

《计算机先进控制》实验报告

实验一：基于离散状态观测器的反馈控制器设计

学 院：

专 业：

学生姓名：

学 号：

北京交通大学

实验内容：

1．离散状态观测器设计：

已知带观测器的反馈控制系统如图1所示。

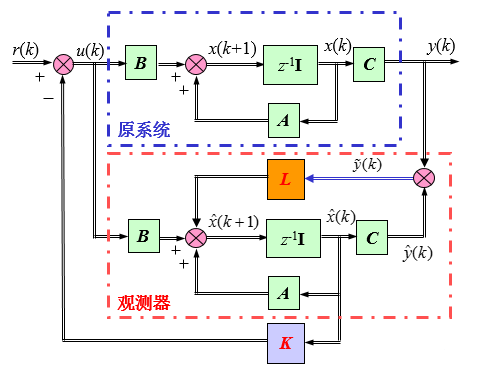
****

图1 带观测器的反馈控制系统

原系统的离散状态矩阵、输入矩阵和输出矩阵分别为

，，

为了得到良好的状态跟踪，设计状态观测器的极点分别为0.5, 0.6, 0.75, *T*0=0.1s。当输入*r*(*k*)为零，初始条件分别为和*x*(0)=0时，试模拟这个观测器，并在一幅图上画上*x*(*k*)和，再在另一幅图上画出*y*(*k*)和。

实验过程（Matlab仿真分析）：

程序如下：exp1.m

T0=0.1; %采样周期

A=[0.81 -0.23 -0.045 %状态矩阵

0.09 0.98 -0.0023

0.005 0.1 1];

B=[0.09 0.0047 0.00016]'; %输入矩阵

C=[1 3.5 3]; %输出矩阵

D=0; %传递矩阵

p=[0.5 0.6 0.75]'; %期望的观测极点

L=place(A',C',p)' %极点配置的评价增益(注：place指令比其它，如acker等，更适合极点配置分析和设计)

F\_ob=A-L\*C %观测系统矩阵

eig(F\_ob) %计算观测系统矩阵的特征值

x0=[1;-0.75;0.5] %初始条件

dtime=[0:T0:6]; %时间向量

u=zeros(1,length(dtime)); %输入零向量

G=ss(A,B,C,D,T0) %创建ss型(离散状态方程State Space)系统为LTI对象(线性时不变系统,Linear Time Invariant)

[y,dtime,x]=lsim(G,u,dtime,x0); %系统仿真获得y(k) (注：lsim模拟状态观测器)

disp('Simulate Observer') %显示

B\_ob=ss(F\_ob,L,C,D,T0) %创建系统为LTI对象

[y\_hat,dtime,x\_hat]=lsim(B\_ob,y,dtime); %仿真观测器

figure

plot(dtime,x,'o',dtime,x\_hat(:,1),'+',dtime,x\_hat(:,2),'x',...

dtime,x\_hat(:,3),'^'),grid %画系统和观测器状态

title('Observer design for Example 2.1')

xlabel('Time(s)') %定义x轴

ylabel('Amplitude') %定义y轴

text(0.6,0.75,'x\_1')

text(0.3,-0.5,'x\_2')

text(0.07,0.5,'x\_3')

legend('x\_1','x\_2','x\_3','x\_1hat','x\_2hat','x\_3hat') %不同曲线标识符

figure

plot(dtime,y,'o',dtime,y\_hat,'\*'),grid %画y和y-hat图

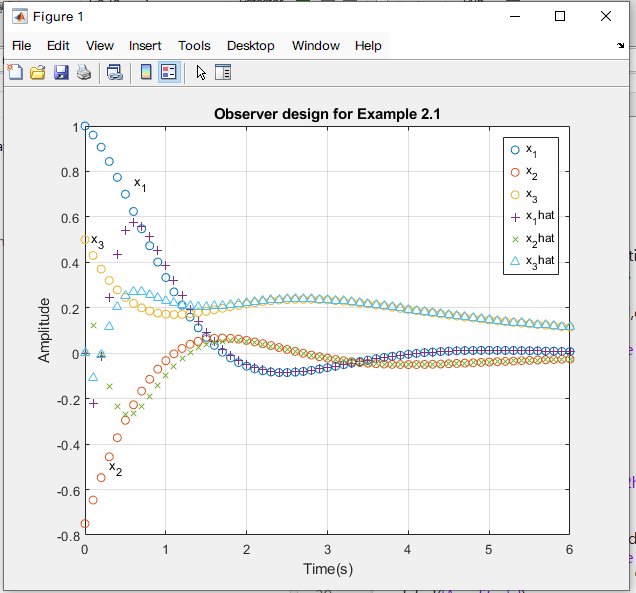
title('Observer design for Example 2.1') %标题

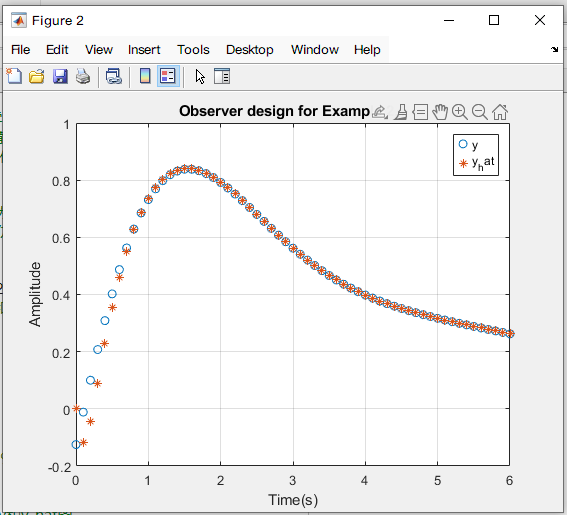
xlabel('Time(s)') %定义x轴

ylabel('Amplitude') %定义y轴

legend('y','y\_hat') %不同曲线标识符

仿真结果如图所示：





仿真结果如下：

>> exp1

L =

1.7737

-0.9762

0.8610

F\_ob =

-0.9637 -6.4380 -5.3661

1.0662 4.3966 2.9262

-0.8560 -2.9134 -1.5829

ans =

0.5000

0.6000

0.7500

x0 =

1.0000

-0.7500

0.5000

G =

A =

x1 x2 x3

x1 0.81 -0.23 -0.045

x2 0.09 0.98 -0.0023

x3 0.005 0.1 1

B =

u1

x1 0.09

x2 0.0047

x3 0.00016

C =

x1 x2 x3

y1 1 3.5 3

D =

u1

y1 0

Sample time: 0.1 seconds

Discrete-time state-space model.

Model Properties

Simulate Observer

B\_ob =

A =

x1 x2 x3

x1 -0.9637 -6.438 -5.366

x2 1.066 4.397 2.926

x3 -0.856 -2.913 -1.583

B =

u1

x1 1.774

x2 -0.9762

x3 0.861

C =

x1 x2 x3

y1 1 3.5 3

D =

u1

y1 0

Sample time: 0.1 seconds

Discrete-time state-space model.

Model Properties

2.带状态观测器的反馈控制器设计：

结合图1系统，确定观测器-控制器传递函数的零、极点并画出它们的频率响应（Bode图），以及闭环系统的状态跟踪图和阶跃响应。设期望的系统极点分别为0.75, 0.88, 0.9, *T*0=0.1s。初始条件分别为，，。

程序如下：

T0=0.1; %采样周期

A=[0.81 -0.23 -0.045 %状态矩阵

0.09 0.98 -0.0023

0.005 0.1 1];

B=[0.09 0.0047 0.00016]'; %输入矩阵

C=[1 3.5 3]; %输出矩阵

D=0;

Gss=ss(A,B,C,D,T0) %创建系统的ss模型

p\_s=[0.75,0.88,0.9]'; %期望的系统极点

K=place(A,B,p\_s) %极点配置的控制器增益

p\_o=[0.5 0.6 0.75]'; %期望的观测器极点

L=place(A',C',p\_o)' %观测器增益

%build observer-controller

F\_oc=A-B\*K-L\*C %观测器-控制器系统矩阵

Goc=ss(F\_oc,-L,-K,0,T0) %创建观测器-控制器为LTI

Goc\_poles=pole(Goc) %控制器极点全为实数

Goc\_zeros=tzero(Goc) %控制器零点

figure

title('Example 2.2:Bode Diagram') %标题

bode(Goc),grid %画控制器的bode图

Gpc=Gss\*Goc; %控制器和被控对象

Gcl=feedback(Gpc,1,-1) %具有非单位的DC增益的闭环系统

cl\_loop\_poles=pole(Gcl) %闭环系统极点

lfg=dcgain(Gcl)

N=1/lfg %归一化常数

T\_ref=N\*Gcl %归一化闭环系统

t=[0:T0:4]; %时间向量

r=0\*t; %零参考输入

z0=[1 -0.75 0.5 0 0 0] %初始状态向量

[y,t,z]=lsim(T\_ref,r,t,z0); %初始状态仿真

figure

plot(t,z(:,1:3),'o',t,z(:,4),'+',t,z(:,5),'x',t,z(:,6),'^'),grid %画系统和观测器状态

title('Example 2.2: Observer-Controller Design') %标题

xlabel('Time(s)') %定义x轴

ylabel('Amplitude') %定义y轴

text(0.4,0.85,'x\_1')

text(0.33,-0.5,'x\_2')

text(0.07,0.5,'x\_3')

legend('x\_1','x\_2','x\_3','x\_1hat','x\_2hat','x\_3hat') %不同曲线标识符

figure

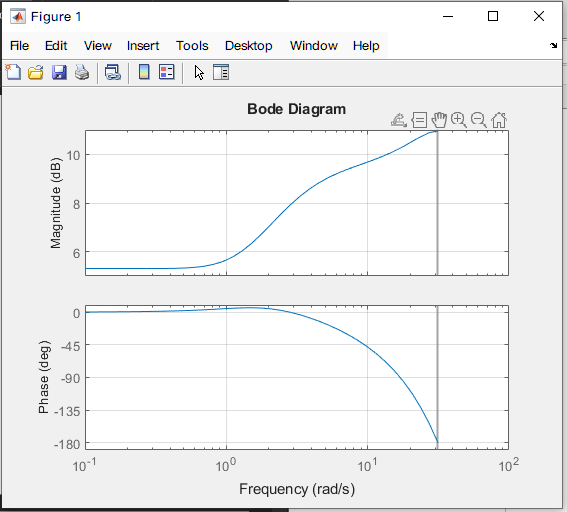
[ys,t,z]=step(T\_ref,4); %计算前4s的阶跃响应

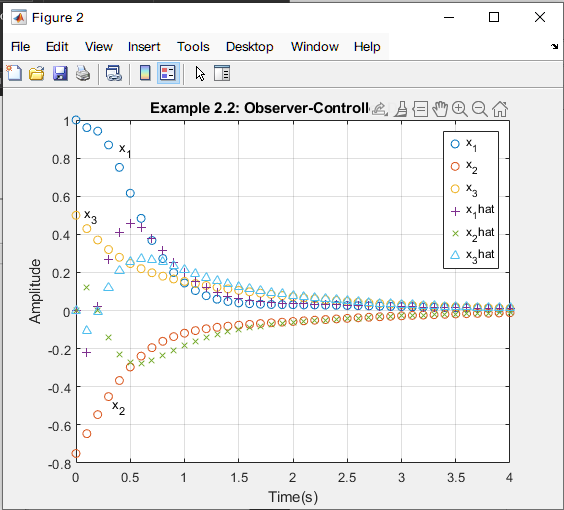
stem(t,ys,':\*');grid

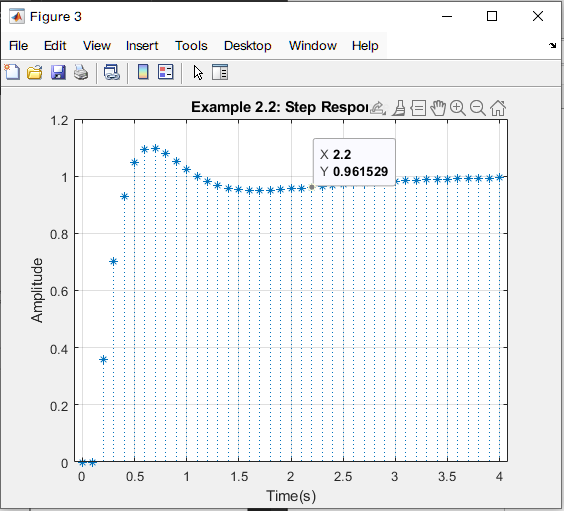
title('Example 2.2: Step Response') %标题

xlabel('Time(s)') %定义x轴

ylabel('Amplitude') %定义y轴







请在图3(c)中画出没有状态观测器时的输出曲线，以作对比。讨论为什么要进行观测器-控制器设计，有什么优点？

在程序中加入如下代码

figure

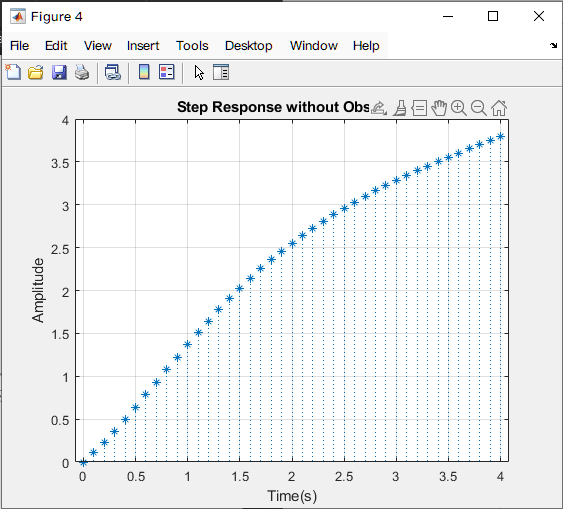
[y\_no\_observer,t]=step(Gss,4); % 没有观测器时的阶跃响应

stem(t,y\_no\_observer,':\*');grid

title('Step Response without Observer') % 标题

xlabel('Time(s)') % x轴

ylabel('Amplitude') % y轴



得到上图。为没有状态观测器时的输出曲线。观测器-控制器设计的主要优点是通过观测器估计不可测的状态变量，并使用这些估计值来生成控制器的反馈。在实际应用中，某些状态变量可能难以直接测量或昂贵，通过观测器设计可以解决这一问题，同时维持系统的稳定性和性能。