Finding Roots of Functions

Project #7

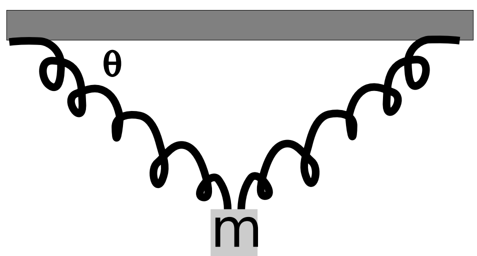
Computational Physics Lab

[Due Friday, February 22]

Part 1

Mass on Two Springs

As a simple example of the usefulness of the root finding techniques we will attempt to solve a deceptively simple problem that does not have an analytic solution, namely that of a mass suspended between two springs as shown below:



The basic problem is to find the value of θ at which this system will be in equilibrium. For our present purposes we will use the following values:

m	5 kg	mass of the object
L_o	0.3 m	half the distance between the two supports
k	1000 N/kg	spring constant
g	9.8 m/s2	acceleration due to gravity

Given the information above, it is straightforward to find an equation that can be solved to find the angle θ:

$$\tan(\theta) - \sin(\theta) = (mg)/(2k\, L_o) \qquad \text{(1.a)} \quad \text{or}$$

$$\tan(\theta) - \sin(\theta) - (mg)/(2k\, L_o) = 0 \qquad \text{(1.b)}$$

The physical problem is to find the angle of the spring when the above system is in equilibrium. If we assume that we can determine the angle to an accuracy of 1/1000 of a degree, then we would like a numerical technique that provides us with at least that much precision.

-Over-

One way to accomplish this is to search for the zero of the function in equation 1.b. We can accomplish this using any one of the methods described in class. Furthermore if we examine the progression of the estimates obtained from our root finding methods and continue to iterate until the change in consecutive estimates is below the desired precision, we can get a result that has a known accuracy.

This problem is a relatively simple one to solve, even with a calculator. Get the source code from cvs for a program which utilizes the Bisection Method to solve this problem. Compile and run this program. Recall the cvs command "cvs checkout rootfinder" to checkout the source code, but first remember to set the environmental variable $CVSROOT ("setenv CVSROOT /export/home/crede/comphy/cvs").

Additional tasks

1. Given the information provided above, create a second program and solve the two spring problem using the *False Position* method.

2. Given the information provide above, create a third program and solve the two spring problem using the *Newton-Raphson* method.

3. Compare the results given by the three methods. Do they obtain the same roots? Do they obtain the same precision? Do they have the same rate of convergence to the answer?

4. Using gnuplot, graph the successive approximations for the three methods on a single graph.

Bonus Assignment:

Use the Newton-Raphson method in order to solve for the roots of the complex function $f(z) \equiv z^3 - 1 = 0$. Plot only the starting complex points which converge using this method. Limit your range in the complex plane to { ± 2, $\pm 2i$ }.

Part 2

Post exercise to your computational physics website. Create a html page for Exercise 5. Create a link from your main project web page to this html page. This html page should include the following heading information: <u>exercise title, exercise number, your name, & today's date.</u> The main content of this page should include the following:

- a short <u>description</u> of the exercise
- a link to the source codes
- separate text regions which contains the actual source code text
- separate text regions which contains the program output (not need for Bonus)
- an image of the generated plot(s)

** For text regions use the html object tag; example:
<object width="600" height="400" type="text/plain" data="program.cc"border="0"></object>