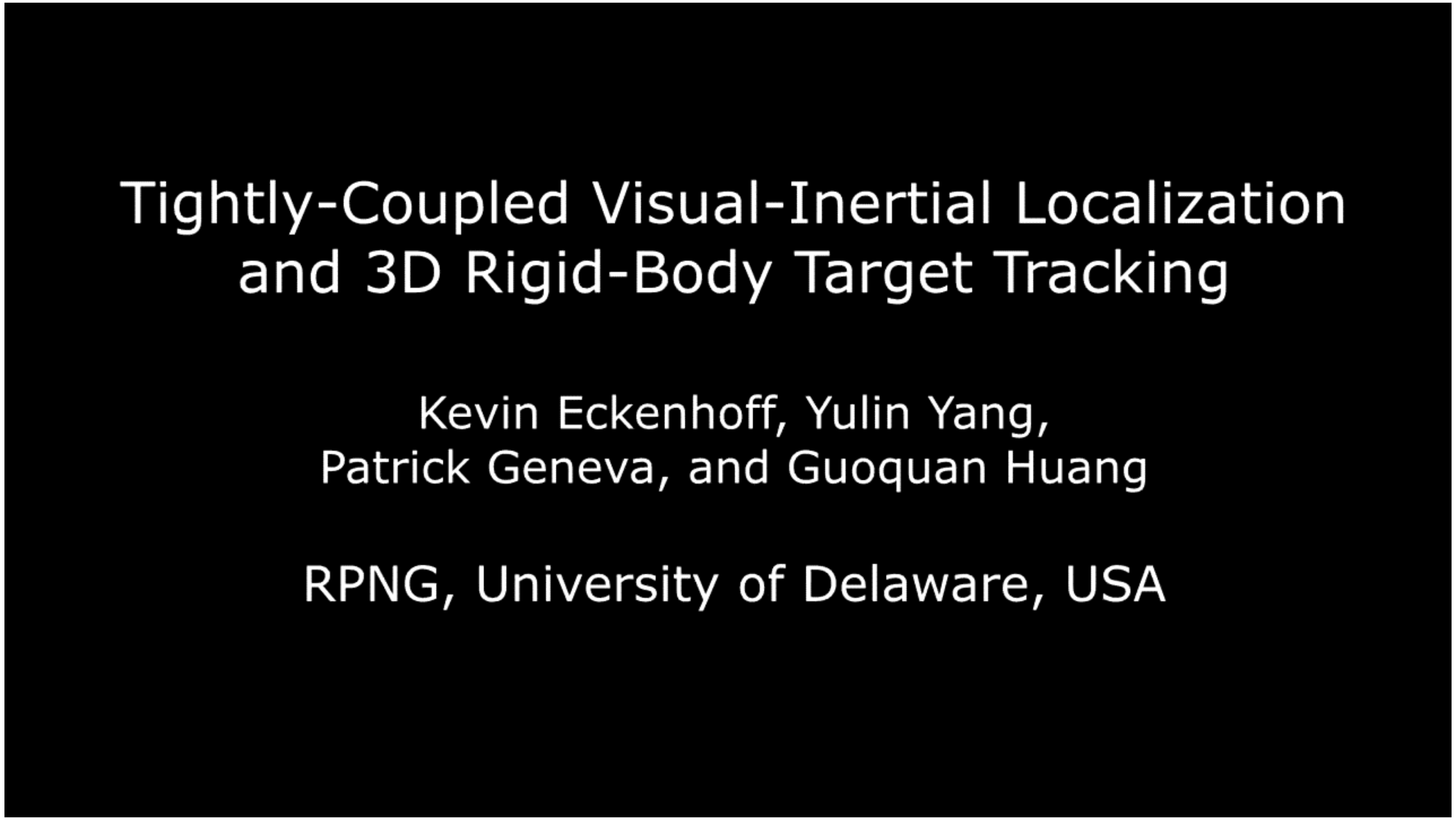
Tightly-Coupled Visual-Inertial Localization and 3-D Rigid-Body Target Tracking

Kevin Eckenhoff, Yulin Yang, Patrick Geneva, and Guoquan Huang

Speaker: Erqun Dong

3-D Tracking 和VIO的联合估计



主要思想

- 将3D Tracking加到MSCKF的状态中，做联合估计
- 如何表示Target的位姿？
- 如何表示Target的移动模型？

MSCKF简单回顾

- MSCKF是一个error-state的EKF

误差状态的EKF模型

- **一个完整的采用误差状态EKF的估计系统**

一套单独的系统状态递推方程（并非滤波器预测方程），用来预测系统状态：

$$\hat{\mathbf{X}}_{k+1|k} = \mathbf{\Phi}_k^{k+1}\left(\hat{\mathbf{X}}_{k|k}\right)$$

误差状态的EKF方程，当有效测量到来时才进行计算：

$$\mathbf{P}_{k+1|k} = \mathbf{f}_k^{k+1}\mathbf{P}_{k|k}\mathbf{f}_k^{k+1T} + \mathbf{Q}_k$$

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

$$\begin{aligned}\delta\hat{\mathbf{X}}_{k+1|k+1} &= \mathbf{K}_{k+1}\mathbf{r}_{k+1} \\ &= \mathbf{K}_{k+1}\left(\mathbf{Z}_{k+1} - \mathbf{H}_{k+1}\left(\hat{\mathbf{X}}_{k+1|k}\right)\right)\end{aligned}$$

$$\mathbf{P}_{k+1|k+1} = \left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{h}_{k+1}\right)\mathbf{P}_{k+1|k}$$

其中有：$\mathbf{r}_{k+1} = \mathbf{Z}_{k+1} - \mathbf{H}_{k+1}\left(\hat{\mathbf{X}}_{k+1|k}\right)$

注意小**f**和小**h**所代表的意义：

$$\delta\mathbf{X}_{k+1} = \mathbf{f}_k^{k+1}\delta\mathbf{X}_k + \mathbf{w}$$

$$\mathbf{r} = \mathbf{h}\delta\mathbf{X} + \mathbf{v}$$

最终的系统输出：$\hat{\mathbf{X}}_{k+1|k+1} = \hat{\mathbf{X}}_{k+1|k} + \delta\hat{\mathbf{X}}_{k+1|k+1}$

MSCKF简单回顾

- 某一时刻的状态

$$\mathbf{x}_I = \begin{bmatrix} {}^I_G\bar{q}^\top & \mathbf{b}_\omega^\top & {}^G\mathbf{v}_I^\top & \mathbf{b}_a^\top & {}^G\mathbf{p}_I^\top \end{bmatrix}^\top$$

- Error-state

$$\delta\mathbf{x}_I = \begin{bmatrix} {}^I\delta\boldsymbol{\theta}_G^\top & \delta\mathbf{b}_\omega^\top & {}^G\delta\mathbf{v}_I^\top & \delta\mathbf{b}_a^\top & {}^G\delta\mathbf{p}_I^\top \end{bmatrix}^\top$$

MSCKF简单回顾

- 测量值

$$\mathbf{a}_m = \mathbf{a} + {}^I_G\mathbf{R}^G\mathbf{g} + \mathbf{b}_a + \mathbf{n}_a, \quad \boldsymbol{\omega}_m = \boldsymbol{\omega} + \mathbf{b}_\omega + \mathbf{n}_\omega$$

- 状态方程。由下面这个方程组就能推出error-state的状态递推方程

$$ {}^I_G\dot{\bar{q}} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}){}^I_G\bar{q}, \quad {}^G\dot{\mathbf{v}} = {}^I_G\mathbf{R}^\top\mathbf{a}, \quad {}^G\dot{\mathbf{p}} = {}^G\mathbf{v}$$

$$\dot{\mathbf{b}}_w = \mathbf{n}_{b\omega}, \quad \dot{\mathbf{b}}_a = \mathbf{n}_{ba}$$

误差状态的EKF模型

- 一个完整的采用误差状态EKF的估计系统

一套单独的系统状态递推方程（并非滤波器预测方程），用来预测系统状态：

$$\hat{\mathbf{X}}_{k+1|k} = \mathbf{\Phi}_k^{k+1}\left(\hat{\mathbf{X}}_{k|k}\right)$$

误差状态的EKF方程，当有效测量到来时才进行计算：

$$\mathbf{P}_{k+1|k} = \mathbf{f}_k^{k+1}\mathbf{P}_{k|k}\mathbf{f}_k^{k+1T} + \mathbf{Q}_k$$

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

$$\delta\hat{\mathbf{X}}_{k+1|k+1} = \mathbf{K}_{k+1}\mathbf{r}_{k+1}$$

$$= \mathbf{K}_{k+1}\left(\mathbf{Z}_{k+1} - \mathbf{H}_{k+1}\left(\hat{\mathbf{X}}_{k+1|k}\right)\right)$$

$$\mathbf{P}_{k+1|k+1} = \left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{h}_{k+1}\right)\mathbf{P}_{k+1|k}$$

其中有： $\mathbf{r}_{k+1} = \mathbf{Z}_{k+1} - \mathbf{H}_{k+1}\left(\hat{\mathbf{X}}_{k+1|k}\right)$

注意小f和小h所代表的意义：

$$\delta\mathbf{X}_{k+1} = \mathbf{f}_k^{k+1}\delta\mathbf{X}_k + \mathbf{w}$$

$$\mathbf{r} = \mathbf{h}\delta\mathbf{X} + \mathbf{v}$$

最终的系统输出： $\hat{\mathbf{X}}_{k+1|k+1} = \hat{\mathbf{X}}_{k+1|k} + \delta\hat{\mathbf{X}}_{k+1|k+1}$

MSCKF简单回顾

- 滑窗内的所有状态

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{x}_I & \mathbf{x}_{cl} \end{bmatrix}^\top$$

$$\mathbf{x}_{cl} = \begin{bmatrix} {}^{I_{k-1}}_{G}\bar{q}^\top & {}^{G}\mathbf{p}^\top_{I_{k-1}} & | \cdots & | & {}^{I_{k-m}}_{G}\bar{q}^\top & {}^{G}\mathbf{p}^\top_{I_{k-m}} \end{bmatrix}^\top$$

- 视觉观测类似VINS，就是滑窗内Track到的特征点，每个点也是用逆深度来参数化。用滑窗里的观测来对逆深度做三角化。

$$\mathbf{z}_i = \mathbf{\Pi}({}^{C_i}\mathbf{p}_{fs}) + \mathbf{n}_{fi}, \quad \mathbf{\Pi}\left([x\, y\, z]^\top\right) = \begin{bmatrix} \frac{x}{z} & \frac{y}{z} \end{bmatrix}^\top$$

$${}^{C_i}\mathbf{p}_{fs} = {}^{C}_{I}\mathbf{R}^{I_i}_{G}\mathbf{R}\left({}^{G}\mathbf{p}_{fs} - {}^{G}\mathbf{p}_{I_i}\right) + {}^{C}\mathbf{p}_{I}$$

$${}^{G}\mathbf{p}_{fs} = {}^{G}_{C_a}\mathbf{R}\left(\frac{1}{\rho}\begin{bmatrix}\alpha \\ \beta \\ 1\end{bmatrix}\right) + {}^{G}\mathbf{p}_{C_a}$$

Feature static

anchor

MSCKF简单回顾

- 上式做一下等价变换（为了保证很远点的数值稳定性，乘以逆深度，里面变成归一化齐次坐标）。系统的观测方程就是

$$\mathbf{z}_i = \mathbf{\Pi}(\rho^{C_i}\mathbf{p}_{fs}) + \mathbf{n}_{fi}$$

- 对上面的方程线性化，就得到观测方程的error-state形式：

$$\delta\mathbf{z} = \mathbf{H}_x\delta\mathbf{x} + \mathbf{H}_f\delta\mathbf{m}_f + \mathbf{n}_f$$

- 但是MSCKF要解决EKF中存储特征点很占空间的问题。Trick：使用H_f的左零空间矩阵Q_2^T左乘上式，把和feature相关的这一项消掉

$$\delta\mathbf{z}' = \mathbf{H}'_x\delta\mathbf{x} + \mathbf{n}'_f$$

- MSCKF中把这个左零空间变换后的方程作为观测的error-state

误差状态的EKF模型

- 一个完整的采用误差状态EKF的估计系统

一套单独的系统状态递推方程（并非滤波器预测方程），用来预测系统状态：

$$\hat{\mathbf{X}}_{k+1|k} = \mathbf{\Phi}_k^{k+1}\left(\hat{\mathbf{X}}_{k|k}\right)$$

误差状态的EKF方程，当有效测量到来时才进行计算：

$$\mathbf{P}_{k+1|k} = \mathbf{f}_k^{k+1}\mathbf{P}_{k|k}\mathbf{f}_k^{k+1T} + \mathbf{Q}_k$$

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

$$\begin{aligned}\delta\hat{\mathbf{X}}_{k+1|k+1} &= \mathbf{K}_{k+1}\mathbf{r}_{k+1} \\ &= \mathbf{K}_{k+1}\left(\mathbf{Z}_{k+1} - \mathbf{H}_{k+1}\left(\hat{\mathbf{X}}_{k+1|k}\right)\right)\end{aligned}$$

$$\mathbf{P}_{k+1|k+1} = \left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{h}_{k+1}\right)\mathbf{P}_{k+1|k}$$

其中有：$\mathbf{r}_{k+1} = \mathbf{Z}_{k+1} - \mathbf{H}_{k+1}\left(\hat{\mathbf{X}}_{k+1|k}\right)$

注意小f和小h所代表的意义：

$$\delta\mathbf{X}_{k+1} = \mathbf{f}_k^{k+1}\delta\mathbf{X}_k + \mathbf{w}$$

$$\mathbf{r} = \mathbf{h}\delta\mathbf{X} + \mathbf{v}$$

最终的系统输出：$\hat{\mathbf{X}}_{k+1|k+1} = \hat{\mathbf{X}}_{k+1|k} + \delta\hat{\mathbf{X}}_{k+1|k+1}$

Target State Estimation

- 在这篇文章之前，Target State往往用单点来描述。但这样回丢失很多几何信息，也会因为遮挡等情况而出问题。
- 这篇文章用一个Representative Point, ${}^{G}\mathbf{p}_{T}$ 和一个orientation ${}^{T}_{G}\bar{q}$ ，来表述Target的状态。
- 此外，还跟踪一些Non-Representative Point。Non-Representative Point像static feature一样，是在整个滑窗内Tracking到的所有点。区别是Non-Representative Point是表述在Target系里的。

Target State Estimation

- Representative Point的观测方程和static feature基本一致，只不过这里 ${}^{G}\mathbf{p}_T$ 是运动的，每个时刻都要优化一个新值：

$$\mathbf{z}_i = \mathbf{\Pi}({}^{C_i}\mathbf{p}_T) + \mathbf{n}_{fi}$$

$${}^{C_i}\mathbf{p}_T = {}^{C}_{I}\mathbf{R}\,{}^{I_i}_{G}\mathbf{R}\left({}^{G}\mathbf{p}_T - {}^{G}\mathbf{p}_{I_i}\right) + {}^{C}\mathbf{p}_I$$

- Non-Representative Point的观测方程。由于每个Non-Representative Point都是在Target的坐标系里描述的，所以要多转换一次坐标系：

$$\mathbf{z}_i = \mathbf{\Pi}({}^{C_i}\mathbf{p}_{ft}) + \mathbf{n}_{fi}$$

$${}^{C_i}\mathbf{p}_{ft} = {}^{C}_{I}\mathbf{R}\,{}^{I_i}_{G}\mathbf{R}\left({}^{G}\mathbf{p}_{T_i} + {}^{T_i}_{G}\mathbf{R}^{\top T}\mathbf{p}_{ft} - {}^{G}\mathbf{p}_{I_i}\right) + {}^{C}\mathbf{p}_I$$

Target State Estimation

- 关于Non-Representative Point的观测就可以和static feature合并在一起。然后也用左零空间消掉。这样EKF中就不用存储Non-Representative Point的信息了。

Target State Estimation

- MSCKF中还要求给出feature的初始估计。Non-Representative Point满足这个方程。其中d是深度，r是feature的归一化齐次坐标：

$$ {}^{T}\mathbf{p}_{ft} = {}^{T_i}_{G}\mathbf{R}\left({}^{G}\mathbf{p}_{C_i} - {}^{G}\mathbf{p}_{T_i}\right) + d_i {}^{T_i}_{G}\mathbf{R}^{G}_{C_i}\mathbf{R}^{C_i}\mathbf{r}_i $$

- 上面的方程左乘 ${}^{T_i}_{G}\mathbf{R}^{G}_{C_i}\mathbf{R}^{C_i}\mathbf{r}_i$ 对应的反对称矩阵，刚好就消掉了这一项：

$$ \left\lfloor {}^{T_i}_{G}\mathbf{R}^{G}_{C_i}\mathbf{R}^{C_i}\mathbf{r}_i \right\rfloor {}^{T}\mathbf{p}_{ft} = \left\lfloor {}^{T_i}_{G}\mathbf{R}^{G}_{C_i}\mathbf{R}^{C_i}\mathbf{r}_i \right\rfloor {}^{T_i}_{G}\mathbf{R}\left({}^{G}\mathbf{p}_{C_i} - {}^{G}\mathbf{p}_{T_i}\right) $$

- 然后就可以解出 ${}^{T}\mathbf{p}_{ft}$ 的初值。之后可以对 ${}^{T}\mathbf{p}_{ft}$ 做一个local BA。

误差状态的EKF模型

- 一个完整的采用误差状态EKF的估计系统

一套单独的系统状态递推方程（并非滤波器预测方程），用来预测系统状态：

$$\hat{\mathbf{X}}_{k+1|k} = \mathbf{\Phi}_k^{k+1}\left(\hat{\mathbf{X}}_{k|k}\right)$$

误差状态的EKF方程，当有效测量到来时才进行计算：

$$\mathbf{P}_{k+1|k} = \mathbf{f}_k^{k+1}\mathbf{P}_{k|k}\mathbf{f}_k^{k+1T} + \mathbf{Q}_k$$

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

$$\begin{aligned}\delta\hat{\mathbf{X}}_{k+1|k+1} &= \mathbf{K}_{k+1}\mathbf{r}_{k+1} \\ &= \mathbf{K}_{k+1}\left(\mathbf{Z}_{k+1} - \mathbf{H}_{k+1}\left(\hat{\mathbf{X}}_{k+1|k}\right)\right)\end{aligned}$$

$$\mathbf{P}_{k+1|k+1} = \left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{h}_{k+1}\right)\mathbf{P}_{k+1|k}$$

其中有：$\mathbf{r}_{k+1} = \mathbf{Z}_{k+1} - \mathbf{H}_{k+1}\left(\hat{\mathbf{X}}_{k+1|k}\right)$

注意小f和小h所代表的意义：

$$\delta\mathbf{X}_{k+1} = \mathbf{f}_k^{k+1}\delta\mathbf{X}_k + \mathbf{w}$$

$$\mathbf{r} = \mathbf{h}\delta\mathbf{X} + \mathbf{v}$$

最终的系统输出：$\hat{\mathbf{X}}_{k+1|k+1} = \hat{\mathbf{X}}_{k+1|k} + \delta\hat{\mathbf{X}}_{k+1|k+1}$

Target Motion Model

- Model 1. Constant Global Linear Velocity. 这个模型假设Target在Global系下的速度和角速度在短时间内是恒定的。这种一般发生在速度和角速度是解耦的情况，比如UAV。
- 状态、状态方程、Error-state的状态方程分别如下：

$$\mathbf{x}_T^{(1)} = \begin{bmatrix} {}^T_G\bar{q}^\top & {}^T\boldsymbol{\omega}^\top & {}^G\mathbf{p}_T^\top & {}^G\mathbf{v}_T^\top \end{bmatrix}^\top$$

$${}^T_G\dot{\bar{q}} = \frac{1}{2}\boldsymbol{\Omega}\left({}^T\boldsymbol{\omega}\right){}^T_G\bar{q},\ {}^G\dot{\mathbf{p}}_T = {}^G\mathbf{v}_T,\ {}^G\dot{\mathbf{v}}_T = \mathbf{n}_{tv},\ {}^T\dot{\boldsymbol{\omega}} = \mathbf{n}_{t\omega}$$

$$\begin{bmatrix} {}^T\delta\dot{\boldsymbol{\theta}}_G \\ {}^T\delta\dot{\boldsymbol{\omega}} \\ {}^G\delta\dot{\mathbf{p}}_T \\ {}^G\delta\dot{\mathbf{v}}_T \end{bmatrix} = \begin{bmatrix} -\lfloor{}^T\hat{\boldsymbol{\omega}}\rfloor & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \begin{bmatrix} {}^T\delta\boldsymbol{\theta}_G \\ {}^T\delta\boldsymbol{\omega} \\ {}^G\delta\mathbf{p}_T \\ {}^G\delta\mathbf{v}_T \end{bmatrix} + \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{I}_3 & \mathbf{0}_3 \end{bmatrix} \begin{bmatrix} \mathbf{n}_{tv} \\ \mathbf{n}_{t\omega} \end{bmatrix}$$

Target Motion Model

- Model 2. Constant Local Linear Velocity. 这个模型假设Target的速度在Target系下是短时间恒定的。但是Target的角速度还是在Global系下是短时间恒定的。比如地面的车辆、固定翼飞行器。
- 状态、状态方程、Error-state的状态方程分别如下：

$$\mathbf{x}_T^{(2)} = \begin{bmatrix} {}^T_G\bar{q}^\top & {}^T\boldsymbol{\omega}^\top & {}^G\mathbf{p}_T^\top & {}^T\mathbf{v}_T^\top \end{bmatrix}^\top$$

$${}^T_G\dot{\bar{q}} = \frac{1}{2}\boldsymbol{\Omega}\left({}^T\boldsymbol{\omega}\right){}^T_G\bar{q},\ {}^G\dot{\mathbf{p}}_T = {}^G_T\mathbf{R}^T\mathbf{v}_T,\ {}^T\dot{\mathbf{v}}_T = \mathbf{n}_{tv},\ {}^T\dot{\boldsymbol{\omega}} = \mathbf{n}_{tw}$$

$$\begin{bmatrix} {}^T\delta\dot{\boldsymbol{\theta}}_G \\ {}^T\delta\dot{\boldsymbol{\omega}} \\ {}^G\delta\dot{\mathbf{p}}_T \\ {}^T\delta\dot{\mathbf{v}}_T \end{bmatrix} = \begin{bmatrix} -\lfloor {}^T\hat{\boldsymbol{\omega}} \rfloor & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -{}^G_T\hat{\mathbf{R}}\lfloor {}^T\hat{\mathbf{v}}_T \rfloor & \mathbf{0}_3 & \mathbf{0}_3 & {}^G_T\hat{\mathbf{R}} \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \begin{bmatrix} {}^T\delta\boldsymbol{\theta}_G \\ {}^T\delta\boldsymbol{\omega} \\ {}^G\delta\mathbf{p}_T \\ {}^T\delta\mathbf{v}_T \end{bmatrix} + \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{I}_3 & \mathbf{0}_3 \end{bmatrix} \begin{bmatrix} \mathbf{n}_{tv} \\ \mathbf{n}_{tw} \end{bmatrix}$$

Target Motion Model

- Model 3. Local Planar Velocity. 这个模型假设速度都发生在一个平面上，这样的话，一方面，速度在Target系里是恒定的；另一方面，角速度只有yaw这个分量。Model 3其实就是Model 2限制角速度只能在yaw上变化。这个模型适用于地面车辆。

$$\mathbf{x}_T^{(3)} = \begin{bmatrix} {}^T_G\bar{q}^\top & \omega_z & {}^G\mathbf{p}_T^\top & v_x & v_y \end{bmatrix}^\top$$

$${}^T_G\dot{\bar{q}} = \frac{1}{2}\boldsymbol{\Omega}\left(\begin{bmatrix} n_{\omega x} \\ n_{\omega y} \\ \omega_z \end{bmatrix}\right){}^T_G\bar{q}, \quad {}^G\dot{\mathbf{p}}_T = {}^G_T\mathbf{R}\begin{bmatrix} v_x \\ v_y \\ n_{vz} \end{bmatrix}$$

$$\dot{v}_x = n_{vx}, \quad \dot{v}_y = n_{vy}, \quad \dot{\omega}_z = n_{\omega z}$$

where

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{K} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{J} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix},$$

and $\mathbf{e}_i$ is the 3×1 unit vector in the i-th axis direction.

$$\begin{bmatrix} {}^T\delta\dot{\boldsymbol{\theta}}_G \\ \delta\dot{\omega}_z \\ {}^G\delta\dot{\mathbf{p}}_T \\ \delta\dot{v}_x \\ \delta\dot{v}_y \end{bmatrix} = \begin{bmatrix} -\lfloor {}^T\hat{\boldsymbol{\omega}} \rfloor & \mathbf{e}_3 & \mathbf{0}_3 & \mathbf{0}_{3\times 2} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} & \mathbf{0}_3 & \mathbf{0}_{3\times 2} \\ -{}^G_T\hat{\mathbf{R}}\lfloor {}^T\hat{\mathbf{v}}_T \rfloor & \mathbf{0}_{3\times 2} & \mathbf{0}_3 & {}^G_T\hat{\mathbf{R}}\begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 \end{bmatrix} \\ \mathbf{0}_{2\times 3} & \mathbf{0}_{2\times 1} & \mathbf{0}_{2\times 3} & \mathbf{0}_{2\times 2} \end{bmatrix} \begin{bmatrix} {}^T\delta\boldsymbol{\theta}_G \\ \delta\omega_z \\ {}^G\delta\mathbf{p}_T \\ \delta v_x \\ \delta v_y \end{bmatrix} + \begin{bmatrix} \mathbf{0}_3 & \mathbf{L} \\ \mathbf{0}_{1\times 3} & \mathbf{e}_3^\top \\ {}^G_T\hat{\mathbf{R}}\mathbf{K} & \mathbf{0}_3 \\ \mathbf{J} & \mathbf{0}_{2\times 3} \end{bmatrix} \begin{bmatrix} n_{vx} \\ n_{vy} \\ n_{vz} \\ n_{\omega x} \\ n_{\omega y} \\ n_{\omega z} \end{bmatrix}$$

能观性分析

- 这块我还是零基础，没有看懂。

附录——Target Motion Model的error-state推导

- Model 1.

推公式时，不要急着把一个整体展开.

true: $\frac{1}{2}\omega_t \otimes q_t = \dot{q}_t$ error-state: $q_t = \delta q \otimes q = \begin{bmatrix} 1 \\ \frac{1}{2}\delta\theta \end{bmatrix} \otimes q$

nominal: $\frac{1}{2}\omega \otimes q = \dot{q}$ $\omega_t = \omega + \delta\omega$

则有: $\frac{1}{2}\omega_t \otimes q_t = \dot{q}_t = \delta\dot{q} \otimes q + \delta q \otimes \dot{q}$

$$\frac{1}{2}\omega_t \otimes \delta q \otimes q = \delta\dot{q} \otimes q + \delta q \otimes \frac{1}{2}\omega \otimes q$$

同时消去 q $\Rightarrow$ 并且两侧同乘2.

$$2\delta\dot{q} = \omega_t \otimes \delta q - \delta q \otimes \omega$$

$$= \left([\omega_t]_R - [\omega]_L\right)\delta q$$

$$= \begin{bmatrix} 0 & -(\omega_t-\omega)^T \\ (\omega_t-\omega) & -[\omega_t+\omega]_\times \end{bmatrix} \delta q$$

所以, $2\begin{bmatrix} 0 \\ \frac{1}{2}\delta\dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & -(\omega_t-\omega)^T \\ (\omega_t-\omega) & -[\omega_t+\omega]_\times \end{bmatrix} \begin{bmatrix} 1 \\ \frac{1}{2}\delta\theta \end{bmatrix}$

$$\Rightarrow \delta\dot{\theta} = -\frac{1}{2}[\omega_t+\omega]_\times \delta\theta + (\omega_t-\omega)$$

$$= -[\omega]_\times \delta\theta + \delta\omega + O(\|\delta\theta\|^2)$$

Model 2.

true: $\dot{P}_t = R_t V_t$ error-state: $P_t = P + \delta P$

nominal: $\dot{P} = RV$

用李代数推.

$$R_t V_t = \dot{P}_t = (\dot{P+\delta P}) = \dot{P} + \delta\dot{P} = RV + \delta\dot{P}$$

$$\Rightarrow \delta\dot{P} = \underline{R_t} V_t - RV = R \cdot \mathrm{Exp}([\delta\theta]_\times) V_t - R \cdot V$$

因为 R_t表示 R_{GT}，所以应该用右乘扰动 $\cancel{= \mathrm{Exp}([\delta\theta]_\times) R V_t - RV}$

$$= R(1 + [\delta\theta]_\times + \frac{1}{2}[\delta\theta]_\times^2 + \cdots)(V + \delta V) - RV$$

$$= R[\delta\theta]_\times V + R\delta V + O(|\delta\theta|^2)$$

$\cancel{= (1 + [\delta\theta]_\times + \frac{1}{2}[\delta\theta]_\times^2 + \cdots) R (V+\delta V) - RV}$ $= -R[V]_\times \delta\theta + R\delta V.$ ✓

$\cancel{= [\delta\theta]_\times RV + R\delta V + O(|\delta\theta|^2)}$.

$\cancel{= -[RV]_\times \delta\theta + R\delta V}$ 错

$$= -R[V]_\times \delta\theta + R\delta V$$

用四元数推不出来…

$$\delta\dot{P} = \delta q \otimes q \otimes (V + \delta V) - q \otimes V$$

$$= \delta q \otimes q \otimes V - q \otimes V + \delta q \otimes q \otimes \delta V$$

$$= \begin{bmatrix} 1 \\ \frac{1}{2}\delta\theta \end{bmatrix} \otimes q \otimes V - \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes q \otimes V + \delta q \otimes q \otimes \delta V$$

$$= \underbrace{\begin{bmatrix} 0 \\ \frac{1}{2}\delta\theta \end{bmatrix}}_{\text{pure}} \otimes q \otimes \underbrace{V}_{\text{pure}} + \delta q \otimes q \otimes \delta V$$

Model 3.

Model 3. 就是 Model 2的变种.

把 Model2的 ${}^T\omega$ 特殊化为 $\begin{bmatrix} n_{\omega x} \\ n_{\omega y} \\ \omega_{zt} \end{bmatrix}$

只需在error-state中，注意一下

$$\omega_t = \omega + \delta\omega$$

其中，nominal 值 $\omega = \begin{bmatrix} 0 \\ 0 \\ \omega_z \end{bmatrix}$ 这两维就是高斯白噪声

$$\delta\omega = \begin{bmatrix} n_{\omega x} \\ n_{\omega y} \\ \delta\omega_z \end{bmatrix}$$