|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Student Name: | Hanlin Cai |  | Student Name: | 蔡汉霖 |
| Maynooth ID: | 20122161 |  | Fuzhou University ID: | 832002117 |

## EE211 MATLAB Assignment 3:

Personal Statement

Alternatives：**A MATLAB/Simulink environment**

The data set given to me by the TA is shown below:

A = 4; = 10; =25

All the Code & Picture were designed and created by myself,

and I never share with others.

Procedure 1

The figure of the dynamical system in this assign3 is shown below:

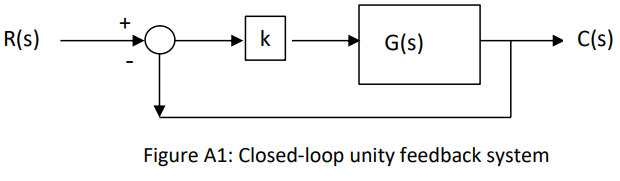


Fig-1

And its transfer function G(s) is given:

Where A = 4; = 10; =25

So, the transfer function is:

Firstly, we split the transfer function into 2 parts by defining an intermediate variable Z(s) as follows:

Then, converting to the time domain gives:

Setting the states as

Then we get the state equation as follows:

Giving:

So, we know that:

Procedure 2

Through Producer1, we have obtained that:

So, we can calculate the eigenvalues of the state space system:

That is：

So, the eigenvalues are:

Procedure 3

Through the producer1, we have got the transfer function:

Then, we can calculate the poles of the transfer function:

So, we can easily get the poles:

Just the same as the eigenvalues calculated in Procedure 2:

Procedure 4

Due to the fact that all the eigenvalues are:

And the poles of the dynamical system are:

**Hence, the dynamical system is asymptotically stable and critically damped.**

Procedure 5

Through Procedure1, we have obtained that:

So, we use the SIMULINK for simulation:

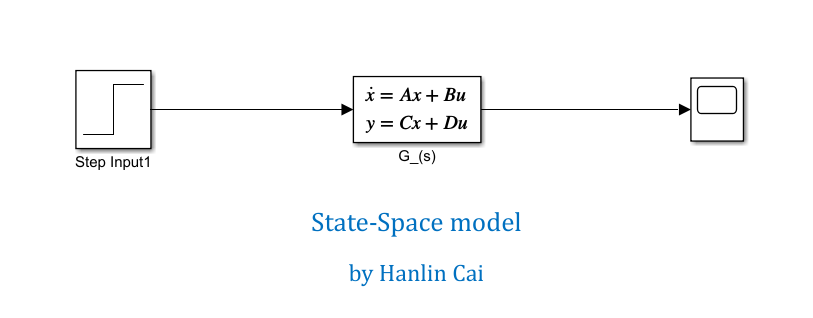


Fig-2

Then we can get the Scope of the transfer function:

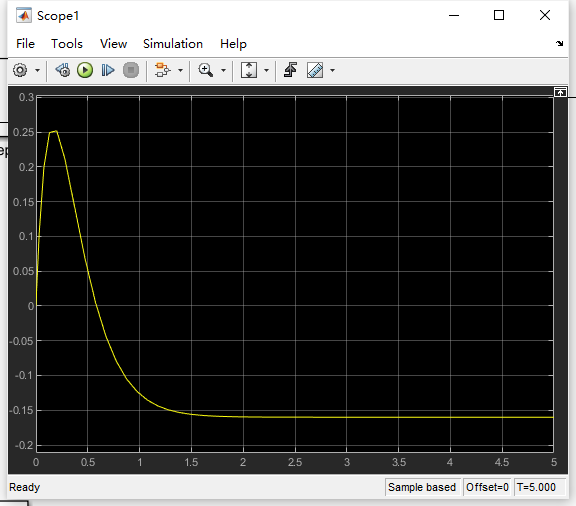


Fig-3

When we change the system input:

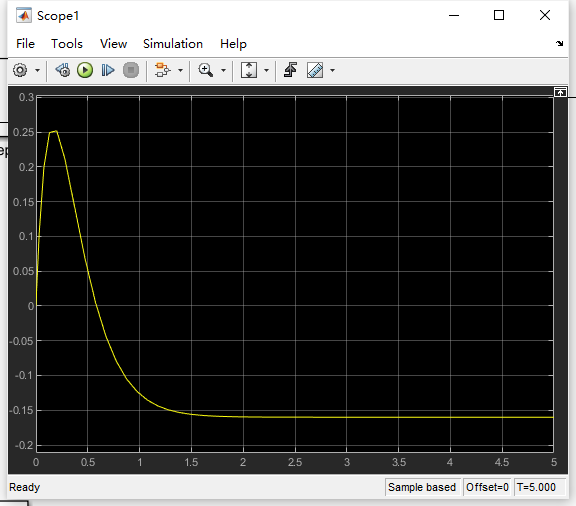


Fig-4

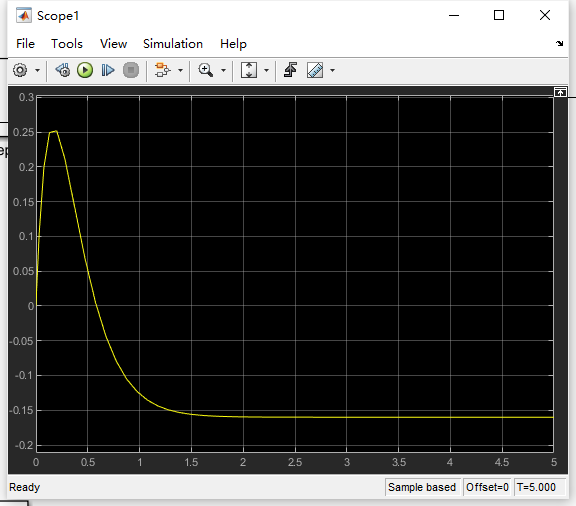


Fig-5

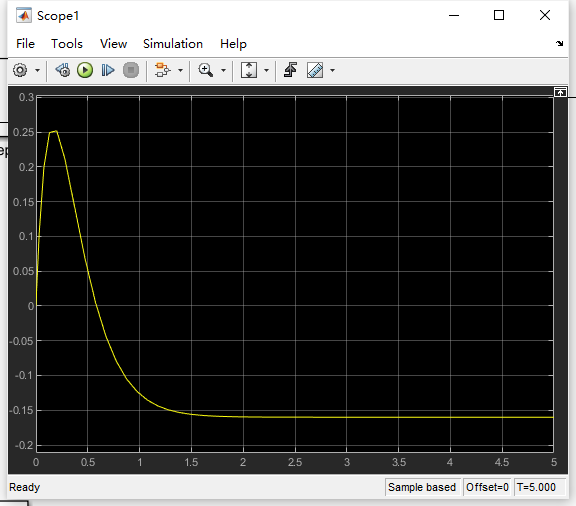


Fig-6

**So, we can conclude that stability is independent of the system input.**

Procedure 6

We have known that the figure of the dynamical system:

图示

描述已自动生成

Fig-7

Then, we can use SIMULINK for system simulation:

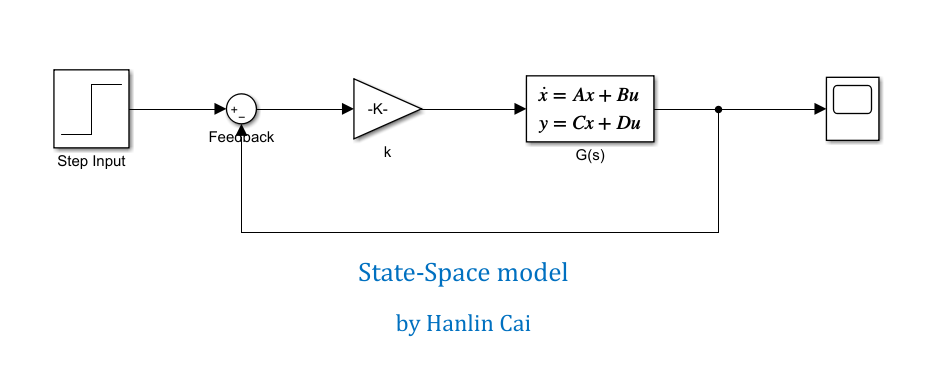


Fig-8

When the

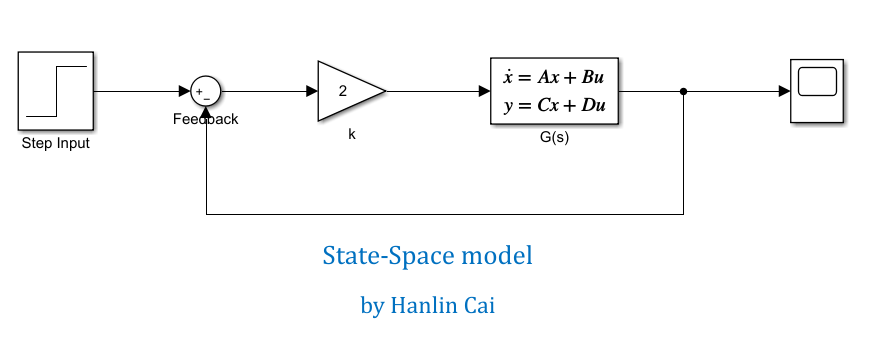


Fig-9

We can get the Scope of the system, as Fig-10 shown, the system is asymptotically stable:

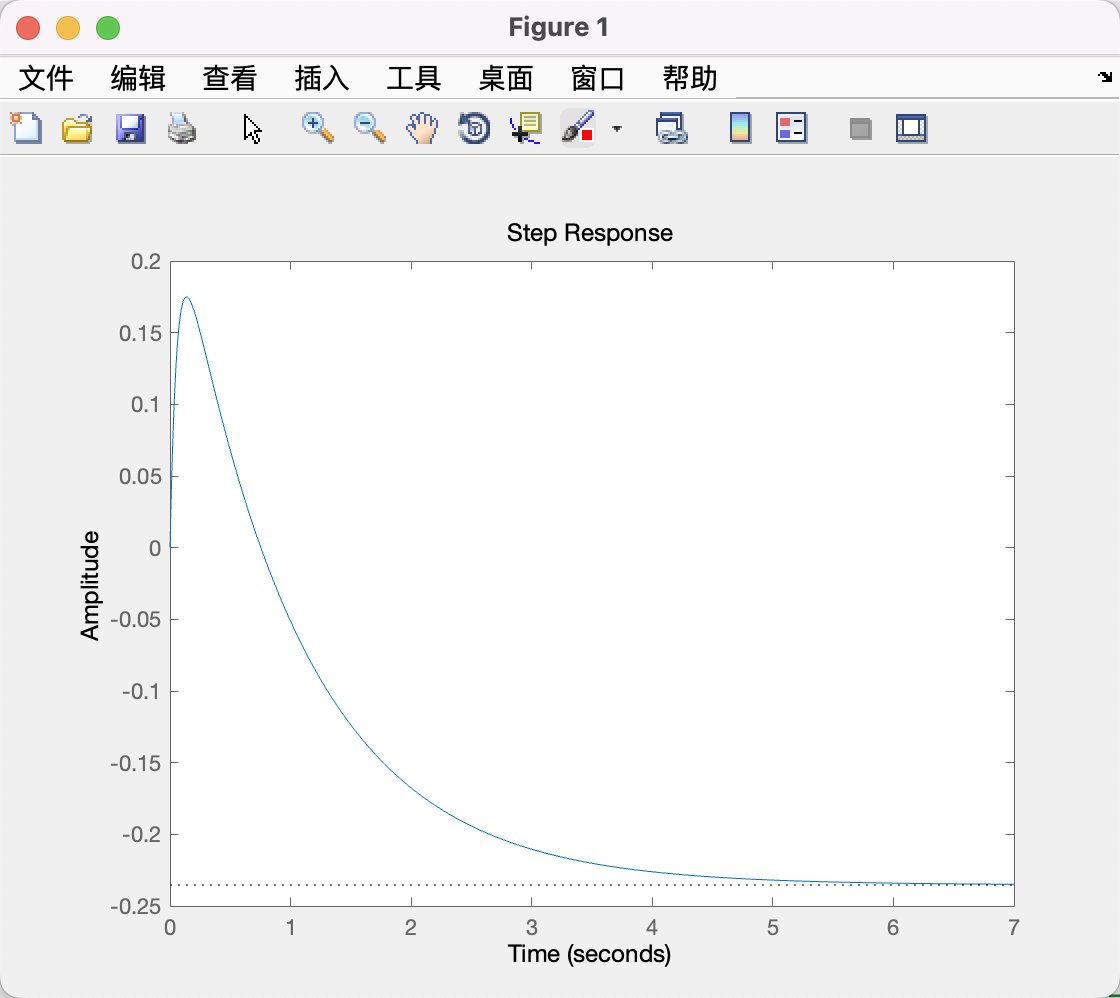


Fig-10

**As Fig-10 shown, the dynamical system is asymptotically stable.**

Procedure 7

We have obtained the SIMULINK simulation for the system:

图示

描述已自动生成

Fig-11

When we change the value of k, we can get different Scope:

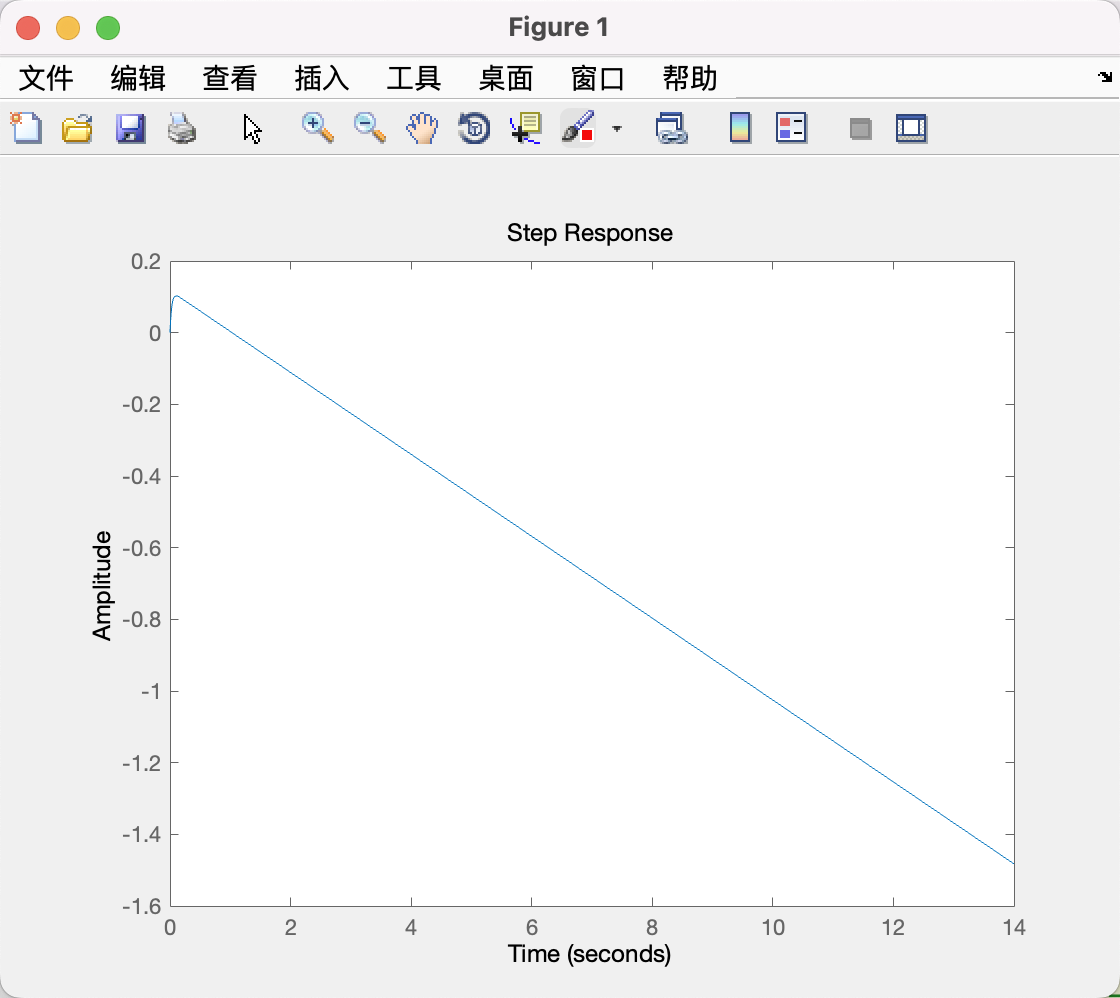


Fig-12 Already Overdamped

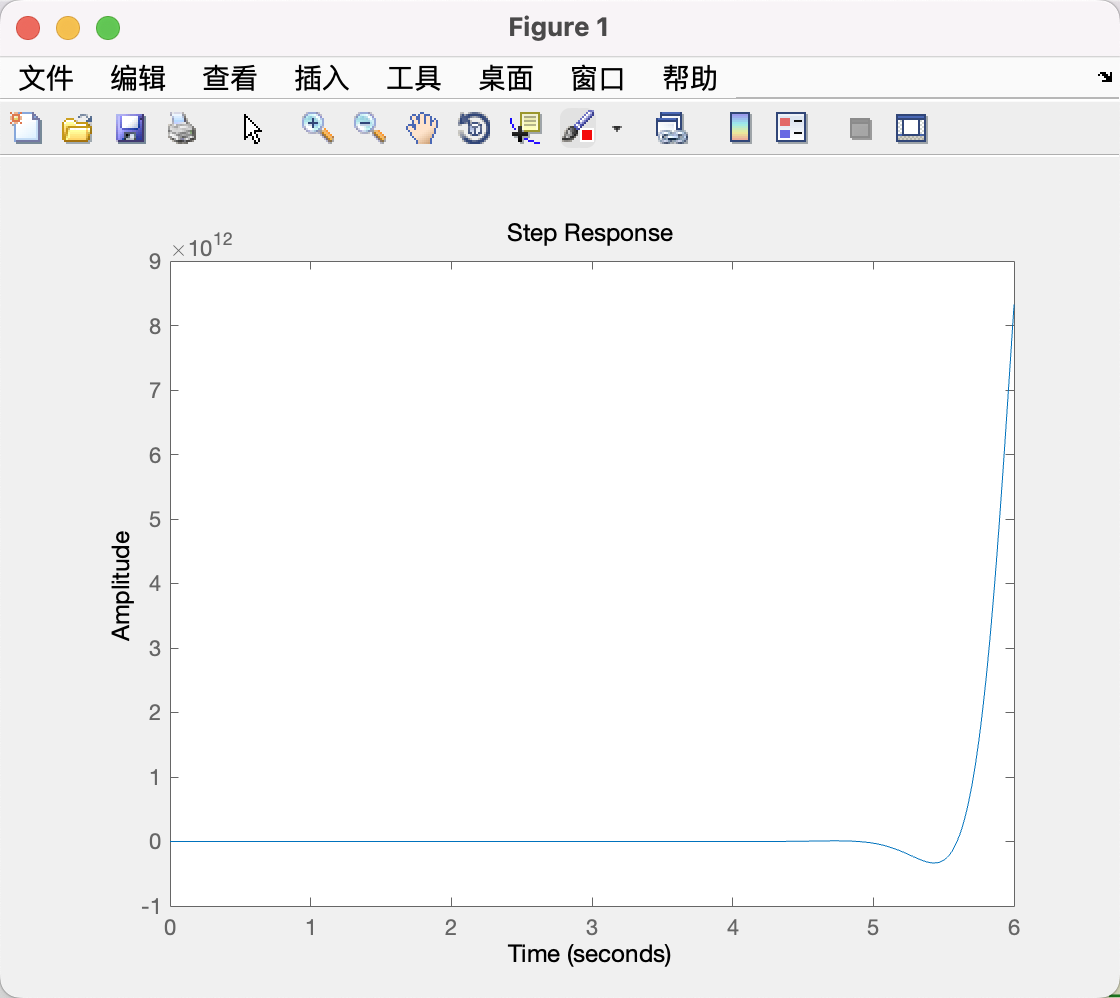


Fig-13 Overdamped

图形用户界面, 应用程序

描述已自动生成

Fig-14 Marginally Stable

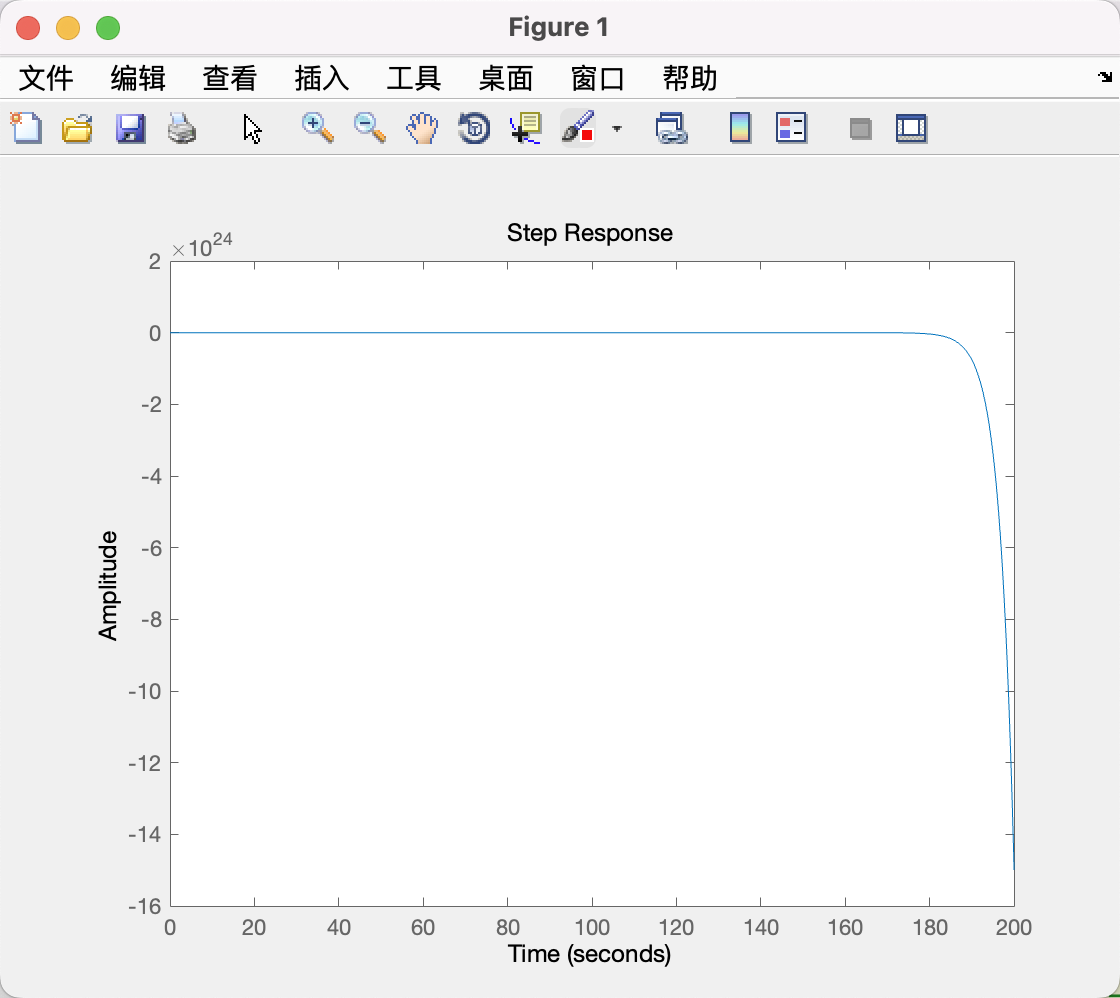


Fig-15 Overdamped

**Comment:**

**As the Fig10-15 shown, when we change different value of k, the transient response will change. When , the system is asymptotically stable.**

**In this scope, when k become larger, the transient response will become weaker, and the system tends to be overdamped. (When , the system will already become overdamped.)**

**Meanwhile, when k become smaller, the transient response will become stronger, and the system tends to be overdamped, too. (When , the system is marginally stable.)**

**Overall, when , the system is asymptotically stable. And when , the system is unstable.**

Procedure 8 & Procedure 9

We have let , and the figure of the closed-loop transfer function (CLTF) for the system is given as Fig-16. So, we can calculate the poles of the system by MATLAB.

手机屏幕截图

中度可信度描述已自动生成

Fig-16

Through MATLAB Code shown below, through ‘root’ function, we can get the poles:

Because , so the system is asymptotically stable.

And through the ‘feedback’ function, we can get the state-space representation for the CLTF for the system, as Fig-17 shown.

表格

低可信度描述已自动生成

Fig-17

So, the state-space representation for the CLTF for the system is:

And through the ‘eig’ function, we can get the eigenvalues of this system:

Due to the fact that:

So, we can draw the conclusion that:

**The dynamical system is asymptotically stable.**

**MATLAB Code:**

%% Assign3 Hanlin Cai 832002117

% The dataset given to me by the TA is: A = 4; B1 = 10; B2 =25

% Statement: All the Code and Pic were created by myself, and I never share with others.

%% Q4-Q5

% 建立系统状态空间表达式模型 State-Space Modelling

A = [0 1 ; -25 -10];

B = [0;1];

C = [-4 4];

D = 0;

G = ss(A,B,C,D); % Continuous-time state-space model.

% step(G,5) % using a unit step input

% 系统多项式形式的传递函数模型 sys1

sys1 = tf(G); % Continuous-time transfer function.

% 系统零-极点形式的传递函数模型 sys2

sys2 = zpk(sys1); % Continuous-time zero/pole/gain model.

% 系统传递函数多项式形式的分子、分母多项式系数

[num ,den] = ss2tf(A,B,C,D); % Numerator and denominator polynomial coefficients

% 系统零-极点: z为零点，p为极点

[z,p] = ss2zp(A,B,C,D); % z is the zero point, and p is the pole.

%% Q8

% calculate the poles of the closed loop system.

syms s;

sys11 = s^2 + 10\*s + 25;

% root(sys11,s);

poles = solve(sys11,s); % poles = -5 & -5

%% Q9

% determine a state-space representation for the closed-loop transfer function (CLTF) for the system in Fig.A1 \

% and calculate the eigenvalues of this system.

k = 2; % when k = 2, the system is asymptotically stable

sign = -1;

SYS = feedback(G,k,sign);

SYS1 = tf(SYS); % Continuous-time transfer function.

eigvalues = eig(SYS); % when k = 2, eigvalues = -1 & -17

%% EE211 Assignment3 MATLAB code by Hanlin Cai

A summary of what I gained in this Assignment3

Through this assignment, I get familiar with how to simulate a complex model by **MATLAB/Simulink**. Although this assignment was very difficult for me at the beginning, I finally finished it well through independent study and continuous trying.

Acknowledgement

I would like to thank my professor, **Zhicong Chen**, and my TA, **Honghui Chen.** I’m really appreciate for your grateful patience, hard-work, and wisdom! Without your help, I could not finish this difficult assignment3 well. Thank you so much!