Michael Bocamazo 2015/04/06

$\boxed{PS\ 10}$ $x = u(t)$

1. $\dot{y} + y = x \longrightarrow y(t) = (1 - e^{-t})\,u(t)$

$sY(s) + Y(s) = \frac{1}{s}$ ← L.T. of ~~unit~~ step function

$Y(s)(1+s) = \frac{1}{s}$

$Y(s) = \left(\frac{1}{s}\right)\left(\frac{1}{1+s}\right) = \frac{A}{s} + \frac{B}{1+s}$

$\frac{1}{s}\cdot\frac{1}{1+s} = \frac{A}{s} + \frac{B}{1+s}$ ~~...~~ $\frac{1}{s}\cdot\frac{1}{1+s} = \frac{1}{s} + \frac{B}{1+s}$

$\frac{1}{1+s} = A + \frac{sB}{1+s}$ $\qquad \frac{1}{1+s} = 1 + \frac{sB}{1+s}$

$s = 0$ $\qquad 1 = s+1+sB$

$1 = A + 0,\ A=1$ $\qquad 0 = s+sB \quad B=-1$

$Y(s) = \frac{1}{s} - \frac{1}{1+s}$

$\mathcal{L}^{-1} \Downarrow$ ROC: $s>0 \qquad s>-1$

Intersection of ROCs

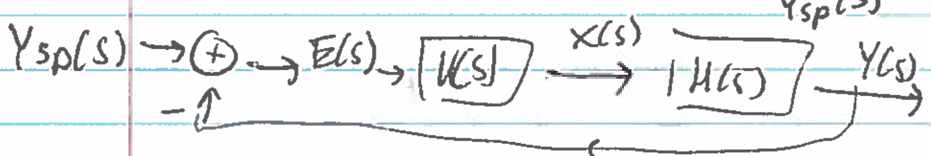
Overall ROC:

$Re\{s\} > 0$

$y(t) = u(t) - u(t)e^{-t}$

$\underline{y(t) = (1 - e^{-t})\,u(t)}$

2.A. Find DC gain of the system $\frac{Y(s)}{Y_{sp}(s)}$ $\quad K(s) = \frac{K_I}{s}$ for any $H(s)$



DC gain $= \lim_{s\to 0} H(s) \quad \frac{Y(s)}{Y_{sp}(s)} = \frac{\cancel{Y(s)K(s)H(s)}}{} \quad \frac{KH}{1+KH} = \frac{K_I/s\ H}{1 + \frac{K_I}{s}H}$

$K(Y_{sp} - HX) = X$

$\frac{X}{Y_{sp}} = \frac{K}{1+KH}$

$\frac{Y}{Y_{sp}} = \frac{KH}{1+KH}$

Using L'Hopital's rule,
the DC gain is $= 1$ as $s \to 0$
And doesn't depend on K_I

B. $H(s) = \dfrac{1/\tau}{s + 1/\tau}$ Find $\dfrac{Y(s)}{Y_{sp}(s)}$ Find poles assuming $K \gg 1/\tau$

Draw pole-zero diagram

$$\frac{Y(s)}{Y_{sp}(s)} = \frac{K_I/s \; H}{1 + K_I/s \; H} = \frac{\frac{K_I}{s}\left(\frac{1/\tau}{s+1/\tau}\right)}{1 + \frac{K_I}{s}\left(\frac{1/\tau}{s+1/\tau}\right)} \; \frac{(s)(s+1/\tau)}{(s)(s+1/\tau)}$$

$$\frac{Y(s)}{Y_{sp}(s)} = \frac{K_I/\tau}{(s)(s+1/\tau) + K_I/\tau} \qquad s^2 + s/\tau + K_I/\tau$$

$a = 1 \quad b = 1/\tau \quad c = K_I/\tau$

$$s = \frac{-1/\tau \pm \sqrt{1/\tau^2 - 4K_I/\tau}}{2}$$

dominates sqrt

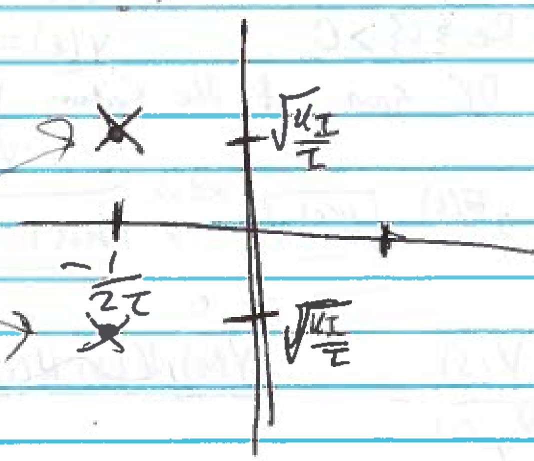
For poles

$$s = \frac{-1}{2\tau} \pm j\sqrt{\frac{K_I}{\tau}}$$

$K_I = 0$

Is a trivial zero

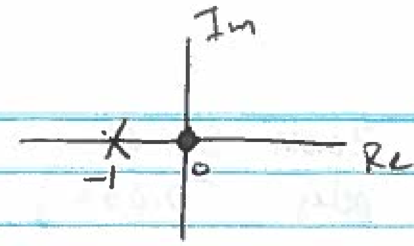
pole



and $K_I \gg 1/\tau$

it is ~~[illegible]~~ much greater and positive

3. A. $\frac{s}{s+1}$ | Zero $s=0$
pole $s=-1$



Step response: $\mathcal{L}^{-1}\left\{\frac{1}{s+1}\right\}$ $X(s)=\frac{1}{s}$

1^{st} order

$Y(s) = u(t)e^{-t}$

The response: similar to a high pass: for high s, $H=1$, low s, $H=0$. Makes sense that the step response is exponentially decays.

B. $\frac{s}{s^2+100s+1}$ 2^{nd} order

Acts as a band pass ~~around~~ between 10^{-2} Hz and 10^{2} Hz. There are poles at ≈ -100 and ≈ 0, and a zero at $s=0$. ↳ -99.99
↳ which should be ~~zero~~ $= -0.01$

The step response quickly rises to the input and then slowly decays.

C. zeros $=0$ 2^{nd} order

poles $= -0.5+0.87j$, $-0.5-0.87j$

This is a band pass filter with ~~an extremely small band between~~ nearly no band, as even at the ~~to~~ greatest point there is attenuation

2^{nd} order

D. Zeros: $s = 0$
poles: $-0.05 + j$, $-0.05 - j$

~~Also~~ Now, the band pass filter ~~as~~ has some gain around $Hz = 1$, gain ≈ 5, with steeper roll-off initially.
The step response oscillates, which is interesting (and slowly decays.)

2^{nd} order

E. The step response slowly oscillates around the input.
Zeros: $s = 0.01 + j$, $0.01 + j$
poles: $-0.01 + j$, $-0.01 - j$
The bode plot shows that there is essentially no gain, as expected, ~~just phase irregularity induced by the~~ The system induces ~~as~~ a phase shift of -2 rad/pi for $\omega > 1$.

2^{nd} order

F. $\dfrac{s^2 + 0.1s + 1}{s^2 + 0.11s + 1}$ The step response again slowly oscillates around the input
Zeros: $s = -0.05 \pm j$
poles: $s = -0.06 \pm j$
The bode plot shows that this is a band-stop system at ~~Hz~~ $\omega = 1$ and induces a phase shift similar to (E)

Plots on next pages

4. $H(s) = \dfrac{1}{s^2 - 0.01s + 1}$

proportional Control

B. $K(s) = K_p$

multiply by denominator

$$\frac{Y}{X} = \frac{\dfrac{K_p}{s^2 - 0.01s + 1}}{1 + \dfrac{K_p}{s^2 - 0.01s + 1}} = \frac{K_p}{s^2 - 0.01s + (1 + K_p)}$$

Poles always on right, unstable (see next pages)

C. $K(s) = K_I/s$

Integral Control

$$\frac{\dfrac{K_I}{s} \cdot \dfrac{1}{s^2 - 0.01s + 1}}{1 + \dfrac{K_I}{s} \cdot \dfrac{1}{s^2 - 0.01s + 1}} = \frac{K_I}{s^3 - 0.01s^2 + s + K_I}$$

doesn't stabilize

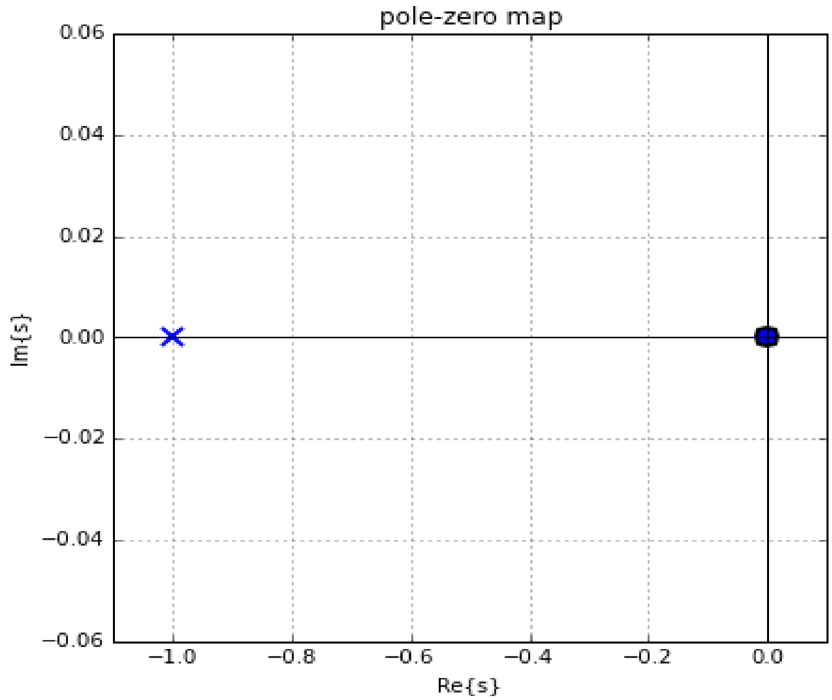
D. $K(s) = K_d s$

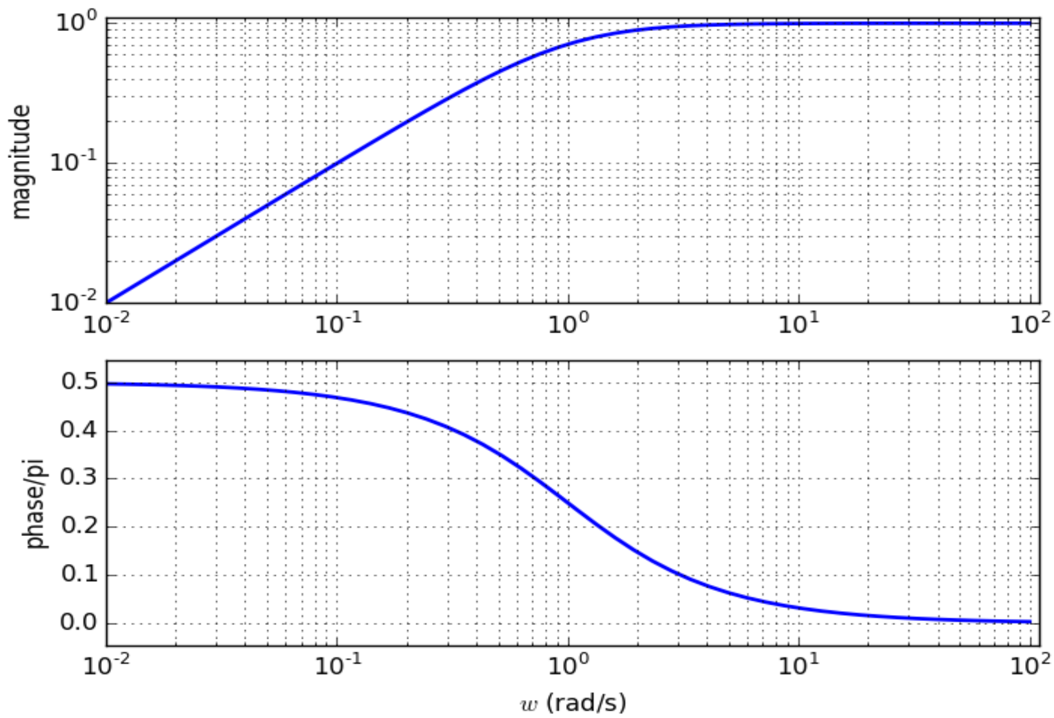
$$\frac{\dfrac{K_d s}{s^2 - 0.01s + 1}}{1 + \dfrac{K_d s}{s^2 - 0.01s + 1}} = \frac{K_d \cdot s}{s^2 + (K_d - 0.01)s + 1}$$

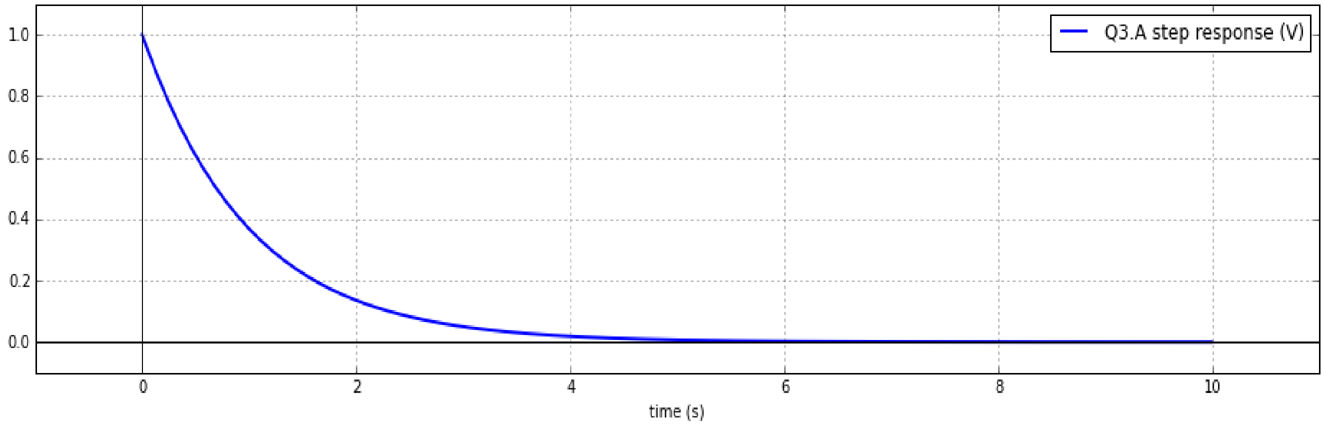
Does stabilize

Michael Bocamazo
SigSys PS10
Q3 Plots

3.A



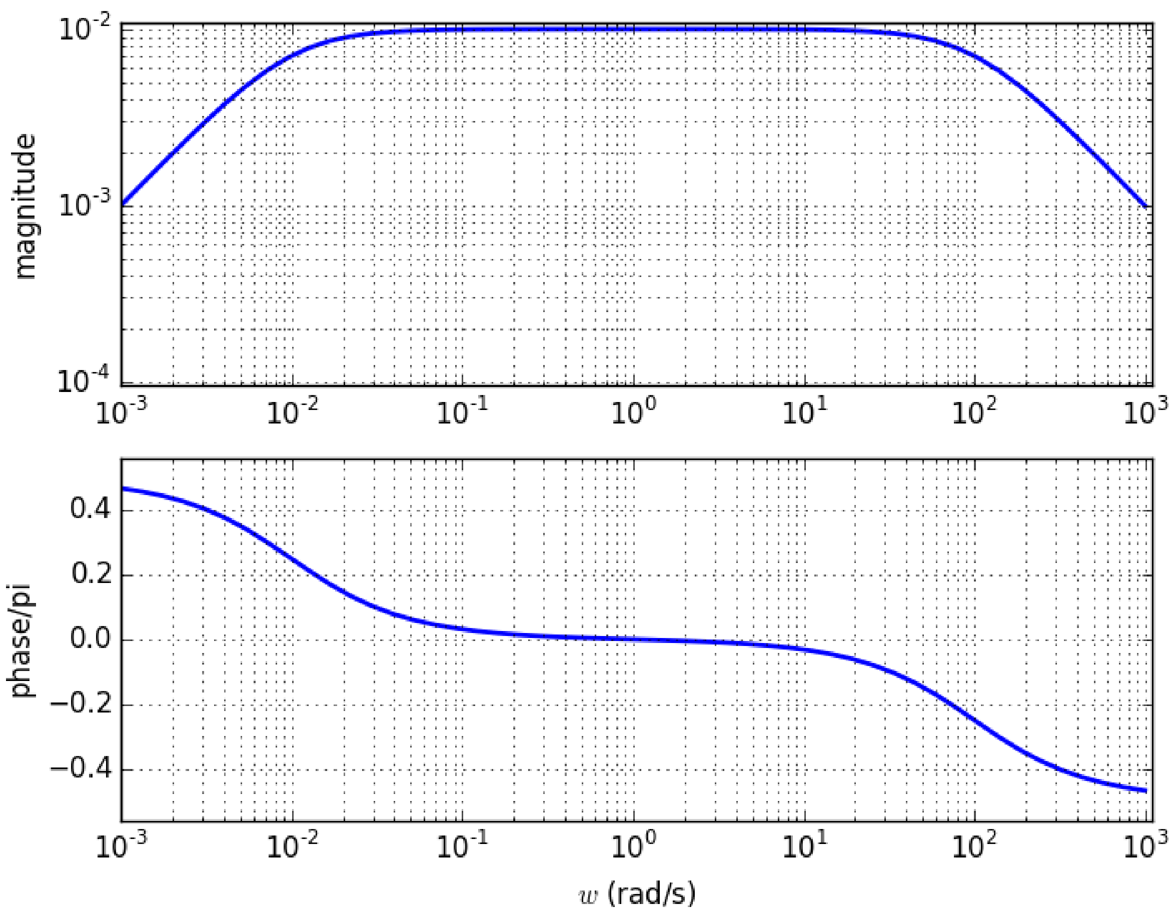




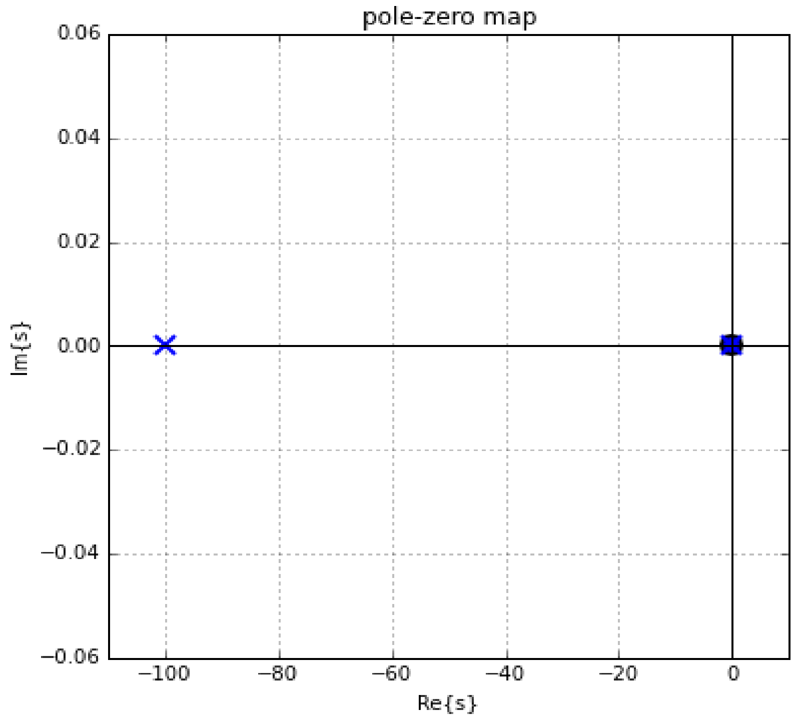
Q3.A step response (V)
1.0
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
10
time (s)

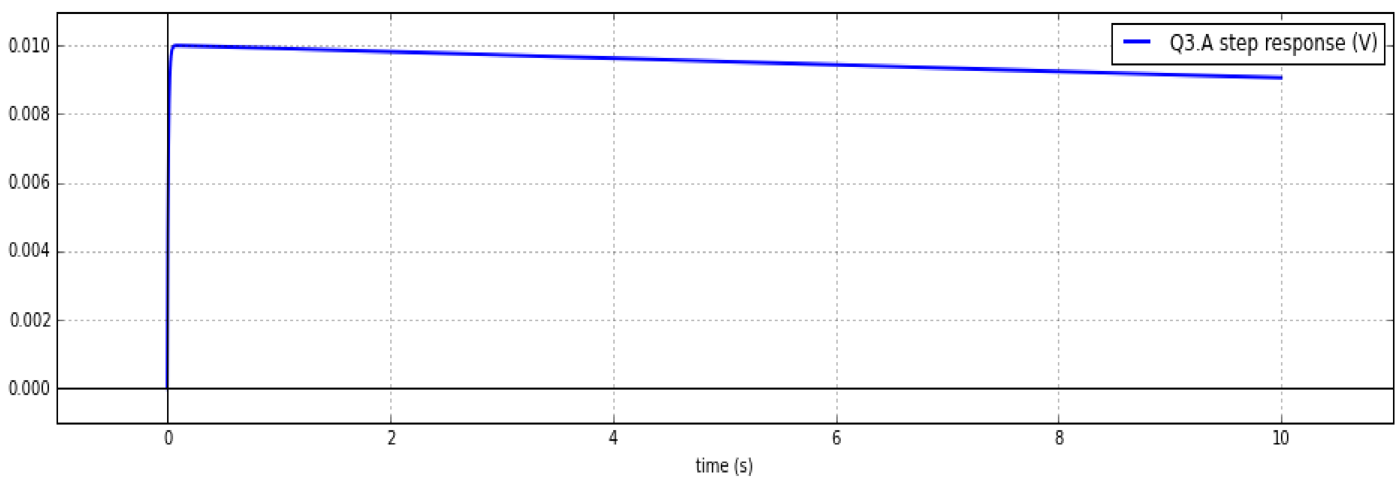
3.B

Bode Plot



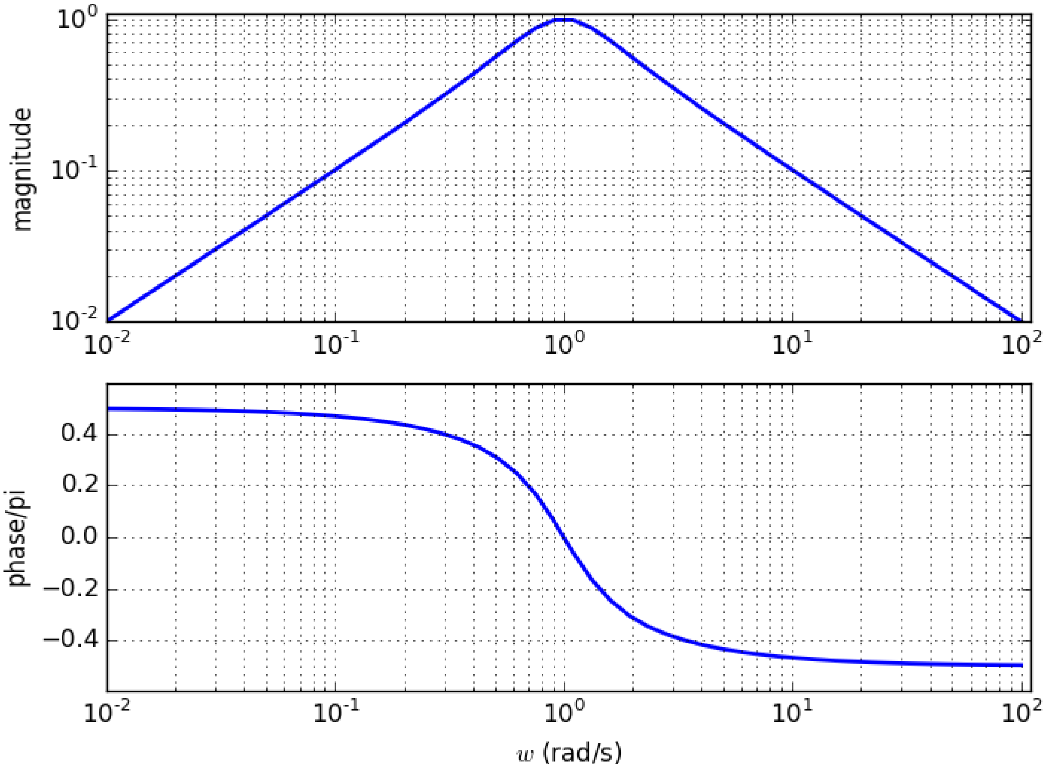
magnitude
10^{-2}
10^{-3}
10^{-4}
10^{-3} 10^{-2} 10^{-1} 10^{0} 10^{1} 10^{2} 10^{3}
phase/pi
0.4
0.2
0.0
−0.2
−0.4
10^{-3} 10^{-2} 10^{-1} 10^{0} 10^{1} 10^{2} 10^{3}
w (rad/s)



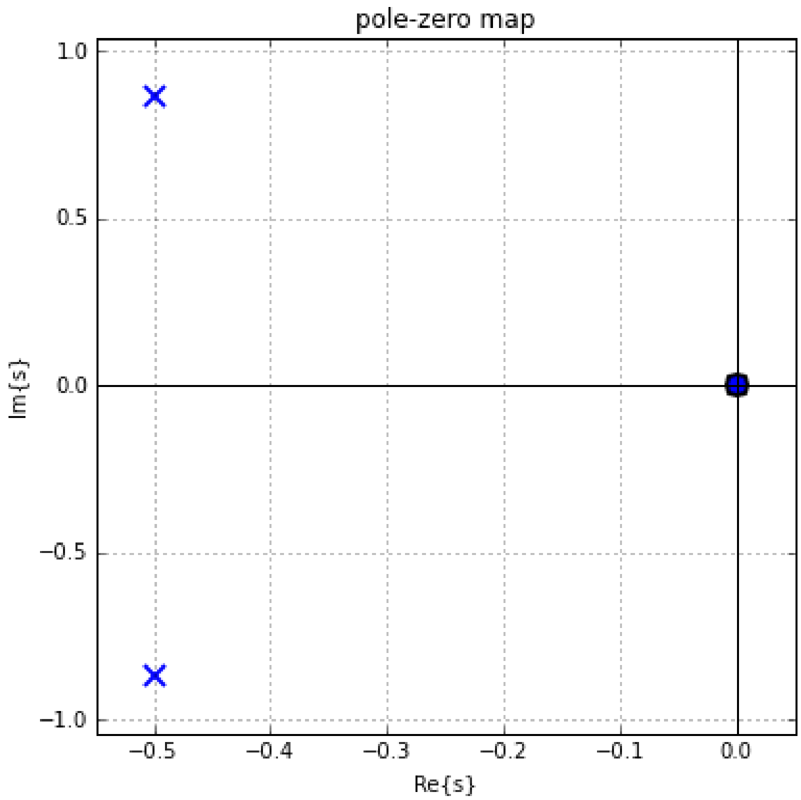


(Ignore legend: Q3.B Step response)

3.C
Bode plot

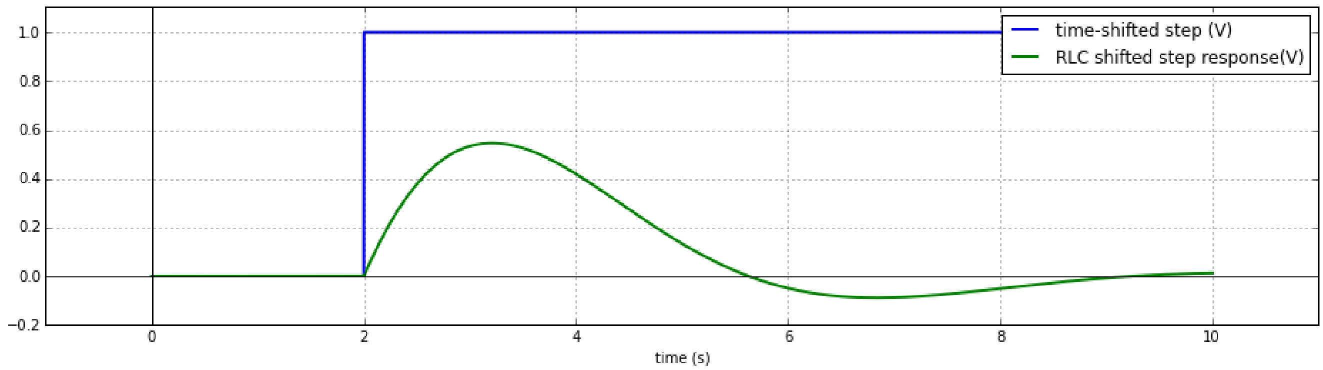


magnitude
10^0
10^{-1}
10^{-2}
phase/pi
0.4
0.2
0.0
−0.2
−0.4
10^{-2}
10^{-1}
10^0
10^1
10^2
w (rad/s)



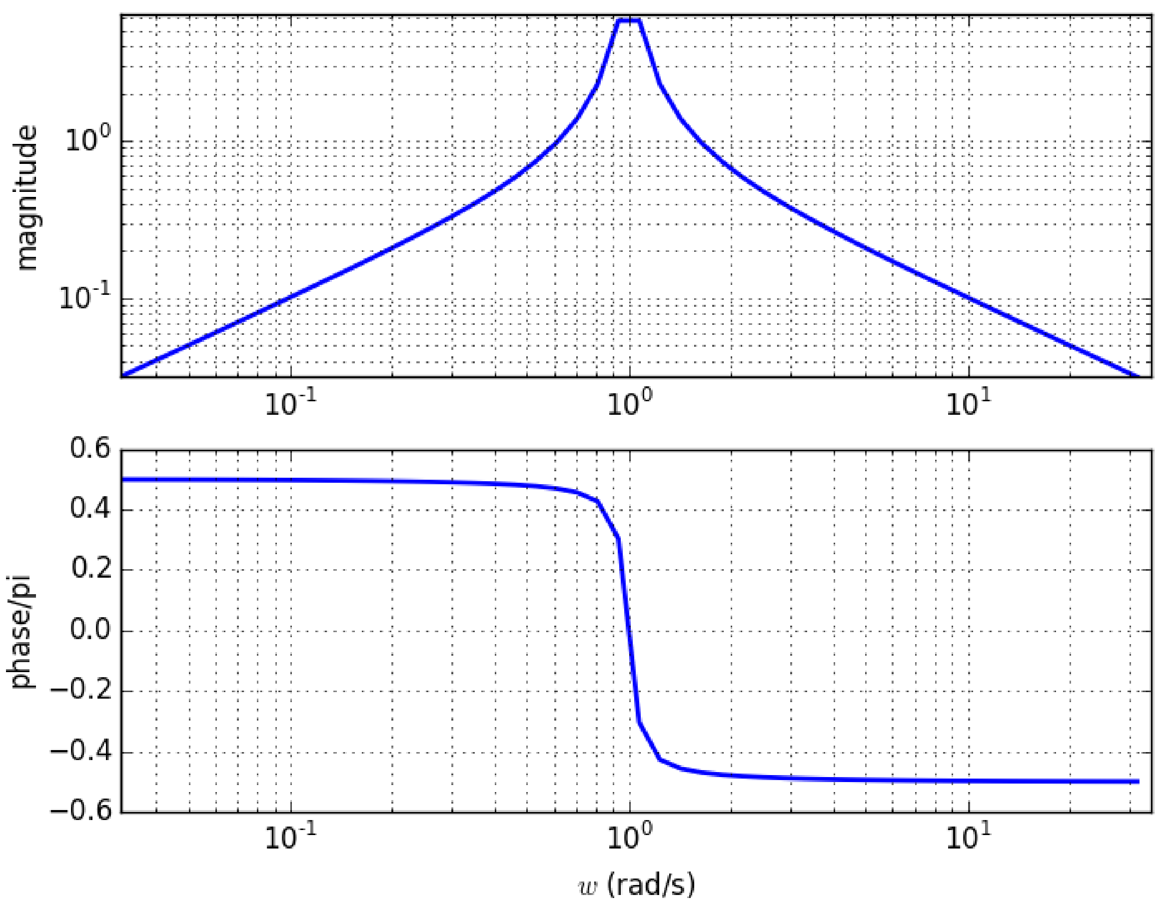
pole-zero map
1.0
0.5
0.0
−0.5
−1.0
Im{s}
−0.5
−0.4
−0.3
−0.2
−0.1
0.0
Re{s}

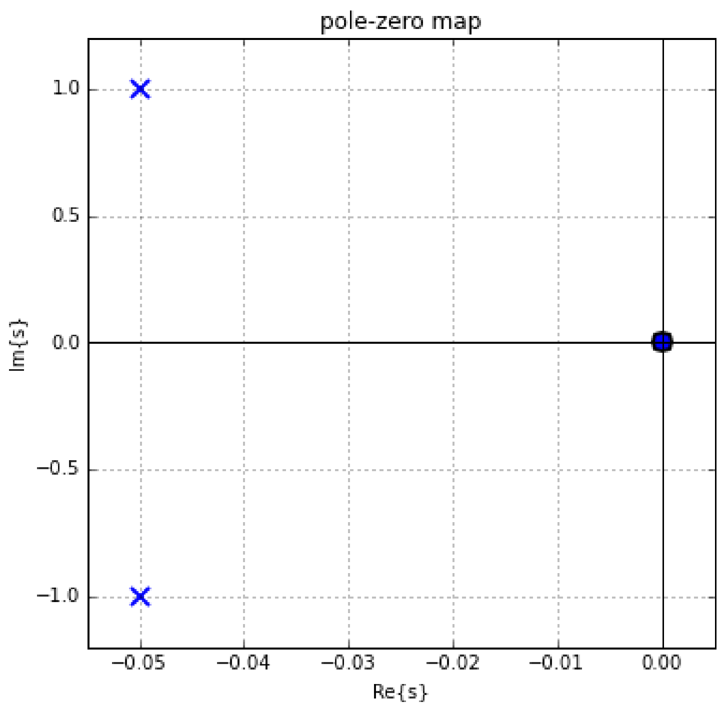
Step Response



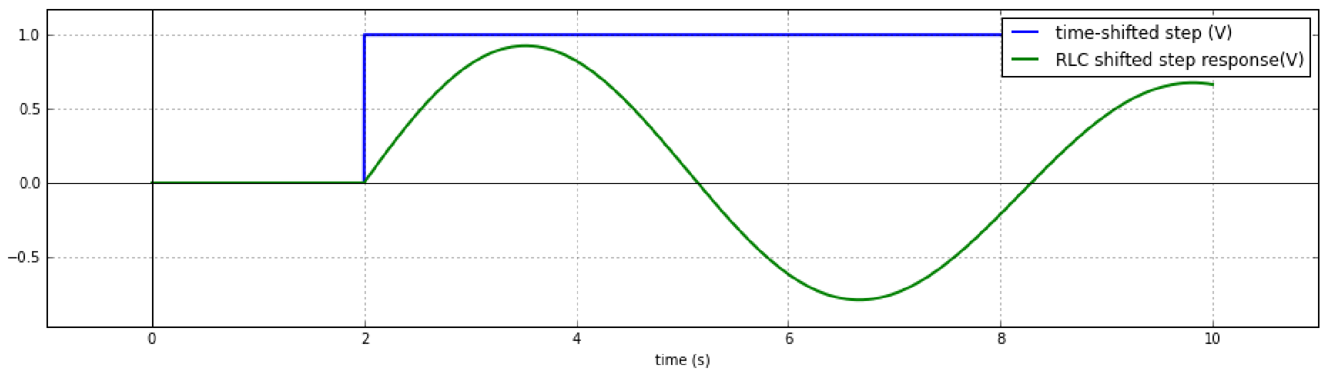
3.D

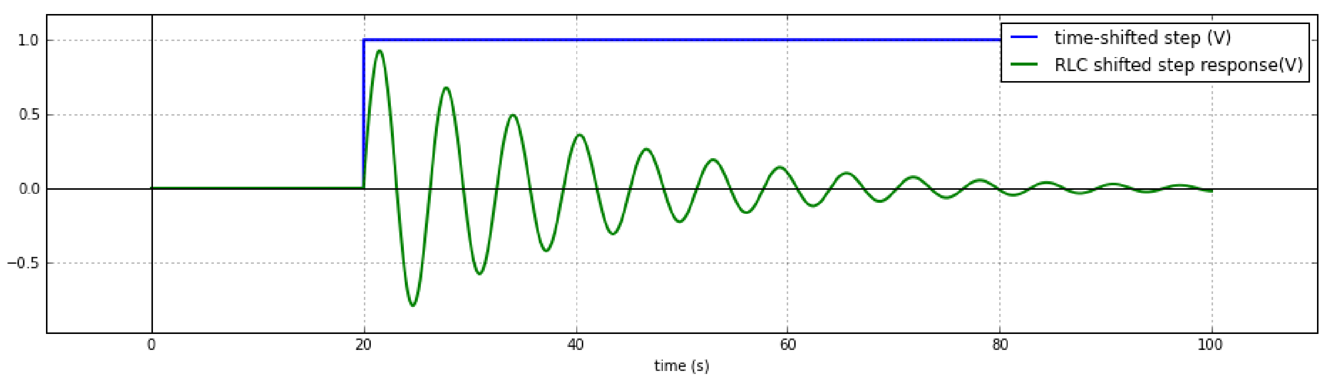
Bode Plot





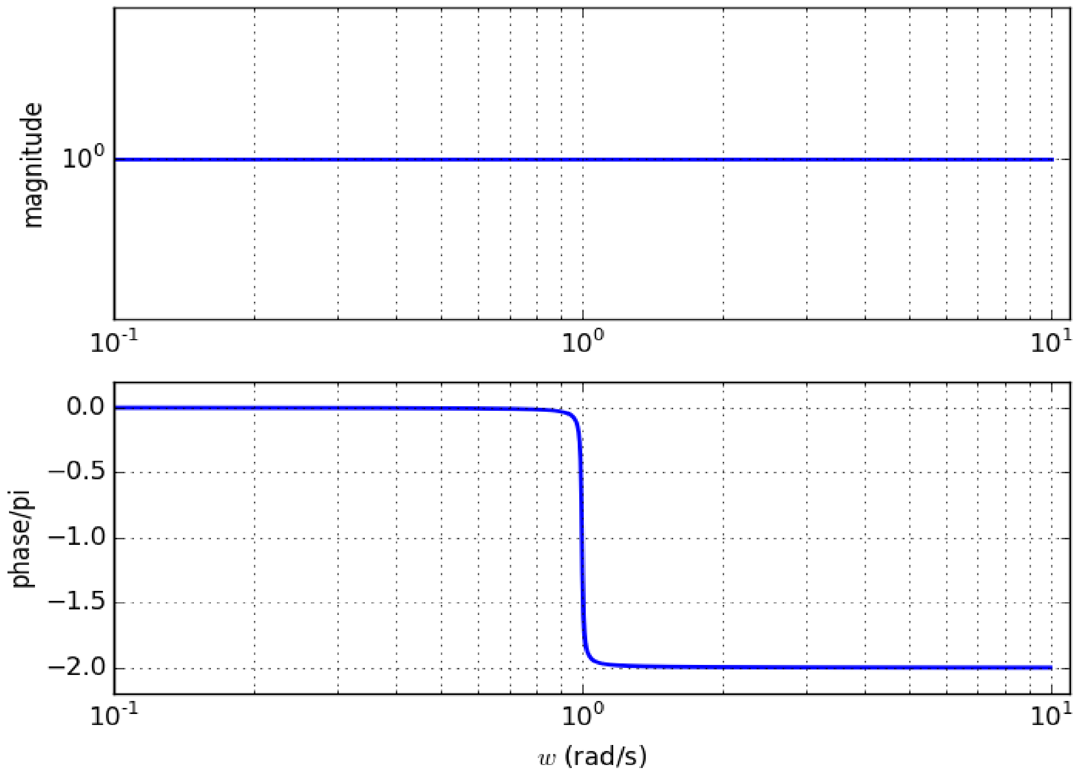
3.D Step Response



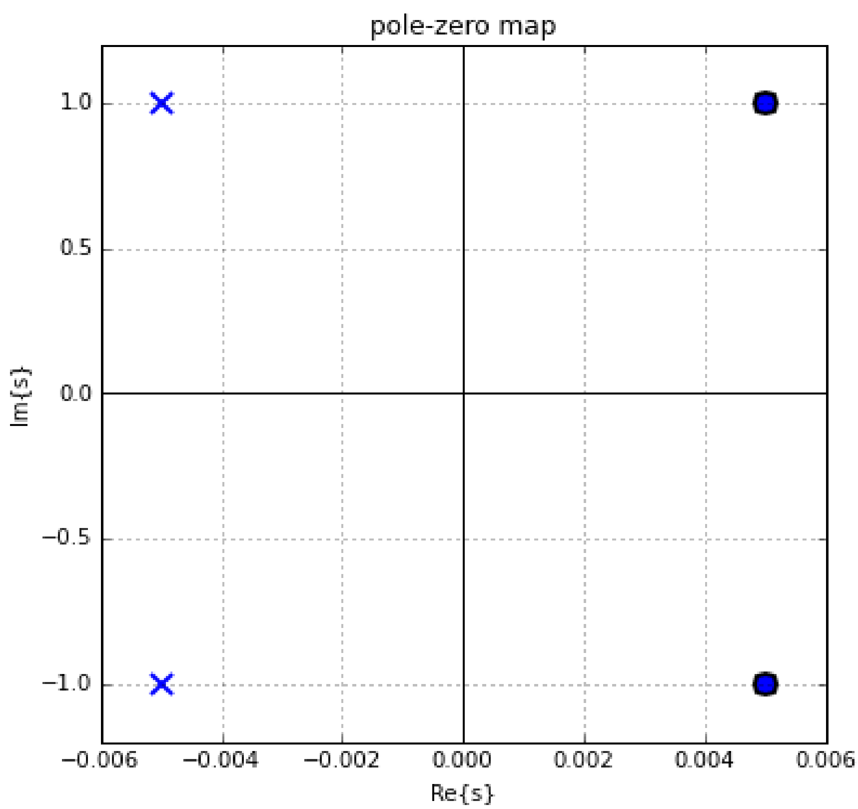


Plotted over a longer timescale show the decay more clearly.

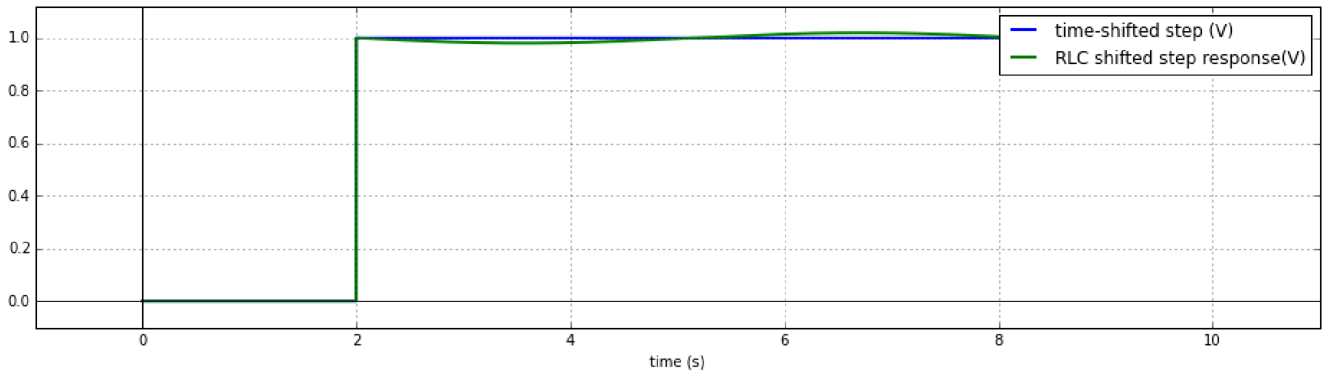
3. E
Bode Plot



There is no change of phase, but the frequencies over 1 Hz are shifted by 2 rad.

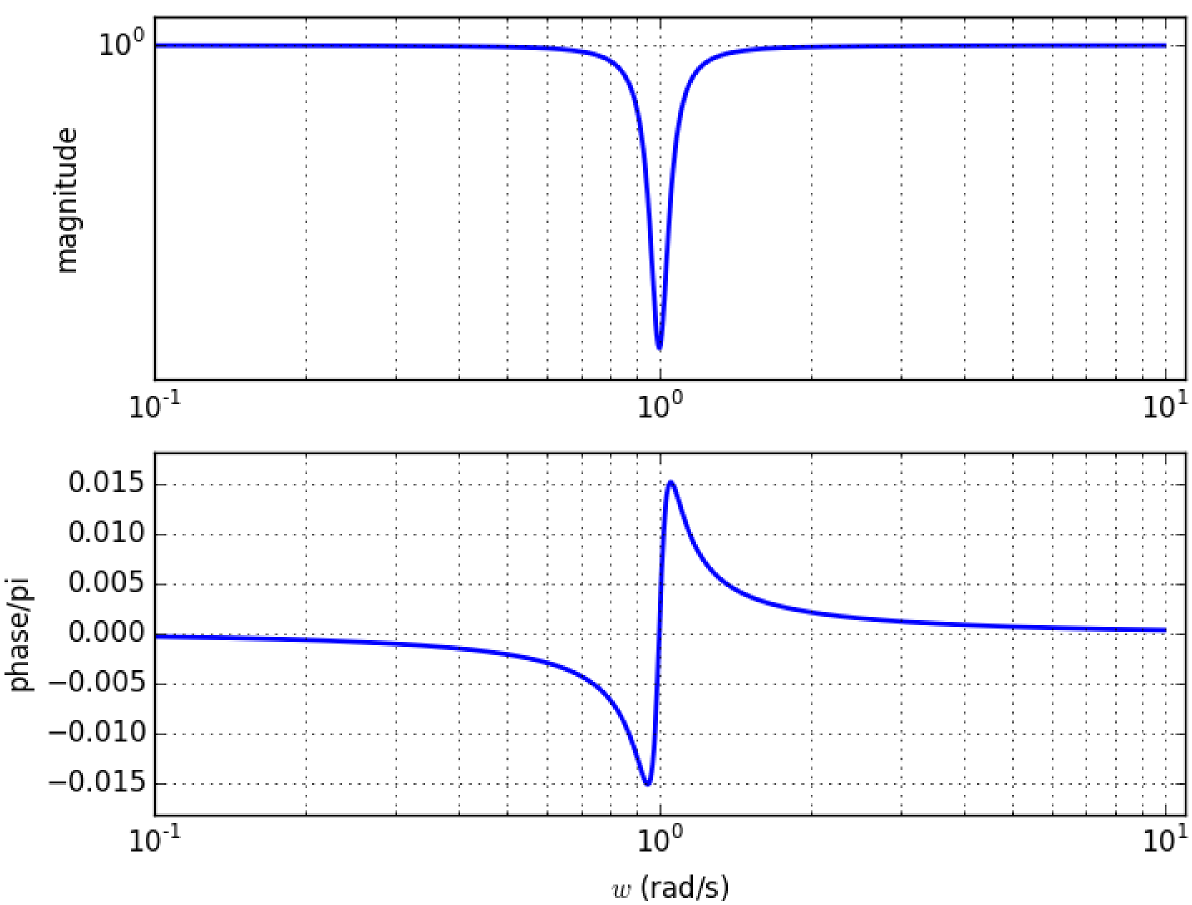


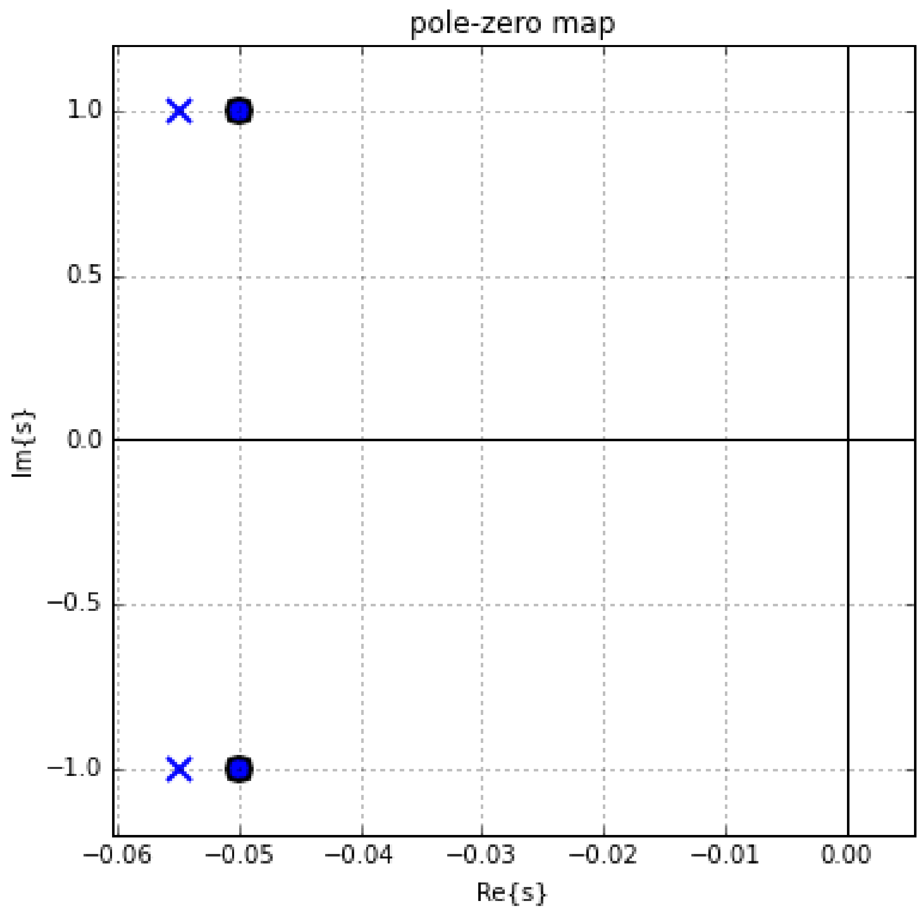
Step Response

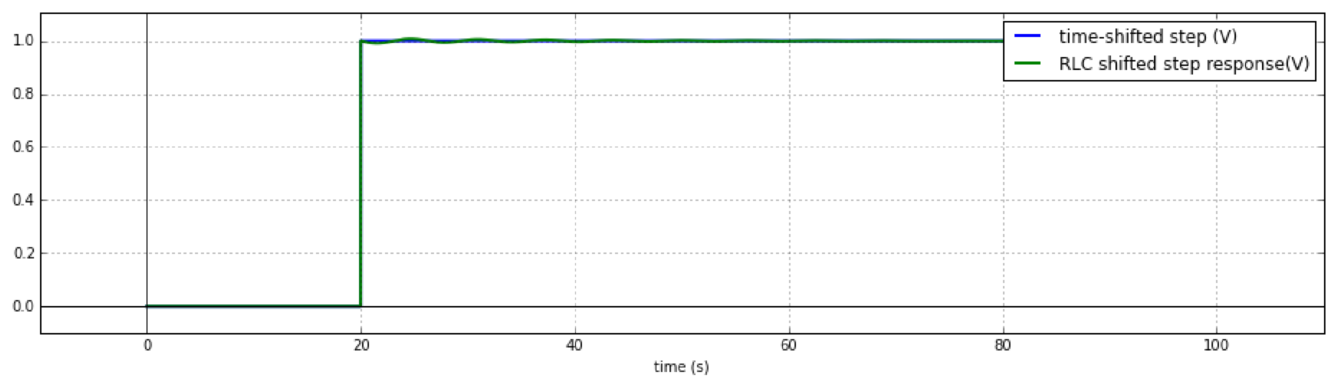


3.F

Bode plot







The oscillations slowly decay to zero, as opposed to the previous, where they appear to continue indefinitely.

Michael Bocamazo

SigSys

PS 10 Q4

In [43]:

```
%matplotlib inline
%run convenience.ipynb
from __future__ import print_function
import numpy as np
import matplotlib.pyplot as plt
from scipy import signal

np.set_printoptions(precision=2,suppress=True) # numpy output options

pi=np.pi
j=1j
```

4a: no control

H = 1/(s^2-0.01s+1)

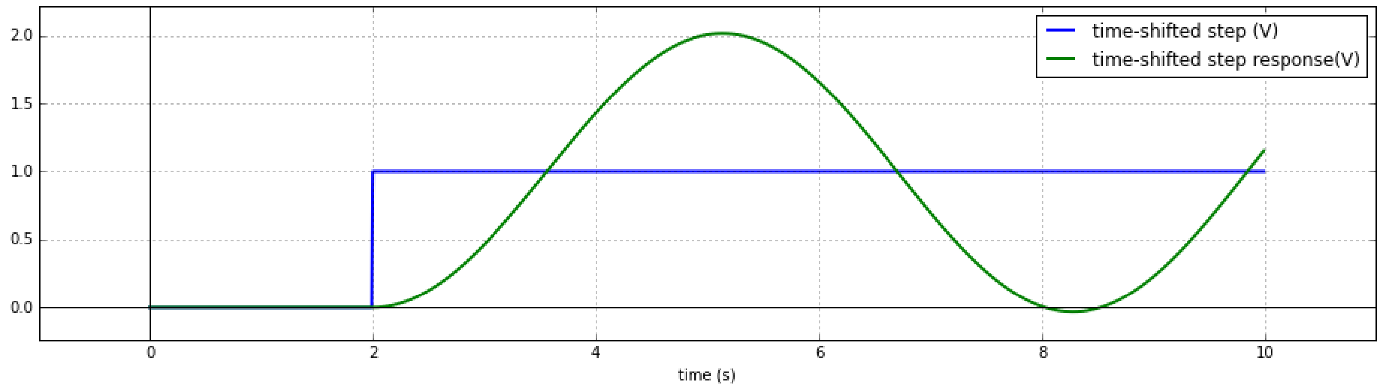
In [44]:

```
k = 1;
#derivative control
#sys = signal.lti([k,0], [1,k-0.01,1])
#proportional control
sys = signal.lti([1], [1,-0.01,1])
#integral control
#sys = signal.lti([k], [1,-0.01,1,k])
def print_sys(sys):
    t=np.arange(0,10,0.01)
    #create a delayed unit step
    x = np.zeros(len(t))
    pulse_start=t[-1]/5 #delay by a fifth of the end time of t_step
    x[t>pulse_start] = 1

    _t,y,state=signal.lsim(sys,x,t)#simulate the output, t is not the same as given in the input?

    timeplot(t,x,label='time-shifted step (V)')
    timeplot(t,y,label='time-shifted step response(V)')#BUG: plot dissapears if you add a space before (V)
    print("Zeros:")
    print(np.roots(sys.num))
    print("Poles:")
    print(np.roots(sys.den))
```

In [45]:

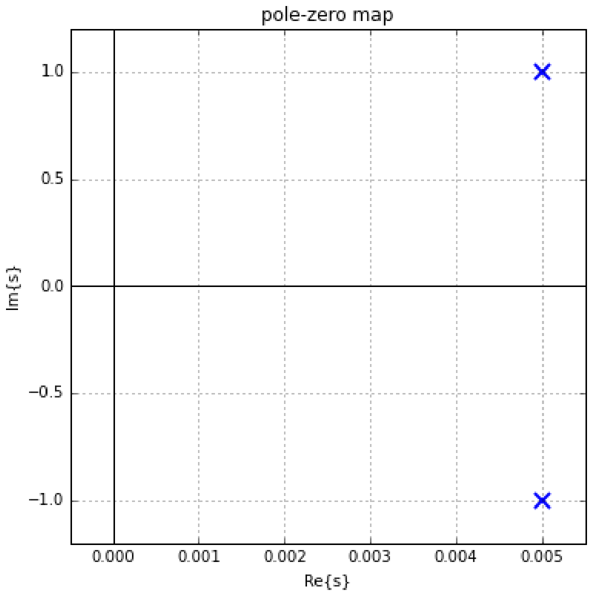
```
print_sys(sys)
```

```
Zeros:
[]
Poles:
[ 0.01+1.j  0.01-1.j]
```



In [46]:

```
pzmap(sys)
```



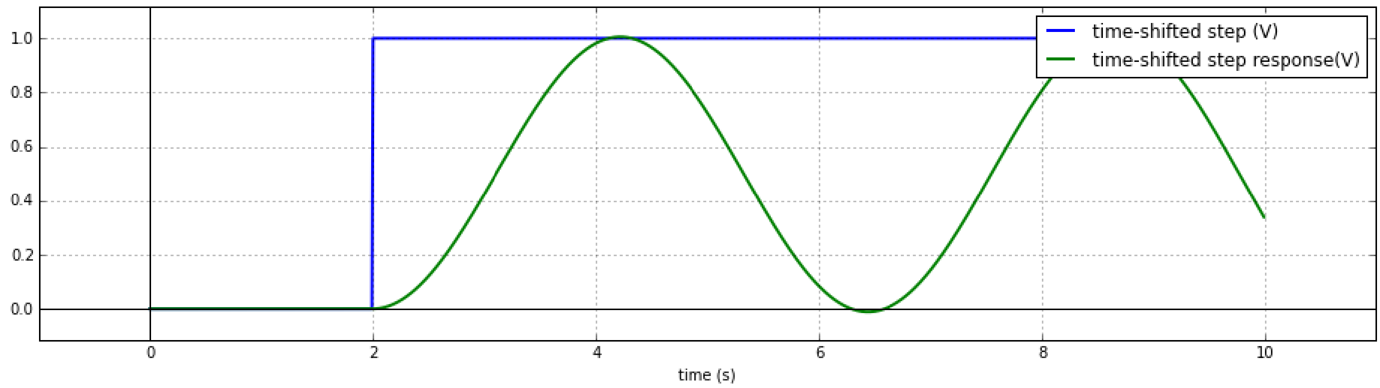
4b: Proportional control

H(s) = Kp/(s^2-0.01s+1+Kp)

In [47]:

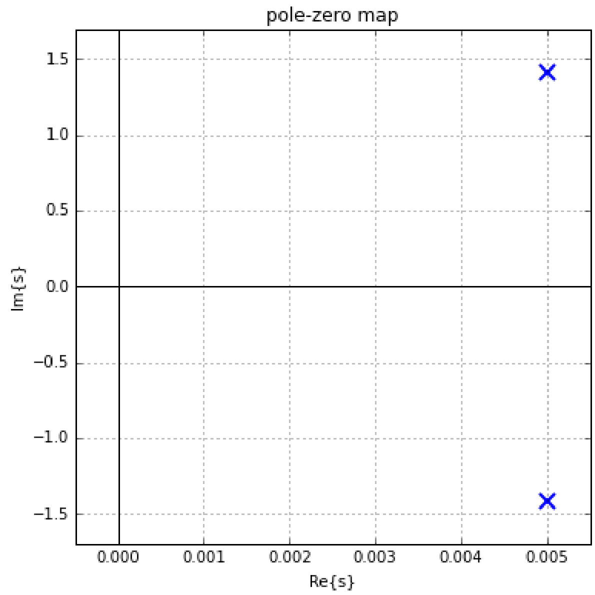
```
k = 1;
sys = signal.lti([k], [1,-0.01,1+k])
print_sys(sys)
```

```
Zeros:
[]
Poles:
[ 0.01+1.41j  0.01-1.41j]
```



In [48]:

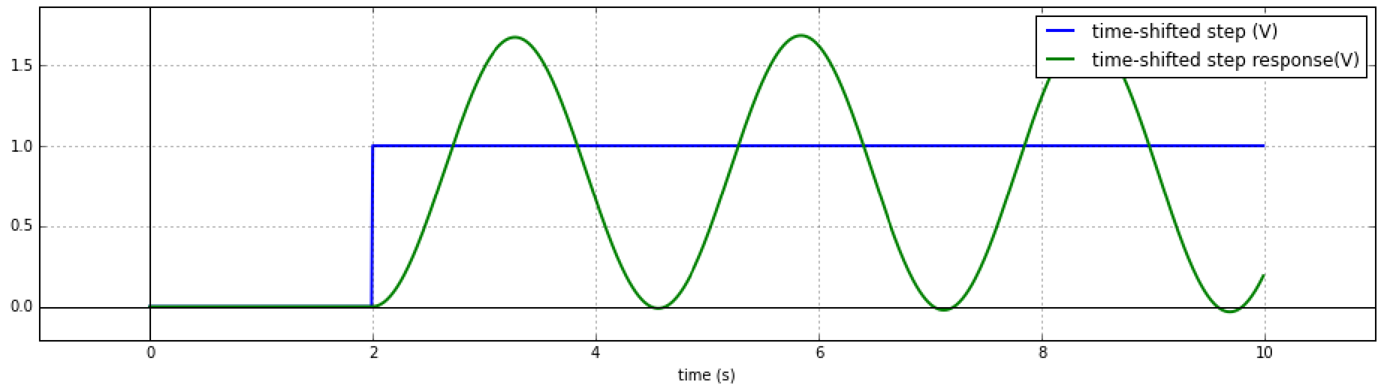
```
pzmap(sys)
```



In [49]:

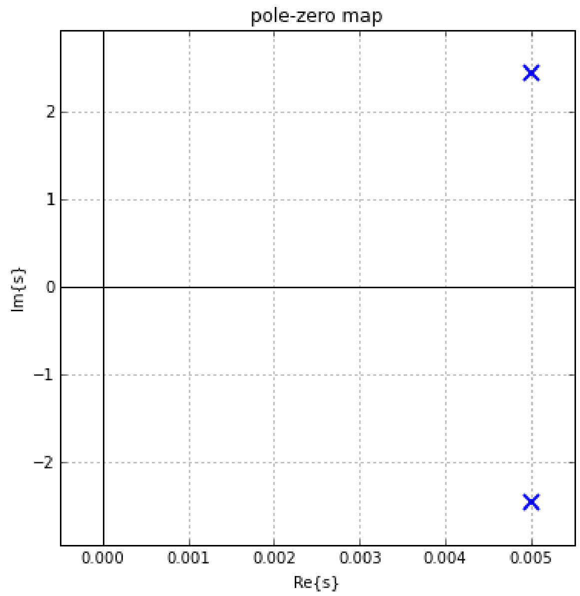
```
k = 5;
sys = signal.lti([k], [1,-0.01,1+k])
print_sys(sys)
```

```
Zeros:
[]
Poles:
[ 0.01+2.45j  0.01-2.45j]
```



In [50]:

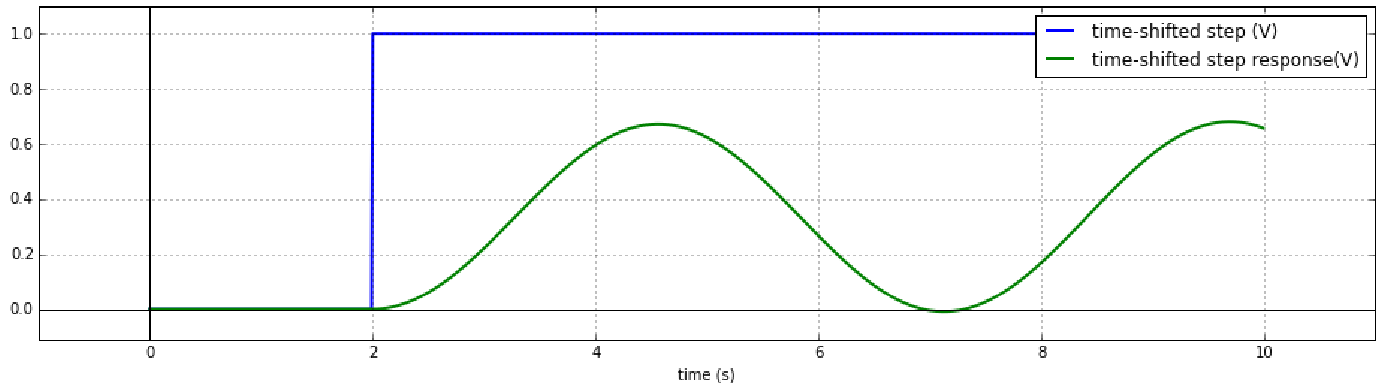
```
pzmap(sys)
```



In [51]:

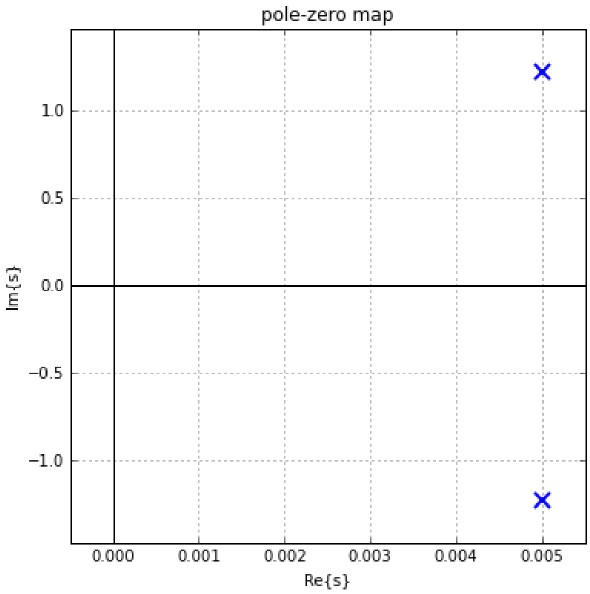
```
k = 0.5;
sys = signal.lti([k], [1,-0.01,1+k])
print_sys(sys)
```

```
Zeros:
[]
Poles:
[ 0.01+1.22j  0.01-1.22j]
```



In [52]:

```
pzmap(sys)
```



Proportional Control **Doesn't stabilize**: poles on the right half-plane

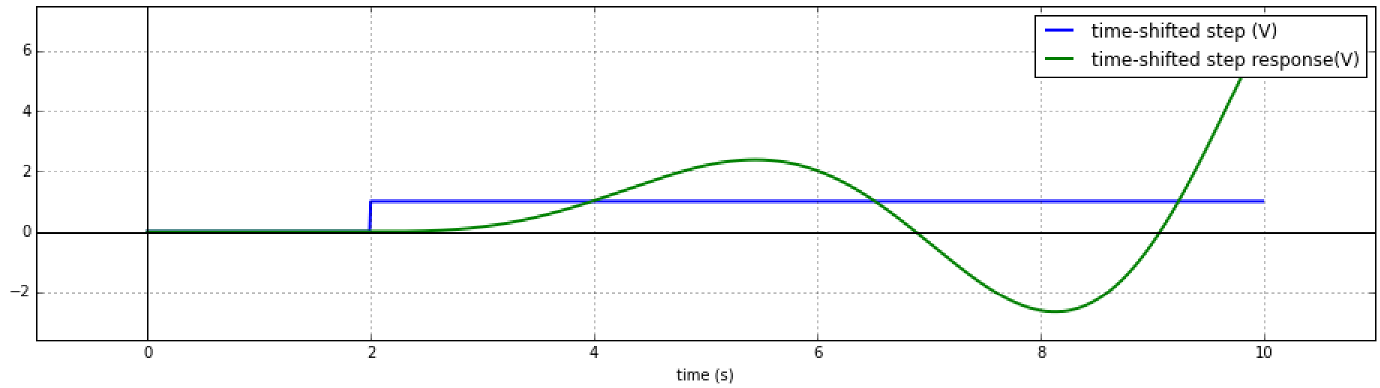
4c Integral Control

H(s) = Ki/(s^3 -0.01s^2 +s +Ki)

In [53]:

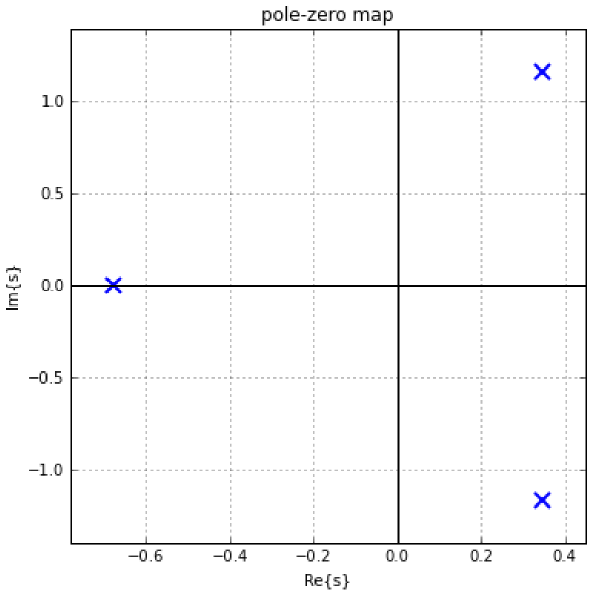
```
k = 1;
#integral control
sys = signal.lti([k], [1,-0.01,1,k])
print_sys(sys)
```

```
Zeros:
[]
Poles:
[ 0.35+1.16j  0.35-1.16j -0.68+0.j   ]
```



In [54]:

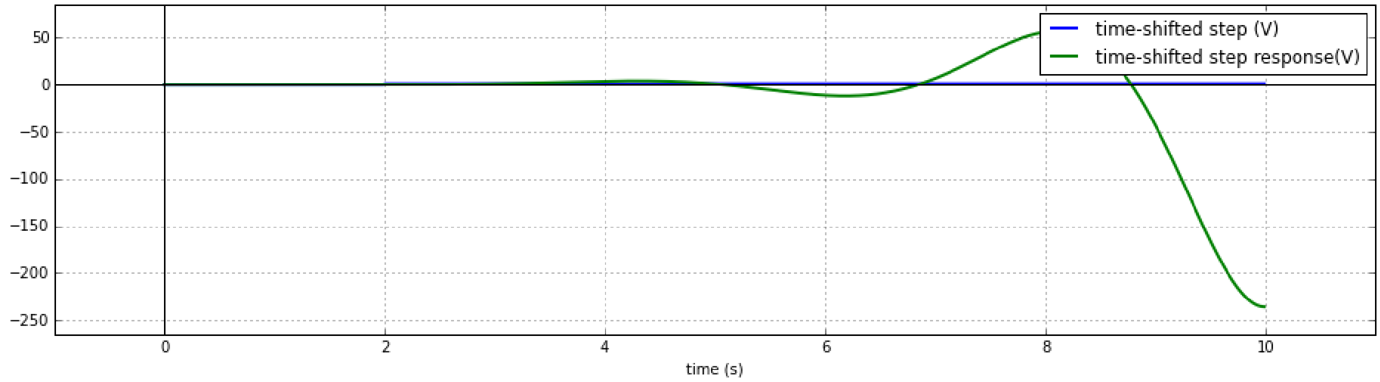
```
pzmap(sys)
```



In [55]:

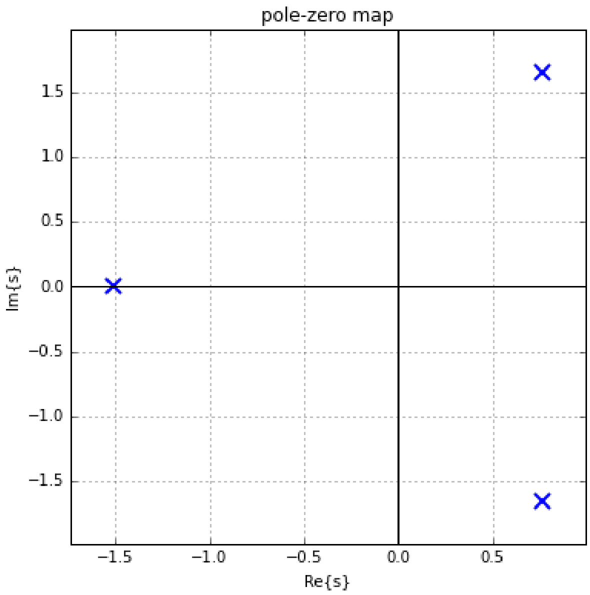
```
k = 5;
#integral control
sys = signal.lti([k], [1,-0.01,1,k])
print_sys(sys)
```

```
Zeros:
[]
Poles:
[ 0.76+1.65j  0.76-1.65j -1.51+0.j  ]
```



In [56]:

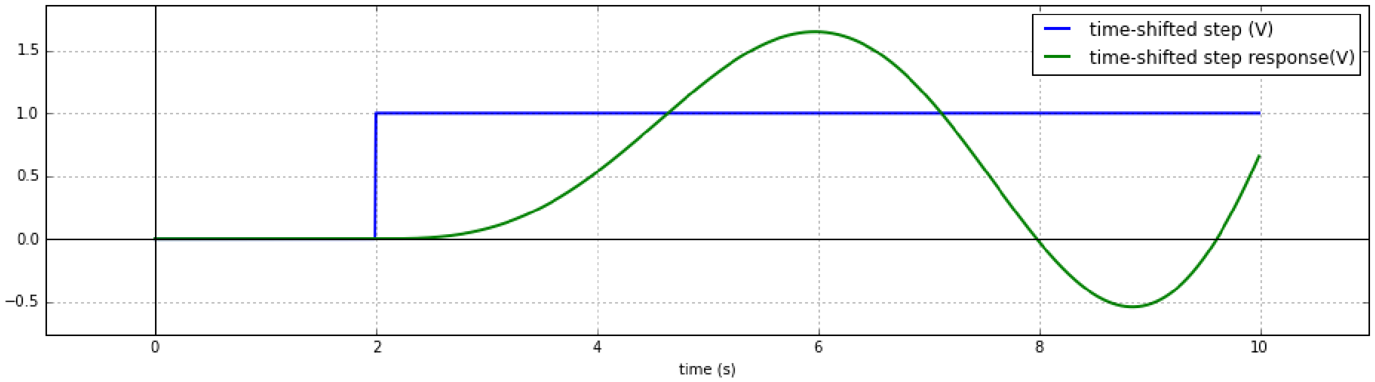
```
pzmap(sys)
```



In [57]:

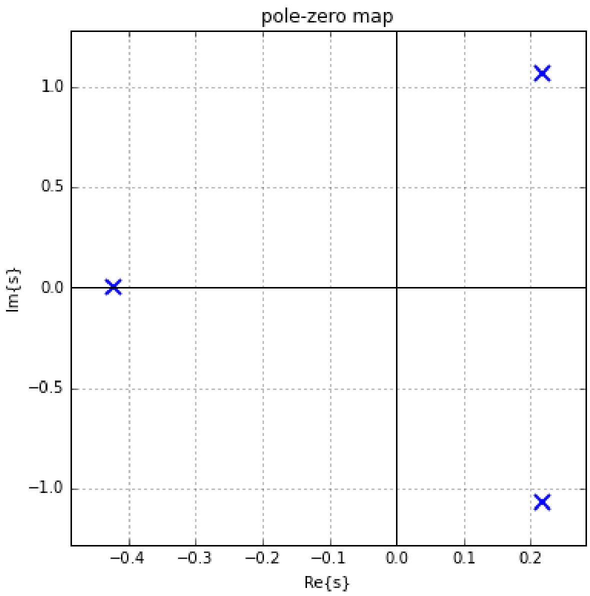
```
k = 0.5;
#integral control
sys = signal.lti([k], [1,-0.01,1,k])
print_sys(sys)
```

```
Zeros:
[]
Poles:
[ 0.22+1.07j  0.22-1.07j -0.42+0.j  ]
```



In [58]:

```
pzmap(sys)
```



Integral Control **Doesn't stabilize**: poles on the right half-plane, no matter the gain

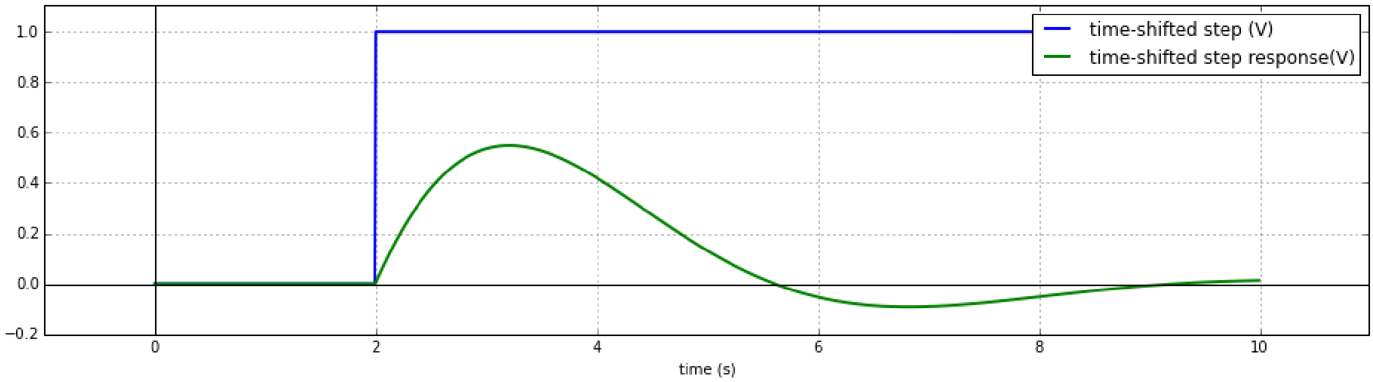
4d Derivative Control

H(s) = Kd.s/(s^2 +(Kd-0.01).s +1)

In [59]:

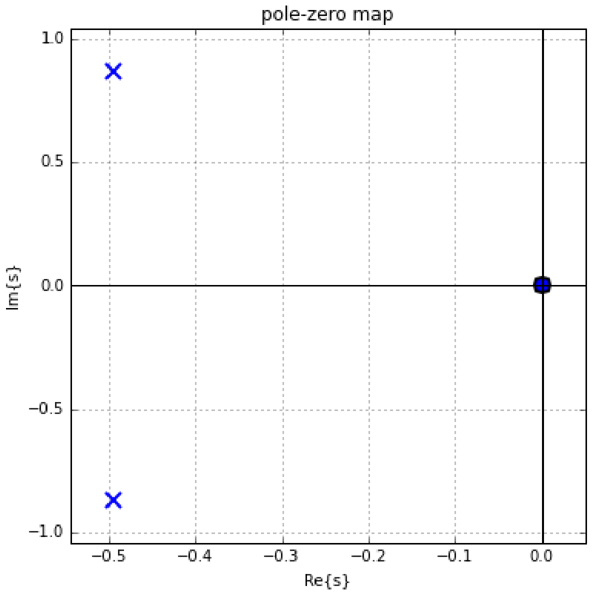
```
k = 1;
#derivative control
sys = signal.lti([k,0], [1,k-0.01,1])
print_sys(sys)
```

```
Zeros:
[ 0.]
Poles:
[-0.49+0.87j -0.49-0.87j]
```



In [60]:

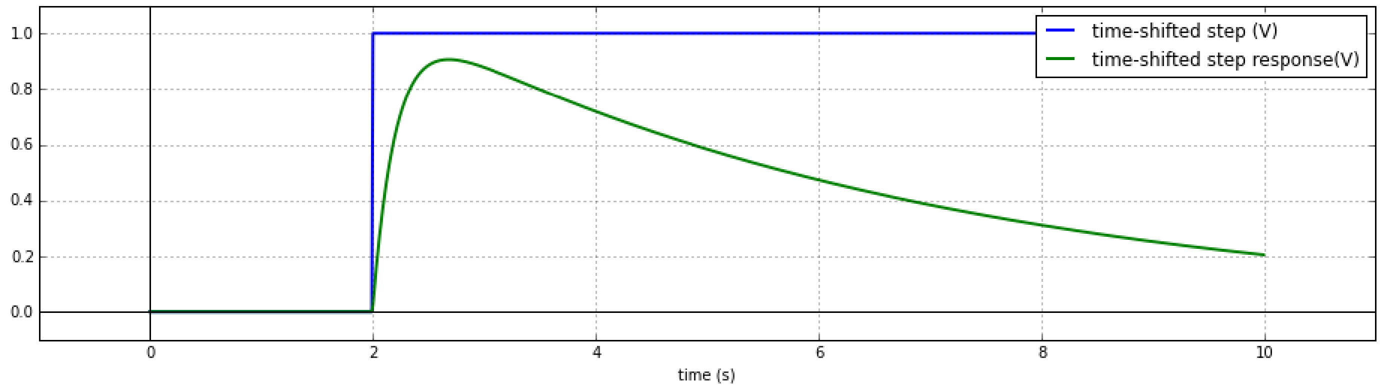
```
pzmap(sys)
```



In [61]:

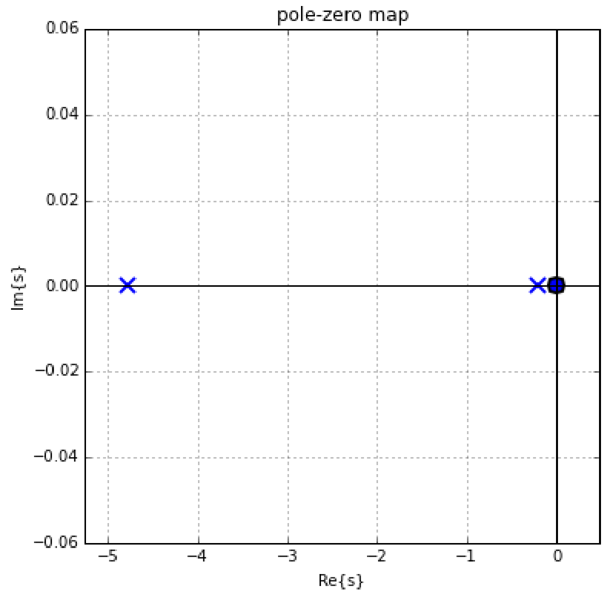
```
k = 5;
#derivative control
sys = signal.lti([k,0], [1,k-0.01,1])
print_sys(sys)
```

```
Zeros:
[ 0.]
Poles:
[-4.78 -0.21]
```



In [62]:

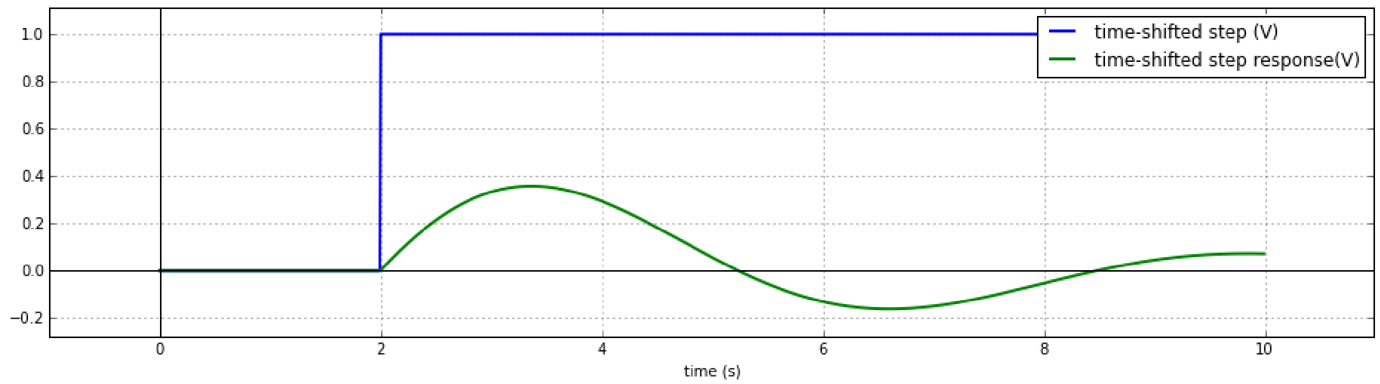
```
pzmap(sys)
```



In [63]:

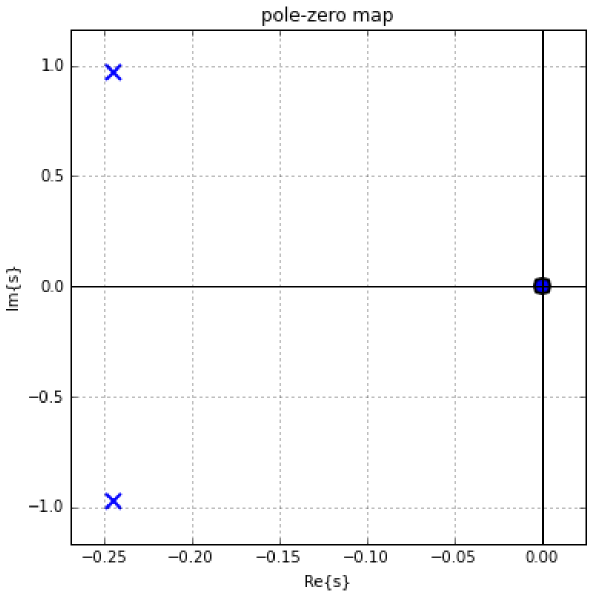
```
k = 0.5;
#derivative control
sys = signal.lti([k,0], [1,k-0.01,1])
print_sys(sys)
```

```
Zeros:
[ 0.]
Poles:
[-0.25+0.97j -0.25-0.97j]
```



In [64]:

```
pzmap(sys)
```



Using derivative control, it **is possible** to stabilize the system after the step input.