Gabrielle Merritt
gmerritt@seas.upenn.edu

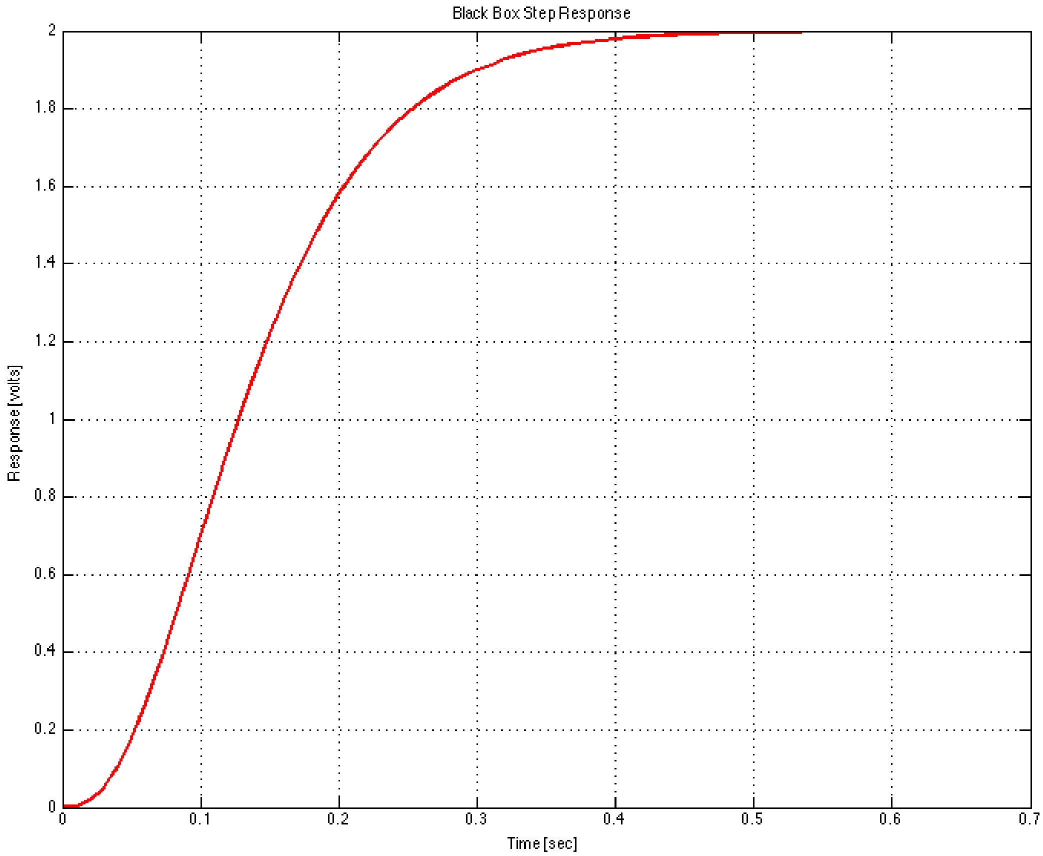
ESE 505 Homework 3

1 Step Response

Below is a picture of the step response for the transfer function described in the homework

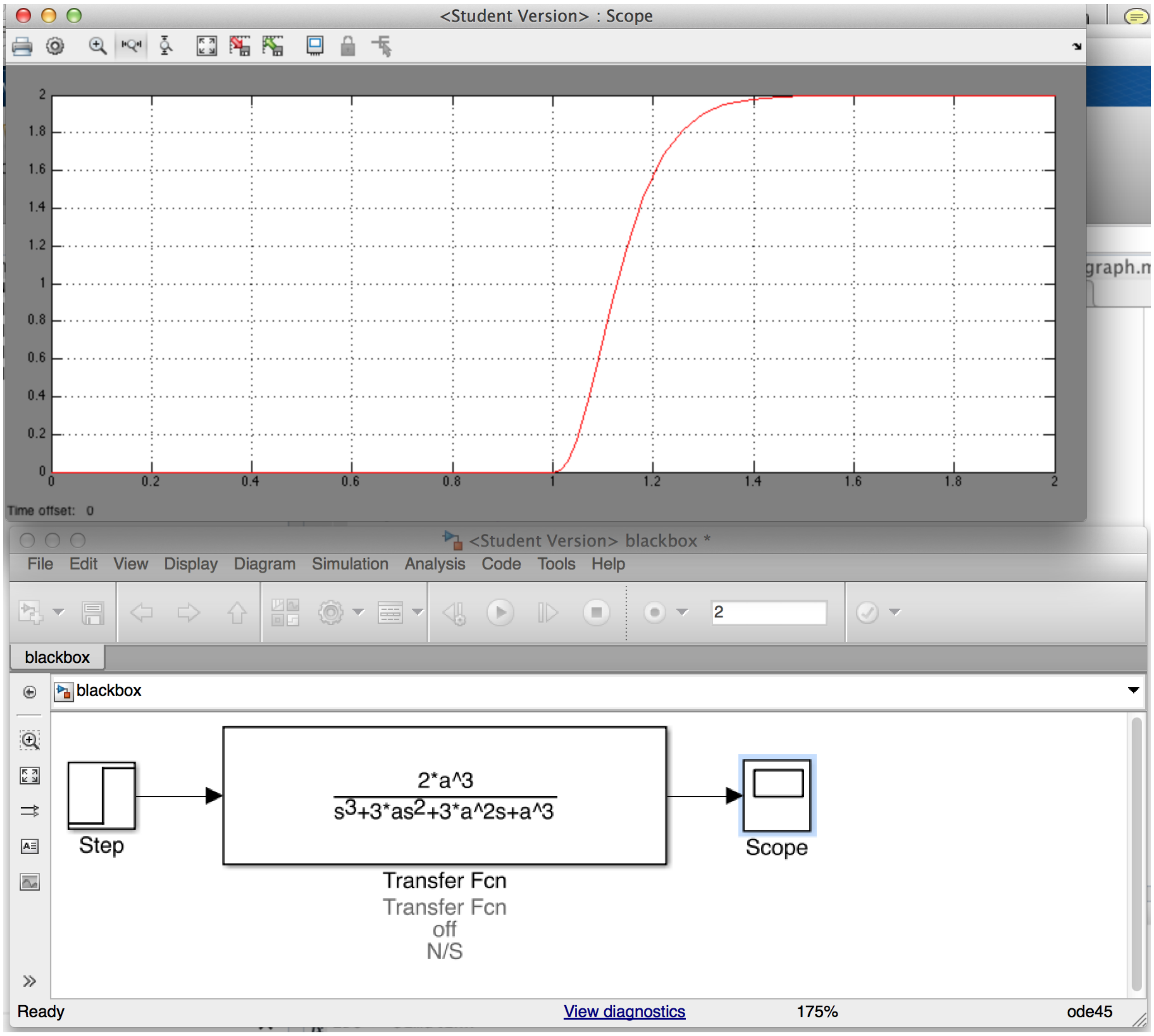
$$\frac{2a^3}{s^3 + 3as^2 + 3a^2s + a^3}$$

Figure 1: Step response



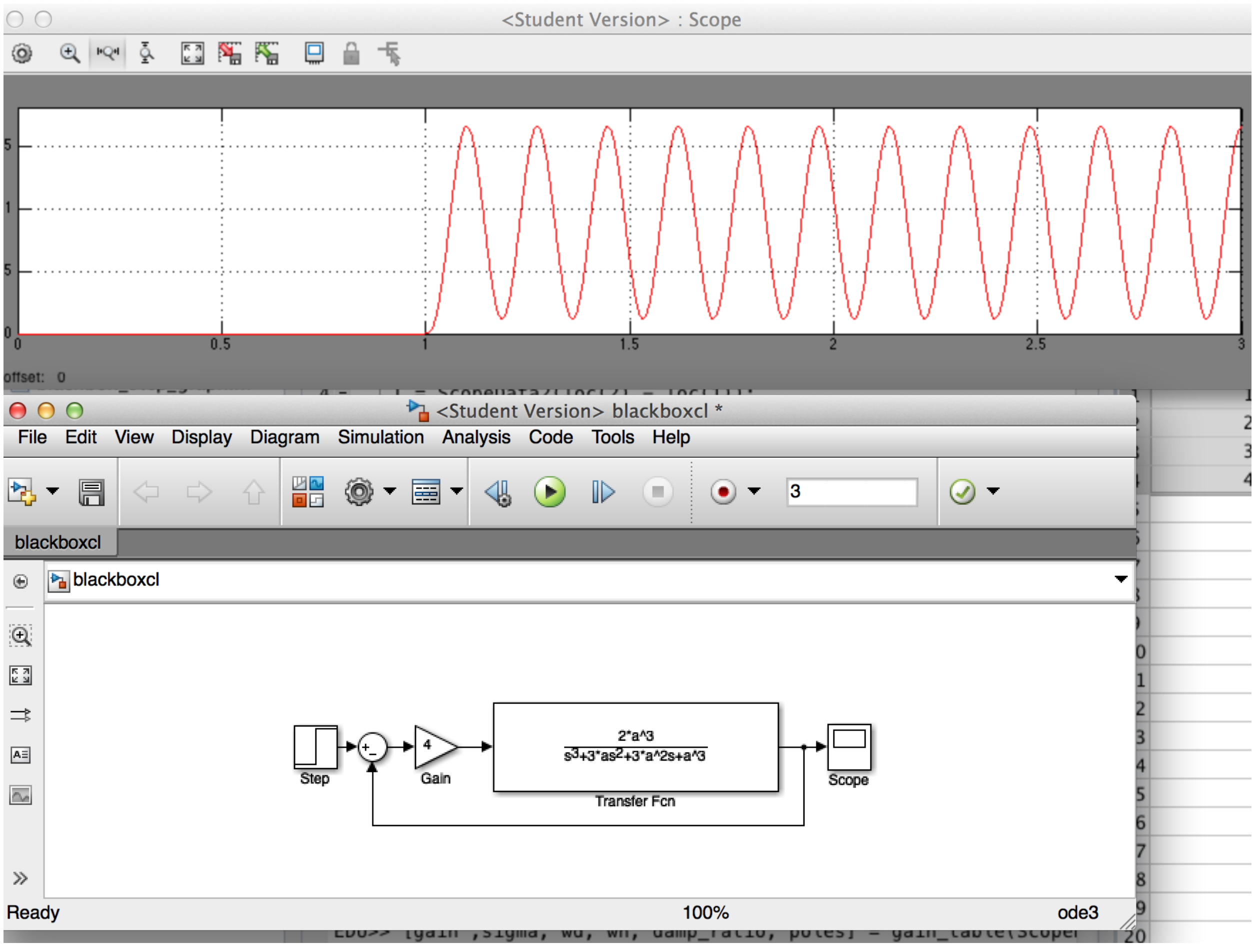
generated using Matlab built in transfer funcition

Figure 2: Simulink step response



Generated using Simulink model

Figure 3: Simulink Proportional Gain (K =4)

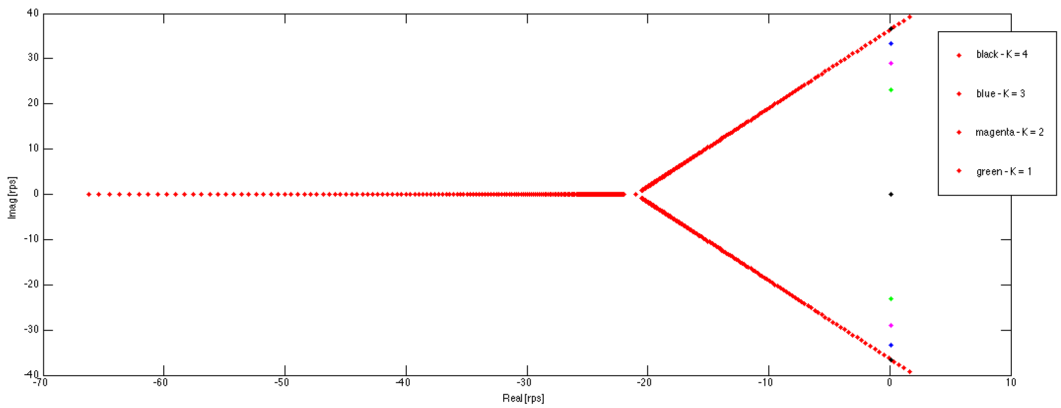


Generated using Simulink model

2 Root Locus and Poles

K	σ	ω_d	ω_n	ζ
1	0.065232138	23.0153308	23.01657113	0.010381458
2	0.11032051	28.954771	28.95923382	0.017555349
3	0.120529727	33.2443667	33.25048283	0.019179373
4	0.105618679	36.53014713	36.53530788	0.01680736

Figure 4: Root Locus for Transfer Function



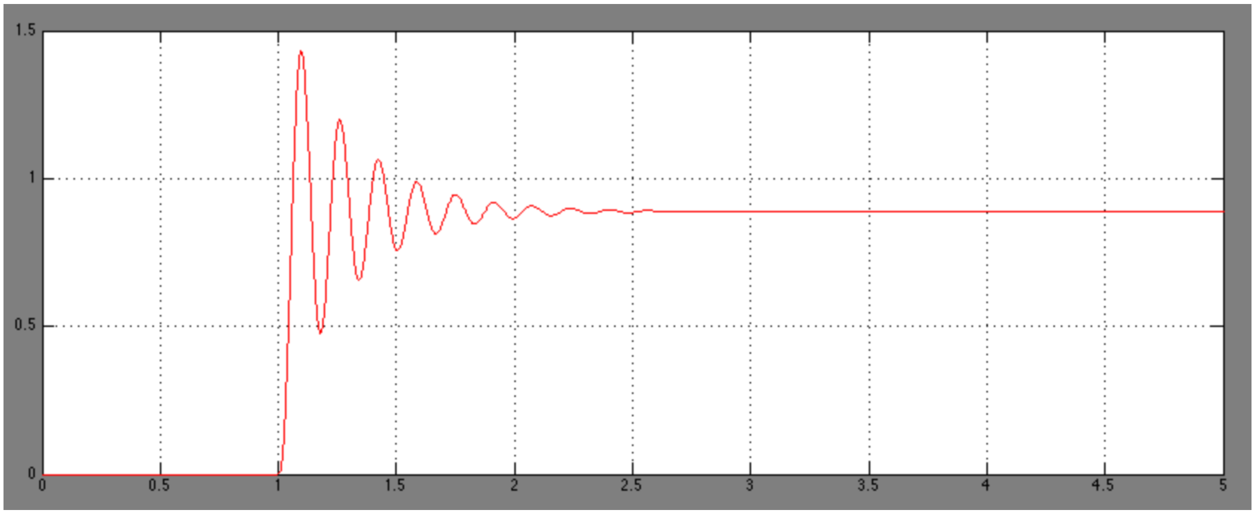
Using different Gains to see effect on pole position

I am a little disconcerted as to why the points don't all lie on the root locus graph. The optimal proportional gain of 4 does, but it would make more sense for all of my poles for all my gains to lie somewhere on the root locus graph.

3 Closed loop control with Derivative Gain

Kd = .0257 To obtain this I used the damping ratio from my table as as well as my natural frequency to find kd. I found Kd using sgrid and rlocfind. S grid to locate the point on the root locust curve that corresponded to damping my proportional gains. I noticed that it is slightly off from the recommended gain of .0257, I think this is due to some errors I might have in finding my damping ratio.

Figure 5: Simulink Proportional Gain (K =4) and Derivative Gain (Kd = .0257)



Generated using Simulink model

4 Adding an Integral Term

Initial damp ratio - .0153 withK_I set to 0. For this part of the assignment I used the form

$$D_p + K_p N(s) + sK_d N(s) + \frac{1}{s} K_I N(S) = 0$$

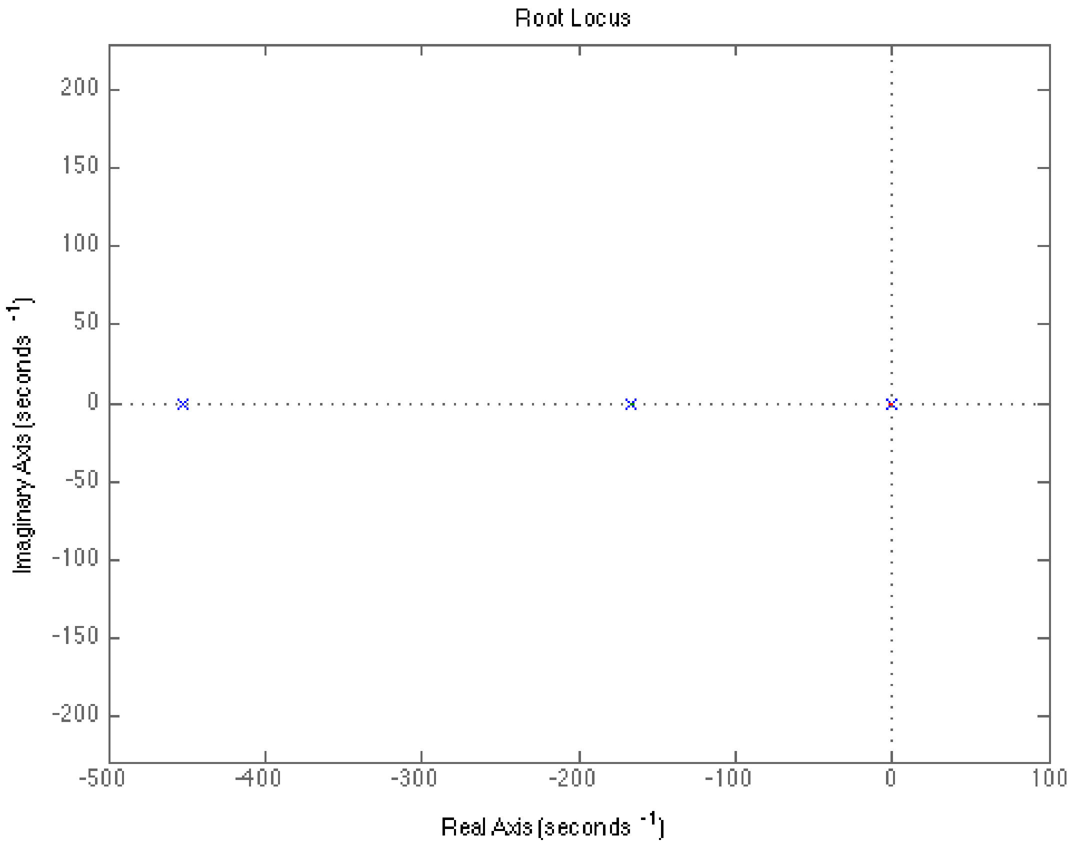
Where

$$A(s) = D_P + K_p N(s) + sK_d N(s)$$

$$B(s) = 1/2N(s)$$

From this I constructed this root locus path,which doesn't seem to return a damping ratio half ofK_I= 0.

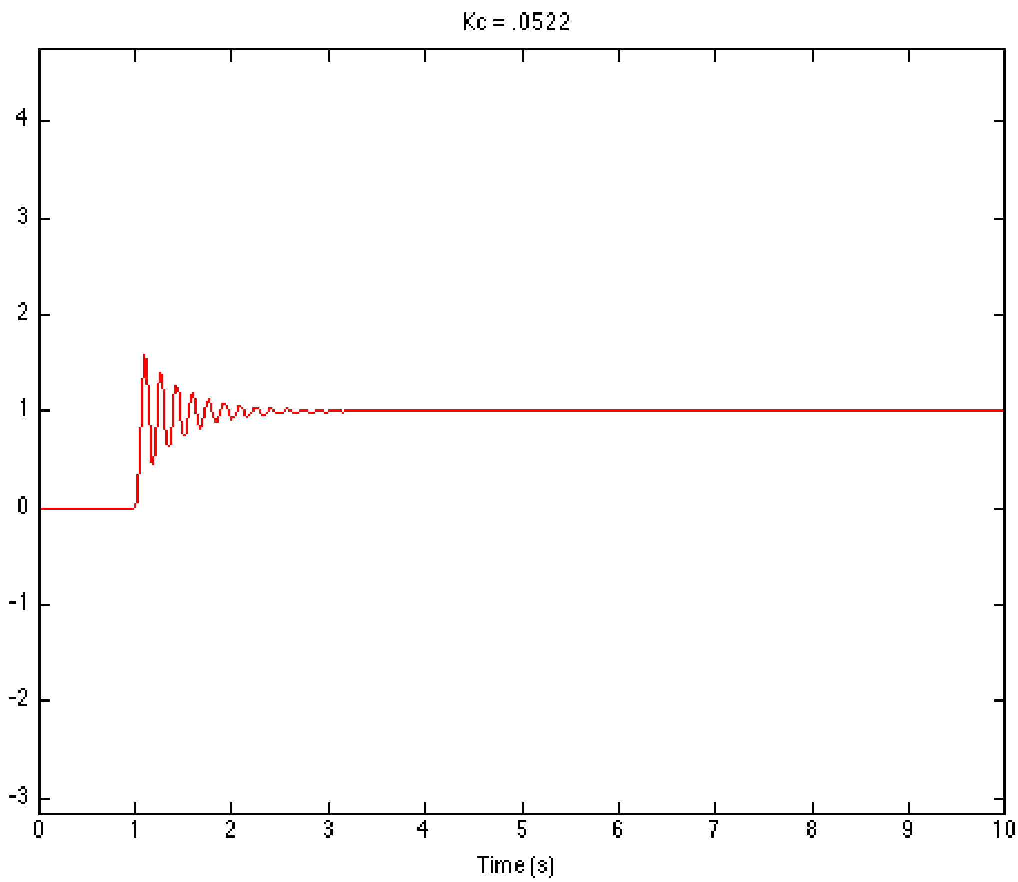
Figure 6: PID transfer function



Generated using Transfer function

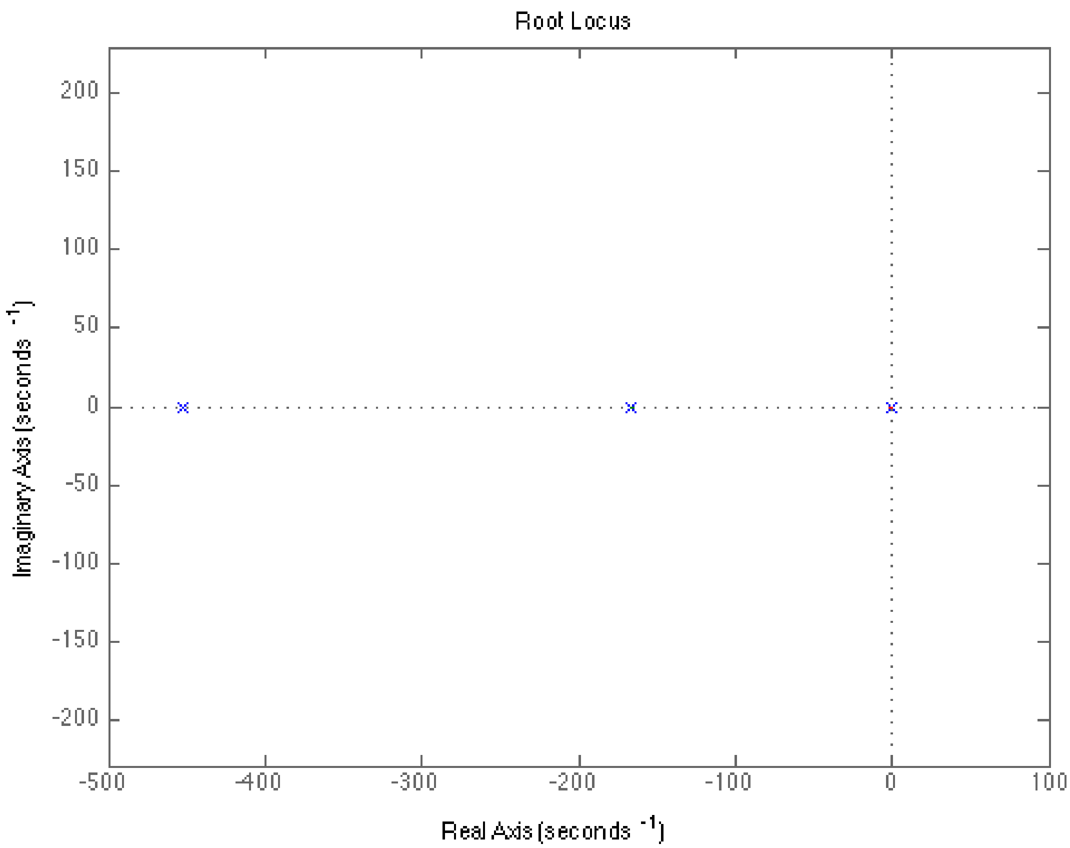
I was able to extract a K_I of .0533 but my damping constant remained the same

Figure 7: PID transfer function



Generated using Simulink

Figure 8: PID transfer function

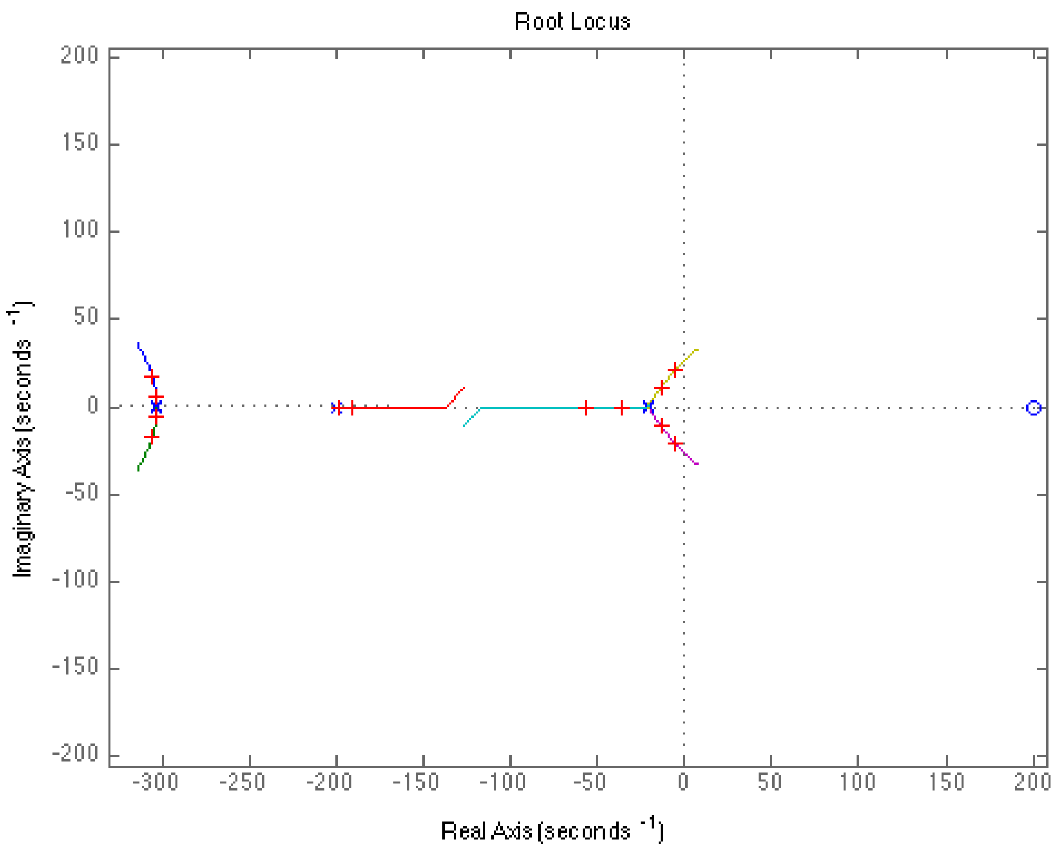


Generated using Transfer functionl

5 Neutral Stablilty using Simulink

I started this section by modelling in Simulink the black box system with the time delay. I added the transfer equation and formed a root locus graph. I noticed that it seemd that any value that I chose that stayed to the left of the real axis and was on the plot would result in a stable system for this function.

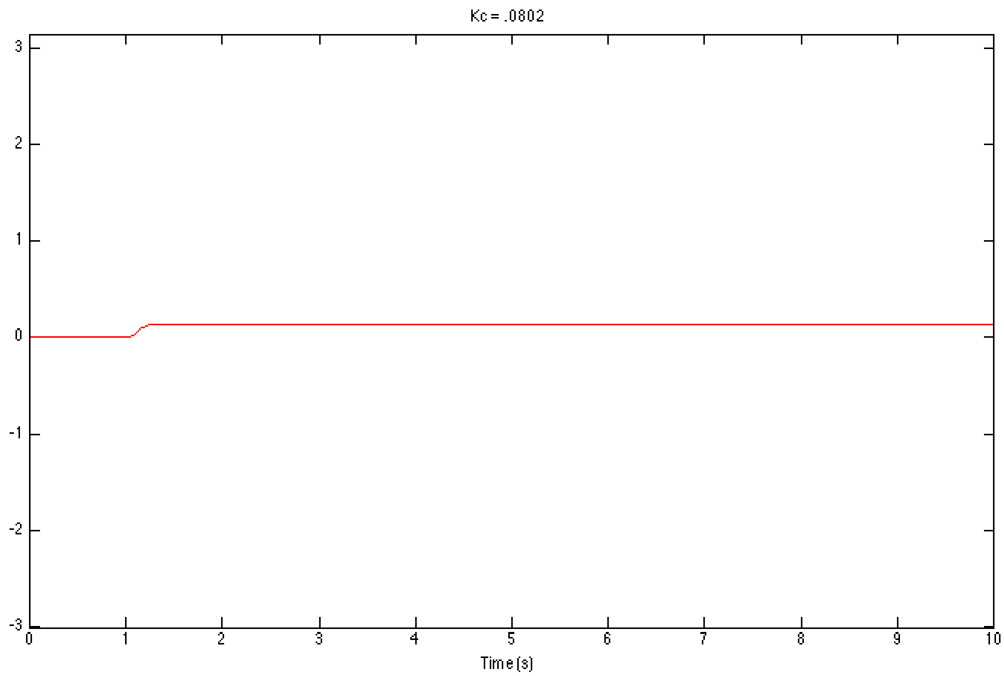
Figure 9: Time delayed function



Generated using Transfer functionl

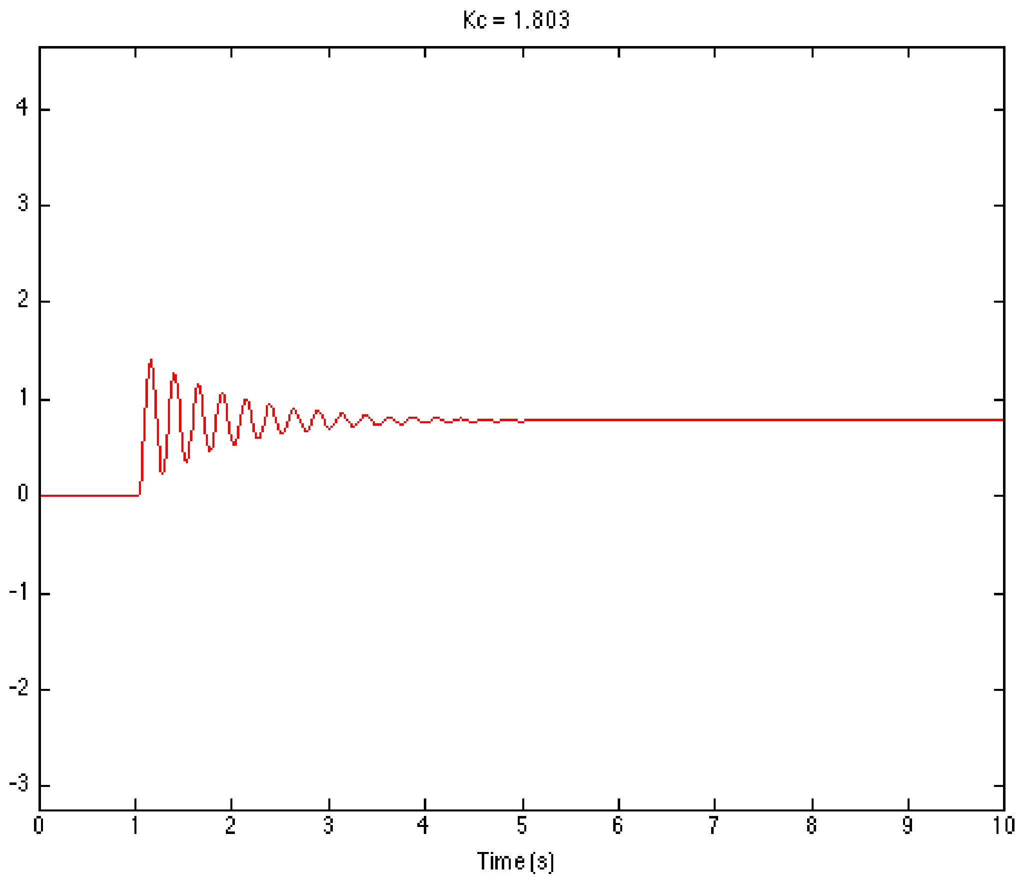
I noticed the signal was stable at lower values for K, but as long as it was left of the real axis and on the plot I was able to produce a stable graph for K values from .0802 up until 2

Figure 10: Time delayed function with K = .0802



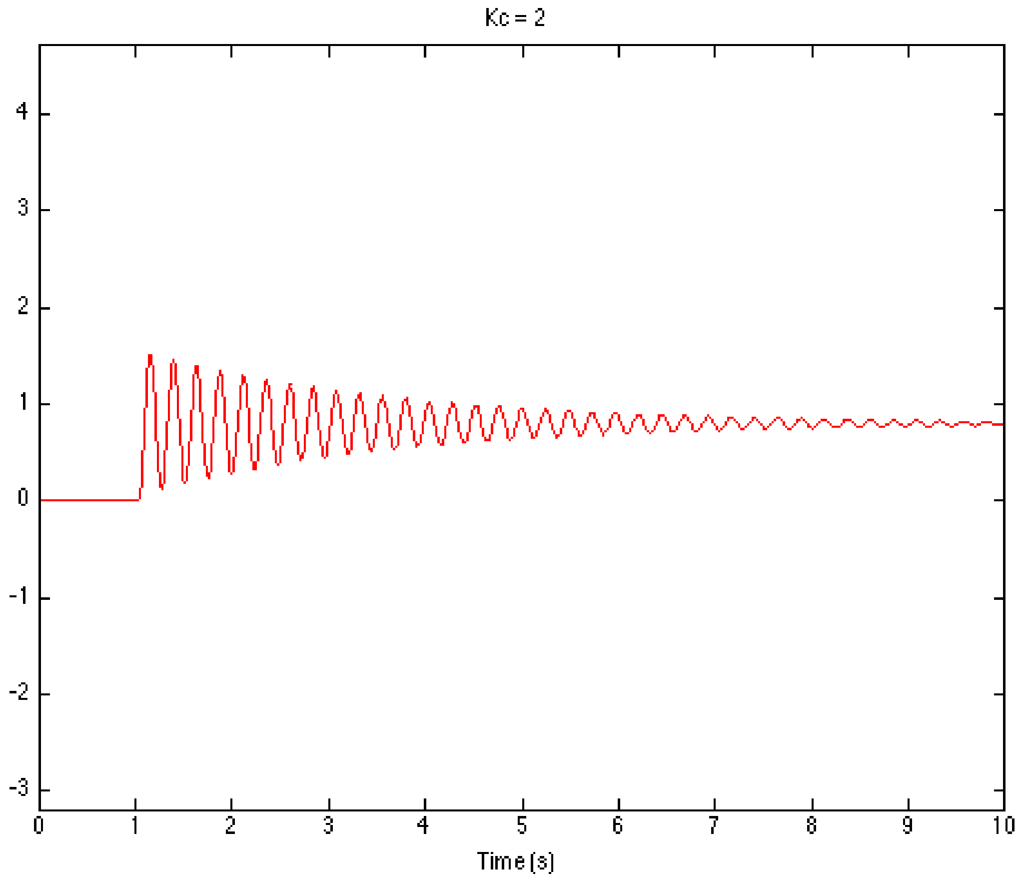
Generated using Transfer functionl

Figure 11: Time delayed function with K = 1.803



Generated using Transfer functionl

Figure 12: Time delayed function with K = 2



Generated using Transfer functionl