

机器人技术基础课程作业



**课 程** 机器人技术基础

**姓 名** 陈志平

**学 号** U201410626

**班 级** 机械1402班

1. SCARA机器人建模
2. 运动学方程（按照图5-6形式建立方程）

SCARA机器人有三个旋转关节，和一个移动关节，建立其D-H表：如下

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 1 | 0 | 0 | 0 |  |
| 2 | 0 |  | 0 |  |
| 3 | 0 |  | 0 |  |
| 4 | 0 | 0 | + | 0 |

由齐次变换知识可得，SCARA各连杆的变换矩阵为：

其中表示cos，表示sin。由上述各连杆变换矩阵之积，可得运动学方程：

1. 运动学逆解

令SCARA机器人末端执行器期望位姿为：

由3.4可得，

所以

由1.4和2.4可得

且有

可得， (式1)

而

所以

将代入式1中，可得

(式2)

故，

1. 雅可比矩阵

SCARA机器人的特点是各个关节的运动旋量的方向都是固定的，各个旋量轴的线矢量上的一定是的函数，取为

，，，

计算各个关节的运动旋量坐标，则得

1. 讨论其在参考形位处（图6-5）的奇异性、可操作性；

在参考形位处，杆一和杆二在同一条直线上，关节三只能沿着切线方向移动，故自由度降低，为奇异点，因为雅可比矩阵非满秩，所以可操作性为0.

1. 讨论其在参考形位处（图6-5）的末端操作刚度。

由于此为四自由度机器人，建立其柔度矩阵过于复杂，且都为参数值，采用matlab计算其特征值，最终值过长，如下：

同上

其中k4为第四关节的刚度，为Z轴方向，所以这个方向柔性较差，刚度较好，而其他方向较差。

1. 采用Matlab编写轨迹生成程序
2. 三阶多项式&五阶多项式

代码程序：

function y=TriF(varargin)

syms t;

B=[];

A=[]\*t;

if nargin == 5

tt=varargin{1};

for j=1:2

for i=0:3

if j==1

A(1,i+1)=0;

else

A(2,i+1)=t.^i;

end

end

end

A(1,1)=1;

for i=2:3

for j=0:3

if rem(i,2)==0

A(i+1,j+1)=0;

else

A(i+1,j+1)=diff(A(i-1,j+1));

end

end

end

A(3,2)=A(4,2);

for i=1:4

B(i,1)=varargin{i+1};

end

m=A(2,:);

n=subs(A\B,tt);

y=m\*n;

elseif nargin == 7

tt=varargin{1};

for j=1:2

for i=0:5

if j==5

A(1,i+1)=0;

else

A(2,i+1)=t.^i;

end

end

end

A(1,1)=1;

for i=2:5

for j=0:5

if rem(i,2)==0

A(i+1,j+1)=0;

else

A(i+1,j+1)=diff(A(i-1,j+1));

end

end

end

A(3,2)=A(4,2);

A(5,3)=A(6,3);

for i=1:6

B(i,1)=varargin{i+1};

end

m=A(2,:);

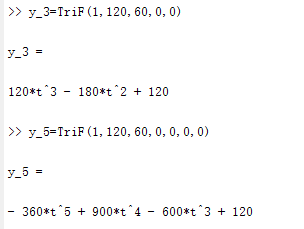
n=subs(A\B,tt);

y=m\*n;

end

end

结果展示：

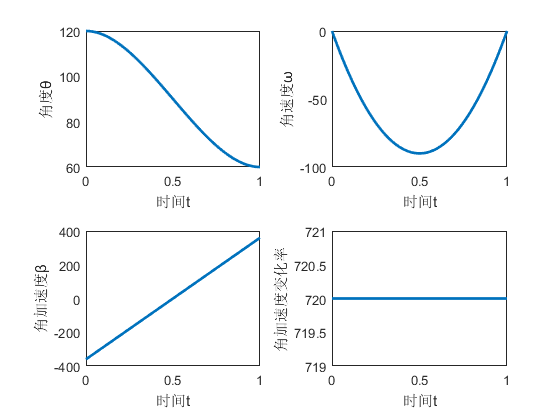


如上图，得出，三阶多项式为：

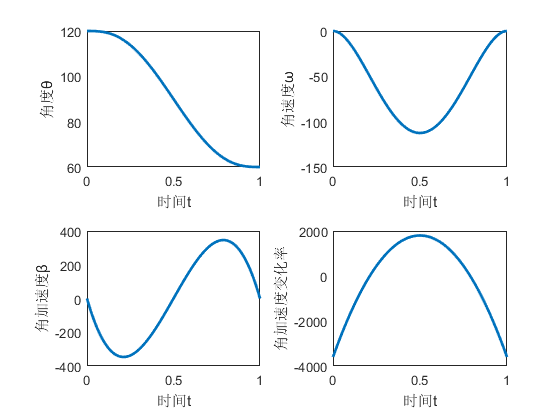
五阶多项式为：

绘图如下

三 阶



五 阶



结果讨论：

三阶和五阶拟合的直线相差不大，但五阶的平滑度较好

1. 两段带有中间点的三阶多项式

代码程序：

function y=BiF(varargin)

syms t;

A=[]\*t;

for i=0:3

A(1,i+1)=t.^i;

end

if nargin == 5

m=varargin{1};

if m==1

for i=1:4

B(1,i)=subs(A(1,i),0);

end

for i=1:4

B(2,i)=subs(A(1,i),1);

end

for i=1:4

B(3,i)=subs(diff(A(1,i)),0);

end

for i=1:4

B(4,i)=subs(diff(diff(A(1,i))),0);

end

end

if m==2

for i=1:4

B(1,i)=subs(A(1,i),1);

end

for i=1:4

B(2,i)=subs(A(1,i),2);

end

for i=1:4

B(3,i)=subs(diff(A(1,i)),2);

end

for i=1:4

B(4,i)=subs(diff(diff(A(1,i))),2);

end

end

C=[];

for i=1:4

C(i,1)=varargin{i+1};

end

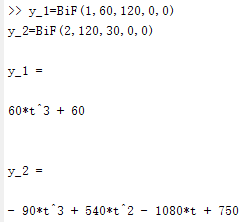
n=B\C;

y=A\*n;

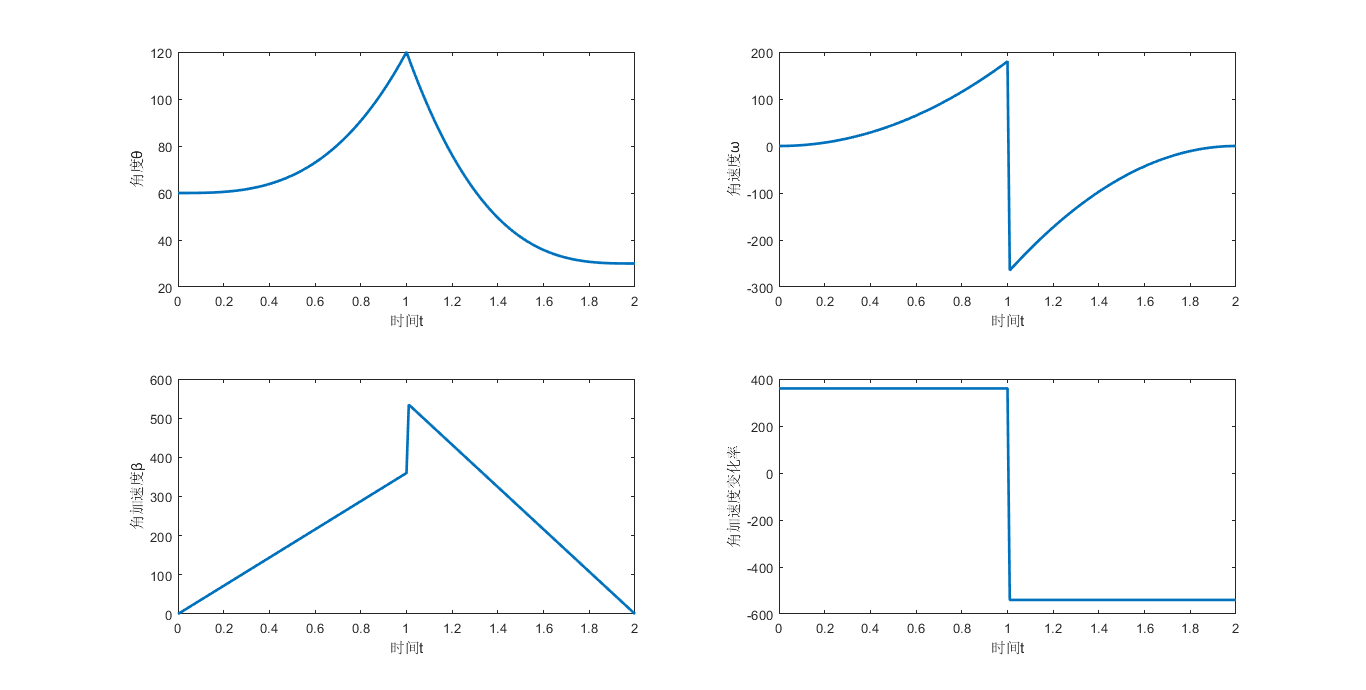
end

end

结果展示：



结果讨论：



题目中给出的条件为，起点和终点的角速度和角加速度为0，条件多余，造成最终两段曲线速度和加速度不能平缓结合，不能实现。

题目有问题，应该去掉起点和终点角加速度为0的条件。

改进后：

代码为：

ttt=[1 0 0 0 0 0 0 0;

1 1 1 1 0 0 0 0;

0 0 0 0 1 0 0 0;

0 0 0 0 1 1 1 1;

0 1 0 0 0 0 0 0;

0 0 0 0 0 1 2 3;

0 1 2 3 0 -1 0 0;

0 0 2 6 0 0 -2 0];

y=[60;120;120;30;0;0;0;0];

syms t;

A=[]\*t;

for i=0:3

A(1,i+1)=t.^i;

end

B=ttt\y;

B1=B(1:4,:);

B2=B(5:8,:);

y=A\*B1;

x=A\*B2;

yy=diff(y);xx=diff(x);

yyy=diff(yy);xxx=diff(xx);

yyyy=diff(yyy);xxxx=diff(xxx);

subplot(2,2,1)

tt=0:0.01:1;

m1=subs(y,tt);

tt=1.01:0.01:2;

m2=subs(x,tt-1);

tt=0:0.01:2;

m=[m1,m2];

plot(tt,m,'LineWidth',2);

xlabel('Ê±¼ät');

ylabel('½Ç¶È¦È');

subplot(2,2,2)

tt=0:0.01:1;

m1=subs(yy,tt);

tt=1.01:0.01:2;

m2=subs(xx,tt-1);

tt=0:0.01:2;

m=[m1,m2];

plot(tt,m,'LineWidth',2);

xlabel('Ê±¼ät');

ylabel('½ÇËÙ¶È¦Ø');

subplot(2,2,3)

tt=0:0.01:1;

m1=subs(yyy,tt);

tt=1.01:0.01:2;

m2=subs(xxx,tt-1);

tt=0:0.01:2;

m=[m1,m2];

plot(tt,m,'LineWidth',2);

xlabel('Ê±¼ät');

ylabel('½Ç¼ÓËÙ¶È¦Â');

subplot(2,2,4)

tt=0:0.01:1;

m1=subs(yyyy,tt);

tt=1.01:0.01:2;

m2=subs(xxxx,tt-1);

tt=0:0.01:2;

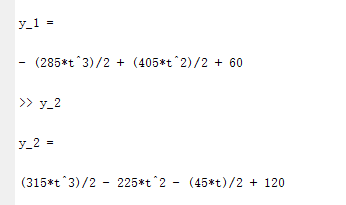
m=[m1,m2];

plot(tt,m,'LineWidth',2);

xlabel('Ê±¼ät');

ylabel('½Ç¼ÓËÙ¶È±ä»¯ÂÊ');

结果展示：

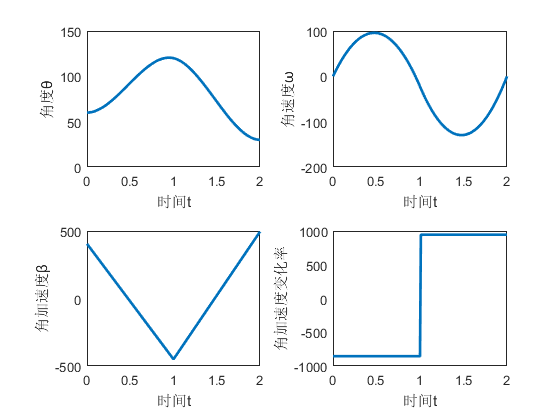


其中第二段的时间段为[0,1],绘图时右移动一个单位

结果讨论：

去掉起点和终点加速度为0的条件后，可以实现题目要求，但在角加速度变化率上，跳跃较大。平滑性较好。

1. 用工具箱检验



代码程序：

clear all;

Q0=120;QF=60;

[Q,QD,QDD] = jtraj(Q0, QF, 1001, 0, 0);

subplot(2,2,1)

tt=0:0.001:1;

plot(tt,Q,'LineWidth',2);

subplot(2,2,2)

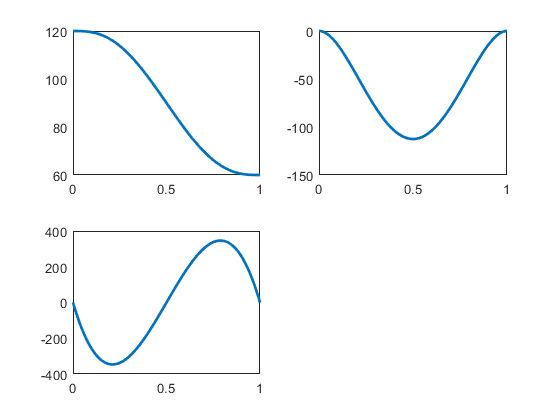
tt=0:0.001:1;

plot(tt,QD,'LineWidth',2);

subplot(2,2,3)

tt=0:0.001:1;

plot(tt,QDD,'LineWidth',2);



结果展示：

结果讨论：

与五阶比较，完全一致，所以，程序生成是正确的。