

گزارش آزمایشگاه کنترل خطی (سروو موتور)

نیما حاجی حیدری (۹۹۲۳۰۱۷)، مهدی شاهینی (۹۹۲۳۰۴۰)

دانشگاه صنعتی امیرکبیر، دانشکده مهندسی برق

فهرست مطالب

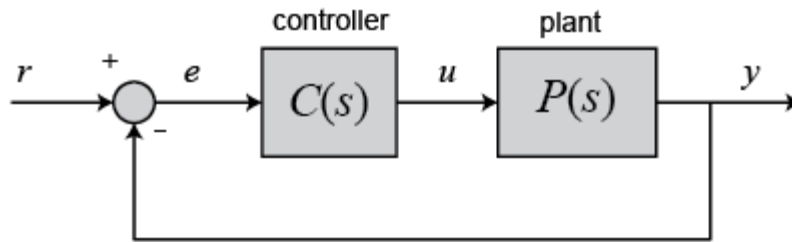
3 جبرانساز PD
3 مقدمه
3 محاسبات
7 جبرانساز Lead
7 مقدمه
7 محاسبات
8 کدنویسی
16 جبرانساز PI
16 مقدمه
17 طراحی:
17 کدنویسی
17 First section
17 Second Section

جبران ساز PD

هدف از این طراحی و آزمایش پیاده سازی یک کنترل کننده PD برای سیستم یک سروو موتور است. مطلوب است اگر

ورودی پله باشد، درصد بالا زدگی کمتر از ۵٪ و زمان نشست ۰.۱ ثانیه باشد.

مقدمه



به ترتیب خواهیم داشت:

$$r = u(t), R(s) = \frac{1}{s}$$

$$c(t) = K_p + K_d \frac{de(t)}{dt}, C(s) = K_p + K_d s$$

$$P(s) = \frac{0.129}{s(0.00213s + 0.0844)}$$

محاسبات

$$T_s = 0.1 \text{ s} = \frac{4}{\xi \omega_n} \rightarrow 40 = \xi \omega_n$$

$$P.O. = 4.7\% \rightarrow \xi = \frac{\sqrt{2}}{2}$$

$$\omega_n = \frac{40}{\sqrt{2}} = 40\sqrt{2} \rightarrow s_d = -40 \pm j(40\sqrt{2}) \sqrt{1 - \left(\frac{1}{2}\right)} = -40 \pm j40$$

$P(s) \rightarrow \text{poles: } 0 - 39.6244, \text{Zeros: Zero's is not limited.}$

$$\theta_z - (90^\circ + 135^\circ) = 180^\circ \rightarrow \theta_z = 45^\circ \quad z_d = -80$$

PD

Plant

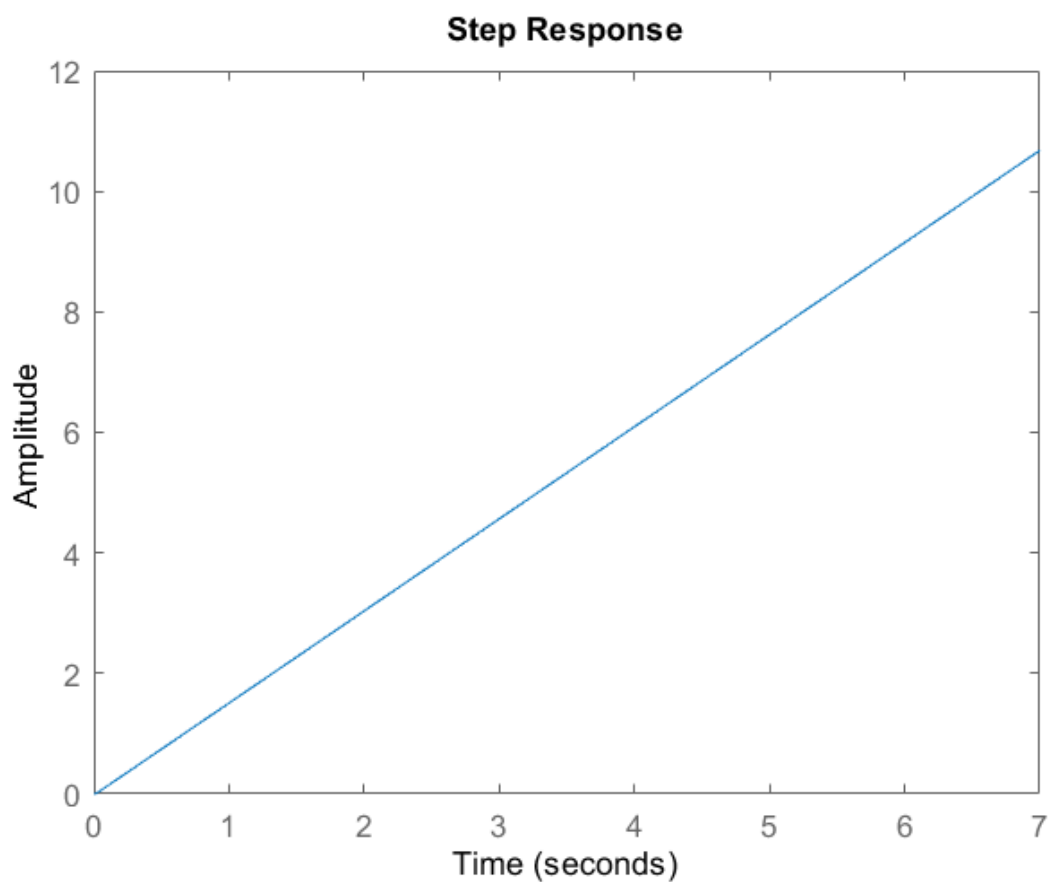
```
Gs = tf(0.129,[0.00113 0.0844 0])
```

Gs =

$$\frac{0.129}{0.00113 s^2 + 0.0844 s}$$

Continuous-time transfer function.

```
step(Gs)
```



Controller

```
k = 1
```

k = 1

```
Cs = tf([1 80], 1)
```

Cs =

$$s + 80$$

Continuous-time transfer function.

```
sysCsGs = series(k*Cs, Gs)
```

sysCsGs =

$$\frac{0.129 s + 10.32}{0.00113 s^2 + 0.0844 s}$$

Continuous-time transfer function.

Evaluation of k

```
% The num of C*G
num = [0.129 , 10.32];
% The den of C*G
den = [0.00213 , 0.0844 , 0];
% num of C*G for s = s_d
numval = polyval(num,-40+40*1j);
% den of C*G for s = s_d
denval = polyval(den,-40+40*1j);
value = denval/numval;
```

$$K = \frac{1}{|C(s) * G(s)|_{s_d}}$$

```
k = abs(value)
```

k = 0.6605

```
T = Cs * Gs
```

T =

$$\frac{0.129 s + 10.32}{0.00113 s^2 + 0.0844 s}$$

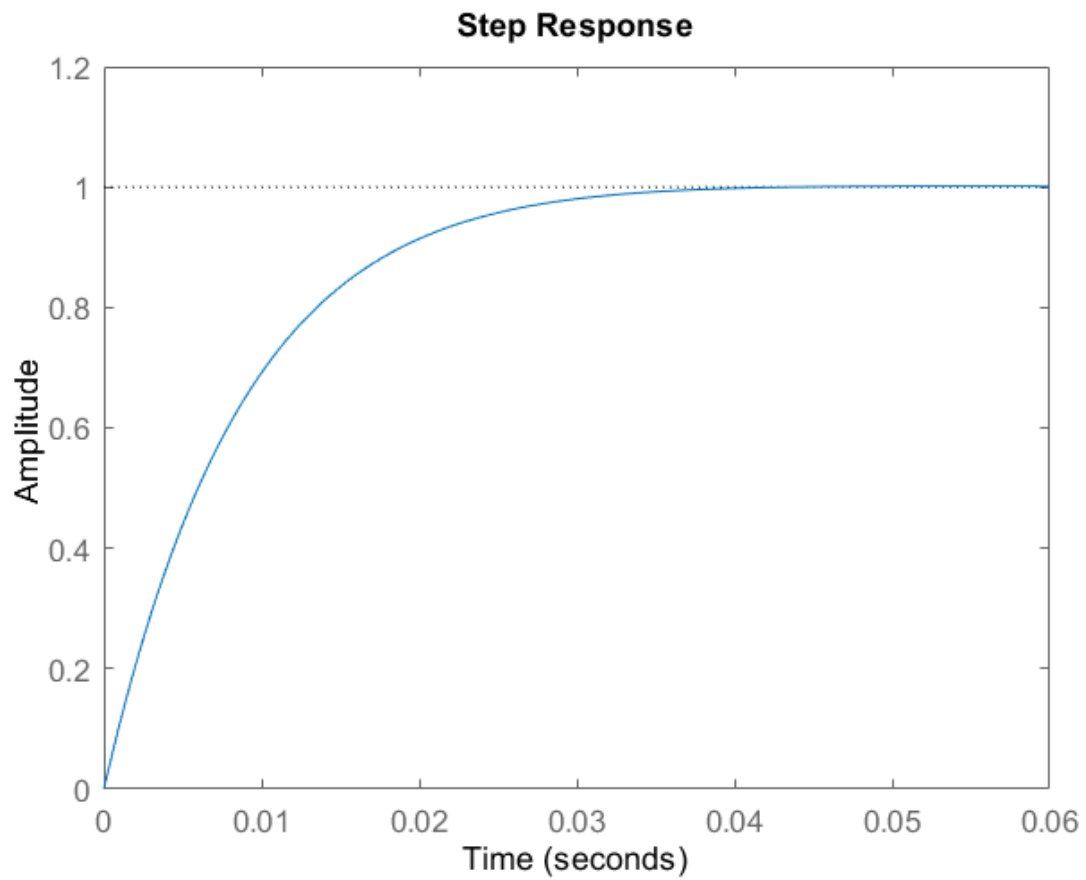
Continuous-time transfer function.

```
T_c = feedback(T, 1);
stepinfo(T_c)
```

```
ans = struct with fields:
    RiseTime: 0.0180
    SettlingTime: 0.0298
    SettlingMin: 0.9020
    SettlingMax: 1.0016
```

Overshoot: 0.1566
Undershoot: 0
Peak: 1.0016
PeakTime: 0.0541

step(T_c)

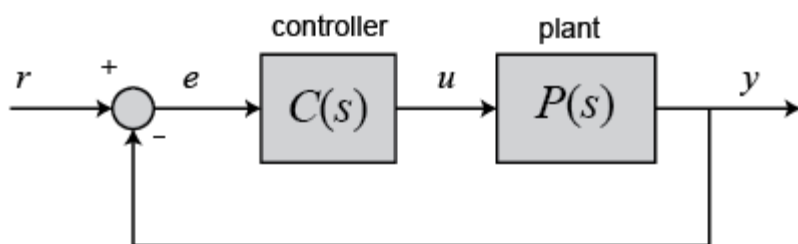


جبران ساز Lead

هدف از این طراحی و آزمایش پیاده سازی یک کنترل کننده Lead برای سیستم یک سروو موتور است. مطلوب است

اگر ورودی پله باشد، درصد بالا زدگی کمتر از ۳۰٪ و زمان نشست ۰.۱ ثانیه باشد.

مقدمه



به ترتیب خواهیم داشت:

$$r = u(t), R(s) = \frac{1}{s}$$

$$c(t) = K_p + K_d \frac{de(t)}{dt}, C(s) = K_p + K_d s$$

$$P(s) = \frac{0.129}{s(0.00213s + 0.0844)}$$

محاسبات

یک صفر را زیر یا کمی سمت چپ مقدار حقیقی قطب مطلوب قرار میدهم. سپس قطب را بر اساس شرط زاویه پیدا

میکنیم.

$$T_s = 0.1 \text{ s} = \frac{4}{\xi \omega_n} \rightarrow 40 = \xi \omega_n$$

$$P.O. = 30\% \rightarrow \xi = 0.36$$

$$\omega_n = 111.8 \rightarrow s_d = -40 \pm j104.4$$

$$P(s) \rightarrow \text{poles: } 0 - 39.6244, \text{Zeros: Zero's is not limited.}$$

$$90^\circ - \left(90^\circ + \tan^{-1} \frac{40}{104.4} + \tan^{-1} \frac{104.4}{-40 - p} \right) = 180^\circ \rightarrow p = -80$$

کدنویسی

Lead

```
% G(s)
sysG = tf([0.129], [0.00213 0.0844 0])
sysG =
```

```
      0.129
-----
0.00213 s^2 + 0.0844 s
```

Continuous-time transfer function.

```
% C(s)
sysGc = tf([1 40], [1 80])
sysGc =
```

```
      s + 40
-----
      s + 80
```

Continuous-time transfer function.

```
% C(s) * G(s)
sysGcG = series(sysG, sysGc)
sysGcG =
```

```
      0.129 s + 5.16
-----
0.00213 s^3 + 0.2548 s^2 + 6.752 s
```

Continuous-time transfer function.

```
% The num of C*G
num = [0.129 , 5.16];
% The den of C*G
den = [0.00213 , 0.2548 , 6.752 0];
% num of C*G for s = s_d
numval = polyval(num, -40+104.37*i);
% den of C*G for s = s_d
denval = polyval(den, -40+104.37*i);
value = denval/numval;
```


$$K = \frac{1}{|C(s) * G(s)|_{s_d}}$$

```
k = abs(value)
```

```
k = 206.2827
```

```
sysT1 = series(sysGcG, k)
```

```
sysT1 =
```

```

      26.61 s + 1064
-----
0.00213 s^3 + 0.2548 s^2 + 6.752 s

```

Continuous-time transfer function.

```
sysT2 = series(sysGcG, 0.95*k) %It has better performance!
```

```
sysT2 =
```

```

      25.28 s + 1011
-----
0.00213 s^3 + 0.2548 s^2 + 6.752 s

```

Continuous-time transfer function.

```
sysT = sysT2
```

```
sysT =
```

```

      25.28 s + 1011
-----
0.00213 s^3 + 0.2548 s^2 + 6.752 s

```

Continuous-time transfer function.

```
sysTc = feedback(sysT, 1)
```

```
sysTc =
```

```

      25.28 s + 1011
-----
0.00213 s^3 + 0.2548 s^2 + 32.03 s + 1011

```

Continuous-time transfer function.

```
stepinfo(sysTc)
```

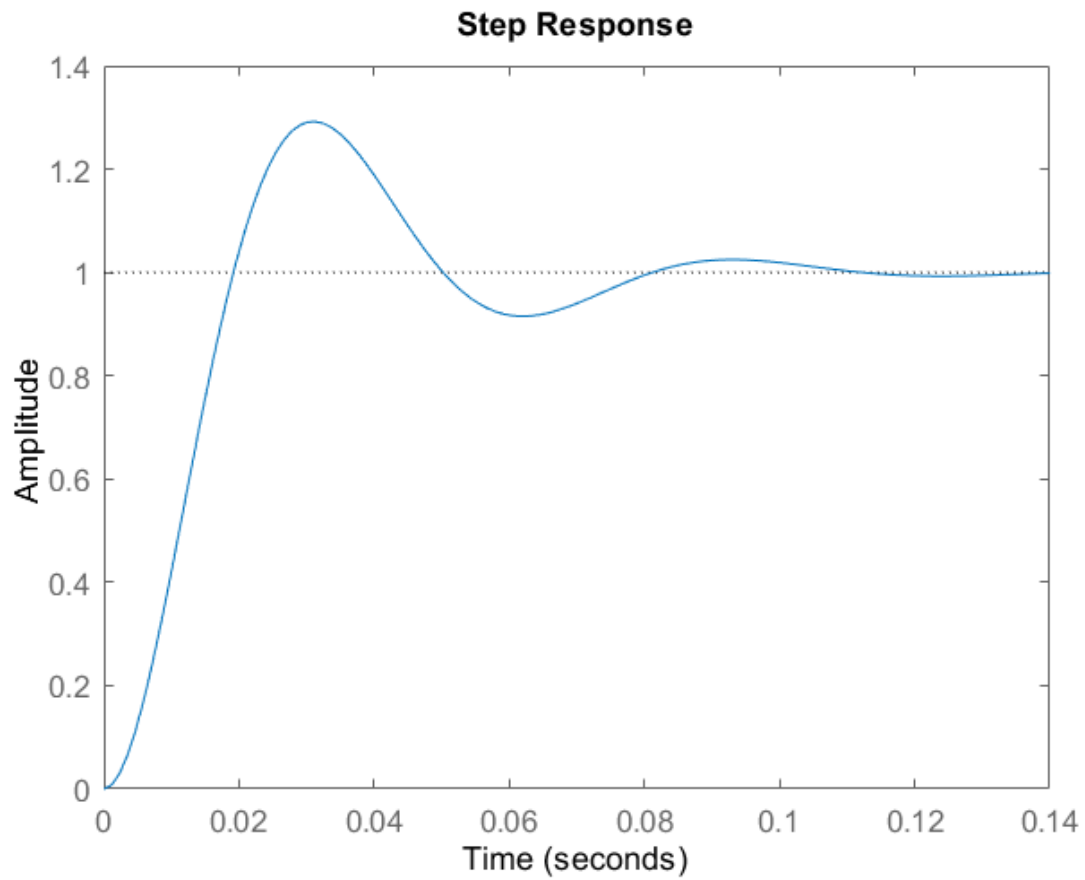
```
ans = struct with fields:
```

```

    RiseTime: 0.0130
  SettlingTime: 0.0993
   SettlingMin: 0.9018
   SettlingMax: 1.2922
    Overshoot: 29.2170
    Undershoot: 0
         Peak: 1.2922
    PeakTime: 0.0313

```

```
step(sysTc)
```



```
zero(sysTc)
```

```
ans = -40
```

```
pole(sysTc)
```

```
ans = 3×1 complex
```

```
102 ×
```

```
-0.3978 + 1.0133i
```

```
-0.3978 - 1.0133i
```

```
-0.4006 + 0.0000i
```

```
syspf = tf([1], [1 40])
```

```
syspf =
```

```
1
```

```
-----
```

```
s + 40
```

```
Continuous-time transfer function.
```

```
syseq = series(syspf, sysTc)
```

```
syseq =
```

```
25.28 s + 1011
```

```
-----
```

```
0.00213 s4 + 0.34 s3 + 42.22 s2 + 2292 s + 4.045e04
```

Continuous-time transfer function.

```
stepinfo(syseq)
```

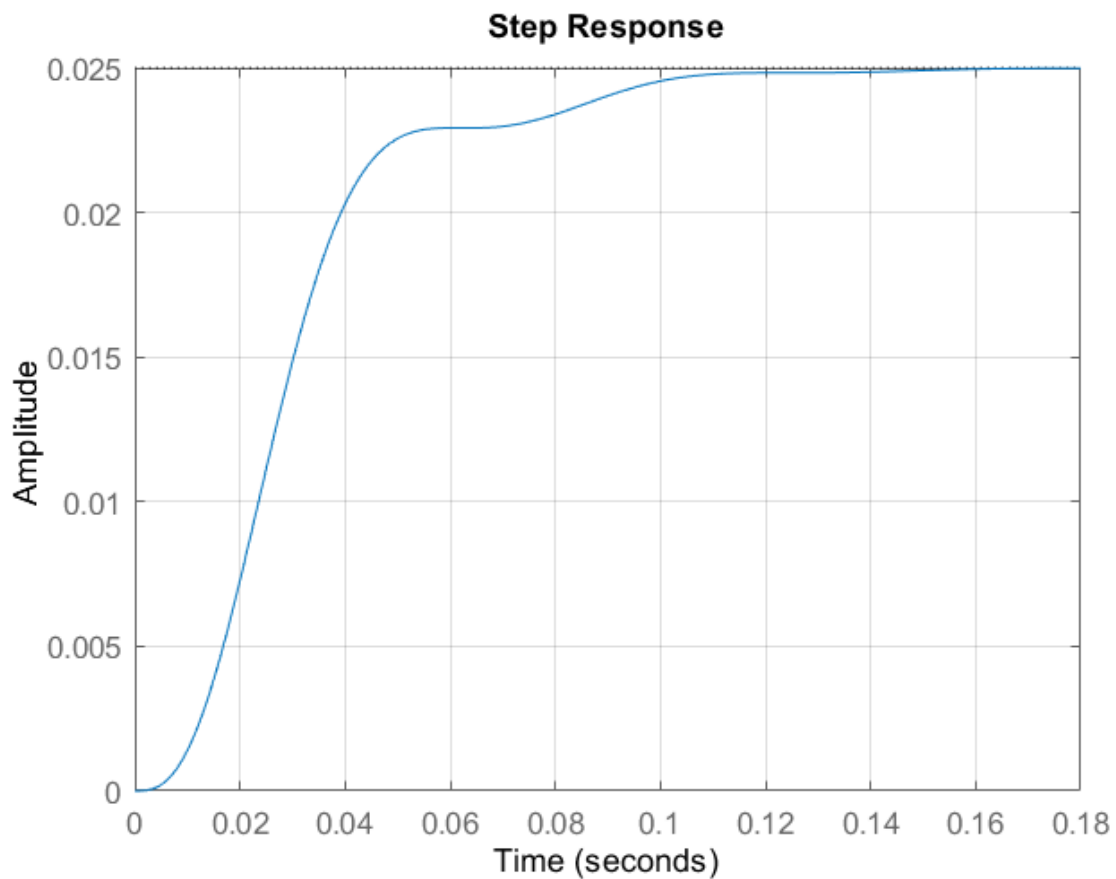
```
ans = struct with fields:
```

```
    RiseTime: 0.0369  
    SettlingTime: 0.0990  
    SettlingMin: 0.0225  
    SettlingMax: 0.0250  
    Overshoot: 0  
    Undershoot: 0  
        Peak: 0.0250  
    PeakTime: 0.2072
```

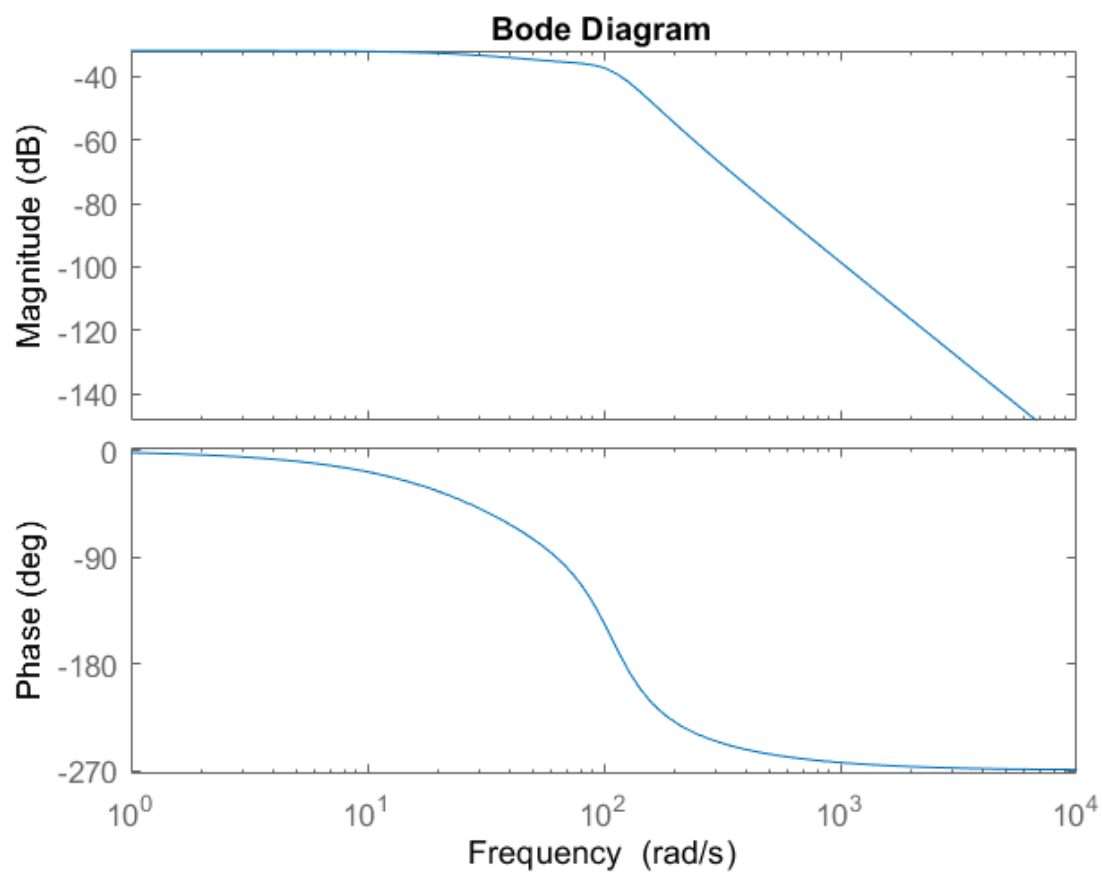
```
step(syseq)
```

```
grid on
```

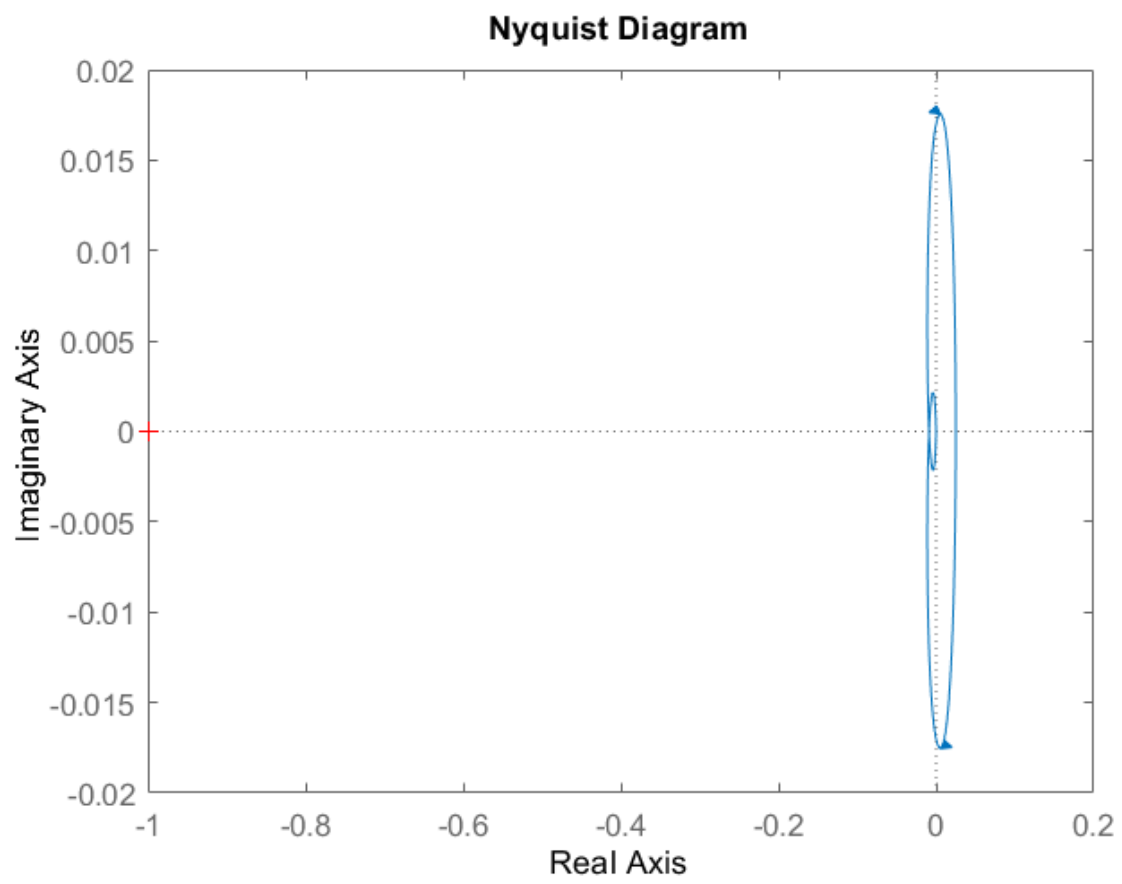
```
grid minor
```



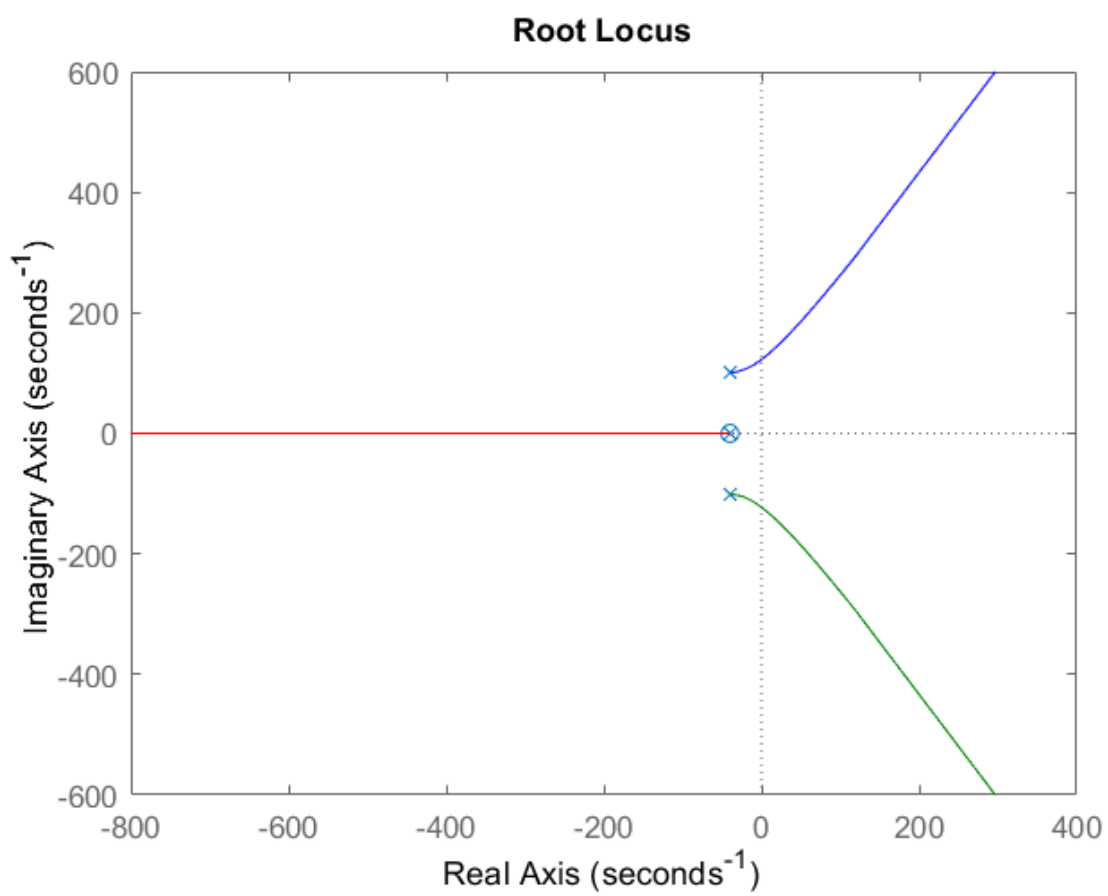
```
bode(syseq)
```



```
nyquist(syseq)
```



```
rlocus(syseq)
```



جبرانساز PI

هدف از این طراحی و آزمایش پیاده سازی یک کنترل کننده PI برای سیستم یک سروو موتور است. ورودی سیستم

شیب است! مطلوب در این آزمایش متغیر است و متناسب با انتخاب مقادیر، میتوان به خطاهای ثابت متفاوتی رسید.

پاسخ سوالات (مطابق نتایجی که در ادامه بدست آمد)

چرایی صفر در ۰.۱-؟

صفر به گونه‌ای نزدیک قطب انتخاب میشود که مکان در محل قطب‌های مطلوب چندان عوض نشود. در واقع شرط زاویه

در محل قطب‌های مطلوب کماکان تقریباً بیش از ۵ درجه تغییر نکند. (مطابق دیدگاه جناب آقای پروفیسور طالبی)

در نظر گرفتن شرایط گذرا؟!!

جبران ساز انتگرالی در بهبود پاسخ حالت دائم کاربرد دارد. به همین خاطر تاثیر محسوسی در پاسخ حالت گذرا ندارد.

فلذا اگر پاسخی گذرای با شرایط خاص مدنظر است میتوان با استفاده از مکان هندسی و بدست آوردن گین مورد نظر کنترلر را

تنظیم نمود.

مقدمه

میدانیم که سروو موتور یک سیستم نوع یک می‌باشد بدین معنی که در تابع تبدیل آن حداقل یک انتگرال گیر موجود

است. این بلوک باعث میشود که خطای حالت ماندگار به ورودی پله در حالت جبران نشده، مقداری متناهی باشد. از سوی دیگر

میدانیم که یک کنترل کننده ی PI به صورت زیر بیان می‌شود:

$$G_c(s) = K_p + \frac{K_I}{s}$$

بنابراین یک انتگرال گیر دیگر در تابع تبدیل حلقه باز سیستم جبران شده ظاهر خواهد شد و سیستم را به نوع دو تبدیل

میکند. این موضوع باعث میشود که خطای ماندگار به ورودی پله صفر گردد. به همین علت در این آزمایش، از ورودی شیب به

عنوان مرجع استفاده میگردد و تلاش می شود که خطای حالت ماندگار آن مقداری متناهی و ثابت باشد. (k_p)

طراحی:

در فرضیات سوال گفته می شود که صفر جبرانساز PI باید در -0.1 قرار گیرد. میتوان تابع تبدیل جبران ساز را به صورت

زیر بازنویسی کرد:

$$G_c(s) = K_c \times \frac{s+z}{s}$$

پس بنابراین، بسته به مشخصات پاسخ مطلوب میتوان مقادیری مختلف از K_c را تعیین کرد.

کدنویسی

Plant

First section

```
Gs=tf([0.129],[0.00113 0.0844 0])
```

Gs =

$$\frac{0.129}{0.00113 s^2 + 0.0844 s}$$

Continuous-time transfer function.

```
T=feedback(Gs,[1])
```

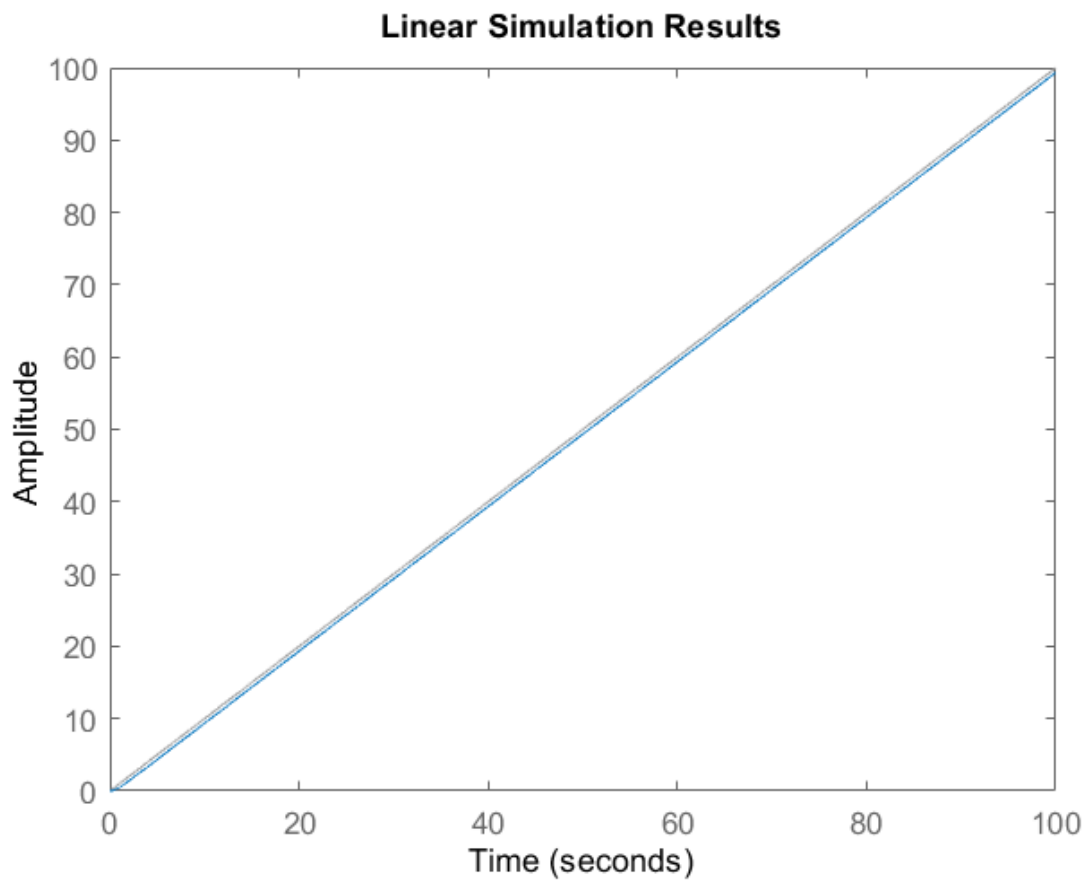
T =

$$\frac{0.129}{0.00113 s^2 + 0.0844 s + 0.129}$$

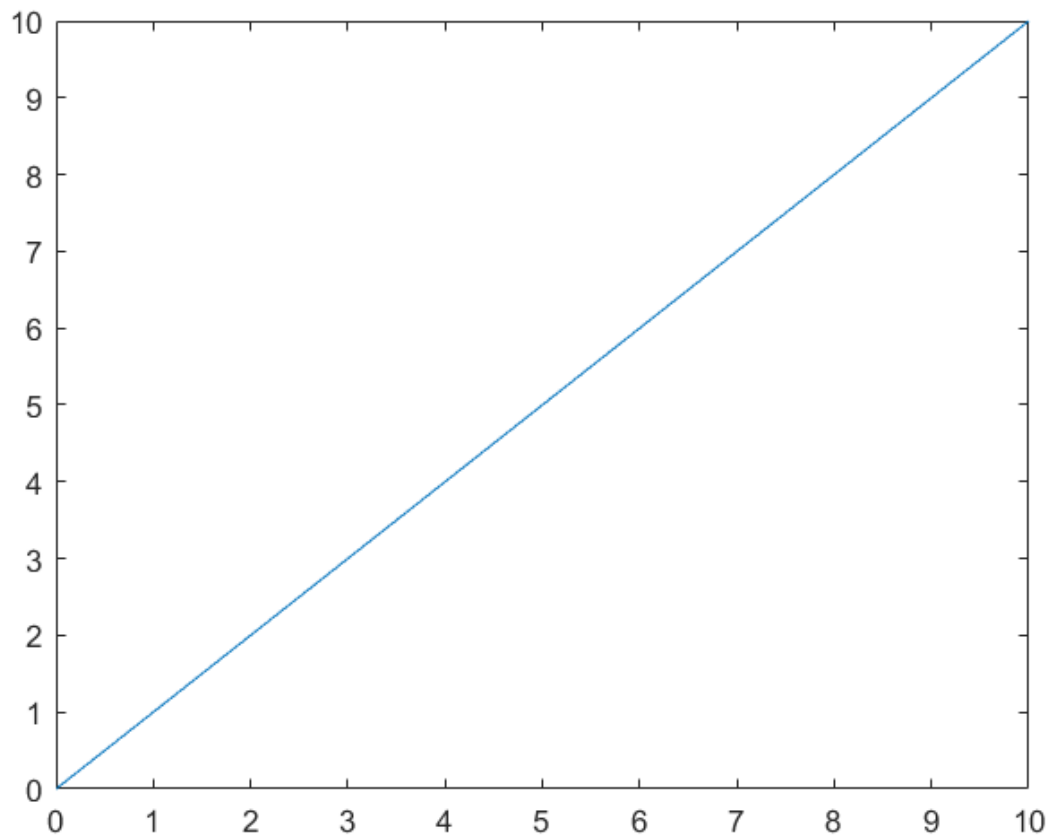
Continuous-time transfer function.

Second Section

```
lsim(T,[0:0.01:100],[0:0.01:100])
```



```
u=[0:0.01:10];  
t=[0:0.01:10];  
plot(t,u)
```



Contorller

```
Gc=tf([1 0.1],[1 0])
```

Gc =

$$\frac{s + 0.1}{s}$$

Continuous-time transfer function.

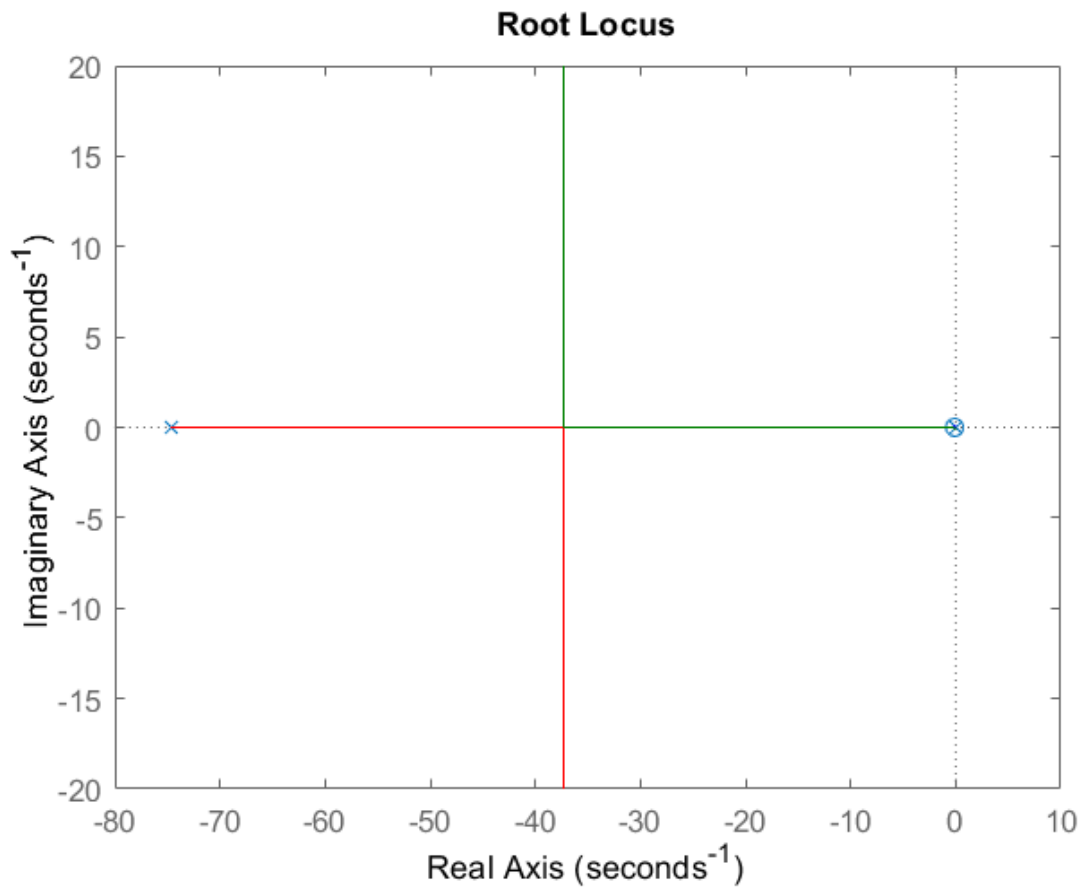
```
TFol=series(Gc,Gs)
```

TFol =

$$\frac{0.129 s + 0.0129}{0.00113 s^3 + 0.0844 s^2}$$

Continuous-time transfer function.

```
rlocus(TFol)
```



Error

```
numtf=[0 0 2.58 0.258];
dentf=[0.00113 0.0844 0 0];
denE=dentf+numtf;
numE=dentf;
Etf=tf(numE,denE)
```

Etf =

$$\frac{0.00113 s^3 + 0.0844 s^2}{0.00113 s^3 + 0.0844 s^2 + 2.58 s + 0.258}$$

Continuous-time transfer function.

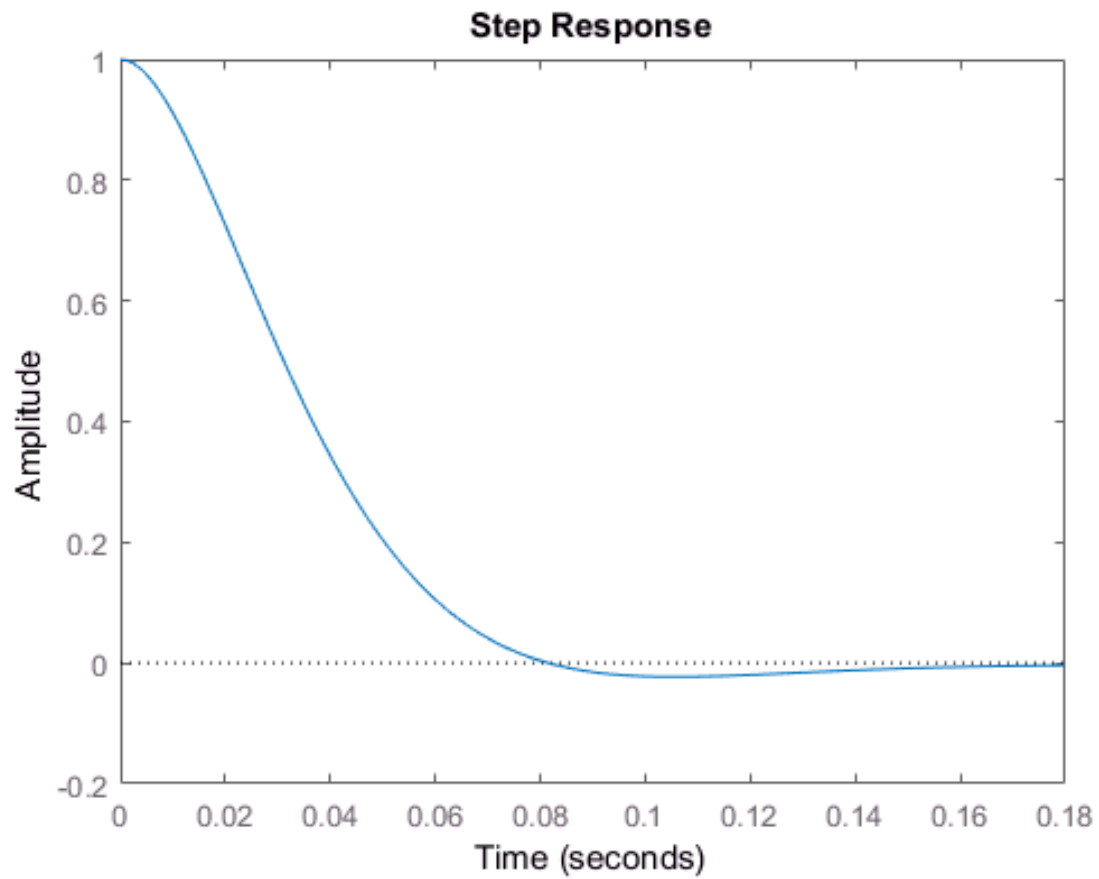
```
stepinfo(Etf)
```

ans =

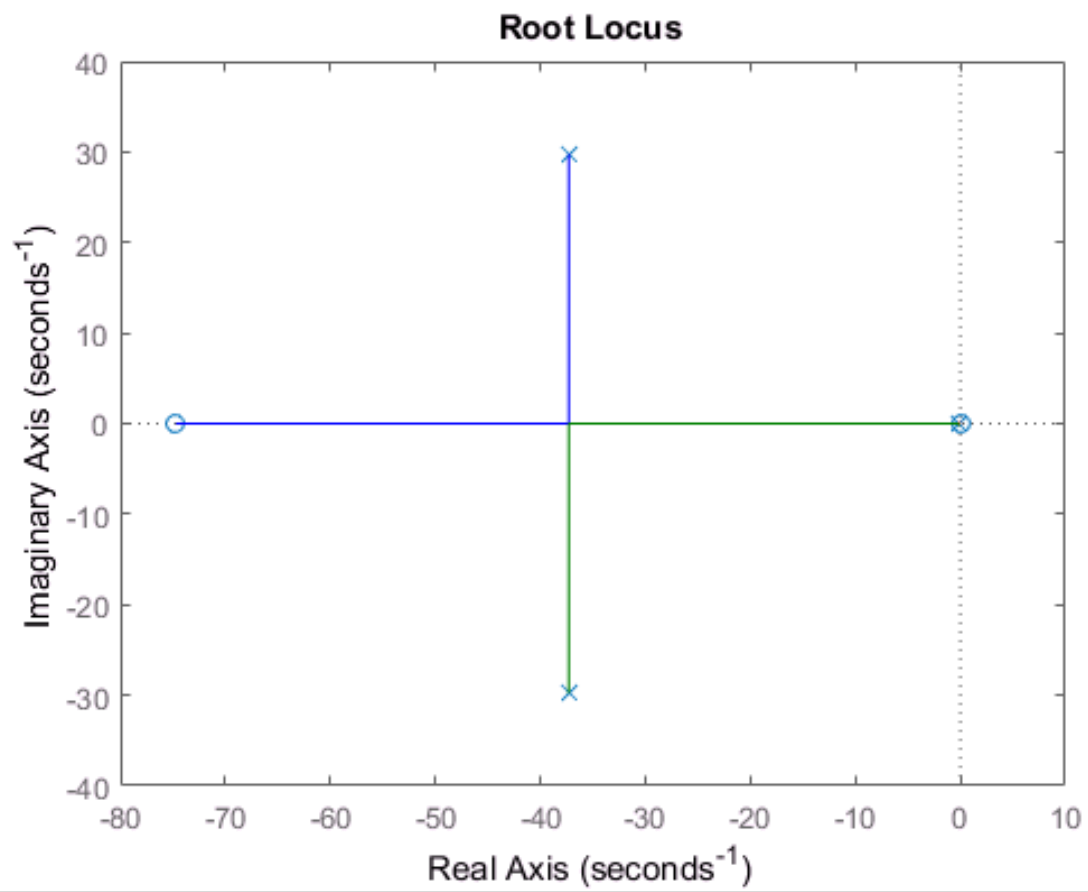
```
RiseTime: 0.0500
SettlingTime: 0.1189
SettlingMin: -0.0228
SettlingMax: 0.0918
Overshoot: 9.0072e+17
Undershoot: 2.0527e+16
```

Peak: 1
PeakTime: 0

step(Etf)



rlocus(Etf)



```
rad2deg(0.6)
```

ans = 34.3775

