MASSACHVSETTS INSTITVTE OF TECHNOLOGY
Department of Electrical Engineering and Computer Science
6.01—Introduction to EECS I
Fall Semester, 2007

Assignment 6, Issued: Tuesday, Oct. 9

To do this week

No Tuesday software lab

...before the lab Thursday, 10/11

1. Do the on-line Tutor problems for week 6 that are due on Thursday (Part 6.1).
2. Do the writeup for the 10/2 and 10/4 software labs (in a previous handout) providing written answers (including code and test cases) for **every** numbered question in this handout.

...in lab on Thursday 10/11

1. Work on the robot steering lab described below.
2. Do the nanoquiz; it will be based on the material in the lecture notes and the on-line Tutor problems due on Thursday.

...before lecture Tuesday, 10/16

1. Do the writeup for the 10/11 design lab (In this handout).

On Athena or the lab laptops make sure you execute:
`athrun 6.01 update`
so that you can get the `Desktop/6.01/lab4` directory which has the files mentioned in this handout.

- You need the file `polynomial.pyc`.
- You need the file `feedback.py`.
- You need the file `differenceEquationWithInput.py`.
- You will need the z-transform manipulation software you wrote.
- You need the file `soar-graph.py` for the software lab.

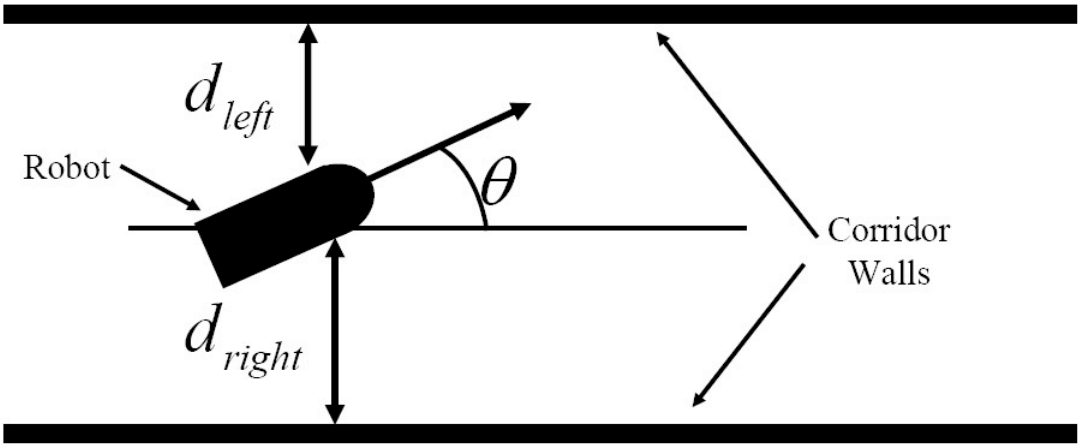

Figure 1: Robot in corridor.

Thursday Design Lab: More sophisticated robot control

Just as in the previous lab, you will be controlling the robot to drive down a narrow corridor as shown in Figure 1. Notice that in the figure, we have denoted the forward speed of the robot, V, the distance to the left wall, d_{left}, the distance to the right wall, d_{right}, and the angle the robot is making with respect to the parallel walls, θ. Unlike two weeks ago, when you tried to steer the robot to drive straight down the center of the hallway, this week you will be trying to steer the robot to stay a desired distance, $d_{desired}$, from the center of the hallway (if $d_{desired} = 0$, the problem is equivalent to last week's problem). Note that the distance from the center is $d = 0.5*(d_{right} - d_{left})$.

Question 1. Briefly explain why the distance to the left of center is $0.5 * (d_{right} - d_{left})$.

In the last lab you used a very simple control strategy to keep the robot in the center of the corridor, $d_{desired} = 0$, by adjusting the rotational speed based on the current value of $d(n)$. You discovered that a model of such a controller predicted robot behavior that would oscillate and eventually hit one of the walls. This time you will design a better controller.

Using Z transforms to analyze the simple controller

Consider trying to steer the robot down a hallway so as to maintain a desired distance, $d_{desired}$, from the center of the hallway while moving forward with a constant speed. Let the error at step n, $e(n)$, be defined as

$$e(n) = d_{desired}(n) - d(n).$$

where $d(n)$ is the measured distance at time n from the center of the hallway. Then, the objective would be to keep the magnitude of $e(n)$ as small as possible.

We can use almost the same difference equation as last lab to model this more general case of an adjustable desired displacement. In particular, the center displacement will still be related to the robot angle by

$$d(n+1) = d(n) + V\delta t\theta(n).$$

where δt is the system's time between samples and V is the robot forward speed. For this lab, use $V = 0.1$ meters per second.

If we use the same control strategy as in the last lab, then the robot angle will satisfy a slightly modified difference equation for this more general input case,

$$\theta(n+1) = \theta(n) + K\delta t e(n)$$

where K is the "gain" of the feedback. You experimented with different values of K in the last lab. BE CAREFUL ABOUT THE SIGN OF K, which depends on the definition of e! Notice that if $d_{desired} = 0$, then $Ke(n) = -Kd(n)$.

Question 2. Determine (by hand) a symbolic expression for the transfer function $\tilde{H}(z)$ in

$$\tilde{D}(z) = \tilde{H}(z)\tilde{D}_{desired}(z),$$

as a function of the gain K. Determine and use the numerical value for δt and use 0.1 for the forward velocity V.

Question 3. Pick a numerical value K, and demonstrate that you can use your transfer function manipulation program to combine the above difference equations. Show that you can both generate an $\tilde{H}(z)$ and a difference equation that relates $d_{desired}(n)$ to $d(n)$.

Question 4. Demonstrate that you can use your program to compute the system's natural frequencies given a numerical value for K, and verify that you program can reproduce your results from lab 4.

Checkpoint: 10:45

- Demonstrate your answers to the above questions to an LA.

Analyzing a more sophisticated controller

In order to design a better controller, one can process the error, $e(n)$, in a more sophisticated way. For example, one could adjust the rotational speed using some combination of the present and previous values of the displacement error. The robot angle would then satisfy the difference equation

$$\theta(n+1) = \theta(n) + \delta t(K_1 e(n) + K_2 e(n-1))$$

Question 5. Determine (by hand) a symbolic expression for the transfer function $\tilde{H}(z)$ of the new controller,

$$\tilde{D}(z) = \tilde{H}(z)\tilde{D}_{desired}(z),$$

as a function of the gains K_1 and K_2. Determine and use the numerical value for δt and use 0.1 for the forward velocity V.

Question 6. Pick numerical values for K_1 and K_2, and for demonstrate that you can use your transfer function manipulation program to combine the difference equations for the more complicated controller. Show that you can both generate an $\tilde{H}(z)$ and a difference equation that relates $d_{desired}(n)$ to $d(n)$.

Question 7. Demonstrate that you can use your program to compute the system's natural frequencies given numerical values for K_1 and K_2.

Designing the controller by placing the natural frequencies.

As you have no doubt discovered, the more sophisticated controller generates a transfer function whose denominator is a cubic polynomial. In addition, two of the denominator polynomial coefficients are functions of K_1 and K_2. Since the roots of the denominator polynomial are the natural frequencies of the feedback system, your design problem is to pick values of K_1 and K_2 so that the natural frequencies are less than one in magnitude.

One approach to determining values of K_1 and K_2 is to use your transfer function program to perform a brute-force search for values of K_1 and K_2. Before trying brute-force, consider a strategy of starting by specifying a set of desired natural frequencies, and then determining the associated feedback gains.

Question 8. Using your expression for the transfer function, can you determine values of K_1 and K_2 such that all the natural frequencies will equal $2/3$ (Hint, try forming the polynomial $(z - \frac{2}{3})^3$ and matching coefficients).

Question 9. Can you determine a set of three natural frequencies that are all less than one in magnitude, but can not be the natural frequencies of your system regardless of the values of K_1 andK_2.

Experiment with your improved dynamic feedback control scheme by writing a brain that on every step, sets the robot speed and updates the robot rotation based on your controller design and a measurement of the distance from the wall. You can start with `feedback.py`, which you may recall using in lab 4, and modify `feedback.py` to use the more sophisticated controller. Try using the gains you determined when all the natural frequencies are $2/3$ on the robot in one of our corridors, with $d_{desired}(n) = 0$ and with the robot initially offset from center.

Question 10. Plot the solution to the difference equation model of the feedback system, using the gains that gave three natural frequencies of $2/3$. Be sure to start with non-zero initial conditions. Does the solution behave as you expected?

Question 11. What happened when you tried to use the gains that gave three natural frequencies of $2/3$ to control the robot? Why?

Checkpoint: 11:45

- Demonstrate your natural-frequency placement controller on the robot.

Searching for a good controller

Optical sensor measurement noise makes it impossible to use very high gains in a feedback controller, the practical limit on any gain (K, K_1 or K_2) is between ten and twenty. Use your transform manipulation program and the system natural frequencies to help you determine effective values for K_1 and K_2 given these practical limits on the gains. You can write a brute-force search method to solve this problem, but you can simplify the search if you think carefully about how to use the constraints on the gain.

Try your new values of K_1 and K_2 on the robot, and compare the performance to using the simple controller from week 4.

Question 12. What happened when you tried your new controller with lower gains.

Question 13. Do you see a pattern in the values of K_1 and K_2 for both your brute-force designed controller and the natural frequencies placement controller? Can you think of an explanation for the pattern you see?

Checkpoint: 12:30

- Demonstrate your new controller on the robot.

Concepts covered in this assignment

Here are the important points covered in this assignment:

- Learn about difference equations and how to solve them.
- Learn about the connection between natural frequencies and the solutions of difference equations.
- Learn to reason about systems using difference equations.
- Begin to learn about the issue of stability in dynamic feedback systems.
- Learn about transfer functions and how to manipulate them
- Learn about how to analyze feedback systems
- Gain a more complete understanding of feedback stability