Controlling with Controllers

Padron, Sebastian & Reyes, David

Sections 12563 & 12588 March 7th, 2024

***Abstract*—The primary objective of this lab report is to use a bang-bang controller, proportional controller, a PID controller and a full-state feedback controller on a motor/flywheel system to modulate the difference from the actual motor rotation to the desired motor rotation, or the error. The most stable system was shown to be the bang-bang controller with a sinusoidal wave input with an error of 4.17 degrees and the least stable system was also the bang-bang controller for a square wave input with an error 103.84 degrees. The proportional controller displayed a more stable system with a reduction of error with an increasing proportional gain value for square wave input signals. The Full-state feedback controller displayed an error of 74.12 degrees, with a percentage overshoot of 0.078%, a rise time of 0.804 seconds and a settling time of 1.173 seconds. Of the two methods that were tuned in this report, the manual tuning method was less stable with an error of 61.89 degrees, a percent overshoot of 0.2%, a rise time of 0.491 seconds and a settling time of 0.175 seconds.**

***Index Terms*— Bang-bang controller, PID controller, Pole placement**

I. Introduction.

THE introduction of a controller to a motorized system leads to a reduction of the difference between the desired output and the actual output. This concept will be explored through the implementation of three different controllers to regulate a motor/flywheel system, a bang-bang (BB) controller, a proportional-integral-derivative (PID) controller, and a full-state (FS) feedback controller. The primary goal of the report is to test these to minimize the error, $e(t)$, or the difference between the actual motor rotation, $\theta_{actual}(t)$, and the desired motor rotation, $\theta_{desired}(t)$, as shown.

$$e(t) = \theta_{desired}(t) - \theta_{actual}(t) \qquad (1)$$

The motor/flywheel system used in this lab will be the same motor/platen system that was built and utilized in the previous report.

Bang-Bang Controller

Considered to be the simplest control scheme amongst the selection of feedback controllers, the bang-bang follows a simple control law that regulates the level of command effort set, $u(t)$, by alternating from two polarizing states [1]. The state is dependent on whether the error is positive or negative as shown in (2).

$$u(t) = \begin{Bmatrix} +c, & e(t) > 0 \\ -c, & e(t) < 0 \end{Bmatrix} \qquad (2)$$

PID Controller

The proportional-integral-derivative controller, the most common control scheme, regulates $u(t)$, improving the steady-state errors and the transient response of the system [2]. It does this through the continuous calculation shown in (3) which involves the multiplication of the proportional, integral, and derivate control gains, or K_P, K_I, and K_D, respectively. These control gains will be multiplied by the error, the integral of the error, where T represents the amount of time over which the error integral will be computed, and the derivative of the error.

$$u(t) = K_P e(t) + K_I \int_{t-T}^{t} e(t)dt + K_D \frac{d(e(t))}{dt} \qquad (3)$$

A manual tuning method will later be used to reduce the rise time, overshoot percentage, and steady-state error will be applied and elaborated in Methods.

FS Controller

The full-state feedback controller will regulate $u(t)$ with the implementation of Ackerman's method or pole-placement routines that multiply the error and derivative of the error with the two gains, K_1 and K_2, which are determined by the desired percent overshoot and settling time on the system [1]. The expanded control law for this control scheme is shown in (4).

$$u(t) = -\left(K_1 e(t) + K_2 \frac{d\big(e(t)\big)}{dt}\right) \qquad (3)$$

Lab Procedure

With the ultimate goal of implementing these controllers to reduce the error within system, a series of test cases giving the response of the motor system will be recorded. There will be 14 unique conditions of which one full cycle will be selected to compute both the error and motor control signals. A table containing the mean, standard deviation, and root mean square (RMS) for both the error and command values of every test case will be generated and displayed in Results.

These test cases will include the use of a BB controller with the varying effort levels for both square and sinusoidal waves, a PID controller with varying proportional gains for both square and sinusoidal waves, a modified PID controller that underwent the manual tuning method for a square wave signal, and a FS controller that used the pole-placement method in MATLAB for a square wave signal.

II. Methods

A. Equipment Preparation

For this lab, the required equipment includes a laptop installed with LabVIEW and MATLAB, , a 9-volt power supply, a multi-function DAQ and a built Furuta Pendulum system with an integrated DC motor. This motor/flywheel device was constructed using the “Pendulum v2.1 Assembly” instructions which were provided in the previous lab.

B. Software Preparation

Once the assembly of the motor flywheel system has been set up, a LabVIEW virtual instrument (VI) is built to record and manipulate the data related to the rotation of the motor. This lab will continue to utilize the signal generator that was added to the standard lab software in the previous report. It produces square and sinusoidal wave input signals for the motor; however, it is modified to compute a desired angle of the motor, or the amplitude, rather than just the commanded effort.

The VI was configured with two Boolean switches to select the type of input wave signal that will be used and the type of controller that will be used for any given test case. Along with this, the VI will also accept various user inputs which will further diverge each test case from each other, such as the values for K_P, K_I, K_D, and K_{BB}, the system gain for the BB controller which correlates with the effort level of the controller.

The BB and PID controllers will also be added onto the control framework VI, where the errors are defined as the difference between the desired value coming from the signal generator and the measurement of the motor rotation from the encoder. Depending on the controller selection, the aforementioned gain values will affect the performance of the motor, where K_{BB} will control the effort level of the motor for only the bang-bang test cases as shown in Fig. 1.

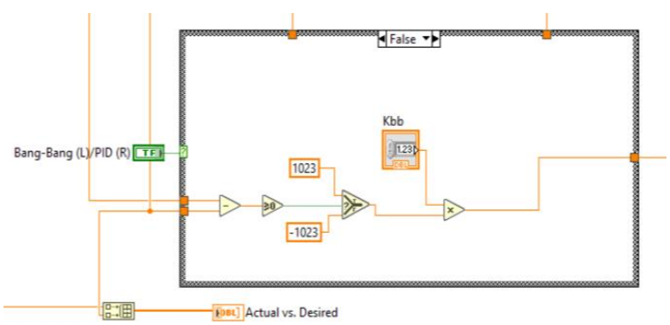

Fig. 1. Bang-bang controller configuration in LabView.

It is significant to note that a separate configuration for the FS controller is not necessary as it only involves position and velocity states which is multiplied by the user-inputted gain values and summed as shown in (3). The PID controller configuration shown in Fig 2. can work as the FS controller, if the states are redefined to be the position and velocity errors, then with K_P representing K_1 and K_I representing K_2. Moreover, this PID controller will also undergo manual tuning to conduct the last test case that reduces overshoot percentage, steady-state error and rise time.

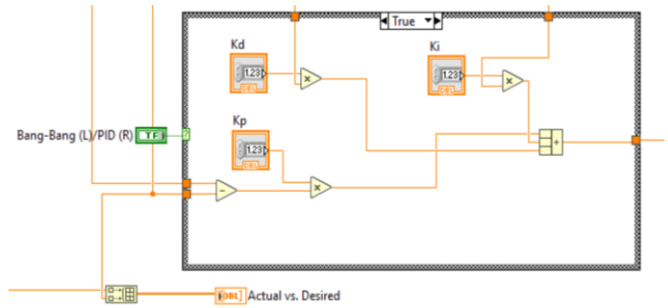

Fig. 2. Proportional-Integral-Derivative controller configuration in LabView.

The various test cases will have constant input values throughout all of them. These include the previously determined value of 69 for the dithering compensation (i.e., Turn On Command), which will continue to provide a smooth open-loop sinusoidal trajectory, a standard value of 0.25 Hz for the frequency and an amplitude of 110 degrees (Instructed: 90+3+8+6+3 = 110 degree) with no offset. This is done for the comparison and analysis of each the test case and their results.

This LabVIEW VI will provide an output that includes the elapsed time, the desired rotation angle, the actual rotation, the command input signal, the frequency, the difference of errors, and all the gain values for a given test to differentiate them. This data will go through filtration to remove any excessive noise and saved into a spreadsheet that can later be exported onto Microsoft Excel, or any data manipulation software.

C. Data Collection

After verifying that the VI is functional through the different controller configurations, the collection of data for the 14 different test cases can commence. It is significant to note that all cases will supply enough data to provide accurate results, therefore, the standard for this report will require at least four complete cycles through every configuration.

The first three tests will record the response of a motor system with a square wave input attached with a BB controller where each case will record the data at 25%, 50 %, and 100% controller effort level. In these cases, all gain levels will be set to zero, except for K_{BB} where it will be set to 0.25, 0.5 and 1.0 respectively. The next three tests will also use square wave inputs but instead use the PID controller with K_I and K_{ID} set to zero. The goal here is to treat the controller as just a proportional controller and observe how the rotation response changes with the increasing gain. In these cases, K_P will be set at the levels 0.75, 2.0, and 5.0. Afterwards, these six tests will be repeated with the exact configurations excluding the input wave signal. This is adjusted to send sinusoidal input signals.

The 13th case will send square wave input signals on modified PID controllers that will use the manual tuning method to find the arrangement of gains to provide the lowest possible rise time with a maximum overshoot percentage of 10% and a reduction in steady-state error (SSE) [3]. This procedure begins with all gains set to zero, where K_P is slowly increased until the response begins to display oscillations occurring. K_P is then halved and K_I will be increased until the SSE becomes negligible. Afterwards, K_D is increased until the response gives a percent overshoot (P.O.), of 10% or less [3].

The final case the PID controller will be used to act as a FS controller as previously mentioned. Here the signal generator will send square wave signals and use pole-placement to give a step response for a 0.25 Hz, square wave that will have a 2% settling time, T_s, less than 0.3 seconds and a P.O. of 5%. In order to determine the poles, we must solve and define the percent overshoot and settling time. The percent overshoot compares the actual rotation of the motor at steady state, θ_{actual}, to the maximum rotation value, θ_{max} as shown in (4). The setting time is the difference between the time in which the step response begins to oscillate within 2% of the steady-state value, $T_{2\%}$, and the time in which the step response started, T_i as shown in (5).

$$P.O. = \frac{(\theta_{max} - \theta_{actual})}{\theta_{actual}} \times 100\% \tag{4}$$

$$T_s = T_{2\%} - T_i \tag{5}$$

After calculating for these values, the location of the poles, x, can be determined by first determining the damping ratio of the motor/flywheel system as shown in (6).

$$\zeta = \sqrt{\frac{(\ln(P.O))^2}{\pi^2 + (\ln(P.O))^2}} \tag{6}$$

Then we must solve for the natural, ω_n, and damped frequency, ω_d, as shown in (7) and (8).

$$\omega_n = \frac{4}{\zeta \cdot T_s} \tag{7}$$

$$\omega_d = \omega_n \sqrt{1 - \zeta^2} \tag{8}$$

$$x_{1,2} = -\zeta\omega_n \pm j\omega_d \tag{9}$$

Lastly, we can calculate for the location of the poles as shown in (9). From here these values are put into the MATLAB code in Fig. 3. Within this code K_{DC} and tau, references the transfer function that was determined in the previous report, where they act as the DC gain and the time constant for the transfer function. Following the acquisition of the poles, they are joined by two matrices that will use K_{DC} and tau to convert the system into a state space system and put into the Ackermann function in MATLAB to provide the gains needed for the controller.

```
clc;clear;

Kdc = 6.3904;
tau = 0.0424;
PO = 5;
Ts = 0.3;
zeta = abs(log(PO/100))/sqrt(pi^2+(log(PO/100))^2);
omega_n = 4/(zeta*Ts);
omega_d = omega_n*sqrt(1-zeta^2);
P = [-zeta*omega_n+1i*omega_d -zeta*omega_n-1i*omega_d];

A = [0 1; 0 -1/tau];
B = [0; Kdc/tau];

K = acker(A,B,P)
```

Fig. 3. Ackerman's method calculated in MATLAB.

D. Response Analysis

For each test case, a single full cycle of data will be selected to find the absolute value mean (μ), standard deviation (σ), and root mean square (RMS), for both the error and the motor signals. The RMS for error utilizes the number of measurements (N) and the squared value of each measurement in a single full cycle, as shown in (10). For the RMS of the command signal, similar variables are used, but with the subscript CS, as shown in (11).

$$RMS_e = \sqrt{\frac{1}{N}\sum_{i}^{N}(e_i)^2} \tag{10}$$

$$RMS_{CS} = \sqrt{\frac{1}{N}\sum_{i}^{N}(CS_i)^2} \tag{11}$$

Further, to quantify the performance of the PID controllers, 2nd order characteristics for the square wave will be performed and placed in Table II in the Results.

III. Results

The statistical values of the error and the command signal for all the 14 tests conducted are shown in Table I. For test 13, the PID gain values obtained from the manual tuning process were 4, 0.85, and 0.125 for K_P, K_I, and K_D, respectively. For test 14, Ackermann's method yielded PID gain values of 2.4768, 0, and 0.0204 for K_P, K_I, and K_D, respectively.

TABLE I
Statistical Observations for Error and Command Signal

#	Test	μ_e	σ_e	RMS_e	μ_{CS}	σ_{CS}	RMS_{CS}
1	BB 0.25	55.05	90.74	90.74	184.83	198.37	198.36
2	BB 0.5	34.49	69.47	69.54	511.50	511.51	511.50
3	BB 1.0	78.81	103.84	103.93	1023.0	1022.6	1023.0
4	P 0.75	95.02	102.53	102.40	71.26	76.90	76.80
5	P 2.0	53.48	79.37	79.37	106.96	158.75	158.74
6	P 5.0	22.42	56.40	56.40	85.96	223.85	223.83
7	BB 0.25*	3.51	4.17	4.17	255.75	255.75	255.75
8	BB 0.5*	6.42	7.35	7.35	511.50	511.52	511.50
9	BB 1.0*	51.11	57.58	57.58	1023.0	1022.7	1023.0
10	P 0.75*	66.88	74.48	74.48	50.16	55.86	55.86
11	P 2.0*	44.22	49.18	49.18	88.45	98.37	98.36
12	P 5.0*	20.55	22.98	22.98	102.77	114.92	114.92
13	PID	29.33	61.91	61.89	103.82	223.59	223.58
14	FS	44.02	74.12	74.12	83.24	177.35	179.17

Test numbers are consistent with references in text. Tests including a * symbol denote the use of a sinusoidal input, if not it is a square wave input.. Units for error are degrees and units for the command signal are bits.

Furthermore, using second-order characteristics, the performance of each controller configuration that makes use of a step input is quantified to clearly compare each controller. Table II displays the percent overshoot, rise time, and settling time for each test. Certain second-order characteristics do not apply for some tests, for example, tests 1-3, which make use of a BB controller which has no settling time as it continuously surpasses the ±2% range of the final value. Moreover, tests 4-5 have no percent overshoot as the motor signal was delayed enough to not reach the input command signal.

TABLE II
SECOND ORDER PERFORMANCE METRICS

#	Test	$P.O.$ (%)	Rise Time (s)	T_s (s)
1	BB 0.25	4.913	0.401	-
2	BB 0.5	11.453	0.196	-
3	BB 1.0	81.242	0.101	-
4	P 0.75	-	1.165	1.290
5	P 2.0	-	1.006	1.397
6	P 5.0	0.023	0.305	0.494
13	PID	0.200	0.491	0.715
14	FS	0.078	0.804	1.174

Showcasing the different controller configurations tested, Fig. 4 shows the different outputs produced by each controller with the same command signal input.

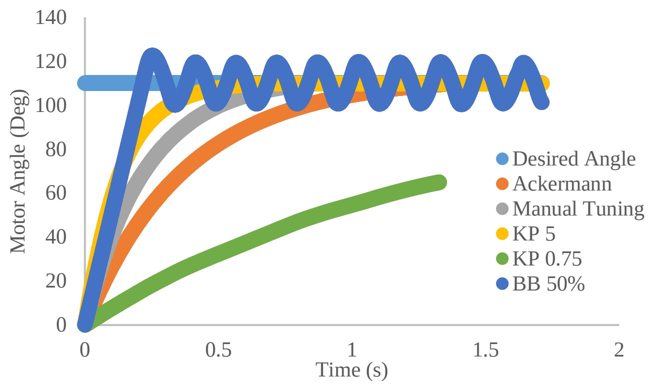

Fig. 4. Plot displays the use of different controller configurations and their respective outputs corresponding to the same command signal as a step-input.

IV. DISCUSSION

The overall objective of this report was to compare and analyze the responses and performances of a motor/flywheel control system with a BB, PID, Proportional and FS controllers. In this section of the report, comparisons on the results of the different controllers will be compared.

A. *Comparison of the Methods*

Through observation, the BB controller with a step-input, there seems to be an irregularity as the error of the system appears to decrease until the controller effort is switched to 100%, causing the error of the system to increase. When the controller effort was set to 50%, the system had the lowest error (69.54 deg) for the BB controller configuration with a step-input. Therefore, the controller effort should be tuned between 25-50% effort. When command signal is switched to sinusoidal-input, as the controller effort is increased, the error of the system heavily increases. As the controller effort for the bang-bang controller configuration with a sinusoidal-input is set to 25%, the error of the system (4.17 deg) is the lowest of all tests. Thus, in this case, the controller effort should be tuned between 0-25% effort. In terms of second-order characteristics, the percent overshoot exponentially increases as the controller effort is increased, with the bang-bang controller configuration with 100% effort having the largest percent overshoot of all tests. Also, there is an almost perfect quadratic relationship between controller effort and rise time for the bang-bang configuration. With the controller effort increasing at an increasing rate, the rise time decreases at a decreasing rate. As stated before, settling time does not apply to the bang-bang controller configuration as the data does not settle within the ±2% range of the final value at any point of the full cycle selected.

In contrast, the proportional controller displays a decrease in error with an increasing controller effort for both step- and sinusoidal-inputs. Error is less prominent in the configuration that makes use of a sinusoidal-input with an averaged difference of 30.51 deg between step- and sinusoidal-input tests with the same controller effort. For tests 4-5, the motor's rotation at no point stabilized and therefore there was no recorded excess past a stabilized value, causing the inability to calculate percent overshoot. Nonetheless, for test 6, the percent overshoot happened to be the lowest of all tests. Further, both the rise time and settling time decrease with increasing controller effort. Based on the low percent overshoot, low rise time and low settling time, test 6 is a suitable controller configuration for ideal controller performance.

The tuned systems performed average in terms of error, which did not behave as expected. This is most likely due to an error in hardware or data acquisition, therefore, in future experiments, more tests will be recommended. Both tests 13-14, which made use of the tuning processes, performed better than the proportional controller test 5 but worse than the proportional controller test 6, staying consistent with the observation that test 6 is a more ideal set-up. Comparing tuning processes, Ackermann's method provided a better percent overshoot but worse rise and settling times. The manual tuning method presented a 2 times shorter rise time and a 1.5 times shorter settling time. It is also important to note that the pole placement denoted by Ackermann's method did not provide the desired percent overshoot (5%) and settling time (0.3s) stipulated before the data collection.

Additionally, for tests 1-12, the command signal increases with increased controller effort, as expected.

B. *Limitations and Recommendations*

As with any real-life experiment, improvements can provide more precise results through decreasing the amount of mechanical friction between the different moving parts within the system. The motor/flywheel system used in this lab will hold natural limitations (i.e., friction) in the hardware, generating discrepancies and inaccuracies, subsequently creating excess noise and disturbances with the acquisition of data. There is also a lack of trials to determine the most effective controller configuration. With more trials, there would be more accurate results.

REFERENCES

[1] S. Ridgeway, "EML 4314C - Spring 2024 - Lab 2 Assignment Revision" in EML4314C, Dynamics and Control System Design Laboratory, Mar-2024

[2] S. Ridgeway, "EML 4314C Spring 2023 - Lecture 7 slides" in EML4314C, Dynamics and Control System Design Laboratory, Mar-2024

[3] J. Brooks, "L13_02_21_24" in EML4314C, Dynamics and Control System Design Laboratory, Mar-2024