Lab 1: System Identification

Domokos, Lorant

Group 49 Date 2/11/2023

***Abstract*—The goal of this lab was to find first order transfer functions for the motor platen system in lab using sine sweep and step response methods and to compare the results with analytical representations of those systems. By observing the signal magnification and phase shift at different frequencies of sine wave input signals, and by observing the step response of the system, DC gain and time constant values were used to create transfer functions of the system. The results showed that a fist order approximation is not entirely representative of the system and that relatively consistent transfer function values can be derived based on whether the system time delay is considered or not.**

I. Introduction

The first step in the development of a controlled system is the identification of the behavior of the dynamics of that system. This behavior of the system is often represented as a transfer function, which relates the ratio of the output to the input signal. The transfer function of a system can be derived analytically based on the a priori knowledge of the physical properties of the system or empirically by observing the output of the due to different inputs. Analytically, transfer functions are derived using mathematical techniques applicable to the system, in chemical plant control systems that may be reaction rate relations, for electrical systems it may electrical dynamics and for mechanical systems it may be dynamics and kinematics. However, analytical models often do not offer an accurate representation of the true behavior of the system due to uncertainties in physical properties as well as other non-linear factors. While uncertainties can be reduced and non-linear dynamic properties can be accounted for, it is often more economical to approximate the system as a lower order linear system with empirically derived constants. Empirical techniques are also often more scalable for identifying many similar systems.

Two different empirical methods were used for system identification and their transfer functions were independently compared to their analytical transfer function behaviors as well as to each other. The first empirical technique is the sine sweep, where the input signal is modulated as sine wave of constant amplitude but varied through a range of wave frequencies. Since the system is assumed linear, it is expected that the system output would follow sine wave output of equal frequency to the input at steady state. The time delay and amplitude ratio between the input and input and output sine waves changes with different input frequencies. These time delays and amplitude changes were used to develop a bode plot for the system which was then used to find the first order system pole. The second empirical technique is the step response analysis. A step response is input and the output response time and rise time are used to develop the transfer function.

The goal of this lab is to develop a first order transfer function for the class provided motor/platen system using empirical techniques and comparing the empirical system behavior to the behavior of an analytically derived system with the same transfer function.

II. Procedure

A. *Building the Motor/Platen system*

The motor platen system was constructed using instructions provided [1]. The system consists of a brushed DC motor with a hall effect encoder and a 50:1 gear reduction, attached to an additional 6:1 3D printed gear reduction and mounted to a 3D printed fixture. The Platen is fixed coaxially to the second gear of the 6:1 reduction and allowed to rotate easily on lubricated ball bearings. The DC motor is powered by a 9V power supply and motor controller which is controlled by a DAQ, connected to a laptop running the class provided LabVIEW VI framework.

B. *LabVIEW VI setup*

1) *Command Signal Generation*

The provided LabVIEW VI handles reading and writing data to and from the DAQ and then to the motor controller. To simulate the sine wave and step input command (cmd sig), signal generators were created such that they may be toggleable. The command input is an integer between -1023:1023 corresponding to a PWM signal used to vary to DC motor input voltage. Due to the static friction in the system, the motor struggled to start at low PWM values, so a Turn-on Command constant, cmd_{TO}, was added to the PWM command value. This constant was tuned while a sine wave command was given, until the output position showed a smooth wave output at a value of 65. The motor controller was connected to a 9V DC power supply so the motor input voltage can be calculated using the command signal sum, PWM resolution and input voltage V_i.

$$V_i = \frac{\text{cmd sig} + \text{cmd}_{\text{TO}}}{1023} * 9\,Volts \quad (1)$$

2) Encoder Ratio Correction

The couts/rev variable was also changed from 3704.4 to 3600 to correctly align with the encoder counts per revolution. The couts/rev value was calculated as the product of the 12 encoder counts per motor revolution, 50:1 motor gear reduction and 6:1 platen spur gear reduction for a total of 3600 encoder counts per platen revolution.

3) Data Filtering and Numerical Derivative

Encoder data and encoder counts per platen revolution were used to calculate the platen angle. The platen angle data was passed through a lowpass Butterworth filter to smooth out the data. A backwards finite difference approximation across the last 20 position points of smooth data was used to calculate the angular velocity of the platen.

4) Data Output

Ultimately the VI output data for time, command signal, filtered platen rotation angle (degrees), platen velocity (degrees/sec), amplitude (PWM), generated wave frequency (Hz), and a binary digit corresponding to the output of a sine (0) or a square (1) wave.

C. Data Collection

To perform the sine sweep data collection, the LabVIEW Turn-on command was set to 65, and the amplitude was set to 350 after advice from a TA, the platen rotation and velocity were zeroed, and the Butterworth filter was initialized. The frequency was set to 0.05 Hz and the system was started. A minimum of five periods were recorded for each of the following frequencies: 0.05, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 7.5, 10, 15 and 20 Hz.

To collect square wave data, the frequency was set to 0.2 Hz to ensure that a steady state angular velocity is reached before the direction of rotation changes.

D. Sine Sweep Approach (bode_plot.ipynb)

1) Data Processing

The sine sweep data was loaded as a .csv into a Python Pandas dataframe using Jupyter Notebook. Dataframe columns were added angular velocity in radians/sec, as well as motor voltage input using (1). The input frequencies were also converted to radians/sec and added to a new column. The platen velocity, ω_p, and motor voltage, V_i, were plotted for the whole range of input frequencies, to assess anomalies.

The data for each input frequency was separated from the whole dataset one at a time by manually iterating through a list of the input frequencies (freq_data). The first and last 150 datapoints of freq_data were dropped to avoid interference from transient phenomena when switching between frequency inputs. The ω_p and V_i columns of freq_data were plotted to visually ensure clean data.

Large peaks were observed in velocity were observed in low frequencies (< 0.05 Hz) likely due to gear cogging and static friction, so a backwards finite difference approximation across the last 40 data points was calculated and plotted to filter the velocity. Data rows with anomalous acceleration were labeled as outliers, and dropped from the dataset. The resulting filtered velocity ω_p was plotted against the input V_i.

2) Magnitude Amplification and Phase Shift Extraction

To find the amplitude and phase shift of velocity, ω_p, and voltage, V_i, as sine function was fit to the data using least squares. The least squares fit functions were plotted to visually compare the fit to empirical data. Magnification was calculated as the decibel ratio of the velocity amplitude A_{ω_p}, to voltage amplitude A_{V_i} (2) in units of radians/(seconds*Volts).

$$M_p = 20 * log_{10}\left(\frac{A_{\omega_p}}{A_{V_i}}\right) \quad (2)$$

The phase shift was calculated using the time delay between the zero-crossing times between the input signal, V_i, and the output signal, ω_p. The zero crossing times were found and listed for both signals. The least squares function plots were used to assess whether one signal had more zero crossings, and the lists were truncated to make sure the list of zero crossings for both signals were aligned. The time delay Δt, between the crossing times of the two signals was calculated (3) and listed.

$$\Delta t = t_{V_i} - t_{\omega_p} \quad (3)$$

The list of Δt time delays were averaged, $\mu_{\Delta t}$. The phase shift, ϕ, was calculated as the angular change in degrees that occurred during the mean time delay given the input frequency in radians/seconds ω_i (4).

$$\phi = \mu_{\Delta t} * \omega_i * \frac{180°}{2\pi} \quad (4)$$

The magnitude and phase shift values for each input frequency were plotted on a Bode plot. To extract the first order transfer function DC gain, k_{DC}, and time constant, τ, from the bode plot. The pole frequency, ω_p, was visually estimated at a phase shift of 45 degrees. The DC gain was calculated as the output input ratio (5) at this frequency and the time constant was calculated as the inverse of the interpolated input frequency (6).

$$k_{DC} = 10^{M_p/20} \quad (6)$$

$$\tau = \frac{1}{\omega_p} \quad (7)$$

E. Step Response Approach (step_response.ipynb)

1) Data Processing

The square wave data was loaded as a dataframe and the same column additions were made as for the Sine Sweep data and the collected data was visualized. The first step response was isolated by creating a subset dataframe (first_step) starting when the input signal first switches to a positive voltage, to when the output signal appears to reach a steady state. The input and output signals were plotted for visual confirmation. The step response showed a system time delay likely due to processing latency t_{sys} which was recorded and removed from the first_step dataframe.

2) DC Gain and Time Constant Feature Extraction

The low and high output signal steady state angular velocities, $\omega_{high}, \omega_{low}$ were recorded, as well as the low and high input voltages V_{high}, V_{low}. The ratio of the angular velocity change to the voltage change was used to calculate the DC gain, k_{DC}, of the system (8).

$$k_{DC} = \frac{\omega_{high} - \omega_{low}}{V_{high} - V_{low}} \tag{8}$$

The first order time constant, τ, was calculated using the 63.2% rise time, which is the time it takes for the output signal to reach 63.2% of its steady state value (9). The pole frequency for the system can be found as the inverse of the time constant (10).

$$\tau = 0.632 * \left(\omega_{high} - \omega_{low}\right) \tag{9}$$

$$\omega_p = \frac{1}{\tau} \tag{10}$$

First order transfer functions (11) were used to represent the system.

$$H(s) = \frac{k_{DC}}{\tau s + 1} \tag{11}$$

Three transfer functions were created using the gain and time constant values from the sine sweep and step response tests. Two transfer functions were produced from the step response data, one which accounts for the system time delay and one which does not. Step responses and bode plots were then plotted in MATLAB for the representative first order transfer functions.

III. Results

A. Sine Sweep Results

The results for the eleven different input sine wave frequencies were calculated and plotted on a Bode plot as shown in Figure 1. The magnitude and phase values were calculated using (2) and (4). The second part of Figure 1 shows the traces that were used to estimate the first order DC gain by finding where the system has a phase shift of 45 degrees and tracing the values for M_p and ω_n and using (6), (7) to find the transfer function constants.

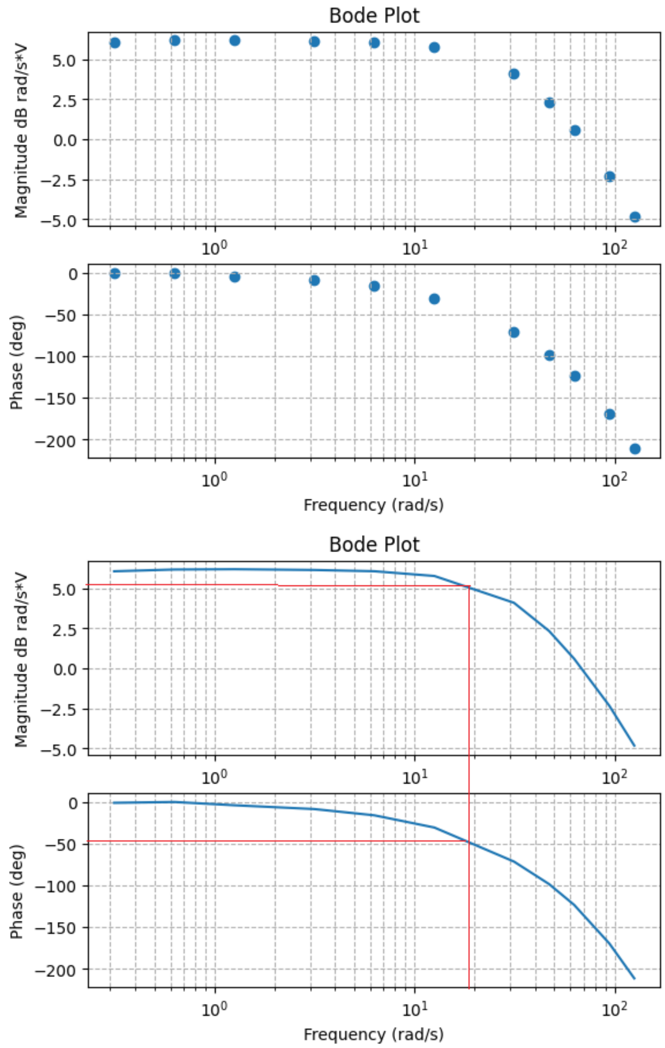

Fig. 1. Bode plot for the Sine Sweep Results. Top graph indicates the data points collected. The bottom plot shows the same data points but with the pole location marked as well as the traces used to find M_p, which was used to find the $H_1(s)$ DC gain.

The step response of the system is shown in Figure 2. The short horizontal section of the step response is the system latency. The time length of this section was used to develop the distinct time delay adjusted and non-adjusted transfer functions.

The first order pole location, DC gain and time constant for the three calculated system properties are shown in Table 1.

TABLE I
DC Gains and Time Constants

Test	Latency Corrected	Pole Location (rad/s)	k_{DC} (rad/sV)	τ (seconds)
Sine Sweep	No	18	1.94984	0.055556
Step	Yes	34.4793	1.83117	0.029003
Step	No	23.2493	1.83117	0.043012

Rad = radians, s = seconds, V = volts.

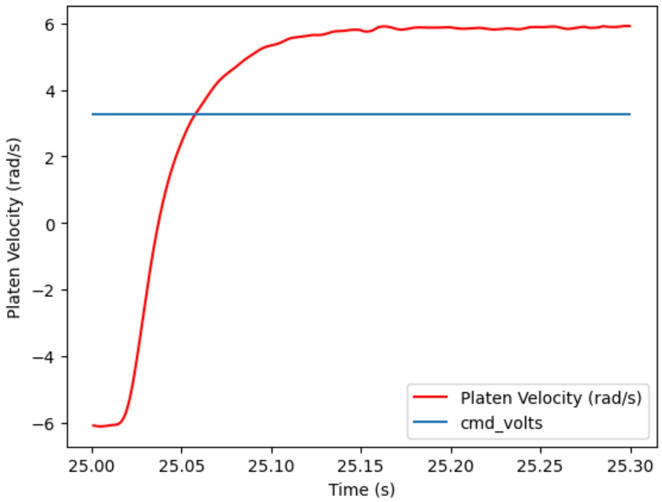

Fig. 2. Step response of the platen system. The horizontal section starting at 25s is the system latency that was dropped before measuring calculating the time constant.

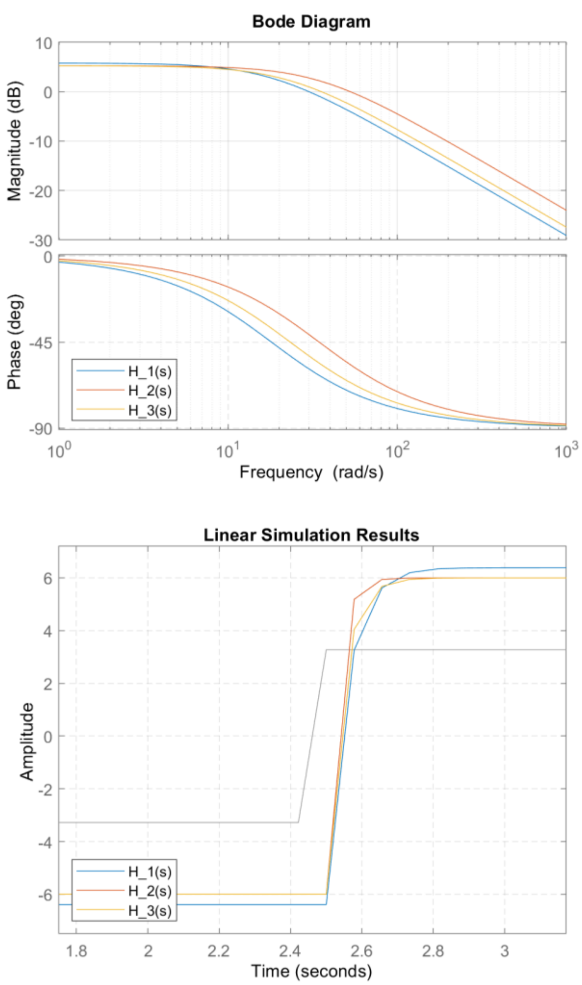

Fig. 3. Simulated Bode plots and step response for the system transfer functions (top). The bottom plot shows the simulated step response generated using the MATLAB gensig() and lsim() commands.

Using these constants, the following three transfer functions were derived (12-14) $H_1(s)$ corresponds to the sine sweep test results and $H_2(s)$ and $H_3(s)$ correspond to the delay adjusted

$$H_1(s) = \frac{1.94984}{0.055556s + 1} \tag{12}$$

$$H_2(s) = \frac{1.83117}{0.029003s + 1} \tag{13}$$

$$H_3(s) = \frac{1.83117}{0.043012s + 1} \tag{14}$$

and non-adjusted transfer functions respectively. The bode plots and step responses for (12-14) were subsequently plotted in MATLAB using the bode(), gensig() and lsim() commands, producing the plots shown in Figure 3. The gensig() and lsim() commands were used instead of the step() command in order to show a proper apples to apples comparison of the input signal jumping from negative to positive instead of starting out with an input signal of zero.

Analyzing the time constant values and the gain values for the three transfer functions, it is clear that the results are most consistent between $H_1(s)$ and $H_3(s)$ showing that the time delay was clearly not accounted for in the sine sweep transfer function. These results are also supported by the step response of the system, where $H_1(s)$ and $H_3(s)$ show similar rise time behavior.

IV. Discussion

The objective of this study was to empirically find the first order transfer functions for the lab platen system, simulate the transfer function in MATLAB and compare the measured response of the system with the simulated transfer function response.

The results of the study indicate that a first order transfer function is a poor approximation of the system. The bode plots indicate that the true system exhibits behavior that is more characteristic of as second order system. Nonetheless the consistency of the transfer function constants indicates that the first order system estimates may be accurate enough at low input frequency sine waves. The simulated step responses also seem to show a faster settling time than the real system, further indicating that a higher order system transfer function would be more appropriate.

The study results are limited by the predictive ability of a first order transfer function to approximate a higher order system.

V. Conclusion

The results of this study show that both a sine sweep approach and step response approach can be used to estimate the first order transfer function of the lab motor platen system. This is supported by the similarity of the transfer functions derived based on both methods. However, it would be wise to explore higher order transfer function models for this system in order to get a system estimate more consistent with the empirical findings.

References

[1] Out of the Box ER. (2024, Jan 12). Pendulum v2.1 Assembly. [Online].