Problem Set 9 Solutions

Problem 1

a) $G_c(z) = \frac{U(z)}{E(z)} = \frac{K}{(z-1)}$, so: $u_{k+1} = u_k + Ke_k$, or equivalently: $u_k = u_{k-1} + Ke_{k-1}$ If e_0=0 and e_k=1 for all k>0, then:

k	$u_k = u_{k-1} + Ke_{k-1}$	u_{k-1}	e_{k-1}
0	0	0	0
1	0	0	0
2	K	0	1
3	2K	K	1
n	(n-1)K	(n-2)K	1

In general, $u_k = \sum_{i-1}^{k-1} Ke_k$. To compare, the other form yields:

$G_c(z) = \frac{U(z)}{E(z)} = \frac{Kz}{(z-1)}$, so: $u_{k+1} = u_k + Ke_{k+1}$, or equivalently: $u_k = u_{k-1} + Ke_k$

k	$u_k = u_{k-1} + Ke_k$	u_{k-1}	e_k
0	0	0	0
1	K	0	1
2	2K	K	1
3	3K	2K	1
n	nK	(n-1)K	1

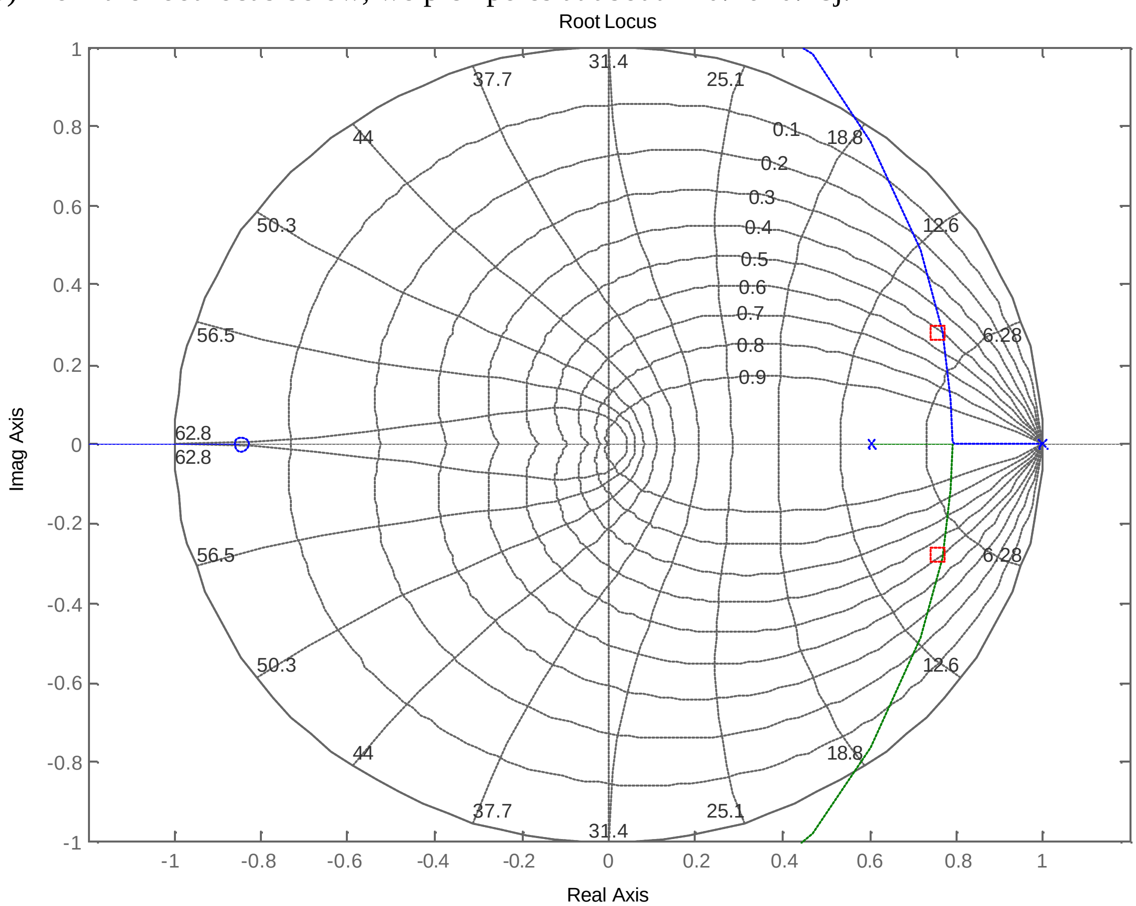
And here, $u_k = \sum_{i-1}^{k} Ke$. If we consider closing a loop where the forward path is simply $G_c(z)$ and the feedback is H=1, then **the steady-state error is the same (zero) for both systems**. However, the first system has only an open-loop pole at z=+1. As K is increased, this pole eventually becomes negative, and we can get an oscillatory (and even an unstable) output. The second system has the open-loop pole at z=+1 and ALSO has a zero at z=0. This bounds the closed-loop pole between 0 and 1 on the real axis, so the output never has an oscillatory response (and is never unstable).

Problem 2

a) $G_p(s) = \dfrac{10}{s(s+10)}$ and $G_{ZOH}(s) = \dfrac{1-e^{-Ts}}{s}$, for T=0.05 s

$$G_p(z) = \frac{0.01065x + 0.00902}{(z-1)(z-0.6065)}$$

b) From the root locus below, we pick poles at about z=0.76±0.28j.



This corresponds to zeta of about 0.5, and Ts=-4T/ln(r)=-4(.05)/ln(.81) = 0.95 sec. There is no steady state error, because $K_p = \lim_{z\to 1} G(z) = \frac{1}{0} = \infty$. (There is a pole at z=+1 in the denominator of G(z).) The value of K to put the poles at z=0.76±0.28j is about K= 6.9.

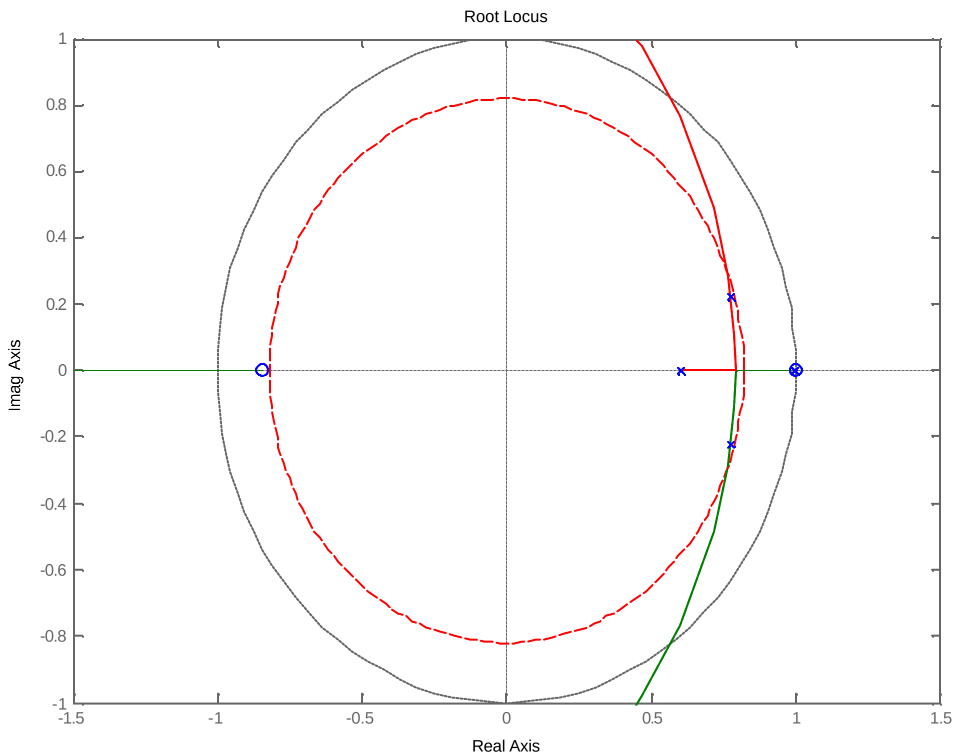
c) $K_v = \dfrac{1}{T}\lim_{z\to 1}(z-1)G(z) = 6.9$. $e_{ramp,ss} = \dfrac{1}{K_v} = 0.145$

d) For T=0.2s, Ts<=1 sec means ln(r)<(-4T/Ts), r<0.45. That is, we look for a location where the root locus is within a radius of 0.45. Since this never occurs for this new G(z), the design specs cannot be met.

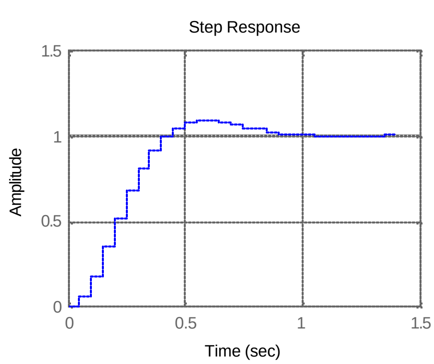
Problem 3

Using T=0.05 sec, we put another pole at z=1 to achieve zero steady-state error to a ramp input. One solution a the controller $C(z) = K\frac{z-.999}{z-1}$. Below is a root locus for the plant plus the controller. Our goal is a settling time under 1 second, which means the poles need to be within a radius of z=.82, and we also want zeta of at least 0.5.

$$G_p(z) = \frac{0.01065x + 0.00902}{(z-1)(z-0.6065)} \cdot K\frac{(z-.999)}{(z-`1)}$$



Selecting K=5 puts the poles at z=0.777±0.22j. Below is a step response:



Problem 4

a)Using partial fraction expansion: $\frac{G(s)}{s} = \frac{(s+3)}{s^2(s+1)(s+2)} = \frac{-0.25}{s+2} + \frac{2}{s+1} - \frac{1.75}{s} + \frac{1.5}{s^2}$

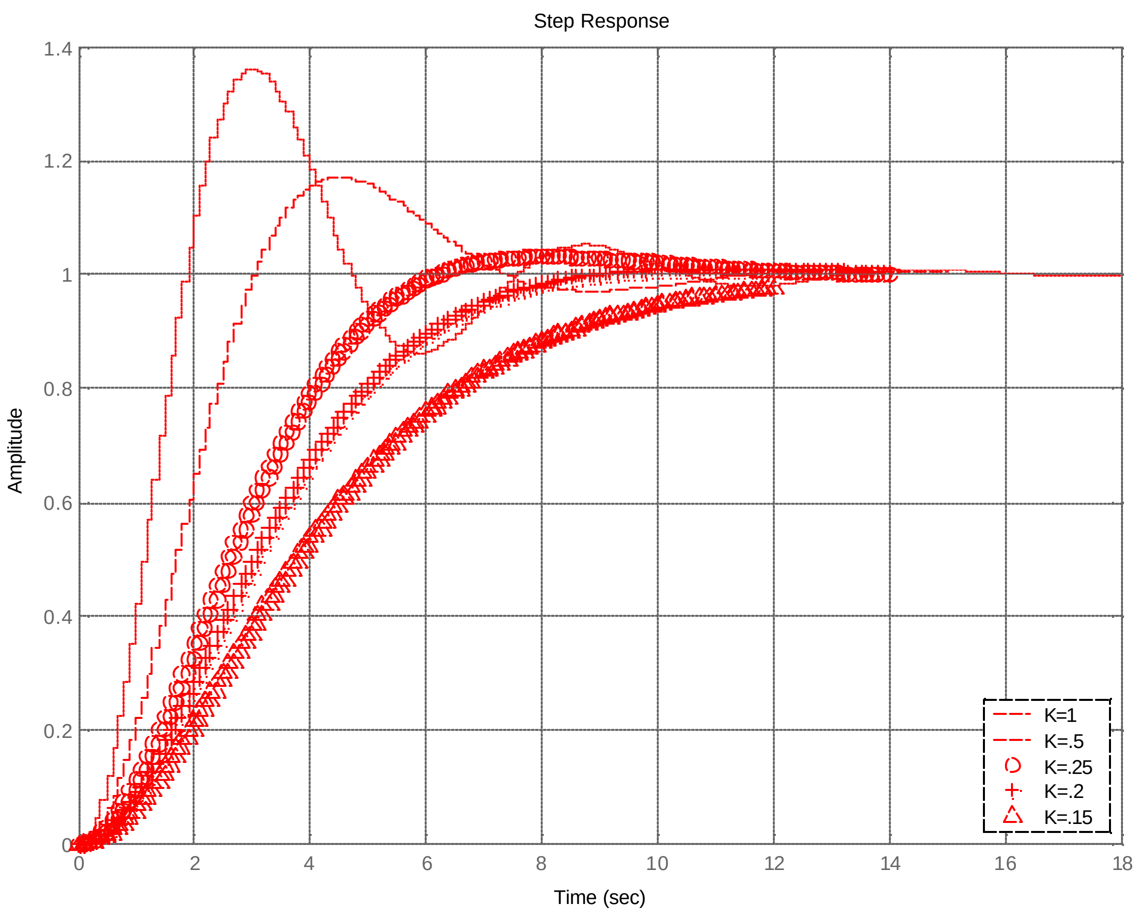
Including the ZOH, the equivalent G(z) will be:

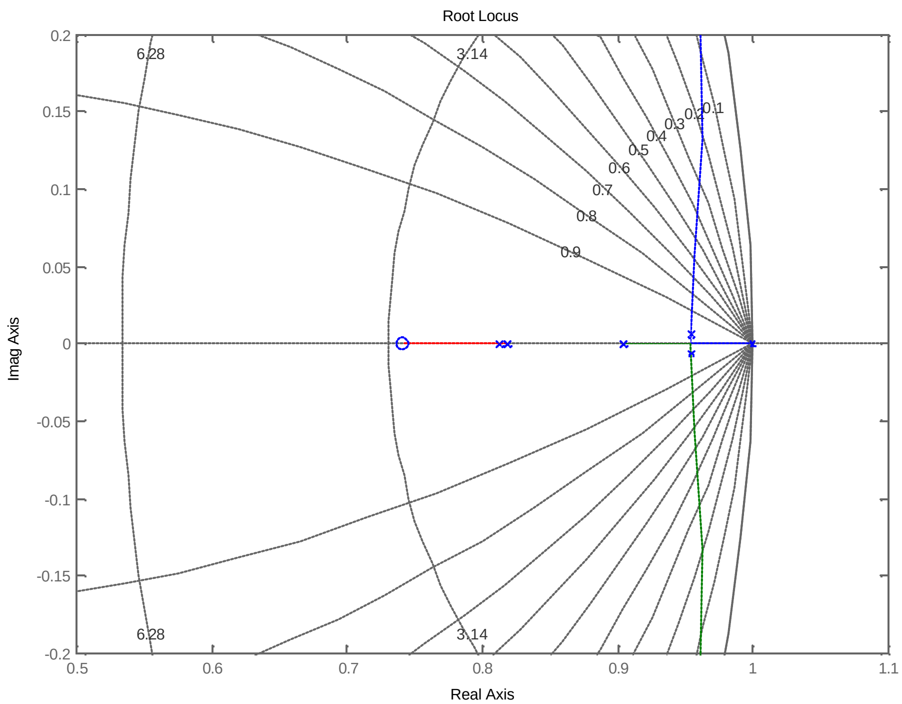
$$G(z) = \frac{z-1}{z} Z\left\{\frac{G(s)}{s}\right\} = \frac{z-1}{z}\left[-.25\frac{z}{z-0.819} + 2\frac{z}{z-0.905} - 1.75\frac{z}{z-1} + 1.5\frac{0.1z}{(z-1)^2}\right]$$

$$G(z) = \frac{0.004992z^2 + 0.001293z - 0.003698}{z^3 - 2.724z^2 + 2.464z - 0.7408}$$

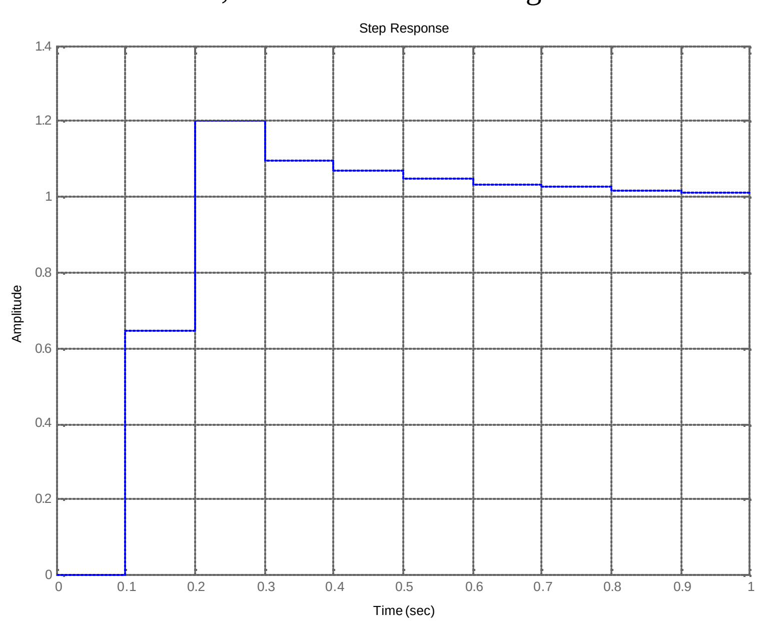
b) …which is identical to the answer MATLAB produces, using c2d.

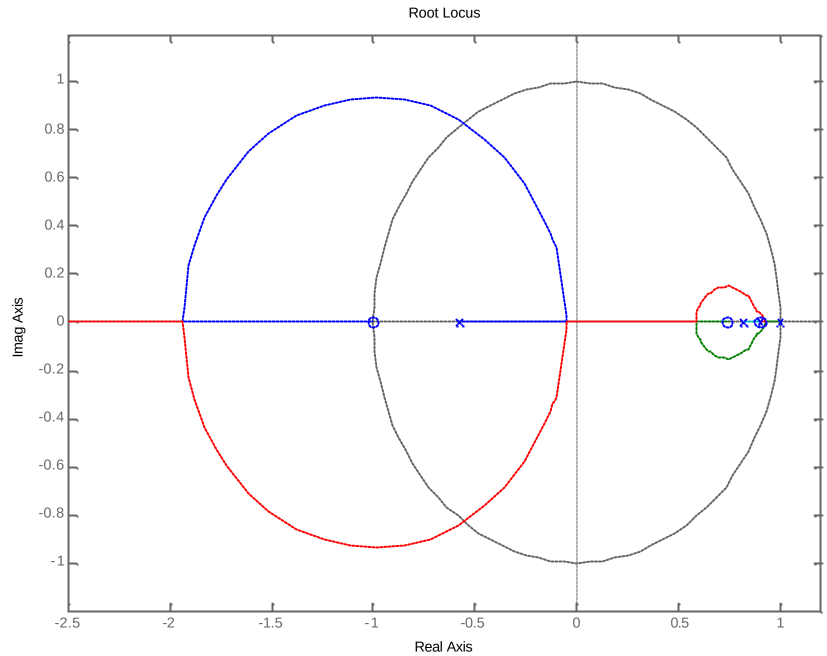
c) Looking at both the root locus and step responses (for several values of K), it looks like something around K=0.2 works "best". This has little overshoot and settles more quickly than smaller values of K will.

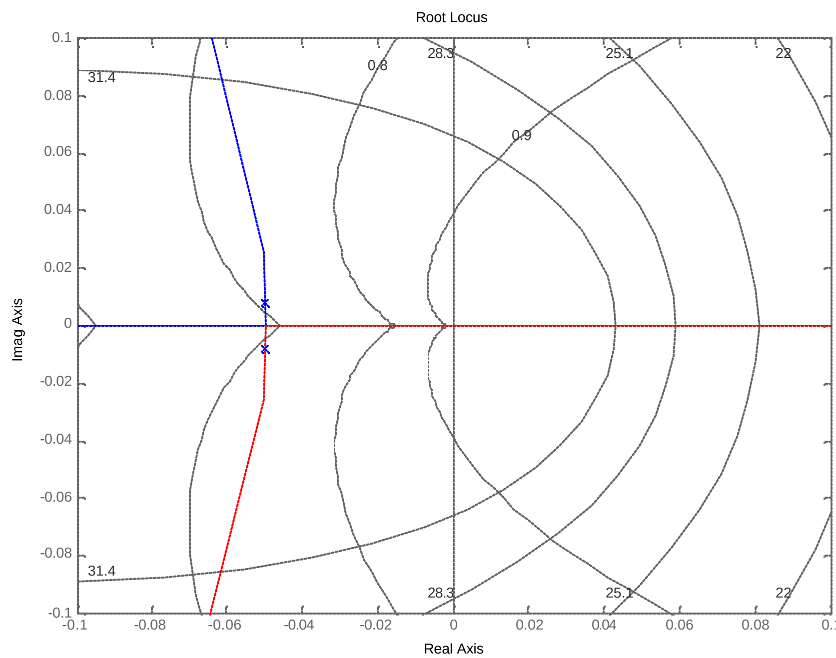




d) One reasonable controller design is to put one zero "near" z=+1 (but not at this location, as that would ruin our steady-state error), and then put a pole in the location that will create a "breakaway" at about -.05 on the real axis. (This corresponds to the "fastest" location on the zeta=0.7 line, approximately.) The "closer" we make that zero to z=+1, the more time it will take to get to the steady-state value, so I pick a reasonable value at z=0.9. The necessary pole location to get a breakaway at z=-.05 is around z=-0.58, and the corresponding K is about 129.6: $G_c(z) = 129.6\frac{z-0.9}{z+0.58}$. Below are a step response and root locus for this controller, which has a 2% settling time of about .8 seconds.







e) We cannot implement a controller where the order of the numerator exceeds the order of the denominator. If $\frac{Y}{X} = \frac{z+a}{b}$, for instance, then $y_k = \frac{x_{k+1} + ax_k}{b}$, and we would need to know a FUTURE value of x_{k+1} to determine the CURRENT value of the controller output, y_k.