**控制系統設計Project2**

**組員**

**機械系 蔡旻霖 E14051075**

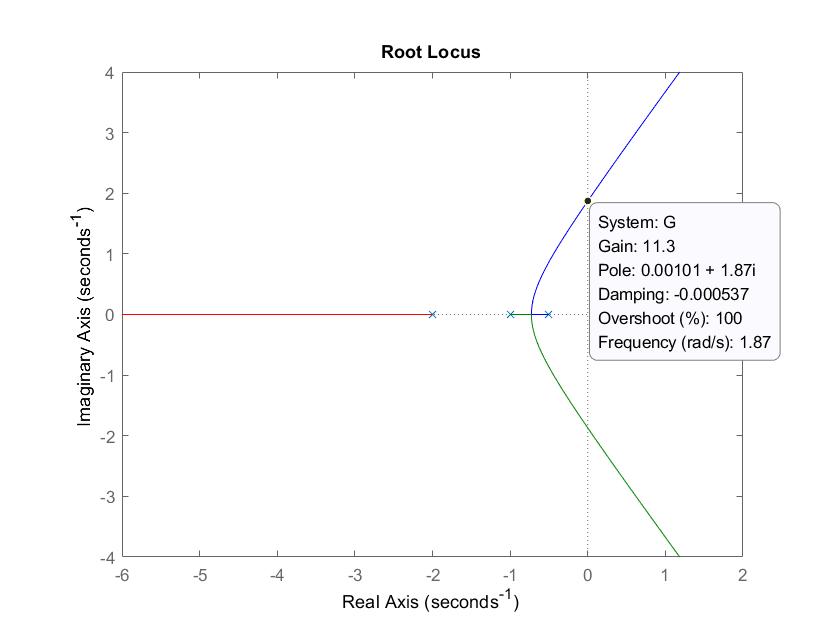
**機械系 林俊佑 E14056499**

**Part1**

**(Q1)**

P控制器可以使系統stable。其k值範圍由根軌跡圖可以知道為

0<k<11.3。



**(Q2)**

k=0.1；系統穩定，沒有overshoot，不振盪，但非常不準確。

一張含有 地圖, 文字, 螢幕擷取畫面 的圖片

自動產生的描述

k=8；系統存在，暫態時震盪幅度非常大，settling time很大。

一張含有 文字, 地圖 的圖片

自動產生的描述

k=20；系統不存在。

一張含有 文字, 地圖 的圖片

自動產生的描述

**(Q3)**

找不到。只使用P控制器過於簡單，無法達到設計要求。

**Prar2 紀錄**

**PD controller**

PD type

s = -1.25+2.1651i;

syms k;

z = -0.038

y = solve((1/(2\*s + 1)\*(s+1)\*(0.5\*s+1))\*k\*(s+z) + 1)

%get z = -0.0835 k=0.721784 帶入系統不合要求且本系統在damping ratio = 0.5 時唯一解 所以it is not the answer%

**D controller**

D

clear

close all

tc = 0.8

damp = 0.5

damp2 = damp^2 % s = -1.25 + 2.1651j

S\_first= -(1/tc)

S\_after= (1/tc)\*(((1-damp2)/damp2)^(1/2))

for i = 1:1:10

k=[1 0];

plant = tf(k,conv(conv([2 1],[1 1]),[0.5 1]))

plant\_new = feedback(plant\*i/10,1,-1)

figure(i)

rlocus(plant\_new)

end

%解出來變成PD control同上 不合要求 且此為唯一解 所以it is not the answer%

**PID controller**

PID finally

clear

close all

tc = 0.8

damp = 0.5

damp2 = damp^2 % s = -1.25 + 2.1651j

S\_first= -(1/tc)

S\_after= (1/tc)\*(((1-damp2)/damp2)^(1/2))

S=S\_first + i\*S\_after

L = S\*(2\*S+1)\*(S+1)\*(0.5\*S+1)

for i = 5.694:0.001:5.694 % a:b:c a->start point b->interval c->final point

%利用for迴圈大量列出各種i

%以i=k 來大量嘗試各種增益值得繪圖

syms a w

k=i

A=solve(17-5\*k+8\*k\*a==0,a);

S=solve(23+k\*(-3)-5\*k\*A+k\*A\*A+k\*w\*w == 0,w);

%將K帶入早已計算好的方程,以此帶入damping ratio 和 constant time 解出的S值,得到必然的Z1 和 Z2%

z1=A+S(1)\*(-1)^(0.5)

z2=A-S(1)\*(-1)^(0.5)

plant = tf([1 2\*eval(A) eval(z1\*z2)],conv(conv(conv([2 1],[1 1]),[0.5 1]),[1 0]))

plant\_new = feedback(plant\*k,1,-1)

figure(1)

hold on

step(plant\_new)

hold off

end

**%經過無數測試找出符合要求的解答**

**Phase lead controller**

d=0.5; %dampig ratio

ts=4; %settling time

tc=ts/5; %time constant

s1=-(1/tc)+(1/tc)\*(((1-d^2)/(d^2))^(0.5))\*1j;

s2=-(1/tc)-(1/tc)\*(((1-d^2)/(d^2))^(0.5))\*1j; %dominant poles

syms k z p

z=0.7;

f=1+k\*((s1+z)/(s1+p))\*(1/((2\*s1+1)\*(s1+1)\*(0.5\*s1+1)));

Re=real(f);

Im=imag(f);

[sol\_k, sol\_p] = vpasolve([Re== 0, Im ==0], [k,p]);

sol\_z=z;

k1=double(sol\_k);

z1=double(sol\_z);

p1=double(sol\_p);

Gc1=zpk([-z1],[-p1],k1);

Gc2=zpk([-0.6],[],1);

Gc3=zpk([],[0],1);

Gc4=series(Gc2,Gc3);

Gc=series(Gc1,Gc4);

Gp=zpk([],[-0.5 -1 -2],1);

G=series(Gp,Gc);

sys\_phaselead=feedback(G,1);

pole\_phaselead=pole(sys\_phaselead);

zero\_phaselead=zero(sys\_phaselead);

figure(1)

rlocus(G);

axis([-5 5 -10 10])

figure(2)

step(sys\_phaselead,20);axis([0 10 -1 1.5])

tf=20;

t=0:0.01:tf;

ramp=t;

figure(3)

lsim(sys\_phaselead,ramp,t);

**Prar3**

**(Q4) PID controller**

**(Q5)**

**(1)-2.19e-01 + 3.39e-01i**

**(2)-2.19e-01 - 3.39e-01i**

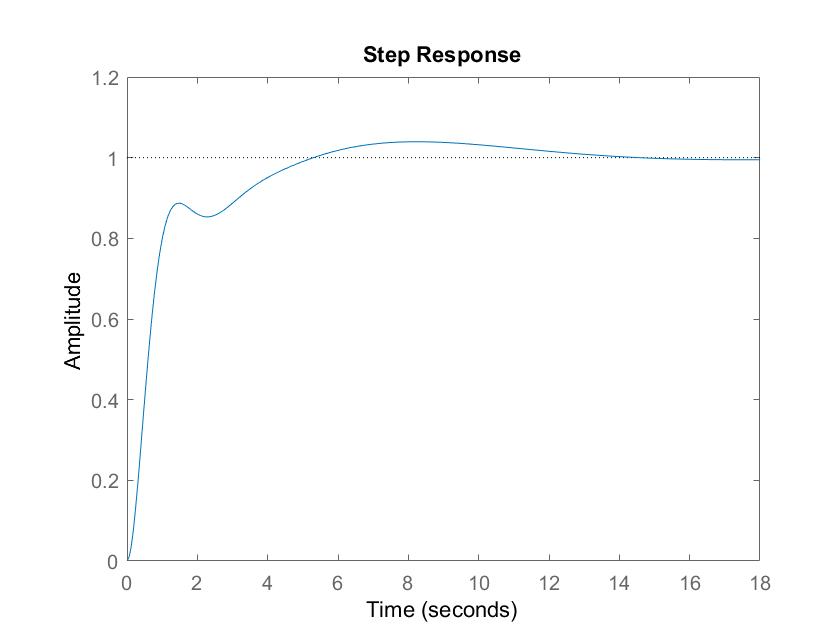
**(3)-1.53e+00 + 2.31e+00i**

**(4)-1.53e+00 - 2.31e+00i**

**It have those four pole, however, (3) and (4) is the dominate point because the response from (1) and (2) is too quick to control the system**

**and damping ratio is about 5.52e-01**

**(Q6)**

****

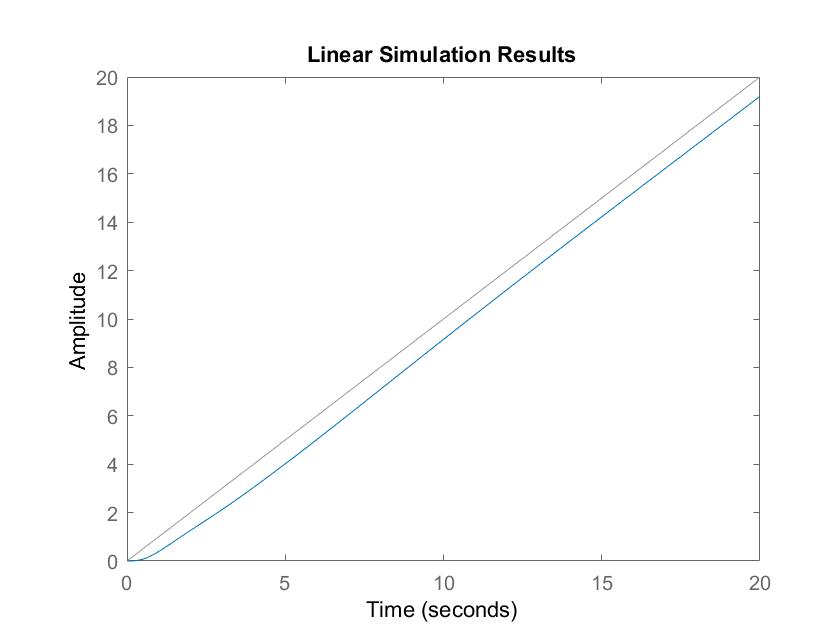
**Setting time is about 4 sec**

**Steady state error is about 0**

**Overshoot is about 0.04**

**(Q7)**

**e1 = 0.8**

****