

EE 324 LAB 2

Using MATLAB to Plot Transfer Function Bode Plots

In this lab, you will practice finding the frequency response and Bode plots of systems using Laplace analysis. The idea is to build a connection between the Laplace transform, frequency response, and time-domain features of the response.

Prelab:

Consider Figure 1.

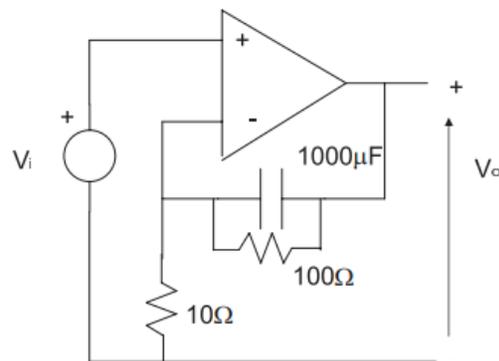
We want to analyze the causal system. We assume that the Op Amp is ideal, i.e., the input impedance is infinite, and that there is no current flowing into the Op Amp and $V^+ = V^-$.

1. Compute the transfer function of the system from $V_i(s)$ to $V_o(s)$.
2. Compute its pole(s), zero(s), and region of convergence.
3. Derive the differential equation which dictates the dynamical evolution of the system.
4. Substitute $s = j\omega$. Compute the magnitude and the phase of the transfer function and describe what the system would do to very low ω frequency and very high ω frequency inputs.
5. Sketch its Bode plot, without computer or calculator.

Laboratory Assignment:

For the circuit:

1. Use the MATLAB function `tf` and `bode` to compute and plot the Bode plot of the system.
2. Derive and build the equivalent Simulink diagrams from the differential equations you have derived in the prelab.
3. Using Simulink, experimentally validate the bode plot. Namely, measure magnitude amplification and phase shift for sinusoidal inputs of various frequencies. Don't forget to wait "long enough" for the transient response to vanish.



Also, note that phase difference can be computed based on time difference t of the zero-crossings of input versus output: $\frac{t}{T} = \frac{\phi}{2\pi}$. Another way can be to LPF the product of the input and the output and then the DC output of the LPF equals $\frac{1}{2} \cos(\phi)$.

$$\cos(\omega t) \cdot \cos(\omega t + \phi) = \frac{1}{2} [\cos(\phi) + \cos(2\omega t + \phi)]$$

4. Collect the frequencies, the associated magnitudes and the associated phase shifts into three vectors called `freq`, `mag`, and `phase`, respectively. Use MATLAB commands `loglog` and `semilogx` to plot the experimental bode magnitude and phase plots. Are there any discrepancies with the one computed in the first part? Explain.

Include all answers to questions, figures and code in your lab report.

EE 324 LAB 2

Using MATLAB to Plot Transfer Function Bode Plots

In this lab, you will practice finding the frequency response and Bode plots of systems using Laplace analysis. The idea is to build a connection between the Laplace transform, frequency response, and time-domain features of the response.

Prelab:

Consider Figure 1.

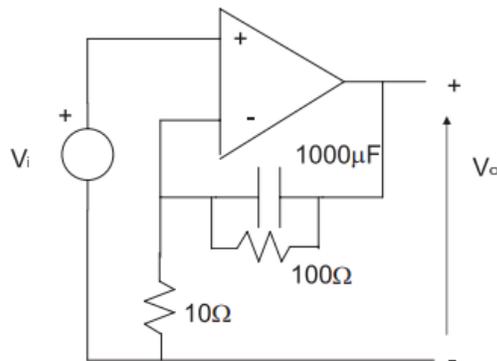
We want to analyze the causal system. We assume that the Op Amp is ideal, i.e., the input impedance is infinite, and that there is no current flowing into the Op Amp and $V^+ = V^-$.

6. Compute the transfer function of the system from $V_i(s)$ to $V_o(s)$.
7. Compute its pole(s), zero(s), and region of convergence.
Causal \Rightarrow zero impulse response for $t < 0 \Rightarrow h(t)$ right-sided $\Rightarrow H(s)$ right-sided RoS
8. Derive the differential equation which dictates the dynamical evolution of the system.
$$\left[\sum_{j=0}^n a_j \frac{d^j y}{dt^j} = \sum_{j=0}^n b_j \frac{d^j x}{dt^j} \right] \Leftrightarrow \left[\sum_{j=0}^n a_j s^j Y(s) = \sum_{j=0}^n b_j s^j X(s) \right] \Leftrightarrow \frac{Y(s)}{X(s)} = \frac{\sum_{j=0}^n b_j s^j}{\sum_{j=0}^n a_j s^j} = H(s)$$
9. Substitute $s = j\omega$. Compute the magnitude and the phase of the transfer function and describe what the system would do to very low ω frequency and very high ω frequency inputs.
10. Sketch its Bode plot, without computer or calculator.

Laboratory Assignment:

For the circuit:

5. Use the MATLAB function `tf` and `bode` to compute and plot the Bode plot of the system.
6. Derive and build the equivalent Simulink diagrams from the differential equations you have derived in the prelab.
7. Using Simulink, experimentally validate the bode plot. Namely, measure magnitude amplification and phase shift for sinusoidal inputs of various frequencies.



Don't forget to wait "long enough" for the transient response to vanish. Also, note that phase difference can be computed based on time difference t of the zero-crossings of input versus output: $\frac{t}{T} = \frac{\phi}{2\pi}$. Another way can be to LPF the product of the input and the output and then the DC output of the LPF equals $\frac{1}{2} \cos(\phi)$. $\cos(\omega t) \cdot \cos(\omega t + \phi) = \frac{1}{2} [\cos(\phi) + \cos(2\omega t + \phi)]$

- Collect the frequencies, the associated magnitudes and the associated phase shifts into three vectors called `freq`, `mag`, and `phase`, respectively. Use MATLAB commands `loglog` and `semilogx` to plot the experimental bode magnitude and phase plots. Are there any discrepancies with the one computed in the first part? Explain.

Include all answers to questions, figures and code in your lab report.

Lab Tips:

- In Simulink, you need to change the solver type (sampling time) to fixed-step. Recommended step size is 0.0001 seconds (which will allow you to simulate with input up to 1000 rad/s). Instructions for this:

Simulink > Modeling (Tab) > Model Settings > Solver > Solver Selection > Type > **Set to Fixed-step**
 > Solver Selection > Solver Details > Fixed-step size (fundamental sample time) > **Set to 0.0001**

- Once you have your block diagram, connect the block "To Workspace" to your input and output signal (one for each). This records time-series data for the input and output and stores them in a variable in your MATLAB workspace. (Q6)
- Your input signal should be a sine wave. (Q6)
- Each time you run the simulation, only change the frequency of the input sine wave (should have no phase shift and amplitude should always be the same). (Q7)
 - Find the time of the first zero-crossing (where the wave crosses the x-axis, i.e. changes from a positive to negative value). Find this time for the input (t_{in}) and output (t_{out}) signal each.
 - The time difference ($t_{in} - t_{out}$) is equal to your time shift which you can use to calculate the phase shift.
 - Find the amplitude of your output. An easy way to do this is by using the `max()` function in the command line. Ex: `max(out.simout)` where `out.simout` is the time-series data vector in your workspace.
 - The ratio between output and input amplitude can be used to calculate the gain in dB.

$$Gain = 20 \log \left(\frac{A_o}{A_i} \right) \text{ dB}$$

$$Phase = (t_i - t_o) \omega \frac{180}{\pi} \text{ degrees}$$

5. For every input frequency (each simulation) you should have a gain and phase (shift). It is recommended to keep a data table (in Excel, Sheets, etc.) like so, and at the minimum, test the frequencies below. (Q8)

Frequency (rad/s)	A_in	A_out	Gain (dB)	t_in (s)	t_out (s)	Time Shift (s)	Phase Shift (degrees)
0.1							
1							
10							
50							
100							
1000							

- a. The more frequencies you test, the smoother your plot will be in part 4.
6. Enter the data into arrays in MATLAB (1 array for frequency, 1 for gain, 1 for phase) and use the specified functions to plot the data. Syntax is the same for both – refer to [this](#). (Q8)

Note that the amplitude of your input (A_in) should always be 1 unless you have changed the parameter of your input sine block.

Essentially, you are manually recreating the bode plot from part 1 by gathering and plotting individual data points. Thus, your plots in part 4 should resemble the one from part 1.

Report Requirements:

1. Bode plot of the system generated using the bode() function
2. Block diagram
3. Simulation data (ex. data table in Excel)
4. Total of 4 plots – Frequency vs. Gain + Frequency vs. Phase (each plotted with both `loglog()` and `semilogx()`)