

IMU 应用讲解 (第一期)

唐宝芳 李文涛 肖斯凯

2020 年 11 月 25 日

主要内容

- 1 旋转运动学
- 2 IMU 测量模型
- 3 近期 IMU 应用中的问题及解决

IMU 应用讲解计划

- 第一期: 旋转运动学; IMU 测量模型
- 第二期: IMU 误差模型; IMU 标定
- 第三期: 预积分 (上)
- 第四期: 预积分 (下)

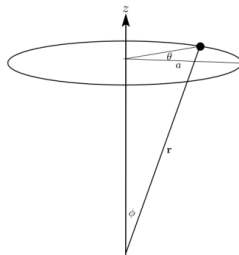
主要内容

- 1 旋转运动学
- 2 IMU 测量模型
- 3 近期 IMU 应用中的问题及解决

旋转运动学

- 粒子在坐标系中 $z = h$ 中的平面做圆周运动, 坐标为: $r = (a \cos \theta, a \sin \theta, h)^T$, 对坐标求导得:

$$\begin{aligned} \dot{r} &= (-a\dot{\theta} \sin \theta, a\dot{\theta} \cos \theta, 0)^T \\ &= \begin{bmatrix} 0 & -\dot{\theta} & 0 \\ \dot{\theta} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a \cos \theta \\ a \sin \theta \\ h \end{bmatrix} \\ &= \hat{w} r \end{aligned} \quad (1)$$



其中, \hat{w} 是一个反对称矩阵, $w = (0, 0, \dot{\theta})$, $\dot{\theta}$ 是角速度.

旋转运动学

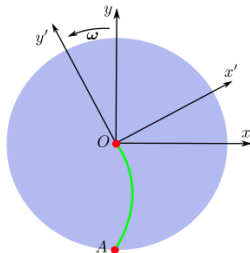
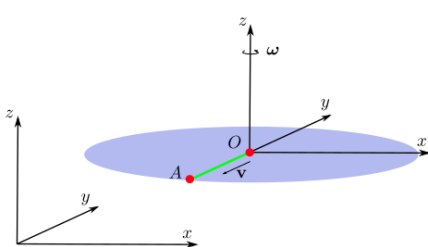
- 旋转矩阵是一个正交矩阵. 它的行列式为 1, 且每个列向量都是单位向量且相互正交, 它的逆等于它的转置.
- 旋转矩阵求导:

$$\begin{aligned}
 \dot{R}_{ib} &= \lim_{\Delta t \rightarrow 0} \frac{R_{ib} \exp([w^b \Delta t]^\wedge) - R_{ib}}{\Delta t} \\
 &= \lim_{\Delta t \rightarrow 0} \frac{R_{ib} (\exp([w^b \Delta t]^\wedge) - I)}{\Delta t} \\
 &\approx R_{ib} [w^b]^\wedge \\
 &= [R_{ib} w^b]^\wedge R_{ib} \\
 &= [w^i]^\wedge R_{ib}
 \end{aligned} \tag{2}$$

旋转运动学

更复杂一点的情况：一个旋转的水平光滑圆盘上，有一个光滑的小球，从圆心沿着半径向外运动。

- 从圆盘旋转坐标系来观察，小球轨迹如何？
- 从世界的坐标系来观察，小球轨迹如何？
- 科氏力, 离心力, 欧拉力？

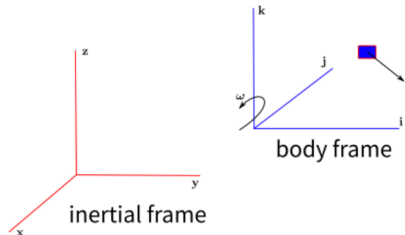


旋转运动学

- 质量块在 body 坐标系下的坐标为: $r^b = (x_1, x_2, x_3)^T$
- 忽略平移, 只考虑旋转, 旋转到惯性坐标系下: $r^i = R_{ib}r^b$
- 对时间求导:

$$\begin{aligned}
 \dot{r} &= R_{ib}\dot{r}^b + \dot{R}_{ib}r^b \\
 &= R_{ib}\dot{r}^b + R_{ib}[w^b]^{\wedge}r^b \\
 &= R_{ib}\dot{r}^b + [R_{ib}w^b]^{\wedge}R_{ib}r^b \\
 &= R_{ib}v^b + [w^i]^{\wedge}r^i \\
 v &= v^i + [w^i]^{\wedge}r^i \Leftrightarrow v^i = v - [w^i]^{\wedge}r_i
 \end{aligned}
 \tag{3}$$

其中, $w^i = R_{ib}w^b$, $v^i = R_{ib}v^b$, 表示 body 坐标系的角速度或线速度在 I 系下的表示.



旋转运动学

● 对速度求导:

$$\begin{aligned}
 \ddot{r} &= R_{ib}\dot{v}^b + \dot{R}_{ib}v^b + [w^i]^{\wedge}\dot{r}^i + [\dot{R}_{ib}w^b + R_{ib}\dot{w}^b]^{\wedge}r^i \\
 &= R_{ib}\dot{v}^b + \dot{R}_{ib}v^b + [w^i]^{\wedge}\dot{r}^i + [R_{ib}\dot{w}^b]^{\wedge}r^i \\
 &= R_{ib}a^b + [w^i]^{\wedge}v^i + [w^i]^{\wedge}[v^i + (w^i)^{\wedge}r^i] + [\dot{w}^i]^{\wedge}r^i \\
 &= R_{ib}a^b + 2[w^i]^{\wedge}v^i + [w^i]^{\wedge}([w^i]^{\wedge}r^i) + [\dot{w}^i]^{\wedge}r^i
 \end{aligned} \tag{4}$$

$$\Rightarrow a^i = a - \underbrace{2[w^i]^{\wedge}v^i}_{\text{Coriolis force}} - \underbrace{[w^i]^{\wedge}([w^i]^{\wedge}r^i)}_{\text{centrifugal force}} - \underbrace{[\dot{w}^i]^{\wedge}r^i}_{\text{Euler force}}$$

其中, $v^i = R_{ib}v^b$, $a^i = R_{ib}a^b$, 表示物体在 body 下的速度或加速度在 I 系下的表示.

在旋转坐标系下观察, 运动的物体 (运动方向和旋转轴不为同一个轴时) 会受到科氏力的作用.

主要内容

1 旋转运动学

2 IMU 测量模型

- 加速度计测量原理
- 陀螺仪的测量原理

3 近期 IMU 应用中的问题及解决

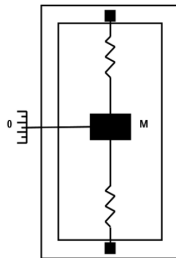
加速度计测量原理

- 其测量原理可以用一个质量块 + 弹簧 + 指示计来表示.
- 加速度计测量值 a_m 为弹簧拉力对应的加速度,

$$a_m = \frac{f}{m} = a - g \quad (5)$$

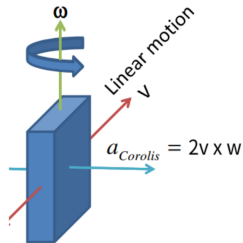
其中, f 为弹簧拉力, a 为物体在惯性系下的加速度, g 为重力加速度.

- 通过受力影响位移, 位移影响电容大小, 通过测量电流的方式获得 a_m



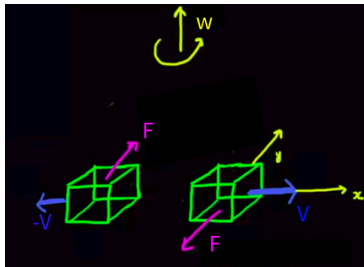
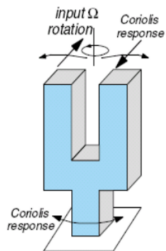
陀螺仪测量原理

- 陀螺仪主要用来测量物体的旋转角速度, 按测量原理分有震动陀螺/光纤陀螺等.
- 一般采用震动陀螺原理, 通过测量 Coriolis force 来间接得到角速度.
- 一个主动运动轴 + 一个敏感轴



音叉振动陀螺原理

- 音叉中间为旋转轴, 音叉左右两个质量块, 做方向相反的正弦运动, 质量块受到的科氏力方向相反.
- 为什么要这么做? 一个质量块行不行?



思考

- 实际上, 两个质量块不可能完全一致, 也就是说陀螺仪的测量会受到外部加速度的影响, 即常称的 G-sensitivity.
- 加速度计不需要考虑科氏力的影响吗?

主要内容

- 1 旋转运动学
- 2 IMU 测量模型
- 3 近期 IMU 应用中的问题及解决

近期 IMU 应用中的问题及解决 (一)

- 问题一: 同样是右手坐标系, 不同的是 z 轴上的数值取反 (原来重力大小为-9.8, 需要改成 9.8), 有两种方式.
- 方式一:

```
1 //pentu_ig1.urdf
2 <joint name="imu_link_joint" type="fixed">
3   <parent link="base_footprint" />
4   <child link="imu" />
5   <origin xyz="0_0_0" rpy="3.14_0_0"/>
6 </joint>
```

```
1 // SensorBridge::HandleImuMessage()
2 imu_data->angular_velocity = Eigen::Vector3d{imu_data->angular_velocity[0],
3   -imu_data->angular_velocity[1], -imu_data->angular_velocity[2]};
4 imu_data->linear_acceleration = Eigen::Vector3d{imu_data->linear_acceleration[0],
5   -imu_data->linear_acceleration[1], imu_data->linear_acceleration[2]};
6 .....
7 trajectory_builder_->AddSensorData( sensor_id,
8   carto::sensor::ImuData{imu_data->time, imu_data->linear_acceleration,
9     imu_data->angular_velocity});
```


近期 IMU 应用中的问题及解决 (一)

● 方式二:

```
1 // pentu_ig1.urdf
2 <joint name="imu_link_joint" type="fixed">
3   <parent link="base_footprint" />
4   <child link="imu" />
5   <origin xyz="0_0_0" rpy="0_0_0"/>
6 </joint>
```

```
1 // SensorBridge::HandleImuMessage()
2 imu_data->linear_acceleration[2] = -imu_data->linear_acceleration[2];
3 .....
4 trajectory_builder_>AddSensorData( sensor_id,
5   carto::sensor::ImuData{ imu_data->time, imu_data->linear_acceleration,
6   imu_data->angular_velocity});
```

近期 IMU 应用中的问题及解决 (二)

- 问题二:IMU 安装不是水平的,base_footprint 到 imu 的静态 TF 的标定困难. 解决方法是开发一个自动标定 IMU 的代码.

```

1 //pentu_ig1.urdf
2 <joint name="imu_link_joint" type="fixed">
3   <parent link="base_footprint" />
4   <child link="imu" />
5   <origin xyz="0_0_0" rpy="-0.00478014_-0.000399168_0"/>
6 </joint>
    
```

近期 IMU 应用中的问题及解决 (二)

```
1 void ImuCallback(const sensor_msgs::ImuConstPtr& msg)
2 {
3     static int nums = 0;
4     static Eigen::Vector3d calibr_sum;
5     if(++nums > 500)
6         return;
7     if(nums <= 500)
8     {
9         std::unique_ptr<cartographer::sensor::ImuData> imu_data = ToImuData(msg);
10        const Eigen::Quaterniond rotation = Eigen::Quaterniond::FromTwoVectors(
11            imu_data->linear_acceleration, Eigen::Vector3d{0, 0, -9.8});
12        Eigen::Vector3d calibr = cartographer::transform::RotationQuaternionToAngleAxisVector(rotation);
13        calibr_sum += calibr;
14    }
15    if(nums == 500)
16        std::cout << " [rpy: ] " << calibr_sum / 500 << std::endl;
17 }
```

讨论一波嗨!!!