

# 机器人学

作业 2：数学基础

**Robotics (2023-2024-2)**

## **Homework 2: Mathematical Foundations of Robotics**



姓名：赵四维

学号：521021910696

班级：ME3403-01

E-mail: racheus.11@sjtu.edu.cn

2024 年 3 月 9 日

## 旋转矩阵的求取、变换和证明

1. 求旋转矩阵它表示坐标系 {B} (一开始与 {A} 重合) 经过了如下运动: (a) 绕  $x_B$  旋转  $60^\circ$ ; (b) 绕  $z_B$  旋转  $60^\circ$ ; (c) 绕  $z_A$  旋转  $90^\circ$ 。

[Solution]:

[Click here to jump to the MATLAB code.](#)

先给出 (a)、(b)、(c) 三个旋转矩阵的表达式:

$$R_{x_B}(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix} \quad (1)$$

$$R_{z_B}(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 \\ \frac{\sqrt{3}}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$R_{z_A}(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

注意到 (1)、(2) 两个变换是在 {B} 坐标系下进行的, 是 "Euler Angle" 范畴, 在求矩阵的过程中要右乘; 而 (3) 是在 {A} 坐标系下进行的, 是 "Fixed Angle" 范畴, 在求矩阵的过程中要左乘。

因此, 最终旋转矩阵为:

$$R = R_{z_A}(\theta) R_{x_B}(\theta) R_{z_B}(\theta) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ 0 & \frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 \\ \frac{\sqrt{3}}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -0.433 & -0.25 & 0.866 \\ 0.50 & -0.866 & 0 \\ 0.75 & 0.433 & 0.5 \end{bmatrix}$$

2. 已知坐标系 {B} 相对于坐标系 {A} 的齐次变换矩阵为

$$\begin{bmatrix} ? & 0 & 1 & 1 \\ ? & 0 & 0 & 2 \\ ? & -1 & 0 & 3 \\ ? & 0 & 0 & 1 \end{bmatrix}$$

(1) 求第一列元素的值

(2) 求旋转运动的旋转轴和旋转角

(3) 若  $p^A = [1, 3, 5]^T$ , 求  $p^B$

[Solution]:

(1) 由增广后旋转矩阵的定义,  $R_{41} = 0$  是显然的。而矩阵的前三行前三列 (记为  $R' = [q_1, q_2, q_3]$ ) 构成的  $3 \times 3$  矩阵是一个正交矩阵, 且行列式为 1。因此, 设  $R_{11} = a, R_{21} = b, R_{31} = c$ , 则有:

$$\begin{cases} q_1 \cdot q_2 = 0 \rightarrow c = 0 \\ q_1 \cdot q_3 = 0 \rightarrow a = 0 \\ \det(R') = 1 \rightarrow (-1)^{(1+3)}b \times (-1) = 1 \rightarrow b = -1 \end{cases} \quad (4)$$

因此, 旋转矩阵的表示为:

$$R = \begin{bmatrix} 0 & 0 & 1 & 1 \\ -1 & 0 & 0 & 2 \\ 0 & -1 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

(2) [Click here to jump to the MATLAB code.](#)

本小题采用 MATLAB 的 `rotm2axang` 函数, 该函数可以将旋转矩阵转换为旋转轴和旋转角。

```
>> Rhw_2_2_main
Axis:
    -0.5774    0.5774   -0.5774

Angle:
    120.0000
```

图 1: 旋转矩阵转换为旋转轴和旋转角

因此旋转轴为  $\mathbf{e} = [-0.5774, 0.5774, -0.5774]$ , 旋转角为  $\psi = 120^\circ$ 。

(3) 由题意,  $p^A = [1, 3, 5]^T$ , 增广为  $[1, 3, 5, 1]^T$ 。则有:

$$p^B = \begin{bmatrix} 0 & 0 & 1 & 1 \\ -1 & 0 & 0 & 2 \\ 0 & -1 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \\ 5 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 1 \\ 0 \\ 1 \end{bmatrix} \quad (6)$$

因此,  $p^B = [6, 1, 0]^T$ 。

### 3. 设 $R$ 是一个旋转矩阵, 求证 $\det R = 1$ (坐标系为右手系)

[Proof:]

由于旋转矩阵是一个正交矩阵, 由正交矩阵的性质  $R^T R = I$ 。因此有:

$$|R^T R| = |I| = 1 \quad (7)$$

由行列式的性质  $|AB| = |A||B|$ , 又由矩阵转置的性质  $|A| = |A^T|$ , 因此有:

$$|R^T R| = |R^T||R| = |R||R| = (\det R)^2 = 1 \quad (8)$$

因此有  $|R| = \pm 1$ 。由题意, 在右手系下参考, 正交矩阵的  $n$  个列向量组成的标准正交基, 行列式必定为正数。因此有  $\det R = 1$ , 证毕。

## MATLAB 编程题

4. MATLAB 函数 “`eul2rotm`” 可以将欧拉角向量转化为旋转矩阵, 而 “`rotm2eul`” 函数可以实现旋转矩阵到欧拉角的反解。该函数有一个选项可决定旋转轴顺序, 选择 “ZYX” 顺序。已知测试数据如表 1 所示, 完成以下任务:

$\alpha$	$\beta$	$\gamma$
10	20	30
30	90	-55

1) 使用第一行数据和 “`rotx`” “`roty`” “`rotz`” 函数验证 “`eul2rotm`” 是绕定轴旋转的还是绕动轴旋转的, 并给出 `eul2rotm([ $\alpha, \beta, \gamma$ ], 'ZYX')` 对应的绕 xyz 轴分解式。

2) 实现该顺序的旋转矩阵-欧拉角反解程序, 并将反解结果与 “`rotm2eul`” 函数的反解结果、原参数作对比。请问结果与原参数是否相同? 若不相同, 请解释原因。

[Solution]:

(1) [Click here to jump to the MATLAB code.](#)

由 MATLAB 计算得到的结果, 可见结果是绕动轴旋转的, 即以每次旋转变换后的坐标系作为新的参考系进行旋转, 在矩阵乘法的过程中体现为右乘。

同样, 由 MATLAB 计算得到的结果, 如果绕 ‘XYZ’ 顺序旋转, 那么得到的分解式 (Euler Angle) 为:  $[28.4518^\circ, 22.2422^\circ, -1.1161^\circ]$ 。

(2) [Click here to jump to the MATLAB code.](#)

由 MATLAB 计算得到的结果, 可以看到, 旋转矩阵-欧拉角反解程序的结果与 “`rotm2eul`” 函数的反解结果、原参数是相同的。

因为在本题的求解过程中笔者使用了 `atan2` 函数，这一函数已在第一次作业中有所说明。该函数在机器人学中至关重要，其优势在于可以正确地处理四个象限的所有角度，并且避免了由于负数或者除数为 0 时而产生的错误。因此，可以保证结果的正确性。

$$\arctan 2(y, x) = \begin{cases} \arctan(\frac{y}{x}) & x > 0 \\ \arctan(\frac{y}{x}) + \pi & x < 0, y \geq 0 \\ \arctan(\frac{y}{x}) - \pi & x < 0, y < 0 \\ \frac{\pi}{2} & x = 0, y > 0 \\ -\frac{\pi}{2} & x = 0, y < 0 \\ \text{undefined} & x = 0, y = 0 \end{cases}$$

## 5. 实现齐次变换矩阵求逆函数，并与 MATLAB 函数 `inv` 比较运算结果和运算时间。

[Click here to jump to the MATLAB code.](#)

基本解决思路为：

$$T_A^B = \begin{bmatrix} R_A^B & p_{A0}^B \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} (R_B^A)^T & -R_B^A p_{A0}^B \\ 0 & 1 \end{bmatrix} = (T_B^A)^{-1}$$

在 MATLAB 中分别用自己的求逆函数和自带的 `inv` 函数，得到的结果是相同的。分别循环  $10^4$  次，求逆函数的运算时间为  $0.0370s$ ，而 `inv` 函数的运算时间为  $0.0060s$ 。因此，可以 MATLAB 自带的 `inv` 函数的运算速度更快。

```
Custom Inverse:
    0.8660    0.5000         0   -1.8660
   -0.5000    0.8660         0   -1.2320
         0         0    1.0000   -3.0000
         0         0         0    1.0000

Built-in Inverse:
    0.8660    0.5000         0   -1.8661
   -0.5000    0.8660         0   -1.2321
         0         0    1.0000   -3.0000
         0         0         0    1.0000

Custom Function Time:
    0.0370

Built-in Function Time:
    0.0060
```

图 2: 齐次变换矩阵求逆函数与 MATLAB 函数 `inv` 函数求解时间对比

---

代码块 求逆函数

---

```
function T_inv = inverse_homogeneous_transform(T)
    R = T(1:3, 1:3);
    p = T(1:3, 4);

    % Rotation matrix inv
    R_inv = R';

    % inverse of translation part
    p_inv = -R_inv * p;

    % Remake a new homogeneous transformation matrix
    T_inv = eye(4);
    T_inv(1:3, 1:3) = R_inv;
    T_inv(1:3, 4) = p_inv;
end
```

---

分析其原因, 经过观察, 在我自己的求逆函数中, 有两次矩阵提取运算, 有一次转置运算, 一次**矩阵乘法**, 七次赋值运算, 其中除了矩阵乘法的复杂度是  $O(n^2)$  外, 其他的复杂度都是  $O(n)$ 。在矩阵规模较小的时候, 时间差异不明显, 因此笔者采用循环  $10^4$  次的方式来进行对比。

经查阅资料, MATLAB 内置函数使用了一些高效的数值计算算法, 这些算法经过了精心设计和优化, 以在各种情况下都能提供良好的性能。包括但不限于 LU 分解、QR 分解、特征值分解等方法, 可以使得 `inv` 函数的运算速度更快<sup>[1]</sup>。

## Reference

[1]MATLAB Documentation. (n.d.). Retrieved October 10, 2021, from <https://ww2.mathworks.cn/help/matlab/>