1. 针对如下一个两连杆机械臂系统，采用鲁棒逆动力学或自适应控制方法设计机械臂控制系统:

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix}+\begin{bmatrix} -h\dot{q}_2 & -h(\dot{q}_1+\dot{q}_2) \\ h\dot{q}_1 & 0 \end{bmatrix}\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}=\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}+\begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix}$$

$$\begin{aligned}
M_{11} &= p_1 + 2p_3\cos(q_2) + 2p_4\sin(q_2) \\
M_{12} &= M_{21} = p_2 + p_3\cos(q_2) + p_4\sin(q_2) \\
M_{22} &= p_2 \\
h &= p_3\sin(q_2) - p_4\cos(q_2) \\
p_1 &= I_1 + m_1 l_{c1}^2 + I_e + m_e l_{ce}^2 + m_e l_1^2 \\
p_2 &= I_e + m_e l_{ce}^2 \\
p_3 &= m_e l_1 l_{ce}\cos(\sigma_e) \\
p_4 &= m_e l_1 l_{ce}\sin(\sigma_e)
\end{aligned}$$

仿真参数：

$$\begin{aligned}
& m_1 = 1, m_e = 2, \ell_1 = 1, \ell_{ce} = 0.6, \ell_{c1} = 0.5 \\
& \sigma_e = 1/6\pi, I_1 = 0.12, I_e = 0.25 \\
& \omega_1 = 0.1\cos(0.2 - 0.01t) \\
& \omega_2 = 0.1\sin(0.2 - 0.01t) \\
& q_1(0) = 0.05\,rad \quad \dot{q}_1(0) = 0.01\,rad/s \\
& q_2(0) = -0.1\,rad \quad q_2(0) = 0.02\,rad/s
\end{aligned}$$

期望控制指令：

$$q_d = \left[\frac{1}{2}\pi, \frac{1}{2}\pi\right]$$

答：题目已给出需要设计的机械臂控制系统的动力学方程形式，并在给定的仿真参数下给出了期望控制指令即关节空间状态终值。从控制任务上分析，这是一种期望轨迹不随时间变化的轨迹跟踪控制。所以本次控制系统设计可采用鲁棒逆运动学进行。常见的平面肘形机械臂结构以及坐标如图 1 所示。

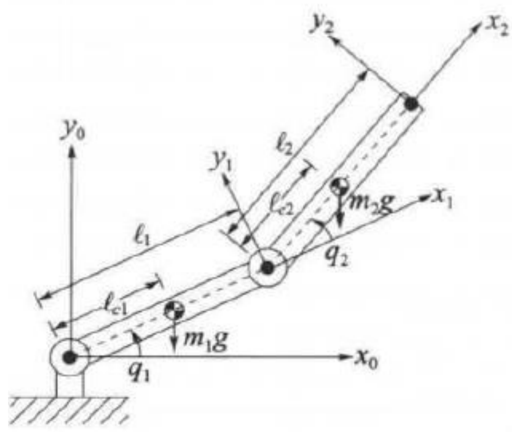

图 1

1) 仿真参数、模型参数解析

本题中的参数与图中存在略微差别，但无明显差别，具体的参数说明如下：

(1) 控制系统参数

将本题中的控制系统方程简写为：

$$W(q)\ddot{q}+H(q,\dot{q})\dot{q}=\tau+w \tag{1}$$

则其中参数分别为：

变量	变量解释
q	关节空间变量，弧度或角度
$\dot{q}$	关节空间变量一阶导，角速度
$\ddot{q}$	关节空间变量二阶导，加加速度
$W(q)$	惯性矩阵
$H(q,\dot{q})$	科里奥利矩阵
τ	关节控制点扭矩输入
w	扰动项

表 1

(2) 仿真参数

变量	变量解释
m	质量，标号 1 为杆 1，标号 e 为杆二，下同
l	杆长度
l_c	关节距离杆中心的长度
σ_e	这个参数没明白
I	转动惯量
w	扰动项
$q(t)$	关节变量随时间的变化，t=0 为初始状态
$\dot{q}(t)$	关节变量一阶倒随时间的变化，t=0 为初始状态
q^d	期望轨迹，或为常数，或为随时间变化的函数，本题为常数

表 2

2) 模型建立

(1) 运动学模型

正运动学为：

$$\begin{aligned} x_1 &= l_1 \cos q_1 \\ y_2 &= l_1 \sin q_1 \\ x_2 &= l_1 \cos q_1 + l_2 \cos(q_1 + q_2) \\ y_2 &= l_1 \sin q_1 + l_2 \sin(q_1 + q_2) \end{aligned} \tag{2}$$

逆运动学为：

$$\begin{aligned} \cos q_2 &= \frac{{x_2}^2 + {y_2}^2 - l_1^2 - l_2^2}{2 l_1 l_2} = C \\ \sin q_2 &= \pm\sqrt{1 - C^2} = D \\ q_2 &= \operatorname{atan} 2(D, C) \\ q_1 &= \operatorname{atan} 2(y, x) - \operatorname{atan} 2(l_2 \sin q_2, l_1 + l_2 \cos q_2) \end{aligned} \tag{3}$$

(2) 动力学模型

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} -h\dot{q}_2 & -h(\dot{q}_1 + \dot{q}_2) \\ h\dot{q}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} + \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} \tag{4}$$

3) 控制器设计

控制对象为式(4)所示，逆向动力学补偿以及控制器的形式应为：

$$\begin{aligned} \tau &= u + M(q)\ddot{q}_d + H(q,\dot{q})\dot{q}_d \\ u &= M(q)\ddot{e} + H(q,\dot{q})\dot{e} - w \end{aligned} \tag{5}$$

其中$e(t) = q(t) - q_d(t)$， u是辅助控制输入信号。

定义辅助信号为：

$$\eta = \dot{e} + \alpha e \tag{6}$$

其中$\alpha > 0$。

由此可将(4)变为

$$M(q)(e + \alpha\dot{e}) = M(q)\alpha\dot{e} - H(q,\dot{q})e + w + u \tag{7}$$

即

$$M(q)\dot{\eta} = -C(q,\dot{q})\eta + M(q)\alpha\dot{e} + H(q,\dot{q})\alpha e + w + u \tag{8}$$

取$W(q,\dot{q},e,\dot{e}) = M(q)\alpha\dot{e} + H(q,\dot{q})\alpha e$，则上式可以表示为

$$M(q)\dot{\eta} = -H(q,\dot{q})\eta + W + w + u \tag{9}$$

设存在正整数$\rho(e,\dot{e})$，使得对于任何的w,下式成立：

$$\|w\| \le p(e,\dot{e}) \tag{10}$$

则可以设反馈控制率为

$$u = -K\eta - W - v \tag{11}$$

其中

$$v = \frac{\eta\rho^2(e,\dot{e})}{\|\eta\|\rho(e,\dot{e}) + \varepsilon} \tag{12}$$

其中$\varepsilon > 0$，为给定的常数； $K = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}$， $k_1 > 0, k_2 > 0$为给定常数，那么我们可以

证明

$$\|e(t)\| \le A\|e(0)| e^{-\alpha t} + Be^{-\lambda t/3} + C, \quad \forall t \tag{13}$$

4) 稳定性证明

首先证明辅助信号的终值有界性。定义正定函数如下：

$$V(t,\eta) = \frac{1}{2}\eta^T M(q)\eta \tag{14}$$

根据机器人 $M(q)$ 的有界性可知，存在正数 $\lambda_1 > 0, \lambda_2 > 0$，满足

$$\lambda_1 \|\eta\|^2 \le V(t,\eta) \le \lambda_2 \|\eta\|^2, \quad \forall \eta \tag{15}$$

根据机器人的斜对称性，有

$$\eta^T \{\dot{M}(q) - 2C(q,\dot{q})\}\eta = 0, \ \forall \eta \tag{16}$$

则

$$\begin{aligned}
\dot{V} &= \eta^T M(q)\dot{\eta} + \frac{1}{2}\eta^T \dot{M}(q)\eta \\
&= \eta^T \left\{-H(q,\dot{q})\eta + W + w + u\right\} + \frac{1}{2}\eta^T \dot{M}(q)\eta \\
&= \eta^T \left\{u + W + w\right\} + \frac{1}{2}\eta^T \left\{\dot{M}(q) - 2H(q,\dot{q})\right\}\eta \\
&= \eta^T \left\{u + W + w\right\}
\end{aligned} \tag{17}$$

将式(11)带入上式可得

$$\dot{V} = -\eta^T v + \eta^T w - \mu^T K\eta \tag{18}$$

利用式(10)以及柯西不等式$|x^T y| \le \|x\| \bullet \|y\|$，得

$$\dot{V} \le -\eta^T v + \|\eta\| \cdot \|w\| - \eta^T K\eta = -\eta^T v + \|\eta\| \bullet \rho - \eta^T K\eta \tag{19}$$

将式(12)的表达式带入，并整理得到

$$\begin{aligned}
\dot{V} &\le -\eta^T K\eta - \eta^T \frac{\eta\rho^2}{\|\eta\|\rho + \varepsilon} + \|\eta\| \bullet \rho \\
&= -\eta^T K\eta + \frac{\|\eta\|\rho}{\|\eta\|\rho + \varepsilon} \bullet \varepsilon \le -\eta^T K\eta + \varepsilon \le -\lambda_3 \|\eta\|^2 + \varepsilon
\end{aligned} \tag{20}$$

其中λ_3表示一个给定的正数，$\lambda_3 \le \| K \|$。

根据式(16)和式(20)以及 Lasalle 定理，将η代替$\| x \|$，辅助信号η满足

$$\| \eta(t) \|^2 \le \frac{\lambda_2}{\lambda_2} \| n(0) \|^2 e^{-\lambda t} + \frac{\varepsilon}{\lambda_1 \lambda}(1 - e^{-\lambda t}) \tag{21}$$

由辅助信号的定义式(6)可得

$$\dot{e} + \alpha e - \eta = 0 \tag{22}$$

此为一阶齐次线性方程组，解得

$$e(t) = e(0)e^{-\alpha t} + \int_0^t e^{-\alpha(t-s)} \eta(s) ds \tag{23}$$

记误差的各个分量为e_1和e_2，则式(23)可以分解为

$$\begin{aligned} e_1(t) &= e_1(0)e^{-\alpha t} + \int_0^t e^{-\alpha(t-s)} \eta_1(s) ds \\ e_2(t) &= e_2(0)e^{-\alpha t} + \int_0^t e^{-\alpha(t-s)} \eta_2(s) ds \end{aligned} \tag{24}$$

其中$\eta^T = [\eta_1 \ \eta_2]$。

将上式两边取绝对值,得

$$\begin{aligned} | e_1(t) | &\le | e_1(0) | e^{-\alpha t} + \int_0^t e^{-\alpha(t-s)} | \eta_1(s) | ds \\ | e_2(t) | &\le | e_2(0) | e^{-\alpha t} + \int_0^t e^{-\alpha(t-s)} | \eta_2(s) | ds \end{aligned} \tag{25}$$

将上述两式相加，再利用欧式范数得关系，有

$$\| e(t) \| \le 2 \| e(0) \| e^{-\alpha t} + 2\int_0^t e^{-\alpha(t-s)} \| \eta(s) \| ds \tag{26}$$

并引入式(21)，有

$$\begin{aligned} &\| \eta(t) \|^2 \le M + N e^{-\lambda t} \\ &\| \eta(t) \| \le \sqrt{M + N e^{-\lambda t}} \le \sqrt{M} + \sqrt{N} e^{-\lambda t/2} \end{aligned} \tag{27}$$

式中的$M = \dfrac{\varepsilon}{\lambda_1 \lambda}, N = \dfrac{\lambda_2}{\lambda_1} \| \eta(0) \|^2 - M$。将式(27)带入式(26)有

$$\begin{aligned}\|e(t)\| &\le 2\|e(0)\|e^{-\alpha t}+2\int_0^t e^{-\alpha(t-s)}(\sqrt{M}+\sqrt{N}e^{-\lambda t/2})ds \\ &=2\|e(0)\|e^{-\alpha t}+2\sqrt{M}\int_0^t e^{-\alpha(t-s)}ds+2\sqrt{N}\int_0^t e^{-\alpha(t-s)}\bullet e^{-\lambda s/2}ds \\ &=2\|e(0)\|e^{-\alpha t}+\frac{2\sqrt{M}}{\alpha}(1-e^{-\alpha t})+\frac{4\sqrt{N}}{2\alpha-\lambda}(e^{-\lambda t/2}-e^{-\alpha t}) \\ &=(2\|e(0)\|-\frac{2\sqrt{M}}{\alpha}-\frac{4\sqrt{N}}{2\alpha-\lambda})e^{-\alpha t}+\frac{4\sqrt{N}}{2\alpha-\lambda}e^{-\lambda t/2}+\frac{2\sqrt{M}}{\alpha}\end{aligned} \tag{28}$$

也就是

$$\|e(t)\|\le Ae^{-\alpha t}+Be^{-\lambda t/2}+C,\ \ \forall t \tag{29}$$

其中 $A=2\|e(0)\|-\frac{2\sqrt{M}}{\alpha}-\frac{4\sqrt{N}}{2\alpha-\lambda},B=\frac{4\sqrt{N}}{2\alpha-\lambda},C=\frac{2\sqrt{M}}{\alpha}=2\sqrt{\frac{\varepsilon}{\lambda_1\lambda}}$ 。

所以由上述推导可知，在实际选择参数时，好的控制效果对应于较小的C，也就是较小的ε，才能使最终的稳态误差足够的小，从而使误差跟踪一致渐进地趋近于零。

5) Matlab 仿真

首先，仿真结果依赖参数的选取，本次选取的参数，除了题目中已经给定的参数除外，其他的参数为：

变量	变量值的选取
k_1	40
k_2	70
α	50
ε	0.02

表 3

注：要取得更好的控制效果，需要在以上四个参数选择上下功夫。但这不是本次题目关注的重点。

本次 Matlab 仿真采用 Simulink + S_function 的形式进行。

搭建的 Simulink 总体结构如下：

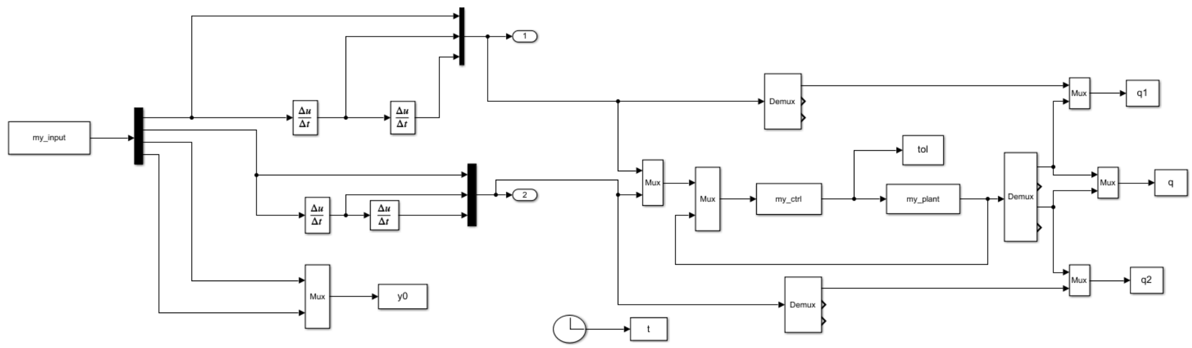

图 2

系统的关节空间变量对输入期望轨迹的跟踪情况为：

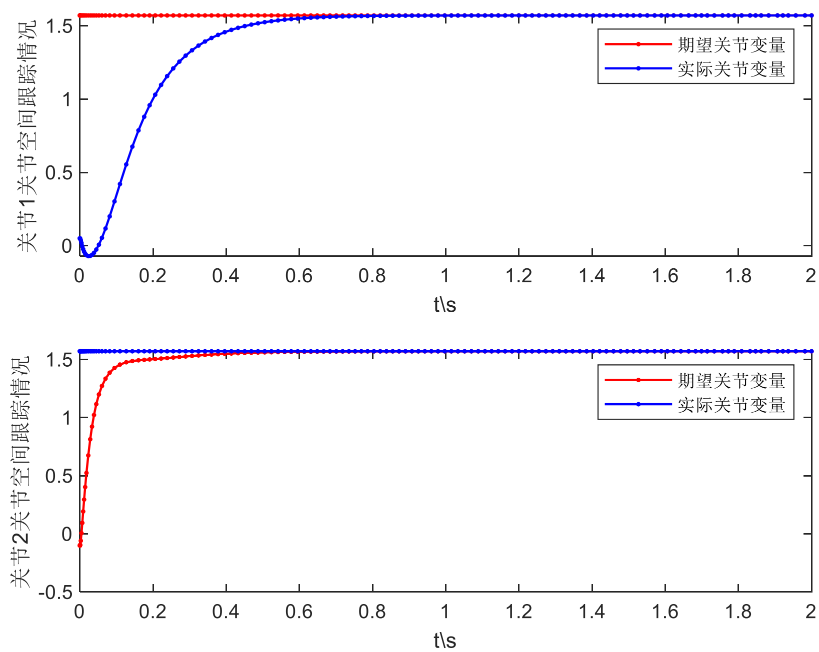

图 3

两个关节的控制力矩变化情况为：

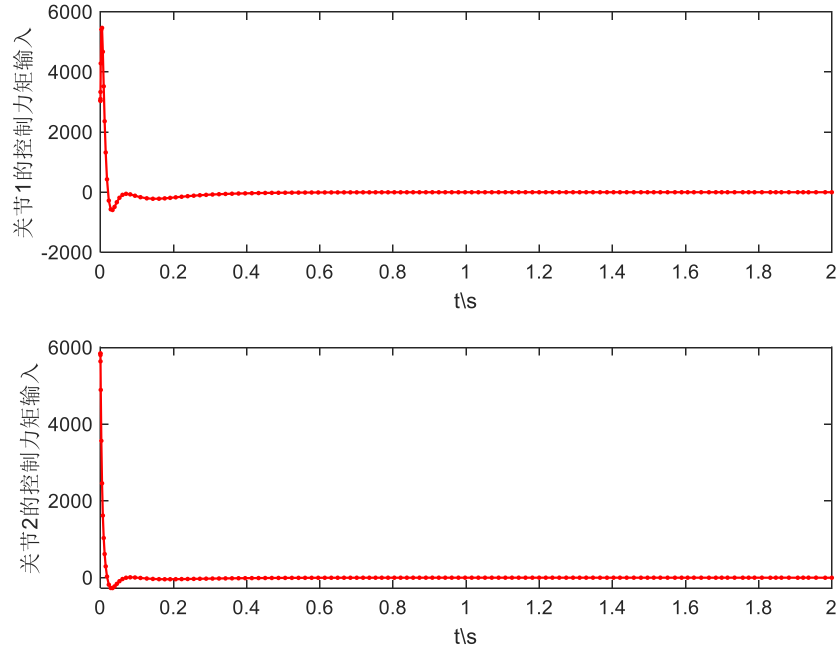

图 4

整个过程的关节运动情况(trajectory)为(红色*为期望关节变量对应的操作空间位置)：

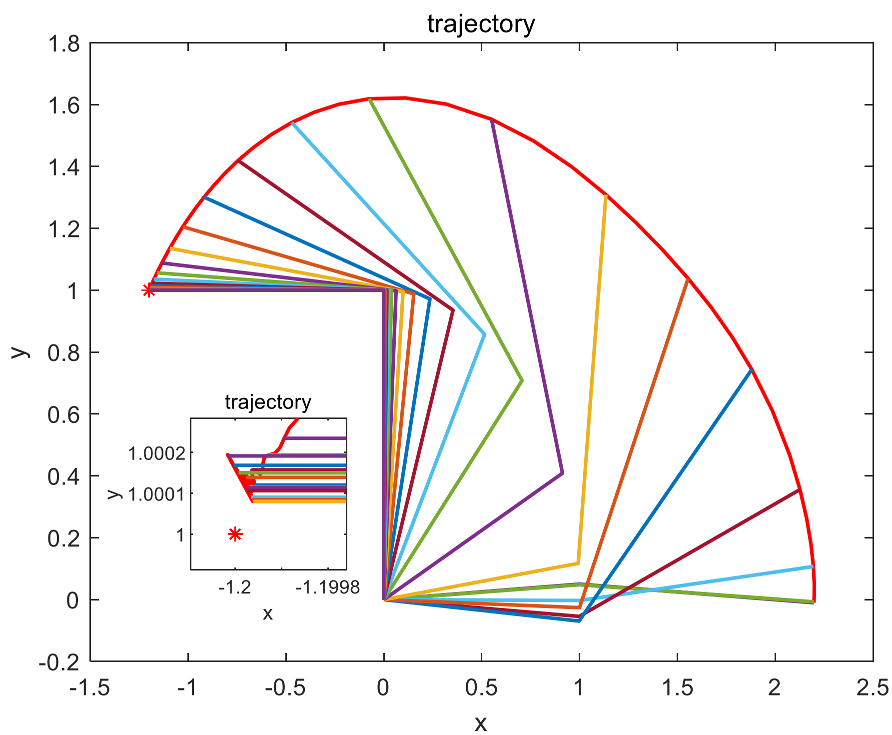

图 5

从图中可以看出：实际控制结果和期望的位置还是有偏差的，这是由于表 3 中相关参数值选取所造成的，但是偏差在小数点后四位，已经可以接受了。

6) 仿真代码

(1) 输入部分 my_input:

```
%input 函数，相当于创建控制指令输入，包含关节空间与操作空间的控制指令输入，实际上就是期望的轨迹
function [sys,x0,str,ts] = inputsignal(t,x,u,flag)
switch flag
case 0
    [sys,x0,str,ts]=mdlInitializeSizes;
case 3
    sys=mdlOutputs(t,x,u);
case {2,4,9}
    sys=[];
otherwise
    error(['Unhandled flag = ',num2str(flag)]);
end
function [sys,x0,str,ts]=mdlInitializeSizes
sizes = simsizes;
sizes.NumOutputs     = 4;
sizes.NumInputs      = 0;
sizes.DirFeedthrough = 0;
sizes.NumSampleTimes = 1;
sys = simsizes(sizes);
x0  = [];
str = [];
ts  = [0 0];
function sys=mdlOutputs(t,x,u)
l1=1;l2=1.2; %初始化机械臂长度参数

xd=-1.2;%初始化期望操作空间轨迹 x
yd=1;   %初始化期望操作空间轨迹 y
qd1=pi/2;
qd2=pi/2;
% 因为逆运动学的多解性，需要判断上肘位和下肘位，从而求解出期望的关节空间轨迹
% if(xd>=0)
%   alfa0=atan(yd/xd);
```

```
% else
%    alfa0=pi+atan(yd/xd);
% end
% beta0=acos((xd^2+yd^2+l1^2-l2^2)/(2*l1*sqrt(xd^2+yd^2)));
% qd2=-acos((xd^2+yd^2-(l1^2+l2^2))/(2*l1*l2));
%
% if(qd2>0)
%    qd1=alfa0-beta0;
% else
%    qd1=alfa0+beta0;
% end
% 但是此次的跟踪控制相当于期望轨迹是不随时间变化的常量，所以并不需要由上述计算而来，只需要填入对应值即可
% 输出为期望的操作空间轨迹以及期望的关节空间轨迹
% 即系统的控制指令输入
sys(1)=qd1;
sys(2)=qd2;
sys(3)=xd;
sys(4)=yd;
```

(2) 控制部分 my_contrl:

```
% 控制器函数
%根据鲁棒逆运动学中控制器的设计来计算控制项 u
function [sys,x0,str,ts] = control_strategy(t,x,u,flag)

switch flag
case 0
    [sys,x0,str,ts]=mdlInitializeSizes;
case 3
    sys=mdlOutputs(t,x,u);
case {2,4,9}
    sys=[];
otherwise
    error(['Unhandled flag = ',num2str(flag)]);
end

function [sys,x0,str,ts]=mdlInitializeSizes
sizes = simsizes;
sizes.NumOutputs     = 4;
sizes.NumInputs      = 10;
```

```
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;
sys = simsizes(sizes);
x0  = [];
str = [];
ts  = [0 0];

function sys=mdlOutputs(t,x,u)
% 输入的控制指令
qd=[u(1) u(4)];
dqd=[u(2) u(5)];
ddqd=[u(3) u(6)];

%实际的关节空间变量输出
q=[u(7) u(9)];
dq=[u(8) u(10)];

%控制指令与输出之间的偏差
e=q-qd;
de=dq-dqd;

% 带入初始参数
m1=1;m2=2;
l1=1;l2=1.5;
r1=0.5;r2=0.6;
J1=0.12;J2=0.25;

q1=q(1);
dq1=dq(1);
q2=q(2);
dq2=dq(2);

% 惯性矩阵
%J=[J1+J2+2*m2*r2*l1*cos(q2) J2+m2*r2*l1*cos(q2);
%    J2+m2*r2*l1*cos(q2) J2];

J=[J1+m1*r1^2+J2+m2*r2^2+m2*l1^2+2*m2*l1*r2*cos(pi/6)*cos(q2)+2*m2*l1*r2*sin(pi/6)*sin(q2) J2+m2*r2^2+m2*l1*r2*cos(pi/6)*cos(q2)+m2*l1*r2*sin(pi/6)*sin(q2);
   J2+m2*r2^2+m2*l1*r2*cos(pi/6)*cos(q2)+m2*l1*r2*sin(pi/6)*sin(q2) J2+m2*r2^2];

%科里奥利矩阵
```

```
%f=[-2*m2*r2*l1*dq2*sin(q2) -m2*r2*l1*dq2*sin(q2);
  %  m2*r2*l1*dq1*sin(q2) 0];

f=[-(m2*l1*r2*cos(pi/6)*sin(q2)-m2*l1*r2*sin(pi/6)*cos(q2))*dq2 -(m2*l1*r2*cos(pi/6)*sin(q2)-m2*l1*r2*sin(pi/6)*cos(q2))*(dq1+dq2);
  (m2*l1*r2*cos(pi/6)*sin(q2)-m2*l1*r2*sin(pi/6)*cos(q2))*dq1 0];

%可调参数
alfa=50;
eq=0.02;
k0=40;
k1=70;

% 控制器需要使用到的参数
K=[k0,0;0,k1];
Kxi=de+alfa*e;

rou=3.0*tanh(2.6*norm(Kxi));
Un=[sin(6*Kxi(1)) sin(6*Kxi(2))];
v=Kxi*rou^2/(norm(Kxi)*rou+eq);
%v=[0,0];
w=J*alfa*de'+f*alfa*e';

%控制器的具体形式
ut=-K*Kxi'-w-v';
% B=[0;1];
% A=[0,1;-k0,-k1];
% if
% deltaa=-
% ut=ddqd-k0*e-k1*de+deltaa;
tol=ut+J*ddqd'+f*dqd';

sys(1)=tol(1);
sys(2)=tol(2);
sys(3)=Kxi(1);
sys(4)=Kxi(2);
```

(3) 被控对象部分 my_plant:

```
function [sys,x0,str,ts]=robotplant(t,x,u,flag,x0,y0)
switch flag,
```

```
case 0
    [sys,x0,str,ts]=mdlInitializeSizes(x0,y0);
case 1
    sys=mdlDerivatives(t,x,u);
case 3
    sys=mdlOutputs(t,x,u);
case {2, 4, 9 }
    sys = [];
otherwise
    error(['Unhandled flag = ',num2str(flag)]);
end
function [sys,x0,str,ts]=mdlInitializeSizes(x0,y0)
sizes = simsizes;
sizes.NumContStates  = 4;
sizes.NumDiscStates  = 0;
sizes.NumOutputs     = 4;
sizes.NumInputs      = 4;
sizes.DirFeedthrough = 0;
sizes.NumSampleTimes = 0;
sys=simsizes(sizes);

l1=1;l2=1.2;
if(x0>=0)
   alfa0=atan(y0/x0);
else
   alfa0=pi+atan(y0/x0);
end

beta0=acos((x0^2+y0^2+l1^2-l2^2)/(2*l1*sqrt(x0^2+y0^2)));
qd2_0=-acos((x0^2+y0^2-(l1^2+l2^2))/(2*l1*l2));
if(qd2_0>0)
   qd1_0=alfa0-beta0;
else
   qd1_0=alfa0+beta0;
end

x0=[0.05;0.01;-0.1;00.02];   %状态变量的初始值
str=[];
ts=[];

function sys=mdlDerivatives(t,x,u)
```

```
% 带入初始参数
m1=1;m2=2;
l1=1;l2=1.5;
r1=0.5;r2=0.6;
J1=0.12;J2=0.25;

q1=x(1);
dq1=x(2);
q2=x(3);
dq2=x(4);

% J=[J1+J2+2*m2*r2*l1*cos(q2) J2+m2*r2*l1*cos(q2);
%    J2+m2*r2*l1*cos(q2) J2];

J=[J1+m1*r1^2+J2+m2*r2^2+m2*l1^2+2*m2*l1*r2*cos(pi/6)*cos(q2)+2*m2*l1*r2*sin(pi/6)*sin(q2) J2+m2*r2^2+m2*l1*r2*cos(pi/6)*cos(q2)+m2*l1*r2*sin(pi/6)*sin(q2);
   J2+m2*r2^2+m2*l1*r2*cos(pi/6)*cos(q2)+m2*l1*r2*sin(pi/6)*sin(q2) J2+m2*r2^2];

% f=[-2*m2*r2*l1*dq2*sin(q2) -m2*r2*l1*dq2*sin(q2);
%     m2*r2*l1*dq1*sin(q2) 0];
f=[-(m2*l1*r2*cos(pi/6)*sin(q2)-m2*l1*r2*sin(pi/6)*cos(q2))*dq2 - (m2*l1*r2*cos(pi/6)*sin(q2)-m2*l1*r2*sin(pi/6)*cos(q2))*(dq1+dq2);
  (m2*l1*r2*cos(pi/6)*sin(q2)-m2*l1*r2*sin(pi/6)*cos(q2))*dq1 0];

tol(1)=u(1);
tol(2)=u(2);
Kxi=[u(3) u(4)];

Delta=[0.1*cos(0.2-0.01*t);0.1*sin(0.2-0.01*t)];

S=inv(J)*(tol'-f*[dq1;dq2]-Delta);

sys(1)=x(2);
sys(2)=S(1);
sys(3)=x(4);
sys(4)=S(2);
function sys=mdlOutputs(t,x,u)
sys(1)=x(1);
sys(2)=x(2);
sys(3)=x(3);
```

```
sys(4)=x(4);
```

(4) 画图部分 my_plot_trajectory:

```
close all;

figure(1);
subplot(211);
plot(t,q1(:,1),'r.-','LineWidth',1);hold on
plot(t,q1(:,2),'b.-','LineWidth',1)
legend('期望关节变量','实际关节变量')
xlabel('t\s');ylabel('关节 1 关节空间跟踪情况');
subplot(212);
plot(t,q2(:,1),'r.-','LineWidth',1);hold on
plot(t,q2(:,2),'b.-','LineWidth',1);
legend('期望关节变量','实际关节变量')
xlabel('t\s');ylabel('关节 2 关节空间跟踪情况');

figure(2);
subplot(211);
plot(t,tol(:,1),'r.-','LineWidth',1);
xlabel('t\s');ylabel('关节 1 的控制力矩输入');
subplot(212);
plot(t,tol(:,2),'r.-','LineWidth',1);
xlabel('t\s');ylabel('关节 2 的控制力矩输入');

figure(3);

l1=1;l2=1.2;
k=size(q);
x_joint1=l1*cos(q(:,1));
y_joint1=l1*sin(q(:,1));
x_joint2=l1*cos(q(:,1))+l2*cos(q(:,1)+q(:,2));
y_joint2=l1*sin(q(:,1))+l2*sin(q(:,1)+q(:,2));
X=[zeros(k(1),1),x_joint1,x_joint2];
Y=[zeros(k(1),1),y_joint1,y_joint2];
plot(X(:,3),Y(:,3),'r','LineWidth',1.5),hold on
plot(y0(:,1),y0(:,2),'r*');hold on
for i=1:3:k(1)
    X(i,:)
    Y(i,:)
```

```
    plot(X(i,:),Y(i,:),'LineWidth',1.5)
    hold on
    drawnow
end
xlabel('x');ylabel('y');title('trajectory')
```