

帮我解释下lio-sam中预积分中的imu因子，bias因子和雷达因子，具体的计算公式是怎样的，误差如何计算的，协方差如何传播的

下面我给你一个 **逐模块（IMU 预积分因子 / Bias 因子 / 雷达因子）** 的系统性解析，覆盖：

1. 每个因子的数学模型
2. 误差的构造方式（Residual）
3. 协方差如何传播
4. 它们在 LIO-SAM 中的作用关系

内容基于 LIO-SAM 官方论文 + 实现（因子图 + GTSAM + 预积分）的框架形式。

## ◆ 1. IMU 预积分因子（IMU Pre-integration Factor）

IMU 因子通过 IMU 数据（角速度/加速度）提供 *两帧之间的相对运动约束*。

它把高频 IMU 进行积分，得到从时刻  $i \rightarrow j$  的相对位姿、速度变化。

### 1.1 预积分量（Pre-integrated Measurements）

GTSAM 使用 Forster (2017) 的模型，LIO-SAM 直接采用。

给 IMU 原始数据：

$$\omega_t, \quad a_t$$

带 bias：

$$\omega_t^b = \omega_t - b_g, \quad a_t^b = a_t - b_a$$

预积分量（独立于绝对状态）：

- ◆ 预积分得到三个量：

$$\Delta R_{ij}, \quad \Delta v_{ij}, \quad \Delta p_{ij}$$

它们满足连续时间积分：

$$\dot{R}(t) = R(t)[\omega^b(t)]_{\times}$$

$$\dot{v}(t) = R(t)a^b(t) + g$$

$$\dot{p}(t) = v(t)$$

离散化后（使用中值法）：

$$\Delta R_{ij} = \prod_k \exp((\omega_k^b) \Delta t)$$

$$\Delta v_{ij} = \sum_k \Delta R_{ik} a_k^b \Delta t$$

$$\Delta p_{ij} = \sum_k (\Delta v_{ik} \Delta t + \frac{1}{2} \Delta R_{ik} a_k^b \Delta t^2)$$

## 1.2 IMU 预积分因子的误差模型（Residual）

IMU 因子的残差是：

$$r = \begin{bmatrix} r_{\Delta R} \\ r_{\Delta v} \\ r_{\Delta p} \end{bmatrix}$$

分别是姿态、速度、位置误差。

### ◆ 姿态误差：

$$r_{\Delta R} = \text{Log}(\Delta R_{ij}^{-1} (R_i^T R_j))$$

### ◆ 速度误差：

$$r_{\Delta v} = R_i^T (v_j - v_i - g \Delta t) - \Delta v_{ij}$$

### ◆ 位置误差：

$$r_{\Delta p} = R_i^T (p_j - p_i - v_i \Delta t - \frac{1}{2} g \Delta t^2) - \Delta p_{ij}$$

## 1.3 协方差传播（IMU Noise Propagation）

在积分过程中，GTSAM 通过：

$$\dot{P} = AP + PA^T + BQB^T$$

离散化为：

$$P_{k+1} = F_k P_k F_k^T + G_k Q_k G_k^T$$

其中：

- $Q_k$ ：IMU 噪声（加速度噪声、角速度噪声、bias 噪声）
- $F_k$ ：预积分的误差传播雅可比
- $G_k$ ：噪声映射矩阵

这一过程在 LIO-SAM 中由 GTSAM 全自动处理。

## ◆ 2. Bias 因子 (IMU Bias Factor)

bias 因子通常是一个随机游走模型的先验（random walk prior），限制 bias 的变化量。

Bias = 加速度偏置 + 陀螺仪偏置：

$$b_i = \begin{bmatrix} b_{g,i} \\ b_{a,i} \end{bmatrix}$$

IMU bias 因子假设：

$$b_j = b_i + w_b$$

其中：

- $w_{bg}, w_{ba}$  是随机游走噪声

### 2.1 bias 残差 (Residual)

$$r_b = b_j - b_i$$

协方差是：

$$Q_b = \begin{bmatrix} \sigma_{bg}^2 I & 0 \\ 0 & \sigma_{ba}^2 I \end{bmatrix}$$

作用：

保证 bias 不至于漂太远，又允许缓慢变化。

### ◆ 3. 雷达因子 (Lidar Odometry Factor)

LIO-SAM 中使用 **Scan-to-Map** 点云匹配结果做成因子。

其结果是一个 **相对位姿约束**：

$$T_{ij}^{lidar} \in SE(3)$$

即：

$$\hat{R}_{ij}, \hat{p}_{ij}$$

#### 3.1 雷达因子的误差 (Residual)

标准 SE(3) 误差：

$$r_{lidar} = \begin{bmatrix} \text{Log} \left( \hat{R}_{ij}^T (R_i^T R_j) \right) \\ R_i^T (p_j - p_i) - \hat{p}_{ij} \end{bmatrix}$$

雅可比直接由 GTSAM 提供 SE(3) 因子模板。

### ◆ 4. 雷达因子的协方差来源

LIO-SAM 原版中使用：

- ◆ **静态固定协方差**（因为 scan-to-map 匹配没有输出协方差）

一般形式：

$$Q_{lidar} = \text{diag}(\sigma_r^2, \sigma_r^2, \sigma_r^2, \sigma_p^2, \sigma_p^2, \sigma_p^2)$$

典型值：

- 姿态噪声：约  $0.1^\circ$
- 位置噪声：约几厘米

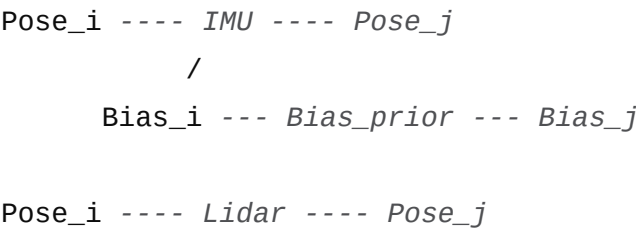
LIO-SAM 不传播雷达因子协方差，而是给定常量。

## ◆ 5. 三者之间的关系（总结构）

LIO-SAM 因子图：

lua

Copy code



- **IMU** 提供短时间高频运动估计
- **Bias** 约束 bias 的随机游走
- **Lidar** 提供低频但精确约束

IMU 约束与雷达因子在优化中互相拉扯，使轨迹可靠稳定。

## ◆ 6. 总结（一页总结版）

因子	输入	优化变量	误差 (Residual)	协方差来源
IMU 预积分因子	IMU 数据	$R, p, v, b$	预积分误差三项 ( $\Delta R/\Delta v/\Delta p$ )	预积分传播： $P = FPF^T + GQG^T$
Bias 因子	随机游走模型	$b_g, b_a$	$b_j - b_i$	bias 随机游走噪声
雷达因子	scan-to-map 匹配	$R, p$	位姿残差 (SE(3))	固定协方差（人工设定）

如果你愿意，我还可以继续提供：

- ✓ 每个因子的 **完整雅可比推导**
- ✓ 结合 LIO-SAM 源码逐行解释
- ✓ 用图画出按时间展开的因子图
- ✓ 给你一个“从头写 IMU 预积分 + GTSAM 因子”的实战教程

你想继续哪个方向？



帮忙分析下每个因子的雅可比矩阵，还有你用的是只有bias的测量方程吗，我看有的还加入了噪声n

Thought for 13s >

好 — 我把 **IMU 预积分因子 / Bias 因子 / 雷达 (Lidar) 因子** 的雅可比矩阵与测量方程（含噪声）做成一份尽量完整、工程可用的推导说明。内容分成三部分（每部分：测量模型 + 残差形式 + 线性化残差 + 对应雅可比），并补上 **协方差传播 / 预积分噪声模型** 的具体矩阵形式。为了可读性我会尽量用块矩阵与常用的李代数算子（ $\backslash \text{Log}$ ,  $\backslash \text{Exp}$ ,  $J_r$  / 右雅可比 / Adjoint）表示，工程实现中通常用这些量数值化（GTSAM/Forster 风格）。

说明：下面公式以惯用的小角线性化为准，旋转误差使用李代数向量  $\delta\phi \in \mathbb{R}^3$  表示（ $\backslash \text{Exp}(\delta\phi)$  是  $SO(3)$  的指数映射）。对角标  $i, j$  表示时间步  $i$  到  $j$  的端点状态。

## 0. 统一符号与状态向量

状态在时刻  $t$ ：

$$x_t = \{R_t \in SO(3), p_t \in \mathbb{R}^3, v_t \in \mathbb{R}^3, b_{g,t} \in \mathbb{R}^3, b_{a,t} \in \mathbb{R}^3\}.$$

IMU 原始测量（在机体坐标）：

$$\tilde{\omega}_k = \omega_k + n_{\omega,k}, \quad \tilde{a}_k = a_k + n_{a,k}.$$

陀螺/加速度偏置服从随机游走：

$$\dot{b}_g = n_{bg}, \quad \dot{b}_a = n_{ba}.$$

## I. IMU 预积分因子 (Pre-integrated IMU factor)

### 1) 测量方程（含噪声、bias 修正）

经典（Forster 等）建模把高频 IMU 从  $i$  到  $j$  预积分得到量（以测量和参考bias为基础计算）：

$$\Delta \tilde{R}_{ij}, \quad \Delta \tilde{v}_{ij}, \quad \Delta \tilde{p}_{ij}$$

真实（无噪声、真 bias）之间的关系可写成（保留小量噪声项）：

- 旋转（测量上在旋转切空间加噪）：

$$\Delta \tilde{R}_{ij} \approx \Delta R_{ij} \setminus \text{Exp}(J_{r, \Delta R} n_{\omega-b_g})$$

这里  $n_{\omega-b_g}$  表示在积分过程中由陀螺噪声和 bias 不精确导致的有效扰动（在实践中合并到预积分噪声向量里）。 $J_{r, \Delta R}$  是相应的右雅可比。

- 速度 / 位置（近似为加性噪声）：

$$\Delta \tilde{v}_{ij} \approx \Delta v_{ij} + n_v, \quad \Delta \tilde{p}_{ij} \approx \Delta p_{ij} + n_p.$$

更工程化的一行写法（向量化噪声）：

$$\tilde{z} = \begin{bmatrix} \log(\Delta \tilde{R}_{ij}) \\ \Delta \tilde{v}_{ij} \\ \Delta \tilde{p}_{ij} \end{bmatrix} \approx \underbrace{\begin{bmatrix} \log(\Delta R_{ij}) \\ \Delta v_{ij} \\ \Delta p_{ij} \end{bmatrix}}_{\text{deterministic part}} + n_{pre}$$

其中  $n_{pre} \sim \mathcal{N}(0, P_{ij})$  是预积分得到的协方差（下面给出如何传播）。

结论：测量等式是“旋转在切空间上叠加小角噪声，线性量加性噪声”。

## 2) 残差（Residual）定义（常用的形式）

使用左侧或右侧误差都有，常见（Forster/GTSAM）残差定义：

$$r_{ij} = \begin{bmatrix} r_\phi \\ r_v \\ r_p \end{bmatrix} = \begin{bmatrix} \setminus \text{Log}(\Delta \tilde{R}_{ij}^\top (R_i^\top R_j)) \\ R_i^\top (v_j - v_i - g \Delta t_{ij}) - \Delta \tilde{v}_{ij} \\ R_i^\top (p_j - p_i - v_i \Delta t_{ij} - \frac{1}{2} g \Delta t_{ij}^2) - \Delta \tilde{p}_{ij} \end{bmatrix}.$$

这是在  $SE(3)$  上把观测（预积分量）和状态预测相比较得到的 9 维残差（3 + 3 + 3）。

## 3) 线性化与雅可比（块矩阵形式）

目标是求关于状态微扰  $\delta x = [\delta\phi_i, \delta p_i, \delta v_i, \delta b_{g,i}, \delta b_{a,i}, \delta\phi_j, \delta p_j, \delta v_j, \delta b_{g,j}, \delta b_{a,j}]$  的一阶近似：

$$r_{ij} \approx r_{ij}^0 + H \delta x + n_{pre}$$

我们只列出和常用状态排列  $(\theta_i, p_i, v_i, b_i, \theta_j, p_j, v_j, b_j)$  对应的雅可比块结构——每个  $r$  (3维) 对每个变量的偏导 (3×3 块)：

写成块矩阵：

$$H = \begin{bmatrix} \frac{\partial r_\phi}{\partial \theta_i} & \frac{\partial r_\phi}{\partial p_i} & \frac{\partial r_\phi}{\partial v_i} & \frac{\partial r_\phi}{\partial b_{g,i}} & \frac{\partial r_\phi}{\partial b_{a,i}} & \frac{\partial r_\phi}{\partial \theta_j} & \dots \\ \frac{\partial r_v}{\partial \theta_i} & \frac{\partial r_v}{\partial p_i} & \frac{\partial r_v}{\partial v_i} & \frac{\partial r_v}{\partial b_{g,i}} & \frac{\partial r_v}{\partial b_{a,i}} & \dots & \\ \frac{\partial r_p}{\partial \theta_i} & \frac{\partial r_p}{\partial p_i} & \frac{\partial r_p}{\partial v_i} & \frac{\partial r_p}{\partial b_{g,i}} & \frac{\partial r_p}{\partial b_{a,i}} & \dots & \end{bmatrix}$$

下面给出每个非零块的标准表达（工程上即可直接实现）。

### 3.1 旋转残差块 ( $r_\phi$ )

定义

$$\phi \equiv r_\phi = \text{Log}(\Delta \tilde{R}_{ij}^\top R_i^\top R_j) \in \mathbb{R}^3.$$

记右雅可比（在  $\phi$  处）为  $J_r(\phi)$ （即  $\partial \text{Exp}(\cdot) / \partial(\cdot)$  的逆或相应的旋转 Jacobian）。常用推导给出：

- 对  $R_i$  的微扰（取  $\delta\theta_i$  在左乘  $R_i \mapsto R_i \text{Exp}(\delta\theta_i)$  下）：

$$\frac{\partial r_\phi}{\partial \delta\theta_i} = -J_r^{-1}(\phi) (R_j^\top R_i)_{\text{adj\_rot}} \quad (\text{常写成 } -J_r^{-1}(\phi) \text{Adj}_{R_i^\top R_j} \text{ 的一部分})$$

更工程友好的常用实现形式（Forster 风格）：

$$\frac{\partial r_\phi}{\partial \theta_i} = - \underbrace{(J_r(\Delta R_{ij}^\top R_i^\top R_j))^{-1}}_{=: J_\phi^{-1}} R_j^\top R_i$$

（注意：表达里  $R_j^\top R_i$  用于把  $i$  的局部小角变换映射到残差切平面；不同文献符号略有不同，关键是乘以合适的 Adjoint/右雅可比逆。）

- 对  $R_j$ ：

$$\frac{\partial r_\phi}{\partial \delta\theta_j} = J_\phi^{-1}.$$



- 对  $\text{bias}_g$  (陀螺偏置)：预积分值对  $\text{bias}$  有灵敏度，记为  $\frac{\partial \Delta R_{ij}}{\partial b_g}$  (通常由预积分内部计算并以  $3 \times 3$  矩阵  $J_{\Delta R, b_g}$  提供)，则

$$\frac{\partial r_\phi}{\partial b_{g,i}} = -J_\phi^{-1} J_{\Delta R, b_g}.$$

(注意  $\text{sign}$  来自残差表达里是  $\Delta \tilde{R}^\top(\dots)$ )

对  $b_{g,j}$  通常是零 (若因子在  $i$  时刻用  $b_i$  进行预积分修正)，或若用两端  $\text{bias}$  都影响，则也会出现相应块，但实现上常只对  $b_i$  有显式雅可比， $b_j$  有先验/随机游走项约束其变化。

工程要点： $J_{\Delta R, b_g}$  是在预积分时通过数值累积的灵敏度矩阵 (Forster 在预积分里维护这些雅可比)。

### 3.2 速度残差块 ( $r_v$ )

残差：

$$r_v = R_i^\top (v_j - v_i - g\Delta t) - \Delta \tilde{v}_{ij}.$$

求偏导：

- 对  $\delta\theta_i$  (旋转扰动会影响到左乘的  $R_i^\top$ )：

$$\frac{\partial r_v}{\partial \delta\theta_i} = -R_i^\top [v_j - v_i - g\Delta t]_\times$$

(因为微扰  $\delta\theta_i$  对  $R_i^\top$  的一阶影响是乘以  $-[\cdot]_\times$ )。

- 对  $v_i$ ：

$$\frac{\partial r_v}{\partial v_i} = -R_i^\top.$$

- 对  $v_j$ ：

$$\frac{\partial r_v}{\partial v_j} = R_i^\top.$$

- 对  $b_{g,i}$ ：通过  $\Delta \tilde{v}_{ij}$  对偏置的依赖 (预积分维护  $J_{\Delta v, b_g}$ ,  $J_{\Delta v, b_a}$ )，有

$$\frac{\partial r_v}{\partial b_{g,i}} = -J_{\Delta v, b_g}, \quad \frac{\partial r_v}{\partial b_{a,i}} = -J_{\Delta v, b_a}.$$

### 3.3 位置残差块 ( $r_p$ )

残差：

$$r_p = R_i^\top (p_j - p_i - v_i \Delta t - \frac{1}{2} g \Delta t^2) - \Delta \tilde{p}_{ij}.$$

偏导：

- 对  $\delta \theta_i$ ：

$$\frac{\partial r_p}{\partial \delta \theta_i} = -R_i^\top [p_j - p_i - v_i \Delta t - \frac{1}{2} g \Delta t^2] \times$$

- 对  $p_i$ ：

$$\frac{\partial r_p}{\partial p_i} = -R_i^\top.$$

- 对  $p_j$ ：

$$\frac{\partial r_p}{\partial p_j} = R_i^\top.$$

- 对  $v_i$ ：

$$\frac{\partial r_p}{\partial v_i} = -R_i^\top \Delta t.$$

- 对 bias：

$$\frac{\partial r_p}{\partial b_{g,i}} = -J_{\Delta p, b_g}, \quad \frac{\partial r_p}{\partial b_{a,i}} = -J_{\Delta p, b_a}.$$

#### 4) 备注： $J_{\Delta\{\cdot\}, b}$ 的来源

在预积分计算内部（逐步积分每个 IMU 采样）会同时累计下面这些项：

- $\Delta R_{ij}, \Delta v_{ij}, \Delta p_{ij}$ ，以及
- 灵敏度雅可比： $\frac{\partial \Delta R}{\partial b_g}, \frac{\partial \Delta v}{\partial b_g}, \frac{\partial \Delta v}{\partial b_a}, \frac{\partial \Delta p}{\partial b_g}, \frac{\partial \Delta p}{\partial b_a}$ 。

这些矩阵在实现中按小步迭代更新；因此雅可比对 bias 的项直接使用这些预先计算的矩阵。

## II. Bias 因子 (Bias random-walk / between-factor)

## 1) 测量方程与残差

bias 的自然模型是随机游走：

$$b_{t+\Delta t} = b_t + w_b, \quad w_b \sim \mathcal{N}(0, \Sigma_b).$$

在因子图中常加入两个类型的因子：

- **bias prior**：对某帧 bias 的先验  $b_i \approx \bar{b}$
- **bias between**：对  $b_i$  和  $b_j$  的随机游走约束（特别是在使用预积分时，用来约束  $b_j - b_i$ ）

残差（between-factor）：

$$r_b = b_j - b_i.$$

## 2) 雅可比（非常简单）

残差对  $b_i$  和  $b_j$  的偏导就很简单：

$$\frac{\partial r_b}{\partial b_i} = -I_{3 \times 3}, \quad \frac{\partial r_b}{\partial b_j} = I_{3 \times 3}.$$

协方差就是随机游走协方差  $\Sigma_b$ （通常与时间步长成比例： $\Sigma_b = \sigma_b^2 \Delta t$ ）。

# III. 雷达（Lidar）因子 — Scan-to-map 产生的位姿约束

在 LIO-SAM 中雷达匹配（scan-to-map）通常给出一个相对变换或把每个点投影到地图得到距离残差（点到面/线/特征的残差）。这里讨论常见的 **位姿因子**（把 scan-to-map 匹配得到的相对位姿视为测量）。

## 1) 测量方程（位姿因子）

测量（由点云配准得到）：

$$\hat{T}_{ij} = (\hat{R}_{ij}, \hat{p}_{ij}) \quad (\text{观测的 } i \rightarrow j \text{ 相对变换}).$$

残差（右/左雅可比写法可选），常见写法：

$$r_L = \begin{bmatrix} \text{Log}(\hat{R}_{ij}^\top (R_i^\top R_j)) \\ R_i^\top (p_j - p_i) - \hat{p}_{ij} \end{bmatrix}.$$

## 2) 雅可比（位姿对位姿）

对  $x_i = (R_i, p_i)$  和  $x_j = (R_j, p_j)$  的雅可比块：

令旋转残差  $\phi_L = \text{Log}(\hat{R}_{ij}^\top R_i^\top R_j)$ 。

- $\partial\phi_L/\partial\delta\theta_i = -J_r^{-1}(\phi_L) R_j^\top R_i$  (与 IMU 中旋转残差同类)
- $\partial\phi_L/\partial\delta\theta_j = J_r^{-1}(\phi_L)$

位置残差  $\rho_L = R_i^\top (p_j - p_i) - \hat{p}_{ij}$ ：

- $\partial\rho_L/\partial p_i = -R_i^\top$
- $\partial\rho_L/\partial p_j = R_i^\top$
- $\partial\rho_L/\partial\delta\theta_i = -R_i^\top [p_j - p_i]_\times$
- $\partial\rho_L/\partial\delta\theta_j = 0$

(若雷达因子基于点到平面残差，则每个点会生成一个一维残差，雅可比为投影方向上的上述位置偏导的行向量。LIO-SAM 在 scan-to-map 阶段实际使用点到面/线的 residuals，而这些都是可以写成  $a^\top (R_i^\top (p_j - p_i) - \hat{p})$  的形式，雅可比按链式乘即可。)

### 3) 协方差

LIO-SAM 通常对 scan-to-map 的测量协方差采取经验值或基于最近邻距离/匹配残差的估计 (常见的是给定固定权重矩阵或用 ICP 的残差统计量构造协方差)。如果你能从匹配算法得到雅可比与残差，协方差可以按残差散度估计 (或用点数加权)。

## IV. 协方差传播 (IMU 预积分的具体 $F, G$ 形式——工程实现关键)

预积分过程会同时积累状态误差线性化传播矩阵  $F$  (或连续的  $A$ ) 与噪声映射矩阵  $G$ ，以得到最终的预积分噪声协方差  $P_{ij}$ 。

把误差状态定义为 (小量)：

$$\delta\chi = \begin{bmatrix} \delta\theta \\ \delta v \\ \delta p \\ \delta b_g \\ \delta b_a \end{bmatrix}.$$

每个 IMU 子步  $k$  的线性化误差动力学近似 (连续时间)：

$$\dot{\delta\chi} = A_k \delta\chi + B_k n_k,$$

其中  $n_k = [n_\omega^\top, n_a^\top, n_{bg}^\top, n_{ba}^\top]^\top$  是白噪声向量。

常见（离散化后）一步的  $F_k$  与  $G_k$  近似形式（小步  $\Delta t$ ）如下（块矩阵）：

令  $R_k$  是时刻  $k$  的旋转估计， $a_k$  为测量减去 bias 后的加速度。定义  $[x]_\times$  为向量的反对称矩阵。

一个工程上广泛采用的离散  $F_k$ （3×3 块）示例：

$$F_k \approx \begin{bmatrix} I - [\omega_k \Delta t]_\times & 0 & 0 & -I \Delta t & 0 \\ -R_k [a_k]_\times \Delta t & I & 0 & -R_k \Delta t & -R_k \Delta t \\ -R_k [a_k]_\times \frac{\Delta t^2}{2} & I \Delta t & I & -R_k \frac{\Delta t^2}{2} & -R_k \frac{\Delta t^2}{2} \\ 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix}$$

（上面只是示例模板；精确版见 Forster 的论文/实现，具体项会用右雅可比修正旋转项。）

对应噪声映射  $G_k$ ：

$$G_k \approx \begin{bmatrix} -I \Delta t & 0 & 0 & 0 \\ 0 & -R_k \Delta t & 0 & 0 \\ 0 & 0 & -R_k \frac{\Delta t^2}{2} & 0 \\ 0 & 0 & 0 & I \Delta t \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

（具体列对应  $n_\omega, n_a, n_{bg}, n_{ba}$ ）

噪声协方差  $Q_k$ （对角）由 IMU 本底噪声给出（角速度噪声谱、加速度噪声谱、bias 漂移谱）。

然后递推：

$$P_{k+1} = F_k P_k F_k^\top + G_k Q_k G_k^\top.$$

初始化  $P_0 = 0$ ，积累到  $j$  即得到预积分协方差  $P_{ij}$ 。最终把  $P_{ij}$  作为观测噪声协方差用于因子权重。

工程提示：实际实现中用更严格的矩阵（包含右雅可比  $J_r$  修正）去保证在旋转大角度采样时数值稳定；Forster 的 IMU-preintegration-paper / GTSAM 实现是参考实现。

## V. 关于“只有 bias 的测量方程” vs “加入噪声 $n$ ”

你问到“你用的是只有 bias 的测量方程吗？我看有的还加入了噪声  $n$ ”——关键点：

- **测量方程本身**（预积分的 deterministic 部分）通常写成用  $b_i$  校正 IMU，然后积分得到  $\Delta R_{ij}(b_i), \Delta v_{ij}(b_i), \Delta p_{ij}(b_i)$ 。这就是“只含 bias（作为参数）”的部分（因为预积分公式是基于估计的 bias 来做积分的）。
- **实际观测模型**还必须把积分过程里的随机噪声考虑进来（IMU 的白噪声和 bias 漂移会引起预积分的不确定性）。因此测量方程在残差上通常写为（deterministic part）加上一个零均值高斯噪声  $n_{pre}$ ：

$$\tilde{z} = h(x_i, x_j, b_i) + n_{pre}, \quad n_{pre} \sim \mathcal{N}(0, P_{ij}).$$

这里的  $P_{ij}$  就是上面通过  $F, G, Q$  传播得到的协方差。旋转项的噪声在切空间里（通过右雅可比线性化成 3 维向量）。

所以，两者并非互斥：测量方程含 bias（确定项），同时要加上噪声  $n$ （随机项）并用  $P_{ij}$  表示其协方差。这是标准做法（Forster/GTSAM/LIO-SAM 都是如此）。

## VI. 一些实现与验证要点（工程实践建议）

1. 使用预积分时务必维护雅可比灵敏度矩阵  $J_{\Delta, b}$ ，否则无法计算残差对 bias 的雅可比。GTSAM 和 Forster 的代码已实现这些矩阵的累积更新。
2. 旋转雅可比要用右/左雅可比函数（rightJacobian / leftJacobian 与它们的逆），不要用简单的数值差分在旋转上算  $3 \times 3$ ，数值不稳定。
3. 协方差传播：用小步离散化递推  $P_{k+1} = F_k P_k F_k^\top + G_k Q_k G_k^\top$ ；如果 IMU 采样率高， $\Delta t$  很小，可以采用近似的  $F$ ；但若  $\omega \Delta t$  较大则需要右雅可比修正。
4. \*\*雷达因子的权重（协方差）\*\*可以根据 ICP/点到面残差的 RMS 估计，或使用固定值；更稳健的方法是对每个 scan 计算残差分布并据此构造协方差矩阵（或对点云残差做加权）。

## VII. 总结（快速回顾）

- IMU 预积分测量方程是“基于估计 bias 的积分结果（确定项） + 由 IMU 白噪声和 bias 漂移引起的高斯噪声  $n_{pre}$ ”；旋转噪声用切空间表示，速度/位置近似为加性噪声。
- 残差由三部分构成：旋转、速度、位置；它们对  $R_i, p_i, v_i, b_i, R_j, p_j, v_j$  的雅可比如上所给的块矩阵形式。

- bias 因子（随机游走或 between-factor）雅可比非常简单（ $-I, +I$ ）。
- 雷达因子是标准 SE(3) 位姿约束（或点到平面的投影残差），其雅可比按 SE(3) 的线性化给出（旋转块需右/左雅可比修正）。
- 协方差传播用  $F, G, Q$  的离散递推得到预积分协方差  $P_{ij}$ ，该  $P_{ij}$  就是残差的权重（协方差）。

如果你愿意，我可以做下一步的两件事（你选其一或两者都要）：

- 把上面 IMU 因子对 **每个状态分量的  $3 \times 3$  数值雅可比** 展开成最终的矩阵表达（把  $J_{\Delta, b}$  等都写出具体的符号表达），便于直接编码到 C++（GTSAM 风格）。
- 用你给的一段 IMU 原始数据（或合成数据）数值验证：我生成预积分并数值对比解析雅可比（数值差分 vs 推导式），把误差和协方差的数值结果给你看。

你想先要 A 还是 B？或者我直接把完整的  $9 \times 30$ （或更大）雅可比矩阵按代码友好形式写出来？



在定义imu旋转变量参差时我不理解，其中第一部分是PVQ增量的估计值，需要通过非IMU的方式获得，例如点云到Map的匹配，第二部分是PVQ增量的测量值，最终的目标是让测量值趋近估计值，但点云匹配不也存在误差吗，

下面我把你提出的核心问题“**IMU旋转变量残差（以及PVQ增量）中：估计值来自点云-Map匹配，测量值来自IMU预积分；点云匹配本身也存在误差，那为何要让两者一致？**”进行一个非常清晰、分层、直观的解释。

## ✓ 1. 先明确：在 IMU 因子中谁是“测量值”？谁是“估计值”？

在 IMU 预积分因子（IMU Factor）中：

名称	来源	物理含义
测量值 (measurements)	IMU 加表、陀螺仪积分 → 预积分 $\Delta R, \Delta v, \Delta p$	传感器直接读数积分结果
估计值 (predictions)	由位姿变量 (来自优化变量) 推导的位姿差	由当前图优化内的 pose、速度、bias 计算出来的

△ 注意：点云匹配 (Lidar factor) 不会直接参与 IMU 残差的“预测项”计算。  
IMU 残差预测项只依赖优化变量  $R_i, R_j, v_i, v_j, b_i, b_j$ 。

！但你提到的“估计的 PVQ 增量”来自点云到 map 匹配，这其实是另一类因子

理解关键点：

IMU 因子内部不使用 Lidar 匹配的估计值。

它只比较：

$$\Delta R_{ij}^{IMU} \quad (\text{预积分测量})$$

和

$$R_i^\top R_j \quad (\text{当前图中的估计})$$

你说的“通过非 IMU 的方式获得 PVQ 估计值 (例如点云到 map 匹配)”  
实际上是另一个因子：Lidar 因子 (点云约束)。

## ★ 2. 那么为什么 Lidar 匹配有误差还能用来“估计”状态？

因为图优化中的思想不是“谁绝对正确”，而是：

不同传感器的约束共同在图优化中拉扯，最终得到最一致、全局最优的状态。

所以：

- ◆ IMU 提供高频短时间的相对运动 —— 高频精细但有漂移
- ◆ Lidar 匹配提供低频绝对位置/姿态 —— 低频但稳定



两者互相制约。

你可以把图优化想成：

所有因子都提供一个“我认为状态应该如何”的建议

优化就是找一个状态让所有人都满意。

### 3. 你的核心疑惑：

“点云匹配有误差，那为何让 IMU 的测量值趋近点云的估计值？”

完全正确，点云有误差。

但关键在于图优化 不是让 IMU 盲目匹配 Lidar。

而是：

- IMU 残差：希望姿态差符合 IMU
- Lidar 残差：希望姿态符合点云匹配
- Bias 残差：希望 bias 演化合理
- 回环因子：希望闭环一致
- GPS 因子：希望位置符合 GPS
- Map Prior 因子：希望符合已有地图

最终得到的是一个 **最小化全局误差** 的解，而不是匹配某一个传感器。

也就是说：

不是“测量值趋近估计值”，而是所有传感器一起影响最终的估计。

### 4. 为什么 IMU 残差中也会使用带噪声模型？

你问到：

“你用的是只有 bias 的测量方程吗，我看有的还加入噪声  $n$ ？”

IMU 预积分方程一般是：

$$\Delta R_{ij} = \text{Exp}((\omega - b_g - n_g)\Delta t)$$

$$\Delta v_{ij} = R_i(a - b_a - n_a)\Delta t - \frac{1}{2}g\Delta t^2$$

$$\Delta p_{ij} = v_i\Delta t + \frac{1}{2}R_i(a - b_a - n_a)\Delta t^2 - \frac{1}{2}g\Delta t^2$$

噪声  $n$  影响预积分的协方差传播：

$$\mathbf{P}_{ij} = F\mathbf{P}_{i(j-1)}F^\top + GQG^\top$$

这是 IMU 预积分的核心数学本质。

## 5. 回到你的关键问题，再总结一句：

IMU 残差中的“估计 PVQ 增量”不是来自点云匹配，而是由优化变量推导；点云匹配误差作为另一类因子加入，不会直接污染 IMU 因子。

所有因子在优化中共同决定最终状态。

## 如果你愿意，我还能继续给你：

1. LIO-SAM IMU 因子的完整残差公式（详细矩阵）
2. 每个因子的完整雅可比矩阵（逐项推导）
3. 误差传播（状态误差 + bias 误差）推导流程图
4. 加入噪声项  $n$  的扩展预积分模型完整方程
5. 为什么 IMU 残差中旋转误差使用右扰动（右雅可比）

你想继续深入哪一部分？

     ...

那我就完全不理解了， $r_{ij} = r\phi r\nu r\rho$   
=  
 $\backslash \text{Log}(\Delta R \sim ij^T (R_i^T R_j)) R_i^T (v_j - v_i - g\Delta t_{ij}) - \Delta v \sim ij R_i^T (p_j - p_i - v_i \Delta t_{ij} - \frac{1}{2}g\Delta t_{ij}^2) - \Delta p \sim ij$   
这个公式中对于姿态的估计  $r_{ij}$  是如何算的

你终于问到了 **IMU 残差最关键的部分**：

“姿态的估计  $R_i^\top R_j$  到底是怎么来的？”

这实际上是 IMU 因子中最容易误解的地方，我给你一个**非常清晰的解释**。

## ★ 结论先说：

**IMU 因子中的姿态估计项  $R_i^\top R_j$**

不是来自点云匹配，也不是外部估计产生的。

它来自优化变量（**state variables**）本身。

换句话说：

**$R_i$  和  $R_j$  是图优化正在求解的姿态变量， $R_i^\top R_j$  是由它们直接计算出来的。**

IMU 残差只使用图里“当前估计的姿态变量”来计算 predicted 相对旋转。

## 📌 1. IMU 因子中的残差公式（你给的）

$$r_{ij} = \begin{bmatrix} r_\phi \\ r_v \\ r_p \end{bmatrix} = \begin{bmatrix} \text{Log} \left( \tilde{\Delta} R_{ij}^\top (R_i^\top R_j) \right) \\ R_i^\top (v_j - v_i - g \Delta t_{ij}) - \tilde{\Delta} v_{ij} \\ R_i^\top (p_j - p_i - v_i \Delta t_{ij} - \frac{1}{2} g \Delta t_{ij}^2) - \tilde{\Delta} p_{ij} \end{bmatrix}$$

注意：所有 tilde ( $\sim$ ) 的量是**IMU 预积分测量**。

而：

- $R_i, v_i, p_i$
- $R_j, v_j, p_j$

是图优化的状态变量。

## 📌 2. 关键疑问：

❓ “姿态估计  $R_i^\top R_j$  是从哪里来的？”

答案如下：

### 🧠 3. 图优化里的状态变量 (State Variables)

LIO-SAM 中优化变量如下：

$$X_i = \{R_i, v_i, p_i, b_i^g, b_i^a\}$$

这些变量存储在 GTSAM 的 `values()` 容器中。

优化算法会不停更新它们，直到所有因子的残差最小。

✨ 所以  $R_i$  是节点  $i$  的姿态变量

是三维旋转矩阵，来源于 GTSAM 的 `Rot3`。

✨  $R_j$  是节点  $j$  的姿态变量

和  $R_i$  一样，也是优化变量。

### ★★ 4. 那 “姿态估计的相对量” 就是两者相乘

$$R_i^\top R_j$$

这是最自然的相对姿态估计：

- 假设  $R_i$  是从世界到节点  $i$  的旋转
  - $R_j$  是从世界到节点  $j$  的旋转
- 则：

$$R_i^\top R_j = R_{i \rightarrow j}$$

这告诉你：

从位姿  $i$  转到位姿  $j$  的相对旋转是多少。

### 📌 5. IMU 因子做的事情非常简单：

## (1) IMU 提供测量

$$\tilde{\Delta R}_{ij}$$

这是预积分得到的。

## (2) 图优化提供估计

$$R_i^\top R_j$$

来自优化变量。

## (3) 两者比较

$$r_\phi = \text{Log} \left( \tilde{\Delta R}_{ij}^\top (R_i^\top R_j) \right)$$

优化希望：

$$R_i^\top R_j \approx \tilde{\Delta R}_{ij}$$

## 6. 那点云匹配的姿在哪里体现？

它不会进入 IMU 残差。

它进入的是：

### Lidar 因子 (Lidar factor)

- 为 pose 提供独立约束
- 通过点到地图距离产生误差项
- 影响  $R_i$  和  $R_j$  的更新

最终结果是：

- Lidar 因子 → 推动  $R_i, R_j$  往点云对齐的方向走
- IMU 因子 → 推动  $R_i^\top R_j$  与预积分一致
- 其他因子 → 推动状态一致

整个系统通过 **多因子共同拉扯** 得到最优解。

## 7. 用一句最简洁的话总结：

**\*\*IMU 残差中的  $R_i^\top R_j$  不是计算出来的，不是估计出来的。**

它就是图优化当前估计的状态变量本身。\*\*

图优化会同时受到：

- IMU 约束
- Lidar 约束
- Bias 约束
- 闭环约束
- ...等共同作用。

最终所有因子一起找到一个最一致的状态。

---

↓ 如果你愿意，我还可以继续给你：

- ✓ IMU 因子中旋转残差的完整雅可比推导（含右雅可比）
- ✓ 为什么 IMU 残差使用右扰动模型（Right perturbation）
- ✓ 点云因子如何影响姿态和 IMU 因子协同工作
- ✓ 预积分中旋转状态的误差传播矩阵  $F$  与  $G$  的详细结构

想继续哪个？

