第三讲作业

崔华坤

达闼科技（北京）有限公司

2018.3.16

2 群的性质

1.{Z,+}是否为群？若是，验证其满足群定义；若不是，说明理由。其中 Z 为整数集

答：封闭性：$\forall \mathrm{Z}_1, Z_2 \in A, \mathrm{Z}_1 + Z_2 \in A$

结合律：$\forall \mathrm{Z}_1, Z_2, Z_3 \in A, (\mathrm{Z}_1 + Z_2) + Z_3 = \mathrm{Z}_1 + (Z_2 + Z_3)$

幺元：$\exists \mathrm{Z}_0 = 0 \in A, \mathrm{Z}_0 + Z_1 = Z_1 + \mathrm{Z}_0 = \mathrm{Z}_1$

逆：$\forall \mathrm{Z} \in A, \exists Z^{-1} = -Z \in A, s.t. Z + (-Z) = 0 = \mathrm{Z}_0$

满足群定义，故为群。

2.{N,+}是否为群？若是，验证其满足群定义；若不是，说明理由。其中 N 为自数集。

答：封闭性：$\forall \mathrm{N}_1, N_2 \in A, \mathrm{N}_1 + N_2 \in A$

结合律：$\forall \mathrm{N}_1, N_2, N_3 \in A, (\mathrm{N}_1 + N_2) + N_3 = \mathrm{N}_1 + (N_2 + N_3)$

幺元：$\exists \mathrm{N}_0 = 0 \in A, \mathrm{N}_0 + N_1 = N_1 + \mathrm{N}_0 = \mathrm{N}_1$

逆：$\forall \mathrm{N} \in A, \exists N^{-1} = -N \notin A$

因自然数为非负整数，对于正整数的逆为负整数，不属于自然数，故不满足群定义，不为群。

3 验证向量叉乘的李代数性质

答：

$\forall \mathrm{X}, \mathrm{Y}, \mathrm{Z} \in R^3, \mathrm{a}, \mathrm{b} \in \mathrm{R}$

封闭性：$[X,Y] = \mathrm{X} \times \mathrm{Y} \in R^3$

双线性：$[aX+bY,Z] = (aX+bY)\times Z = a(X\times Z) + b(Y\times Z)$

$[Z,aX+bY] = \mathrm{Z}\times(aX+bY) = a(Z\times X) + b(Z\times Y)$

自反性：$[X,X] = \mathrm{X}\times\mathrm{X} = 0$

雅可比等价：$[X,[Y,Z]] + [Y,[Z,X]] + [Z,[X,Y]] = X\times Y\times Z + Y\times Z\times X + Z\times X\times Y = 0$

故向量叉乘均满足李代数性质。

4 推导 SE(3) 的指数映射

答：$\xi^\wedge = \begin{bmatrix} \phi^\wedge & \rho \\ 0^T & 0 \end{bmatrix}$

$\xi^\wedge\xi^\wedge = \begin{bmatrix} \phi^\wedge & \rho \\ 0^T & 0 \end{bmatrix}\begin{bmatrix} \phi^\wedge & \rho \\ 0^T & 0 \end{bmatrix} = \phi^\wedge\xi^\wedge$

$\xi^\wedge\xi^\wedge\xi^\wedge = (\phi^\wedge)^2\xi^\wedge$

$$exp(\xi^\wedge) = \sum_{n=0}^{\infty}\frac{(\xi^\wedge)^n}{n!} = \mathrm{I} + \xi^\wedge + \frac{(\xi^\wedge)^2}{2!} + \frac{(\xi^\wedge)^3}{3!} + \frac{(\xi^\wedge)^4}{4!} + \cdots$$

$$= \mathrm{I} + \xi^\wedge + \frac{\phi^\wedge\xi^\wedge}{2!} + \frac{(\phi^\wedge)^2\xi^\wedge}{3!} + \frac{(\phi^\wedge)^3\xi^\wedge}{4!} + \cdots$$

$$= \mathrm{I} + \left[\frac{(\phi^\wedge)^0}{1!} + \frac{(\phi^\wedge)^1}{2!} + \frac{(\phi^\wedge)^2}{3!} + \frac{(\phi^\wedge)^3}{4!} + \cdots\right]\xi^\wedge$$

$$= \mathrm{I} + \sum_{n=0}^{\infty}\frac{(\phi^\wedge)^n}{(n+1)!}\xi^\wedge$$

$$= \mathrm{I} + \begin{bmatrix} \sum_{n=0}^{\infty}\frac{(\phi^\wedge)^n}{(n+1)!}\phi^\wedge & \sum_{n=0}^{\infty}\frac{(\phi^\wedge)^n}{(n+1)!}\rho \\ 0^T & 0 \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{n=0}^{\infty}\frac{(\phi^\wedge)^n}{n!} & \sum_{n=0}^{\infty}\frac{(\phi^\wedge)^n}{(n+1)!}\rho \\ 0^T & 1 \end{bmatrix}$$

下面我们讨论左雅可比形式：

$$\sum_{n=0}^{\infty}\frac{(\phi^\wedge)^n}{(n+1)!} \tag{1}$$

令$\phi^\wedge = \theta\alpha^\wedge$，则有如下关系成立：

$\alpha^\wedge\alpha^\wedge = \alpha\alpha^T - I$

$(\alpha^\wedge)^3 = -\alpha^\wedge$

$(\alpha^\wedge)^4 = (\alpha^\wedge)^2$

则式(1)可展开为：

$$\sum_{n=0}^{\infty}\frac{(\phi^\wedge)^n}{(n+1)!} = 1 + \frac{\theta\alpha^\wedge}{2!} + \frac{\theta^2(\alpha^\wedge)^2}{3!} - \frac{\theta^3\alpha^\wedge}{4!} - \frac{\theta^4(\alpha^\wedge)^2}{5!} + \frac{\theta^5\alpha^\wedge}{6!} + \frac{\theta^6(\alpha^\wedge)^2}{7!} - \frac{\theta^7\alpha^\wedge}{8!} + \cdots$$

$$= 1 + \left[-\frac{1}{\theta}\left(1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \frac{\theta^8}{8!} - \cdots\right) + \frac{1}{\theta}\right]\alpha^\wedge$$

$$+ \left[-\frac{1}{\theta}\left(\theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \cdots\right) + 1\right](\alpha^\wedge)^2$$

$$= 1 + \frac{1-\cos\theta}{\theta}\alpha^\wedge + \frac{\theta - \sin\theta}{\theta}(\alpha^\wedge)^2$$

$$= 1 + \frac{1-\cos\theta}{\theta}\alpha^\wedge + \frac{\theta - \sin\theta}{\theta}(\alpha\alpha^T - I)$$

$$= \frac{\sin\theta}{\theta}I + \left(1 - \frac{\sin\theta}{\theta}\right)\alpha\alpha^T + \frac{1-\cos\theta}{\theta}\alpha^\wedge$$

5 伴随　$R exp(\xi^\wedge)R^T = exp\left[(R\xi)^\wedge\right]$

答：左边将矩阵的指数映射展开可得：

$$\mathrm{R}\exp(\xi^\wedge)R^T = R\sum_{n=0}^{\infty}\frac{(\xi^\wedge)^n}{n!}R^T = \sum_{n=0}^{\infty}\frac{R(\xi^\wedge)^n R^T}{n!}$$

$$= \sum_{n=0}^{\infty}\frac{(R\xi^\wedge R^T)(R\xi^\wedge R^T)(R\xi^\wedge R^T)\cdots}{n!} = \sum_{n=0}^{\infty}\frac{(R\xi^\wedge R^T)^n}{n!}$$

$$= \exp(R\xi^\wedge R^T)$$

因此，我们只要证明$R\xi^\wedge R^T = (R\xi)^\wedge$即可，下面进行推导：

我们将式子左右两边右乘一个向量 v，则左式可化简为：

$$R\xi^\wedge R^T v = R(\xi \times R^T v) = (R\xi) \times (RR^T v) = (R\xi)^\wedge v = 右边$$

6 轨迹的描绘

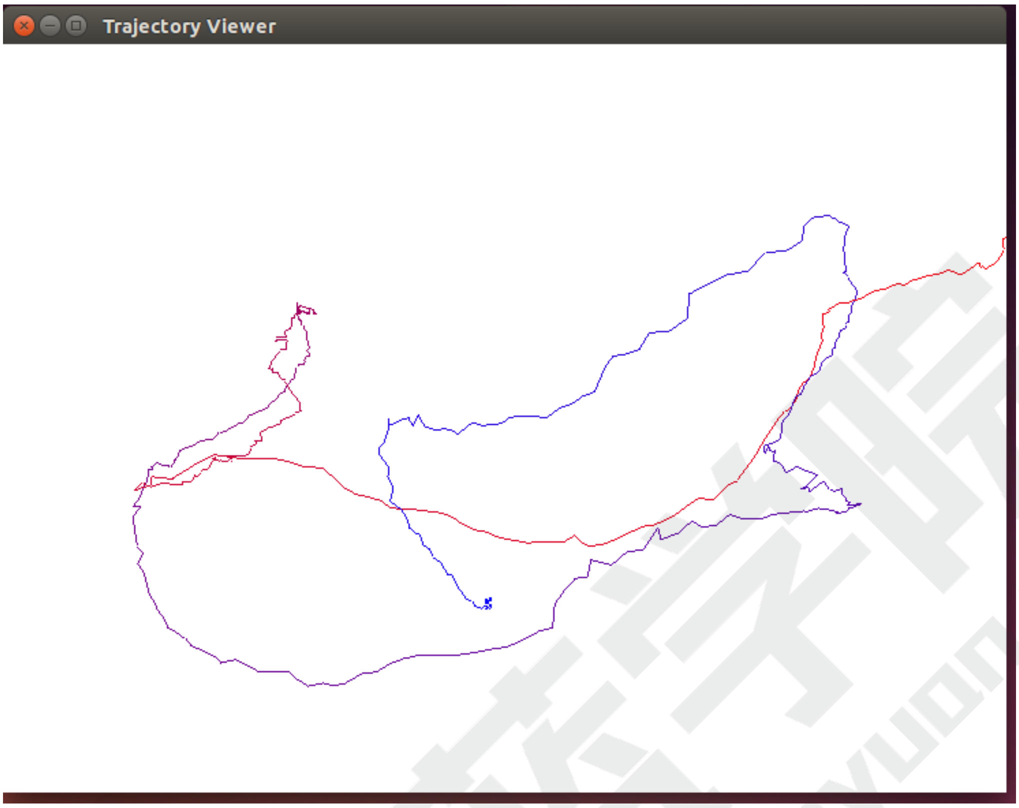
1. 事实上，TWC 的平移部分即构成了机器人的轨迹。它的物理意义是什么？为何画出 TWC 的平移部分就得到了机器人的轨迹？

答：TWC 平移部分的物理意义是表示经过旋转校正后的相机坐标系 C 原点在世界坐标系 W 中的坐标位置，而相机坐标系原点就代表了机器人，因此 TWC 的平移部分就是机器人在世界坐标系中的位置轨迹。

2. 画轨迹：

答：

```
// path to trajectory file
string trajectory_file = "../trajectory.txt";
ifstream ifTraject;
Sophus::SE3 T;
Eigen::Vector3d t;
Eigen::Quaterniond q;
double timestamp;

// function for plotting trajectory, don't edit this code
// start point is red and end point is blue
void DrawTrajectory(vector<Sophus::SE3, Eigen::aligned_allocator<Sophus::SE3> >);

int main(int argc, char **argv) {
    cout<<"main..."<<endl;
    vector<Sophus::SE3, Eigen::aligned_allocator<Sophus::SE3> > poses;

    /// implement pose reading code
    // start your code here (5~10 lines)
    ifTraject.open(trajectory_file.c_str());
    if(!ifTraject.is_open())
    {
        cout<<"file is empty!"<<endl;
        return -1;
    }

    string sFileLine;
    while(getline(ifTraject, sFileLine) && !sFileLine.empty())
    {
    istringstream iss(sFileLine);
    iss >> timestamp>>t[0]>>t[1]>>t[2]>>q.x()>>q.y()>>q.z()>>q.w();
    T = Sophus::SE3(q,t);
    poses.push_back(T);
    }
    ifTraject.close();
    // end your code here

    // draw trajectory in pangoli
    DrawTrajectory(poses);
    return 0;
}
```



7 * 轨迹的误差

```
int main()
{
cout<<"traceError main ..."<<endl;
ifsGT.open(GTPath.c_str());
ifsEs.open(EsPath.c_str());
if(!ifsGT.is_open() || !ifsEs.is_open())
{
cerr<<"txt is not opened!"<<endl;
return -1;
}

int num = 0;
string sGTLine,sEsLine;
while(getline(ifsGT,sGTLine) && getline(ifsEs,sEsLine)
        &&!sGTLine.empty() && !sEsLine.empty())
{
istringstream issGT(sGTLine);
istringstream issEs(sEsLine);
issGT >>time_g>>t_g[0]>>t_g[1]>>t_g[2]>>q_g.x()>>q_g.y()>>q_g.z()>>q_g.w();
issEs >>time_e>>t_e[0]>>t_e[1]>>t_e[2]>>q_e.x()>>q_e.y()>>q_e.z()>>q_e.w();
T_W_Cg = SE3(q_g, t_g);
T_W_Ce = SE3(q_e, t_e);
kesi = (T_W_Cg.inverse() * T_W_Ce).log();
err += kesi.transpose() * kesi;
num++;
}

RMSE = sqrt(err/num);
cout<<"RMSE: "<<RMSE<<endl;
return 1;
}
```

```
stevencui@ubuntu:~/Project/SLAMCourse/l3-7/build$ make
Scanning dependencies of target traceError
[ 50%] Building CXX object CMakeFiles/traceError.dir/traceError.cpp.o
[100%] Linking CXX executable traceError
[100%] Built target traceError
stevencui@ubuntu:~/Project/SLAMCourse/l3-7/build$ ./traceError
traceError main ...
RMSE: 2.20728
stevencui@ubuntu:~/Project/SLAMCourse/l3-7/build$
```