中国空间站机械臂运动控制仿真

——《机器人技术与实践》实验报告

郑乔译 崔哲珺 陈启锐 金杰鹏 王伟杰

2024 年 12 月 7 日

目录

1 实验内容与要求

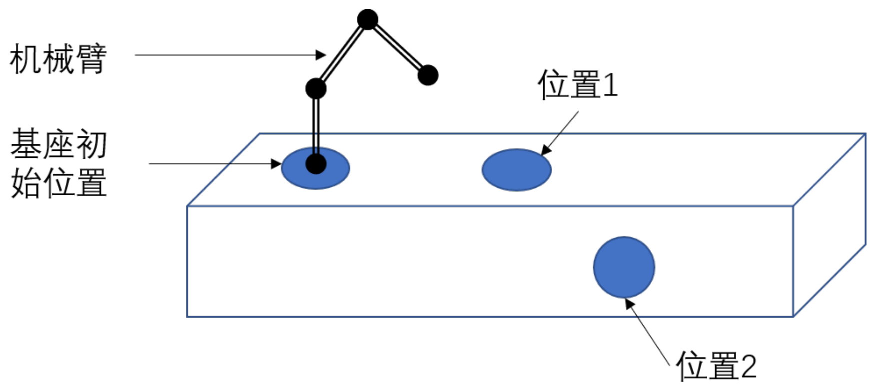

图 1: 空间站机械臂简化示意图

对于中国空间站机械臂的仿真模型，要求相应完成以下任务

1. 根据缩小版的机械臂结构和尺寸，利用DH方法正运动学建模；
2. 分析并给出该机械臂的逆运动学解析解；
3. 采用一种轨迹规划方法，控制该机械臂在仿真环境中在空间站上行走，至少行走2步，即从初始位置移动到位置1，再移动到位置2，运动时速度在合理范围内自定义，保证速度、加速度连续；
4. 假设机械臂末端于基座之间接触后可以通过电磁铁紧密吸合，借助仿真环境进行可视化演示。

2 列写DH参量表

*本次作业中，我们采用了**标准DH法**来进行正逆运动学的解算，而不是改进DH法*

机械臂机构图如下图所示

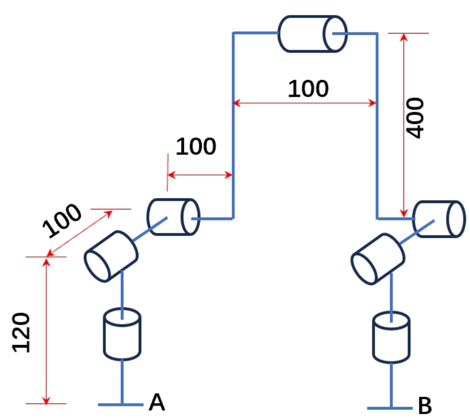

图 2: 机械臂机构图

根据机械臂的机构图，我们可以列写出仿真空间站机械臂的DH参量表如下(此处是以左端为基座的DH参量表，以右端为基座的DH参量表只需把其中的d_2和d_6取为相反数即可)

表 1: 标准DH法DH参量表

关节i	a_i	α_i	d_i	θ_i
1	0	$-\frac{\pi}{2}$	120	θ_1
2	0	$-\frac{\pi}{2}$	-100	θ_2
3	400	0	-150	θ_3
4	-400	0	-150	θ_4
5	0	$\frac{\pi}{2}$	0	θ_5
6	0	$\frac{\pi}{2}$	100	θ_6
7	0	0	120	θ_7

由DH参量表，根据标准DH法即可得到各相邻关节坐标系之间的齐次变换矩阵，其表达式为

$$
{}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

从基座到末端的总齐次变换矩阵由各相邻坐标系的其次末端矩阵相乘得到

$$
{}^{0}_{7}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T\,{}^{5}_{6}T\,{}^{6}_{7}T
$$

3 逆运动学求解推导

下面，我们对于按标准DH法建立的齐次变换矩阵进行各关节角的逆运动学求解。

设机械臂末端位姿为

$$
{}^{0}_{7}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T\,{}^{5}_{6}T\,{}^{6}_{7}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & x \\ r_{21} & r_{22} & r_{23} & y \\ r_{31} & r_{32} & r_{33} & z \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

从齐次变换矩阵 ${}^{0}_{7}T$, 可得末端执行器的位置 $[x, y, z]^T$, 并计算RPY角

$$
[r, p, y]^T = \begin{bmatrix} \operatorname{atan2}(-r_{23}, r_{33}) \\ \operatorname{asin}(r_{13}) \\ \operatorname{atan2}(-r_{12}, r_{11}) \end{bmatrix}
$$

当基座为 L_base 时,求解过程如下:

对位姿变换矩阵作变换得

$$
\left({}^{1}_{2}T\right)^{-1}\left({}^{0}_{1}T\right)^{-1}{}^{0}_{7}T = {}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T\,{}^{5}_{6}T\,{}^{6}_{7}T = {}^{2}_{7}T \tag{1}
$$

以下推导中，我们将采用一些简写的符号，如

- ${}^2_7T(3,4)$表示${}^2_7T$的第3行第4列元素
- s_1表示$\sin\theta_1$
- s_{12}表示$\sin(\theta_1+\theta_2)$

将DH参数代入各齐次变换矩阵，计算(1), 我们可以获得如下等式:

$$ {}^2_7T(3,3) = -s_6 = r_{13}c_2s_1 - r_{23}c_1c_2 - r_{33}s_2 \tag{2} $$

$$ {}^2_7T(3,4) = -120s_6 - 300 = (120-z)s_2 + (xs_1 - yc_1)\,c_2 \tag{3} $$

联立方程(2)(3)，并根据三角函数恒等式$\sin^2 x + \cos^2 x = 1$，即可解得s_1, c_1, s_1, c_2，于是可以得到

$$ \theta_6 = -\arcsin({}^2_7T(3,3))\text{或}\pi + \arcsin({}^2_7T(3,3)) $$

$$ \theta_1 = \operatorname{atan2}(s_1, c_1) $$

$$ \theta_2 = \operatorname{atan2}(s_2, c_2) $$

继续计算(1)中的元素，有

$$ {}^2_7T(3,1) = -c_6c_7 = r_{11}c_2s_1 - r_{21}c_1c_2 - r_{31}s_2 \tag{4} $$

$$ {}^2_7T(3,2) = c_6s_7 = r_{12}c_2s_1 - r_{22}c_1c_2 - r_{32}s_2 \tag{5} $$

由于θ_6前面已解出，所以此时c_6是已知量，则有

$$ \theta_7 = \operatorname{atan2}\left(\frac{{}^2_7T(3,2)}{c_6},\ -\frac{{}^2_7T(3,1)}{c_6}\right) $$

继续计算(1)，得

$$ {}^2_7T(1,4) = 400c_3 - 400c_{34} - 120c_6c_{345} + 100s_{345} = s_1s_2x - c_1s_2y + c_2z - 120c_2 \tag{6} $$

$$ {}^2_7T(2,4) = 400s_3 - 400s_{34} - 120c_6s_{345} - 100c_{345} = -c_1x - s_1y - 100 \tag{7} $$

$$ {}^2_7T(2,1) = -s_7c_{345} + c_7s_6s_{345} = -c_1r_{11} - s_1r_{21} \tag{8} $$

$$ {}^2_7T(2,2) = -c_7c_{345} - s_7s_6s_{345} = -c_1r_{12} - s_1r_{22} \tag{9} $$

由于前面已经解出θ_1和θ_7，所以联立(8)(9)即可解得s_{345}和c_{345}的值。此时再将s_{345}和c_{345}的值代入方程(6)(7)即可解出s_3, c_3, s_{34}, c_{34}的值。

此时，$s_3, c_3, s_{34}, c_{34}, s_{345}, c_{345}$的值均已被求出，于是有

$$ \theta_3 = \operatorname{atan2}(s_3, c_3) $$

$$ \theta_4 = \operatorname{atan2}(s_{34}, c_{34}) - \theta_3 $$

$$ \theta_5 = \operatorname{atan2}(s_{345}, c_{345}) - \theta_3 - \theta_4 $$

至此，关节角$\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7$均已被解出，逆运动学求解完成。

以上是以左端为基座的逆运动学求解过程，以右端为基座时，DH参量表中的d_2和d_6变为相反数，再次将DH参量代入各齐次变换矩阵中求解方程(1)即可。除参数变化外，求解思路和过程与上述求解过程完全一致，故此处不再赘述。

进行逆运动学解算时，一共有8组逆运动学解，此时比较并选取使得关节角改变量最小的解作为最终的逆运动学解即可。

4 五次多项式规划

为了保证速度、加速度的连续性，我们采用分段五次多项式进行关节角的轨迹规划。

4.1 分段五次多项式规划设计

为了实现更优的机械臂轨迹控制，我们给出若干个中间点，利用前面求出的逆运动学解求出各中间点位姿对应的关节角向量$q_1, q_2, ..., q_n$，并指定通过每个中间点的时间$t_1, t_2, ..., t_n$。

在进行轨迹规划我们先按运动学规律，指定机械臂通过每个中间点时各关节角的角速度和角加速度如下

$$v_i = \frac{q_{i+1} - q_{i-1}}{t_{i+1} - t_{i-1}} \qquad a_i = \frac{v_{i+1} - v_{i-1}}{t_{i+1} - t_{i-1}}$$

即使用一段区间内的平均速度和平均加速度来指定在中间点处的角速度和角加速度[1]。

而机械臂运动时，应当从静止开始启动，完成移动任务后到静止停止。因此，特别地，起始点和结束点处的关节角速度、加速度均指定为0

通过选取$n-1$个中间点，我们就将从起始点运动到结束点的关节空间轨迹分成了n段，对每一段均采用五次多项式规划，分段求解系数即可。

同时，通过指定起始点、结束点、各中间点的关节角向量、速度向量、加速度向量，我们就对分段五次多项式轨迹规划中的每一段五次多项式的端点处的函数值、导数值、二阶导值做出了约束，只需根据这些约束求解出每一段中五次多项式的各个系数，就完成了关节空间的分段五次多项式轨迹规划。

由于每一段内都是五次多项式轨迹，所以每一段内的关节角、速度、角速度都是连续的。而机械臂关节角轨迹需要在指定时间以指定角度、速度、角速度通过中间点，这就使得各段五次多项式之间的速度、角速度也是连续的，从而保证在整个运动过程中关节角、速度、角速度都是连续的。

4.2 多项式系数求解

对每一个关节的关节角，我们对分段五次多项式轨迹规划的每一段进行分析。

对每一段而言，其五次多项式表示为

$$\theta = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

[1]这里使用符号v, a是为方便表述，事实上指代的是关节角的角速度ω和角加速度α

对应的角速度v和角加速度a分别为

$$v = 5a_5t^4 + 4a_4t^3 + 3a_3t^2 + 2a_2t + a_1$$

$$a = 20a_5t^3 + 12a_4t^2 + 6a_3t + 2a_2$$

其满足初始角度θ_s、速度v_s、加速度a_s约束和最终角度θ_e、速度v_e、加速度a_e约束。(其中下标s和e分别表示start和end)

根据这些对角度约束，我们可以对每一段五次多项式列出如下方程

$$\begin{cases}
a_5t_s^5 + a_4t_s^4 + a_3t_s^3 + a_2t_s^2 + a_1t_s + a_0 = \theta_s & \text{(起始角度约束)} \\
5a_5t_s^4 + 4a_4t_s^3 + 3a_3t_s^2 + 2a_2t_s + a_1 = v_s & \text{(起始速度约束)} \\
20a_5t_s^3 + 12a_4t_s^2 + 6a_3t_s + 2a_2 = a_s & \text{(起始加速度约束)} \\
a_5t_e^5 + a_4t_e^4 + a_3t_e^3 + a_2t_e^2 + a_1t_e + a_0 = \theta_e & \text{(最终角度约束)} \\
5a_5t_e^4 + 4a_4t_e^3 + 3a_3t_e^2 + 2a_2t_e + a_1 = v_e & \text{(最终速度约束)} \\
20a_5t_e^3 + 12a_4t_e^2 + 6a_3t_e + 2a_2 = a_e & \text{(最终加速度约束)}
\end{cases}$$

通过求解以上方程组，即得这一段五次多项式的系数$a_5, a_4, a_3, a_2, a_1, a_0$；只需对每个关节角的分段五次多项式规划的每一段都进行这样的求解，就完成了了整个运动过程的分段五次多项式轨迹规划的求解。

4.3 代码实现

首先，计算各中间点的速度和加速度，对应的代码如下

```
def GetPointsVA(self):
    self.V = []
    self.A = []
    for i in range(self.N):
        if i == 0 or i == self.N - 1: # 起终点速度为0*
            v = np.array([0, 0, 0, 0, 0, 0, 0])
        else:
            v = (self.Q[i+1] - self.Q[i-1]) / (self.T[i+1] - self.T[i-1])
        self.V.append(v)
    for i in range(self.N):
        if i == 0 or i == self.N - 1: # 起终点加速度为0*
            a = np.array([0, 0, 0, 0, 0, 0, 0])
        else:
            a = (self.V[i+1] - self.V[i-1]) / (self.T[i+1] - self.T[i-1])
        self.A.append(a)
```

此后，即可按上面列写出的方程对每一段五次多项式的系数进行求解。为此，我们编写了下面一段函数，用于求解从第k个中间点到第$k+1$个中间点这一段五次多项式的系数。只要取$k = 0, 1, 2, \ldots, n-1$，调用此函数，即可求解全部n段的多项式系数，完成轨迹规划。(其中，第0个和第n个中间点分别代值机械臂的出发点和最终到达点)

```
def QuinticPoly(self, k):
    Ts = self.T[k]
    Te = self.T[k+1]
```

```
        Qs = self.Q[k]
        Qe = self.Q[k+1]
        Vs = self.V[k]
        Ve = self.V[k+1]
        As = self.A[k]
        Ae = self.A[k+1]
        A = np.array([
            [ Ts**5, Ts**4, Ts**3, Ts**2, Ts, 1], #q
            [ 5 * Ts**4, 4 * Ts**3, 3 * Ts**2, 2 * Ts, 1, 0], #v
            [20 * Ts**3, 12 * Ts**2, 6 * Ts, 2, 0, 0], #a
            [ Te**5, Te**4, Te**3, Te**2, Te, 1], #q
            [ 5 * Te**4, 4 * Te**3, 3 * Te**2, 2 * Te, 1, 0], #v
            [20 * Te**3, 12 * Te**2, 6 * Te, 2, 0, 0], #a
        ])
        b = np.array([Qs, Vs, As, Qe, Ve, Ae])
        self.Coefficient.append(np.linalg.solve(A, b))
```

完成分段五次多项式轨迹规划后，对于机械臂运动过程中的每一个时刻t，只需要找到t属于那一段五次多项式，利用对应段系数代入即可得到该时刻对应的关节角，对应代码如下

```
    def GetCurvePosition(self, t):
        for i in range(self.N-1):
            if t < self.T[i+1]:
                ans = []
                for j in range(7):
                    ans.append(np.dot(np.array(self.Coefficient)[i, :, j], [t**5, t**4, t**3, t**2, t, 1]))
                return ans
        return self.Q[self.N - 1]
```

得到该时刻的关节角后，只需要调用CoppeliaSim提供的sim.setJointPosition函数API即可。

5 实验结果与分析

由于报告中无法展示视频，故此处主要展示机械臂运动过程中的重要截图。对机械臂的完整运行过程请查看提交的机械臂运动视频，或运行提交的源代码文件并在*CoppeliaSim*中查看效果。

5.1 机械臂运行过程截图

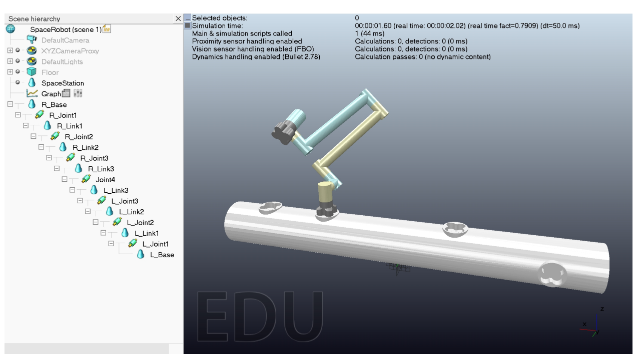

图 3: 机械臂运行过程截图

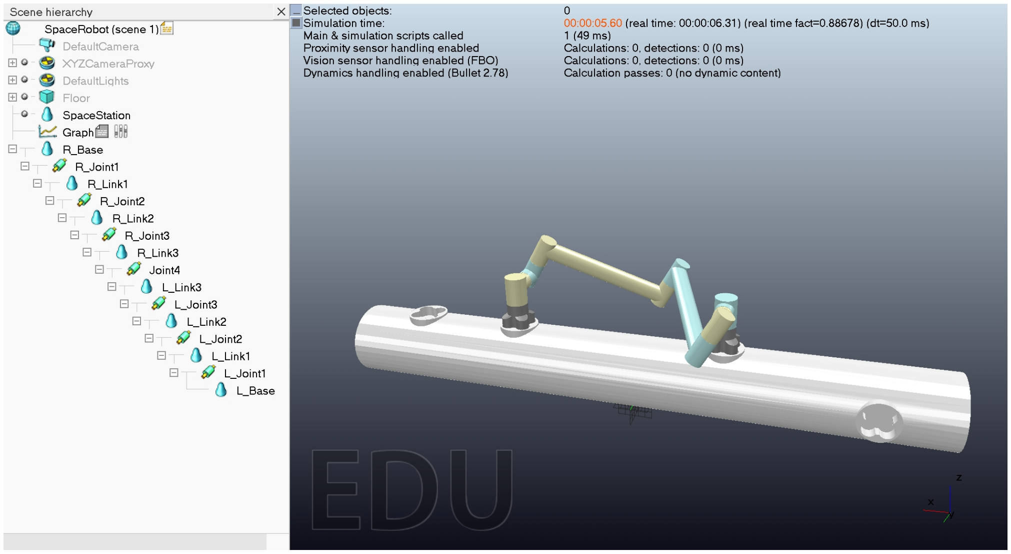

图 4: 机械臂到达位置1，完成第一步

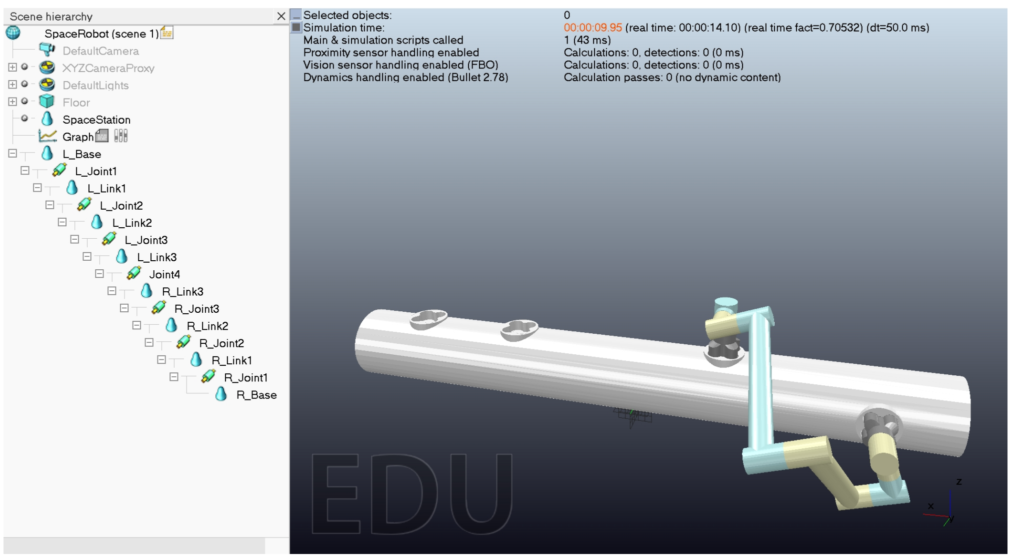

图 5: 机械臂到达位置2，完成运动

从上述截图中可以看出，机械臂可以正常进行运动，并按题目要求完成两步行走(即从初始位置移动到位置1，在移动到位置2)

机械臂整体运动正常，完成题目要求。

5.2 角度连续性展示

我们绘制出机械臂运动过程中，各关节的角速度随时间的变化情况如下

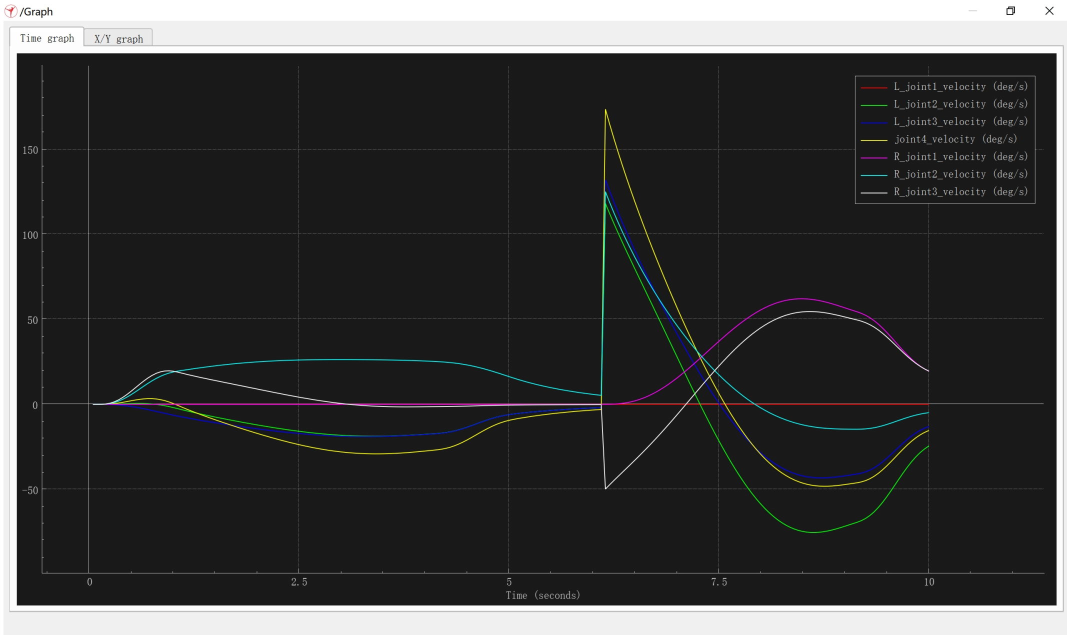

图 6: 关节角速度随时间的变化曲线

图中关节角速度在$t = 6$s时发生突变是因为在这一时刻机械臂到达位置1，此时切换了基座base和各关节的父子关系，使得原本的左右关节互换了，才显得关节角和对应的角速度发生了“突变”。事实上，实际的关节角并没有发生变化，关节角的角度、角速度、角加速度仍然是连续的。

从关节角速度随时间的变化曲线中可以看出，机械臂运动过程中，各关节的角速度是连续变化的。进一步，我们可以看出，关节角速度曲线是光滑的，说明其导数(即关节角加速度)也是连续变化的。

即**在整个机械臂的运动过程中，机械臂的关节角速度、角加速度是连续的**，符合题目要求。