深蓝学院
shenlanxueyuan.com

多传感器融合定位

第9讲 基于图优化的建图方法

主讲人 任 乾

北京理工大学本硕
自动驾驶从业者

目录

1. 基于预积分的融合方案流程
2. 预积分模型设计
3. 预积分在优化中的使用
4. 典型方案介绍
5. 融合编码器的优化方案

目录

1. 基于预积分的融合方案流程
2. 预积分模型设计
3. 预积分在优化中的使用
4. 典型方案介绍
5. 融合编码器的优化方案

基于预积分的融合方案流程

1. 优化问题分析

优化问题可以等效为如下形式

$$\min_{\mathbf{X}} \left\{ \underbrace{\boxed{\sum \left\|\mathbf{r}_{L}(\boldsymbol{L}_{k}^{k+1}, \mathbf{X})\right\|^{2}}}_{\text{激光里程计约束}} + \underbrace{\boxed{\sum \left\|\mathbf{r}_{B}\left(\boldsymbol{B}_{k}^{k+1}, \mathbf{X}\right)\right\|^{2}}}_{\text{IMU约束}} + \underbrace{\boxed{\sum \left\|\mathbf{r}_{G}(\boldsymbol{G}_{k}, \boldsymbol{X})\right\|^{2}}}_{\text{RTK约束}} \right\}$$

三种约束分别通过以下方式获得：

1) 激光里程计约束：使用激光里程计，计算每个关键帧位姿，进而得到相对位姿；

2) IMU约束：在上一个关键帧位姿基础上，进行惯性积分，从而得到两关键帧相对位姿；

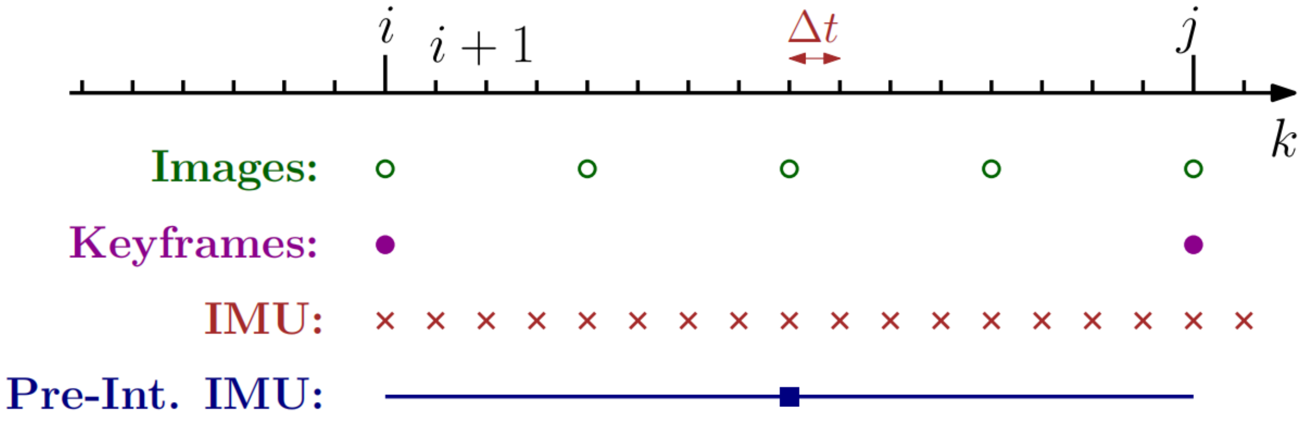
3) RTK约束：直接测量得到。

基于预积分的融合方案流程

2. 预积分的作用

问题：位姿每次优化后会发生变化，其后的IMU惯性积分就要重新进行，运算量过大。

解决思路：直接计算两帧之间的相对位姿，而不依赖初始值影响，即所谓的预积分。

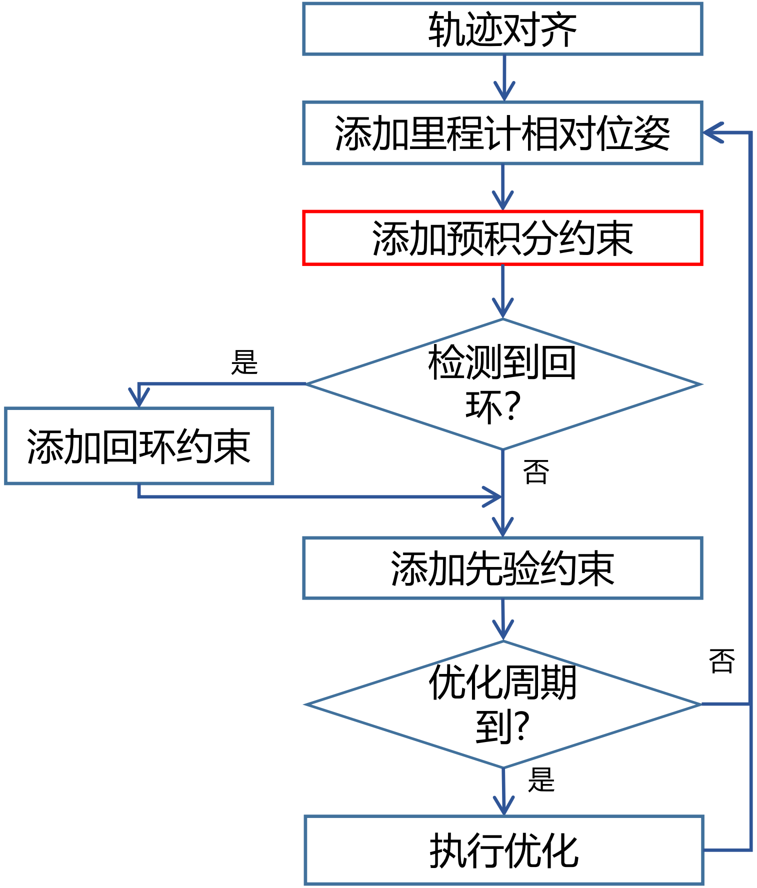


基于预积分的融合方案流程

3. 基于预积分的建图方案流程

由于此处讨论的优化方案包含组合导航系统，且认为外参已标定，因此会和常见的lio/vio中的方案有所不同，它不包含以下内容：

1) 初始化lidar和IMU之间的外参；
2) 初始化速度、陀螺仪bias等；
3) 初始化重力；
4) 世界坐标系对齐(组合导航已经对齐)。



点云地图建立流程图

目录

1. 基于预积分的融合方案流程

2. 预积分模型设计

3. 预积分在优化中的使用

4. 典型方案介绍

5. 融合编码器的优化方案

预积分模型设计

在第6讲中，已知导航的微分方程如下

$$\dot{\mathbf{p}}_{wb_t} = \mathbf{v}_t^w \tag{1}$$

$$\dot{\mathbf{v}}_t^w = \mathbf{a}_t^w \tag{2}$$

$$\dot{\mathbf{q}}_{wb_t} = \mathbf{q}_{wb_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2}\boldsymbol{\omega}^{b_t} \end{bmatrix} \tag{3}$$

根据该微分方程，可知从 i 时刻到 j 时刻 IMU的积分结果为

$$\mathbf{p}_{wb_j} = \mathbf{p}_{wb_i} + \mathbf{v}_i^w \Delta t + \iint_{t\in[i,j]} \left(\mathbf{q}_{wb_t}\mathbf{a}^{b_t} - \mathbf{g}^w\right) \delta t^2 \tag{4}$$

$$\mathbf{v}_j^w = \mathbf{v}_i^w + \int_{t\in[i,j]} \left(\mathbf{q}_{wb_t}\mathbf{a}^{b_t} - \mathbf{g}^w\right) \delta t \tag{5}$$

$$\mathbf{q}_{wb_j} = \int_{t\in[i,j]} \mathbf{q}_{wb_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2}\boldsymbol{\omega}^{b_t} \end{bmatrix} \delta t \tag{6}$$

预积分模型设计

根据预积分的要求，需要求相对结果，而且不依赖于上一时刻位姿，因此需要对上式做转换。

由于 $\mathbf{q}_{wb_t} = \mathbf{q}_{wb_i} \otimes \mathbf{q}_{b_ib_t}$ ，把它带入(4)-(6)式可得

$$\mathbf{p}_{wb_j} = \mathbf{p}_{wb_i} + \mathbf{v}_i^w \Delta t - \frac{1}{2}\mathbf{g}^w \Delta t^2 + \mathbf{q}_{wb_i} \boxed{\iint_{t\in[i,j]} \left(\mathbf{q}_{b_ib_t}\mathbf{a}^{b_t}\right) \delta t^2} \tag{7}$$

$$\mathbf{v}_j^w = \mathbf{v}_i^w - \mathbf{g}^w \Delta t + \mathbf{q}_{wb_i} \boxed{\int_{t\in[i,j]} \left(\mathbf{q}_{b_ib_t}\mathbf{a}^{b_t}\right) \delta t} \tag{8}$$

$$\mathbf{q}_{wb_j} = \mathbf{q}_{wb_i} \boxed{\int_{t\in[i,j]} \mathbf{q}_{b_ib_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2}\boldsymbol{\omega}^{b_t} \end{bmatrix} \delta t} \tag{9}$$

可见，此时需要积分的项，就完全和i时刻的状态无关了。

预积分模型设计

为了整理公式，把积分相关的项用下面的式子代替

$$\boldsymbol{\alpha}_{b_i b_j} = \iint_{t\in[i,j]} \left(\mathbf{q}_{b_i b_t}\mathbf{a}^{b_t}\right) \delta t^2 \tag{10}$$

$$\boldsymbol{\beta}_{b_i b_j} = \int_{t\in[i,j]} \left(\mathbf{q}_{b_i b_t}\mathbf{a}^{b_t}\right) \delta t \tag{11}$$

$$\mathbf{q}_{b_i b_j} = \int_{t\in[i,j]} \mathbf{q}_{b_i b_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2}\boldsymbol{\omega}^{b_t} \end{bmatrix} \delta t \tag{12}$$

实际使用中使用离散形式，而非连续形式，由于在解算中，一般采用中值积分方法，即

$$\boldsymbol{\omega} = \frac{1}{2}\left[\left(\boldsymbol{\omega}^{b_k} - \mathbf{b}_k^g\right) + \left(\boldsymbol{\omega}^{b_{k+1}} - \mathbf{b}_k^g\right)\right] \tag{13}$$

$$\mathbf{a} = \frac{1}{2}\left[\mathbf{q}_{b_i b_k}\left(\mathbf{a}^{b_k} - \mathbf{b}_k^a\right) + \mathbf{q}_{b_i b_{k+1}}\left(\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a\right)\right] \tag{14}$$

那么预积分的离散形式可以表示为

$$\boldsymbol{\alpha}_{b_i b_{k+1}} = \boldsymbol{\alpha}_{b_i b_k} + \boldsymbol{\beta}_{b_i b_k}\delta t + \frac{1}{2}\mathbf{a}\delta t^2 \tag{15}$$

$$\boldsymbol{\beta}_{b_i b_{k+1}} = \boldsymbol{\beta}_{b_i b_k} + \mathbf{a}\delta t \tag{16}$$

$$\mathbf{q}_{b_i b_{k+1}} = \mathbf{q}_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\boldsymbol{\omega}\delta t \end{bmatrix} \tag{17}$$

预积分模型设计

经过以上的推导，此时状态更新的公式可以整理为

$$\begin{bmatrix} \mathbf{p}_{wb_j} \\ \mathbf{v}_j^w \\ \mathbf{q}_{wb_j} \\ \mathbf{b}_j^a \\ \mathbf{b}_j^g \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{wb_i} + \mathbf{v}_i^w \Delta t - \frac{1}{2}\mathbf{g}^w \Delta t^2 + \mathbf{q}_{wb_i} \boldsymbol{\alpha}_{b_i b_j} \\ \mathbf{v}_i^w - \mathbf{g}^w \Delta t + \mathbf{q}_{wb_i} \boldsymbol{\beta}_{b_i b_j} \\ \mathbf{q}_{wb_i} \mathbf{q}_{b_i b_j} \\ \mathbf{b}_i^a \\ \mathbf{b}_i^g \end{bmatrix}$$

需要注意的是，陀螺仪和加速度计的模型为

$$\mathbf{b}_{k+1}^a = \mathbf{b}_k^a + \mathbf{n}_{\mathbf{b}_k^a} \delta t$$
$$\mathbf{b}_{k+1}^g = \mathbf{b}_k^g + \mathbf{n}_{\mathbf{b}_k^g} \delta t$$

即认为bias是在变化的，这样便于估计不同时刻的bias值，而不是整个系统运行时间内都当做常值对待。这更符合低精度mems的实际情况。但在预积分时，由于两个关键帧之间的时间较短，因此认为i和j时刻的bias相等。

预积分模型设计

需要注意的一点是，预积分的结果中包含了bias，在优化过程中，bias作为状态量也会发生变化，从而引起预积分结果变化。

为了避免bias变化后，重新做预积分，可以把预积分结果在bias处泰勒展开，表达成下面的形式，这样就可以根据bias的变化量直接算出新的预积分结果。

$$\boldsymbol{\alpha}_{b_ib_j} = \bar{\boldsymbol{\alpha}}_{b_ib_j} + \mathbf{J}^{\alpha}_{b_i^a}\delta\mathbf{b}_i^a + \mathbf{J}^{\alpha}_{b_i^g}\delta\mathbf{b}_i^g$$

$$\boldsymbol{\beta}_{b_ib_j} = \bar{\boldsymbol{\beta}}_{b_ib_j} + \mathbf{J}^{\beta}_{b_i^a}\delta\mathbf{b}_i^a + \mathbf{J}^{\beta}_{b_i^g}\delta\mathbf{b}_i^g$$

$$\mathbf{q}_{b_ib_j} = \bar{\mathbf{q}}_{b_ib_j} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\mathbf{J}^{q}_{b_i^g}\delta\mathbf{b}_i^g \end{bmatrix}$$

其中

$$\mathbf{J}^{\alpha}_{b_i^a} = \frac{\partial \boldsymbol{\alpha}_{b_ib_j}}{\partial \delta \mathbf{b}_i^a}$$

$$\mathbf{J}^{\alpha}_{b_i^g} = \frac{\partial \boldsymbol{\alpha}_{b_ib_j}}{\partial \delta \mathbf{b}_i^g}$$

$$\mathbf{J}^{\beta}_{b_i^a} = \frac{\partial \boldsymbol{\beta}_{b_ib_j}}{\partial \delta \mathbf{b}_i^a}$$

$$\mathbf{J}^{\beta}_{b_i^g} = \frac{\partial \beta_{b_ib_j}}{\partial \delta \mathbf{b}_i^g}$$

$$\mathbf{J}^{q}_{b_i^g} = \frac{\mathbf{q}_{b_ib_j}}{\partial \mathbf{b}_i^g}$$

注：此处暂时不直接给出以上各雅可比的结果，它的推导放在后面进行。

目录

1. 基于预积分的融合方案流程
2. 预积分模型设计
3. 预积分在优化中的使用
4. 典型方案介绍
5. 融合编码器的优化方案

预积分在优化中的使用

1. 使用方法

1）凸优化回顾

损失函数由残差函数组成

$$\min_x F(x) = \frac{1}{2}\|f(x)\|_2^2$$

当考虑方差时，可以写为

$$\min_x F(x) = f(x)^T \Omega f(x)$$

利用高斯牛顿方法，求解该优化问题，需要解下面的方程

$$\underbrace{J^T \Omega J}_{H} \Delta x = \underbrace{-J^T \Omega f(x)}_{g}$$

即，在优化中使用一项信息，需要设计残差，并推导它的雅可比和方差

2）预积分的使用

按照优化的套路，在优化中使用预积分，需要：

a. 设计残差
b. 推导残差关于待优化变量的雅可比
c. 计算残差的方差

除此以外，预积分中还多一项计算，即推导bias变化时，预积分量重新计算的方法。

预积分在优化中的使用

2. 残差设计

在优化时，需要知道残差关于状态量的雅可比。由于已知姿态位姿更新的方法如下

$$\begin{bmatrix} \mathbf{p}_{wb_j} \\ \mathbf{q}_{wb_j} \\ \mathbf{v}_j^w \\ \mathbf{b}_j^a \\ \mathbf{b}_j^g \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{wb_i} + \mathbf{v}_i^w \Delta t - \frac{1}{2}\mathbf{g}^w \Delta t^2 + \mathbf{q}_{wb_i} \boldsymbol{\alpha}_{b_i b_j} \\ \mathbf{q}_{wb_i} \mathbf{q}_{b_i b_j} \\ \mathbf{v}_i^w - \mathbf{g}^w \Delta t + \mathbf{q}_{wb_i} \boldsymbol{\beta}_{b_i b_j} \\ \mathbf{b}_i^a \\ \mathbf{b}_i^g \end{bmatrix}$$

因此，可以很容易写出一种残差形式如下：

$$\begin{bmatrix} \mathbf{r}_p \\ \mathbf{r}_q \\ \mathbf{r}_v \\ \mathbf{r}_{ba} \\ \mathbf{r}_{bg} \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{wb_j} - \mathbf{p}_{wb_i} - \mathbf{v}_i^w \Delta t + \frac{1}{2}\mathbf{g}^w \Delta t^2 - \mathbf{q}_{wb_i} \boldsymbol{\alpha}_{b_i b_j} \\ 2\left[\mathbf{q}_{b_i b_j}^* \otimes \left(\mathbf{q}_{wb_i}^* \otimes \mathbf{q}_{wb_j}\right)\right]_{xyz} \\ \mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t - \mathbf{q}_{wb_i} \boldsymbol{\beta}_{b_i b_j} \\ \mathbf{b}_j^a - \mathbf{b}_i^a \\ \mathbf{b}_j^g - \mathbf{b}_i^g \end{bmatrix}$$

2. 残差设计

但是和预积分相关的量，仍然与上一时刻的姿态有关，无法直接加减，因此，把残差修正为以下形式

$$\begin{bmatrix} \mathbf{r}_p \\ \mathbf{r}_q \\ \mathbf{r}_v \\ \mathbf{r}_{ba} \\ \mathbf{r}_{bg} \end{bmatrix} = \begin{bmatrix} \mathbf{q}_{wb_i}^* \left(\mathbf{p}_{wb_j} - \mathbf{p}_{wb_i} - \mathbf{v}_i^w \Delta t + \frac{1}{2}\mathbf{g}^w \Delta t^2\right) - \boldsymbol{\alpha}_{b_i b_j} \\ 2\left[\mathbf{q}_{b_i b_j}^* \otimes \left(\mathbf{q}_{wb_i}^* \otimes \mathbf{q}_{wb_j}\right)\right]_{xyz} \\ \mathbf{q}_{wb_i}^* \left(\mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t\right) - \boldsymbol{\beta}_{b_i b_j} \\ \mathbf{b}_j^a - \mathbf{b}_i^a \\ \mathbf{b}_j^g - \mathbf{b}_i^g \end{bmatrix}$$

待优化的变量是$\begin{bmatrix} \mathbf{p}_{wb_i} & \mathbf{q}_{wb_i} & \mathbf{v}_i^w & \mathbf{b}_i^a & \mathbf{b}_i^g \end{bmatrix}$ $\begin{bmatrix} \mathbf{p}_{wb_j} & \mathbf{q}_{wb_j} & \mathbf{v}_j^w & \mathbf{b}_j^a & \mathbf{b}_j^g \end{bmatrix}$

但在实际使用中，往往都是使用扰动量，因此实际是对以下变量求雅可比

$$\begin{bmatrix} \delta\mathbf{p}_{wb_i} & \delta\theta_{wb_i} & \delta\mathbf{v}_i^w & \delta\mathbf{b}_i^a & \delta\mathbf{b}_i^g \end{bmatrix}$$

$$\begin{bmatrix} \delta\mathbf{p}_{wb_j} & \delta\theta_{wb_j} & \delta\mathbf{v}_j^w & \delta\mathbf{b}_j^a & \delta\mathbf{b}_j^g \end{bmatrix}$$

此处只对几个比较复杂的雅可比进行推导，其余比较简单，感兴趣的可自行完成。

3. 残差雅可比的推导

3.1 姿态残差的雅可比

1)对i时刻姿态误差的雅可比

$$\frac{\partial \mathbf{r}_q}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} = \frac{\partial 2\left[\mathbf{q}_{b_j b_i} \otimes \left(\mathbf{q}_{b_i w} \otimes \mathbf{q}_{w b_j}\right)\right]_{xyz}}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}}$$

$$= \frac{\partial 2\left[\mathbf{q}_{b_i b_j}^{*} \otimes \left(\mathbf{q}_{w b_i} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\delta \boldsymbol{\theta}_{b_i b_i'} \end{bmatrix}\right)^{*} \otimes \mathbf{q}_{w b_j}\right]_{xyz}}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}}$$

$$= \frac{\partial -2\left[\left(\mathbf{q}_{b_i b_j}^{*} \otimes \left(\mathbf{q}_{w b_i} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\delta \boldsymbol{\theta}_{b_i b_i'} \end{bmatrix}\right)^{*} \otimes \mathbf{q}_{w b_j}\right)^{*}\right]_{xyz}}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}}$$

$$= \frac{\partial -2\left[\mathbf{q}_{w b_j}^{*} \otimes \left(\mathbf{q}_{w b_i} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\delta \boldsymbol{\theta}_{b_i b_i'} \end{bmatrix}\right) \otimes \mathbf{q}_{b_i b_j}\right]_{xyz}}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}}$$

预积分在优化中的使用

3. 残差雅可比的推导

3.1 姿态残差的雅可比

上式可以化简为

$$
\begin{aligned}
\frac{\partial \mathbf{r}_q}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} &= -2 \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \frac{\partial \mathbf{q}_{wb_j}^* \otimes \left(\mathbf{q}_{wb_i} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\delta \boldsymbol{\theta}_{b_i b_i'} \end{bmatrix} \right) \otimes \mathbf{q}_{b_i b_j}}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} \\
&= -2 \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \frac{\partial \left[\mathbf{q}_{wb_j}^* \otimes \mathbf{q}_{wb_i} \right]_L \left[\mathbf{q}_{b_i b_j} \right]_R \begin{bmatrix} 1 \\ \frac{1}{2}\delta \boldsymbol{\theta}_{b_i b_i'} \end{bmatrix}}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} \\
&= -2 \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \left[\mathbf{q}_{wb_j}^* \otimes \mathbf{q}_{wb_i} \right]_L \left[\mathbf{q}_{b_i b_j} \right]_R \begin{bmatrix} \mathbf{0} \\ \frac{1}{2}\mathbf{I} \end{bmatrix}
\end{aligned}
$$

3. 残差雅可比的推导

3.1 姿态残差的雅可比

2)对j时刻姿态误差的雅可比

$$
\begin{aligned}
\frac{\partial \mathbf{r}_q}{\partial \delta \boldsymbol{\theta}_{b_j b_j'}} &= \frac{\partial 2 \left[\mathbf{q}_{b_i b_j}^* \otimes \mathbf{q}_{w b_i}^* \otimes \mathbf{q}_{w b_j} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} \delta \boldsymbol{\theta}_{b_j b_j'} \end{bmatrix} \right]_{xyz}}{\partial \delta \boldsymbol{\theta}_{b_j b_j'}} \\
&= \frac{\partial 2 \left[\left[\mathbf{q}_{b_i b_j}^* \otimes \mathbf{q}_{w b_i}^* \otimes \mathbf{q}_{w b_j} \right]_L \begin{bmatrix} 1 \\ \frac{1}{2} \delta \boldsymbol{\theta}_{b_j b_j'} \end{bmatrix} \right]_{xyz}}{\partial \delta \boldsymbol{\theta}_{b_j b_j'}} \\
&= 2 \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \left[\mathbf{q}_{b_i b_j}^* \otimes \mathbf{q}_{w b_i}^* \otimes \mathbf{q}_{w b_j} \right]_L \begin{bmatrix} \mathbf{0} \\ \frac{1}{2} \mathbf{I} \end{bmatrix}
\end{aligned}
$$

预积分在优化中的使用

3. 残差雅可比的推导

3.1 姿态残差的雅可比

3)对i时刻陀螺仪bias误差的雅可比

$$
\begin{aligned}
\frac{\partial \mathbf{r}_q}{\partial \delta \mathbf{b}_i^g} &= \frac{\partial 2\left[\left(\mathbf{q}_{b_i b_j} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\mathbf{J}_{b_i^g}^q \delta \mathbf{b}_i^g \end{bmatrix}\right)^* \otimes \mathbf{q}_{wb_i}^* \otimes \mathbf{q}_{wb_j}\right]_{xyz}}{\partial \delta \mathbf{b}_i^g} \\
&= \frac{\partial -2\left[\left(\left(\mathbf{q}_{b_i b_j} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\mathbf{J}_{b_i^g}^q \delta \mathbf{b}_i^g \end{bmatrix}\right)^* \otimes \mathbf{q}_{wb_i}^* \otimes \mathbf{q}_{wb_j}\right)^*\right]_{xyz}}{\partial \delta \mathbf{b}_i^g} \\
&= \frac{\partial -2\left[\mathbf{q}_{wb_j}^* \otimes \mathbf{q}_{wb_i} \otimes \left(\mathbf{q}_{b_i b_j} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\mathbf{J}_{b_i^g}^q \delta \mathbf{b}_i^g \end{bmatrix}\right)\right]_{xyz}}{\partial \delta \mathbf{b}_i^g} \\
&= -2\begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix}\left[\mathbf{q}_{wb_j}^* \otimes \mathbf{q}_{wb_i} \otimes \mathbf{q}_{b_i b_j}\right]_L \begin{bmatrix} \mathbf{0} \\ \frac{1}{2}\mathbf{J}_{b_i^g}^q \end{bmatrix}
\end{aligned}
$$

3. 残差雅可比的推导

3.2 速度残差的雅可比

1)对i时刻姿态误差的雅可比

$$
\begin{aligned}
\frac{\partial \mathbf{r}_v}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} &= \frac{\partial \left(\mathbf{q}_{wb_i} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\delta \boldsymbol{\theta}_{b_i b_i'} \end{bmatrix} \right)^{-1} \left(\mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t \right)}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} \\
&= \frac{\partial \left(\mathbf{R}_{wb_i} \exp \left(\left[\delta \boldsymbol{\theta}_{b_i b_i'} \right]_\times \right) \right)^{-1} \left(\mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t \right)}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} \\
&= \frac{\partial \exp \left(\left[-\delta \boldsymbol{\theta}_{b_i b_i'} \right]_\times \right) \mathbf{R}_{b_i w} \left(\mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t \right)}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} \\
&= \frac{\partial \left(\mathbf{I} - \left[\delta \boldsymbol{\theta}_{b_i b_i'} \right]_\times \right) \mathbf{R}_{b_i w} \left(\mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t \right)}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} \\
&= \frac{\partial - \left[\delta \boldsymbol{\theta}_{b_i b_i'} \right]_\times \mathbf{R}_{b_i w} \left(\mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t \right)}{\partial \delta \boldsymbol{\theta}_{b_i b_i'}} \\
&= \left[\mathbf{R}_{b_i w} \left(\mathbf{v}_j^w - \mathbf{v}_i^w + \mathbf{g}^w \Delta t \right) \right]_\times
\end{aligned}
$$

3. 残差雅可比的推导

3.2 速度残差的雅可比

2)对i时刻速度误差的雅可比

$$\frac{\partial \mathbf{r}_v}{\partial \delta \mathbf{v}_i^w} = -\mathbf{R}_{b_i w}$$

3)对i时刻加速度bias误差的雅可比

$$\frac{\partial \mathbf{r}_v}{\partial \delta \mathbf{b}_i^a} = -\frac{\partial \boldsymbol{\beta}_{b_i b_j}}{\partial \delta \mathbf{b}_i^a} = -\mathbf{J}_{b_i^a}^{\beta}$$

3.3 位置残差的雅可比

由于位置残差的形式与速度残差极其相似，因此不再重复推导。

预积分在优化中的使用

4. 预积分方差计算

4.1 核心思路

在融合时，需要给不同信息设置权重，而权重由方差得来，因此对于IMU积分，也要计算其方差。方差的计算形式与第四章相同，即

$$\boldsymbol{P}_{i,k+1} = \mathbf{F}_k \boldsymbol{P}_{i,k} \mathbf{F}_k^\top + \mathbf{B}_k \boldsymbol{Q} \mathbf{B}_k^\top$$

但需注意的是，此处$\boldsymbol{F}_k$和$\boldsymbol{G}_k$是离散时间下的状态传递方程中的矩阵，而我们一般是在连续时间下推导微分方程，再用它计算离散时间下的传递方程。

4.2 连续时间下的微分方程

连续时间下的微分方程一般写为如下形式

$$\dot{\boldsymbol{x}} = \boldsymbol{F}_t \boldsymbol{x} + \boldsymbol{B}_t \boldsymbol{w}$$

$$\boldsymbol{x} = \begin{bmatrix} \delta\boldsymbol{\alpha}_t^{b_k} \\ \delta\boldsymbol{\theta}_t^{b_k} \\ \delta\boldsymbol{\beta}_t^{b_k} \\ \delta\boldsymbol{b}_{\boldsymbol{a}_t} \\ \delta\boldsymbol{b}_{\boldsymbol{w}_t} \end{bmatrix}$$

$$\boldsymbol{w} = \begin{bmatrix} \boldsymbol{n}_a \\ \boldsymbol{n}_w \\ \boldsymbol{n}_{b_a} \\ \boldsymbol{n}_{b_w} \end{bmatrix}$$

以上变量排列顺序是为了与lio/vio等常见系统顺序保持一致。

4. 预积分方差计算

4.2 连续时间下的微分方程

推导方法已经在第6讲介绍，此处直接给出结果

$\delta\dot{\boldsymbol{\theta}}_t^{b_k}$ 的微分方程

$$\delta\dot{\boldsymbol{\theta}}_t^{b_k} = -\left[\boldsymbol{\omega}_t - \boldsymbol{b}_{\omega_t}\right]_\times \delta\boldsymbol{\theta}_t^{b_k} + \boldsymbol{n}_\omega - \delta\boldsymbol{b}_{\omega_t}$$

$\delta\dot{\boldsymbol{\beta}}_t^{b_k}$ 的微分方程

$$\delta\dot{\boldsymbol{\beta}}_t^{b_k} = -\boldsymbol{R}_t\left[\boldsymbol{a}_t - \boldsymbol{b}_{a_t}\right]_\times \delta\boldsymbol{\theta}_t^{b_k} + \boldsymbol{R}_t(\boldsymbol{n}_a - \delta\boldsymbol{b}_{a_t})$$

$\delta\dot{\boldsymbol{\alpha}}_t^{b_k}$ 的微分方程

$$\delta\dot{\boldsymbol{\alpha}}_t^{b_k} = \delta\boldsymbol{\beta}_t^{b_k}$$

4. 预积分方差计算

4.3 离散时间下的传递方程

得到连续时间微分方程以后，就可以计算离散时间的递推方程了，表示为

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{B}_k \boldsymbol{w}_k$$

其中

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} \delta\boldsymbol{\alpha}_{k+1} \\ \delta\boldsymbol{\theta}_{k+1} \\ \delta\boldsymbol{\beta}_{k+1} \\ \delta\boldsymbol{b}_{a_{k+1}} \\ \delta\boldsymbol{b}_{\omega_{k+1}} \end{bmatrix} \qquad \boldsymbol{x}_k = \begin{bmatrix} \delta\boldsymbol{\alpha}_k \\ \delta\boldsymbol{\theta}_k \\ \delta\boldsymbol{\beta}_k \\ \delta\boldsymbol{b}_{a_k} \\ \delta\boldsymbol{b}_{\omega_k} \end{bmatrix} \qquad \boldsymbol{w}_k = \begin{bmatrix} \boldsymbol{n}_{a_k} \\ \boldsymbol{n}_{w_k} \\ \boldsymbol{n}_{a_{k+1}} \\ \boldsymbol{n}_{w_{k+1}} \\ \boldsymbol{n}_{b_a} \\ \boldsymbol{n}_{b_w} \end{bmatrix}$$

4. 预积分方差计算

4.3 离散时间下的传递方程

1） $\delta\boldsymbol{\theta}_{k+1}$ 的求解

由于连续时间下有

$$\delta\dot{\boldsymbol{\theta}} = -\left[\boldsymbol{\omega}_t - \boldsymbol{b}_{\omega_t}\right]_\times \delta\boldsymbol{\theta} + \boldsymbol{n}_\omega - \delta\boldsymbol{b}_{\omega_t}$$

则离散时间下应该有

$$\delta\dot{\boldsymbol{\theta}}_k = -\left[\frac{\boldsymbol{\omega}_k + \boldsymbol{\omega}_{k+1}}{2} - \boldsymbol{b}_{\omega_t}\right]_\times \delta\boldsymbol{\theta}_k + \frac{\boldsymbol{n}_{\omega_k} + \boldsymbol{n}_{\omega_{k+1}}}{2} - \delta\boldsymbol{b}_{\omega_k}$$

因此有

$$\delta\boldsymbol{\theta}_{k+1} = \left[\boldsymbol{I} - \left[\frac{\boldsymbol{\omega}_k + \boldsymbol{\omega}_{k+1}}{2} - \boldsymbol{b}_{\omega_k}\right]_\times \delta t\right] \delta\boldsymbol{\theta}_k + \delta t \frac{\boldsymbol{n}_{\omega_k} + \boldsymbol{n}_{\omega_{k+1}}}{2} - \delta t \delta\boldsymbol{b}_{\omega_k}$$

令 $\bar{\boldsymbol{\omega}} = \frac{\boldsymbol{\omega}_k + \boldsymbol{\omega}_{k+1}}{2} - \boldsymbol{b}_{\omega_k}$ ，则上式可以重新写为

$$\delta\boldsymbol{\theta}_{k+1} = \left[\boldsymbol{I} - [\bar{\boldsymbol{\omega}}]_\times \delta t\right] \delta\boldsymbol{\theta}_k + \frac{\delta t}{2}\boldsymbol{n}_{\omega_k} + \frac{\delta t}{2}\boldsymbol{n}_{\omega_{k+1}} - \delta t \delta\boldsymbol{b}_{\omega_k}$$

4. 预积分方差计算

4.3 离散时间下的传递方程

2） $\delta\boldsymbol{\beta}_{k+1}$ 的求解

由于连续时间下有

$$\delta\dot{\boldsymbol{\beta}} = -\boldsymbol{R}_t[\boldsymbol{a}_t - \boldsymbol{b}_{a_t}]_\times\delta\boldsymbol{\theta} + \boldsymbol{R}_t(\boldsymbol{n}_a - \delta\boldsymbol{b}_{a_t})$$

则离散时间下应该有

$$\begin{aligned}
\delta\dot{\boldsymbol{\beta}}_k = &-\frac{1}{2}\boldsymbol{R}_k[\boldsymbol{a}_k - \boldsymbol{b}_{a_k}]_\times\delta\boldsymbol{\theta}_k \\
&-\frac{1}{2}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times\delta\boldsymbol{\theta}_{k+1} \\
&+\frac{1}{2}\boldsymbol{R}_k\boldsymbol{n}_{a_k} \\
&+\frac{1}{2}\boldsymbol{R}_{k+1}\boldsymbol{n}_{a_{k+1}} \\
&-\frac{1}{2}(\boldsymbol{R}_k + \boldsymbol{R}_{k+1})\delta\boldsymbol{b}_{a_k}
\end{aligned}$$

4. 预积分方差计算

4.3 离散时间下的传递方程

2） $\delta\boldsymbol{\beta}_{k+1}$ 的求解

把前面求得的 $\delta\boldsymbol{\theta}_{k+1}$ 的表达式代入上式，可得

$$
\begin{aligned}
\delta\dot{\boldsymbol{\beta}}_k = & -\frac{1}{2}\boldsymbol{R}_k[\boldsymbol{a}_k - \boldsymbol{b}_{a_k}]_\times \delta\boldsymbol{\theta}_k \\
& -\frac{1}{2}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times \left\{ \left[\boldsymbol{I} - [\bar{\boldsymbol{\omega}}]_\times \delta t\right] \delta\boldsymbol{\theta}_k + \frac{\delta t}{2}\boldsymbol{n}_{\omega_k} + \frac{\delta t}{2}\boldsymbol{n}_{\omega_{k+1}} - \delta t \delta\boldsymbol{b}_{\omega_k} \right\} \\
& + \frac{1}{2}\boldsymbol{R}_k \boldsymbol{n}_{a_k} \\
& + \frac{1}{2}\boldsymbol{R}_{k+1} \boldsymbol{n}_{a_{k+1}} \\
& - \frac{1}{2}(\boldsymbol{R}_k + \boldsymbol{R}_{k+1})\delta\boldsymbol{b}_{a_k}
\end{aligned}
$$

4. 预积分方差计算

4.3 离散时间下的传递方程

2）$\delta\boldsymbol{\beta}_{k+1}$ 的求解

经过一系列合并同类项以后，最终的合并结果为

$$
\begin{aligned}
\delta\dot{\boldsymbol{\beta}}_k = & -\frac{1}{2}\left[\boldsymbol{R}_k[\boldsymbol{a}_k-\boldsymbol{b}_{a_k}]_\times+\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\left(\boldsymbol{I}-[\bar{\boldsymbol{\omega}}]_\times\delta t\right)\right]\delta\boldsymbol{\theta}_k \\
& -\frac{\delta t}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\boldsymbol{n}_{\omega_k} \\
& -\frac{\delta t}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\boldsymbol{n}_{\omega_{k+1}} \\
& +\frac{\delta t}{2}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\delta\boldsymbol{b}_{\omega_k} \\
& +\frac{1}{2}\boldsymbol{R}_k\boldsymbol{n}_{a_k} \\
& +\frac{1}{2}\boldsymbol{R}_{k+1}\boldsymbol{n}_{a_{k+1}} \\
& -\frac{1}{2}(\boldsymbol{R}_k+\boldsymbol{R}_{k+1})\delta\boldsymbol{b}_{a_k}
\end{aligned}
$$

4. 预积分方差计算

4.3 离散时间下的传递方程

2） $\delta\boldsymbol{\beta}_{k+1}$ 的求解

由导数形式可以得到递推形式如下

$$
\begin{aligned}
\delta\boldsymbol{\beta}_{k+1} =&\delta\boldsymbol{\beta}_k \\
&-\frac{\delta t}{2}\left[\boldsymbol{R}_k[\boldsymbol{a}_k-\boldsymbol{b}_{a_k}]_\times+\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\left(\boldsymbol{I}-[\bar{\boldsymbol{\omega}}]_\times\delta t\right)\right]\delta\boldsymbol{\theta}_k \\
&-\frac{\delta t^2}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\boldsymbol{n}_{\omega_k} \\
&-\frac{\delta t^2}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\boldsymbol{n}_{\omega_{k+1}} \\
&+\frac{\delta t^2}{2}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\delta\boldsymbol{b}_{\omega_k} \\
&+\frac{\delta t}{2}\boldsymbol{R}_k\boldsymbol{n}_{a_k} \\
&+\frac{\delta t}{2}\boldsymbol{R}_{k+1}\boldsymbol{n}_{a_{k+1}} \\
&-\frac{\delta t}{2}(\boldsymbol{R}_k+\boldsymbol{R}_{k+1})\delta\boldsymbol{b}_{a_k}
\end{aligned}
$$

4. 预积分方差计算

4.3 离散时间下的传递方程

3）$\delta\boldsymbol{\alpha}_{k+1}$ 的求解

由于连续时间下有 $\delta\dot{\boldsymbol{\alpha}}_t = \delta\boldsymbol{\beta}_t$，则离散时间下应该有

$$
\begin{aligned}
\delta\dot{\boldsymbol{\alpha}}_k =& \frac{1}{2}\delta\boldsymbol{\beta}_k + \frac{1}{2}\delta\boldsymbol{\beta}_{k+1} \\
=& \delta\boldsymbol{\beta}_k \\
& - \frac{\delta t}{4}\left[\boldsymbol{R}_k[\boldsymbol{a}_k - \boldsymbol{b}_{a_k}]_\times + \boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times\left(\boldsymbol{I} - [\bar{\boldsymbol{\omega}}]_\times\delta t\right)\right]\delta\boldsymbol{\theta}_k \\
& - \frac{\delta t^2}{8}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times\boldsymbol{n}_{\omega_k} \\
& - \frac{\delta t^2}{8}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times\boldsymbol{n}_{\omega_{k+1}} \\
& + \frac{\delta t^2}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times\delta\boldsymbol{b}_{\omega_k} \\
& + \frac{\delta t}{4}\boldsymbol{R}_k\boldsymbol{n}_{a_k} \\
& + \frac{\delta t}{4}\boldsymbol{R}_{k+1}\boldsymbol{n}_{a_{k+1}} \\
& - \frac{\delta t}{4}(\boldsymbol{R}_k + \boldsymbol{R}_{k+1})\delta\boldsymbol{b}_{a_k}
\end{aligned}
$$

4. 预积分方差计算

4.3 离散时间下的传递方程

3）$\delta\boldsymbol{\alpha}_{k+1}$ 的求解

由导数形式可以写出递推形式

$$
\begin{aligned}
\delta\boldsymbol{\alpha}_{k+1} =& \delta\boldsymbol{\alpha}_k \\
& + \delta t \delta\boldsymbol{\beta}_k \\
& - \frac{\delta t^2}{4}\left[\boldsymbol{R}_k[\boldsymbol{a}_k - \boldsymbol{b}_{a_k}]_\times + \boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times\left(\boldsymbol{I} - [\bar{\boldsymbol{\omega}}]_\times \delta t\right)\right]\delta\boldsymbol{\theta}_k \\
& - \frac{\delta t^3}{8}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times \boldsymbol{n}_{\omega_k} \\
& - \frac{\delta t^3}{8}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times \boldsymbol{n}_{\omega_{k+1}} \\
& + \frac{\delta t^3}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1} - \boldsymbol{b}_{a_k}]_\times \delta\boldsymbol{b}_{\omega_k} \\
& + \frac{\delta t^2}{4}\boldsymbol{R}_k \boldsymbol{n}_{a_k} \\
& + \frac{\delta t^2}{4}\boldsymbol{R}_{k+1} \boldsymbol{n}_{a_{k+1}} \\
& - \frac{\delta t^2}{4}(\boldsymbol{R}_k + \boldsymbol{R}_{k+1})\delta\boldsymbol{b}_{a_k}
\end{aligned}
$$

4. 预积分方差计算

4.3 离散时间下的传递方程

由以上的推导结果，便可以写出 $\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{B}_k \boldsymbol{w}_k$ 中的矩阵

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & \mathbf{f}_{12} & \mathbf{I}\delta t & -\frac{1}{4}\left(\boldsymbol{R}_k + \boldsymbol{R}_{k+1}\right)\delta t^2 & \mathbf{f}_{15} \\ \mathbf{0} & \mathbf{I} - [\bar{\boldsymbol{\omega}}]_\times \delta t & \mathbf{0} & \mathbf{0} & -\mathbf{I}\delta t \\ \mathbf{0} & \mathbf{f}_{32} & \mathbf{I} & -\frac{1}{2}\left(\boldsymbol{R}_k + \boldsymbol{R}_{k+1}\right)\delta t & \mathbf{f}_{35} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}$$

$$\mathbf{B}_k = \begin{bmatrix} \frac{1}{4}\boldsymbol{R}_k \delta t^2 & \mathbf{g}_{12} & \frac{1}{4}\boldsymbol{R}_{k+1}\delta t^2 & \mathbf{g}_{14} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{2}\mathbf{I}\delta t & \mathbf{0} & \frac{1}{2}\mathbf{I}\delta t & \mathbf{0} & \mathbf{0} \\ \frac{1}{2}\boldsymbol{R}_k \delta t & \mathbf{g}_{32} & \frac{1}{2}\boldsymbol{R}_{k+1}\delta t & \mathbf{g}_{34} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\delta t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\delta t \end{bmatrix}$$

4. 预积分方差计算

4.3 离散时间下的传递方程

上面的矩阵中，有

$$\boldsymbol{f}_{12}=-\frac{\delta t^2}{4}\left[\boldsymbol{R}_k[\boldsymbol{a}_k-\boldsymbol{b}_{a_k}]_\times+\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times(\boldsymbol{I}-[\overline{\boldsymbol{\omega}}]_\times\delta t)\right]$$

$$\boldsymbol{f}_{15}=\frac{\delta t^3}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times\delta\boldsymbol{b}_{\omega_k}$$

$$\boldsymbol{f}_{32}=-\frac{\delta t}{2}\left[\boldsymbol{R}_k[\boldsymbol{a}_k-\boldsymbol{b}_{a_k}]_\times+\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times(\boldsymbol{I}-[\overline{\boldsymbol{\omega}}]_\times\delta t)\right]$$

$$\boldsymbol{f}_{35}=\frac{\delta t^2}{2}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times$$

$$\boldsymbol{g}_{12}=-\frac{\delta t^3}{8}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times$$

$$\boldsymbol{g}_{14}=-\frac{\delta t^3}{8}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times$$

$$\boldsymbol{g}_{32}=-\frac{\delta t^2}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times$$

$$\boldsymbol{g}_{34}=-\frac{\delta t^2}{4}\boldsymbol{R}_{k+1}[\boldsymbol{a}_{k+1}-\boldsymbol{b}_{a_k}]_\times$$

预积分在优化中的使用

5. 预积分更新

回到bias变化时，预积分结果怎样重新计算的问题，再次给出它的泰勒展开形式：

$$\boldsymbol{\alpha}_{b_i b_j} = \bar{\boldsymbol{\alpha}}_{b_i b_j} + \mathbf{J}^{\alpha}_{b_i^a} \delta \mathbf{b}_i^a + \mathbf{J}^{\alpha}_{b_i^g} \delta \mathbf{b}_i^g$$

$$\boldsymbol{\beta}_{b_i b_j} = \bar{\boldsymbol{\beta}}_{b_i b_j} + \mathbf{J}^{\beta}_{b_i^a} \delta \mathbf{b}_i^a + \mathbf{J}^{\beta}_{b_i^g} \delta \mathbf{b}_i^g$$

$$\mathbf{q}_{b_i b_j} = \bar{\mathbf{q}}_{b_i b_j} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} \mathbf{J}^{q}_{b_i^g} \delta \mathbf{b}_i^g \end{bmatrix}$$

这里，雅可比没有明确的闭式解，但是在推导方差的更新时，我们得到了

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{B}_k \boldsymbol{w}_k$$

通过该递推形式，可以知道

$$\mathbf{J}_{k+1} = \mathbf{F}_k \mathbf{J}_k$$

即预积分结果的雅可比，可以通过这种迭代方式计算。

$\boldsymbol{J}_j$ 中关于 bias 的项，就是预积分泰勒展开时，各 bias 对应的雅可比。

目录

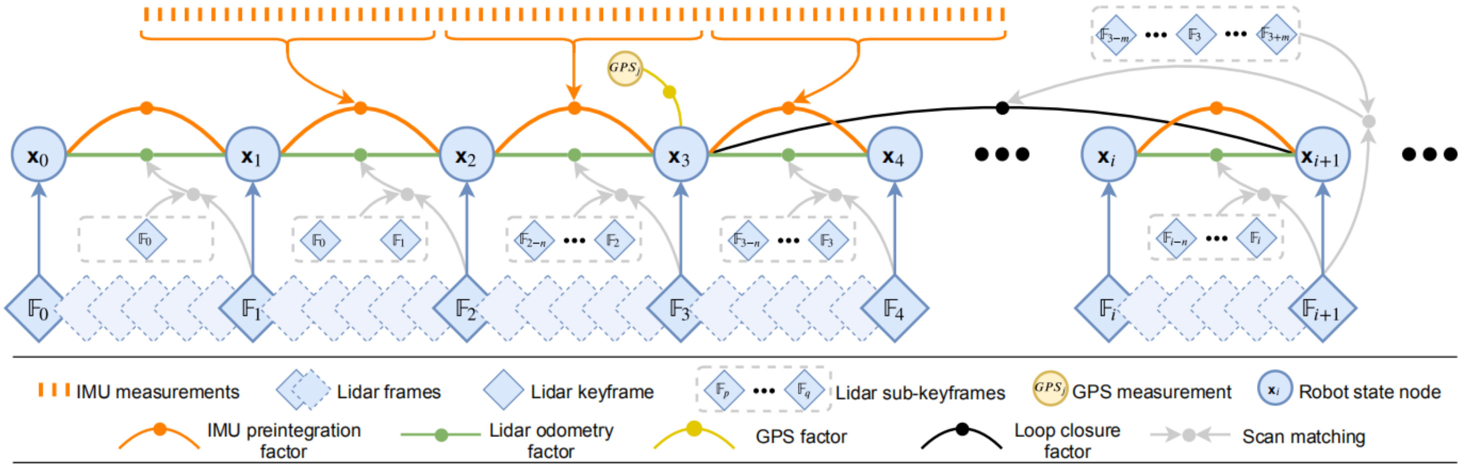
1. 基于预积分的融合方案流程
2. 预积分模型设计
3. 预积分在优化中的使用
4. 典型方案介绍
5. 融合编码器的优化方案

典型方案介绍

1. LIO-SAM介绍

论文名称：LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping

代码地址：https://github.com/TixiaoShan/LIO-SAM



典型方案介绍

1. LIO-SAM介绍

最大特点：分两步完成，先通过点云特征计算出相对位姿，再利用相对位姿、IMU预积分和GPS做融合。
相比于直接一步做紧耦合，大大提高了效率，而且实测性能也很优异。

代码讲解环节：讲解LIO-SAM代码

目录

1. 基于预积分的融合方案流程
2. 预积分模型设计
3. 预积分在优化中的使用
4. 典型方案介绍
5. 融合编码器的优化方案

融合编码器的优化方案

1. 整体思路介绍

理论上，只要是在载体系下测量，且频率比关键帧的提取频率高，就都可以做预积分。

编码器与IMU基于预积分的融合有多种方法，此处选相对简单的一种：
把它们当做一个整体的传感器来用，IMU提供角速度，编码器提供位移增量，且不考虑编码器的误差。

此时对预积分模型，输入为

角速度： $\boldsymbol{\omega}_k = \begin{bmatrix} \boldsymbol{\omega}_{xk} \\ \boldsymbol{\omega}_{yk} \\ \boldsymbol{\omega}_{zk} \end{bmatrix}$　　位移增量： $\boldsymbol{\phi}_k = \begin{bmatrix} \boldsymbol{\phi}_{xk} \\ 0 \\ 0 \end{bmatrix}$

输出中包含位置、姿态，但不包含速度。

2. 预积分模型设计

连续时间下，从 i 时刻到 j 时刻 IMU的积分结果为

$$\mathbf{p}_{wb_j} = \mathbf{p}_{wb_i} + \int_{t\in[i,j]} \mathbf{q}_{wb_t} \boldsymbol{\phi}^{b_t} \delta t$$

$$\mathbf{q}_{wb_j} = \int_{t\in[i,j]} \mathbf{q}_{wb_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2}\boldsymbol{\omega}^{b_t} \end{bmatrix} \delta t$$

把 $\mathbf{q}_{wb_t} = \mathbf{q}_{wb_i} \otimes \mathbf{q}_{b_ib_t}$ 代入上式可得

$$\mathbf{p}_{wb_j} = \mathbf{p}_{wb_i} + \mathbf{q}_{wb_i} \boxed{\int_{t\in[i,j]} \left(\mathbf{q}_{b_ib_t} \boldsymbol{\phi}^{b_t}\right) \delta t}$$

$$\mathbf{q}_{wb_j} = \mathbf{q}_{wb_i} \boxed{\int_{t\in[i,j]} \mathbf{q}_{b_ib_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2}\boldsymbol{\omega}^{b_t} \end{bmatrix} \delta t}$$

2. 预积分模型设计

为了整理公式，把积分相关的项用下面的式子代替

$$\boldsymbol{\alpha}_{b_i b_j} = \int_{t\in[i,j]} \left(\mathbf{q}_{b_i b_t} \boldsymbol{\phi}^{b_t}\right) \delta t$$

$$\mathbf{q}_{b_i b_j} = \int_{t\in[i,j]} \mathbf{q}_{b_i b_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2}\boldsymbol{\omega}^{b_t} \end{bmatrix} \delta t$$

使用中值积分方法，把连续方法改成离散形式如下

$$\boldsymbol{\omega}^{b} = \frac{1}{2}\left[\left(\boldsymbol{\omega}^{b_k} - \mathbf{b}_k^g\right) + \left(\boldsymbol{\omega}^{b_{k+1}} - \mathbf{b}_k^g\right)\right]$$

$$\boldsymbol{\phi}^{w} = \frac{1}{2}\left(\mathbf{q}_{b_i b_k}\boldsymbol{\phi}^{b_k} + \mathbf{q}_{b_i b_{k+1}}\boldsymbol{\phi}^{b_{k+1}}\right)$$

那么预积分的离散形式可以表示为

$$\boldsymbol{\alpha}_{b_i b_{k+1}} = \boldsymbol{\alpha}_{b_i b_k} + \boldsymbol{\phi}^{w}\delta t$$

$$\mathbf{q}_{b_i b_{k+1}} = \mathbf{q}_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\boldsymbol{\omega}^{b}\delta t \end{bmatrix}$$

融合编码器的优化方案

2. 预积分模型设计

此时状态更新的公式可以整理为

$$\begin{bmatrix} \mathbf{p}_{wb_j} \\ \mathbf{q}_{wb_j} \\ \mathbf{b}_j^g \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{wb_i} + \mathbf{q}_{wb_i}\boldsymbol{\alpha}_{b_ib_j} \\ \mathbf{q}_{wb_i}\mathbf{q}_{b_ib_j} \\ \mathbf{b}_i^g \end{bmatrix}$$

残差形式可以写为

$$\begin{bmatrix} \mathbf{r}_p \\ \mathbf{r}_q \\ \mathbf{r}_{bg} \end{bmatrix} = \begin{bmatrix} \mathbf{q}_{wb_i}^* \left(\mathbf{p}_{wb_j} - \mathbf{p}_{wb_i}\right) - \boldsymbol{\alpha}_{b_ib_j} \\ 2\left[\mathbf{q}_{b_ib_j}^* \otimes \left(\mathbf{q}_{wb_i}^* \otimes \mathbf{q}_{wb_j}\right)\right]_{xyz} \\ \mathbf{b}_j^g - \mathbf{b}_i^g \end{bmatrix}$$

其余部分(方差递推、残差对状态量雅可比、bias更新等)的推导，留作作业(仅优秀标准需要做)。

作业

在IMU的预积分中，课程只提供了残差对部分变量的雅可比，请推导残差对其它变量的雅可比，并在建图流程的代码中补全基于IMU预积分的融合方法中的待填内容，随后与不加IMU融合时的效果进行对比。

备注：

1）对比是全方位的，既包括轨迹精度的对比，也包括地图质量的对比(因为IMU会增加估计的平滑性)；

2）由于数据集的老问题，部分指标可能与预期不一致，且地图质量无法量化，因此给出自己的分析即可。

评价标准：

及格：公式推导正确，补全代码之后功能正常；

良好：在及格基础上，实现和不加IMU时的效果对比和分析；

优秀：在良好的基础上，完成融合编码器时预积分公式的推导(方差递推、残差对状态量雅可比、bias更新等）。

作业

附加题(不参与考核)：

基于预积分的融合编码器的方法，近年来层出不穷，众多论文中给出了众多方法，请调研相关文献，梳理不同的推导思路，并从原理上对比各种方案的不同于优缺点。

如有余力，在优秀作业的基础上，实现不同方案，并对比效果。

深蓝学院
shenlanxueyuan.com

感谢聆听!

Thanks for Listening