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

Work for Week 5

Issued: Tuesday, March 6

This handout contains

- Software Lab for March 6th
- Prelab exercises due Thursday March 8th before lab
- Post-lab exercises due Tuesday March 13th in Lecture.

Dynamic Feedback Control

In many of the labs so far, you developed progressively more sophisticated approaches for measuring the robot's environment, as well as progressively more intelligent approaches to controlling the robot's actions based on those measurements. In those labs, our primary goal was to have you learn a little about the robot as well as have you investigate some ideas in software engineering. In this lab, we will also be using measurements to control the robot, also known as feedback control, but we will be investigating a situation in which detailed analysis is needed to achieve a desired performance. The problem we are asking you to investigate is how to control the robot so that it drives down the center of a narrow corridor at a constant forward velocity. Such a task is similar to what an automobile driver does when keeping a car on the road. We will suggest a simple feedback scheme to keep the robot in the center of corridor, and ask you to develop and analyze a mathematical model based on difference equations that explains the behavior of that simple feedback system. In next week's lab, you will use more sophisticated mathematical tools to develop improvements to the simple feedback scheme suggested in this lab.

The summary of tasks for this week are:

- Post-lecture software lab on solving second order difference equations
- Pre-lab tutor exercises to practice solving difference equations
- Robot lab on implementing and analyzing a simple feedback scheme
- Post-lab writeup and exercises due Tuesday in lecture.

Tuesday's Software Lab: Solving second-order difference equations

For this lab, you will be writing a python program that solves arbitrary second-order homogeneous difference equations analytically, by computing natural frequencies. Your program should take as inputs the initial values $x[0]$ and $x[1]$ as well as coefficients a_1 and a_2 for the difference equation in the form

$$y[n] = a_1 * y[n-1] + a_2 * y[n-2].$$

Your program should print out the solution to the difference equation and also return a procedure that, when called with n, returns $y[n]$. For example, if the difference equation is

$$y[n] = 2 * y[n-1] - 2 * y[n-2],$$

and the initial conditions are $y[0] = 0$, $y[1] = 1$, your program should print something equivalent to

```
y[n] = (2.77555756156e-017+0.5j)*(1-1j)**n  +  (-2.77555756156e-017-0.5j)*(1+1j)**n
```

and your program should ALSO return a function which can be used to evaluate $y[n]$ for any n.

Getting Started

To save time, you should use our implementation of the polynomial manipulation program from the recent post-lab exercises in your second-order difference equation solver. Please download *importPolynomial.py* and *polynomial.pyc*. Executing the module *importPolynomial.py* in IDLE will import our version of the polynomial manipulation routines implemented in the class Polynomial. You can see the class interface by typing

```
help(Polynomial)
```

at the IDLE command line. Note that the Polynomial class has functions to add, multiply, print and find the roots of polynomials.

Demonstrate to your LA that you understand the Polynomial class. Then, before you write any code, describe your approach to the difference equation solver to your LA.

A note on complex numbers

For this problem, your procedure will sometimes need to compute z^n where z is a complex number. Python understands complex numbers to some extent, but you have to write them in the form $a + bj$, where a is the real part and b is the imaginary part. Note that Python uses the convention that $j = \sqrt{-1}$. This should not surprise you, as entering EECS students, you already appreciate that the variable i must be reserved for electrical current.

So, for example, the Python command

(-4)**0.5

will produce an error, but the Python command

(-4+0j)**0.5

produces (1.2246063538223773e-016+2j). (Why the very small real part?)

Note: use `y**0.5` to compute the square root of a number. The python *sqrt()* function does not understand complex numbers.

Demonstrate that your program works

Use your program to solve the difference equation

$$y[n] = 2 * y[n-1] - 2 * y[n-2],$$

using the initial conditions $y[0] = 0$, $y[1] = 1$. Demonstrate to your LA that the program prints the correct information and also returns the correct procedure.

Use your program to find difference equations with given properties

Use your program to find a difference equation whose solution oscillates rapidly, but the amplitude of the oscillation decays slowly. Use your program to find a difference equation whose solution oscillates rapidly, but the amplitude of the oscillation remains constant.

What to turn in

For this software lab, hand in a brief (few page) lab report along with your post-lab report on March 13th. Also, hand in a copy of your carefully commented python code with an explanation, *written in actual English sentences*, of how and why it works.

Homework in preparation for Thursday's lab

This section includes problems to complete with the online tutor. These are due before lab on Thursday. The examples are to help you solidify your understanding of difference equations.

Read the entire document!

Please read the entire section on the lab before coming to lab.

Exercises with the online tutor

Use the online tutor to complete the problems due Thursday, March 8th.

Thursday's Lab

As we are sure you have come to expect, Thursday's lab will end with a nano-quiz that is based both on tutorial problems due on thursday, and on material covered this week. It should be no great surprise that you will be asked a question about the solution to a difference equation. Please keep in mind that the point of the quizzes is as a diagnostic for us (did we succeed in teaching you?) and to encourage you to do the tutorial problems before coming to lab and participate in lab.