FAST-LIO 论文解读

zlwang 2022.11

摘要

本文是对 FAST-LIO 论文的详细解读，主要包含公式的详细推导以及一些个人理解，如有错误欢迎批评指正。

目录

1 Abstract

- 高计算效率、高鲁棒性的雷达里程计
- 紧耦合迭代扩展卡尔曼滤波
- 新卡尔曼增益计算公式
- 1200 特征点 25ms

2 Introduction

- 视觉传感器对光照敏感、雷达（旋转式）体积大成本高，不适用于小型机器人
- 固态雷达低成本、重量轻
- 固态雷达带来的新挑战：
  1）特征少的场景易退化，小 FOV 雷达更明显
  2）数千个特征点 +IMU 数据导致计算量太大
  3）激光点采样时间不同，运动失真
- 本文提出了：
  1）迭代卡尔曼滤波融合雷达和 IMU 数据，反向传播来补偿运动失真
  2）新卡尔曼增益计算公式，减少计算量
  3）实验表明能在四旋翼机载计算机实时运行

3 Methodology

3.1 Framework Overview

- 雷达输入到预处理模块，经过一段时间的累积后进行特征提取，提取出面点和角点
- IMU 输入进行前向传播 (积分得到粗略位姿估计)，反向传播进行运动补偿
- 计算雷达里程计的残差，利用迭代卡尔曼滤波估计位姿变换，直到收敛。最后根据位姿建图，并更新特征地图。

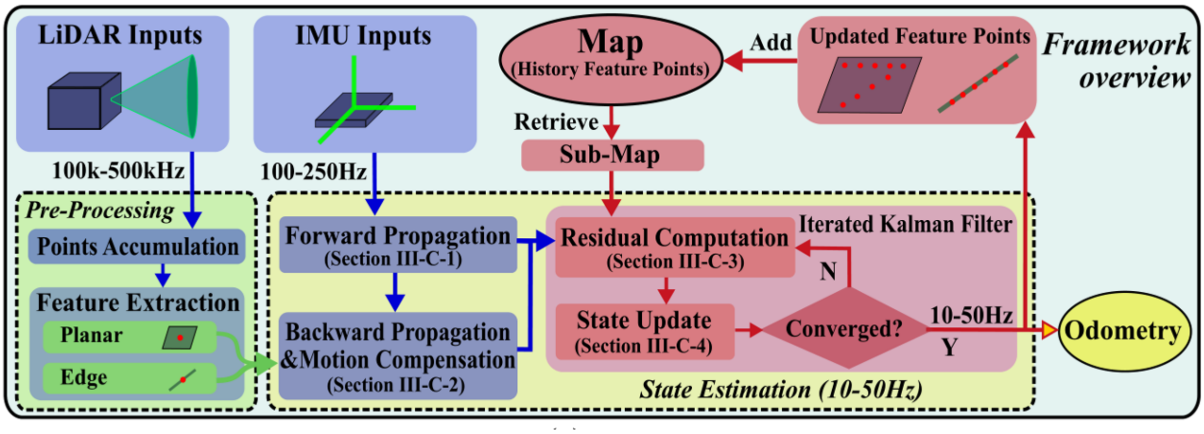


Figure 1: System overview.

3.2 System description

3.2.1 ⊞ / ⊟ operator

首先论文定义了一个新的运算符 ⊞ 和 ⊟:

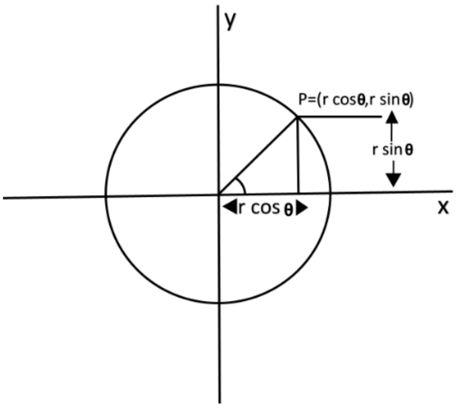
$$\begin{aligned}
&\boxplus : \mathcal{M} \times \mathbb{R}^n \to \mathcal{M}; \quad \boxminus : \mathcal{M} \times \mathcal{M} \to \mathbb{R}^n \\
&\mathcal{M} = SO(3) : \mathbf{R} \boxplus \mathbf{r} = \mathbf{R}\mathrm{Exp}(\mathbf{r}); \qquad \mathbf{R}_1 \boxminus \mathbf{R}_2 = \log\left(\mathbf{R}_2^T \mathbf{R}_1\right) \\
&\mathcal{M} = \mathbb{R}^n : \quad \mathbf{a} \boxplus \mathbf{b} = \mathbf{a} + \mathbf{b}; \qquad \mathbf{a} \boxminus \mathbf{b} = \mathbf{a} - \mathbf{b}
\end{aligned} \tag{1}$$

这里的 $\mathcal{M}$ 表示一种 n 维的流形。$\times$ 表示运算，例如 $\mathcal{M}$ 上的一个元素和 $\mathbb{R}^n$ 上的向量进行 ⊞ 运算, 结果还是 $\mathcal{M}$ 上的元素，⊟ 同理。下文将简单介绍流形相关的知识。当然，**读者也可以完全不用了解流形相关的知识**，只是把它当作是一种简化的方式，或者类似 C++ 的运算符重载即可。这不会影响后续对于论文的理解。

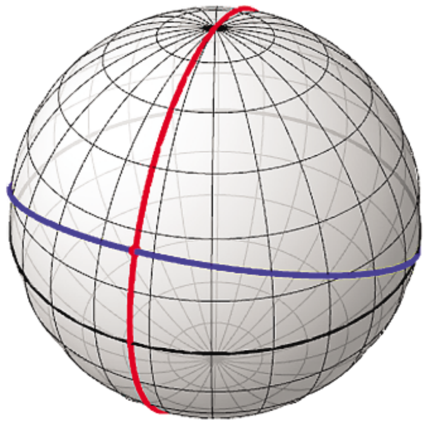
首先我们讨论一下什么是流形 (Manifold)。流形官方的定义是：**局部具有欧几里得空间性质的空间**，在数学中用于描述几何形体。然而，读者看到这句话可能并不清楚流形到底是什么，因此我们首先介绍两个示例。我们要表示一个空间，这个空间的所有点都在单位圆上。读者很自然的想到用直角坐标系来表达它，例如 (1,0) 和 (0,1) 都属于这个空间。但是这种表达方式是冗余的，因为直角坐标系的大部分点都不在单位圆上。我们希望能建立某一种描述方法，让这个描述方法所确定的所有点的集合都能在圆上，甚至能连续不间断地表示圆上的点。这种方式就是极坐标。在原点的圆只需要给定一个半径 R，就能产生连续不断的圆上的点。因此，二维空间中的圆是一个一维流形。另一个例子是三维空间的球面。如果用 (x,y,z) 三维坐标来表述这个空间会产生冗余，因此我们可以用经纬度或球面极坐标来表述它，因此，三维空间中的球面是一个二维流形。如图2所示。

流形学的观点是认为：**我们所能观察到的数据实际上是由一个低维流形映射到高维空**

间上的，即这些数据所在的空间是"嵌入在高维空间的低维流形"。由于数据内部特征的限制，一些高维中的数据会产生维度上的冗余，实际上只需要比较低的维度就能唯一地表示。注意，**流形是一个"空间"**，而不是一个"形状"。(以上关于流形的理解参考 [2])



(a) 二维空间的圆是一个一维流形



(b) 三维空间的球面是一个二维流形

Figure 2: 流形的示例

在论文中我们经常会看到局部同胚 (locally homeomorphic) 这个词。一种非正式但便于理解的描述如下。设流形为 $\mathcal{M}$, 对于 $\forall x \in \mathcal{M}$ 和一个包含 x 的开子集 $U \subset \mathcal{M}$, 存在一个双射函数 (又叫同胚) ϕ, 它可以把 U 中的点映射到 $\mathbb{R}^n$ 上的一个开子集。如图 3所示。

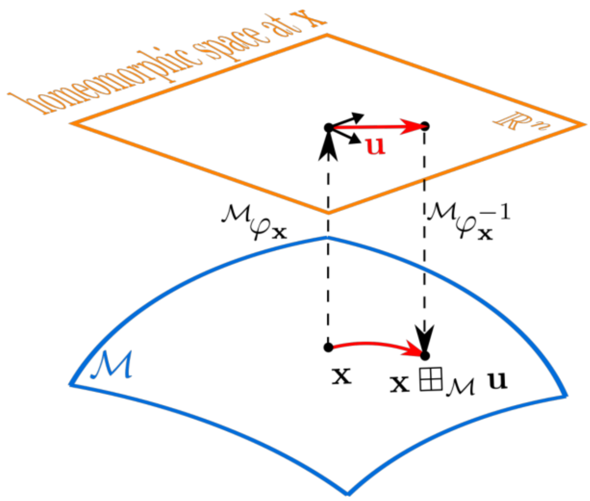


Figure 3: 运算符 ⊞ 的说明 [3]。$\mathbb{R}^n$ 空间在点 x 处局部同胚于 $\mathcal{M}$。

其中 $\mathcal{M}_{\varphi_{\mathbf{x}}}$ 是点 x 附近的同胚 (双射函数)。在 [3] 中，广义的 ⊞ 和 ⊟ 是这样定义的:

$$
\begin{aligned}
\boxplus : \mathcal{M} \times \mathbb{R}^n \to \mathcal{M} \qquad & \mathbf{x} \boxplus_{\mathcal{M}} \mathbf{u} = \mathcal{M}_{\varphi_{\mathbf{x}}^{-1}(\mathbf{u})} \\
\boxminus : \mathcal{M} \times \mathcal{M} \to \mathbb{R}^n \qquad & \mathbf{y} \boxminus_{\mathcal{M}} \mathbf{x} = \mathcal{M}_{\varphi_{\mathbf{x}}}(\mathbf{y})
\end{aligned} \tag{2}
$$

其中 $\forall \mathbf{x}, \mathbf{y} \in \mathcal{M}, \mathbf{u} \in \mathbb{R}^n$。$\boxplus$ 的物理含义可以理解为在 $x \in \mathcal{M}$ 处添加一个小的扰动 $u \in \mathbb{R}^n$, 逆运算 $\boxminus$ 则是确定扰动 u。这看起来可能还是有些抽象，但当 $\mathcal{M}$ 是李群时就比较清楚 (例如 $\mathcal{M} = SO(3)$), 此时 $\boxplus, \boxminus$ 定义为:

$$\mathbf{x} \boxplus_{\mathcal{M}} \mathbf{u} = \mathbf{x} \cdot \operatorname{Exp}(\mathbf{u}); \quad \mathbf{y} \boxminus_{\mathcal{M}} \mathbf{x} = \log\left(\mathbf{x}^{-1} \cdot \mathbf{y}\right) \tag{3}$$

也就是说，此时 $\mathcal{M}_{\varphi_{\mathbf{x}}}$ 是对数映射，$\mathcal{M}_{\varphi_{\mathbf{x}}^{-1}}$ 是指数映射。$\mathbf{x} \boxplus_{\mathcal{M}} \mathbf{u}$ 可以理解为 x 乘上扰动 u 的指数映射，相当于在矩阵 x 上添加了一个李代数上的扰动 u，而 $\mathbf{y} \boxminus_{\mathcal{M}} \mathbf{x}$ 是求李代数上的扰动。至此的知识已经足够我们理解相关论文中关于流形的描述。其他关于流形的知识请读者自行查阅。

回到论文中，定义复合流形的运算:

$$\begin{bmatrix} \mathbf{R} \\ \mathbf{a} \end{bmatrix} \boxplus \begin{bmatrix} \mathbf{r} \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{R} \boxplus \mathbf{r} \\ \mathbf{a} + \mathbf{b} \end{bmatrix}; \quad \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{a} \end{bmatrix} \boxminus \begin{bmatrix} \mathbf{R}_2 \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1 \boxminus \mathbf{R}_2 \\ \mathbf{a} - \mathbf{b} \end{bmatrix} \tag{4}$$

以及以下易于验证的结论:

$$(\mathbf{x} \boxplus \mathbf{u}) \boxminus \mathbf{x} = \mathbf{u}; \quad \mathbf{x} \boxplus (\mathbf{y} \boxminus \mathbf{x}) = \mathbf{y}; \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{M}, \forall \mathbf{u} \in \mathbb{R}^n. \tag{5}$$

3.2.2 Continuous model

这一节主要讲的是 IMU 的运动模型及其积分。IMU 坐标系记为 I, 雷达坐标系记为 L，IMU 和雷达之间的外参定义为 ${}^I\mathbf{T}_L = \left({}^I\mathbf{R}_L, {}^I\mathbf{p}_L\right)$。IMU 的运动模型如下 (第一帧 IMU 的坐标系，定义为全局坐标系 G):

$$\begin{aligned} {}^G\dot{\mathbf{p}}_I &= {}^G\mathbf{v}_I, {}^G\dot{\mathbf{v}}_I = {}^G\mathbf{R}_I\left(\mathbf{a}_m - \mathbf{b}_{\mathbf{a}} - \mathbf{n}_{\mathbf{a}}\right) + {}^G\mathbf{g}, {}^G\dot{\mathbf{g}} = \mathbf{0} \\ {}^G\dot{\mathbf{R}}_I &= {}^G\mathbf{R}_I\left\lfloor \boldsymbol{\omega}_m - \mathbf{b}_{\boldsymbol{\omega}} - \mathbf{n}_{\boldsymbol{\omega}} \right\rfloor_{\wedge}, \dot{\mathbf{b}}_{\boldsymbol{\omega}} = \mathbf{n}_{\mathbf{b}\boldsymbol{\omega}}, \dot{\mathbf{b}}_{\mathbf{a}} = \mathbf{n}_{\mathbf{ba}} \end{aligned} \tag{6}$$

我们先来看其中比较好理解的几个。${}^G\mathbf{g}$ 是重力矢量，我们认为它是不随时间变化的，因此它的导数为 0。$\mathbf{n}_{\mathbf{a}}$ 和 $\mathbf{n}_{\boldsymbol{\omega}}$ 是 IMU 测量值的白噪声。$\mathbf{b}_{\mathbf{a}}$ 和 $\mathbf{b}_{\boldsymbol{\omega}}$ 分别是 IMU 的零偏，我们认为它符合随机游走模型，即它们的导数服从高斯分布。要理解其他几个量，首先我们要知道 IMU 的测量值和真值的关系:

$$\begin{aligned} \mathbf{a}_m &= \mathbf{a} + \mathbf{b}_a + \mathbf{n}_a - {}^I\mathbf{R}_G{}^G\mathbf{g} \\ \boldsymbol{\omega}_m &= \boldsymbol{\omega} + \mathbf{b}_w + \mathbf{n}_w \end{aligned} \tag{7}$$

其中 $\mathbf{a}, \boldsymbol{\omega}$ 表示加速度和角速度真值。这个式子是通用的 IMU 的测量模型，它是易于

理解的，即测量值 = 真值 + 零偏 + 噪声。需要注意的是加速度测量值多包含了一项重力分量，由于它是不随时间变化的，因此只需要把第一帧 (Global 坐标系) 的重力分量乘上变换矩阵 ${}^{I}\mathbf{R}_G$ 转换到当前 IMU 的坐标系即可 (这里令成减去 ${}^{I}\mathbf{R}_G{}^{G}\mathbf{g}$ 是为了与论文公式统一，实际代码中可根据 IMU 重力加速度的正负号来决定是加还是减号)。

接下来回到公式 (6)。${}^{G}\mathbf{p}_I$,${}^{G}\mathbf{v}_I$ 和 ${}^{G}\mathbf{R}_I$ 分别是全局坐标系的位置，速度和姿态。其中位置的导数是速度，速度的导数是加速度 (式中转换到 G 坐标系)，这是易于理解的。关于 ${}^{G}\mathbf{R}_I$ 的导数的推导如下，其中主要参考了 [4]。

考虑考虑一个从原点出发的向量 r 绕单位轴 u 旋转，角速度大小为 $\dot{\theta}$, 角速度可表示为 $\boldsymbol{\omega} = \dot{\theta} u$，如图 4所示。则 r 的导数为:

$$\frac{d\mathbf{r}}{dt} = \boldsymbol{\omega} \times \mathbf{r} \tag{8}$$

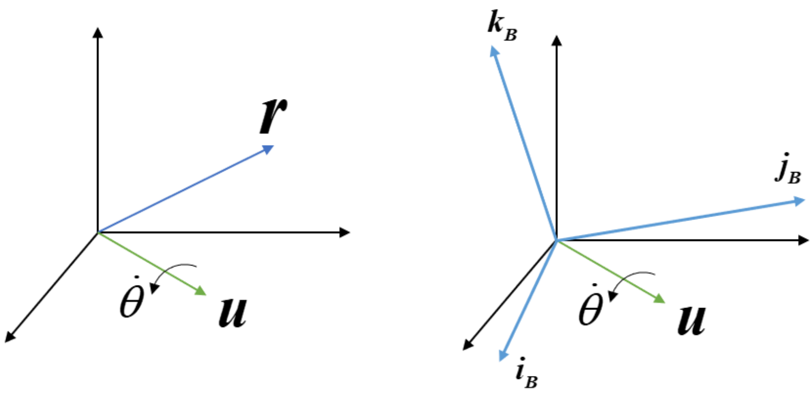


Figure 4: IMU 运动模型推导.

现在考虑图 (右)，坐标系 B 绕单位轴 u 旋转，其三个单位轴的时间一阶导为:

$$\frac{d\mathbf{i}_B}{dt} = \boldsymbol{\omega} \times \mathbf{i}_B, \frac{d\mathbf{j}_B}{dt} = \boldsymbol{\omega} \times \mathbf{j}_B, \frac{d\mathbf{k}_B}{dt} = \boldsymbol{\omega} \times \mathbf{k}_B \tag{9}$$

实际上，$[\mathbf{i}_B, \mathbf{j}_B, \mathbf{k}_B]$ 是一组单位正交基，它和原坐标系的基的内积就是旋转矩阵 (这一部分可以参考十四讲第 3 讲)。而原坐标系的基就是单位阵，因此，$[\mathbf{i}_B, \mathbf{j}_B, \mathbf{k}_B]$ 实际上就是坐标系 B 相对于原坐标系的旋转矩阵 R，因此有:

$$\dot{\mathbf{R}} = \begin{bmatrix} \boldsymbol{\omega} \times \mathbf{i}_B & \boldsymbol{\omega} \times \mathbf{j}_B & \boldsymbol{\omega} \times \mathbf{k}_B \end{bmatrix} = \boldsymbol{\omega} \times \mathbf{R} = \boldsymbol{\omega}^{\wedge}\mathbf{R} \tag{10}$$

其中 ^ 表示向量对应的反对称矩阵 (论文中同理)。然而这里的角速度 $\boldsymbol{\omega}$ 是原坐标系下的，对应论文中的 Global 坐标系; 而图中变换后的坐标系对应论文中的 IMU 坐标系。已知 ${}^{G}\boldsymbol{\omega} = {}^{G}\mathbf{R}_I \cdot {}^{I}\boldsymbol{\omega}$，以及对任意旋转矩阵 R 和三维向量 p, 都有 $(Rp)^{\wedge} = Rp^{\wedge}R^T$(证明

见 [4]), 于是有:

$$
{}^{\mathbf{G}}\dot{\mathbf{R}}_{\mathbf{I}} = ({}^{\mathbf{G}}\mathbf{R}_{\mathbf{I}} \cdot {}^{I}\boldsymbol{\omega})^{\wedge}\,{}^{\mathbf{G}}\mathbf{R}_{\mathbf{I}} = {}^{\mathbf{G}}\mathbf{R}_{\mathbf{I}}({}^{I}\boldsymbol{\omega})^{\wedge} \tag{11}
$$

这个式子和 (6) 是一致的。

3.2.3 Discrete model

这一节主要讲的是离散化上述的状态模型。通过前文求得的导数，有如下离散模型:

$$
\mathbf{x}_{i+1} = \mathbf{x}_i \boxplus \left(\Delta t \mathbf{f}\left(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i\right)\right) \tag{12}
$$

这个公式其实就代表了文中提到的**前向传播 (Forward Propagation)**, 其中 i 是 IMU 测量的 index，其他定义如下:

$$
\begin{aligned}
&\mathcal{M} = SO(3) \times \mathbb{R}^{15}, \dim(\mathcal{M}) = 18 \\
&\mathbf{x} \doteq \begin{bmatrix} {}^{G}\mathbf{R}_I^T & {}^{G}\mathbf{p}_I^T & {}^{G}\mathbf{v}_I^T & \mathbf{b}_{\boldsymbol{\omega}}^T & \mathbf{b}_{\mathbf{a}}^T & {}^{G}\mathbf{g}^T \end{bmatrix}^T \in \mathcal{M} \\
&\mathbf{u} \doteq \begin{bmatrix} \boldsymbol{\omega}_m^T & \mathbf{a}_m^T \end{bmatrix}^T, \mathbf{w} \doteq \begin{bmatrix} \mathbf{n}_{\boldsymbol{\omega}}^T & \mathbf{n}_{\mathbf{a}}^T & \mathbf{n}_{\mathbf{b}\boldsymbol{\omega}}^T & \mathbf{n}_{\mathbf{ba}}^T \end{bmatrix}^T \\
&\mathbf{f}\left(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i\right) = \begin{bmatrix} \boldsymbol{\omega}_{m_i} - \mathbf{b}_{\boldsymbol{\omega}_i} - \mathbf{n}_{\boldsymbol{\omega}_i} \\ {}^{G}\mathbf{v}_{I_i} \\ \mathbf{R}_{I_i}\left(\mathbf{a}_{m_i} - \mathbf{b}_{\mathbf{a}_i} - \mathbf{n}_{\mathbf{a}_i}\right) + {}^{G}\mathbf{g}_i \\ \mathbf{n}_{\mathbf{b}\boldsymbol{\omega}_i} \\ \mathbf{n}_{\mathbf{ba}} \\ \mathbf{0}_{3\times 1} \end{bmatrix}
\end{aligned} \tag{13}
$$

这个公式是易于推导的，读者可自行展开验证。可以看出我们要实时估计的是一个 18 维的量 x, 它包含位置，姿态，速度以及 IMU 的参数。因此它是一个紧耦合的框架。

3.2.4 Discrete model

这一节是雷达数据的预处理，因为作者使用的是固态雷达，因此对固态雷达数据的处理做了如下额外的说明:

- 原始点的采集频率很高 (200kHZ), 因此通常累计一定时间 (20ms, 即 50HZ) 的点作为一次 scan
- 从原始点中提取平面点和边缘点 (LOAM), 设特征点的总数量为 m，每个特征点的时间戳为 ρ_j，并定义每个 IMU 帧的时间戳为 τ_i，如图 5所示

- 最后一个特征点是一次 scan 的终点，即 $\rho_m = t_k$，但 IMU 测量值不与起点或终点对齐

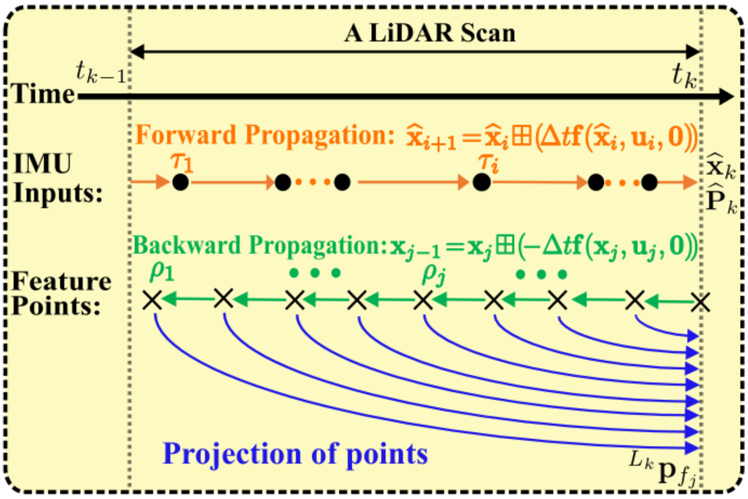


Figure 5: 前向和反向传播.

3.3 State Estimation

本节讲的利用误差卡尔曼滤波来估计状态量 x。我们首先明确文中的定义：

- x 表示真值
- $\bar{x}$ 表示滤波算法结束后得到的最优估计
- $\hat{x}$ 表示滤波算法过程中得到的估计值
- $\tilde{x}$ 表示误差

假设我们现在准备进行第 k 帧的状态估计，那么第 $k-1$ 帧的估计是已完成的。因此，我们定义误差量：

$$\widetilde{\mathbf{x}}_{k-1} \doteq \mathbf{x}_{k-1} \boxminus \overline{\mathbf{x}}_{k-1} = \begin{bmatrix} \delta\boldsymbol{\theta}^T & {}^G\widetilde{\mathbf{p}}_I^T & {}^G\widetilde{\mathbf{v}}_I^T & \widetilde{\mathbf{b}}_\omega^T & \widetilde{\mathbf{b}}_\mathbf{a}^T & {}^G\widetilde{\mathbf{g}}^T \end{bmatrix}^T \tag{14}$$

这里由于第 $k-1$ 帧的估计是已完成的，因此该式中误差使用的是 $\overline{\mathbf{x}}_{k-1}$ 而不是像后文中使用 $\hat{\mathbf{x}}_{k-1}$，此处易混淆，请读者注意。其中 $\delta\boldsymbol{\theta} = \log\left({}^G\overline{\mathbf{R}}_I^{TG}\mathbf{R}_I\right)$ 表示真实姿态和估计姿态之间的小偏差。其他项的误差是向量空间的，直接相减即可。此外，定义协方差矩阵为 $\bar{\mathbf{P}}_{k-1}$。

3.3.1 Forward Propagation

本节讲的是前向传播。前向传播实际上有 2 个内容：一个是**通过 IMU 积分计算一个粗略的状态量 $\hat{x}_k$(见图 5)，这个状态量会用于后续的反向传播来补偿运动失真**。公式如下：

$$\widehat{\mathbf{x}}_{i+1} = \widehat{\mathbf{x}}_i \boxplus \left(\Delta t \mathbf{f}\left(\widehat{\mathbf{x}}_i, \mathbf{u}_i, \mathbf{0}\right)\right); \widehat{\mathbf{x}}_0 = \overline{\mathbf{x}}_{k-1} \tag{15}$$

其中 $\Delta t = \tau_{i+1} - \tau_i$，表示相邻两帧 IMU 之间的时间差。这里的噪声量 $\mathbf{w}_i = 0$ 是因为我们并不知道噪声的实际大小，因此在传播过程中设为零 (噪声实际上会在后续的误差状态方程中考虑)。这个公式和离散模型的公式是一致的。我们每接收一个 IMU 都会进行一次上述计算，直到计算到最后一个 IMU 帧为止。

另一个内容是**传播误差量，并计算对应的协方差矩阵**。这里读者可能会疑惑：我们不知道真值，怎么计算误差呢？实际上，我们计算的误差量，也是一个近似值，因此它才会有对应的协方差矩阵来评判置信度。和传统的卡尔曼滤波器不同的是，传统的卡尔曼滤波器直接估计状态量 x, 它的运动方程和观测方程通常长这样：

$$\begin{aligned} \boldsymbol{x}_k &= f\left(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k\right) + \boldsymbol{w}_k \\ \boldsymbol{z}_k &= h\left(\boldsymbol{x}_k\right) + \boldsymbol{v}_k \end{aligned} \tag{16}$$

而文中使用的**误差状态卡尔曼滤波器 (Error state Kalman filter,ESKF)**，以误差量作为待估计量，也就是把上式的 x 用 $\tilde{x}$ 代替：

$$\begin{aligned} \widetilde{\boldsymbol{x}}_k &= \widetilde{f}\left(\widetilde{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_k\right) + \widetilde{\boldsymbol{w}}_k \\ \boldsymbol{z}_k &= \widetilde{h}\left(\widetilde{\boldsymbol{x}}_k\right) + \widetilde{\boldsymbol{v}}_k \end{aligned} \tag{17}$$

上式中的 $\widetilde{f}, \widetilde{h}$ 等可能并不是一个严谨的表达，但读者应能明白它的含义。也就是说，我们现在要估计的是误差量 $\tilde{x}$，而不是直接估计状态量 x。而有了误差量的估计，再直接加上状态量的估计就是我们求得的最优估计 ($\widetilde{\mathbf{x}}_{k-1} \doteq \mathbf{x}_{k-1} \boxminus \overline{\mathbf{x}}_{k-1}$, 公式 (14))。这将带来以下好处 [5]：

- 在旋转的处理上，ESKF 的状态变量可以采用最小化的参数表达，也就是使用三维变量来表达旋转的增量。而传统 KF 需要用到四元数 (4 维) 或者更高维的表达 (旋转矩阵，9 维)，或采用带有奇异性的表达方式 (欧拉角)
- ESKF 总是在原点附近，离奇异点较远，并且也不会由于离工作点太远而导致线性化近似不够的问题
- ESKF 的状态量为小量，其二阶变量相对来说可以忽略。同时大多数雅可比矩阵

在小量情况下变得非常简单，甚至可以用单位阵代替

- 误差状态的运动学也相比原状态变量要来得更小，因为我们可以把大量更新部分放到原状态变量中

了解了 ESKF 的好处之后，我们现在要做的就是求公式 (17) 中的 $\widetilde{f}, \widetilde{h}$ 的具体表达式，并且将它们线性化。回到论文中，对于运动方程，误差模型如下：

$$\begin{aligned}\widetilde{\mathbf{x}}_{i+1} &= \mathbf{x}_{i+1} \boxminus \widehat{\mathbf{x}}_{i+1} \\ &= \left(\mathbf{x}_i \boxplus \Delta t \mathbf{f}\left(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i\right)\right) \boxminus \left(\widehat{\mathbf{x}}_i \boxplus \Delta t \mathbf{f}\left(\widehat{\mathbf{x}}_i, \mathbf{u}_i, \mathbf{0}\right)\right) \\ &\simeq \mathbf{F}_{\widetilde{\mathbf{x}}} \widetilde{\mathbf{x}}_i + \mathbf{F}_{\mathbf{w}} \mathbf{w}_i .\end{aligned} \tag{18}$$

这里的 $\mathbf{F}_{\widetilde{\mathbf{x}}}$ 和 $\mathbf{F}_{\mathbf{w}}$ 是两个大的稀疏矩阵，具体形式见论文。这两个麻烦的家伙我们稍后再推导它。假设白噪声 $\mathbf{w}$ 的协方差矩阵为 $\mathbf{Q}$，则可以按照下式传播协方差矩阵：

$$\widehat{\mathbf{P}}_{i+1} = \mathbf{F}_{\widetilde{\mathbf{x}}} \widehat{\mathbf{P}}_i \mathbf{F}_{\widetilde{\mathbf{x}}}^T + \mathbf{F}_{\mathbf{w}} \mathbf{Q} \mathbf{F}_{\mathbf{w}}^T ; \widehat{\mathbf{P}}_0 = \overline{\mathbf{P}}_{k-1} \tag{19}$$

这个公式推导自高斯分布运算的基本性质。至此我们完成了前向传播的 2 个内容，即求得了状态估计 $\hat{\mathbf{x}}_k$, 又推导了误差状态的传播，并求得了真实状态和估计状态之间误差的协方差 $\hat{\mathbf{P}}_k$。

接下来我们来推导式 (18)。实际上 ESKF 的误差状态传播有两种推导方式。一种是类似 [5] 中的方法，即**基于误差随时间变化的递推方程**。如果我们能够推导状态误差随时间变化的导数关系，例如 $\dot{\delta \mathbf{x}} = \mathbf{A} \delta \mathbf{x} + \mathbf{B} \mathbf{w}$。则误差状态的传递方程为：

$$\delta \mathbf{x}_k = \delta \mathbf{x}_{k-1} + \dot{\delta \mathbf{x}}_{k-1} \Delta t \tag{20}$$

这种推导方式可以得到和文中类似的结论 (矩阵内有些项可能与 FAST-LIO 不同，其实是简化成了单位阵)。关于这种推导方式可以参考 [5][6]。而我们着重介绍的是论文中使用的**基于一节泰勒展开**的误差传递方法。非线性系统 $\mathbf{x}_k = f\left(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}\right)$ 的状态误差的线性递推关系如下：

$$\begin{aligned}&\mathbf{x}_k = f\left(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}\right) \\ &\rightarrow \hat{\mathbf{x}}_k + \delta \mathbf{x}_k = f\left(\hat{\mathbf{x}}_{k-1} + \delta \mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}\right) \\ &\xrightarrow{Taylor} \underline{\hat{\mathbf{x}}_k} + \delta \mathbf{x}_k = \underline{f\left(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, 0\right)} + \mathbf{F} \delta \mathbf{x}_{k-1} + \mathbf{G} \mathbf{w}_{k-1} \\ &\rightarrow \delta \mathbf{x}_k = \mathbf{F} \delta \mathbf{x}_{k-1} + \mathbf{G} \mathbf{w}_{k-1}\end{aligned} \tag{21}$$

其中, $\mathbf{F}$ 是状态量 $\mathbf{x}_k$ 对状态量 $\mathbf{x}_{k-1}$ 的雅克比矩阵, $\mathbf{G}$ 是状态量 $\mathbf{x}_k$ 对噪声量 $\mathbf{w}_{k-1}$ 的雅克比矩阵。接下来我们的任务就是推导这两个矩阵。

回到论文中，论文中的 $\mathbf{F}_{\widetilde{\mathbf{x}}}$ 对应上式的 $\mathbf{F}$，$\mathbf{F}_{\mathbf{w}}$ 对应上式的 $\mathbf{G}$。我们已知:

- 由公式 (18)：$\widetilde{\mathbf{x}}_{i+1}=\mathbf{x}_{i+1}\boxminus\widehat{\mathbf{x}}_{i+1}=\left(\mathbf{x}_i\boxplus\Delta t\mathbf{f}\left(\mathbf{x}_i,\mathbf{u}_i,\mathbf{w}_i\right)\right)\boxminus\left(\widehat{\mathbf{x}}_i\boxplus\Delta t\mathbf{f}\left(\widehat{\mathbf{x}}_i,\mathbf{u}_i,\mathbf{0}\right)\right)$
- $\mathbf{x}_i=\widehat{\mathbf{x}}_i\boxplus\widetilde{\mathbf{x}}_i$
- 定义 $\mathbf{g}\left(\widetilde{\mathbf{x}}_i,\mathbf{w}_i\right)=\mathbf{f}\left(\mathbf{x}_i,\mathbf{u}_i,\mathbf{w}_i\right)\Delta t=\mathbf{f}\left(\widehat{\mathbf{x}}_i\boxplus\widetilde{\mathbf{x}}_i,\mathbf{u}_i,\mathbf{w}_i\right)\Delta t$

则公式 (18) 可以写为:

$$\widetilde{\mathbf{x}}_{i+1}=\left(\left(\widehat{\mathbf{x}}_i\boxplus\widetilde{\mathbf{x}}_i\right)\boxplus\mathbf{g}\left(\widetilde{\mathbf{x}}_i,\mathbf{w}_i\right)\right)\boxminus\left(\widehat{\mathbf{x}}_i\boxplus\mathbf{g}(\mathbf{0},\mathbf{0})\right)\stackrel{def}{=}\mathbf{G}\left(\widetilde{\mathbf{x}}_i,\mathbf{g}\left(\widetilde{\mathbf{x}}_i,\mathbf{w}_i\right)\right)\tag{22}$$

以上内容见原文附录。为了简化，**在后续推导中我们省去上式中烦人的下标** i。我们要求的是 $\mathbf{G}(\tilde{\mathbf{x}},\mathbf{g}(\tilde{\mathbf{x}},\mathbf{w}))$ 对 $\tilde{\mathbf{x}}$ 和 $\mathbf{w}$ 的偏导 (在 $\tilde{x}=w=0$ 附近，因为 $\tilde{x}$ 是误差量接近 0，w 是噪声)。这是一个复合函数，因此有:

$$F_{\widetilde{x}}=\left.\frac{\partial G(\widetilde{x},g(0,0))}{\partial\widetilde{x}}+\frac{\partial G(0,g(\widetilde{x},0))}{\partial g(\widetilde{x},0)}\frac{\partial g(\widetilde{x},0)}{\partial\widetilde{x}}\right|_{\widetilde{x}=0}\xlongequal{simplify}\left.\frac{\partial G}{\partial\widetilde{x}}+\frac{\partial G}{\partial g}\frac{\partial g}{\partial\widetilde{x}}\right|_{\widetilde{x}=0}\tag{23}$$

$$F_w=\left.\frac{\partial G(0,g(0,w))}{\partial g(0,w)}\frac{\partial g(0,w)}{\partial w}\right|_{w=0}\xlongequal{simplify}\left.\frac{\partial G}{\partial g_w}\frac{\partial g}{\partial w}\right|_{w=0}\tag{24}$$

接下来，我们先考虑这样一个流形的偏导数。对于流形 $\mathbf{E}=((\mathbf{a}\boxplus\mathbf{b})\boxplus\mathbf{c})\boxminus\mathbf{d}$，有以下结论 [3]:

$$\frac{\partial\mathbf{E}}{\partial\mathbf{b}}=\frac{\partial\mathbf{E}}{\partial\mathbf{c}}=\mathbf{I}_{\mathbf{n}\times\mathbf{n}},\ a,b,c,d\in\mathbb{R}^n\tag{25}$$

$$\frac{\partial\mathbf{E}}{\partial\mathbf{b}}=\mathbf{A}(\mathbf{E})^{-\mathbf{T}}\mathbf{Exp}(-\mathbf{c})\mathbf{A}(\mathbf{b})^{\mathbf{T}},\frac{\partial\mathbf{E}}{\partial\mathbf{c}}=\mathbf{A}(\mathbf{E})^{-\mathbf{T}}\mathbf{A}(\mathbf{c})^{\mathbf{T}},\ a,d\in SO(3),b,c\in\mathbb{R}^3\tag{26}$$

其中

$$\begin{aligned}\mathbf{A}(\mathbf{u})^{-1}&=\mathbf{I}-\frac{1}{2}\lfloor\mathbf{u}\rfloor\wedge+(1-\boldsymbol{\alpha}(\|\mathbf{u}\|))\frac{\lfloor\mathbf{u}\rfloor_\wedge^2}{\|\mathbf{u}\|^2}\\ \alpha(\mathrm{m})&=\frac{\mathrm{m}}{2}\cot\left(\frac{\mathrm{m}}{2}\right)=\frac{\mathrm{m}}{2}\frac{\cos(\mathrm{m}/2)}{\sin(\mathrm{m}/2)}\end{aligned}\tag{27}$$

也就是 BCH 近似中的雅可比矩阵 (见 14 讲)。对于式 (25)，这是显然的，因为欧式空

间下 ⊞⊟ 就是常用的加减法。我们接下来推导式 (26)。关于 **b** 求偏导如下:

$$
\begin{aligned}
&E = ((a \boxplus b) \boxplus c) \boxminus d = Log(d^{-1} \cdot (a \boxplus b) \boxplus c)) \\
&\rightarrow Exp(E) = d^{-1} \cdot a \cdot Exp(b) \cdot Exp(c) \\
&\rightarrow Exp(E + \Delta E) = d^{-1} \cdot a \cdot Exp(b + \Delta b) \cdot Exp(c) \\
&\xrightarrow{BCH} Exp(E)Exp(A(E)^T \Delta E) = d^{-1} \cdot a \cdot Exp(b)Exp(A(b)^T \Delta b) \cdot Exp(c) \\
&\rightarrow Exp(A(E)^T \Delta E) = Exp(-c)Exp(A(b)^T \Delta b) \cdot Exp(c) \\
&\rightarrow Exp(A(E)^T \Delta E) \overset{Property}{=\!=\!=} Exp(Exp(-c)A(b)^T \Delta b) \\
&\rightarrow \frac{\Delta E}{\Delta b} = A(E)^{-T} Exp(-c) A(b)^T
\end{aligned}
\tag{28}
$$

其中第四行用了 BCH 近似公式，倒数第二行用了 SO(3) 上矩阵的伴随性质。关于 **c** 求偏导同理:

$$
\begin{aligned}
&\rightarrow Exp(E + \Delta E) = d^{-1} \cdot a \cdot Exp(b) \cdot Exp(c + \Delta c) \\
&\xrightarrow{BCH} Exp(E)Exp(A(E)^T \Delta E) = d^{-1} \cdot a \cdot Exp(b) \cdot Exp(c)Exp(A(c)^T \Delta c) \\
&\rightarrow Exp(A(E)^T \Delta E) = Exp(A(c)^T \Delta c) \\
&\rightarrow \frac{\Delta E}{\Delta c} = A(E)^{-T} A(c)^T
\end{aligned}
\tag{29}
$$

至此我们完成了式 (26) 的推导。回到式 (23)(24), 实际上 $\mathbf{G}$ 就是形如 $((\mathbf{a} \boxplus \mathbf{b}) \boxplus \mathbf{c}) \boxminus \mathbf{d}$ 的形式，因为 $\mathbf{G} = (\hat{\mathbf{x}} \boxplus \tilde{\mathbf{x}}) \boxplus \mathbf{g}(\tilde{\mathbf{x}}, \mathbf{w}) \boxminus (\hat{\mathbf{x}} \boxplus \mathbf{g}(\mathbf{0}, \mathbf{0}))$。由 $F_{\tilde{x}} = \frac{\partial G}{\partial \tilde{x}} + \frac{\partial G}{\partial g}\frac{\partial g}{\partial \tilde{x}}\Big|_{\tilde{x}=0}$, 我们推导:

$$
\begin{aligned}
\frac{\partial G}{\partial \tilde{x}}\Big|_{\tilde{x}=0} &\overset{(26)}{=\!=\!=} \begin{bmatrix} \mathbf{A}(\mathbf{0})^{-1}\mathbf{Exp}(-\mathbf{g}(\mathbf{0},\mathbf{0}))\mathbf{A}(\mathbf{0})^{\mathbf{T}} & 0 \\ 0 & \mathbf{I_{15\times15}} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{Exp}(-\mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}, \mathbf{0})\mathbf{\Delta t}) & 0 \\ 0 & \mathbf{I_{15\times15}} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{Exp}(-(\omega_{\mathbf{m}} - \hat{\mathbf{b}}_{\omega})\mathbf{\Delta t}) & 0 \\ 0 & \mathbf{I_{15\times15}} \end{bmatrix}
\end{aligned}
\tag{30}
$$

其中，当 $\tilde{x} = 0$ 时，$G = 0$ 且 $A(0) = I$(易于验证)。由于状态量 x 第一项是 SO(3) 的，其它项都是向量，因此式中的矩阵只有前三行代入式 (26), 其他行代入式 (25)。同

理，我们推导：

$$\left.\frac{\partial G}{\partial g}\right|_{\widetilde{x}=0} \xlongequal{(26)} \begin{bmatrix} \mathbf{A}(\mathbf{0})^{-1}\mathbf{A}(\mathbf{g}(\mathbf{0},\mathbf{0}))^{\mathbf{T}} & 0 \\ 0 & \mathbf{I_{15\times 15}} \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{A}((\omega_{\mathbf{m}} - \hat{\mathbf{b}}_{\omega})\mathbf{\Delta t}) & 0 \\ 0 & \mathbf{I_{15\times 15}} \end{bmatrix} \tag{31}$$

最后，关于 $\frac{\partial g}{\partial x}$ 的求解，由于 $\tilde{\mathbf{x}} \doteq \mathbf{x} \boxminus \hat{\mathbf{x}} = \begin{bmatrix} \delta\boldsymbol{\theta}^T & {}^G\tilde{\mathbf{p}}_I^T & {}^G\tilde{\mathbf{v}}_I^T & \tilde{\mathbf{b}}_{\boldsymbol{\omega}}^T & \tilde{\mathbf{b}}_{\mathbf{a}}^T & {}^G\tilde{\mathbf{g}}^T \end{bmatrix}^T$,$\mathbf{w} \doteq \begin{bmatrix} \mathbf{n}_{\boldsymbol{\omega}}^T & \mathbf{n}_{\mathbf{a}}^T & \mathbf{n}_{\mathbf{b}\boldsymbol{\omega}}^T & \mathbf{n}_{\mathbf{ba}}^T \end{bmatrix}^T$, 以及：

$$\mathbf{g}(\tilde{\mathbf{x}},\mathbf{w}) = \mathbf{f}(\mathbf{x},\mathbf{u},\mathbf{w})\Delta t = \begin{bmatrix} \boldsymbol{\omega}_m - \mathbf{b}_{\omega} - \mathbf{n}_{\omega} \\ {}^G\mathbf{v}_I \\ \mathbf{R}_I(\mathbf{a}_m - \mathbf{b}_{\mathbf{a}} - \mathbf{n}_{\mathbf{a}}) + {}^G\mathbf{g} \\ \mathbf{n}_{\mathbf{b}\omega} \\ \mathbf{n}_{\mathbf{ba}} \\ \mathbf{0}_{3\times 1} \end{bmatrix}\Delta t$$
$$= \begin{bmatrix} \boldsymbol{\omega}_m - \hat{\mathbf{b}}_{\omega} - \tilde{\mathbf{b}}_{\omega} - \mathbf{n}_{\omega} \\ {}^G\hat{\mathbf{v}}_I + {}^G\tilde{\mathbf{v}}_I \\ \hat{\mathbf{R}}_{\mathbf{I}}\mathbf{Exp}(\delta\boldsymbol{\theta}^{\mathbf{T}})\left(\mathbf{a}_m - \hat{\mathbf{b}}_{\mathbf{a}} - \tilde{\mathbf{b}}_{\mathbf{a}} - \mathbf{n}_{\mathbf{a}}\right) + {}^G\mathbf{g} \\ \mathbf{n}_{\mathbf{b}\omega} \\ \mathbf{n}_{\mathbf{b}a} \\ \mathbf{0}_{3\times 1} \end{bmatrix}\Delta t \tag{32}$$

则：

$$\left.\frac{\partial g(\tilde{\mathbf{x}},0)}{\partial \widetilde{x}}\right|_{\widetilde{x}=0} = \begin{pmatrix} 0 & 0 & 0 & -\mathbf{I}_{3\times 3}\Delta t & 0 & 0 \\ 0 & 0 & \mathbf{I}_{3\times 3}\Delta t & 0 & 0 & 0 \\ -\hat{\mathbf{R}}_{\mathbf{I}}(\mathbf{a_m} - \hat{\mathbf{b}}_{\mathbf{a}})^{\wedge}\Delta t & 0 & 0 & 0 & -\hat{\mathbf{R}}_{\mathbf{I}}\Delta t & \mathbf{I}_{3\times 3}\Delta t \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{33}$$

其中，第三行关于 R 的求导利用了右乘扰动模型。至此，根据式 (23) 中：$F_{\widetilde{x}} = \frac{\partial G}{\partial \widetilde{x}} + \frac{\partial G}{\partial g}\frac{\partial g}{\partial \widetilde{x}}\Big|_{\widetilde{x}=0}$，

再加上不烦人的下标 i, 得到:

$$\mathbf{F}_{\widetilde{\mathbf{x}}}=\begin{bmatrix}\mathbf{Exp}(-(\omega_{\mathbf{m}}-\hat{\mathbf{b}}_{\omega})\Delta t) & 0 & 0 & -\mathbf{A}((\omega_{\mathbf{m}}-\hat{\mathbf{b}}_{\omega})\Delta t)\Delta t & 0 & 0\\ 0 & \mathbf{I} & \mathbf{I}\Delta t & 0 & 0 & 0\\ -{}^{G}\widehat{\mathbf{R}}_{I_i}(\mathbf{a}_{\mathbf{m}}-\hat{\mathbf{b}}_{\mathbf{a}})^{\wedge}\Delta t & 0 & \mathbf{I} & 0 & -{}^{G}\widehat{\mathbf{R}}_{I_i}\Delta t & \mathbf{I}\Delta t\\ 0 & 0 & 0 & \mathbf{I} & 0 & 0\\ 0 & 0 & 0 & 0 & \mathbf{I} & 0\\ 0 & 0 & 0 & 0 & 0 & \mathbf{I}\end{bmatrix} \tag{34}$$

这和论文中是一致的结论 (注意论文中 $\widehat{\omega}_i=\boldsymbol{\omega}_{m_i}-\widehat{\mathbf{b}}_{\boldsymbol{\omega}_i},\widehat{\mathbf{a}}_i=\mathbf{a}_{m_i}-\widehat{\mathbf{b}}_{\mathbf{a}_i}$)。

同理，根据 $F_w=\frac{\partial G}{\partial g_w}\frac{\partial g}{\partial w}\Big|_{w=0}$，我们推导:

$$\begin{aligned}\frac{\partial G}{\partial g_w}\Big|_{w=0} &\xlongequal{(26)}\begin{bmatrix}\mathbf{A}(\mathbf{0})^{-\mathbf{1}}\mathbf{A}(\mathbf{g}(\mathbf{0},\mathbf{0}))^{\mathbf{T}} & 0\\ 0 & \mathbf{I}_{\mathbf{15}\times\mathbf{15}}\end{bmatrix}\\ &=\begin{bmatrix}\mathbf{A}((\omega_{\mathbf{m}}-\hat{\mathbf{b}}_{\omega})\mathbf{\Delta t}) & 0\\ 0 & \mathbf{I}_{\mathbf{15}\times\mathbf{15}}\end{bmatrix}\end{aligned} \tag{35}$$

以及:

$$\frac{\partial g}{\partial w}\Big|_{w=0}=\begin{pmatrix}-\mathbf{I}_{3\times3}\Delta t & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & -\hat{\mathbf{R}}_{\mathbf{I}}\Delta t & 0 & 0\\ 0 & 0 & \mathbf{I}_{3\times3} & 0\\ 0 & 0 & 0 & \mathbf{I}_{3\times3}\\ 0 & 0 & 0 & 0\end{pmatrix} \tag{36}$$

加上下标 i, 得到:

$$F_w=\begin{pmatrix}-\mathbf{A}((\omega_{\mathbf{m}}-\hat{\mathbf{b}}_{\omega})\Delta t)\Delta t & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & -\hat{\mathbf{R}}_{\mathbf{I}}\Delta t & 0 & 0\\ 0 & 0 & \mathbf{I}\Delta t & 0\\ 0 & 0 & 0 & \mathbf{I}\Delta t\\ 0 & 0 & 0 & 0\end{pmatrix} \tag{37}$$

至此，我们终于推导完成这个大家伙了！有了这个递推关系，我们相当于已经完成了 ESKF 的运动方程的线性化的推导。后文我们将进行 ESKF 观测方程的推导。这里值得

一提的是，有部分读者可能会问，这与其它论文的结果或代码中的矩阵不太一样呢？实际上式中的部分量是可以简化的，例如 $-\mathbf{A}((\omega_{\mathbf{m}}-\hat{\mathbf{b}}_\omega)\Delta t)\Delta t$ 可以简化为 $\mathbf{I}$, 因为 Δt 足够小。例如，简化后与 [5] 中的矩阵是一致的。

3.3.2 Backward Propagation and Motion Compensation

这一部分我们主要进行反向传播，即运动补偿。前文我们已经提到，由于雷达每个点的采样时间不同，而雷达在运动，会导致运动失真。我们希望拿到的数据是所有点在同一时刻的采样，文中是在 t_k 时刻。因此，我们根据 IMU 积分估计的位姿，把每个点转到 t_k 时刻，如图 6所示。

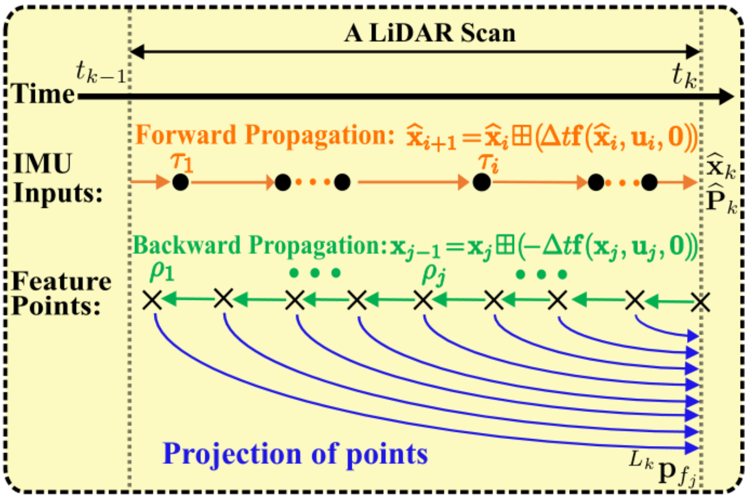


Figure 6: 前向和反向传播.

从 $\hat{x}_k$ 开始 (此处设为零点，因为我们实际上计算的是相对于 $\hat{x}_k$ 的位姿)，利用 $\check{\mathbf{x}}_{j-1}=\check{\mathbf{x}}_j\boxplus(-\Delta t\mathbf{f}(\check{\mathbf{x}}_j,\mathbf{u}_j,\mathbf{0}))$ 计算相对位姿。通常点的频率是大于 IMU 频率的，因此对于两帧 IMU 之间的多个特征点，我们以左 IMU 帧为准。此外，考虑到 (13) 中后三行 (加速度噪声) 为零，我们把反向传播简化：

$$
\begin{aligned}
&{}^{I_k}\check{\mathbf{p}}_{I_{j-1}}={}^{I_k}\check{\mathbf{p}}_{I_j}-{}^{I_k}\check{\mathbf{v}}_{I_j}\Delta t, \quad \text{s.f. } {}^{I_k}\check{\mathbf{p}}_{I_m}=\mathbf{0};\\
&{}^{I_k}\check{\mathbf{v}}_{I_{j-1}}={}^{I_k}\check{\mathbf{v}}_{I_j}-{}^{I_k}\check{\mathbf{R}}_{I_j}\left(\mathbf{a}_{m_{i-1}}-\widehat{\mathbf{b}}_{\mathbf{a}_k}\right)\Delta t-{}^{I_k}\widehat{\mathbf{g}}_k\Delta t\\
&\quad \text{s.f. } {}^{I_k}\check{\mathbf{v}}_{I_m}={}^{G}\widehat{\mathbf{R}}_{I_k}^{T}\,{}^{G}\widehat{\mathbf{v}}_{I_k}\,{}^{I_k}\widehat{\mathbf{g}}_k={}^{G}\widehat{\mathbf{R}}_{I_k}^{T}\widehat{\mathbf{g}}_k;\\
&{}^{I_k}\check{\mathbf{R}}_{I_{j-1}}={}^{I_k}\check{\mathbf{R}}_{I_j}\operatorname{Exp}\left(\left(\widehat{\mathbf{b}}_{\boldsymbol{\omega}_k}-\boldsymbol{\omega}_{m_{i-1}}\right)\Delta t\right), \text{ s.f. } {}^{I_k}\mathbf{R}_{I_m}=\mathbf{I}.
\end{aligned}
\tag{38}
$$

其中 s.f. 表示"starting from"。定义我们反向传播得到的点的相对位姿是 ${}^{I_k}\check{\mathbf{T}}_{I_j}=\left({}^{I_k}\check{\mathbf{R}}_{I_j},{}^{I_k}\check{\mathbf{p}}_{I_j}\right)$, 我们把该点 ${}^{L_j}\mathbf{p}_{f_j}$ 投影到扫描结束时间 t_k:

$$
{}^{L_k}\mathbf{p}_{f_j}={}^{I}\mathbf{T}_L^{-1}\,{}^{I_k}\check{\mathbf{T}}_{I_j}\,{}^{I}\mathbf{T}_L\,{}^{L_j}\mathbf{p}_{f_j}. \tag{39}
$$

其中 ${}^{I}\mathbf{T}_L$ 表示雷达坐标系转到 IMU 坐标系的外参矩阵，上式的含义是先把 L 坐标系的特征点转到 I 坐标系，然后乘上反向传播的补偿矩阵，再转回 L 坐标系。所有的投影点都像这样转好后，就可以开始计算残差了。

3.3.3 Residual computation

设迭代卡尔曼滤波的当前迭代为 κ(用于表示迭代的代数)，我们可以把补偿后的特征点转换到全局坐标系：

$$
{}^{G}\widehat{\mathbf{p}}_{f_j}^{\kappa} = {}^{G}\widehat{\mathbf{T}}_{I_k}^{\kappa}\, {}^{I}\mathbf{T}_L\, {}^{L_k}\mathbf{p}_{f_j}; j = 1, \cdots, m \tag{40}
$$

其中 $\widehat{\mathbf{T}}_{I_k}^{\kappa}$ 是待求的位姿变换。残差的计算和 LOAM 中类似，采用点面距离和点线距离作为残差。定义 $\mathbf{u}_j$ 为平面或边的法向量，${}^{G}\mathbf{q}_j$ 是平面或边上的一个点，则残差 $\mathbf{z}_j^{\kappa}$ 表示为：

$$
\mathbf{z}_j^{\kappa} = \mathbf{G}_j \left({}^{G}\widehat{\mathbf{p}}_{f_j}^{\kappa} - {}^{G}\mathbf{q}_j \right) \tag{41}
$$

其中 $\mathbf{G}_j = \mathbf{u}_j^T$ 当面点 p 为平面点，$\mathbf{G}_j = \mathbf{u}_j^{\wedge}$ 当面点 p 为角点。公式的含义表示点面距离或点线距离，和 LOAM 一致，读者可自行推导。近邻点搜索采用 KD-tree。

3.3.4 Iterated state update

本节开始进行迭代更新。前文说到我们已经完成了 ESKF 的运动方程的线性化，现在我们需要线性化观测方程。其中，观测方程的噪声来自雷达测距的误差，因此对于点 ${}^{L_j}\mathbf{p}_{f_j}$，我们定义如下的噪声量：

$$
{}^{L_j}\mathbf{p}_{f_j}^{\mathrm{gt}} = {}^{L_j}\mathbf{p}_{f_j} - {}^{L_j}\mathbf{n}_{f_j} \tag{42}
$$

理论上，如果我们使用状态量的真值 $\mathbf{x}_k$ 以及雷达点的真值 ${}^{L_j}\mathbf{p}_{f_j}^{\mathrm{gt}}$，其残差应当为零，把上式代入到残差的计算 (把 (42) 代入到 (39), 再代入到 (40)(41))：

$$
\mathbf{0} = \mathbf{h}_j \left(\mathbf{x}_k, {}^{L_j}\mathbf{n}_{f_j} \right) = \mathbf{G}_j \left({}^{G}\mathbf{T}_{I_k}\, {}^{I_k}\check{\mathbf{T}}_{I_j}\, {}^{I}\mathbf{T}_L \left({}^{L_j}\mathbf{p}_{f_j} - {}^{L_j}\mathbf{n}_{f_j} \right) - {}^{G}\mathbf{q}_j \right) \tag{43}
$$

其中 $h(x, n)$ 表示观测方程的函数，这个方程也就是观测方程。我们需要将它线性化，在 $\widehat{\mathbf{x}}_k^{\kappa}$ 附近展开 (即在 $\widetilde{\mathbf{x}}_k^{\kappa} = 0$ 处，因为 $\widetilde{\mathbf{x}}_k^{\kappa} = \mathbf{x}_k \boxminus \widehat{\mathbf{x}}_k^{\kappa}$ 或 $\mathbf{x}_k = \widehat{\mathbf{x}}_k^{\kappa} \boxplus \widetilde{\mathbf{x}}_k^{\kappa}$)：

$$
\begin{aligned}
\mathbf{0} &= \mathbf{h}_j \left(\mathbf{x}_k, {}^{L_j}\mathbf{n}_{f_j} \right) = \mathbf{h}_j \left(\widehat{\mathbf{x}}_k^{\kappa} \boxplus \widetilde{\mathbf{x}}_k^{\kappa}, {}^{L_j}\mathbf{n}_{f_j} \right) \\
&\simeq \mathbf{h}_j \left(\widehat{\mathbf{x}}_k^{\kappa}, \mathbf{0} \right) + \mathbf{H}_j^{\kappa} \widetilde{\mathbf{x}}_k^{\kappa} + \mathbf{v}_j \\
&= \mathbf{z}_j^{\kappa} + \mathbf{H}_j^{\kappa} \widetilde{\mathbf{x}}_k^{\kappa} + \mathbf{v}_j
\end{aligned} \tag{44}
$$

其中 $\mathbf{H}_j^\kappa$ 是 $h(x,n)$ 关于 $\tilde{\mathbf{x}}_k^\kappa$ 的雅可比矩阵 (且取值为 $\tilde{\mathbf{x}}_k^\kappa=0$)，$\mathbf{v}_j \in \mathcal{N}(\mathbf{0}, \mathbf{R}_j)$ 来自于原始测量的噪声 ${}^{L_j}\mathbf{n}_{f_j}$。其中 $\mathbf{H}$ 的计算可以利用链式法则:

$$\boldsymbol{H} = \frac{\partial \mathbf{h}}{\partial \boldsymbol{p}} \frac{\partial \boldsymbol{p}}{\partial \widetilde{\boldsymbol{x}}} \tag{45}$$

其中，$\boldsymbol{p}$ 表示转换到 {G} 系的特征点。而论文中并没有给出 $\boldsymbol{H}$ 的详细推导，此处我们推导一下 $\boldsymbol{H}$ 的具体形式，它与代码中的结果是完全相同的。设在 {L} 坐标系下的特征点为 ${}^{L}p$, 有以下转换关系:

$$\boldsymbol{p} = {}^{G}\hat{\mathbf{R}}_I({}^{I}\mathbf{R}_L{}^{L}\mathbf{p} + {}^{I}\mathbf{T}_L) + {}^{G}\hat{\mathbf{T}}_I \tag{46}$$

这里以面点为例 (实际上在 fast-lio2 以及代码中，都省去了特征提取，只计算面点的残差)，设 $\boldsymbol{p}$ 的近邻点所构成的平面的单位法向量为 $\vec{n}$，则残差定义为

$$\boldsymbol{h} = (\boldsymbol{p} - \boldsymbol{p}_a) \cdot \vec{\boldsymbol{n}} \tag{47}$$

其中 p_a 为该平面上的一点。则

$$\frac{\partial \mathbf{h}}{\partial \boldsymbol{p}} = \vec{\boldsymbol{n}} \tag{48}$$

接下来我们求解 $\frac{\partial \boldsymbol{p}}{\partial \delta \boldsymbol{x}}$。实际上该矩阵只与 ${}^{G}\hat{\mathbf{R}}_I$ 和 ${}^{G}\hat{\mathbf{T}}_I$ 有关，也就是该矩阵只需要求前 6 列，其他行都为零。即:

$$\frac{\partial \boldsymbol{p}}{\partial \widetilde{\boldsymbol{x}}} = [-{}^{G}\hat{\mathbf{R}}_I({}^{I}\mathbf{R}_L{}^{L}\mathbf{p} + {}^{I}\mathbf{T}_L)^{\wedge}, I_{3\times3}, 0, 0, 0, 0] \tag{49}$$

其中第一项利用了李代数求导的右扰动模型。将以上 2 式相乘即可得到:

$$H = [-\vec{\boldsymbol{n}}{}^{G}\hat{\mathbf{R}}_I({}^{I}\mathbf{R}_L{}^{L}\mathbf{p} + {}^{I}\mathbf{T}_L)^{\wedge}, \vec{\boldsymbol{n}}, 0, 0, 0, 0] \tag{50}$$

至此我们完成了观测方程的推导。但还并没有到发起总攻的时候。我们现在得到的是 EKF 的表达式，而文中使用的 IEKF(迭代扩展卡尔曼滤波) 就是在 EKF 进行了改进，其直接的想法就是在 $x_k \rightarrow x_{k+1}$ 的过程中进行多次迭代，以达到消减非线性的影响 (关于 IEKF 读者可参考 [7])。这里比较有意思的是，IEKF 可以证明和高斯牛顿 GN 法是等价的，因此 IEKF 可以保证全局收敛 ([8] 中证明了文中的 IEKF 和 GN 是等价的)。

在 IEKF 中，迭代过程不会更新协方差矩阵 P。而文中用了如下公式，使得我们在迭

代过程中也能够更新协方差矩阵：

$$\widetilde{\mathbf{x}}_k = \mathbf{x}_k \boxminus \widehat{\mathbf{x}}_k = (\widehat{\mathbf{x}}_k^\kappa \boxplus \widetilde{\mathbf{x}}_k^\kappa) \boxminus \widehat{\mathbf{x}}_k = \widehat{\mathbf{x}}_k^\kappa \boxminus \widehat{\mathbf{x}}_k + \mathbf{J}^\kappa \widetilde{\mathbf{x}}_k^\kappa \tag{51}$$

其中：

$$\mathbf{J}^\kappa = \begin{bmatrix} \mathbf{A}\left({}^G\widehat{\mathbf{R}}_{I_k}^\kappa \boxminus {}^G\widehat{\mathbf{R}}_{I_k}\right)^{-T} & \mathbf{0}_{3\times15} \\ \mathbf{0}_{15\times3} & \mathbf{I}_{15\times15} \end{bmatrix} \tag{52}$$

为雅可比矩阵，上式同样是通过式 (26) 推导而来。由于每迭代一次后 $\tilde{\mathbf{x}}$ 就会发生一次变化，理论上其协方差矩阵并不是初始的 $\mathbf{P}_k$ 了，因此协方差矩阵可通过 $\mathbf{P} = (\mathbf{J}^\kappa)^{-1}\widehat{\mathbf{P}}_k(\mathbf{J}^\kappa)^{-T}$ 在迭代过程中更新。可以看到该矩阵实际上也是接近单位阵的。

至此，我们完成了 ESKF 的所有前置任务。最大后验估计：

$$\min_{\widetilde{\mathbf{x}}_k^\kappa}\left(\|\mathbf{x}_k \boxminus \widehat{\mathbf{x}}_k\|^2_{\widehat{\mathbf{P}}_k^{-1}} + \sum_{j=1}^m \left\|\mathbf{z}_j^\kappa + \mathbf{H}_j^\kappa \widetilde{\mathbf{x}}_k^\kappa\right\|^2_{\mathbf{R}_j^{-1}}\right) \tag{53}$$

令 $\mathbf{H} = \left[\mathbf{H}_1^{\kappa^T}, \cdots, \mathbf{H}_m^{\kappa^T}\right]^T, \mathbf{R} = \mathrm{diag}(\mathbf{R}_1, \cdots \mathbf{R}_m), \mathbf{P} = (\mathbf{J}^\kappa)^{-1}\widehat{\mathbf{P}}_k(\mathbf{J}^\kappa)^{-T}$, 以及 $\mathbf{z}_k^\kappa = \left[\mathbf{z}_1^{\kappa^T}, \cdots, \mathbf{z}_m^{\kappa^T}\right]^T$, 则卡尔曼增益：

$$\mathbf{K} = \mathbf{P}\mathbf{H}^T\left(\mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R}\right)^{-1} \tag{54}$$

迭代公式：

$$\widehat{\mathbf{x}}_k^{\kappa+1} = \widehat{\mathbf{x}}_k^\kappa \boxplus \left(-\mathbf{K}\mathbf{z}_k^\kappa - (\mathbf{I} - \mathbf{K}\mathbf{H})(\mathbf{J}^\kappa)^{-1}(\widehat{\mathbf{x}}_k^\kappa \boxminus \widehat{\mathbf{x}}_k)\right) \tag{55}$$

接下来我们推导这个公式 (参考了 [8])。省略掉下标 k，令 $X = \widehat{\mathbf{x}}^\kappa \boxminus \widehat{\mathbf{x}}$，式 (53) 总误差可以表示为如下的最小二乘的形式：

$$E = \min_{\widetilde{x}^\kappa} \frac{1}{2}\begin{pmatrix} z^\kappa + H\widetilde{x}^\kappa \\ X + J^\kappa\widetilde{x}^\kappa \end{pmatrix}^T \begin{pmatrix} R & \\ & \hat{P} \end{pmatrix}^{-1} \begin{pmatrix} z^\kappa + H\widetilde{x}^\kappa \\ X + J^\kappa\widetilde{x}^\kappa \end{pmatrix} = \min_{\widetilde{x}^\kappa}\frac{1}{2}M^TM \tag{56}$$

其中

$$M = S\begin{pmatrix} z^\kappa + H\widetilde{x}^\kappa \\ X + J^\kappa\widetilde{x}^\kappa \end{pmatrix}, S^TS = \begin{pmatrix} R & \\ & \hat{P} \end{pmatrix}^{-1} \tag{57}$$

$X + J^\kappa\widetilde{x}^\kappa$ 由式 (51) 得来。显然 M 关于 $\widetilde{x}^\kappa$ 的雅可比为：

$$\mathcal{J} = S\begin{pmatrix} H \\ J^\kappa \end{pmatrix} \tag{58}$$

根据高斯牛顿法，每次迭代的增量：

$$\Delta\widetilde{x}^{\kappa} = -(\mathcal{J}^T\mathcal{J})^{-1}\mathcal{J}^T M \tag{59}$$

对其分别推导，其中：

$$\begin{aligned}
(\mathcal{J}^T\mathcal{J})^{-1} &= \left[\begin{pmatrix} H^T & J^{\kappa T}\end{pmatrix}\begin{pmatrix} R^{-1} & \\ & \hat{P}^{-1}\end{pmatrix}\begin{pmatrix} H \\ J^{\kappa}\end{pmatrix}\right]^{-1} \\
&= (J^{\kappa T}\hat{P}^{-1}J^{\kappa} + H^T R^{-1} H)^{-1} \\
&\overset{SMW}{=} P - PH^T(R + HPH^T)^{-1}HP \\
&= (I - KH)P \\
&= KRH^{-T}
\end{aligned} \tag{60}$$

其中 $P = J^{\kappa-1}\hat{P}J^{\kappa-T}$，$K = PH^T(HPH^T + R)^{-1}$。第三行利用了 SMW 公式：

$$(A^{-1} + BD^{-1}C)^{-1} = A - AB(D + CAB)^{-1}CA \tag{61}$$

最后一行推导如下：

$$\begin{aligned}
&K = PH^T(HPH^T + R)^{-1} \\
&K(HPH^T + R) = PH^T \\
&KR - (I - KH)PH^T = 0 \\
&(I - KH)P = KRH^{-T}
\end{aligned} \tag{62}$$

回到式 (59), 后面部分：

$$\begin{aligned}
-\mathcal{J}^T M &= -\begin{pmatrix} H^T & J^{\kappa T}\end{pmatrix}\begin{pmatrix} R^{-1} & \\ & \hat{P}^{-1}\end{pmatrix}\begin{pmatrix} z^{\kappa} + H\widetilde{x}^{\kappa} \\ X + J^{\kappa}\widetilde{x}^{\kappa}\end{pmatrix} \\
&= -H^T R^{-1}(z^{\kappa} + H\widetilde{x}^{\kappa}) - J^{\kappa T}\hat{P}^{-1}(X + J^{\kappa}\widetilde{x}^{\kappa})
\end{aligned} \tag{63}$$

式 (60) 和式 (63) 相乘，得：

$$\begin{aligned}
\Delta\widetilde{x}^{\kappa} &= KRH^{-T}(-H^T R^{-1}(z^{\kappa} + H\widetilde{x}^{\kappa})) - (I - KH)P(J^{\kappa T}\hat{P}^{-1}(X + J^{\kappa}\widetilde{x}^{\kappa})) \\
&= -Kz^{\kappa} - KH\widetilde{x}^{\kappa} - (I - KH)J^{\kappa-1}\hat{P}J^{\kappa-T}(J^{\kappa T}\hat{P}^{-1}(X + J^{\kappa}\widetilde{x}^{\kappa})) \\
&= -Kz^{\kappa} - KH\widetilde{x}^{\kappa} - (I - KH)J^{\kappa-1}(X + J^{\kappa}\widetilde{x}^{\kappa}) \\
&= -Kz^{\kappa} - \widetilde{x}^{\kappa} - (I - KH)J^{\kappa-1}X
\end{aligned} \tag{64}$$

则

$$\widetilde{x}^{\kappa+1} = \widetilde{x}^{\kappa} + \Delta\widetilde{x}^{\kappa} = -Kz^{\kappa} - (I - KH)J^{\kappa-1}(\widehat{\mathbf{x}}^{\kappa} \boxminus \widehat{\mathbf{x}}) \tag{65}$$

推导完毕！可以看出 ESKF 和 GN 法是等效的。

最后，文中提到了一个工程问题，在计算卡尔曼增益时，矩阵 $\mathbf{HPH}^T + \mathbf{R}$ 的维度是很大的 ($m \times m$, 因为 $H_{m\times18}, P_{18\times18}, R_{m\times m}$)，特征点 m 的数量可能为上千个，如此高维的矩阵求逆是十分耗时的，因此文中给出了新的卡尔曼增益计算方式，它和原增益是等效的：

$$\mathbf{K} = \left(\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} + \mathbf{P}^{-1}\right)^{-1}\mathbf{H}^T\mathbf{R}^{-1} \tag{66}$$

证明是易懂的，见文中附录及 [9]。式中 $\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}$ 是 18×18 的，避免了高维矩阵求逆的情况。

3.3.5 The algorithm

算法总结如图 7所示。

Algorithm 1: State Estimation

Input : Last optimal estimation $\bar{\mathbf{x}}_{k-1}$ and $\bar{\mathbf{P}}_{k-1}$,
IMU inputs $(\mathbf{a}_m, \boldsymbol{\omega}_m)$ in current scan;
LiDAR feature points ${}^{L_j}\mathbf{p}_{f_j}$ in current scan.

Forward propagation to obtain state prediction $\widehat{\mathbf{x}}_k$ via (4) and covariance prediction $\widehat{\mathbf{P}}_k$ via (8);
Backward propagation to obtain ${}^{L_k}\mathbf{p}_{f_j}$ via (9), (10);
$\kappa = -1$, $\widehat{\mathbf{x}}_k^{\kappa=0} = \widehat{\mathbf{x}}_k$;
repeat
    $\kappa = \kappa + 1$;
    Compute $\mathbf{J}^{\kappa}$ via (16) and $\mathbf{P} = (\mathbf{J}^{\kappa})^{-1}\widehat{\mathbf{P}}_k(\mathbf{J}^{\kappa})^{-T}$;
    Compute residual $\mathbf{z}_j^{\kappa}$ (12) and Jocobin $\mathbf{H}_j^{\kappa}$ (14);
    Compute the state update $\widehat{\mathbf{x}}_k^{\kappa+1}$ via (18) with the Kalman gain $\mathbf{K}$ from (20);
until $\|\widehat{\mathbf{x}}_k^{\kappa+1} \boxminus \widehat{\mathbf{x}}_k^{\kappa}\| < \epsilon$;
$\bar{\mathbf{x}}_k = \widehat{\mathbf{x}}_k^{\kappa+1}$; $\bar{\mathbf{P}}_k = (\mathbf{I} - \mathbf{KH})\,\mathbf{P}$.

Output: Current optimal estimation $\bar{\mathbf{x}}_k$ and $\bar{\mathbf{P}}_k$.

Figure 7: 算法总结.

3.4 Map Update

地图的更新就是根据估计的最优状态变量 $\bar{\mathbf{x}}_k$(写成 SE(3):${}^{G}\overline{\mathbf{T}}_{I_k} = \left({}^{G}\overline{\mathbf{R}}_{I_k}, {}^{G}\overline{\mathbf{p}}_{I_k}\right)$) 把特征点全部投影到全局坐标系：

$$ {}^{G}\overline{\mathbf{p}}_{f_j} = {}^{G}\overline{\mathbf{T}}_{I_k} {}^{I}\mathbf{T}_{L} {}^{L_k}\mathbf{p}_{f_j}; j = 1, \cdots, m. \tag{67} $$

3.5 Initialization

初始化：在开始时刻保持 2s，利用收集到的数据初始化重力向量，IMU 零偏，噪声等

4 EXPERIMENT RESULTS

4.1 Computational Complexity Experiments

- Table II 两种卡尔曼增益计算效率对比，明显新公式提升较大

4.2 UAV Flight Experiments

- 无人机实验
- 70°FOV Livox Livox Avia LiDAR, DJI Manifold 2-C 1.8 GHz Intel i7-8550U CPU
- 最大 50HZ 的里程计输出 + 建图
- 漂移小于 0.3%（32 米轨迹上的漂移为 0.08 米）。

4.3 Indoor Experiments

- 室内手持快速旋转的场景 (>100°/s)
- 与 LOAM 对比，紧耦合 vs 松耦合

4.4 Outdoor Experiments

- 室外手持，香港大学主楼
- 漂移小于 0.05%(140 米轨迹上的漂移为 0.07 米)，10HZ, 平均处理时间为 25ms，平均有效特征点为 1497 个

- 在 LINS 数据集上比较，FAST-LIO 平均 7.3ms，LINS 平均 34.5ms(个人感觉理论上的差异主要在计算增益 K 的公式上)

References

[1] FAST-LIO: A Fast, Robust LiDAR-inertial Odometry Package by Tightly-Coupled Iterated Kalman Filter.

[2] 流形-Manifold 学习理解与应用. https://www.cnblogs.com/icmzn/p/11082509.html.

[3] Kalman Filters on Differentiable Manifolds.

[4] 从零开始的 IMU 状态模型推导：https://fzheng.me/2016/11/20/imu_model_eq.

[5] 简明 ESKF 推导：https://zhuanlan.zhihu.com/p/441182819

[6] Quaternion kinematics for the error-state Kalman filter.

[7] 迭代扩展卡尔曼滤波 (IEKF)：https://zhuanlan.zhihu.com/p/141018958

[8] FAST-LIO2 简明公式推导:https://zhuanlan.zhihu.com/p/533920262

[9] 关于 FAST-LIO 中使用卡尔曼滤波增益更新求逆简化问题推导: https://blog.csdn.net/weixin_44023934/article/details/123506688