

University of Utah
Electrical & Computer Engineering Department
ECE 3510 Lab 10
Flexible Beam

M. Bodson, A. Stolp, 4/6/07
rev, 4/27/07

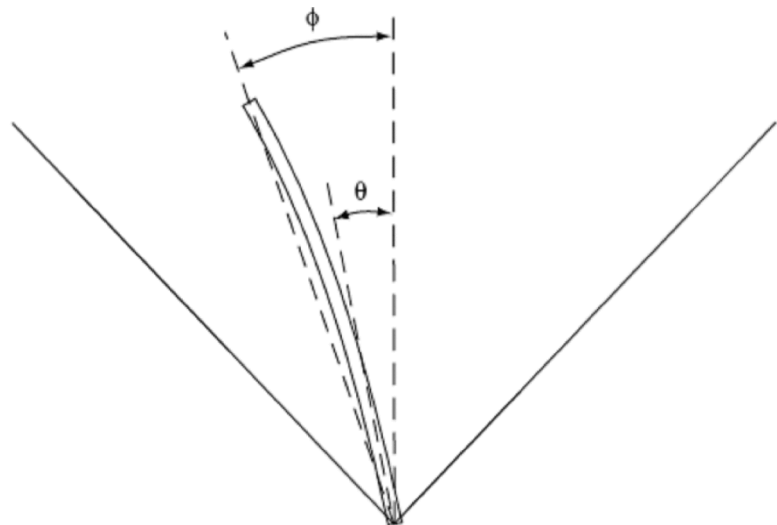
Objectives

The objective of this lab is to learn about the challenges posed by resonances in feedback systems. You'll gain an intuitive understanding of the difficulty through the manual control of a flexible beam resembling a large space robotic arm. The lab will lead you through a control design in the frequency domain using a lead controller. A second design will include a notch filter in the feedback loop to reduce the excitation of resonances.

Introduction

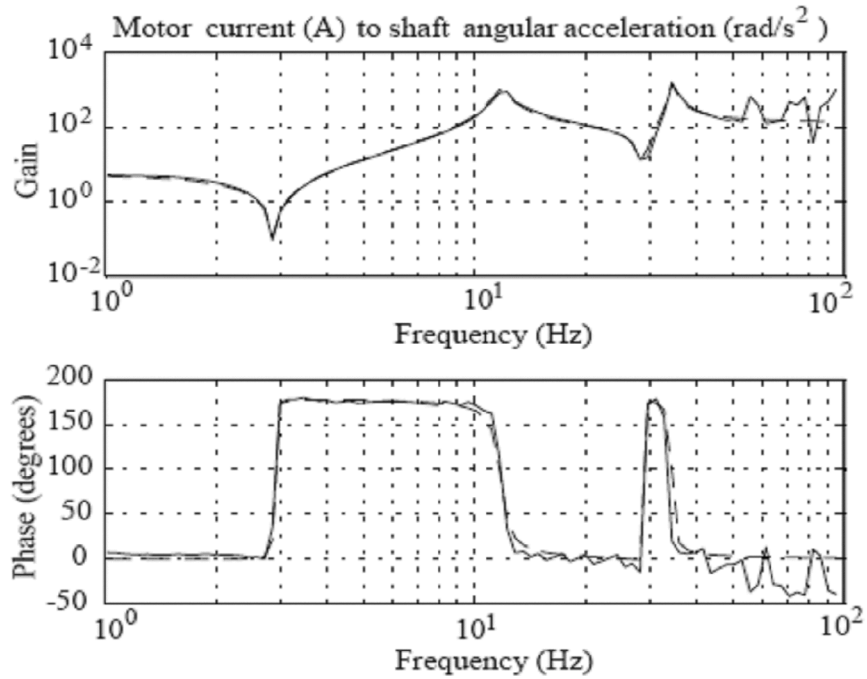
Systems with lightly-damped, complex poles (resonances), are encountered in many applications. An example is a large robotic arm in space, whose transversal dimensions are made small to reduce weight. The arm will bend and oscillate if moved rapidly. In a computer disk drive, a read/write head is attached to the end of a small, rigid structure which is rotated rapidly to access various tracks. When the head is positioned within fractions of microns, even such a rigid structure behaves like a flexible structure.

The diagram of the flexible beam is shown at right. The angle of the beam at the shaft which rotates the beam at the bottom, is denoted θ , while ϕ is the angle at the tip. If there was no flexibility, the two angles would be equal.

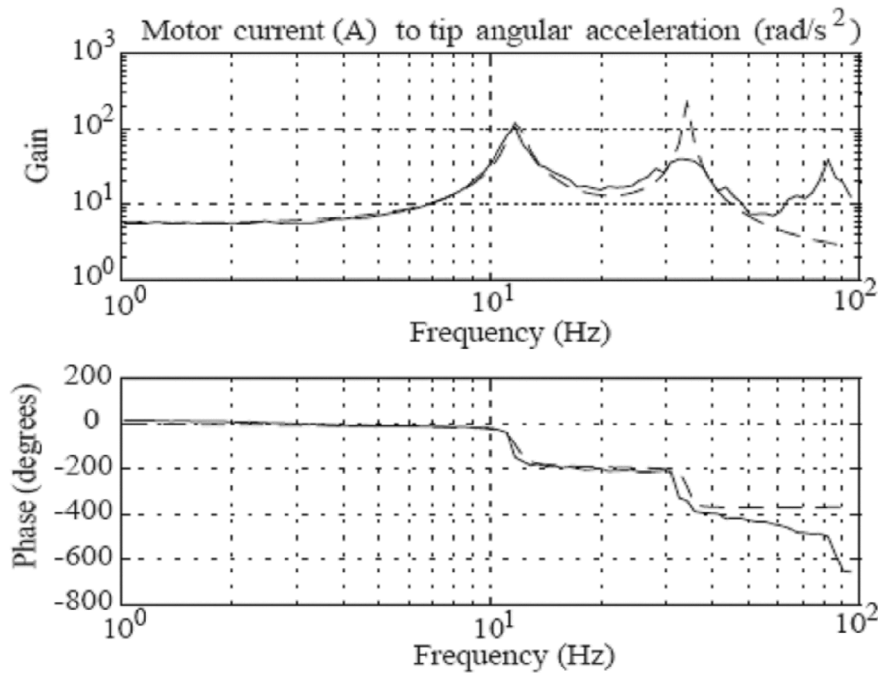


The Bode plot on the next page was drawn from experimental data collected from a 0.4-meter-long flexible beam. An electric motor was hooked to the bottom of the beam and was driven by a variable-frequency current. The

frequency response was measured from the angular acceleration of the motor shaft. The acceleration at the shaft was obtained by measuring the position with an encoder. This position needs to be differentiated twice (multiplied by s^2) to get the acceleration. (The frequency response is multiplied by ω^2 to obtain the acceleration). The next Bode plot shows the response measured from the angular acceleration of the tip. The acceleration at the tip was obtained with an accelerometer. The plots show the experimental data, as well as approximate fits obtained with fourth-order models (as dashed lines). The fourth-order models are shown beneath the respective plots. Note that the poles are very lightly damped.



Dashed line:
$$\frac{s^2 \Theta(s)}{I(s)} = \frac{k_{\theta}((s + 0.07)^2 + 18^2)((s + 0.07)^2 + 180^2)}{((s + 3)^2 + 74^2)((s + 3)^2 + 215^2)}$$



Dashed line:
$$\frac{s^2 \Phi(s)}{I(s)} = \frac{k_{\phi}(s - 100)(s + 120)(s - 200)(s + 300)}{((s + 3)^2 + 74^2)((s + 3)^2 + 215^2)}$$

The previous plots were made for angular acceleration. That makes the plots flatter and it makes the complex peaks easier to see, but we will actually control the beam position. The upshot of this is that the s^2 will move from the numerator on the left to the denominator on the right and the bode plots will actually be sloped downward at an extra 40 dB/decade.

The input variable is the current driving the motor, measured in amps (A), and the angles θ and ϕ are measured in radians. The poles of the transfer functions are equal. So are the DC gains. That makes sense since there is no bending of the beam near zero frequency. The angular acceleration DC gains come out to about 5.5, so for low frequencies, the transfer functions for the angles are approximately given by

$$\frac{\Theta(s)}{I(s)} \approx \frac{\Phi(s)}{I(s)} \approx \frac{k_p}{s^2} = \frac{5.5}{s^2}$$

This approximation of the system is like the double integrator encountered in the ball and beam lab. In this case, however, we also have additional poles and zeros close to the $j\omega$ -axis, and even two zeros in the right half-plane. All this makes the feedback design more difficult than for the ball & beam.

Include some of this introductory material in your lab notebook or report.

Manual control

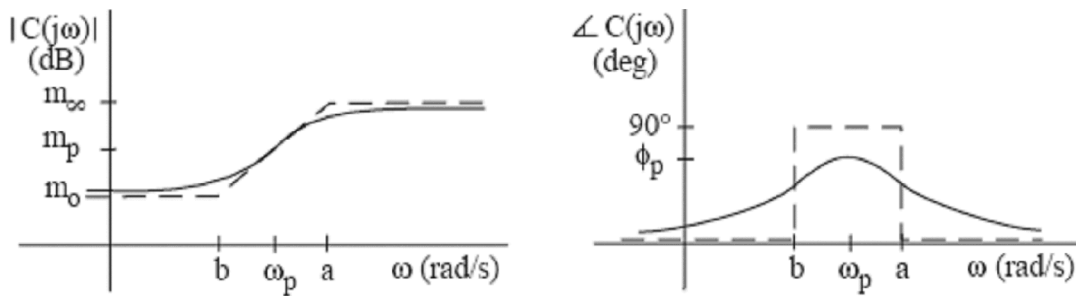
Download a copy of the simulation file called *flex.m*. Run it in the manual mode and try to move the beam to one of the 45° lines and stabilize it there. Try to move the beam back and forth between the lines. You will encounter two difficulties: the $1/s^2$ (lagging) behavior and the flexibility of the beam. You can avoid resonances by moving the beam slowly, but performance will be unimpressive. It can be enlightening (and fun) to “excite” the resonances by applying commands in the same frequency range as the flexible modes. The beam will bend to large angles. Once you understand this mechanism, you may return to the task of rapidly moving the beam from side to side without exciting such resonances. Describe the difficulties of manual control as well as way(s) to deal with them. Is your response time good?

Now before you get too cocky, you should know that the simulation was implemented differently from previous simulations. This continuous time model was discretized with a sampling period of 200Hz. The program runs at a rate of approximately 20Hz, so the visualization actually runs at about 1/10th real-time, slowing the dynamics by about a factor of 10. The actual system dynamics are too fast to be controlled manually. Previous simulations ran at approximately real time.

Lead controller design

The objective is to design a *lead controller*

$$C(s) = \frac{I(s)}{\Phi_{ref}(s) - \Phi(s)} = k_c \frac{(s + b)}{(s + a)}$$



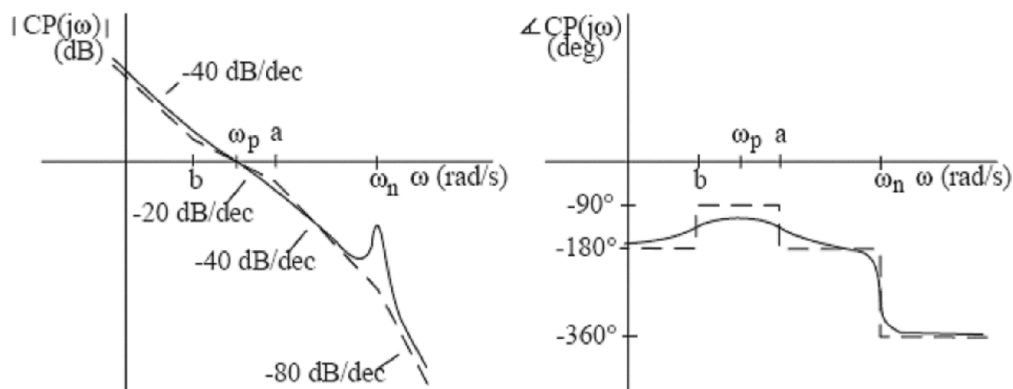
Such a controller is described in the text at the end of chapter 5. Here, as there, the design will be performed in the frequency domain. The motor current i is the control signal, and the tip angle ϕ is the output to be regulated, with a reference value ϕ_{ref} .

The Bode plots of the lead controller are shown above. In the notes for the course, the variables are shown to satisfy the following constraints

$$\omega_p = \sqrt{ab} \quad , \quad m_p = k_c \sqrt{\frac{b}{a}} \quad \text{and} \quad \frac{a}{b} = \frac{1 + \sin(\phi_p)}{1 - \sin(\phi_p)}$$

To determine the controller parameters, we consider an approximate model of the plant with the two poles at the origin and only the first resonant modes. The transfer function from i to ϕ is then

$$P(s) = \frac{\Phi(s)}{I(s)} = \frac{k_p}{s^2} \cdot \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$



These are Bode plots of the loop transfer function for the approximate plant with the lead compensator. The magnitude plot shows as the crossover frequency, a condition that will be enforced by proper choice of the parameters. Assuming that $\omega_n \gg \omega_p$, show that

$$|C(j\omega_p)P(j\omega_p)| \approx \frac{k_p k_c}{\omega_p^2} \sqrt{\frac{b}{a}}$$

and

$$|C(j\omega_n)P(j\omega_n)| \approx \frac{k_p k_c}{2\zeta\omega_n^2}$$

Automatic control

Design #1

For this design, we'd like a phase margin of 60° . Next, we'd like to suppress the gain enough to achieve a gain margin of 10 at that first peak. Finally we'll find the crossover frequency at ω_p and from that, the pole (a) and zero (b) of the lead controller.

$$\phi_p = 60^\circ, \quad |C(j\omega_n)P(j\omega_n)| = 0.1, \text{ and } |C(j\omega_p)P(j\omega_p)| = 1$$

Find the values of the controller parameters a/b, k_c , ω_p , a, and b in that order. This is about the best you can do with this system if you don't add extra filters.

Implement the controller in the simulation with files *flex.m*, *flexc.m* and *flexcinit.m*. Only the last file will have to be modified. A discrete-time equivalent of the control system will be computed, as was done in the previous labs. Discretization will be based on a 200Hz sampling frequency, so that visualization will show the system at a rate slowed down by a factor of 10.

The file *flexcinit.m* will plot the Bode plots with the gain and phase margins (function "margin" in Matlab) and the step response of the closed-loop system (function "step" in Matlab). The feedback system is assembled using a state-space method in Matlab, but could also be done with the functions "series" and "feedback".

Use the stepping reference option, which will cause ϕ_{ref} to switch between 45° and -45° every 2 seconds, and let the record the data for 6 seconds (which will take 60 seconds in the simulation). Plot the responses of i , θ , ϕ , and $\theta - \phi$ to step inputs (run *flexpl.m*). The visualization should show a slow response, with a large overshoot, due to the low frequency zero at $s = -b$. If you reduce the gain margin by increasing $|C(j\omega_n)P(j\omega_n)|$, you will find that the crossover frequency increases and that the response speeds up, but oscillations are observed due to excitation of the beam's flexible modes. Print all the plots.

Try controlling this system with a manual reference. Describe how this works differently than the original manual control and how, with a little practice, you can beat the performance of the totally automatic system. Notice how smoothly you apply the input reference to get this better performance.

Design #2

One way to improve performance is by adding some more filters. A low-pass prefilter, $C_F(s)$, will smooth-out the input steps and a notch filter, $C_{notch}(s)$, will suppress signals which excite the resonance of the plant. The lead controller, $C(s)$, will have the same form as before but different parameters.

$$I(s) = C(s)C_{notch}(s)(C_F(s)\Phi_{ref}(s) - \Phi(s))$$

Let the prefilter be
$$C_F(s) = \frac{1.2b}{s + 1.2b} \quad \text{where } b \text{ is the zero of the compensator.}$$

Let the notch filter be

$$C_{notch}(s) = \frac{s^2 + \omega_n^2}{s^2 + 2\omega_n s + \omega_n^2}$$

where ω_n is the natural frequency of the first resonant mode. The prefilter will eliminate the overshoot, and the notch filter will allow you to increase the crossover frequency and, as a result, the speed of response. Re-design the lead controller with $\omega_p = 12$ rad/s, keeping the condition that $\phi_p = 60^\circ$ and the crossover frequency condition

$$\frac{k_p k_c}{\omega_p^2} \sqrt{\frac{b}{a}} = 1$$

Implement the new design as "Design #2" in *flexcinit.m*. Run the simulation and make all the plots and printouts as before. Compare the responses to those of the previous controller.

Try controlling this system with a manual reference. It is no longer easy to beat the performance of the totally automatic system. Can you tell the difference between manually controlling design#2 and manually controlling design #1?

Checkoff and conclusions

Show your code and demonstrate the real-time operation to the TA.

Be sure to include some description of challenges and strategies for manual control and it changed when the manual became the reference input to the closed-loop system. Describe the improvements achieved by the automatic control systems. Include observations and comments.

