sad-L3-hw

Yixiao Feng

04/06/2023

1 推导题

$\boldsymbol{r}_{\Delta \boldsymbol{p}_{ij}}$ 对 $\boldsymbol{\phi}_i$ 的雅可比和 $\boldsymbol{r}_{\Delta \boldsymbol{v}_{ij}}$ 对 $\boldsymbol{\phi}_i$ 的雅可比推导方式几乎一致。具体推导如下：

$$
\begin{aligned}
\boldsymbol{r}_{\Delta \boldsymbol{p}_{ij}}(\boldsymbol{R}_i \mathrm{Exp}(\delta \boldsymbol{\phi}_i)) &= (\boldsymbol{R}_i \mathrm{Exp}(\delta \boldsymbol{\phi}_i))^\top (\boldsymbol{p}_j - \boldsymbol{p}_i - \boldsymbol{v}_i \Delta t_{ij} - \frac{1}{2}\boldsymbol{g}\Delta t_{ij}^2) - \Delta \tilde{\boldsymbol{p}}_{ij}, \\
&= \mathrm{Exp}(-\delta \boldsymbol{\phi}_i) \boldsymbol{R}_i^\top (\boldsymbol{p}_j - \boldsymbol{p}_i - \boldsymbol{v}_i \Delta t_{ij} - \frac{1}{2}\boldsymbol{g}\Delta t_{ij}^2) - \Delta \tilde{\boldsymbol{p}}_{ij}, \\
&= (\boldsymbol{I} - \delta \boldsymbol{\phi}_i^\wedge) \boldsymbol{R}_i^\top (\boldsymbol{p}_j - \boldsymbol{p}_i - \boldsymbol{v}_i \Delta t_{ij} - \frac{1}{2}\boldsymbol{g}\Delta t_{ij}^2) - \Delta \tilde{\boldsymbol{p}}_{ij}, \\
&= \boldsymbol{r}_{\Delta \boldsymbol{p}_{ij}}(\boldsymbol{R}_i) + (\boldsymbol{R}_i^\top (\boldsymbol{p}_j - \boldsymbol{p}_i - \boldsymbol{v}_i \Delta t_{ij} - \frac{1}{2}\boldsymbol{g}\Delta t_{ij}^2))^\wedge \delta \boldsymbol{\phi}_i.
\end{aligned} \tag{1.1}
$$

所以我们得到：

$$
\frac{\partial \boldsymbol{r}_{\Delta \boldsymbol{p}_{ij}}}{\partial \boldsymbol{\phi}_i} = (\boldsymbol{R}_i^\top (\boldsymbol{p}_j - \boldsymbol{p}_i - \boldsymbol{v}_i \Delta t_{ij} - \frac{1}{2}\boldsymbol{g}\Delta t_{ij}^2))^\wedge. \tag{1.2}
$$

2 实现由 Odom 数据触发的图优化

实现 Odom 数据触发的图优化只需要在 AddOdom 函数中加入触发优化的函数 Optimize()，使得每次接收到轮速计的数据，就触发图优化。并将 GNSS 触发图优化的那部分去掉。具体代码如图 1：

```
    // 积分到GNSS时刻
    pre_integ_->Integrate(last_imu_, gnss.unix_time_ - current_time_);

    current_time_ = gnss.unix_time_;
    *this_frame_ = pre_integ_->Predict(*last_frame_, options_.gravity_);

    // Optimize();

    last_frame_ = this_frame_;
    last_gnss_ = this_gnss_;
}

void GinsPreInteg::AddOdom(const sad::Odom& odom) {
    last_odom_ = odom;
    last_odom_set_ = true;
    Optimize();
}
```

图 1: Odom 数据触发图优化代码实现

Odom 触发的图优化运行的结果如图 2：

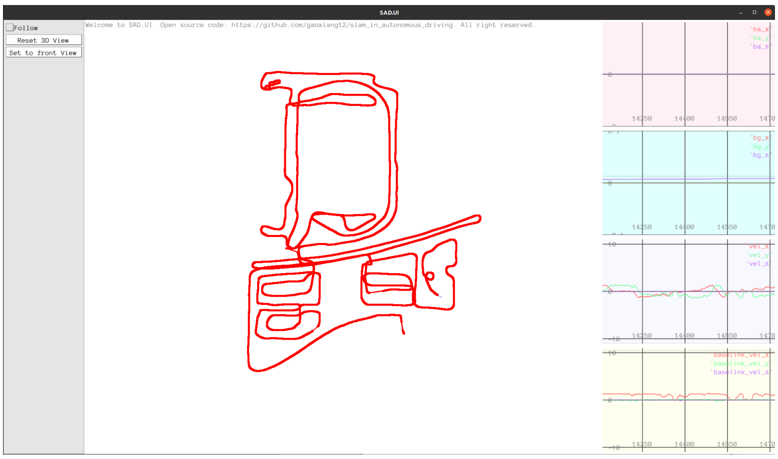
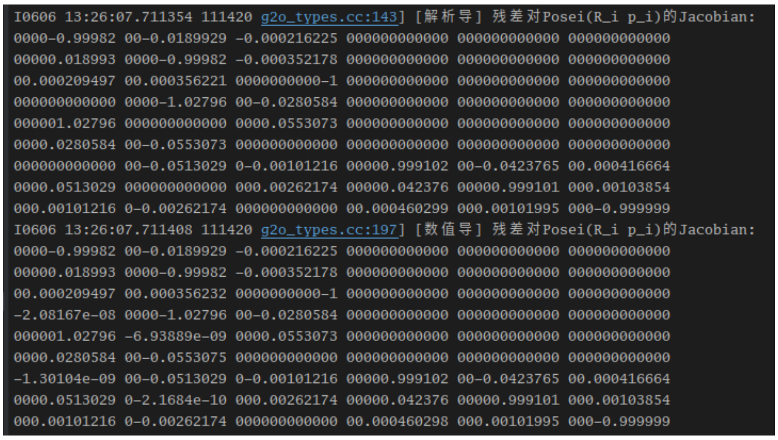


图 2: 代码运行结果

3 利用数值求导工具，验证本书实验中的雅可比矩阵的正确性

应用导数的定义，给图优化关联的几个状态量的不同维度应用广义加法加上一个小量（代码实现中把这个扰动定义为 1e-9，与 g2o 自动求导相同），计算出加入扰动后的残差项，与未加入扰动的残差项相减后除以这个小量，就能得到数值雅克比。具体的代码实现见 g2o_types.cc 文件的 linearizeOplus() 函数，代码实现了残差对 Posei 的数值雅克比计算，残差对其他状态量的雅克比计算类似。代码运行中某次优化的对比结果如图 3：



```
I0606 13:26:07.711354 111420 g2o_types.cc:143] [解析导] 残差对Posei(R_i p_i)的Jacobian:
0000-0.99982 00-0.0189929 -0.000216225 000000000000 000000000000 000000000000
00000.018993 0000-0.99982 -0.000352178 000000000000 000000000000 000000000000
00.000209497 00.000356221 0000000000-1 000000000000 000000000000 000000000000
000000000000 0000-1.02796 00-0.0280584 000000000000 000000000000 000000000000
000001.02796 000000000000 0000.0553073 000000000000 000000000000 000000000000
0000.0280584 00-0.0553073 000000000000 000000000000 000000000000 000000000000
000000000000 00-0.0513029 0-0.00101216 00000.999102 00-0.0423765 00.000416664
0000.0513029 000000000000 000.00262174 00000.042376 00000.999101 000.00103854
000.00101216 0-0.00262174 000000000000 00.000460299 000.00101995 000-0.999999
I0606 13:26:07.711408 111420 g2o_types.cc:197] [数值导] 残差对Posei(R_i p_i)的Jacobian:
0000-0.99982 00-0.0189929 -0.000216225 000000000000 000000000000 000000000000
00000.018993 0000-0.99982 -0.000352178 000000000000 000000000000 000000000000
00.000209497 00.000356232 0000000000-1 000000000000 000000000000 000000000000
-2.08167e-08 0000-1.02796 00-0.0280584 000000000000 000000000000 000000000000
000001.02796 -6.93889e-09 0000.0553073 000000000000 000000000000 000000000000
0000.0280584 00-0.0553075 000000000000 000000000000 000000000000 000000000000
-1.30104e-09 00-0.0513029 0-0.00101216 00000.999102 00-0.0423765 00.000416664
0000.0513029 0-2.1684e-10 000.00262174 00000.042376 00000.999101 000.00103854
000.00101216 0-0.00262174 000000000000 00.000460298 000.00101995 000-0.999999
```

图 3: 数值求导和解析求导对比结果