惯性里程计的计算

王永才

目录

- 惯性里程计问题描述
- 姿态解算方法
  - 旋转矩阵表示
  - 四元数表示
- 惯性导航方法
- 偏差Kalman滤波
- 李群和李代数的求导

惯性里程计问题描述

以IMU在0时刻位置和姿态定义世界坐标系，即**IMU初始位置为[0,0,0]，初始姿态角为[0,0,0]**，载体在后续运动过程中通过IMU在IMU坐标系连续检测加速度和角速度。

t时刻测量的自身坐标系的加速度 $\tilde{\mathbf{a}}^i(t) = [a_x,\ a_y,\ a_z]$

t时刻测量的自身坐标系的角速度 $\tilde{\boldsymbol{\omega}}^i(t) = [\omega_x,\ \omega_y,\ \omega_z]$

检测频率均为f，检测时间间隔为$\delta_t=1/f$，并设在δ_t时间内加速度和角速度不变

基于加速度和角速度由0-t时刻的上述测量序列，计算载体t时刻在世界坐标系中的姿态 $\mathbf{T}_{wi}(t) = ?$

$\mathbf{T}_{wi}(t)$ 中包括**姿态** $\mathbf{R}_{wi}(t)$ 和**位置** $\mathbf{p}^w(t)$ $=[x_t,y_t,z_t]^T$

惯性导航系统

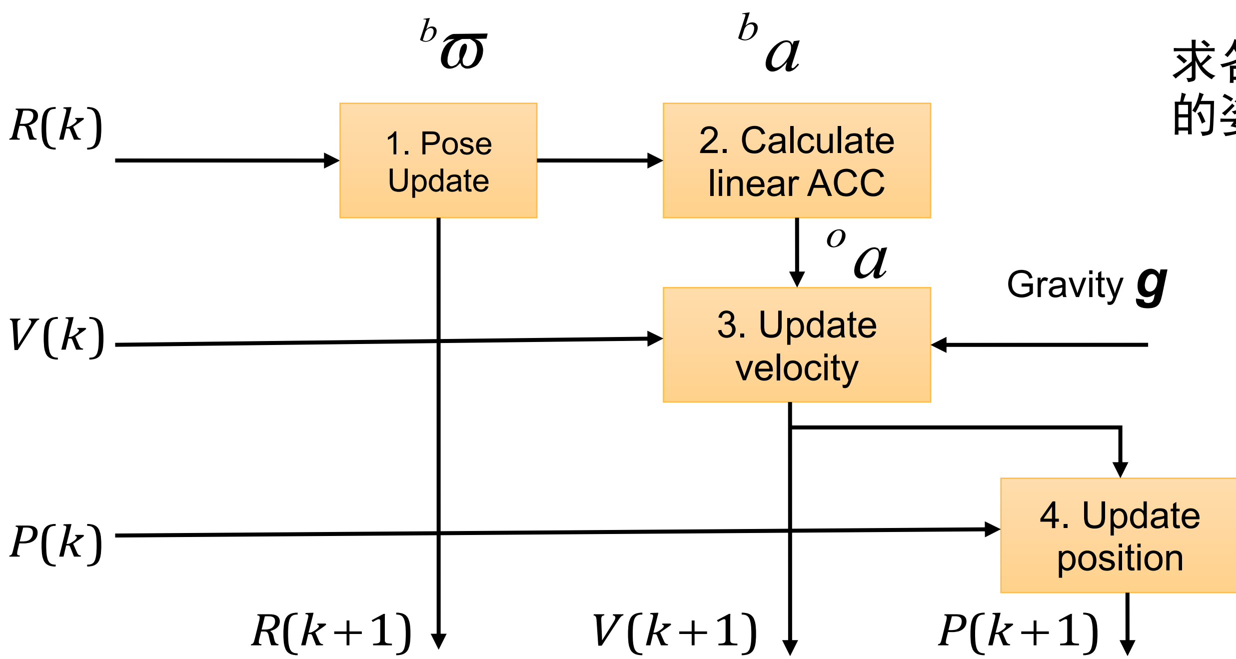
初始状态：$R=I$, t=0开始

- 对象开始运动，陀螺仪提供body frame角速度

$$\varpi=(\omega_x,\omega_y,\omega_z)$$

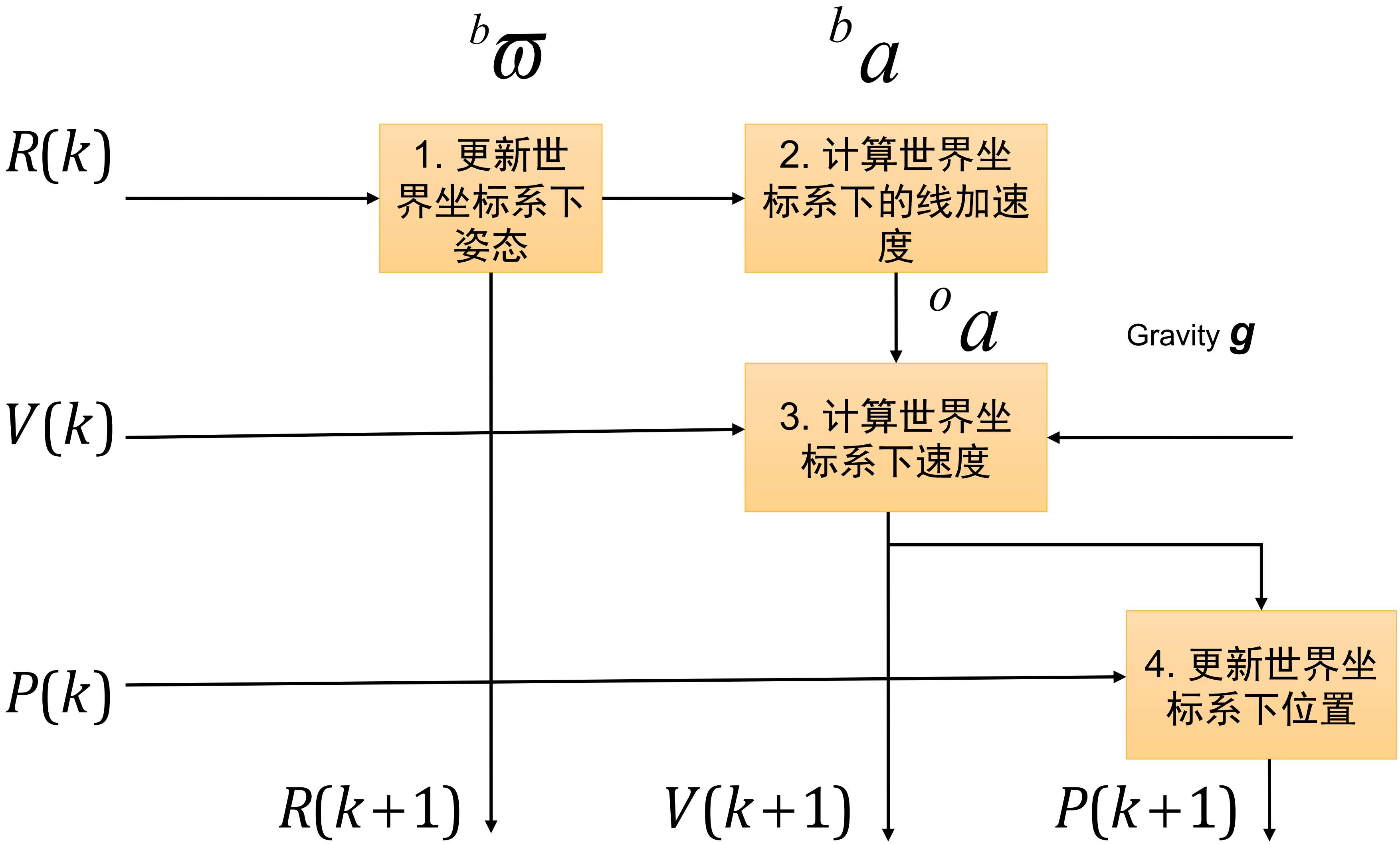
- 加速度传感器提供body frame的加速度信息

$$a=(a_x,a_y,a_z)$$



求各个时刻的设备的姿态和位置

惯性导航流程图



$^{b}\varpi$
^{b}a
$R(k)$
1. 更新世界坐标系下姿态
2. 计算世界坐标系下的线加速度
^{o}a
Gravity $\boldsymbol{g}$
$V(k)$
3. 计算世界坐标系下速度
$P(k)$
4. 更新世界坐标系下位置
$R(k+1)$
$V(k+1)$
$P(k+1)$

姿态更新方法

1. 旋转矩阵表示：指数映射

为表述简练，我们用 $\mathbf{R}(t)$代替$\mathbf{R}_{wi}(t)$，用 $\tilde{\mathbf{a}}^i$ 代替 $\tilde{\mathbf{a}}^i(t)$，用 $\tilde{\boldsymbol{\omega}}^i$ 代替 $\tilde{\boldsymbol{\omega}}^i(t)$

用$\mathbf{R}_t^{t+\delta t}$ 表示 t 到 $t+\delta_t$ 时刻的位姿变换

要求的是 $\mathbf{R}(t+\delta t)=\mathbf{R}(t)\mathbf{R}_t^{t+\delta t}$

我们从 $t=0$, $R(0)=I$开始

在δt这个很短的时间里，设备旋转总是可以描述成绕着某个轴旋转了一定角度

既然测量值是 $\tilde{\boldsymbol{\omega}}^i(t)=[\omega_x, \omega_y, \omega_z]$

把它看做是，绕着 $\tilde{\boldsymbol{\omega}}^i(t_1)/\left|\tilde{\boldsymbol{\omega}}^i(t_1)\right|$ 这个旋转轴，旋转了 $\delta_t\left|\tilde{\boldsymbol{\omega}}^i(t_1)\right|$ 角度

求IMU的姿态变化？ $\mathbf{R}_t^{t+\delta t}\approx\exp\left(\delta_t\cdot\left|\tilde{\boldsymbol{\omega}}^i(t)\right|\left(\tilde{\boldsymbol{\omega}}^i(t)/\left|\tilde{\boldsymbol{\omega}}^i(t)\right|\right)^\wedge\right)=\exp\left(\left(\delta_t\cdot\tilde{\boldsymbol{\omega}}^i(t)\right)^\wedge\right)$

姿态更新方法

2. 旋转矩阵表示：两种罗德里格斯公式

- 根据罗德里格斯公式计算

$$\boldsymbol{R}=\cos\theta\boldsymbol{I}+(1-\cos\theta)\boldsymbol{n}\boldsymbol{n}^T+\sin\theta\boldsymbol{n}^{\wedge}.$$

其中 $\boldsymbol{n}=\tilde{\boldsymbol{\omega}}^i(t_1)/\left|\tilde{\boldsymbol{\omega}}^i(t_1)\right|$ $\boldsymbol{\theta}=\delta_t\left|\tilde{\boldsymbol{\omega}}^i(t_1)\right|$

从而得到旋转矩阵：

$$\mathbf{R}_t^{t+\delta_t}\approx\cos\left(\left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|\right)\mathbf{I}+\left(\mathbf{I}-\cos\left(\left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|\right)\right)\mathbf{n}\mathbf{n}^T+\sin\left(\left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|\right)\cdot(\mathbf{n})^{\wedge}$$

其中 $\mathbf{n}=\dfrac{\delta_t\tilde{\boldsymbol{\omega}}^i(t)}{\left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|}$

另一种罗德里格斯公式：

$$\mathbf{R}_t^{t+\delta t}\approx\begin{cases}I & \text{if } \left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|<10^{-8}\\ I+\dfrac{\sin\left(\left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|\right)}{\left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|}\cdot(\delta_t\tilde{\boldsymbol{\omega}}^i)^{\wedge}+\left(\dfrac{1-\cos\left(\left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|\right)}{\left|\delta_t\tilde{\boldsymbol{\omega}}^i(t)\right|^2}\right)\cdot\left((\delta_t\tilde{\boldsymbol{\omega}}^i(t))^{\wedge}\right)^2 & \text{else}\end{cases}$$

姿态更新方法

3. 角速度矩阵近似方法

- 例如δ_t=0.01，[w_x=0.1, w_y=0.2, w_z=0.3]时，可知两次采样间沿三个轴的body frame转角为[0.001, 0.002, 0.003]，计算：

$$\mathbf{R}_{t-1}^{t} = \mathrm{rotx}(0.001)*\mathrm{roty}(0.002)*\mathrm{rotz}(0.003)$$
$$\mathbf{R}_{t-1}^{t} = \mathrm{roty}(0.002)*\mathrm{rotx}(0.001)*\mathrm{rotz}(0.003)$$

在旋转角度很小时，旋转矩阵是可交换的

$$\mathbf{R}_{t-1}^{t} = \begin{matrix} 1.0000 & -0.0030 & 0.0020 \\ 0.0030 & 1.0000 & -0.0010 \\ -0.0020 & 0.0010 & 1.0000 \end{matrix} = \begin{matrix} 1 & -\delta_t w_z & \delta_t w_y \\ \delta_t w_z & 1 & -\delta_t w_x \\ -\delta_t w_y & \delta_t w_x & 1 \end{matrix}$$

所以在微小旋转的情况下，旋转矩阵更新方法即为

$$\mathbf{R}_{t}^{t+\delta_t} = \delta_t S(\omega) + I$$

$$\mathbf{R}(t+\delta_t) = \mathbf{R}(t)\mathbf{R}_{t}^{t+\delta_t} = \mathbf{R}(t)\left(\delta_t S(\omega) + I\right)$$

角速度矩阵法

4. 四元数表示的姿态更新方法

四元数的时间导数

$$\dot{\mathbf{q}} \triangleq \lim_{\Delta t \to 0} \frac{\mathbf{q}(t+\Delta t) - \mathbf{q}(t)}{\Delta t}$$

[JoanSola2017], 公式(198)

$$= \frac{1}{2}\mathbf{q} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega}_{\mathcal{L}} \end{bmatrix}$$

$\omega_{\mathcal{L}}$ 是局部坐标系的角速度向量

定义:

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}$$

则:

$$\dot{\mathbf{q}} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_{\mathcal{L}})\,\mathbf{q} = \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega}_{\mathcal{L}}$$

注意：世界坐标系姿态与局部坐标系的加速度的转换为:

$$\boldsymbol{\omega}_{\mathcal{G}} = \mathbf{q} \otimes \boldsymbol{\omega}_{\mathcal{L}} \otimes \mathbf{q}^* = \mathbf{R}\,\boldsymbol{\omega}_{\mathcal{L}}$$

[1] Quaternion kinematics for the error-state Kalman filter, Joan Sola, 2017

回顾四元数乘法

$$\mathbf{p} \otimes \mathbf{q} = \begin{bmatrix} p_w q_w - p_x q_x - p_y q_y - p_z q_z \\ p_w q_x + p_x q_w + p_y q_z - p_z q_y \\ p_w q_y - p_x q_z + p_y q_w + p_z q_x \\ p_w q_z + p_x q_y - p_y q_x + p_z q_w \end{bmatrix}$$

$$\mathbf{p} \otimes \mathbf{q} = \begin{bmatrix} p_w q_w - \mathbf{p}_v^\top \mathbf{q}_v \\ p_w \mathbf{q}_v + q_w \mathbf{p}_v + \mathbf{p}_v \times \mathbf{q}_v \end{bmatrix}$$

$$\mathbf{q}_1 \otimes \mathbf{q}_2 = [\mathbf{q}_1]_L \, \mathbf{q}_2$$ 其中 $$[\mathbf{q}]_L = \begin{bmatrix} q_w & -q_x & -q_y & -q_z \\ q_x & q_w & -q_z & q_y \\ q_y & q_z & q_w & -q_x \\ q_z & -q_y & q_x & q_w \end{bmatrix}$$

$$\mathbf{q}_1 \otimes \mathbf{q}_2 = [\mathbf{q}_2]_R \, \mathbf{q}_1 \qquad [\mathbf{q}]_R = \begin{bmatrix} q_w & -q_x & -q_y & -q_z \\ q_x & q_w & q_z & -q_y \\ q_y & -q_z & q_w & q_x \\ q_z & q_y & -q_x & q_w \end{bmatrix}$$

4. 四元数表示：姿态更新方法

- 由前面微分方程的解可得：

$$\boxed{\mathbf{q}_{n+1} = \mathbf{q}_n \otimes \mathbf{q}\{\boldsymbol{\omega}_n \Delta t\}}$$

设这段时间内角速度均值为 $\overline{\boldsymbol{\omega}} = \dfrac{\boldsymbol{\omega}_{n+1} + \boldsymbol{\omega}_n}{2}$ ，则有：

$$\mathbf{q}_{n+1} = \mathbf{q}_n \otimes \mathbf{q}\{\overline{\boldsymbol{\omega}} \Delta t\}$$

四元数乘法可以写成矩阵左乘四元数向量的形式

$$\mathbf{q}_1 \otimes \mathbf{q}_2 = [\mathbf{q}_2]_R \, \mathbf{q}_1$$

其中左乘矩阵为： $$[\mathbf{q}]_R = \begin{bmatrix} q_w & -q_x & -q_y & -q_z \\ q_x & q_w & q_z & -q_y \\ q_y & -q_z & q_w & q_x \\ q_z & q_y & -q_x & q_w \end{bmatrix}$$

姿态更新方法

4. 四元数表示：姿态更新方法

$$\mathbf{q}_{n+1} = \mathbf{q}_n \otimes \mathbf{q}\{\overline{\varpi}\Delta t\}$$

再改写一下：

$$\mathbf{q}(t+\delta_t) \approx \Omega\left(\tilde{\boldsymbol{\omega}}^i(t), \delta_t\right)\mathbf{q}(t)$$

其中：

$$\Omega(\tilde{\omega}^i(t), \delta_t) = \begin{pmatrix} q_{w4} & -q_{w1} & -q_{w2} & -q_{w3} \\ q_{w1} & q_{w4} & q_{w3} & -q_{w2} \\ q_{w2} & -q_{w3} & q_{w4} & q_{w1} \\ q_{w3} & q_{w2} & -q_{w1} & q_{w4} \end{pmatrix}$$

$$q_{w1} = \frac{\tilde{\boldsymbol{\omega}}_x^i(t)}{\left\|\tilde{\boldsymbol{\omega}}^i(t)\right\|}\sin\left(\frac{\left\|\tilde{\boldsymbol{\omega}}^i(t)\delta_t\right\|}{2}\right), q_{w2} = \frac{\tilde{\boldsymbol{\omega}}_y^i}{\left\|\tilde{\boldsymbol{\omega}}^i(t)\right\|}\sin\left(\frac{\left\|\tilde{\boldsymbol{\omega}}^i(t)\delta_t\right\|}{2}\right)$$

$$q_{w3} = \frac{\tilde{\boldsymbol{\omega}}_z^i(t)}{\left\|\tilde{\boldsymbol{\omega}}^i(t)\right\|}\sin\left(\frac{\left\|\tilde{\boldsymbol{\omega}}^i(t)\delta_t\right\|}{2}\right), q_{w4} = \cos\left(\frac{\left\|\tilde{\boldsymbol{\omega}}^i(t)\delta_t\right\|}{2}\right)$$

这是有δt时间的角速度所构造的旋转向量，所对应的四元数

四元数需进行归一化以保持四元数的性质，模长为1

$$\mathbf{q}(t+\delta_t) = \frac{\mathbf{q}(t+\delta_t)}{\left\|\mathbf{q}(t+\delta_t)\right\|}$$

四元数表达的姿态更新方法

4. 四元数和旋转矩阵相互转换

- 四元数姿态表示和旋转矩阵之间可以互相转化

$$\mathbf{R}\left(t+\delta_t\right)=q2dcm\left(\mathbf{q}\left(t+\delta_t\right)\right)$$

若表示 $\mathbf{q}\left(t\right)=\left[b,c,d,a\right]$

$$\mathbf{R}\left(t\right)=q2dcm\left(\mathbf{q}(t)\right)$$

$$\begin{pmatrix} aa+bb-cc-dd & 2\left(bc-ad\right) & 2\left(ac+bd\right) \\ 2\left(ad+bc\right) & aa-bb+cc-dd & 2\left(cd-ab\right) \\ 2\left(bd-ac\right) & 2\left(cd+ab\right) & aa-bb-cc+dd \end{pmatrix}$$

姿态更新的方法总结

1. 角速度反对称矩阵方法

$$\mathbf{R}_t^{t+\delta_t} = \delta_t S(\omega) + I \qquad \mathbf{R}(t+\delta_t) = \mathbf{R}(t)\mathbf{R}_t^{t+\delta_t}$$

2 罗德里格斯公式1： Titterton, Eq. 11.10, p. 312

$$\mathbf{R}_t^{t+\delta t} = \begin{cases} I & \text{if } \ |\delta_t \varpi_t| < 10^{-8} \\ I + \dfrac{\sin(|\delta_t \varpi_t|)}{|\delta_t \varpi_t|} \cdot S(\delta_t \varpi_t) + \left(\dfrac{1-\cos(|\delta_t \varpi_t|)}{|\delta_t \varpi_t|^2} \right) \cdot S^2(\delta_t \varpi_t) & \text{else} \end{cases}$$

3 罗德里格斯公式2： 视觉SLAM 14讲

$$\mathbf{R}_t^{t+\delta t} = \cos\left(\left|\delta_t \varpi_t\right|\right) I + \left(1-\cos\left(\left|\delta_t \varpi_t\right|\right)\right)\mathbf{n}\mathbf{n}^T + \sin\left(\left|\delta_t \varpi_t\right|\right) S(\mathbf{n}), \ \text{where } \mathbf{n} = \frac{\delta_t \varpi_t}{\left|\delta_t \varpi_t\right|}$$

4. 指数映射方法： Groves, Eq. 5.69.

$$\mathbf{R}_t^{t+\delta_t} = \operatorname{expm}\left(S(\delta_t \varpi_t)\right) \qquad \mathbf{R}(t+\delta_t) = \mathbf{R}(t)\mathbf{R}_t^{t+\delta_t}$$

5. 四元数方法：

$$\mathbf{Q}(t+\delta_t) = \Omega(\varpi_t, \delta_t)\mathbf{Q}(t) \qquad \mathbf{Q}(t+\delta_t) = \frac{\mathbf{Q}(t+\delta_t)}{\left\|\mathbf{Q}(t+\delta_t)\right\|} \qquad \mathbf{R}(t+\delta_t) = q2dcm\left(\mathbf{Q}(t+\delta_t)\right)$$

三种姿态更新方法的特点总结

方法	罗德里格斯公式、指数映射方法	四元数方法	角速度反对称矩阵法
特点	计算量适中	计算虽最复杂、但仍然是解析的矩阵计算	计算量最小
适用情况	适合于PDR、视觉惯性里程计、雷达惯性里程计等。	适合于PDR、视觉惯性里程计、雷达惯性里程计等，无Gimbal lock问题，可适用于俯仰角大幅度变化的场景如无人机，手机定位导航等	是一种近似计算，适用于行人计步导航PDR，视觉惯性里程计、雷达惯性里程计等有周期性外部修正信号的场景。

五种方法的对比

- 例如δ_t=0.01， [w_x=0.1, w_y=0.2, w_z=0.3]时， 可知两次采样间沿三个轴的转角为[0.001, 0.002, 0.003]， 计算：

1. 反对称矩阵法

$$R(\delta_t)=\begin{bmatrix} 1.0000 & -0.0030 & 0.0020 \\ 0.0030 & 1.0000 & -0.0010 \\ -0.0020 & 0.0010 & 1.0000 \end{bmatrix}$$

2. 罗德里格斯公式1

$$R(\delta_t)=\begin{bmatrix} 1.0000 & -0.0030 & 0.0020 \\ 0.0030 & 1.0000 & -0.0010 \\ -0.0020 & 0.0010 & 1.0000 \end{bmatrix}$$

3. 罗德里格斯公式2

$$R(\delta_t)=\begin{bmatrix} 1.0000 & -0.0030 & 0.0020 \\ 0.0030 & 1.0000 & -0.0010 \\ -0.0020 & 0.0010 & 1.0000 \end{bmatrix}$$

4. 指数映射

$$R(\delta_t)=\begin{bmatrix} 1.0000 & -0.0030 & 0.0020 \\ 0.0030 & 1.0000 & -0.0010 \\ -0.0020 & 0.0010 & 1.0000 \end{bmatrix}$$

5. 四元数方法

$$R(\delta_t)=\begin{bmatrix} 1.0000 & -0.0030 & 0.0020 \\ 0.0030 & 1.0000 & -0.0010 \\ -0.0020 & 0.0010 & 1.0000 \end{bmatrix}$$

旋转角度小时，五种方法单步结果完全一样

五种方法的对比

- 例如δ_t=0.5，[w_x=0.1, w_y=0.2, w_z=0.3]时，可知两次采样间沿三个轴的转角为[0.05, 0.10, 0.15]，计算100个周期后：

1. 反对称矩阵法

$$R(100\delta_t)=\begin{bmatrix} 0.9988 & -0.0412 & 0.0264 \\ 0.0416 & 0.9990 & -0.0150 \\ -0.0258 & 0.0161 & 0.9995 \end{bmatrix}$$

2. 罗德里格斯公式1

$$R(100\delta_t)=\begin{bmatrix} 0.9907 & 0.1143 & -0.0731 \\ -0.1115 & 0.9929 & 0.0419 \\ 0.0774 & -0.0334 & 0.9964 \end{bmatrix}$$

3. 罗德里格斯公式2

$$R(100\delta_t)=\begin{bmatrix} 0.9907 & 0.1143 & -0.0731 \\ -0.1115 & 0.9929 & 0.0419 \\ 0.0774 & -0.0334 & 0.9964 \end{bmatrix}$$

4. 指数映射

$$R(100\delta_t)=\begin{bmatrix} 0.9907 & 0.1143 & -0.0731 \\ -0.1115 & 0.9929 & 0.0419 \\ 0.0774 & -0.0334 & 0.9964 \end{bmatrix}$$

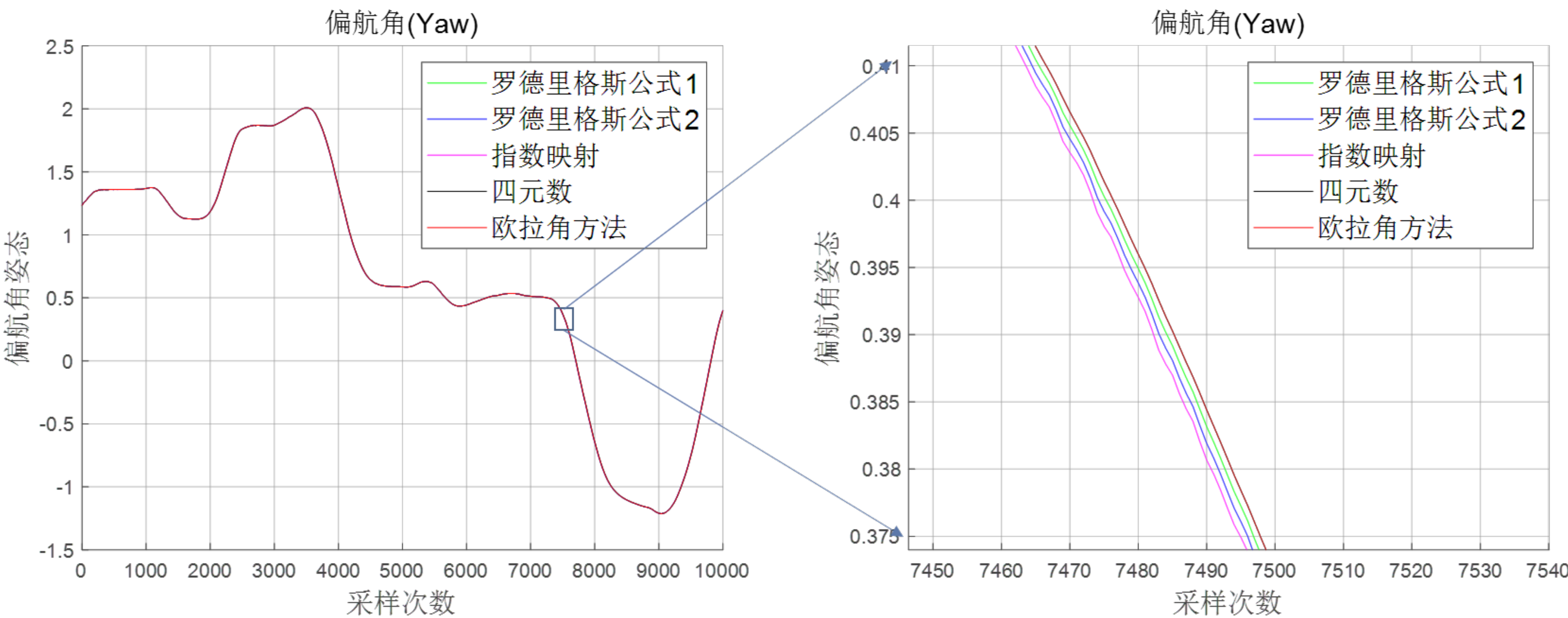
5. 四元数方法

$$R(100\delta_t)=\begin{bmatrix} 0.9907 & 0.1143 & -0.0731 \\ -0.1115 & 0.9929 & 0.0419 \\ 0.0774 & -0.0334 & 0.9964 \end{bmatrix}$$

100个周期后，后四种方法结果完全相同

几种计算方法对比

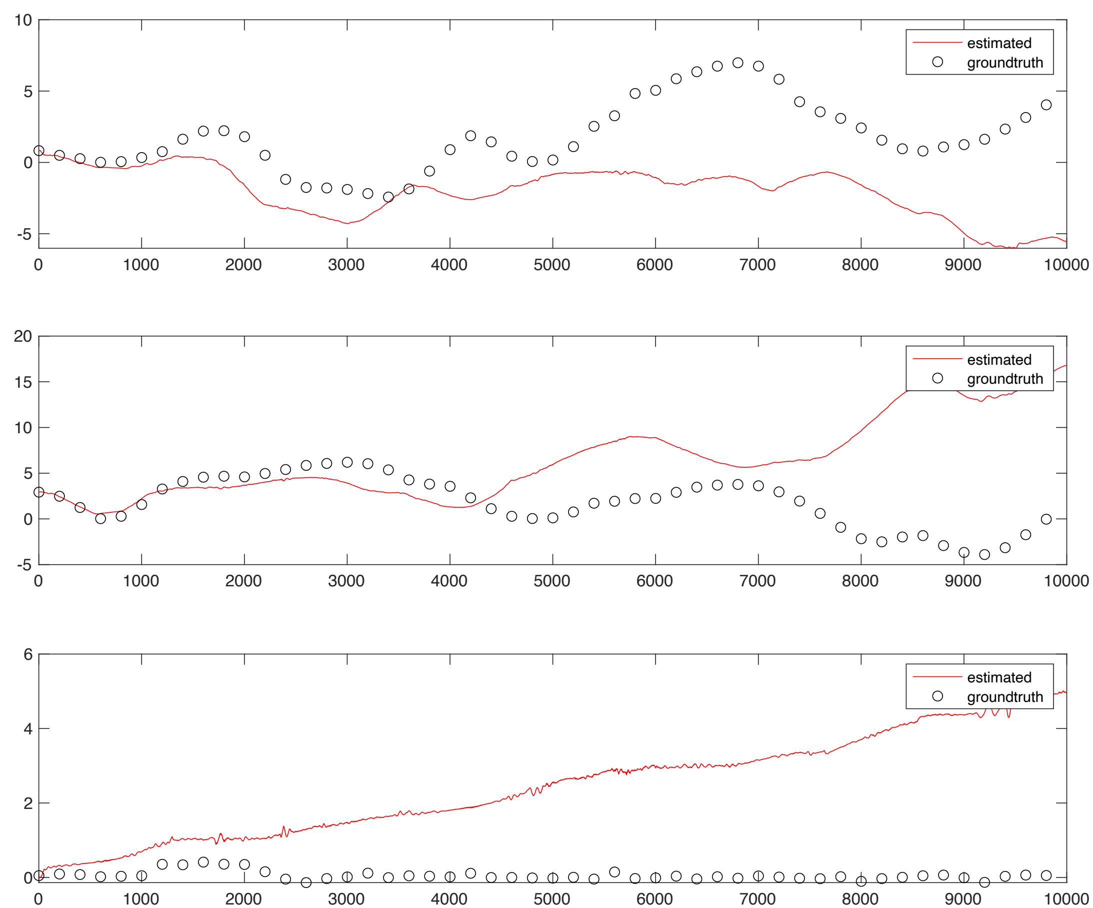
- Ekinox IMU是一种高精度、低漂移、集成高精度姿态解算算法的传感器设备，采样频率为200Hz。
- 计算10分钟之内的位姿变化



几种方法计算效果几乎完全相同

使用比较差的IMU进行姿态解算

- MPU6000是一种手机用的低成本IMU传感器，采样频率为200Hz。
- 计算10分钟之内的位姿变化



很快就出现了
较大的偏差

MEMS陀螺仪的各种噪声

Error type	Description	Result of single integration
Bias	A constant bias offset	A linearly growing error in angle
Random noise	Random noise from a variety of sources (usually specified in terms of its root PSD)	An angle random walk whose standard deviation grows proportional to the square root of time
Calibration	Deterministic errors in scale factors, alignments and gyro linearities	Orientation drift proportional to the rate and duration of motion
Temperature effects	Temperature dependent residual bias	Any residual bias is integrated into the orientation, causing an orientation error which grows linearly with time
Bias instability	Bias fluctuations, usually modelled as a bias random walk	A second-order random walk

$$e_t = \underbrace{N(0,N(0,t))}_{\text{bias}} + \underbrace{N(0,t)}_{\text{Random noise}} + \underbrace{c}_{\text{calibration}} + \underbrace{N(0,Tt)}_{\text{Temperature effects}}$$

目录

惯性导航方法

速度位置更新

- 表示t时刻IMU在自身坐标系采集的真实加速度为 $\mathbf{a}^i(t)$，真实角速度为 $\boldsymbol{\omega}^i(t)$，真实姿态为 $\mathbf{R}_{wi}(t)$，重力加速度为 $\mathbf{g}$
- 通过连续时间的加速度和速度积分，可以得到理想化的速度和位置更新的精确计算公式:

$$\mathbf{a}^w(t) = \mathbf{R}_{wi}(t)\mathbf{a}^i(t) + \mathbf{g}$$

$$\mathbf{v}^w(t) = \int_0^t \mathbf{a}^w(\tau) d\tau$$

$$\mathbf{p}^w(t) = \int_0^t \mathbf{v}^w(\tau) d\tau$$

- 表示IMU的加速度计和陀螺仪t时刻测量偏差（bias）分别为 $\mathbf{b}_a(\mathrm{t})$ 和 $\mathbf{b}_g(\mathrm{t})$，加速度计和陀螺仪测量的噪声表示为 $\mathbf{n}_a(t)$和 $\mathbf{n}_g(t)$

测量值和真实值之间的关系

$$\tilde{\boldsymbol{\omega}}^i(\mathrm{t}) = \boldsymbol{\omega}^i(\mathrm{t}) + \mathbf{b}_g(\mathrm{t}) + \mathbf{n}_g(t)$$

$$\tilde{\mathbf{a}}^i(\mathrm{t}) = \mathbf{R}_{wb}^T(t)\left(\mathbf{a}^w(\mathrm{t}) - \mathbf{g}\right) + \mathbf{b}_a(\mathrm{t}) + \mathbf{n}_a(t)$$

速度位置更新

- 从而速度和位置的更新的精确公式可以表示为：

$$\mathbf{R}_{wi}(t) = \exp\left(\int_0^t \left(\tilde{\boldsymbol{\omega}}^i(\tau) - \mathbf{b}_g(\tau) - \mathbf{n}_g(\tau)\right)^\wedge d\tau\right)$$

$$\mathbf{v}^w(t) = \int_0^t \left(\mathbf{R}_{wi}(\tau)\left(\tilde{\mathbf{a}}^i(\tau) - \mathbf{b}_a(\tau) - \mathbf{n}_a(\tau)\right) + \mathbf{g}\right) d\tau$$

$$\mathbf{p}^w(t) = \int_0^t \mathbf{v}^w(\tau) d\tau$$

$\mathbf{b}_a(\mathrm{t})$, $\mathbf{b}_g(\mathrm{t})$ 也都是随时间变化的，需要在线进行估计

$\mathbf{n}_a(t)$, $\mathbf{n}_g(t)$ 是传感器的随机噪声

速度和位置更新的近似计算方法

- 实际应用IMU的采样是离散的，设IMU采样时间间隔为δ_{t}，则可得到惯性里程计计算近似公式如下：

$$\mathbf{R}_{wi}(t+\delta_{\mathrm{t}}) = \mathbf{R}_{wi}(\mathrm{t})\exp\left[\left(\delta_{\mathrm{t}}\left(\tilde{\boldsymbol{\omega}}^{i}(\mathrm{t})-\mathbf{b}_{g}(t)\right)\right)^{\wedge}\right]$$

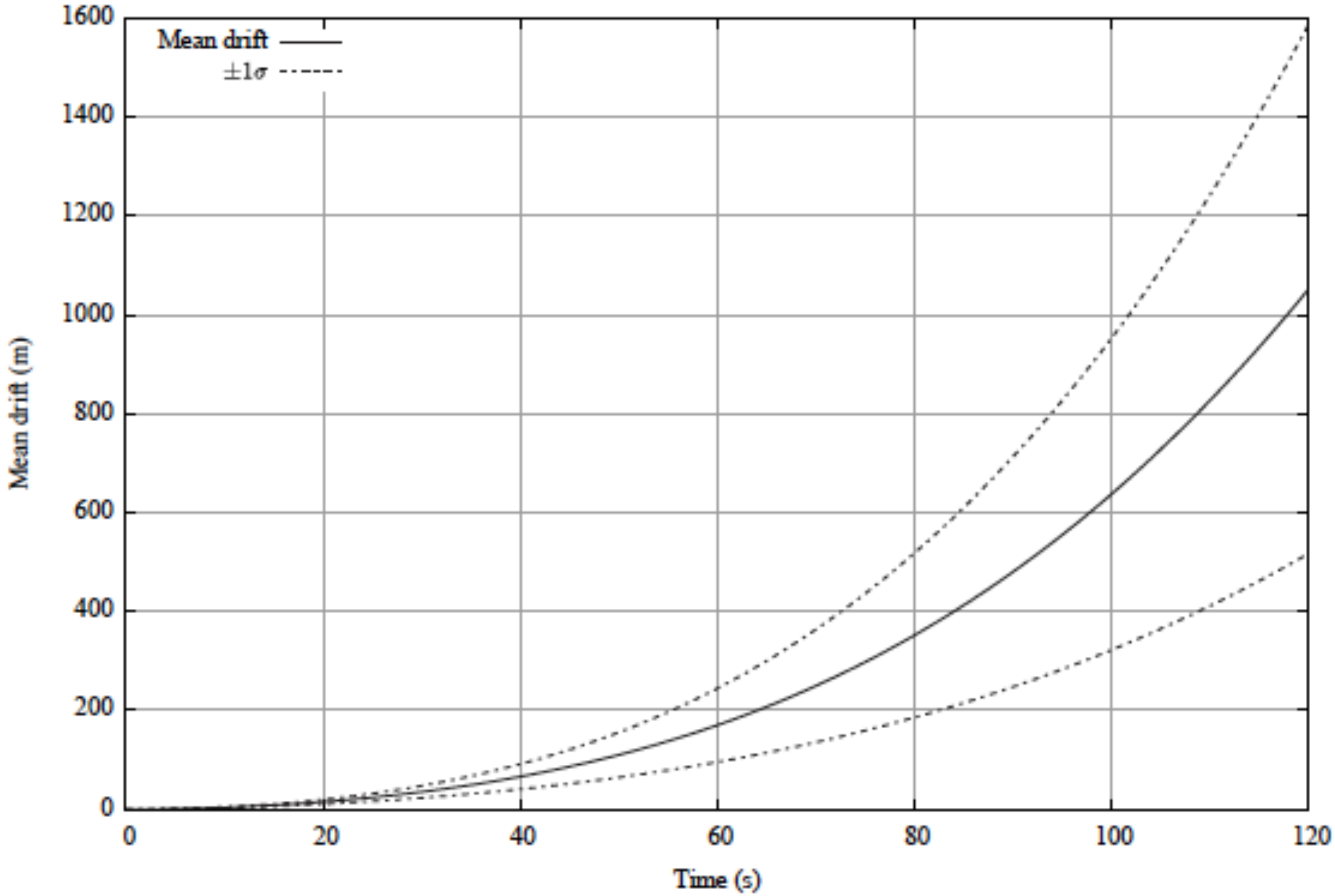
$$\mathbf{v}_{w}(t+\delta_{\mathrm{t}}) = \mathbf{v}_{w}(t)+\mathbf{g}\delta_{\mathrm{t}}+\mathbf{R}_{wi}(\mathrm{t})\left[\tilde{\mathbf{a}}^{i}(\mathrm{t})-\mathbf{b}_{a}(\mathrm{t})\right]\delta_{\mathrm{t}}$$

$$\mathbf{P}_{w}(t+\delta_{\mathrm{t}}) = \mathbf{P}_{w}(t)+\mathbf{v}_{w}(t)\delta_{\mathrm{t}}+\frac{1}{2}\mathbf{g}\delta_{\mathrm{t}}^{2}+\frac{1}{2}\mathbf{R}_{wi}(\mathrm{t})[\tilde{\mathbf{a}}^{i}(\mathrm{t})-\mathbf{b}_{a}(\mathrm{t})]\delta_{\mathrm{t}}^{2}$$

$\mathbf{b}_{g}(\mathrm{t})$ $\mathbf{b}_{a}(\mathrm{t})$ 需要有外部辅助信息才能计算

据经验设置，会导致速度和位置随时间累积产生较大的误差

INS Example



Mean drift
±1σ
Mean drift (m)
Time (s)
0
200
400
600
800
1000
1200
1400
1600
0
20
40
60
80
100
120

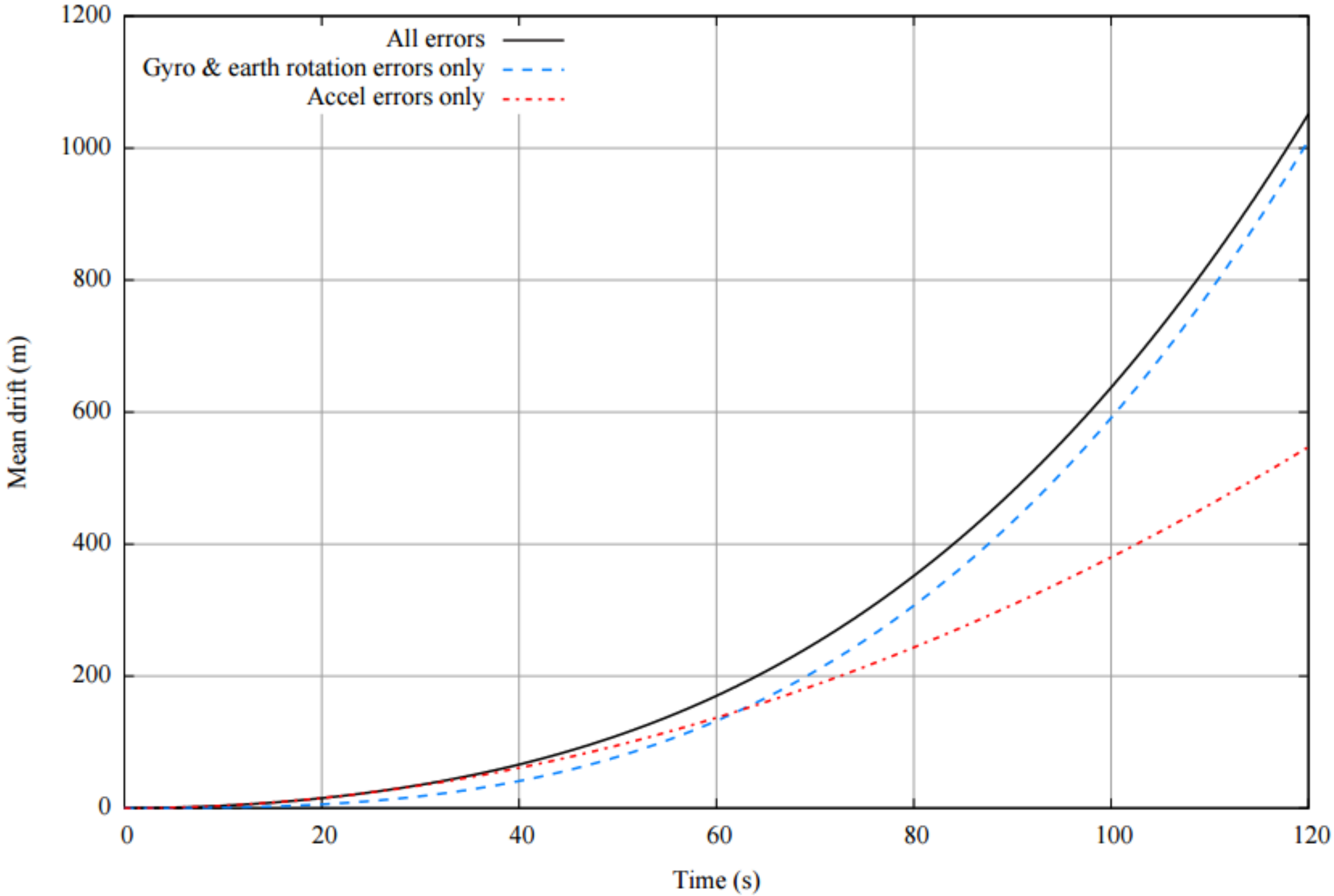


Figure 3.4: The mean drifts incurred when error sources were selectively removed.

目录

惯性导航方法

速度位置更新

- 由于嵌入式加速度传感器、陀螺仪存在时变的测量Bias，前述直接积分的方法产生很大的速度和位置计算的误差。

- 尤其在位置计算中，双重积分会使得偏差快速累计，惯性导航结果很快变差。

- 我们可以利用一些外部观测消除由Bias造成的影响，例如零速检测，加速度传感器和陀螺仪传感器读数都为零时，载体处于静止状态，可以利用这一机会，估计Bias的值，消除由Bias导致的偏差。

- 使用误差状态Kalman滤波器ESKF，来实时追踪速度、位置、Bias的偏差值，在合适的时机修正惯性导航结果。

偏差状态Kalman Filter (ESKF)

- Kalman 滤波器的状态向量定义为

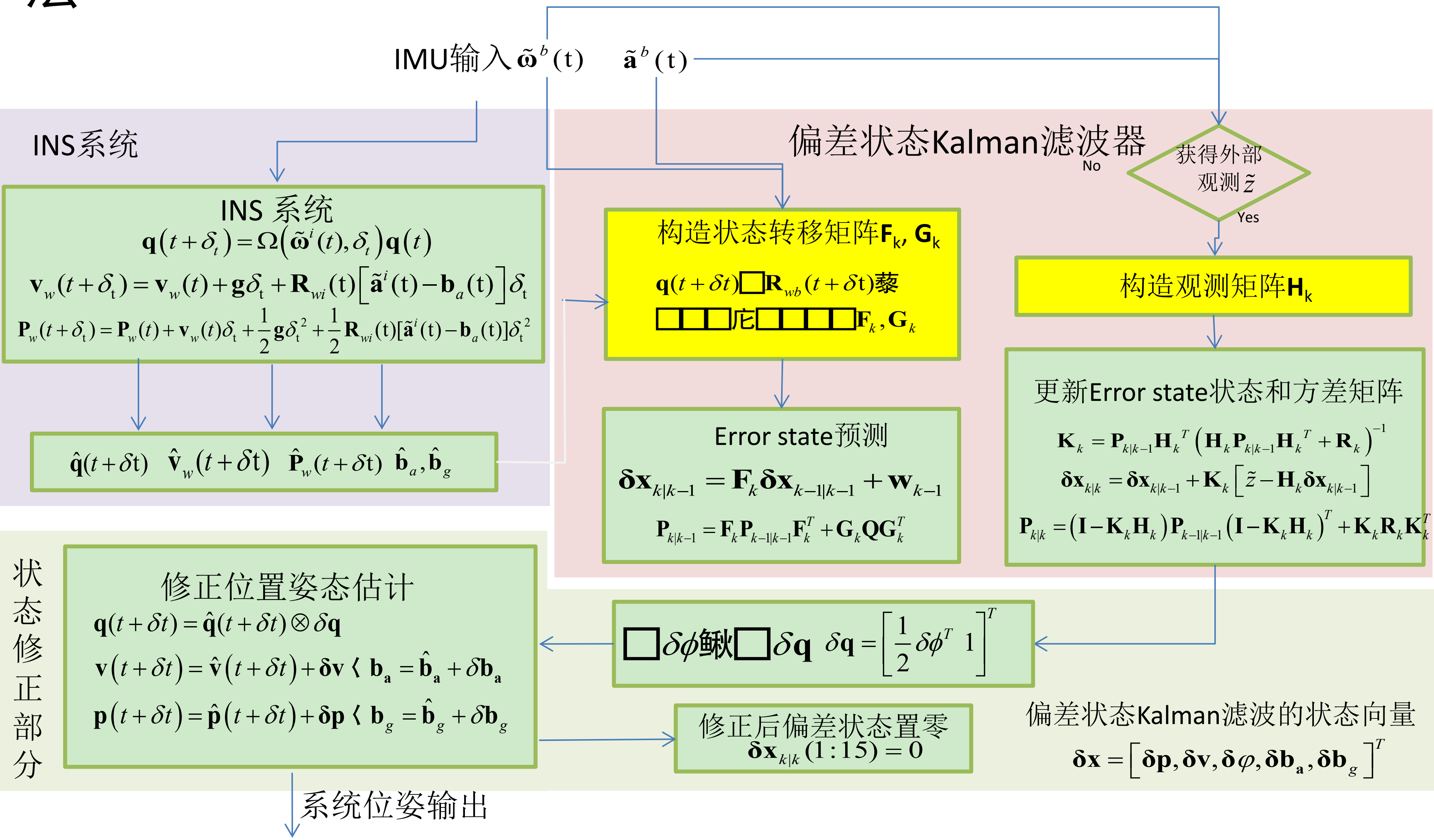
$$\delta x = [\delta p, \delta v, \delta \phi, \delta b_a, \delta b_g]^T$$

追踪的是INS系统的输出与真实状态之间的偏差。

符号表

Magnitude	True	Nominal	Error	Composition	Measured	Noise
Full state (1)	$\mathbf{x}_t$	$\mathbf{x}$	$\delta\mathbf{x}$	$\mathbf{x}_t = \mathbf{x} \oplus \delta\mathbf{x}$		
Position	$\mathbf{p}_t$	$\mathbf{p}$	$\delta\mathbf{p}$	$\mathbf{p}_t = \mathbf{p} + \delta\mathbf{p}$		
Velocity	$\mathbf{v}_t$	$\mathbf{v}$	$\delta\mathbf{v}$	$\mathbf{v}_t = \mathbf{v} + \delta\mathbf{v}$		
Quaternion (2,3)	$\mathbf{q}_t$	$\mathbf{q}$	$\delta\mathbf{q}$	$\mathbf{q}_t = \mathbf{q} \otimes \delta\mathbf{q}$		
Rotation matrix (2,3)	$\mathbf{R}_t$	$\mathbf{R}$	$\delta\mathbf{R}$	$\mathbf{R}_t = \mathbf{R}\,\delta\mathbf{R}$		
Angles vector (4)			$\delta\boldsymbol{\theta}$	$\delta\mathbf{q} = e^{\delta\boldsymbol{\theta}/2}$ $\delta\mathbf{R} = e^{[\delta\boldsymbol{\theta}]_\times}$		
Accelerometer bias	$\mathbf{a}_{bt}$	$\mathbf{a}_b$	$\delta\mathbf{a}_b$	$\mathbf{a}_{bt} = \mathbf{a}_b + \delta\mathbf{a}_b$		$\mathbf{a}_w$
Gyrometer bias	$\boldsymbol{\omega}_{bt}$	$\boldsymbol{\omega}_b$	$\delta\boldsymbol{\omega}_b$	$\boldsymbol{\omega}_{bt} = \boldsymbol{\omega}_b + \delta\boldsymbol{\omega}_b$		$\boldsymbol{\omega}_w$
Gravity vector	$\mathbf{g}_t$	$\mathbf{g}$	$\delta\mathbf{g}$	$\mathbf{g}_t = \mathbf{g} + \delta\mathbf{g}$		
Acceleration	$\mathbf{a}_t$				$\mathbf{a}_m$	$\mathbf{a}_n$
Angular rate	$\boldsymbol{\omega}_t$				$\boldsymbol{\omega}_m$	$\boldsymbol{\omega}_n$

融合外部信息并采用Kalman滤波的状态更新方法



融合外部信息并采用Kalman滤波的状态更新方法

ESKF的INS系统部分

- 整个计算过程中分为三部分：（1）INS系统；（2）Error State Kalman滤波；（3）状态修正部分。

IMU输入 $\tilde{\boldsymbol{\omega}}^b(\mathrm{t})$ $\tilde{\mathbf{a}}^b(\mathrm{t})$

INS系统

$$\mathbf{q}(t+\delta_t)=\Omega\left(\tilde{\boldsymbol{\omega}}^i(t),\delta_t\right)\mathbf{q}(t)$$

$$\mathbf{v}_w(t+\delta_t)=\mathbf{v}_w(t)+\mathbf{g}\delta_t+\mathbf{R}_{wi}(\mathrm{t})\left[\tilde{\mathbf{a}}^i(\mathrm{t})-\mathbf{b}_a(\mathrm{t})\right]\delta_t$$

$$\mathbf{P}_w(t+\delta_t)=\mathbf{P}_w(t)+\mathbf{v}_w(t)\delta_t+\frac{1}{2}\mathbf{g}\delta_t^{\,2}+\frac{1}{2}\mathbf{R}_{wi}(\mathrm{t})[\tilde{\mathbf{a}}^i(\mathrm{t})-\mathbf{b}_a(\mathrm{t})]\delta_t^{\,2}$$

$$\hat{\mathbf{q}}(t+\delta\mathrm{t}) \quad \hat{\mathbf{v}}_w(t+\delta\mathrm{t}) \quad \hat{\mathbf{P}}_w(t+\delta\mathrm{t}) \quad \hat{\mathbf{b}}_a,\hat{\mathbf{b}}_g$$

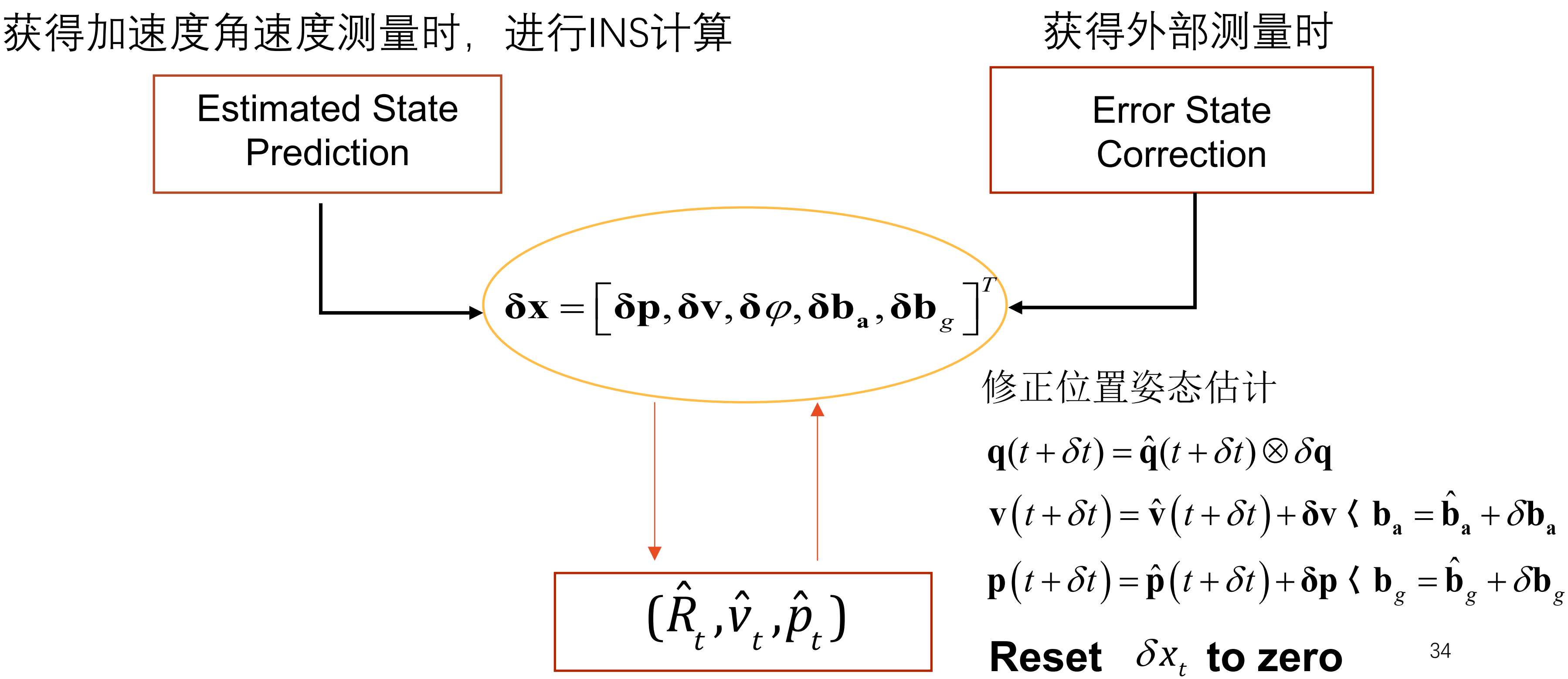
偏差状态Kalman滤波器

ESKF的总体介绍

- Kalman滤波器的状态向量定义为：

$$\boldsymbol{\delta}\mathbf{x} = \left[\boldsymbol{\delta}\mathbf{p}, \boldsymbol{\delta}\mathbf{v}, \boldsymbol{\delta}\varphi, \boldsymbol{\delta}\mathbf{b}_{\mathbf{a}}, \boldsymbol{\delta}\mathbf{b}_{g}\right]^{T}$$

- 即追踪的是INS的估计值和真实值的偏差和Bias项的变化量。



总结

- 惯性导航分为如下几个关键步骤:
  - 姿态解算（有多种方法）

$$\mathbf{R}_{wi}(t+\delta_{\mathrm{t}})=\mathbf{R}_{wi}(\mathrm{t})\exp\left[\left(\delta_{\mathrm{t}}\left(\tilde{\boldsymbol{\omega}}^{i}(\mathrm{t})-\mathbf{b}_{g}(t)\right)\right)^{\wedge}\right]$$

  - 速度和位置计算:

$$\mathbf{v}_{w}(t+\delta_{\mathrm{t}})=\mathbf{v}_{w}(t)+\mathbf{g}\delta_{\mathrm{t}}+\mathbf{R}_{wi}(\mathrm{t})\left[\tilde{\mathbf{a}}^{i}(\mathrm{t})-\mathbf{b}_{a}(\mathrm{t})\right]\delta_{\mathrm{t}}$$

$$\mathbf{P}_{w}(t+\delta_{\mathrm{t}})=\mathbf{P}_{w}(t)+\mathbf{v}_{w}(t)\delta_{\mathrm{t}}+\frac{1}{2}\mathbf{g}\delta_{\mathrm{t}}^{2}+\frac{1}{2}\mathbf{R}_{wi}(\mathrm{t})[\tilde{\mathbf{a}}^{i}(\mathrm{t})-\mathbf{b}_{a}(\mathrm{t})]\delta_{\mathrm{t}}^{2}$$

  - 偏差状态更新，基于零速检测或视觉里程计信息的偏差状态估计

$$\boldsymbol{\delta}\mathbf{x}=\left[\boldsymbol{\delta}\mathbf{p},\boldsymbol{\delta}\mathbf{v},\boldsymbol{\delta}\varphi,\boldsymbol{\delta}\mathbf{b}_{\mathbf{a}},\boldsymbol{\delta}\mathbf{b}_{g}\right]^{T}$$

偏差状态Kalman滤波的整体流程

惯性导航的运动学方程

$$\dot{\mathbf{p}}_t = \mathbf{v}_t$$

位置的导数为速度

$$\dot{\mathbf{v}}_t = \mathbf{a}_t$$

速度的导数为加速度

$$\dot{\mathbf{q}}_t = \frac{1}{2}\mathbf{q}_t \otimes \boldsymbol{\omega}_t$$

姿态四元数的导数

$$\dot{\mathbf{a}}_{bt} = \mathbf{a}_w$$

加速度bias的导数为加速度噪声

$$\dot{\boldsymbol{\omega}}_{bt} = \boldsymbol{\omega}_w$$

角速度bias的导数为角速度噪声

$$\dot{\mathbf{g}}_t = 0$$

重力加速度的导数为0

偏差状态Kalman滤波的整体流程

ESKF的运动学模型

bodyframe测量值和worldframe真实角速度角速度值之间的关系

$$\mathbf{a}_m = \mathbf{R}_t^\top(\mathbf{a}_t - \mathbf{g}_t) + \mathbf{a}_{bt} + \mathbf{a}_n$$
$$\boldsymbol{\omega}_m = \boldsymbol{\omega}_t + \boldsymbol{\omega}_{bt} + \boldsymbol{\omega}_n$$

worldframe加速度角速度的真实值用bodyframe测量值和当前状态描述

$$\mathbf{a}_t = \mathbf{R}_t(\mathbf{a}_m - \mathbf{a}_{bt} - \mathbf{a}_n) + \mathbf{g}_t$$
$$\boldsymbol{\omega}_t = \boldsymbol{\omega}_m - \boldsymbol{\omega}_{bt} - \boldsymbol{\omega}_n.$$

重写前面的运动学方程

$$\dot{\mathbf{p}}_t = \mathbf{v}_t$$
$$\dot{\mathbf{v}}_t = \mathbf{R}_t(\mathbf{a}_m - \mathbf{a}_{bt} - \mathbf{a}_n) + \mathbf{g}_t$$

将a_t用测量值表达式替换

$$\dot{\mathbf{q}}_t = \frac{1}{2}\mathbf{q}_t \otimes (\boldsymbol{\omega}_m - \boldsymbol{\omega}_{bt} - \boldsymbol{\omega}_n)$$

将ω_t用测量值表达式替换

$$\dot{\mathbf{a}}_{bt} = \mathbf{a}_w$$
$$\dot{\boldsymbol{\omega}}_{bt} = \boldsymbol{\omega}_w$$
$$\dot{\mathbf{g}}_t = 0$$

偏差状态Kalman滤波的整体流程

ESKF的运动学模型

标准状态下的运动学模型，去掉加速度和角速度的噪声项

$$
\begin{aligned}
\dot{\mathbf{p}} &= \mathbf{v} \\
\dot{\mathbf{v}} &= \mathbf{R}(\mathbf{a}_m - \mathbf{a}_b) + \mathbf{g} \\
\dot{\mathbf{q}} &= \frac{1}{2}\mathbf{q} \otimes (\boldsymbol{\omega}_m - \boldsymbol{\omega}_b) \\
\dot{\mathbf{a}}_b &= 0 \\
\dot{\boldsymbol{\omega}}_b &= 0 \\
\dot{\mathbf{g}} &= 0.
\end{aligned}
$$

偏差状态下的运动学模型，建立偏差状态之间的运动学方程

$$
\begin{aligned}
\dot{\delta\mathbf{p}} &= \delta\mathbf{v} \\
\dot{\delta\mathbf{v}} &= -\mathbf{R}\left[\mathbf{a}_m - \mathbf{a}_b\right]_\times \delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g} - \mathbf{R}\mathbf{a}_n \\
\dot{\delta\boldsymbol{\theta}} &= -\left[\boldsymbol{\omega}_m - \boldsymbol{\omega}_b\right]_\times \delta\boldsymbol{\theta} - \delta\boldsymbol{\omega}_b - \boldsymbol{\omega}_n \\
\dot{\delta\mathbf{a}}_b &= \mathbf{a}_w \\
\dot{\delta\boldsymbol{\omega}}_b &= \boldsymbol{\omega}_w \\
\dot{\delta\mathbf{g}} &= 0.
\end{aligned}
$$

ESKF追踪的是偏差状态量

$$
\delta\mathbf{x} = \left[\delta\mathbf{p}, \delta\mathbf{v}, \delta\boldsymbol{\theta}, \delta\mathbf{a}_b, \delta\boldsymbol{\omega}_b\right]
$$

推导详见[JoanSola2017, 57-59页]

偏差状态Kalman滤波的整体流程

ESKF的运动方程的紧凑模式

考虑两次测量之间的时间间隔为Δt，则上述微分方程可以求解为：

$$
\begin{aligned}
\delta\mathbf{p} &\leftarrow \delta\mathbf{p} + \delta\mathbf{v}\,\Delta t \\
\delta\mathbf{v} &\leftarrow \delta\mathbf{v} + (-\mathbf{R}\left[\mathbf{a}_m - \mathbf{a}_b\right]_\times \delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g})\Delta t + \mathbf{v_i} \\
\delta\boldsymbol{\theta} &\leftarrow \mathbf{R}^\top\{(\boldsymbol{\omega}_m - \boldsymbol{\omega}_b)\Delta t\}\delta\boldsymbol{\theta} - \delta\boldsymbol{\omega}_b\Delta t + \boldsymbol{\theta}_\mathbf{i} \\
\delta\mathbf{a}_b &\leftarrow \delta\mathbf{a}_b + \mathbf{a_i} \\
\delta\boldsymbol{\omega}_b &\leftarrow \delta\boldsymbol{\omega}_b + \boldsymbol{\omega}_\mathbf{i} \\
\delta\mathbf{g} &\leftarrow \delta\mathbf{g}\ .
\end{aligned}
$$

做如下向量表示：

$$
\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{q} \\ \mathbf{a}_b \\ \boldsymbol{\omega}_b \\ \mathbf{g} \end{bmatrix} \quad , \quad \delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p} \\ \delta\mathbf{v} \\ \delta\boldsymbol{\theta} \\ \delta\mathbf{a}_b \\ \delta\boldsymbol{\omega}_b \\ \delta\mathbf{g} \end{bmatrix} \quad , \quad \mathbf{u}_m = \begin{bmatrix} \mathbf{a}_m \\ \boldsymbol{\omega}_m \end{bmatrix} \quad , \quad \mathbf{i} = \begin{bmatrix} \mathbf{v_i} \\ \boldsymbol{\theta}_\mathbf{i} \\ \mathbf{a_i} \\ \boldsymbol{\omega}_\mathbf{i} \end{bmatrix}
$$

则上述误差状态的运动方程可以写作：

$$
\delta\mathbf{x} \leftarrow f(\mathbf{x}, \delta\mathbf{x}, \mathbf{u}_m, \mathbf{i}) = \mathbf{F_x}(\mathbf{x}, \mathbf{u}_m)\cdot\delta\mathbf{x} + \mathbf{F_i}\cdot\mathbf{i},
$$

偏差状态Kalman滤波的整体流程

ESKF的状态预测方程

为将偏差状态运动方程写成紧凑模式，考虑如下状态向量:

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{q} \\ \mathbf{a}_b \\ \boldsymbol{\omega}_b \\ \mathbf{g} \end{bmatrix} \quad , \quad \delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p} \\ \delta\mathbf{v} \\ \delta\boldsymbol{\theta} \\ \delta\mathbf{a}_b \\ \delta\boldsymbol{\omega}_b \\ \delta\mathbf{g} \end{bmatrix} \quad , \quad \mathbf{u}_m = \begin{bmatrix} \mathbf{a}_m \\ \boldsymbol{\omega}_m \end{bmatrix} \quad , \quad \mathbf{i} = \begin{bmatrix} \mathbf{v_i} \\ \boldsymbol{\theta}_\mathbf{i} \\ \mathbf{a_i} \\ \boldsymbol{\omega}_\mathbf{i} \end{bmatrix}$$

$$\delta\mathbf{x} \leftarrow f(\mathbf{x}, \delta\mathbf{x}, \mathbf{u}_m, \mathbf{i}) = \mathbf{F_x}(\mathbf{x}, \mathbf{u}_m) \cdot \delta\mathbf{x} + \mathbf{F_i} \cdot \mathbf{i},$$

ESKF的状态预测方程写为：

$$\hat{\delta\mathbf{x}} \leftarrow \mathbf{F_x}(\mathbf{x}, \mathbf{u}_m) \cdot \hat{\delta\mathbf{x}}$$

$$\mathbf{P} \leftarrow \mathbf{F_x} \, \mathbf{P} \, \mathbf{F_x}^\top + \mathbf{F_i} \, \mathbf{Q_i} \, \mathbf{F_i}^\top \quad ,$$

将偏差状态用高斯分布描述，方差矩阵为P.
F_x and F_i分别是$F_x(x,u_m)$对x和对i的雅克比矩阵

偏差状态Kalman滤波的整体流程

ESKF的状态预测方程

$$\hat{\delta\mathbf{x}} \leftarrow \mathbf{F_x}(\mathbf{x}, \mathbf{u}_m) \cdot \hat{\delta\mathbf{x}}$$

$$\mathbf{P} \leftarrow \mathbf{F_x}\,\mathbf{P}\,\mathbf{F_x}^\top + \mathbf{F_i}\,\mathbf{Q_i}\,\mathbf{F_i}^\top \quad ,$$

$$\mathbf{F_x} = \left.\frac{\partial f}{\partial \delta\mathbf{x}}\right|_{\mathbf{x},\mathbf{u}_m} = \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & 0 & 0 & 0 & 0 \\ 0 & \mathbf{I} & -\mathbf{R}\left[\mathbf{a}_m - \mathbf{a}_b\right]_\times \Delta t & -\mathbf{R}\Delta t & 0 & \mathbf{I}\Delta t \\ 0 & 0 & \mathbf{R}^\top\{(\boldsymbol{\omega}_m - \boldsymbol{\omega}_b)\Delta t\} & 0 & -\mathbf{I}\Delta t & 0 \\ 0 & 0 & 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{I} \end{bmatrix}$$

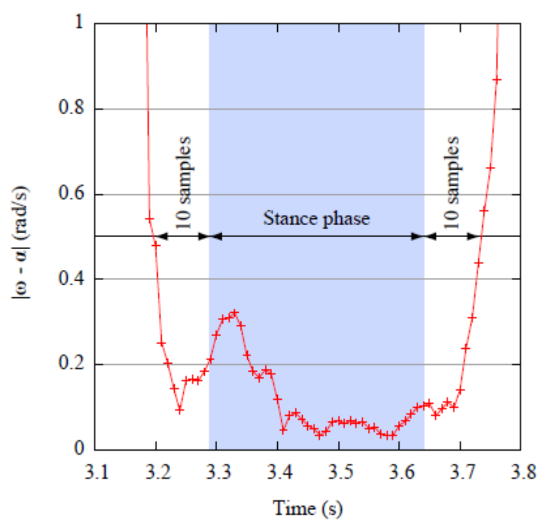
$$\mathbf{F_i} = \left.\frac{\partial f}{\partial \mathbf{i}}\right|_{\mathbf{x},\mathbf{u}_m} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \mathbf{I} & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & \mathbf{I} \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad , \quad \mathbf{Q_i} = \begin{bmatrix} \mathbf{V_i} & 0 & 0 & 0 \\ 0 & \boldsymbol{\Theta}_\mathbf{i} & 0 & 0 \\ 0 & 0 & \mathbf{A_i} & 0 \\ 0 & 0 & 0 & \boldsymbol{\Omega}_\mathbf{i} \end{bmatrix} .$$

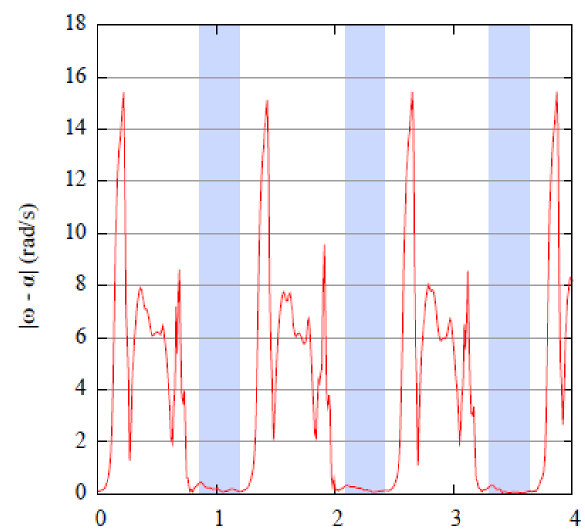
偏差状态Kalman滤波器：状态更新

- 根据外部观测进行偏差状态更新
- 在没有外部观测的时候，一直只进行状态预测的计算，**在获得外部观测的时候，通过外部观测对偏差状态进行更新**。
- **例如，以“零速检测”作为外部观测的偏差状态Kalman滤波**

从连续采集的IMU数据中，进行零速事件检测：**Zero Velocity detection: ZUPT，一般用于行人定位导航中。**

$$|\boldsymbol{\omega}_t^b - \hat{\alpha}_t| < 0.5\,\mathrm{rad/s}\,.$$





偏差状态Kalman滤波器：零速检测

- 在仅使用IMU时，基于IMU本身状态可以获取的外部观测信息是**“零速检测”（Zero Velocity Update: ZUPT）**[9][12]。当IMU固定在鞋子上行走时，在每一步脚掌落地的时候，IMU都会在短暂时刻处于速度为零的状态。
- 文献[9]因此提出零速检测手段，**通过设计零速检测的阈值函数，检测出IMU处于零速的状态，作为外部观测**，建立对速度的观测方程，再通过偏差状态Kalman滤波修正惯性里程计的速度、位置、和bias的追踪结果。
- 由于IMU处于零速时，加速度和角速度的数值都会很小，所以零速检测主要取当前时刻之前一个**滑窗*W***内的加速度、陀螺仪的幅度等判断IMU是否处于零速的状态，有多种零速检测的方法被提出[9] 。

例如最简单的： $|\boldsymbol{\omega}_t^b - \hat{\boldsymbol{\alpha}}_t| < 0.5\,\mathrm{rad/s}\,.$

$$is\_stance = zupt(\tilde{\mathbf{a}}^i, \tilde{\boldsymbol{\omega}}^i, W)$$

偏差状态Kalman滤波器：零速检测

- 当ZUPT检测IMU处于零速时，由于取观测向量为当前的速度估计与零速之间的偏差，即 $\tilde{\mathbf{z}} = \mathbf{v}(t+\delta t) - \mathbf{0}$
- 这是对 $\delta\mathbf{v}$ 的观测，因此观测矩阵

$$\mathbf{H}_k = \left[\mathbf{0}_{3\times3}, \mathbf{I}_{3\times3}, \mathbf{0}_{3\times3}, \mathbf{0}_{3\times3}, \mathbf{0}_{3\times3}\right]$$

- 然后根据观测对偏差状态进行更新。

$$\boldsymbol{\delta}\mathbf{x}_{k|k} = \boldsymbol{\delta}\mathbf{x}_{k|k-1} + \mathbf{K}_k\left[\tilde{z} - \mathbf{H}_k\boldsymbol{\delta}\mathbf{x}_{k|k-1}\right]$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k{}^T\left(\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k{}^T + \mathbf{R}_k\right)^{-1}$$

$$\mathbf{P}_{k|k} = \left(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k\right)\mathbf{P}_{k-1|k-1}\left(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k\right)^T + \mathbf{K}_k\mathbf{R}_k\mathbf{K}_k^T$$

更新后的偏差状态，修正INS结果

□$\delta\phi$鳅□$\delta\mathbf{q}$ $\qquad \delta\mathbf{q} = \begin{bmatrix} \frac{1}{2}\delta\phi^T & 1 \end{bmatrix}^T$

修正位置姿态估计

$$\mathbf{q}(t+\delta t) = \hat{\mathbf{q}}(t+\delta t) \otimes \delta\mathbf{q}$$

$$\mathbf{v}\left(t+\delta t\right) = \hat{\mathbf{v}}\left(t+\delta t\right) + \boldsymbol{\delta}\mathbf{v} \; \langle \; \mathbf{b}_{\mathbf{a}} = \hat{\mathbf{b}}_{\mathbf{a}} + \delta\mathbf{b}_{\mathbf{a}}$$

$$\mathbf{p}\left(t+\delta t\right) = \hat{\mathbf{p}}\left(t+\delta t\right) + \boldsymbol{\delta}\mathbf{p} \; \langle \; \mathbf{b}_{g} = \hat{\mathbf{b}}_{g} + \delta\mathbf{b}_{g}$$

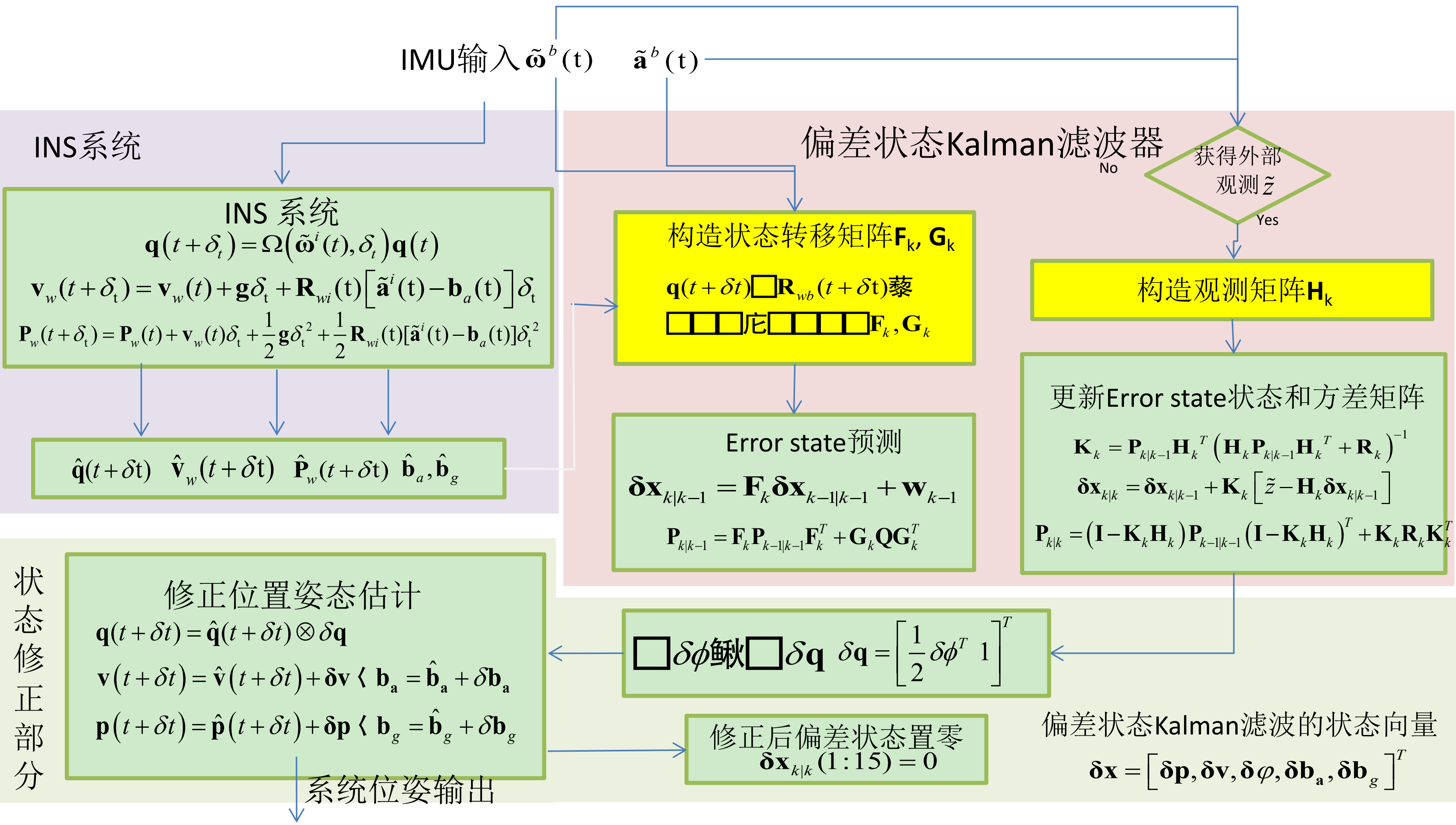
之后把偏差状态重新置为0
将方差矩阵也调整一下

$$\hat{\delta\mathbf{x}} \leftarrow 0$$

$$\mathbf{P} \leftarrow \mathbf{G}\,\mathbf{P}\,\mathbf{G}^\top$$

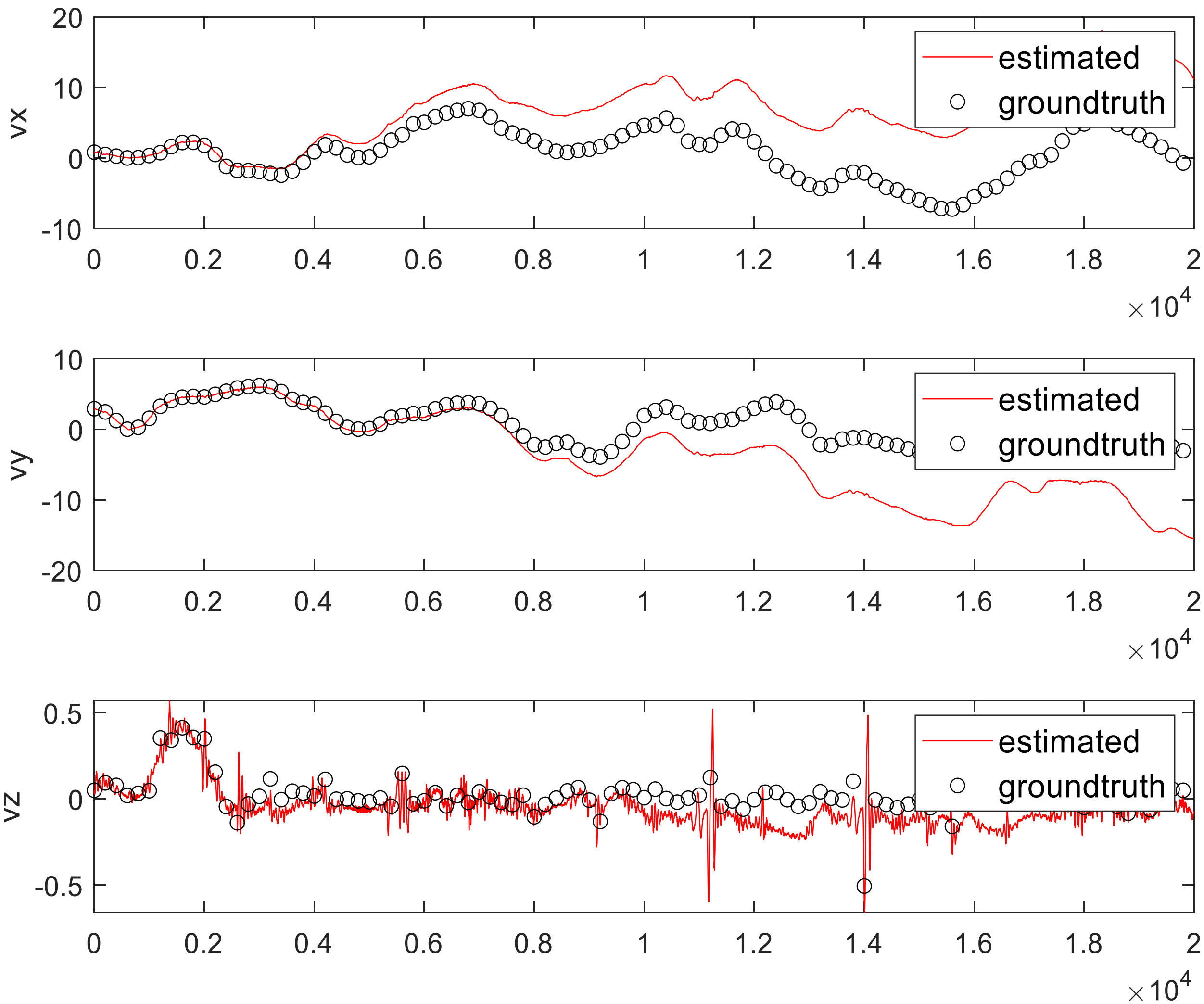
$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_6 & 0 & 0 \\ 0 & \mathbf{I} - \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times & 0 \\ 0 & 0 & \mathbf{I}_9 \end{bmatrix}$$

融合外部信息并采用Kalman滤波的状态更新方法



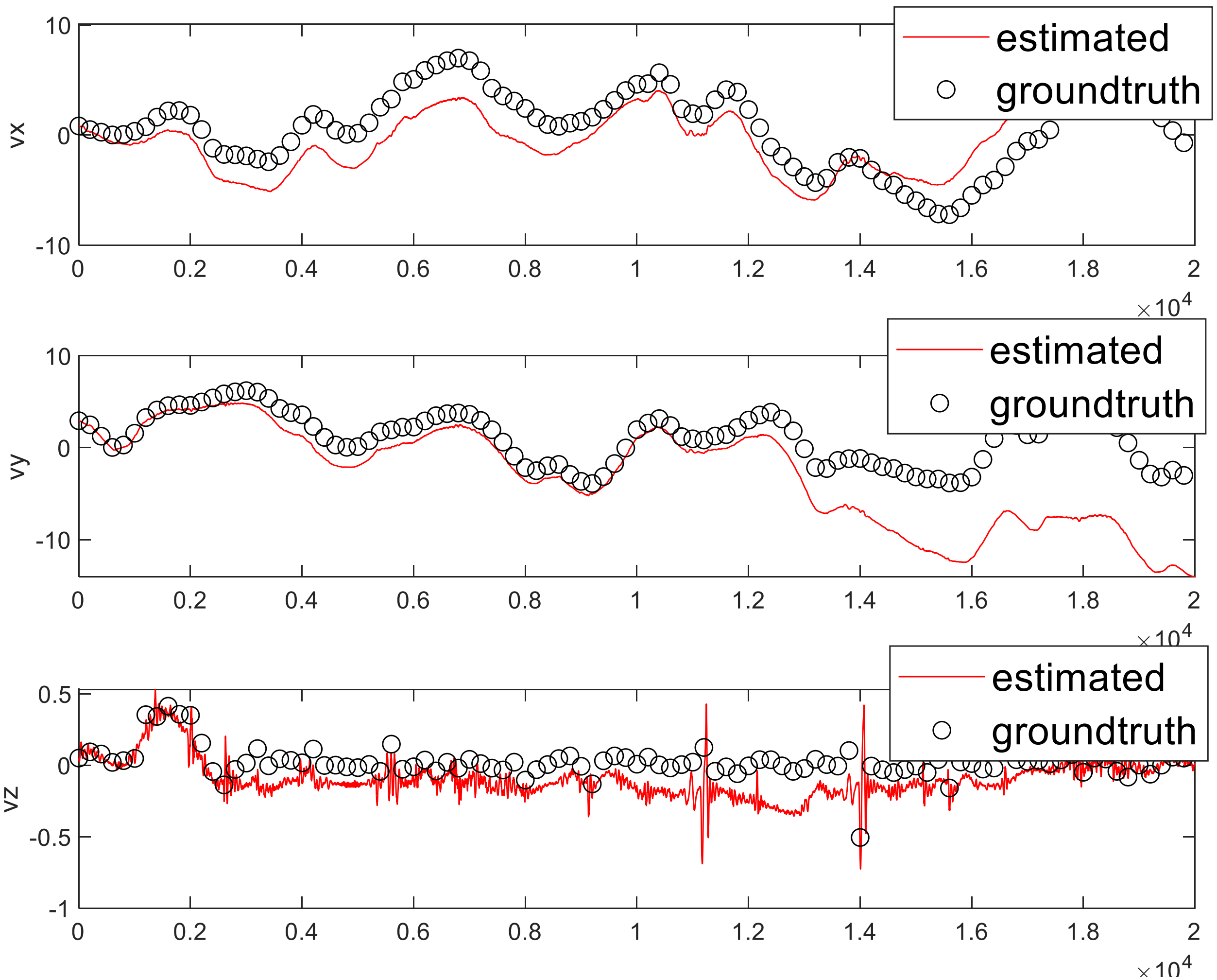
速度与位置追踪的实验结果

Bias设为0时的速度追踪结果



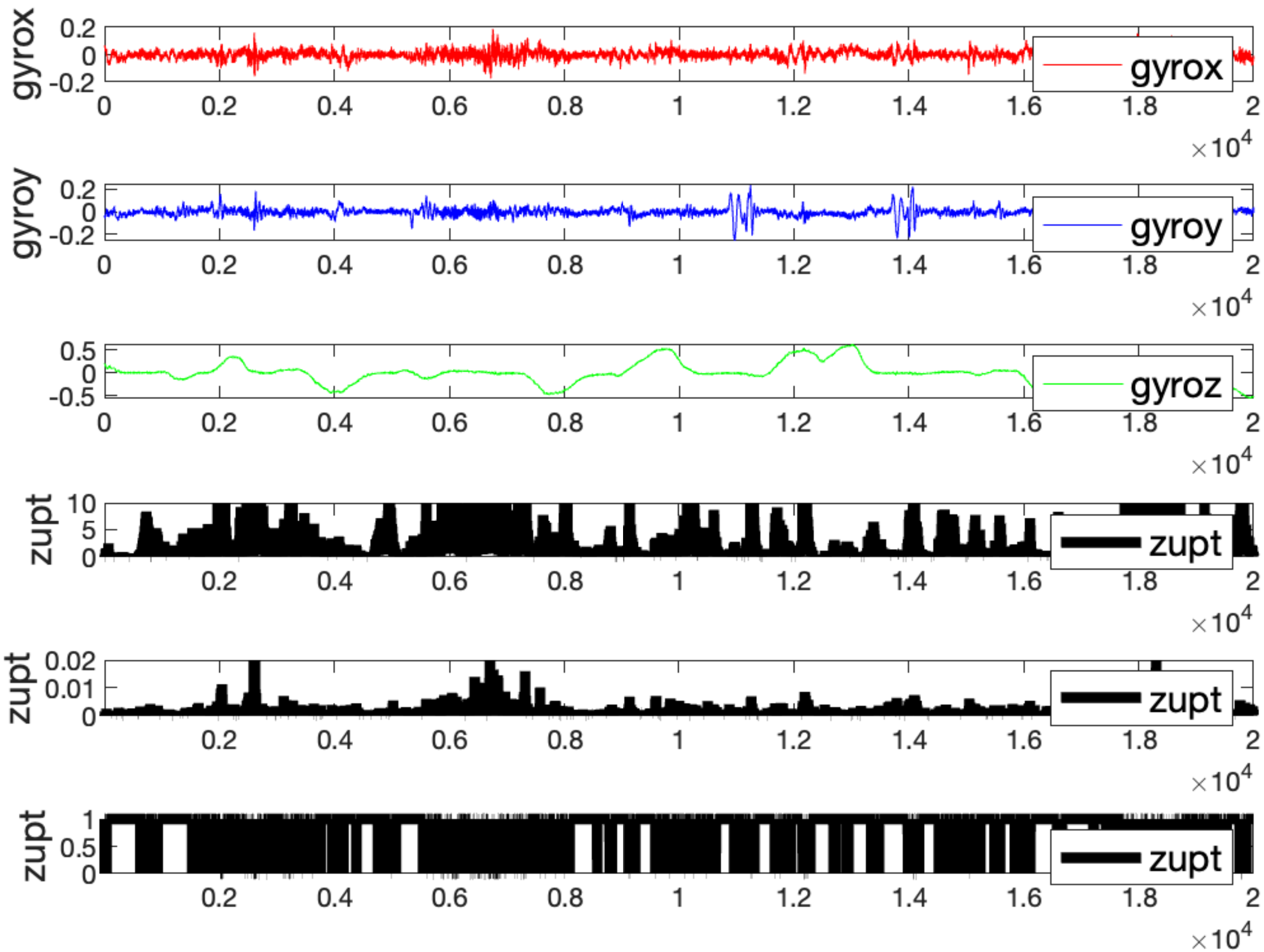
速度与位置追踪的实验结果

Bias设为测量的经验偏差的速度追踪结果



速度与位置追踪的实验结果

- 添加零速检测的结果

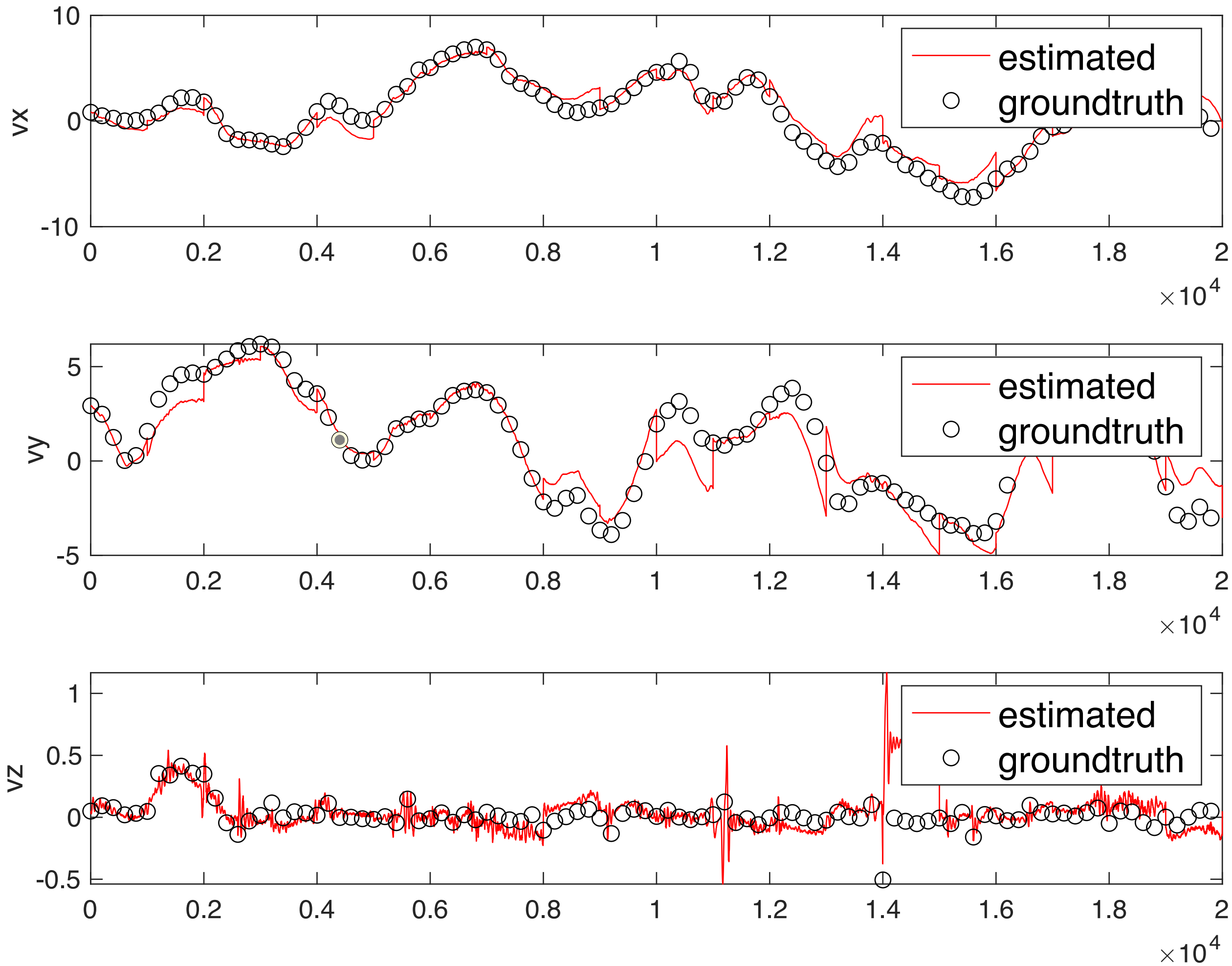


角速度测量值

零速检测的结果

速度与位置追踪的实验结果

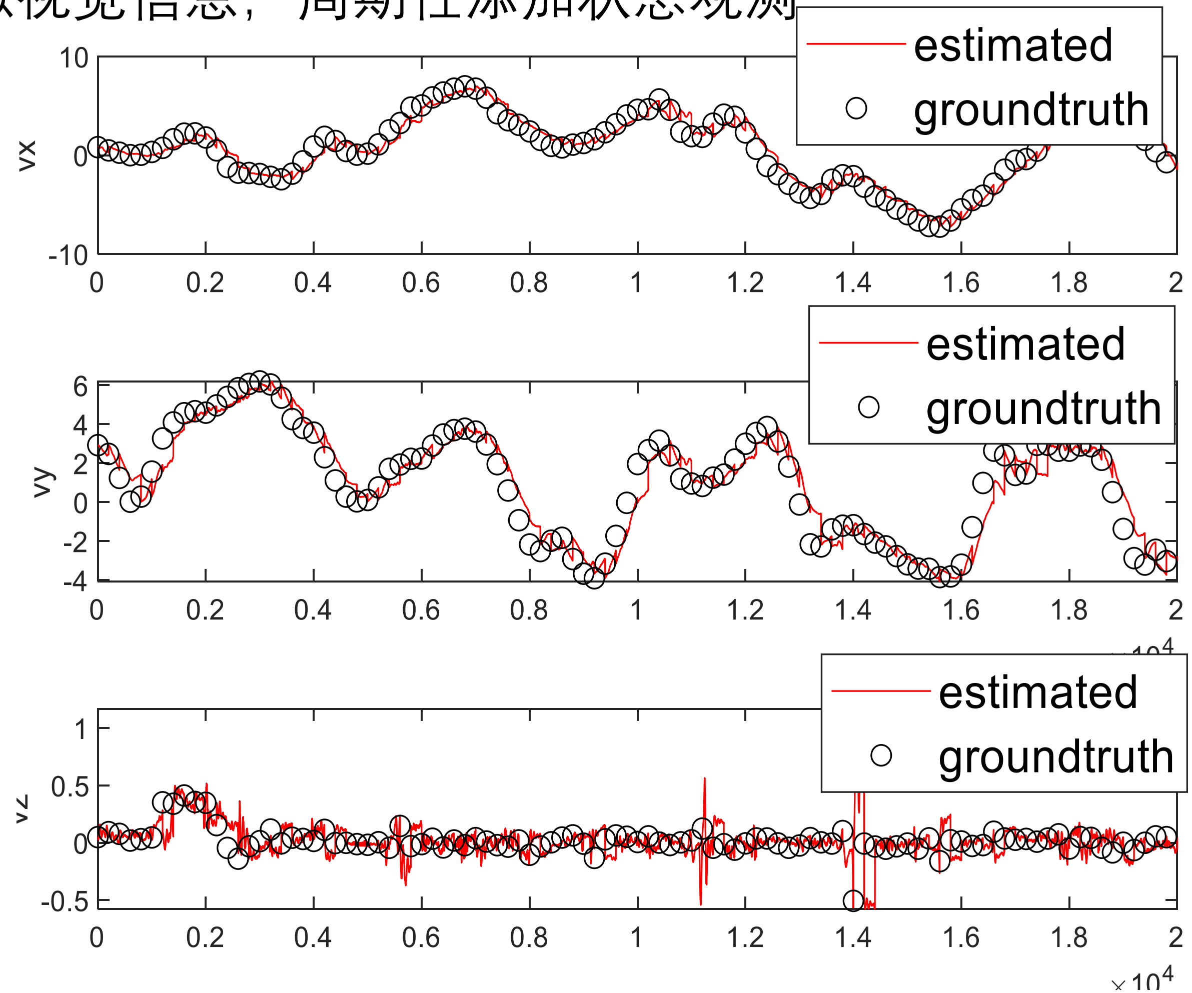
- 添加零速检测的结果



偏差减小了，但是产生了些毛刺

速度与位置追踪的实验结果

- 模拟视觉信息，周期性添加状态观测



惯性导航总结