**SY202 – Cyber Systems Engineering**

**Intro**

**CSE**

**Due Date: 19 February 2019**

**LABORATORY INVESTIGATION #04: Simulation of PID Control**

**Objectives**

* Reinforce familiarity and use of MATLAB and Simulink as simulation and analytical tools for the study of cyber-physical systems.
* Explore and understand the use and effect of PID controllers.
* Modify the time response of a system using PID control.
* Explore the effect of external disturbances in a cyber-physical system.

**Introduction**

In this lab, the students will simulate the speed control of a motor-wheel system using MATLAB and Simulink. The students will first compare the use of open-loop vs. closed-loop control with and without external disturbances. Then, the students will explore and compare the effect that each control gain in a Proportional-Integral-Derivative (PID) controller has on the closed-loop time response of the system. The comparison criteria will be based on transient and steady-state behavior. At the end, the students will attempt to design and validate via simulation a PID controller according to some performance specifications.

**Procedure**

**PART A: Understanding the Control System and Simulation Files**

1. Create a working folder in your Laptop for this lab. From the course’s GAfG drive folder, download Lab04\_MATLAB\_Script.m and PID\_Control\_Wheel.slx files to your folder.
2. Open the PID\_Control\_Wheel.slx and explore the simulation diagram. There are two transfer functions representing the dynamics of the wheel (plant) and the motor (actuator). The transfer function of the sensor has been assumed to be 1, meaning that the sensor’s speed reading is 100% accurate with no added delays or errors.
3. Note that there is a conditional switch to change the diagram between open-loop and closed-loop control. You can manipulate the selection by setting CM=0 for open-loop and CM=1 for closed-loop within the script file Lab04\_MATLAB\_Script.m.
4. Open (double click) the PID controller subsystem block and explore the proportional, integral, and derivative actions. Go back to the main diagram.
5. There are two step inputs. One represents the desired or reference signal and comes before the summing junction that generates the input to the controller. The other represents a constant, unexpected disturbance to the wheel, such as a sudden constant inclination of the terrain. The disturbance has been placed between the transfer functions of the motor and wheel.
6. Note that three variables are being saved, the wheel’s speed or output (“y”), the control signal or effort (“u”), and the error signal (“error”).
7. Open the Lab04\_MATLAB\_Script.m file and explore each section and comments. Note that the system parameters are defined and plotting functions for data illustration have been created.

**PART B: Simulation of Wheel’s Speed Control with no Disturbances**

1. Open Lab04\_MATLAB\_Script.m. Make sure the reference signal “r” is set to 10 rev/sec. Set the control mode to open-loop by setting CM = 0. In addition, set KP = 1, KI=0, and KD = 0 to pass the reference signal intact to the motor transfer function. Make sure that the disturbance value is set to zero (Disturbance = 0). This simulation correspond to the system in open-loop control mode.
2. Run Lab04\_MATLAB\_Script.m. It should open a window with the simulated response, including the wheel’s speed, the error signal, and the control signal in volts. If the signal did not settled properly, you can change the simulation time by changing tfinal in the script. In your MATLAB’s command window, you should also see a display of the performance parameters. Using these values, fill out the first row in Table 1 from the Appendix.
3. Now, set the control mode to closed-loop by setting CM = 1. Set your control gains to KP = 0.5, KI=0, and KD = 0. Run the file. Adjust the time of your simulation if necessary and fill out the second row of Table 1 in the Appendix.
4. Simulate the response in closed-loop for all other controllers in Table 1. Make sure that in all, the reference signal is 10 rev/sec and that the disturbance value is set to zero.
5. Compare the response of all controllers and try to identify trends as the control gains were increased.

**PART C: Simulation of Wheel’s Speed Control with Disturbances**

1. In the MATLAB script, set the control mode back to open-loop (CM=0). Make sure the reference signal is 10 rev/sec. Set the value of the Disturbance = -5. Set the time of disturbance tdisturbance = 8. Set KP = 1, KI=0, and KD = 0.
2. Run the MATLAB Script and fill out Table 2 in the Appendix.
3. Set the control mode to closed-loop (CM=1). Run the MATLAB script for each controller in Table 2 and fill out the table.
4. Draw conclusions/trends.

**PART D: Design of PID Control**

1. By means of trial-and-error, design a closed-loop controller of your choice (P, PI, PD or PID) such that:

* Steady-state error is less than 1.0%
* Reject constant disturbances
* Rise time is between 1.0-2.0 seconds
* % OS is less than 5.0%
* Settling time is less than 4.0 sec
* Control effort does not exceed 50 Volts at any given time

1. Validate your design by simulating your control system to a step input (reference signal) of 10 rev/sec with no disturbance. Include a plot to your report illustrating the response of your system, including the control signal, and discuss results. You may use the same plot generated by the given MATLAB script (use: Edit 🡪 Copy Figure, from top menu and paste into your report).
2. Validate your design against disturbances by repeating step 2 with a Disturbance= -5 at tdisturbance = 8. Note that for this step, you only need to make sure that the system is able to go back to desired steady-state error despite the disturbance (in general, you will not be able to meet other specifications such as settling time under disturbances). Include the plot of the system response in the report and discuss results.

**Deliverables**

Follow the lab report template and the general lab guidelines for SY202 lab reports. Refer to the lab rubric for the grading of the lab.

**APPENDIX: TABLES**

Table 1. Simulation Results with No Disturbance

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Controller  # | Control Type | Control Gains | Rise Time  (sec) | Overshoot | Settling Time  (sec) | Steady-State Error  (rev/sec) | Steady-State Control Effort (V) |
| 1 | Open-Loop |  |  |  |  |  |  |
| 2 | P |  |  |  |  |  |  |
| 3 | P |  |  |  |  |  |  |
| 4 | P |  |  |  |  |  |  |
| 5 | PI |  |  |  |  |  |  |
| 6 | PI |  |  |  |  |  |  |
| 7 | PI |  |  |  |  |  |  |
| 8 | PD |  |  |  |  |  |  |
| 9 | PD |  |  |  |  |  |  |
| 10 | PD |  |  |  |  |  |  |
| 11 | PID |  |  |  |  |  |  |
| 12 | PID |  |  |  |  |  |  |
| 13 | PID |  |  |  |  |  |  |
| 14 | PID |  |  |  |  |  |  |

Table 2. Simulation Results with Disturbance

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Controller  # | Control Type | Control Gains | Steady-State Error  (rev/sec) | Steady-State Control Effort (V) |
| 1 | Open-Loop |  |  |  |
| 3 | P |  |  |  |
| 4 | P |  |  |  |
| 5 | PI |  |  |  |
| 7 | PI |  |  |  |
| 9 | PD |  |  |  |
| 12 | PID |  |  |  |