5 Post-lab Analysis (Week of March 26-30)

This section should be done independently outside lab. Please see Sect 5.2 and run the MATLAB Code on your data as soon after lab as possible! If there are problems with the data, we will help you solve them.

The main point of this lab was to measure the impulse and stochastic response for a mass-pring system where the input is force on the system (as applied by current in a coil) and the output is vertical acceleration. In the post-lab, you will perform a full analysis in order to learn how you can determine system parameters and compare your non-parametric (measured) Bode plots to those computed from a 2^{nd} order system model.

Please keep in mind throughout the course and your future career the important distinction between "**measured**" and "**calculated**". Results derived from measurements, even if calculations are done to convert the measurements into another form, are "measured" (or, if you prefer, "computed" or "derived").

"Calculated" refers to results computed directly from a model or from first principles, i.e. not resulting from a measurement (although sometimes models may include parameters that are experimentally determined, as in the present case). As an example, the frequency response (Bode plot) you generated from *tfestimate* in MATLAB in the last lab is a **measured** frequency response, even though it is "calculated" (computed) from the measured acceleration impulse data.

5.1 Overview of System Identification Analysis Methods

System Identification methods typically fall in one of two categories: those that compute the frequency response function (Bode plot) in the **frequency domain**, and those that compute the impulse response function in the **time domain**. Recall that the frequency response function and the impulse response function are Fourier transform pairs, so that determining one allows you to immediately determine.

Both approaches are extremely powerful, as the system output for arbitrary input can be obtained either by (1) taking the Fourier transform of the input, multiplying by the gain as a function of frequency to obtain the Fourier transform of the output, and then taking the inverse Fourier transform to obtain the output in the time domain **or** by (2) convolving the input with the impulse response function to directly obtain the output in the time domain. Convolution is a method that expresses the overlap of one function with another, as they are shifted apart in time, somewhat related to correlation functions that you will learn about in the next lab. This advanced topic may or may not be covered in lecture; in this post-lab, we will introduce you to both methods, but will not delve deeply into the math behind the methods, instead, as is the philosophy in ME 425 laboratory exercises, teach you how to **use** the advanced techniques to analyze your data, because that is helpful to you even if you haven't fully been exposed to the background theory.

The approach to system identification for a **Linear, Time-Invariant System (LTI)** is outlined in the schematic diagram on the next page, courtesy of Ashin Modak, .

Linear Time-Invariant System Analysis

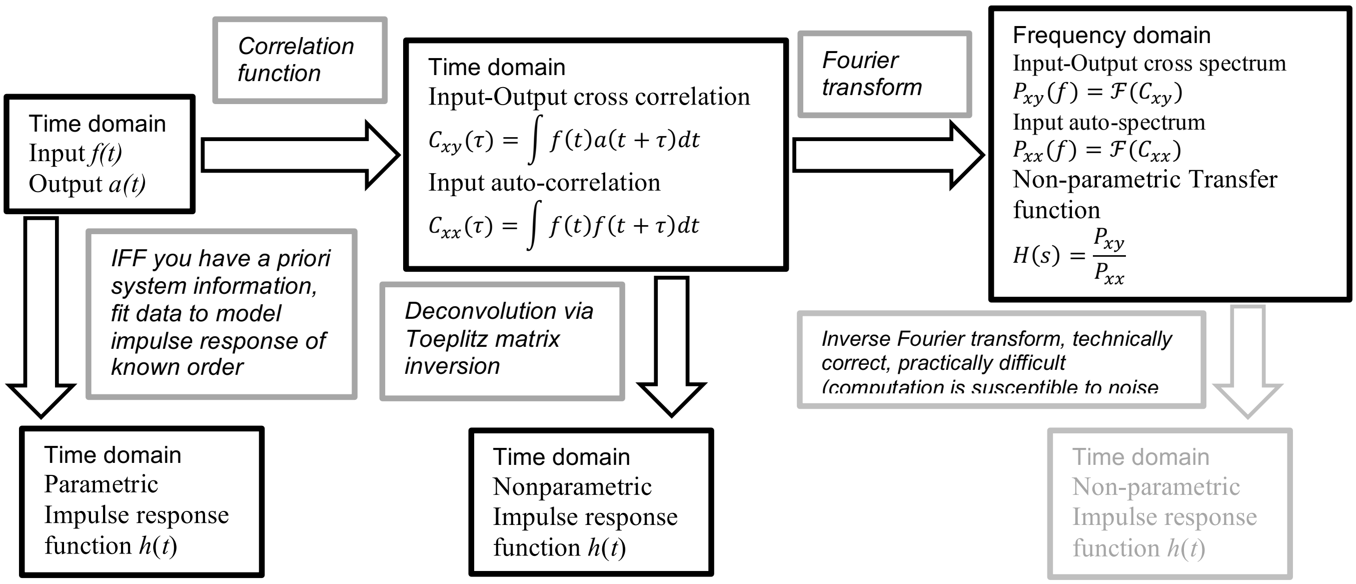
Nomenclature:

$H(s)$ is the transfer function, which for purely imaginary $s=i\omega$ is referred to as the frequency response function and is displayed on a Bode plot

$h(t)$ is the impulse response function. $H(s)$ and $h(t)$ are Fourier transform pairs.

Shown below are the two methods to obtain the **non-parametric** impulse response function from measured system input and output, either directly in the time domain via correlation and convolution, or by using Fourier transform methods to determine the **non-parametric** frequency response function and then inverse Fourier transform to obtain the impulse response function.

As a reminder, **non-parametric** simply means the function is determined directly from the data, with no assumptions made about system order, i.e. the function is specified at every time (or frequency) point, rather than being determined from an analytical expression using derived system parameters such as α, ω_n, and ζ.



Once the non-parametric transfer function $H(s)$ or non-parametric impulse response function $h(t)$ have been determined, if the system appears to have a given order (i.e. 1st or 2nd order, etc.), least square fitting methods can be used to determine the system parameters by fitting the non-parametric function to the functional form for a system of the selected order.

Analysis of Impulse and Stochastic Response Data

Frequency Domain Approach:

Input/Output Signals $\xrightarrow{\text{xcorr}}$ Correlation functions $\xrightarrow{\text{fft}}$ Auto/cross spectra $\xrightarrow{\text{divisio}}$ $H(s)$ $\xrightarrow{\text{tfest}}$ α, ω_n, ζ.

Since you found by the impulse response experiment in lab that the system is well-characterized as a second order system, we will make that assumption when analyzing the data in the time domain, which will proceed as follows (furthest left option in flow chart above):

Time Domain Approach: (done in lab in Logger Pro for impulse response, repeat in MATLAB and also do for Stochastic data)

Input/output Signals: $\xrightarrow{\text{lsqcurvefit}}$ α, ω_n, ζ.

8.2 Comments on the Analysis

A suite of MATLAB scripts has been created to help you analyze your data. They are meant to be generally useful for other experiments, such as some or your Go Forth projects, so please read the comments in the m-files in order to correctly use them to analyze your data.

Please run the code on your data as soon after lab as possible, so that you have time for us to help you if need be before your post-lab is due. Simply running the code will not take much time, and you should be able to quickly tell if you have problems with the analysis. That being said, we would prefer not to just give you MATLAB code, because our experience is that students simply use the code without reading the comments and understanding what the program is actually doing. It is **crucial** that you understand this code, so that if you need to do similar calculations in the future, you can modify the code to apply to your specific problem.

After checking to make the code seems to be working, please then spend some time going through it, referring to the simple flow charts on the previous page, and the ordered list of the program to use below, while reading the code (and comments in the code) to make sure you understand the function of each block of code. MATLAB has excellent help, both in the code and via Google, please take advantage of it as needed!

The code has been tested on a number of sample student files, and in most cases, the frequency domain approach is not very good at fitting a model to the data. The time domain approach was found to be the most successful. In general, the gain is computed more successfully than the phase. We suspect this is a result of the essentially undamped nature of the system (when you fit the impulse response with Logger Pro in lab, you probably found $\zeta < 0.01$). With such low damping, the resonant peak is extremely narrow. As you will learn in Lab 4, the resolution of a Fourier transform is inversely proportional to the sample time, so a 10 s sample time will have a frequency resolution of 0.1 Hz. The width of the resonant peak in the models you will generate is 0.1 Hz or less, which is of order the frequency resolution, so the time domain method will probably do a better job of fitting a model to your data. However, it is still instructive for you to apply both methods, because for some of your Go Forth projects, the frequency domain approach may be preferable.

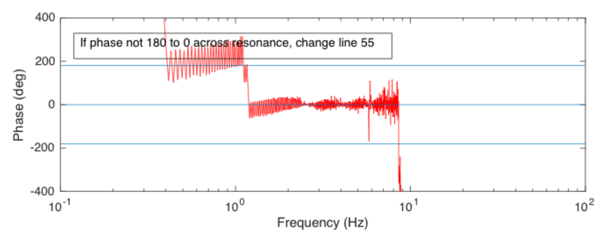
You already used a frequency domain approach in Lab 1 to analyze the swept sine data – this method was essentially the same as described above, but used the MATLAB function *tfestimate*() which splits the data into multiple shorter samples and then averages the system behavior over the multiple samples. This would exacerbate the frequency resolution problem because deceasing the sample time makes the frequency resolution worse (i.e. increases the space between adjacent frequency points). Running the Lab 2 code on your sample data results in a much broader resonance peak, and thus a larger ζ than is actually the case.

As stated above, **please try running the code on your data as soon as possible after you complete the lab experiment!**

5.3 Analyzing the Impulse Response

Start with the Frequency Domain Approach:

1. ***load_CSV_file.m*:** Use this to import your data from the CSV file. *Note it starts with "clc" to clear the command window, "close all" to close all open figure windows, and "clear all" to clear the workspace variables. You may comment out these lines if you prefer.*
2. ***FixTimeShift.m*:** This script assigns the Coil and Accelerometer data columns , converts coil current to force (*you must change the code and put in the calibration constants you found in lab in Line 85*), and then removes the unphysical time delay between input (force) and output (acceleration). The script will generate a plot to show the raw and synchronized data. Make sure that both acceleration and coil start at 0 (i.e. both sensors were properly zeroed), and that force and acceleration have the same sign (positive force, positive acceleration). If the readings do not start at zero before the impulse, subtract that offset value from the entire column of data. ***Use Edit – Copy Figure to copy the graph created by the program into a Word file.***
3. ***BodePlotWithFrequencyDivision.m:*** This uses the path shown by following the top boxes in the diagram on the previous page, by computing the auto- and cross-spectra from the FFT of the auto- and cross-correlation functions, and then dividing the two to obtain $H(s)$. *Phase computations are complicated by the "wrapping" problem, i.e. phase differences of integer multiples of 360° are indistinguishable. The desired final phase decreases from 180° above resonance to 0° below resonance. The script attempts to find a correct multiple.* ***Use Edit – Copy Figure to copy all three graphs created by the program <u>in order</u> into the same Word file.***



4. ***SecondOrderFit_Freq.m*:** Uses the MATLAB function ***tfest*** to fit the non-parametric transfer function found in step 3 to a parametric 2nd order acceleration model. Output of this program is the parameters α, ω_n, and ζ, as well as a plot comparing the non-parametric (measured) transfer function to the parametric (model) one. ***Use Edit – Copy Figure to copy the two graphs created by the program <u>in order</u> into the same Word file.***

Record in the Word file the values of the 2nd order system parameters. Is $\zeta << 1$? What is $\sqrt{1-\zeta^2}$? Are we justified in neglecting it? (You will answer these questions in text paragraphs in your post-lab).

End with the Time Domain Approach:

In the same workspace, after running the 4 programs above, run

1. **SecondOrderFit_Time.m:** This program uses linear least square curve fitting, MATLAB function lsqcurvefit() to fit the input and output time signals to a 2nd order function. The m-file mass_spring_output.m must be in the same directory as it contains

the model. Output of this program is the parameters α, ω_n, and ζ, a plot of the model output vs time compared to the measured output, and a plot of the non-parametric transfer function found in step 3 above compared to the parametric transfer function defined by the model parameters found from lsqcurvefit(). *You need to input initial guesses for the model parameters – use the ones you found in Logger Pro in lab.* ***Use Edit – Copy Figure to copy the two graphs created by the program <u>in order</u> into the same Word file.***

5. 4 Analyzing the Stochastic Response

Perform the same steps listed above using your stochastic data file, rather than your pulse data file. Save the graphs and parameters in the same Word file.

5..5 Post-lab Report

Task 1: You must make two professional quality figures with descriptive figure captions as described below and include in your post-lab submission.

Figure 1: Bode Gain Plot: For the **Impulse Response** data, modify the final Bode Gain Plot from either SecondOrderFit_Freq.m or SecondOrderFit_Time.m to make it "pretty". We suggest you select the one for which the model provides a better fit to the data! We do not want you to show us the Phase plot, because in many physical systems, the phase is less relevant than the gain, and also because you may have noticed that phase computation is challenging. You will have to therefore create a new figure with only the Gain graph. You must make the figure "pretty", either by using the interactive "Edit Plot" GUI (Graphical User Interface), or by learning some of the command line functions in MATLAB to create professionally formatted plots, as discussed in the Lecture on Graphs. Please refer to the Pretty Plots in MATLAB lecture notes, found in the Lecture Notes on the Wiki as well as on the MATLAB page and the Graphic Communication – Graphs – Creating Good Graphs – MATLAB page. You may also find it helpful to Google "MATLAB basic plotting functions" and take the first link on mathworks.com. Remember that a "pretty" graph may look "cartoony" in the MATLAB screen! **If you submit an essentially unmodified graph, you will have points taken off.**

Figure 2: Comparison of Output Signal to Model: For the **Stochastic Response** data, modify the final plot of Output Signal vs Time (data and model) from either SecondOrderFit_Freq.m or SecondOrderFit_Time.m to make it "pretty" as described above. We suggest you select the one for which the model provides a better fit to the data!

Figure Captions: Your figure captions must do the following: **Control** the audience's viewing process; **Guide** the reader toward significance; **Inspire** thinking and discussion; **Educate** the reader in a memorable way (acknowledgment to Jared Berezin).

Your captions must address all the questions described in the Graphs Lecture:

- What am I looking at?
- How should I navigate the image?
- What should I focus on?
- Why does this matter?

Your figure captions should include a brief description of the physical system, the 3 model parameters with units (as found in MATLAB for the method shown in the graph, i.e. the Frequency or the Time method), as well as anything else you think important to include in the caption (for example, a condensed discussion of the comparison we ask you to make below and on the next page). It is not necessary to repeat information in the Fig 2 caption that was stated in Fig 1, but please refer back to the previous caption as necessary.

Task 2:

Write 1 – 3 paragraphs answering the following questions, as well as providing more description of the system and the analysis method than you included in the caption.

Paragraph Text Part 1: *Describe in your own words the frequency domain method and the time method that you used to analyze the data. Discuss the differences between them, and how well the system models created with each method matched the data and/or non-parametric transfer function obtained from the data. Discuss system damping, are we justified in assuming that the resonant (damped) frequency is essentially equal to the natural frequency? You make this assumption when you equate the "B" parameter from the Logger Pro fit with the natural (not resonant) frequency. Make any other descriptive and ideally quantitative comments as desired, i.e. when talking about a discrepancy (different between data and model), use numbers (or % difference).*

Using Measured 2nd Order Parameters to Obtain Physical Information about the System

We will now show you the value of the system transfer function, as represented by the impulse response and/or the frequency response. The general differential equation and transfer function for a 2nd order system with input is given in terms of α, ω_n, and ζ are given as

$$\ddot{x} + 2\varsigma\omega_n\dot{x} + \omega_n^2 x = f(t) \quad (4)$$

$$H(s) = \frac{\alpha \cdot \omega_n^2}{s^2 + 2\zeta\,\omega_n s + \omega_n^2} \quad (5)$$

The corresponding equations for a physical system with mass M, spring constant (or stiffness) K, and damping factor C are as follows:

$$M\ddot{x} + C\dot{x} + Kx = F(t) \quad (6)$$

$$H(s) = \frac{1}{Ms^2 + Cs + K} \quad (7)$$

Comparison of Eqs (5) and (7), which express the same transfer function using different parameters, immediately yields the expected result of

$$\omega_n = \sqrt{\frac{K}{M}} \tag{8}$$

The definitions of the remaining parameters in Eq. (5) in terms of those in Eq. (7) are

$$\alpha = \frac{1}{K}, \text{ and} \tag{9}$$

$$\zeta = \frac{C}{2\sqrt{M \cdot K}}. \tag{10}$$

1. Compute *K, M*, α, and ω_n from first principles as follows: *M* = the system mass that was listed on the apparatus (and should be in your lab notebook). Determine the expected value for *K* from the manufacturer's specification of the spring constant as 0.10 lbf/in (convert to N/m please!). Use these values to determine the expected values of α and ω_n from the equations above
2. Create a table with the column headings shown below (but please add units where needed!) and 6 rows containing values obtained from: (1) First Principles (no value for ζ); (2) Logger Pro fit to Impulse Response done in lab (only values for ω_n and ζ, see below); (3 & 4) Impulse Response with both Frequency and Time Domain Analysis; and (5 & 6) Stochastic Response, also with both Frequency and Time Domain Analysis. **NOTE ABOUT LOGGER PRO:** (1) only list values for ω_n and ζ because the input was not a true impulse so the value obtained by Logger Pro for α has neither the correct units nor magnitude. (2) Logger Pro results must be listed with uncertainty, which means you will need to do propagation of uncertainty with partial derivatives to get from the Logger Pro uncertainty (remember the t-factor!) to the uncertainty of ω_n and ζ.

Method	ω_n	ζ	α	K	M

Task 3:

Table and Paragraph Text Part 2: *Make the table "pretty" so that the information is easily understood. Make sure there is sufficient white space so that the numbers do not look cramped, the font is consistent, and that you use the ± symbol when indicating uncertainty for the Logger Pro parameters, being sure to follow the sig fig rules. Think about the text you put for each row to describe the input signal (impulse or stochastic) and the analysis method (frequency or time domain). You may do some grouping of rows to make the table easier to interpret. Add a* ***Table 1*** *caption* ***above*** *the table (figure captions go below the graphic; table captions go above). Make sure your caption answers all the questions discussed in the Graphics lecture, listed on the previous page. Then write a comparing the measured and model values for the parameters, providing enough information so that the graders know what you are talking about without reading the questions we asked you! You may assume they have already read the paragraph you wrote in the previous section, so you may refer back to it, as well as back to Figure 1 and Figure 2 and their captions. Make sure to point out which, if any, of the five measured sets of parameters agrees well with the First Principles calculations.*

Required Content

1) Your name and lab section on the first page! Also put your time estimate for completing the post-lab on the first page, clearly labeled and with units (i.e. Time to complete post-lab: 3.5 hr.)
2) "Pretty" (professional quality) figure (Figure 1) including Bode Gain graph for the Impulse data and model with a descriptive figure caption. **NOTE:** If you simply submit the graphs as created by the MATLAB script we gave you, you will receive very little credit.
3) Professional quality figure (Figure 2) including output signal vs time comparing the data to the model for the stochastic input with a descriptive figure caption. **NOTE:** If you simply submit the graphs as created by the MATLAB script we gave you, you will receive very little credit.
4) One to three paragraphs as described in "Paragraph Text Part 1"
5) Table and paragraph text as described in "Table and Paragraph Text Part 2"
6) ALL of the graphs output by MATLAB at the end, not necessary to "prettify" or provide Figure Captions. However, please do copy them in correctly to obtain high resolution graphs (Copy Figure works well, but if that doesn't work for you, save as BMP or EPS (but NOT JPG) and then copy into Word). Please make them slightly smaller so that at least 2 fit on a page.

Please submit electronically on Stellar as a PDF file by the due date and time.