

实验一

给定 Puma560 机械臂构型，使用正运动学在关节空间随机采样得到机械臂的工作空间，具体 Matlab 代码如下所示：

```
1. %%Define of puma
2. deg = pi/180;
3. L(1) = Revolute('d', 0, 'a', 0, 'alpha', pi/2, ...
4.     'I', [0, 0.35, 0, 0, 0, 0], ...
5.     'r', [0, 0, 0], ...
6.     'm', 0, ...
7.     'Jm', 200e-6, ...
8.     'G', -62.6111, ...
9.     'B', 1.48e-3, ...
10.    'Tc', [0.395 -0.435], ...
11.    'qlim', [-90 90]*deg );
12. L(2) = Revolute('d', 0, 'a', 100, 'alpha', 0, ...
13.     'I', [0.13, 0.524, 0.539, 0, 0, 0], ...
14.     'r', [-0.3638, 0.006, 0.2275], ...
15.     'm', 17.4, ...
16.     'Jm', 200e-6, ...
17.     'G', 107.815, ...
18.     'B', .817e-3, ...
19.     'Tc', [0.126 -0.071], ...
20.     'qlim', [-90 90]*deg );
21. L(3) = Revolute('d', 20, 'a', 10, 'alpha', -pi/2, ...
22.     'I', [0.066, 0.086, 0.0125, 0, 0, 0], ...
23.     'r', [-0.0203, -0.0141, 0.070], ...
24.     'm', 4.8, ...
25.     'Jm', 200e-6, ...
26.     'G', -53.7063, ...
27.     'B', 1.38e-3, ...
28.     'Tc', [0.132, -0.105], ...
29.     'qlim', [-90 90]*deg );
30. L(4) = Revolute('d', 100, 'a', 0, 'alpha', pi/2, ...
31.     'I', [1.8e-3, 1.3e-3, 1.8e-3, 0, 0, 0], ...
32.     'r', [0, 0.019, 0], ...
33.     'm', 0.82, ...
34.     'Jm', 33e-6, ...
35.     'G', 76.0364, ...
```

```
36. 'B', 71.2e-6, ...
37. 'Tc', [11.2e-3, -16.9e-3], ...
38. 'qlim', [-90 90]*deg);
39. L(5) = Revolute('d', 0, 'a', 0, 'alpha', -pi/2, ...
40. 'I', [0.3e-3, 0.4e-3, 0.3e-3, 0, 0, 0], ...
41. 'r', [0, 0, 0], ...
42. 'm', 0.34, ...
43. 'Jm', 33e-6, ...
44. 'G', 71.923, ...
45. 'B', 82.6e-6, ...
46. 'Tc', [9.26e-3, -14.5e-3], ...
47. 'qlim', [-90 90]*deg );
48. L(6) = Revolute('d', 0, 'a', 0, 'alpha', 0, ...
49. 'I', [0.15e-3, 0.15e-3, 0.04e-3, 0, 0, 0], ...
50. 'r', [0, 0, 0.032], ...
51. 'm', 0.09, ...
52. 'Jm', 33e-6, ...
53. 'G', 76.686, ...
54. 'B', 36.7e-6, ...
55. 'Tc', [3.96e-3, -10.5e-3], ...
56. 'qlim', [-90 90]*deg );
57.
58. p560 = SerialLink(L, 'name', 'Puma 560', ...
59. 'manufacturer', 'Unimation', 'ikine', 'puma', 'comment', 'viscous fricti
on; params of 8/95');
60. p560.model3d = 'UNIMATE/puma';
61. %%solve the reachable workspace
62. qq = (rand(1000,6)*180 - 90*ones(1000,6))*pi/180;
63. tt = p560.fkine(qq);
64. for i = 1:1000
65.     dat=tt(:, :, i);
66.     x(i) =dat(1,4);
67.     y(i) =dat(2,4);
68.     z(i) =dat(3,4);
69. end
70. plot3(x,y,z, 'bd', 'LineWidth', 0.5, 'MarkerSize', 2, 'MarkerEdgeColor', 'm', 'Marke
rFaceColor', 'r');
71. axis equal;
72. hold on
73. p560.plot([0 0 0 0 0 0])
74. set(gcf, 'position', [0,0,1024,768]);
```

在Matlab中求解机器人的可达工作空间可以采用随机采样方法。通过产生中 $(-90^\circ, +90^\circ)$ 范围内的1000个随机关节参数并计算相应的正向，并利用fkin()进行运动学的计算，可以求解末端执行器的位置。

利用可视化工具，我们可以看到最终可达空间的不同视角散点图如图1-4所示。

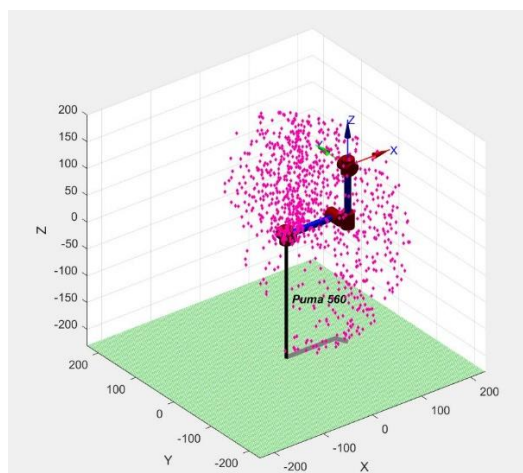


图 1 散点图

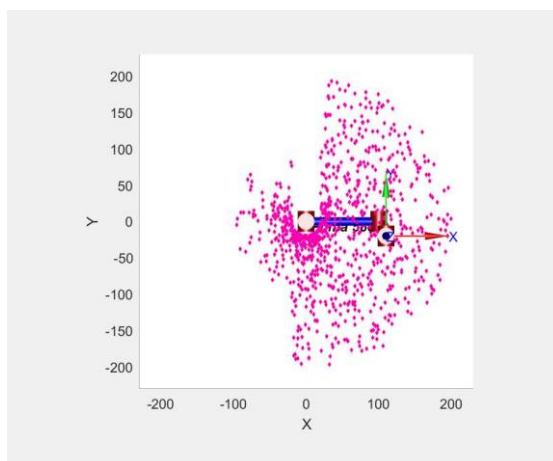


图 2 散点图 (X-Y)

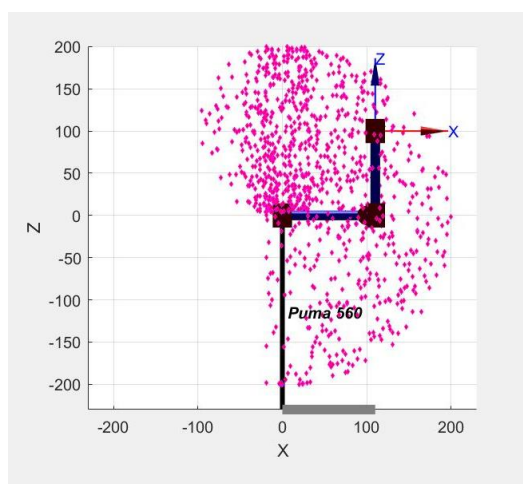


图 3 散点图 (X-Z)

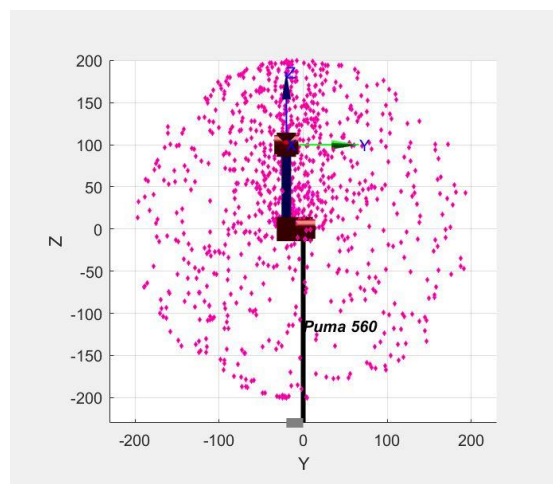


图 4 散点图 (Y-Z)

实验二

工作空间的 $\text{origin}=[100,0,50]$ (中心) 位置处有一块 $\text{size}=[200,30,150]$ 的障碍物, 使用逆运动学规划一条从起点 $\text{pini}=[100\ 100\ 10]'$ 到终点 $\text{pend}=[100\ -100\ 10]'$ 的路径。画出机械臂从起点到终点的工作空间路径及关节角度变化曲线。

具体 Matlab 代码如下:

```
1. L(1) = Revolute('d', 0, 'a', 0, 'alpha', pi/2, ...
2.     'I', [0, 0.35, 0, 0, 0, 0], ...
3.     'r', [0, 0, 0], ...
4.     'm', 0, ...
5.     'Jm', 200e-6, ...
6.     'G', -62.6111, ...
7.     'B', 1.48e-3, ...
8.     'Tc', [0.395 -0.435], ...
9.     'qlim', [-90 90]*deg );
10.
11. L(2) = Revolute('d', 0, 'a', 100, 'alpha', 0, ...
12.     'I', [0.13, 0.524, 0.539, 0, 0, 0], ...
13.     'r', [-0.3638, 0.006, 0.2275], ...
14.     'm', 17.4, ...
15.     'Jm', 200e-6, ...
16.     'G', 107.815, ...
17.     'B', .817e-3, ...
18.     'Tc', [0.126 -0.071], ...
19.     'qlim', [-90 90]*deg );
20.
21. L(3) = Revolute('d', 20, 'a', 10, 'alpha', -pi/2, ...
22.     'I', [0.066, 0.086, 0.0125, 0, 0, 0], ...
23.     'r', [-0.0203, -0.0141, 0.070], ...
24.     'm', 4.8, ...
25.     'Jm', 200e-6, ...
26.     'G', -53.7063, ...
27.     'B', 1.38e-3, ...
28.     'Tc', [0.132, -0.105], ...
29.     'qlim', [-90 90]*deg );
30.
31. L(4) = Revolute('d', 100, 'a', 0, 'alpha', pi/2, ...
```

```
32. 'I', [1.8e-3, 1.3e-3, 1.8e-3, 0, 0, 0], ...
33. 'r', [0, 0.019, 0], ...
34. 'm', 0.82, ...
35. 'Jm', 33e-6, ...
36. 'G', 76.0364, ...
37. 'B', 71.2e-6, ...
38. 'Tc', [11.2e-3, -16.9e-3], ...
39. 'qlim', [-90 90]*deg);
40.
41. L(5) = Revolute('d', 0, 'a', 0, 'alpha', -pi/2, ...
42. 'I', [0.3e-3, 0.4e-3, 0.3e-3, 0, 0, 0], ...
43. 'r', [0, 0, 0], ...
44. 'm', 0.34, ...
45. 'Jm', 33e-6, ...
46. 'G', 71.923, ...
47. 'B', 82.6e-6, ...
48. 'Tc', [9.26e-3, -14.5e-3], ...
49. 'qlim', [-90 90]*deg );
50.
51.
52. L(6) = Revolute('d', 0, 'a', 0, 'alpha', 0, ...
53. 'I', [0.15e-3, 0.15e-3, 0.04e-3, 0, 0, 0], ...
54. 'r', [0, 0, 0.032], ...
55. 'm', 0.09, ...
56. 'Jm', 33e-6, ...
57. 'G', 76.686, ...
58. 'B', 36.7e-6, ...
59. 'Tc', [3.96e-3, -10.5e-3], ...
60. 'qlim', [-90 90]*deg );
61.
62. p560 = SerialLink(L, 'name', 'Puma 560', ...
63. 'manufacturer', 'Unimation', 'ikine', 'puma', 'comment', 'viscous fricti
on; params of 8/95');
64. p560.model3d = 'UNIMATE/puma';
65. puma560.base = transl(100,0,80);
66. T0 = transl(100,100,10);
67. q0 = puma560.ikine6s(T0,'1');
68. T1 = transl(100,20,-30);
69. q1 = puma560.ikine6s(T1,'1');
70. T2 = transl(100,-20,-30);
71. q2 = puma560.ikine6s(T2,'1');
72. T3 = transl(100,-100,10);
73. q3 = puma560.ikine6s(T3,'1');
74. t = [0:0.05:2];
```

```
75. Q1 = jtraj(q0,q1,t);
76. Q2 = jtraj(q1,q2,t);
77. Q3 = jtraj(q2,q3,t);
78. Q = [Q1;Q2;Q3];
79. t1=[0:0.05:6.1];
80. p560.plot(Q)
81. figure(1)
82. tt = puma560.fkine(Q);
83. qq =transl(tt);
84. plot3(qq(:,1),qq(:,2),qq(:,3),'r','LineWidth',1,'MarkerSize',2);
85. hold on;
86. plot3([0,0],[-15,15],[125,125],'b')
87. plot3([0,0],[-15,15],[-25,-25],'b')
88. plot3([0,0],[-15,-15],[125,-25],'b')
89. plot3([0,0],[15,15],[125,-25],'b')
90. plot3([0,200],[-15,-15],[125,125],'b')
91. plot3([0,200],[15,15],[125,125],'b')
92. plot3([0,200],[-15,-15],[-25,-25],'b')
93. plot3([0,200],[15,15],[-25,-25],'b')
94. plot3([200,200],[-15,-15],[125,-25],'b')
95. plot3([200,200],[15,15],[125,-25],'b')
96. plot3([200,200],[-15,15],[125,125],'b')
97. plot3([200,200],[-15,15],[-25,-25],'b')
98. xlabel('x');ylabel('y');zlabel('z');
99. camlight
100. grid on;
101.
102. figure(2)
103. for i=1:6
104.     subplot(2,3,i)
105.     plot(t1, Q(:,i),'r')
106.     grid on;
107. end
```

在Matlab中，使用p560.base将puma机器人的base基数设为(100,0,80),使用jtraj来计算机器人的运动轨迹，并使用逆运动学来计算关节的相关参数，最终进行了各求解过程的可视化。

利用可视化工具，我们可以看到实验部分结果图5-7如下所示。

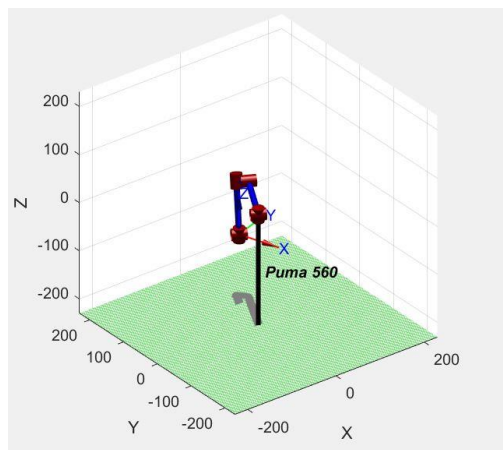


图 5 puma 轨迹（动图无法插入）

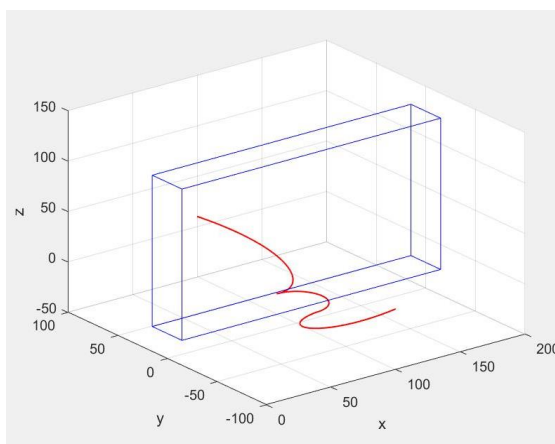


图 6 完整轨迹

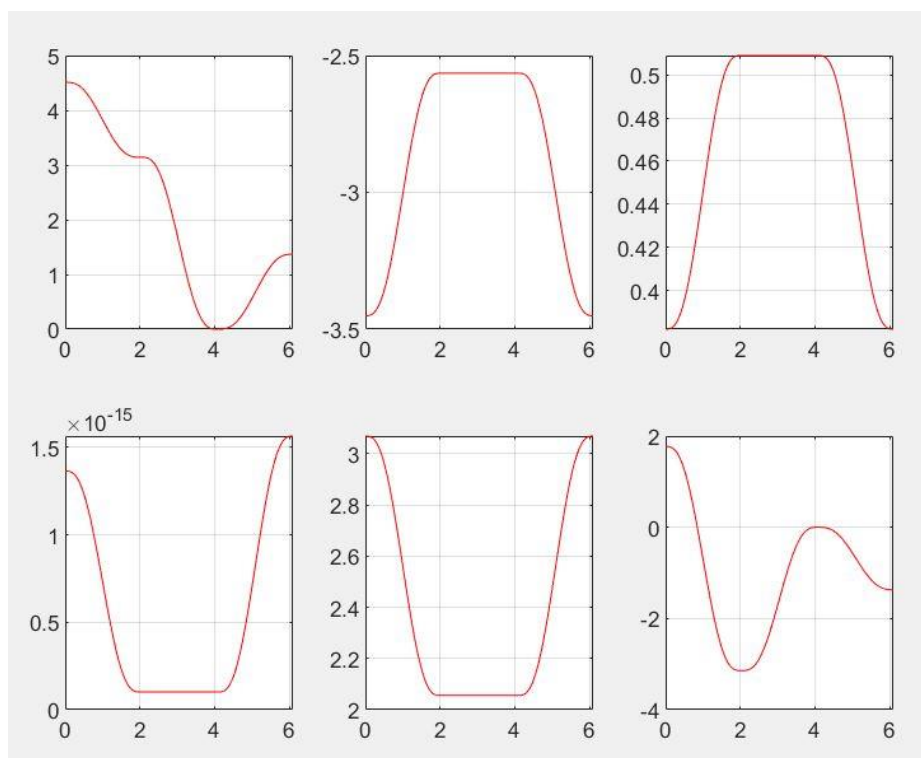


图 7 关节角度变化曲线