这是标题

陈烁龙

2022 年 5 月 6 日

目录

插图

表格

摘要

SLAM 中估计的手段一般有两种：基于滤波的方式和基于非线性优化的方式。本文基于卡尔曼滤波的方式，给出了对一个位姿轨迹的估计滤波过程。

关键词：卡尔曼滤波，PCL，模拟

1 数值模拟

一般，我们 SLAM 后端优化都使用的是非线性优化，我们认为其优化的结果由于传统滤波的方式。所以本次我们使用的卡尔曼滤波主要是用于位姿预测，以加快特征的匹配[1]。

为了实验的进行，我们首先要模拟出一组数据。我们知道，即使对位姿图进行了 BA 优化，但是仍然会出现累积误差[2]。基于此经验，我们会首先设计一条无误差的标准位姿轨迹。每一帧我们计算标准轨迹位姿变化量：

$$\xi_i^{update} \leftarrow \begin{cases} R_i^{update} = R_Z(\frac{\pi}{180})R_Y(\frac{\pi}{180})R_X(\frac{\pi}{180}) \\ t_i^{update} = \begin{pmatrix} 0.0 & 0.0 & 1.0 \end{pmatrix}^T \end{cases}$$

对于标准轨迹，我们直接对其进行更新：

$$\xi_i^{truth} = \xi_i^{update}\xi_{i-1}^{truth}$$

而对于观测轨迹，我们会在标准轨迹位姿变化量的基础上增加误差，得到附有误差的位姿变化量，而后在对观测值进行更新：

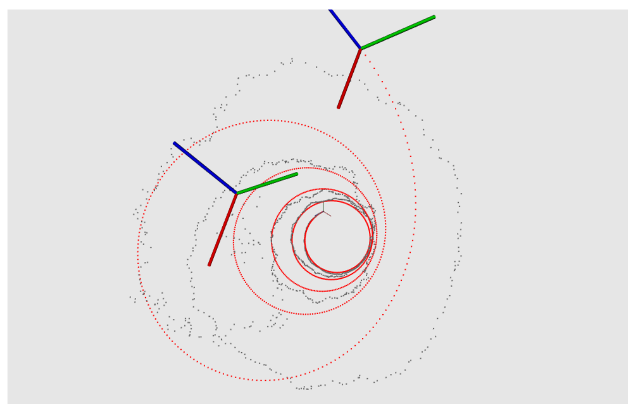
$$\xi_i^{error} \leftarrow \begin{cases} R_i^{error} = R_Z(N_R)R_Y(N_R)R_X(N_R) \\ t_i^{error} = \begin{pmatrix} N_t & N_t & N_t \end{pmatrix}^T \end{cases}$$

$$\xi_i^{obs} = \xi_i^{error}\xi_i^{update}\xi_{i-1}^{obs}$$

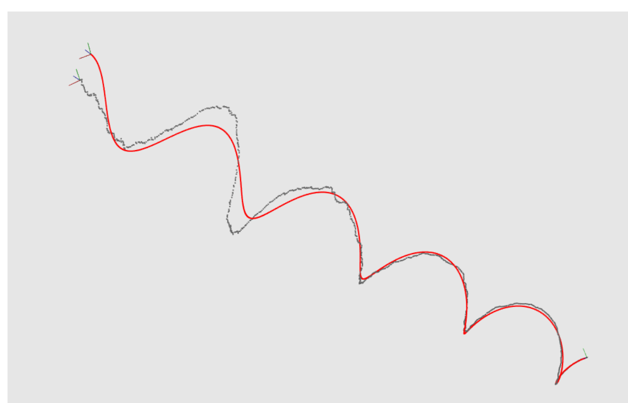
其中 N_R、N_t 为服从在正太分布的 0 均值随机变量，我们事先假定了其标准差。当然，也可在代码中显式指定其值。

$$\begin{cases} N_R \sim N(0, \sigma_R = \frac{\pi}{1800}) \\ N_t \sim N(0, \sigma_t = 0.01) \end{cases}$$

图 1 为本次我们模拟出来的位姿轨迹图。其中红色的位姿轨迹为实（理想）的位姿轨迹，而黑色的则为我们引入累积误差的轨迹，为我们所“观测到的”位姿轨迹。可以看到，随着附带误差的新估计位姿变化量的到来，我们所观测的位姿轨迹逐渐偏离真实的位姿轨迹。这正是我们所需要的。



(a) 效果图 1



(b) 效果图 2

图 1: 模拟效果

[1]当预测了下一帧的位姿后，我们可以推算出当前地图点在下一帧影响上可能出现的位置。

[2]这也是为什么有回环检测的原因。

2 Kalman 滤波公式

2.1 运动方程

首先我们得取定我们所研究系统的运动方程。不是一般性，我们假设我们到估计的位姿是均匀变化的，同时考虑模型的系统误差，我们有：

$$\ddot{\xi}(t) = \begin{pmatrix} \ddot{\xi}_R(t) \\ \ddot{\xi}_t(t) \end{pmatrix} = \begin{pmatrix} e_R(t) \\ e_t(t) \end{pmatrix}$$

其中 $\ddot{\xi}(t)$ 为 SE3 对应的李代数对时间的二次导数，是一个有六个元素的列向量[3]。$e(t)$ 为白噪声，满足：

$$\begin{cases} E(e(t)) = 0 \\ Cov(e(t), e(\tau)) = D_e(t)\delta(t-\tau) \\ D_e(t) = \begin{pmatrix} D_{e_R}(t) & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & D_{e_t}(t) \end{pmatrix} \end{cases}$$

其中 $\delta(t-\tau)$ 为狄拉克-δ 函数。我们设我们的状态参量为：

$$X(t) = \begin{pmatrix} X_1(t) & X_2(t) \end{pmatrix}^T = \begin{pmatrix} \xi(t) & \dot{\xi}(t) \end{pmatrix}^T$$

则有：

$$\begin{pmatrix} \dot{\xi}(t) \\ \ddot{\xi}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{0_{6\times6}} & \mathbf{1_{6\times6}} \\ \mathbf{0_{6\times6}} & \mathbf{0_{6\times6}} \end{pmatrix} \begin{pmatrix} \xi(t) \\ \dot{\xi}(t) \end{pmatrix} + \begin{pmatrix} \mathbf{0_{6\times6}} \\ \mathbf{1_{6\times6}} \end{pmatrix} e(t)$$

我们将其记为：

$$\dot{X}(t) = A(t)X(t) + C(t)e(t)$$

通过解这个微分方程，并将其离散化，我们可以得到：

$$X(t_k) = \Phi(t_k, t_{k-1})X(t_{k-1}) + w(t_{k-1})$$

其中 $\Phi(t_k, \tau)$ 为状态转移矩阵：

$$\Phi(t_k, \tau) = \begin{pmatrix} \mathbf{1_{6\times6}} & (t_k-\tau)\mathbf{1_{6\times6}} \\ \mathbf{0_{6\times6}} & \mathbf{1_{6\times6}} \end{pmatrix}$$

而 $w(t_{k-1})$ 为噪声项 (系统噪声)：

$$w(t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau)C(\tau)D_e(\tau)d\tau$$

其方差为：

$$D_w(t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau)C(\tau)D_e(\tau)C(\tau)^T\Phi(t_k, \tau)^T d\tau$$

易知：

$$\Phi(t_k, \tau)C(\tau) = \begin{pmatrix} (t_k-\tau)\mathbf{1_{3\times3}} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & (t_k-\tau)\mathbf{1_{3\times3}} \\ \mathbf{1_{3\times3}} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & \mathbf{1_{3\times3}} \end{pmatrix}$$

所以被积分的矩阵为：

$$\begin{pmatrix} (t_k-\tau)^2 D_{e_R} & \mathbf{0_{3\times3}} & (t_k-\tau)D_{e_R} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & (t_k-\tau)^2 D_{e_t} & \mathbf{0_{3\times3}} & (t_k-\tau)D_{e_t} \\ (t_k-\tau)D_{e_R} & \mathbf{0_{3\times3}} & D_{e_R} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & (t_k-\tau)D_{e_t} & \mathbf{0_{3\times3}} & D_{e_t} \end{pmatrix}$$

积分求解可得方差矩阵为：

$$\begin{pmatrix} \frac{(\Delta t)^3 D_{e_R}}{3} & \mathbf{0_{3\times3}} & \frac{(\Delta t)^2 D_{e_R}}{2} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & \frac{(\Delta t)^3 D_{e_t}}{3} & \mathbf{0_{3\times3}} & \frac{(\Delta t)^2 D_{e_t}}{2} \\ \frac{(\Delta t)^2 D_{e_R}}{2} & \mathbf{0_{3\times3}} & \Delta t D_{e_R} & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & \frac{(\Delta t)^2 D_{e_t}}{2} & \mathbf{0_{3\times3}} & \Delta t D_{e_t} \end{pmatrix}$$

其中 $\Delta t = t_k - t_{k-1}$。

2.2 观测方程

相较于运动方程。我们的观测方程就变得比较简单了，因为我们所观测的参量中的一部分正是我们所估计的参数。所以，我们有：

$$Z(t_k) = \begin{pmatrix} \mathbf{0_{6\times6}} & \mathbf{1_{6\times6}} \end{pmatrix} \begin{pmatrix} X_1(t_k) \\ X_2(t_k) \end{pmatrix} + \Delta(t_k)$$

我们将其记为：

$$Z(t_k) = H(t_k)X(t_k) + \Delta(t_k)$$

其中，$\Delta(t)$ 为白噪声，满足：

$$Cov(\Delta(t), \Delta(\tau)) = D_\Delta(t)\delta(t-\tau)$$

[3]其中三维与旋转有关，三维与平移有关。

2.3 Kalman 滤波

基于上文，我们有如下的公式：

2.3.1 一步预测

$$
\begin{cases}
\hat{X}(t_k, t_{k-1}) = \Phi(t_k, t_{k-1})\hat{X}(t_{k-1}) \\
D_{\hat{X}}(t_k, t_{k-1}) = \Phi(t_k, t_{k-1})D_{\hat{X}}(t_{k-1})\Phi(t_k, t_{k-1})^T \\
\qquad\qquad\qquad + D_w(t_{k-1})
\end{cases}
$$

我们基于上一时刻 t_{k-1} 的状态量 $\hat{X}(t_{k-1})$，通过 t_{k-1} 到 t_k 的状态转移矩阵 $\Phi(t_k, t_{k-1})$，预测当前时刻 t_k 的状态量 $\hat{X}(t_k, t_{k-1})$，并对给出当前预测状态的方差矩阵 $D_{\hat{X}}(t_k, t_{k-1})$。

2.3.2 二步更新

$$
\begin{cases}
K(t_k) = D_{\hat{X}}(t_k, t_{k-1})H(t_k)^T \\
\qquad\qquad (H(t_k)D_{\hat{X}}(t_k, t_{k-1})H(t_k)^T + D_\Delta(t_k))^{-1} \\
\hat{X}(t_k) = \hat{X}(t_k, t_{k-1}) \\
\qquad\quad + K(t_k)(Z(t_k) - H(t_k)\hat{X}(t_{k-1})) \\
D_{\hat{X}}(t_k) = (I - H(t_k)K(t_k))D_{\hat{X}}(t_k, t_{k-1})
\end{cases}
$$

我们基于当前预测状态的方差矩阵 $D_{\hat{X}}(t_k, t_{k-1})$ 和观测矩阵 $H(t_k)$、观测方差矩阵 $D_\Delta(t_k)$，计算卡尔曼增益 $K(t_k)$，而后更新 t_k 时刻的状态量 $\hat{X}(t_k)$ 和方差矩阵 $D_{\hat{X}}(t_k)$。