EEE 481 Homework Problems

P.3.1

Consider the system $x(k+1) = 0.95x(k) + 0.05u(k)$, where the multiplications are quantized to 0.01. Use simulation to assess the mean, and variance of the error due to quantization (compared to non-quantized operations). Apply various inputs u(k), e.g., random, sinusoid, quantized to 0.01.

The multiplication quantization is modeled as random noise n(k) of uniform distribution ½ LSB. We will ignore addition. The system now is $x(k+1) = 0.95x(k) + 0.05u(k) + n_1(k) + n_2(k)$ .Then, the output contribution of that noise (x_n(k)) is described by the transfer function G(z)=1/(z-0.95) (the quantization occurs after taking the product 0.05u(k)) and is bounded as follows:

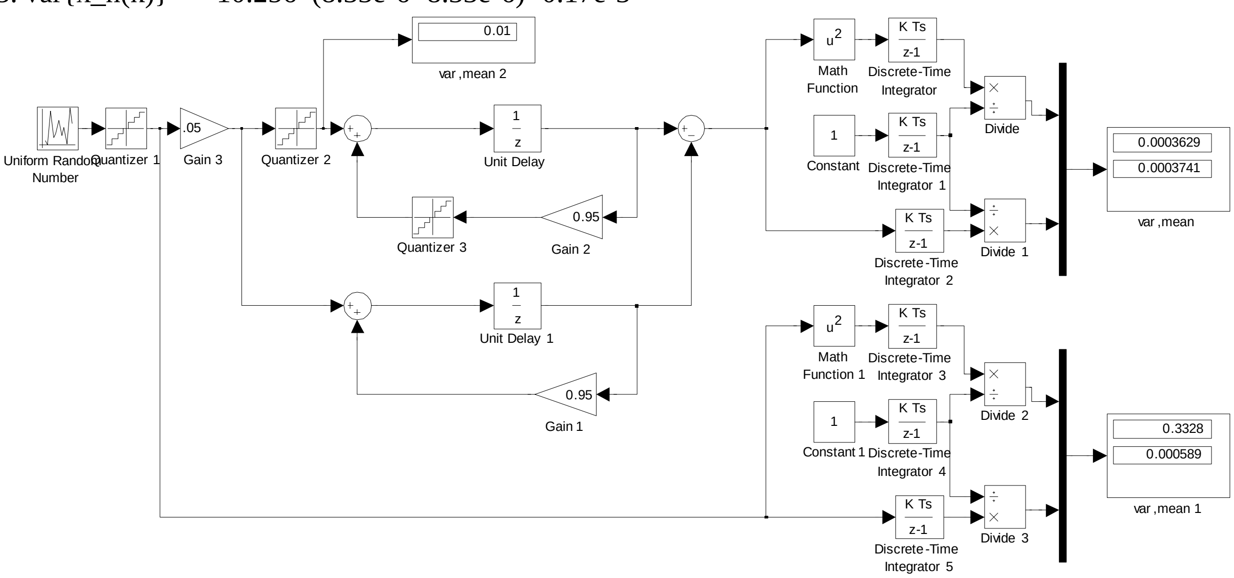
1. mean(x_n) = G(1)mean(n1)+ G(1)mean(n2)
2. var{x_n(k)} <= $|G(e^{j\Omega})|_2^2$var{n1(k)} +$|G(e^{j\Omega})|_2^2$var{n2(k)}

Notice that both quantization noises enter at the same node so there is only one transfer function.

Computing the theoretical estimates, G(1)=1/0.05=20. $|G|_2^2$ = sum$|g(k)|^2$=1/(1-0.95^2)=10.256. (The last one is based on Parseval, or simply using Matlab.)

For a round-off quantization, whose mean is 0 LSB, max(|n|)= ½ LSB = 0.005 and var = (½ LSB)^2/^{3} =8.33e-6. Thus,

1. mean(x_n) = 20*(0+0)=0
3. var{x_n(k)} <= 10.256*(8.33e-6+8.33e-6)=0.17e-3



The Simulink implementation of a simulator is shown above. We try different input signals: random numbers (uniform or gaussian), sinusoids, etc. In general, we must try different frequencies and different amplitudes as well, since the quantization makes the system nonlinear and the response to a scaled input is not simply the scaled response. The main results are tabulated below. Notice that, for stochastic inputs, the variance estimate is the least conservative one, but the estimate does not bound the actual signal, especially for low amplitude excitation where the nonlinearity is more prevalent (for input amplitude 0.05, the input to the quantized system is actually zero!). For slow sinusoids of small amplitude, the stochastic variance bound is too optimistic. This is where the more conservative norm(G,inf) estimate of the system gain becomes more appropriate. Also, it is possible that the mean is nonzero. In this case, the variance estimate should be adjusted (for simplicity, we skip the adjustment, keeping track of the mean to correctly interpret the results).

A priori estimate	Computed rand{-0.05,0.05}	Computed rand{-10,10}	Computed 0.05sin(0.001k)	Computed 10sin(0.001k)
var = 0.17e-3	0.022e-3	0.168e-3	1.28e-3	2.3e-3
mean = 0	-0.028e-3	0.32e-3	-0.07e-3	0.06e-3

As a last remark, the mean estimate becomes relevant if we use a different quantization scheme (floor, ceil) that have a nonzero mean. This estimate is fairly accurate.

P.3.2

Ziegler-Nichols Tuning: Apply the two Z-N methods from the notes to tune a PID for the plant $P(s)=\frac{5(-0.2s+1)}{s^2+3s+1}$. Compare the results with a PID designed for a gain crossover frequency of 2 rad/s and 50deg. phase margin.

*Hint: Define P as a transfer function object and use step(P) to get an estimate of R,L for the first Z-N tuning. Then iterate k on step(feedback(k*P,1)) until the system is marginally stable (slowly increasing or slowly decreasing response). Then estimate Ku,Pu for the second Z-N tuning. Define the compensators and compare step responses and bode plots for the transfer functions command-to-output and input disturbance-to-output*

```
P=tf(5*[-.2 1],[1 3 1])
step(P),grid
R=(2.4-.52)/(2.33-.8),L=0.37, %from graph, R=1.22
s=tf([1 0],1);
Kp=1.2/R/L;Ki=0.6/R/L/L;Kd=0.6/R;ZN1=Kp+Ki/s+Kd*s/(.01*s+1)
%use a fast pole for the pseudo-differentiator
step(fbk(P*5,1))
step(fbk(P*3,1))
step(fbk(P*2,1))
Ku=3,Pu=2.58-0.942,   %Pu=1.56
Kp=0.6*Ku;Ki=1.2*Ku/Pu;Kd=0.075*Ku*Pu;ZN2=Kp+Ki/s+Kd*s/(.01*s+1)

% design a pid using crossover/pm methods for a similar BW
[m,p]=bode(P*1/s/(.01*s +1),2)
Ph=-(p-360)-130
Tz=tan(Ph/2*pi/180)/2
C=tf(conv([Tz 1],[Tz 1]),[.01 1 0])
k=1/bode(P*C,2)
C=tf(conv([Tz 1],[Tz 1]),[.01 1 0])*k
%k=1.04, Tz=0.59,Ph=99.5
step(fbk(P*C,1),fbk(P*ZN1,1),fbk(P*ZN2,1))
bodemag(fbk(P*C,1),fbk(P*ZN1,1),fbk(P*ZN2,1))
bodemag(fbk(1,P*C),fbk(1,P*ZN1),fbk(1,P*ZN2))
```

Notice that both ZN methods yield much smaller phase margins than the classical design (20-30 deg). They do, however, offer smaller sensitivity at low frequencies without increasing the loop bandwidth too much. (They do increase the Sensitivity peak and resonance effect). The closed-loop test ZN is somewhat more reliable.