基于模糊控制的吊车模型仿真实验

学院（系）：电子与信息工程学院

专　　业：自动化

作　　者：姚天亮(2150248)

指导老师：朱芳来、蔚瑞华

日　　期：2023 年 12 月

《智能控制》期末大作业

考虑如下的吊车模型

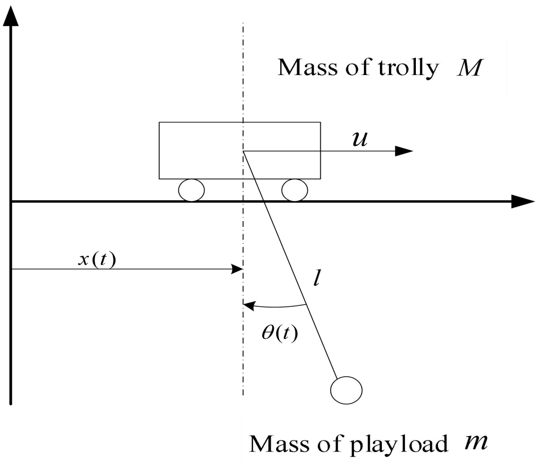


图 1：吊车模型

对吊车模型各个部分进行受力分析，可以得到该物理模型的动力学方程为：

$$\begin{cases}(M+m)\ddot{x}+ml\ddot{\theta}\cos\theta-ml\dot{\theta}^2\sin\theta=u+d\\ ml^2\ddot{\theta}+ml\ddot{x}\cos\theta+mgl\sin\theta=0\end{cases}\tag{1}$$

1. 如果选取状态变量为 $x_1=x$， $x_2=\dot{x}$， $x_3=\theta$， $x_4=\dot{\theta}$，试推导吊车模型的动力学方程(1)可以由如下的仿射非线性系统状态空间模型表示(20 分)

$$\dot{x}=f(x)+g(x)(u+d)\tag{2}$$

其中

$$f(\mathbf{x})=\begin{bmatrix}x_2\\ \dfrac{mlx_4^2\sin x_3+mg\sin x_3\cos x_3}{(M+m\sin^2 x_3)}\\ x_4\\ -\dfrac{mlx_4^2\sin x_3\cos x_3+(M+m)g\sin x_3}{l(M+m\sin^2 x_3)}\end{bmatrix},g(\mathbf{x})=\frac{1}{M+m\sin^2 x_3}\begin{bmatrix}0\\1\\0\\-\dfrac{\cos x_3}{l}\end{bmatrix}$$

其中，$\mathbf{x}=\begin{bmatrix}x_1 & x_2 & x_3 & x_4\end{bmatrix}^T$ 是状态向量，d 是未知干扰，如下可以置 d 为零(即不存在)。

联立消 $\ddot{\theta}$，将②代入①：

$$(M+m)\ddot{x}+\left(\frac{-ml\ddot{x}\cos\theta-mgl\sin\theta}{l}\cos\theta\right)-ml\dot{\theta}^2\sin\theta=u+d.$$

即：$(M+m)\ddot{x}-m\ddot{x}\cos^2\theta-mg\sin\theta\cos\theta-ml\dot{\theta}^2\sin\theta=u+d$

$$(M+m\sin^2\theta)\ddot{x}=u+d+ml\dot{\theta}^2\sin\theta+mg\sin\theta\cos\theta \quad ③$$

③代入②：

$$ml^2\ddot{\theta}+ml\cos\theta\,\frac{(u+d)^2+ml\dot{\theta}^2\sin\theta+mg\sin\theta\cos\theta}{M+m\sin^2\theta}+mgl\sin\theta=0.$$

$$\therefore \ddot{\theta}=-\frac{\cos\theta}{(M+m\sin^2\theta)l}-\frac{ml\dot{\theta}^2\sin\theta\cos\theta+(M+m)g\sin\theta}{l(M+m\sin^2\theta)}.$$

状态变量：$z_1(t)=x$，$z_2(t)=\dot{x}$，

$z_3(t)=\theta$，$z_4(t)=\dot{\theta}$.

求状态变量的导数：

$$\begin{bmatrix}\frac{dz_1(t)}{dt}\\ \frac{dz_2(t)}{dt}\\ \frac{dz_3(t)}{dt}\\ \frac{dz_4(t)}{dt}\end{bmatrix}=\begin{bmatrix}x_2\\ \frac{mlx_4^2\sin x_3+mg\sin x_3\cos x_3}{M+m\sin^2x_3}\\ x_4\\ -\frac{mlx_4^2\sin x_3\cos x_3+(M+m)g\sin x_3}{l(M+m\sin^2x_3)}\end{bmatrix}+\begin{bmatrix}0\\ \frac{u+d}{M+m\sin^2x_3}\\ 0\\ -\frac{(u+d)\cos x_3}{(M+m\sin^2x_3)l}\end{bmatrix}$$

结果与题设一致，得证！

2. 按本作业附件给出的算法，取三个操作点 $\mathbf{x}_1=\begin{bmatrix}0 & 0 & 0 & 0\end{bmatrix}^T$，$\mathbf{x}_2=\begin{bmatrix}10 & 0 & \pi/18 & 0\end{bmatrix}^T$，$\mathbf{x}_1=\begin{bmatrix}10 & 1 & \pi/18 & 0\end{bmatrix}^T$，计算 T-S 模糊模型子系统系统矩阵 A_1,B_1,A_2,B_2,A_3,B_3（20 分）

（1）验证第一个操作点：

```
第一个操作点的A1=                          第一个操作点的B1=
[0, 1,                      0, 0]            0
[0, 0,              (g*m)/M, 0]            1/M
[0, 0,                      0, 1]            0
[0, 0, -(g*(M + m))/(M*l), 0]   -1/(M*l)
```

（2）计算第二个操作点：

由于屏幕限制，分多张图给出结果

```
第二个操作点的A2=
[                                                                                                                                                                     0, 1,
[-(10*((pi*((g*m*cos(pi/18)^2 - g*m*sin(pi/18)^2)/(M + m*sin(pi/18)^2) - (2*g*m^2*cos(pi/18)^2*sin(pi/18)^2)/(M + m*sin(pi/18)^2)^2))/18 - (g*m*cos(pi/18)*sin(pi/18))/(M + m*sin(pi/18)^2)))/(pi^2/324 + 100), 0,
[                                                                                                                                                                     0, 0,
[     (10*((pi*((g*cos(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2)) - (2*g*m*cos(pi/18)*sin(pi/18)^2*(M + m))/(l*(M + m*sin(pi/18)^2)^2)))/18 - (g*sin(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2))))/(pi^2/324 + 100), 0,
```

```
(g*m*cos(pi/18)^2 - g*m*sin(pi/18)^2)/(M + m*sin(pi/18)^2) - (pi*((pi*((g*m*cos(pi/18)^2 - g*m*sin(pi/18)^2)/(M + m*sin(pi/18)^2) - (2*g*m^2*cos(pi/18)^2*sin(pi/18)^2)/(M + m*sin(pi/18)^2)^2))/18 - (g*m*cos(pi/18)*sin(pi/18))/(M + m*si
      (pi*((pi*((g*cos(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2)) - (2*g*m*cos(pi/18)*sin(pi/18)^2*(M + m))/(l*(M + m*sin(pi/18)^2)^2)))/18 - (g*sin(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2))))/(18*(pi^2/324 + 100)) - (g*cos(pi/18)*(M + m
```

```
                                                                                              0, 0]
)/(M + m*sin(pi/18)^2)))/(18*(pi^2/324 + 100)) - (2*g*m^2*cos(pi/18)^2*sin(pi/18)^2)/(M + m*sin(pi/18)^2)^2, 0]
                                                                                              0, 1]
/18)*(M + m))/(l*(M + m*sin(pi/18)^2)) + (2*g*m*cos(pi/18)*sin(pi/18)^2*(M + m))/(l*(M + m*sin(pi/18)^2)^2), 0]
```

```
第二个操作点的B2=
                                 0
             1/(M + m*sin(pi/18)^2)
                                 0
-cos(pi/18)/(l*(M + m*sin(pi/18)^2))
```

（3）计算第三个操作点：

```
第三个操作点的A3=
[                                                                                                                                                                     0,
[-(10*((pi*((g*m*cos(pi/18)^2 - g*m*sin(pi/18)^2)/(M + m*sin(pi/18)^2) - (2*g*m^2*cos(pi/18)^2*sin(pi/18)^2)/(M + m*sin(pi/18)^2)^2))/18 - (g*m*cos(pi/18)*sin(pi/18))/(M + m*sin(pi/18)^2)))/(pi^2/324 + 101),
[                                                                                                                                                                     0,
[     (10*((pi*((g*cos(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2)) - (2*g*m*cos(pi/18)*sin(pi/18)^2*(M + m))/(l*(M + m*sin(pi/18)^2)^2)))/18 - (g*sin(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2))))/(pi^2/324 + 101),
```

```
                                                                                                                                                                     1,
-((pi*((g*m*cos(pi/18)^2 - g*m*sin(pi/18)^2)/(M + m*sin(pi/18)^2) - (2*g*m^2*cos(pi/18)^2*sin(pi/18)^2)/(M + m*sin(pi/18)^2)^2))/18 - (g*m*cos(pi/18)*sin(pi/18))/(M + m*sin(pi/18)^2))/(pi^2/324 + 101),
                                                                                                                                                                     0,
     ((pi*((g*cos(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2)) - (2*g*m*cos(pi/18)*sin(pi/18)^2*(M + m))/(l*(M + m*sin(pi/18)^2)^2)))/18 - (g*sin(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2)))/(pi^2/324 + 101),
```

```
(g*m*cos(pi/18)^2 - g*m*sin(pi/18)^2)/(M + m*sin(pi/18)^2) - (pi*((pi*((g*m*cos(pi/18)^2 - g*m*sin(pi/18)^2)/(M + m*sin(pi/18)^2) - (2*g*m^2*cos(pi/18)^2*sin(pi/18)^2)/(M + m*sin(pi/18)^2)^2))/18 - (g*m*cos(pi/18)*sin(pi/18))/(M + m*
      (pi*((pi*((g*cos(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2)) - (2*g*m*cos(pi/18)*sin(pi/18)^2*(M + m))/(l*(M + m*sin(pi/18)^2)^2)))/18 - (g*sin(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2))))/(18*(pi^2/324 + 101)) - (g*cos(pi/18)*(M +
```

```
                                                                                                                                                                     0, 0]
i/18)^2)/(M + m*sin(pi/18)^2) - (2*g*m^2*cos(pi/18)^2*sin(pi/18)^2)/(M + m*sin(pi/18)^2)^2))/18 - (g*m*cos(pi/18)*sin(pi/18))/(M + m*sin(pi/18)^2)))/(18*(pi^2/324 + 101)) - (2*g*m^2*cos(pi/18)^2*sin(pi/18)^2)/(M + m*sin(pi/18)^2)^2, 0]
                                                                                                                                                                     0, 1]
 + m))/(l*(M + m*sin(pi/18)^2)^2)))/18 - (g*sin(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2))))/(18*(pi^2/324 + 101)) - (g*cos(pi/18)*(M + m))/(l*(M + m*sin(pi/18)^2)) + (2*g*m*cos(pi/18)*sin(pi/18)^2*(M + m))/(l*(M + m*sin(pi/18)^2)^2), 0]
```

```
第三个操作点的B3=
                                 0
             1/(M + m*sin(pi/18)^2)
                                 0
-cos(pi/18)/(l*(M + m*sin(pi/18)^2))
```

因此，模糊子系统矩阵计算完毕。如果能够给出 m、M、g、l、和 pi 的取值，则结果可以美观很多。经过手算平衡点的验证，此计算结果没有问题。

这里代入仿真参数的选取：

```
第一个操作点的A1=
         0    1.0000         0         0
         0         0    4.9050         0
         0         0         0    1.0000
         0         0   -3.6787         0

第一个操作点的B1=
         0
    0.0100
         0
   -0.0025

第二个操作点的A2=
         0    1.0000         0         0
    0.0058         0    4.4016         0
         0         0         0    1.0000
   -0.0025         0   -3.4631         0

第二个操作点的B2=
         0
    0.0099
         0
   -0.0024
```

```
第三个操作点的A3=
         0    1.0000         0         0
    0.0058    0.0006    4.4016         0
         0         0         0    1.0000
   -0.0025   -0.0002   -3.4631         0

第三个操作点的B3=
         0
    0.0099
         0
   -0.0024
```

本题代码如下：

```
clear all;close all;
%syms M m l g ;%设定变量 M、m、l、g 为符号变量
syms x1 x2 x3 x4;%设定变量 x1、x2、x3、x4
M=100;m=50;l=4;g=9.81;
x01=[0 0 0 0]';
x02=[10 0 pi/18 0]';
x03=[10 1 pi/18 0]';%给出题目要求的三个操作点
x=[x1;x2;x3;x4];
```

```
f=[x2;(m*l*x4^2*sin(x3)+m*g*sin(x3)*cos(x3))/(M+m*sin(x3)*sin(x3));x4;-(m*l
*x4^2*sin(x3)*cos(x3)+(M+m)*g*sin(x3))/l/(M+m*sin(x3)*sin(x3))];
gx=1/(M+m*sin(x3)*sin(x3))*[0;1;0;-cos(x3)/l];
J=jacobian(f,x);
Ai=J+(f-J*x)*x'/(x'*x);
A1=double(subs(J,x,x01));
B1=double(subs(gx,x,x01));
A2=double(subs(Ai,x,x02));
B2=double(subs(gx,x,x02));
A3=double(subs(Ai,x,x03));
B3=double(subs(gx,x,x03));
disp('第一个操作点的 A1=');disp(A1);
disp('第一个操作点的 B1=');disp(B1);
disp('第二个操作点的 A2=');disp(A2);
disp('第二个操作点的 B2=');disp(B2);
disp('第三个操作点的 A3=');disp(A3);
disp('第三个操作点的 B3=');disp(B3);
```

3. 取如下的三个权值：$h_1(x_3)=\dfrac{M\sin^2 x_3}{M+m\sin^2 x_3}$，$h_2(x_3)=\dfrac{M\cos^2 x_3}{M+m\sin^2 x_3}$，

$h_3(x_3)=\dfrac{m\sin^2 x_3}{M+m\sin^2 x_3}$，验证开环系统的 T-S 模糊模型

$$\dot{x}(t)=\sum_{i=1}^{3}h_i\left(x_3(t)\right)\left[A_i x(t)+B_i\left(u(t)+d(t)\right)\right] \qquad (3)$$

与仿射非线性系统的吻合程度：(20 分)

(1)提供四张仿真图，每张仿真图，分别画出 T-S 模糊系统(3)和仿射非线性系统(2)的状态 $x_i(t)(i=1,2,3,4)$ 的轨迹图；(时间区间：[0,30])

(2)给出仿真程序；

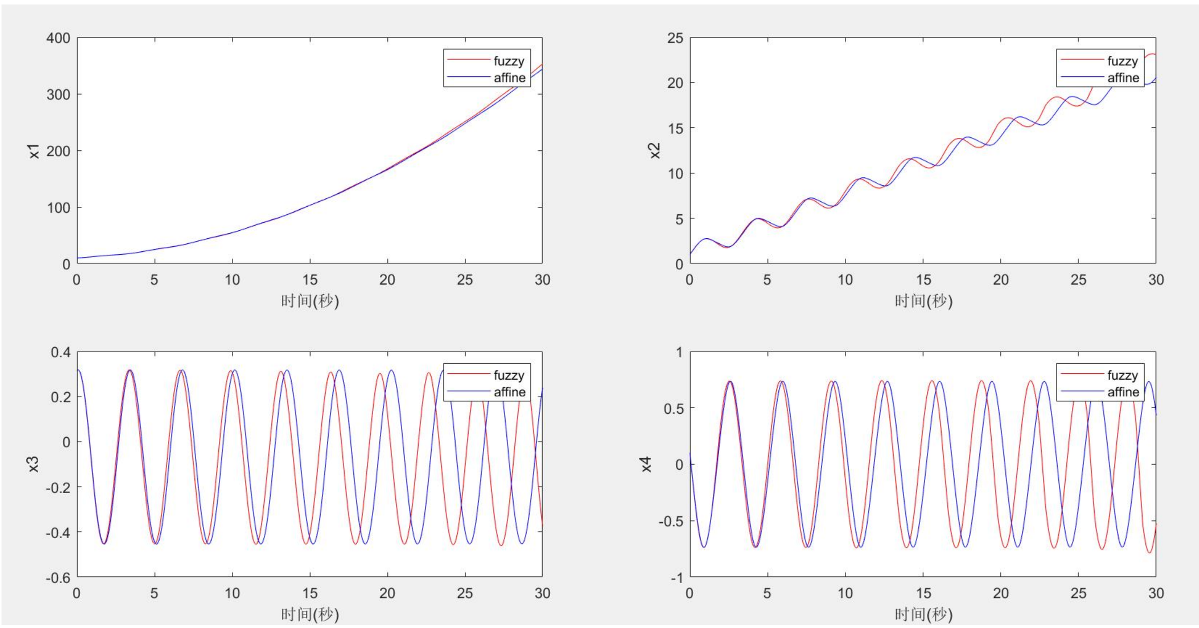
1、提供四张模糊系统与仿射非线性系统的轨迹仿真图

模糊控制方法由如下给出：

x(3)<=0.25 时隶属度函数为 1；

0.25<x(3)<=0.75 时隶属度函数为-2x+1.5

0.75<x(3)时隶属度函数为 0；



2、给出仿真程序

核心新增部分：

```
x0(1:4)=[10 1 pi/10 0.1]';
[t,open_fuzzy]=ode45('open_fuzzy_module',0:0.01:30, x0);
[t,open_affine]=ode45('open_affine_module',0:0.01:30, x0);
figure(1);
subplot(2,2,1);
plot(t, open_fuzzy(:,1),'r',t,open_affine(:,1),'b');
xlabel('时间(秒)');
ylabel('x1');
legend('fuzzy','affine');
subplot(2,2,2);
plot(t, open_fuzzy(:,2),'r',t,open_affine(:,2),'b');
xlabel('时间(秒)');
ylabel('x2');
legend('fuzzy','affine');
subplot(2,2,3);
plot(t, open_fuzzy(:,3),'r',t,open_affine(:,3),'b');
xlabel('时间(秒)');
ylabel('x3');
legend('fuzzy','affine');
subplot(2,2,4);
plot(t, open_fuzzy(:,4),'r',t,open_affine(:,4),'b');
xlabel('时间(秒)');
ylabel('x4');
legend('fuzzy','affine');
```

```
function xdot = close_fuzzy_module(t, x);
global A0 A1 A2 B0 B1 B2 u M m K0 K1 K2;
if(x(3)<=0.25)
    h0=1;
elseif(0.25<x(3)<=0.75)
    %h0=sin(x(3))/x(3);
    h0=-2*x(3)+1.5;
else
    h0=0;
end
h1=(1-h0)/4;
h2=3*h1;
u0=u-h0*K0*x-h1*K1*x-h2*K2*x;
```

```
xdot=h0*(A0*x+B0*u0)+h1*(A1*x+B1*u0)+h2*(A2*x+B2*u0);
```

```
function xdot = open_affine_module(t, x);
global M m l u g;
f=[x(2);(m*l*x(4)^2*sin(x(3))+m*g*sin(x(3))*cos(x(3)))/(M+m*sin(x(3))*sin(x(3)));x(4);-(m*l*x(4)^2*sin(x(3))*cos(x(3))+(M+m)*g*sin(x(3)))/l/(M+m*sin(x(3))*sin(x(3)))];
gx=1/(M+m*sin(x(3))*sin(x(3)))*[0;1;0;-cos(x(3))/l];
xdot=f+gx*u;
```

4. 基于 T-S 模糊模型，设计状态反馈控制器：$u(t)=\sum_{i=1}^{N}h_i\left(x_3(t)\right)K_i x(t)$，将各个子系统的极点配置到左半平面，仿真验证在该控制律下：

(1)T-S 模糊闭环系统的控制效果，并画出闭环系统的状态轨迹图；

(2)仿射非线性系统(2)的闭环系统的效果，并画出闭环系统的状态轨迹图；

(3)给出仿真代码；(20 分)

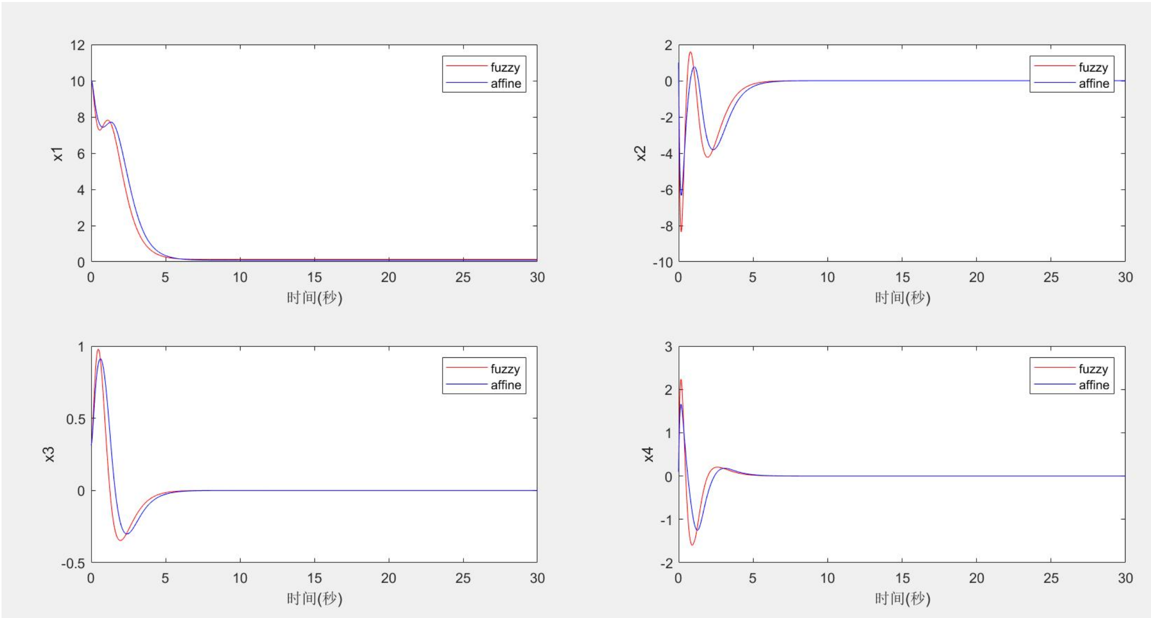
注：三个操作点分别配置到极点

$$p_1=\begin{bmatrix}-2.7534 & -1.5464 & -1.6355 & -2.7453\end{bmatrix}$$

$$p_2=\begin{bmatrix}-1.5324 & -2.4353 & -2.6961 & -2.6343\end{bmatrix}$$

$$p_3=\begin{bmatrix}-2.6594 & -1.6445 & -2.3595 & -3.2659\end{bmatrix}$$

1、T-S 模糊闭环系统的轨迹图　　2、仿射非线性系统闭环轨迹图



3、给出仿真代码

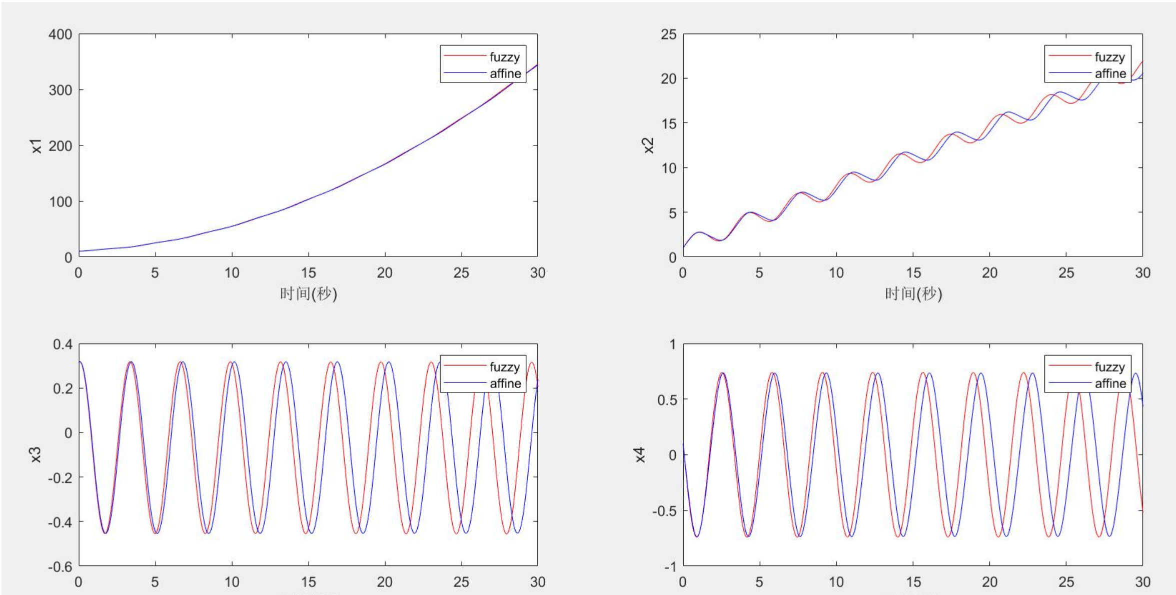
核心新增部分：

```
x0(1:4)=[10 1 pi/10 0.1]';
p1=[-2.7534 -1.5464 -1.6355 -2.7453];
p2=[-1.5324 -2.4353 -2.6961 -2.6343];
p3=[-2.6594 -1.6445 -2.3595 -3.2659];
K0=place(A0,B0,p1);K1=place(A1,B1,p2);K2=place(A2,B2,p3);
[t,close_fuzzy]=ode45('close_fuzzy_module',0:0.01:30, x0);
[t,close_affine]=ode45('close_affine_module',0:0.01:30, x0);
figure(2);
subplot(2,2,1);
plot(t, close_fuzzy(:,1),'r',t,close_affine(:,1),'b');
xlabel('时间(秒)');
ylabel('x1');
legend('fuzzy','affine');
subplot(2,2,2);
plot(t, close_fuzzy(:,2),'r',t,close_affine(:,2),'b');
xlabel('时间(秒)');
ylabel('x2');
```

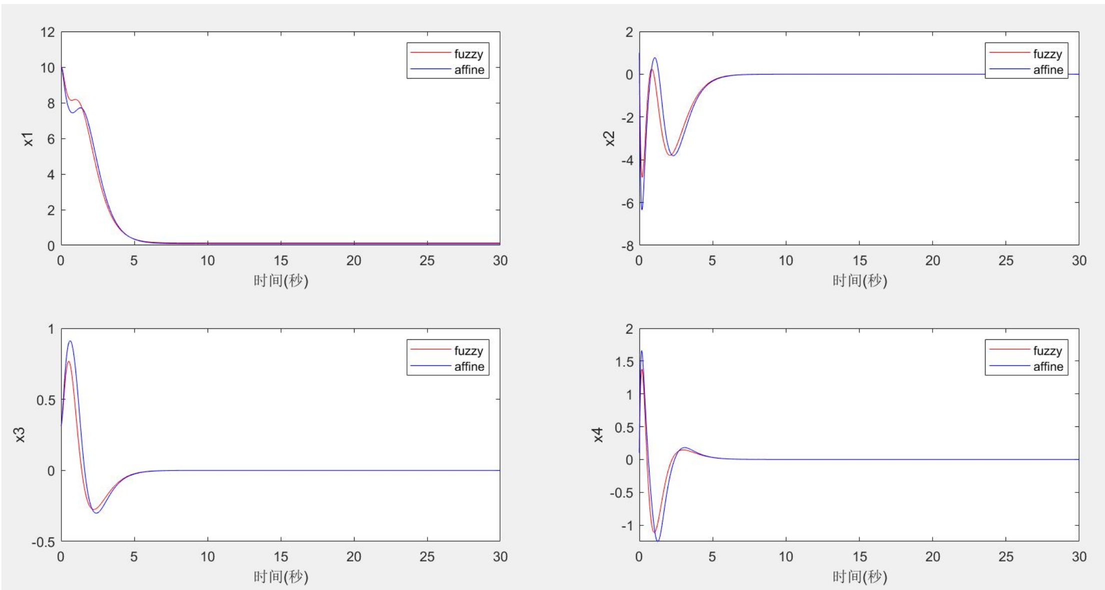
```
legend('fuzzy','affine');
subplot(2,2,3);
plot(t, close_fuzzy(:,3),'r',t,close_affine(:,3),'b');
xlabel('时间(秒)');
ylabel('x3');
legend('fuzzy','affine');
subplot(2,2,4);
plot(t, close_fuzzy(:,4),'r',t,close_affine(:,4),'b');
xlabel('时间(秒)');
ylabel('x4');
legend('fuzzy','affine');


function xdot = close_fuzzy_module(t, x);
global A0 A1 A2 B0 B1 B2 u M m K0 K1 K2;
if(x(3)<=0.25)
    h0=1;
elseif(0.25<x(3)<=0.75)
    %h0=sin(x(3))/x(3);
    h0=-2*x(3)+1.5;
else
    h0=0;
end
h1=(1-h0)/4;
h2=3*h1;
u0=u-h0*K0*x-h1*K1*x-h2*K2*x;
xdot=h0*(A0*x+B0*u0)+h1*(A1*x+B1*u0)+h2*(A2*x+B2*u0);


function xdot = close_affine_module(t, x);
global M m l u g K0 K1 K2;
f=[x(2);(m*l*x(4)^2*sin(x(3))+m*g*sin(x(3))*cos(x(3)))/(M+m*sin(x(3))*sin(x
(3)));x(4);-(m*l*x(4)^2*sin(x(3))*cos(x(3))+(M+m)*g*sin(x(3)))/l/(M+m*sin(x
(3))*sin(x(3)))];
gx=1/(M+m*sin(x(3))*sin(x(3)))*[0;1;0;-cos(x(3))/l];
h0=M*(sin(x(3)))^2/(M+m*sin(x(3))*sin(x(3)));
h1=M*(cos(x(3)))^2/(M+m*sin(x(3))*sin(x(3)));
h2=m*(sin(x(3)))^2/(M+m*sin(x(3))*sin(x(3)));
xdot=f+gx*(u-h0*K0*x-h1*K1*x-h2*K2*x);
```

5. 对仿真结果进行分析和讨论(20 分)

从开环轨迹图的对比可以看出，两张图之间的轨迹差异比较明显，说明模糊控制器的模型选择的不够好 用 trimf 类型实现模糊控制效果不太好，可以改进尝试其他的隶属度函数。将 h(0)的取值改为 sin(x)/x 之后，图像得到了极大的改善。



改善后的开环轨迹图



改善后的闭环轨迹图

附件：

1、仿真参数选取：

仿真参数选取：吊车车体的质量$M=100kg$,货物的质量$m=50kg$,吊绳的长度$l=4m$,吊车的动力$u=100N$，重力加速度取 $g=9.81N/kg$，系统初值设为$x_0=\begin{bmatrix}10 & 1 & pi/10 & 0.1\end{bmatrix}$。

2、附件算法

对非线性项 $f(x)=\left[f_1^T(x) \quad \cdots \quad f_n^T(x)\right]^T$，记

$$\nabla f_k(x)=\begin{bmatrix}\dfrac{\partial f_k(x)}{\partial x_1}\\ \vdots \\ \dfrac{\partial f_k(x)}{\partial x_n}\end{bmatrix},(k=1,\cdots,n)$$

它是$f(x)$的梯度，在记它的 Jacobian 矩阵为

$$J(x)\triangleq\frac{\partial f(x)}{\partial x^T}\triangleq\begin{bmatrix}\nabla^T f_1(x)\\ \nabla^T f_2(x)\\ \vdots \\ \nabla^T f_n(x)\end{bmatrix}=\begin{bmatrix}\dfrac{\partial f_1(x)}{\partial x_1} & \dfrac{\partial f_1(x)}{\partial x_2} & \cdots & \dfrac{\partial f_1(x)}{\partial x_n}\\ \dfrac{\partial f_2(x)}{\partial x_1} & \dfrac{\partial f_2(x)}{\partial x_2} & \cdots & \dfrac{\partial f_2(x)}{\partial x_n}\\ \cdots & \cdots & \cdots & \cdots \\ \dfrac{\partial f_n(x)}{\partial x_1} & \dfrac{\partial f_n(x)}{\partial x_2} & \cdots & \dfrac{\partial f_n(x)}{\partial x_n}\end{bmatrix}\in\mathbf{R}^{n\times n}$$

为了构造含 N 个子系统的 T-S 模糊系统，选取 N 个操作点$x_{0,i}\,(i=1,\cdots,N)$，其中必须包含系统的平衡点$x_{0,1}$

对平衡点$x_{0,1}$ 对于的子系统计算如下

$$A_1=J\left(x_{0,1}\right) \text{ and } B_1=g\left(x_{0,1}\right) \tag{A1}$$

对其他操作点$x_{0,i}\,(i=2,\cdots,N)$，对于的子系统计算如下

$$A_i=J\left(x_{0,i}\right)+\frac{1}{\left\|x_{0,i}\right\|^2}\left[f\left(x_{0,i}\right)-J\left(x_{0,i}\right)x_{0,i}\right]x_{0,i}^T \text{ and } B_i=g\left(x_{0,i}\right) \tag{A2}$$

附录：整理好的完整代码

fuzzy_trolly.m

```
clear all;close all;
global M m l u g;
global A0 A1 A2 B0 B1 B2 x3 K0 K1 K2;
syms M m l g ;
M=100;m=50;l=4;g=9.81;u=100;
syms x1 x2 x3 x4;
x00=[0 0 0 0]';x01=[10 0 pi/18 0]';x02=[10 1 pi/18 0]';
x=[x1;x2;x3;x4];
f=[x2;(m*l*x4^2*sin(x3)+m*g*sin(x3)*cos(x3))/(M+m*sin(x3)*sin(x3));x4;-(m*l
*x4^2*sin(x3)*cos(x3)+(M+m)*g*sin(x3))/l/(M+m*sin(x3)*sin(x3))];
gx=1/(M+m*sin(x3)*sin(x3))*[0;1;0;-cos(x3)/l];
J=jacobian(f,x);
Ai=J+(f-J*x)*x'/(x'*x);
A0=double(subs(J,x,x00));
B0=double(subs(gx,x,x00));
A1=double(subs(Ai,x,x01));
B1=double(subs(gx,x,x01));
A2=double(subs(Ai,x,x02));
B2=double(subs(gx,x,x02));
disp('第一个操作点的 A1=');disp(A0);
disp('第一个操作点的 B1=');disp(B0);
disp('第二个操作点的 A2=');disp(A1);
disp('第二个操作点的 B2=');disp(B1);
disp('第三个操作点的 A3=');disp(A2);
disp('第三个操作点的 B3=');disp(B2);

x0(1:4)=[10 1 pi/10 0.1]';
[t,open_fuzzy]=ode45('open_fuzzy_module',0:0.01:30, x0);
[t,open_affine]=ode45('open_affine_module',0:0.01:30, x0);
figure(1);
subplot(2,2,1);
plot(t, open_fuzzy(:,1),'r',t,open_affine(:,1),'b');
xlabel('时间(秒)');
ylabel('x1');
legend('fuzzy','affine');
subplot(2,2,2);
plot(t, open_fuzzy(:,2),'r',t,open_affine(:,2),'b');
xlabel('时间(秒)');
ylabel('x2');
```

```
legend('fuzzy','affine');
subplot(2,2,3);
plot(t, open_fuzzy(:,3),'r',t,open_affine(:,3),'b');
xlabel('时间(秒)');
ylabel('x3');
legend('fuzzy','affine');
subplot(2,2,4);
plot(t, open_fuzzy(:,4),'r',t,open_affine(:,4),'b');
xlabel('时间(秒)');
ylabel('x4');
legend('fuzzy','affine');

x0(1:4)=[10 1 pi/10 0.1]';
p1=[-2.7534 -1.5464 -1.6355 -2.7453];
p2=[-1.5324 -2.4353 -2.6961 -2.6343];
p3=[-2.6594 -1.6445 -2.3595 -3.2659];
K0=place(A0,B0,p1);K1=place(A1,B1,p2);K2=place(A2,B2,p3);
[t,close_fuzzy]=ode45('close_fuzzy_module',0:0.01:30, x0);
[t,close_affine]=ode45('close_affine_module',0:0.01:30, x0);
figure(2);
subplot(2,2,1);
plot(t, close_fuzzy(:,1),'r',t,close_affine(:,1),'b');
xlabel('时间(秒)');
ylabel('x1');
legend('fuzzy','affine');
subplot(2,2,2);
plot(t, close_fuzzy(:,2),'r',t,close_affine(:,2),'b');
xlabel('时间(秒)');
ylabel('x2');
legend('fuzzy','affine');
subplot(2,2,3);
plot(t, close_fuzzy(:,3),'r',t,close_affine(:,3),'b');
xlabel('时间(秒)');
ylabel('x3');
legend('fuzzy','affine');
subplot(2,2,4);
plot(t, close_fuzzy(:,4),'r',t,close_affine(:,4),'b');
xlabel('时间(秒)');
ylabel('x4');
legend('fuzzy','affine');
```

open_fuzzy_module.m

```
function xdot = open_fuzzy_module(t, x);
global A0 A1 A2 B0 B1 B2 u M m;
% if(x(3)<=0.25)
%     h0=1;
% elseif(0.25<x(3)<=0.75)
%     h0=-2*x(3)+1.5;
% else
%     h0=0;
% end
h0=sin(x(3))/x(3);
h1=(1-h0)/4;
h2=3*h1;
xdot=h0*(A0*x+B0*u)+h1*(A1*x+B1*u)+h2*(A2*x+B2*u);
```

close_fuzzy_module.m

```
function xdot = close_fuzzy_module(t, x);
global A0 A1 A2 B0 B1 B2 u M m K0 K1 K2;
% if(x(3)<=0.25)
%     h0=1;
% elseif(0.25<x(3)<=0.75)
%     %h0=sin(x(3))/x(3);
%     h0=-2*x(3)+1.5;
% else
%     h0=0;
% end
h0=sin(x(3))/x(3);
h1=(1-h0)/4;
h2=3*h1;
u0=u-h0*K0*x-h1*K1*x-h2*K2*x;
xdot=h0*(A0*x+B0*u0)+h1*(A1*x+B1*u0)+h2*(A2*x+B2*u0);
```

open_affine_module.m

```
function xdot = open_affine_module(t, x);
global M m l u g;
f=[x(2);(m*l*x(4)^2*sin(x(3))+m*g*sin(x(3))*cos(x(3)))/(M+m*sin(x(3))*sin(x
(3)));x(4);-(m*l*x(4)^2*sin(x(3))*cos(x(3))+(M+m)*g*sin(x(3)))/l/(M+m*sin(x
(3))*sin(x(3)))];
gx=1/(M+m*sin(x(3))*sin(x(3)))*[0;1;0;-cos(x(3))/l];
xdot=f+gx*u;
```

close_affine_module.m

```
function xdot = close_affine_module(t, x);
global M m l u g K0 K1 K2;
f=[x(2);(m*l*x(4)^2*sin(x(3))+m*g*sin(x(3))*cos(x(3)))/(M+m*sin(x(3))*sin(x
(3)));x(4);-(m*l*x(4)^2*sin(x(3))*cos(x(3))+(M+m)*g*sin(x(3)))/l/(M+m*sin(x
(3))*sin(x(3)))];
gx=1/(M+m*sin(x(3))*sin(x(3)))*[0;1;0;-cos(x(3))/l];
h0=M*(sin(x(3)))^2/(M+m*sin(x(3))*sin(x(3)));
h1=M*(cos(x(3)))^2/(M+m*sin(x(3))*sin(x(3)));
h2=m*(sin(x(3)))^2/(M+m*sin(x(3))*sin(x(3)));
xdot=f+gx*(u-h0*K0*x-h1*K1*x-h2*K2*x);
```