ELEC ENG 4CL4 – CONTROL SYSTEM DESIGN

Lab #5: Simulation of digital control within MATLAB/Simulink

Objectives:

To gain experience in digital-controller design and analysis within MATLAB/Simulink.

Assessment:

Your grade for this lab will be based on your ability to apply design methods for digital controllers and create Simulink models to simulate the control of continuous time plants, and on your reporting of the results. The report should contain pre-lab derivations and calculations and any other mathematical calculations carried out, schematics of your Simulink models with brief descriptions, MATLAB plots of results with brief descriptions, and answers to specific questions below.

The total grade will be out of 20 points. The written report will be worth 15 points, and the remaining 5 points will be based on your demonstration of your Simulink model and MATLAB code to the TA and/or a pre-lab quiz on the theory required for this lab.

Clearly label all plots and their axes (points for style will be deducted otherwise)

DON'T COPY OTHERS BLINDLY!!

Please attend the lab section to which you have been assigned.

You should complete this lab with one lab partner. If there are an odd number of students, then one group of three will be created by the TA, or you can choose to work on your own if a computer is available.

You may choose to complete the lab assignment partially or entirely in your own time (in groups preferably of two students but definitely no larger than three). However, if you choose to do this, you must show up at the start of your scheduled lab time to give the TA a brief demonstration of your MATLAB/Simulink code and model.

Each pair of students should complete one lab report together, which is to be submitted one week from the date of the lab.

1. Introduction:

The nominal model of a continuous-time system is given by:

$$G_o(s) = \frac{12}{(s+3)(s+4)}. \tag{1}$$

We wish to implement digital control of this plant via a sampled-data control loop of the form shown in Fig. 13.1 of Goodwin et al, with sampling interval $\Delta = 0.1$ second.

The discrete-time (shift form) transfer function $G_{oq}(z)$ (with $\Delta = 0.1$ second) that approximates the continuous-time plant model with a zero-order hold (ZOH) at the input is:

$$G_{oq}(z) = \frac{0.047687(z+0.7918)}{(z-0.7408)(z-0.6703)}. \tag{2}$$

In this lab we will compare the performance of three different digital-controller design methodologies.

a. The first digital controller design methodology is "Approximate Continuous Design"—see Section 13.5 of Goodwin et al. The basic idea of this technique is to carry out a normal continuous-time design and then map the resultant controller into the discrete domain.

   In this case, we can use the pole-assignment method to synthesize a continuous-time controller $C(s)$ with forced integration such that the closed-loop characteristic polynomial is $A_{cl}(s)=(s+2)^4$. The resulting continuous-time controller transfer function is:

$$C(s)=\frac{\frac{5}{12}s^2+\frac{5}{3}s+\frac{4}{3}}{s(s+1)}. \tag{3}$$

   We then convert the controller to a zero-order-hold discrete equivalent via the *step-invariant transformation* (with Δ = 0.1 second), giving the first discrete-time controller:

$$C_{1q}(z)=\frac{0.41667(z-0.8956)(z-0.7082)}{(z-1)(z-0.9048)}. \tag{4}$$

b. The second controller design methodology is the "Minimal Prototype" for "At-Sample Digital Design"—see Section 13.6.2 of Goodwin et al. The basic idea of this strategy is to achieve zero error at the sample points in the minimum number of sampling periods for step references.

   Applying this methodology to the discrete-time approximate plant model $G_{oq}(z)$ given above, we obtain the second discrete-time controller:

$$C_{2q}(z)=\frac{20.9701(z-0.7408)(z-0.6703)}{(z+0.7918)(z-1)}. \tag{5}$$

c. The third controller design methodology is "Minimum-Time Dead-Beat Control"—see Section 13.6.3 of Goodwin et al. The idea of this method is the same as the "Minimal Prototype" case, with the added requirement that for step references the controller output $u[k]$ reach its steady-state value in the same number of intervals that it takes to obtain zero error at the sample points. This prevents ringing in the continuous-time plant output between the sample points.

   Applying this methodology to the discrete-time approximate plant model $G_{oq}(z)$ given above, we obtain the third discrete-time controller:

$$C_{3q}(z)=\frac{11.7034(z-0.7408)(z-0.6703)}{(z+0.4419)(z-1)}. \tag{6}$$

2. Pre-lab

a. Show that the continuous-time controller given by Eq. (3) above, when placed in a one-d.o.f., unity-feedback control loop with the plant model given by Eq. (1), produces the closed-loop characteristic polynomial $A_{cl}(s) = (s+2)^4$.

b. Show that the "Minimal Prototype" discrete-time controller given by Eq. (5) above, when placed in a one-d.o.f., unity-feedback control loop with the discrete-time plant model given by Eq. (2), makes the discrete-time plant output be the discrete-time reference signal delayed by one sample.

c. Show that the "Minimum-Time Dead-Beat Control" discrete-time controller given by Eq. (6) above, when placed in a one-d.o.f., unity-feedback control loop with the discrete-time plant model given by Eq. (2), makes the discrete-time plant output and the controller output go to their steady-state values after two samples, i.e., for $k \geq 2$.

3. Analysis of discrete-time control of the approximate discrete-time model

a. Create three different Simulink models by placing the approximate discrete-time plant model given by Eq. (2) in a one-d.o.f. unity-feedback loop with each of the three different discrete-time controllers given by Eqs. (4)–(6).

   Note that the model blocks need to have their "sample time" parameter set to 0.1 second for this discrete-time implementation.

b. Test out the control loops' response to step reference changes and to sinusoidal reference signals with various frequencies. Generate MATLAB plots to compare the results obtained with the three different controllers and confirm that the controllers are working correctly for the discrete-time approximate plant model.

c. Describe the similarities and differences in the performance of the three different controller designs.

   Do they appear to achieve their design criteria, at least when connected to the discrete-time approximate model?

   What are the apparent advantages and disadvantages of each?

4. Analysis of discrete-time control of the continuous-time model

a. Create three new Simulink models by placing the *continuous-time* plant model given above in a one-d.o.f. unity-feedback loop with each of the three different discrete-time controllers given by Eqs. (4)–(6).

   Hint #1: You will need to use a ZOH block with the "sample time" parameter set to something like 1e–3 second to give an "approximately continuous-time" input to the continuous-time plant model.

   Hint #2: You can use a ZOH block with the "sample time" parameter set to 0.1 second to obtain the sampled-output signal from the continuous-time plant model that is required for the feedback signal. Note that the closed-loop bandwidth is less than the Nyquist frequency and we are not considering the effects of any disturbances, so it is not necessary to include an anti-aliasing filter.

b. Test out the control loops' response to step reference changes and to sinusoidal reference signals with various frequencies. Generate MATLAB plots to compare the results obtained with the three different controllers.

c. Describe the similarities and differences in the performance of the three different controller designs.

   Do they achieve their design criteria?

   What are the apparent advantages and disadvantages of each?