 Xiaobuyi 的文稿标
- 以下【标签】将用于标记这篇文稿：

•

•

•

•

- [下载客户端](#)
- [关注开发者](#)
- [报告问题, 建议](#)
- [联系我们](#)

•

添加新批注



保存 取消

在作者公开此批注前, 只有你和作者可见。



保存 取消



修改 保存 取消 删除

• 私有

- 公开
- 删除

查看更早的 5 条回复
回复批注



通知

取消 确认

- ☐
- ☐