sad-L2-hw

Yixiao Feng

23/05/2023

1 证明题

因为该高斯随机变量为零均值，协方差为对角线矩阵且大小相同，设该变量为 $\boldsymbol{x}$，均值为 $\boldsymbol{\eta}$，协方差为 $\boldsymbol{Q}$。该变量满足以下形式：

$$\begin{aligned} \boldsymbol{x} &\sim N(\boldsymbol{\eta}, \boldsymbol{Q}), \\ \boldsymbol{\eta} &= \mathbf{0}, \\ \boldsymbol{Q} &= \sigma^2 \boldsymbol{I}. \end{aligned} \tag{1.1}$$

当该变量乘上任意一个旋转矩阵 $\boldsymbol{R}$ 时，$\boldsymbol{R}\boldsymbol{x}$ 的均值可以表达为：

$$\mathbb{E}[\boldsymbol{R}\boldsymbol{x}] = \boldsymbol{R}\mathbb{E}[\boldsymbol{x}] = \boldsymbol{R}\boldsymbol{\eta} = \boldsymbol{R}\mathbf{0} = \mathbf{0}. \tag{1.2}$$

$\boldsymbol{R}\boldsymbol{x}$ 的协方差可以表达为：

$$\begin{aligned} Cov[\boldsymbol{R}\boldsymbol{x}] &= \mathbb{E}[(\boldsymbol{R}\boldsymbol{x} - \mathbb{E}[\boldsymbol{R}\boldsymbol{x}])(\boldsymbol{R}\boldsymbol{x} - \mathbb{E}[\boldsymbol{R}\boldsymbol{x}])^\top], \\ &= \mathbb{E}[(\boldsymbol{R}\boldsymbol{x} - \mathbf{0})(\boldsymbol{R}\boldsymbol{x} - \mathbf{0})^\top], \\ &= \mathbb{E}[\boldsymbol{R}\boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{R}^\top], \\ &= \boldsymbol{R}\boldsymbol{Q}\boldsymbol{R}^\top, \\ &= \boldsymbol{R}\sigma^2 \boldsymbol{I}\boldsymbol{R}^\top, \\ &= \sigma^2 \boldsymbol{I}. \end{aligned} \tag{1.3}$$

至此，可证。

2 每种状态变量的运动方程和代码实现

将 $\boldsymbol{F}$ 矩阵拆开，由于 $\boldsymbol{\eta}_{ba}$ 和 $\boldsymbol{\eta}_{bg}$ 的均值为 $\mathbf{0}$，所以运动方程中将其省去。每个状态变量的运动方程如下：

$$\delta \boldsymbol{p}(t + \Delta t) = \delta \boldsymbol{p} + \delta \boldsymbol{v} \Delta t, \tag{2.1a}$$

$$\delta \boldsymbol{v}(t + \Delta t) = \delta \boldsymbol{v} + (-\boldsymbol{R}(\tilde{\boldsymbol{a}} - \boldsymbol{b}_a)^\wedge \delta \boldsymbol{\theta} - \boldsymbol{R}\delta \boldsymbol{b}_a + \delta \boldsymbol{g})\Delta t, \tag{2.1b}$$

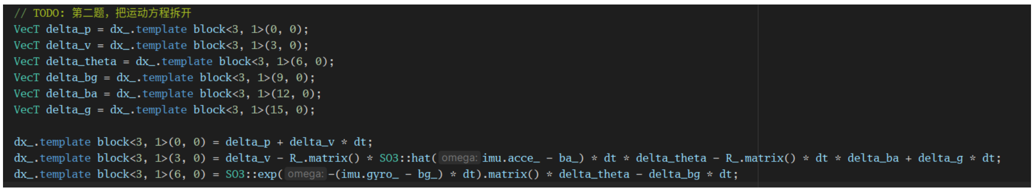
$$\delta \boldsymbol{\theta}(t + \Delta t) = \mathrm{Exp}(-(\tilde{\boldsymbol{\omega}} - \boldsymbol{b}_g)\Delta t)\delta \boldsymbol{\theta} - \delta \boldsymbol{b}_g \Delta t, \tag{2.1c}$$

$$\delta \boldsymbol{b}_g(t + \Delta t) = \delta \boldsymbol{b}_g, \tag{2.1d}$$

$$\delta \boldsymbol{b}_a(t + \Delta t) = \delta \boldsymbol{b}_a, \tag{2.1e}$$

$$\delta \boldsymbol{g}(t + \Delta t) = \delta \boldsymbol{g}. \tag{2.1f}$$

代码实现如图 1：



```
// TODO: 第二题，把运动方程拆开
VecT delta_p = dx_.template block<3, 1>(0, 0);
VecT delta_v = dx_.template block<3, 1>(3, 0);
VecT delta_theta = dx_.template block<3, 1>(6, 0);
VecT delta_bg = dx_.template block<3, 1>(9, 0);
VecT delta_ba = dx_.template block<3, 1>(12, 0);
VecT delta_g = dx_.template block<3, 1>(15, 0);

dx_.template block<3, 1>(0, 0) = delta_p + delta_v * dt;
dx_.template block<3, 1>(3, 0) = delta_v - R_.matrix() * SO3::hat( omega: imu.acce_ - ba_) * dt * delta_theta - R_.matrix() * dt * delta_ba + delta_g * dt;
dx_.template block<3, 1>(6, 0) = SO3::exp( omega: -(imu.gyro_ - bg_) * dt).matrix() * delta_theta - delta_bg * dt;
```

图 1: 运动方程的代码实现

3 左乘模型下的 ESKF 运动方程、噪声方程

定义左乘扰动的形式为：

$$\boldsymbol{R}_t = \mathrm{Exp}(\delta \boldsymbol{\theta})\boldsymbol{R}. \tag{3.1}$$

左乘扰动模型下，名义状态变量的离散时间运动学方程可以写为：

$$\boldsymbol{p}(t+\Delta t)=\boldsymbol{p}(t)+\boldsymbol{v}\Delta t+\frac{1}{2}(\boldsymbol{R}(\tilde{\boldsymbol{a}}-\boldsymbol{b}_a))\Delta t^2+\frac{1}{2}\boldsymbol{g}\Delta t^2, \tag{3.2a}$$

$$\boldsymbol{v}(t+\Delta t)=\boldsymbol{v}(t)+\boldsymbol{R}(\tilde{\boldsymbol{a}}-\boldsymbol{b}_a)\Delta t+\boldsymbol{g}\Delta t, \tag{3.2b}$$

$$\boldsymbol{R}(t+\Delta t)=\operatorname{Exp}((\tilde{\boldsymbol{\omega}}-\boldsymbol{b}_g)\Delta t)\boldsymbol{R}(t), \tag{3.2c}$$

$$\boldsymbol{b}_g(t+\Delta t)=\boldsymbol{b}_g(t), \tag{3.2d}$$

$$\boldsymbol{b}_a(t+\Delta t)=\boldsymbol{b}_a(t), \tag{3.2e}$$

$$\boldsymbol{g}(t+\Delta t)=\boldsymbol{g}(t). \tag{3.2f}$$

3.a 误差状态的旋转项

首先计算左乘模型下误差状态的旋转项，对式(3.1)两侧求时间导数，可得：

$$\begin{aligned}\dot{\boldsymbol{R}}_t&=\dot{\operatorname{Exp}}(\delta\boldsymbol{\theta})\boldsymbol{R}+\operatorname{Exp}(\delta\boldsymbol{\theta})\dot{\boldsymbol{R}},\\&=\boldsymbol{R}_t(\tilde{\boldsymbol{\omega}}-\boldsymbol{b}_{gt}-\boldsymbol{\eta}_g)^\wedge.\end{aligned} \tag{3.3}$$

式(3.3)的第一个式子可以进一步写为：

$$\dot{\operatorname{Exp}}(\delta\boldsymbol{\theta})\boldsymbol{R}+\operatorname{Exp}(\delta\boldsymbol{\theta})\dot{\boldsymbol{R}}=\operatorname{Exp}(\delta\boldsymbol{\theta})\delta\dot{\boldsymbol{\theta}}^\wedge\boldsymbol{R}+\operatorname{Exp}(\delta\boldsymbol{\theta})\boldsymbol{R}(\tilde{\boldsymbol{\omega}}-\boldsymbol{b}_g)^\wedge. \tag{3.4}$$

式(3.3)的第二个式子可以进一步写为：

$$\boldsymbol{R}_t(\tilde{\boldsymbol{\omega}}-\boldsymbol{b}_{gt}-\boldsymbol{\eta}_g)^\wedge=\operatorname{Exp}(\delta\boldsymbol{\theta})\boldsymbol{R}(\tilde{\boldsymbol{\omega}}-\boldsymbol{b}_{gt}-\boldsymbol{\eta}_g)^\wedge. \tag{3.5}$$

将两式合并，约掉两侧左边的 $\operatorname{Exp}(\delta\boldsymbol{\theta})$，并将 $\delta\dot{\boldsymbol{\theta}}^\wedge\boldsymbol{R}$ 移到左侧，可得：

$$\delta\dot{\boldsymbol{\theta}}^\wedge\boldsymbol{R}=\boldsymbol{R}(\tilde{\boldsymbol{\omega}}-\boldsymbol{b}_{gt}-\boldsymbol{\eta}_g)^\wedge-\boldsymbol{R}(\tilde{\boldsymbol{\omega}}-\boldsymbol{b}_g)^\wedge. \tag{3.6}$$

整理一下，可得：

$$\begin{aligned}\delta\dot{\boldsymbol{\theta}}^\wedge\boldsymbol{R}&=\boldsymbol{R}(-\boldsymbol{b}_{gt}+\boldsymbol{b}_g-\boldsymbol{\eta}_g)^\wedge,\\&=\boldsymbol{R}(-\delta\boldsymbol{b}_g-\boldsymbol{\eta}_g)^\wedge.\end{aligned} \tag{3.7}$$

两侧同时右乘 $\boldsymbol{R}^\top$，化简，并利用伴随性质，可得：

$$\delta\dot{\boldsymbol{\theta}}^\wedge=(\boldsymbol{R}(-\delta\boldsymbol{b}_g-\boldsymbol{\eta}_g))^\wedge, \tag{3.8a}$$

$$\delta\dot{\boldsymbol{\theta}}=\boldsymbol{R}(-\delta\boldsymbol{b}_g-\boldsymbol{\eta}_g). \tag{3.8b}$$

3.b 误差状态的速度项

下面推导左乘模型下的误差状态的速度项，对速度误差项的公式两侧求导，等式左侧为：

$$\begin{aligned}\dot{\boldsymbol{v}}_t&=\boldsymbol{R}_t(\tilde{\boldsymbol{a}}-\boldsymbol{b}_{at}-\boldsymbol{\eta}_a)+\boldsymbol{g}_t,\\&=\operatorname{Exp}(\delta\boldsymbol{\theta})\boldsymbol{R}(\tilde{\boldsymbol{a}}-\boldsymbol{b}_a-\delta\boldsymbol{b}_a-\boldsymbol{\eta}_a)+\boldsymbol{g}+\delta\boldsymbol{g},\\&\approx(\boldsymbol{I}+\delta\boldsymbol{\theta}^\wedge)\boldsymbol{R}(\tilde{\boldsymbol{a}}-\boldsymbol{b}_a-\delta\boldsymbol{b}_a-\boldsymbol{\eta}_a)+\boldsymbol{g}+\delta\boldsymbol{g},\\&\approx\boldsymbol{R}\tilde{\boldsymbol{a}}-\boldsymbol{R}\boldsymbol{b}_a-\boldsymbol{R}\delta\boldsymbol{b}_a-\boldsymbol{R}\boldsymbol{\eta}_a+\delta\boldsymbol{\theta}^\wedge\boldsymbol{R}\tilde{\boldsymbol{a}}-\delta\boldsymbol{\theta}^\wedge\boldsymbol{R}\boldsymbol{b}_a+\boldsymbol{g}+\delta\boldsymbol{g},\\&=\boldsymbol{R}\tilde{\boldsymbol{a}}-\boldsymbol{R}\boldsymbol{b}_a-\boldsymbol{R}\delta\boldsymbol{b}_a-\boldsymbol{R}\boldsymbol{\eta}_a-(\boldsymbol{R}\tilde{\boldsymbol{a}})^\wedge\delta\boldsymbol{\theta}+(\boldsymbol{R}\boldsymbol{b}_a)^\wedge\delta\boldsymbol{\theta}+\boldsymbol{g}+\delta\boldsymbol{g}.\end{aligned} \tag{3.9}$$

等式右侧为：

$$\dot{\boldsymbol{v}}+\delta\dot{\boldsymbol{v}}=\boldsymbol{R}(\tilde{\boldsymbol{a}}-\boldsymbol{b}_a)+\boldsymbol{g}+\delta\dot{\boldsymbol{v}}. \tag{3.10}$$

由此，可得：

$$\delta\dot{\boldsymbol{v}}=(\boldsymbol{R}(\boldsymbol{b}_a-\tilde{\boldsymbol{a}}))^\wedge\delta\boldsymbol{\theta}-\boldsymbol{R}\delta\boldsymbol{b}_a-\boldsymbol{R}\boldsymbol{\eta}_a+\delta\boldsymbol{g}. \tag{3.11}$$

3.c 左乘模型下，误差状态变量的离散时间运动学方程

至此，我们可以写出左乘模型下的离散时间的误差状态的运动学方程，如下：

$$
\begin{aligned}
\delta \boldsymbol{p}(t+\Delta t) &= \delta \boldsymbol{p} + \delta \boldsymbol{v}\Delta t, && (3.12a)\\
\delta \boldsymbol{v}(t+\Delta t) &= \delta \boldsymbol{v} + ((\boldsymbol{R}(\boldsymbol{b}_a - \tilde{\boldsymbol{a}}))^{\wedge}\delta\boldsymbol{\theta} - \boldsymbol{R}\delta\boldsymbol{b}_a + \delta\boldsymbol{g})\Delta t - \boldsymbol{\eta}_v, && (3.12b)\\
\delta \boldsymbol{\theta}(t+\Delta t) &= \delta\boldsymbol{\theta} - \boldsymbol{R}\delta\boldsymbol{b}_g\Delta t - \boldsymbol{\eta}_\theta, && (3.12c)\\
\delta \boldsymbol{b}_g(t+\Delta t) &= \delta\boldsymbol{b}_g + \boldsymbol{\eta}_{bg}, && (3.12d)\\
\delta \boldsymbol{b}_a(t+\Delta t) &= \delta\boldsymbol{b}_a + \boldsymbol{\eta}_{ba}, && (3.12e)\\
\delta \boldsymbol{g}(t+\Delta t) &= \delta\boldsymbol{g}. && (3.12f)
\end{aligned}
$$

$\boldsymbol{F}$ 矩阵的形式也会随之改变，如下：

$$
\boldsymbol{F} = \begin{bmatrix}
\boldsymbol{I} & \boldsymbol{I}\Delta t & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & \boldsymbol{I} & (\boldsymbol{R}(\boldsymbol{b}_a - \tilde{\boldsymbol{a}}))^{\wedge}\Delta t & \mathbf{0} & -\boldsymbol{R}\Delta t & \boldsymbol{I}\Delta t\\
\mathbf{0} & \mathbf{0} & \boldsymbol{I} & -\boldsymbol{R}\Delta t & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{I}
\end{bmatrix}. \tag{3.13}
$$

代码中的改动如图 2：

```cpp
    // nominal state 递推
    VecT new_p = p_ + v_ * dt + 0.5 * (R_ * (imu.acce_ - ba_)) * dt * dt + 0.5 * g_ * dt * dt;
    VecT new_v = v_ + R_ * (imu.acce_ - ba_) * dt + g_ * dt;
//    SO3 new_R = R_ * SO3::exp((imu.gyro_ - bg_) * dt);
    // TODO: 左乘运动学方程
    SO3 new_R = SO3::exp((imu.gyro_ - bg_) * dt) * R_;

    R_ = new_R;
    v_ = new_v;
    p_ = new_p;
    // 其余状态维度不变

    // error state 递推
    // 计算运动过程雅可比矩阵 F，见(3.47)
    // F实际上是稀疏矩阵，也可以不用矩阵形式进行相乘而是写成散装形式，这里为了教学方便，使用矩阵形式
//    Mat18T F = Mat18T::Identity();                                                 // 主对角线
//    F.template block<3, 3>(0, 3) = Mat3T::Identity() * dt;                         // p 对 v
//    F.template block<3, 3>(3, 6) = -R_.matrix() * SO3::hat(imu.acce_ - ba_) * dt;  // v 对 theta
//    F.template block<3, 3>(3, 12) = -R_.matrix() * dt;                             // v 对 ba
//    F.template block<3, 3>(3, 15) = Mat3T::Identity() * dt;                        // v 对 g
//    F.template block<3, 3>(6, 6) = SO3::exp(-(imu.gyro_ - bg_) * dt).matrix();     // theta 对 theta
//    F.template block<3, 3>(6, 9) = -Mat3T::Identity() * dt;                        // theta 对 bg
    // TODO: 左乘误差状态的离散运动学方程
    Mat18T F = Mat18T::Identity();                                                       // 主对角线
    F.template block<3, 3>(0, 3) = Mat3T::Identity() * dt;                               // p 对 v
    F.template block<3, 3>(3, 6) = SO3::hat(R_.matrix() * (ba_ - imu.acce_)) * dt;       // v 对 theta
    F.template block<3, 3>(3, 12) = -R_.matrix() * dt;                                   // v 对 ba
    F.template block<3, 3>(3, 15) = Mat3T::Identity() * dt;                              // v 对 g
    F.template block<3, 3>(6, 9) = -R_.matrix() * dt;                                    // theta 对 bg
```

图 2: 左乘模型下的运动学方程的代码实现

3.d 左乘模型下的 RTK 观测方程

左乘模型下，RTK 的观测方程也需要改动，具体改动如下：

$$
\boldsymbol{R}_{\mathrm{gnss}} = \mathrm{Exp}(\delta\boldsymbol{\theta})\boldsymbol{R}. \tag{3.14}
$$

此时，观测方程可以写为：

$$
\boldsymbol{z}_{\delta\boldsymbol{\theta}} = \boldsymbol{h}(\delta\boldsymbol{\theta}) = \mathrm{Log}(\boldsymbol{R}_{\mathrm{gnss}}\boldsymbol{R}^{\top}). \tag{3.15}
$$

$z_{\delta\theta}$ 仍是对 $\delta\boldsymbol{\theta}$ 的直接观测，雅可比矩阵形式不变。ESKF 中的更新量 $\boldsymbol{z}-\boldsymbol{h}(\boldsymbol{x})$ 的形式如下：

$$\boldsymbol{z}-\boldsymbol{h}(\boldsymbol{x})=[\boldsymbol{p}_{\mathrm{gnss}}-\boldsymbol{p},\mathrm{Log}(\boldsymbol{R}_{\mathrm{gnss}}\boldsymbol{R}^{\top})]^{\top}. \tag{3.16}$$

代码中的改动如图 3：

```
    // 更新x和cov
    Vec6d innov = Vec6d::Zero();
    innov.template head<3>() = (pose.translation() - p_);          // 平移部分
//    innov.template tail<3>() = (R_.inverse() * pose.so3()).log();  // 旋转部分(3.67)
    // TODO: 左乘模型下的观测更新量
    innov.template tail<3>() = (pose.so3() * R_.inverse()).log();
```

图 3: 使用 RTK 的观测，在左乘模型下的 ESKF 的更新量的代码实现

3.e 左乘模型下的切空间投影

各变量定义与书中一致，左乘扰动模型下，书中的式 (3.58) 改写为：

$$\mathrm{Exp}(\delta\boldsymbol{\theta}^{+})\boldsymbol{R}^{+}=\mathrm{Exp}(\delta\boldsymbol{\theta}^{+})\mathrm{Exp}(\delta\boldsymbol{\theta}_k)\boldsymbol{R}_k=\mathrm{Exp}(\delta\boldsymbol{\theta})\boldsymbol{R}_k. \tag{3.17}$$

易得：

$$\mathrm{Exp}(\delta\boldsymbol{\theta}^{+})=\mathrm{Exp}(\delta\boldsymbol{\theta})\mathrm{Exp}(-\delta\boldsymbol{\theta}_k). \tag{3.18}$$

利用线性化后的 BCH 公式，可以得到：

$$\begin{aligned}\delta\boldsymbol{\theta}^{+}&=\delta\boldsymbol{\theta}-\delta\boldsymbol{\theta}_k-\frac{1}{2}\delta\boldsymbol{\theta}^{\wedge}\delta\boldsymbol{\theta}_k,\\&=\delta\boldsymbol{\theta}-\delta\boldsymbol{\theta}_k+\frac{1}{2}\delta\boldsymbol{\theta}_k^{\wedge}\delta\boldsymbol{\theta}.\end{aligned} \tag{3.19}$$

最后得到：

$$\boldsymbol{J}_{\boldsymbol{\theta}}=\frac{\partial\delta\boldsymbol{\theta}^{+}}{\partial\delta\boldsymbol{\theta}}\approx\boldsymbol{I}+\frac{1}{2}\delta\boldsymbol{\theta}_k^{\wedge}. \tag{3.20}$$

除去以上讨论的这些，ESKF 旋转量的更新也改用左乘误差状态量的形式。
代码实现如图 4：

```
    /// 更新名义状态变量，重置error state
    void UpdateAndReset() {
        p_ += dx_.template block<3, 1>(0, 0);
        v_ += dx_.template block<3, 1>(3, 0);
//        R_ = R_ * SO3::exp(dx_.template block<3, 1>(6, 0));
        // TODO: R的误差更新也用左乘形式
        R_ = SO3::exp( omega: dx_.template block<3, 1>(6, 0)) * R_;

        if (options_.update_bias_gyro_) {
            bg_ += dx_.template block<3, 1>(9, 0);
        }

        if (options_.update_bias_acce_) {
            ba_ += dx_.template block<3, 1>(12, 0);
        }

        g_ += dx_.template block<3, 1>(15, 0);

        ProjectCov();
        dx_.setZero();
    }

    /// 对P阵进行投影，参考式(3.63)
    void ProjectCov() {
        Mat18T J = Mat18T::Identity();
//        J.template block<3, 3>(6, 6) = Mat3T::Identity() - 0.5 * SO3::hat(dx_.template block<3, 1>(6, 0));
        // TODO: 左乘模型下的切空间投影Jacobian
        J.template block<3, 3>(6, 6) = Mat3T::Identity() - 0.5 * SO3::hat( omega: dx_.template block<3, 1>(6, 0));

        cov_ = J * cov_ * J.transpose();
    }
```

图 4: 切空间投影和 ESKF 旋转量的更新的代码实现

3.f 代码运行结果

左乘扰动模型下，组合导航运行结果如图 5，与右乘扰动模型的结果一致：

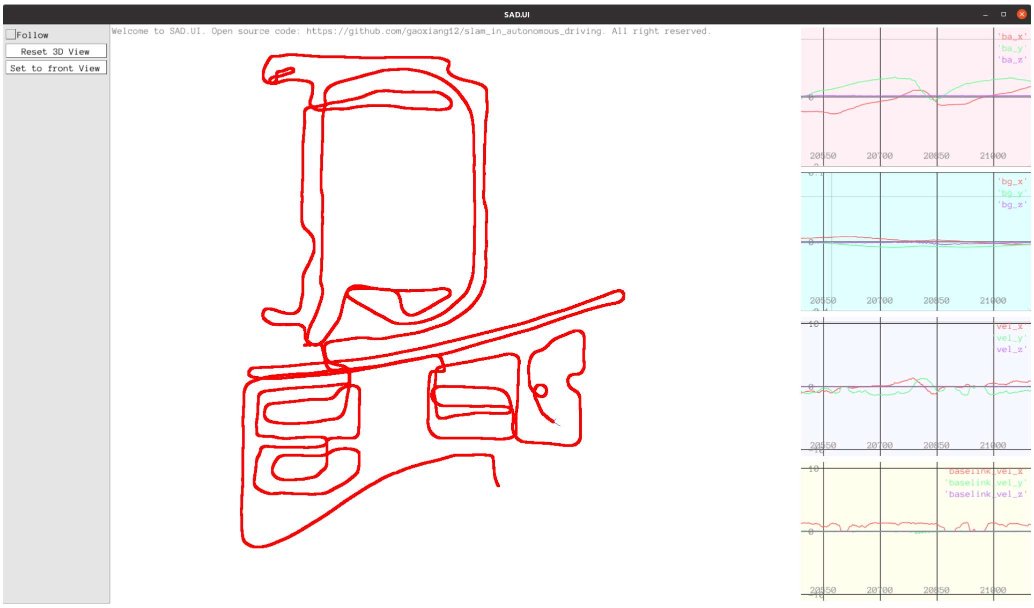


图 5: 代码运行结果