Homework Set #4, Math 475A

Part I Matlab Part. So far you have learned how to get on-line help for MATLAB commands, and how to write the two types of M-files: script files and function files. You also have had some knowledge how to draw a graph. Here you’ll learn more about graphics and how to use them to best present your numerical results. This part is divided into two sections: the practice section and exercise section.

You don’t need to hand in anything for this section. But you’ll be taught some useful MATLAB commands that should be useful for solving the problems in the exercise section.

1. When you try to find the root of a function, you would love to see the graph of that function because the graph can give you some rough idea which method to use and how to pick initial constants, etc. Of course, it’s not possible in general, but it’s very easy to do in MATLAB for the functions you will see in this homework. The MATLAB can draw the graph of a function if you specify the function and the interval. For example,

\[
\text{fplot('sin(x)',[-pi,pi])}
\]

You’ll see in the graphics window the graph of \(\sin x\) in the interval \([-\pi, \pi]\). Use \text{help fplot} to find more examples. You can also add a title to the graph by

\[
\text{title(' \sin(x) ')}
\]

You can also control the ticks for the x-axis and y-axis. For example, to change the ticks on the y-axis, use

\[
\text{set(gca,'YTICK',[-1,-0.5,0.0,0.5,1.0])}
\]

Notice that the ticks on y-axis now only appear at the values specified in the brackets, i.e. \([-1, -0.5, 0.0, 0.5, 1.0]\). \text{gca} here means get current axis. To change the ticks on x-axis, simply change ’YTICK’ to ’XTICK’ and change the values in the brackets properly. The function that MATLAB can draw for you can also be specified by a function file. Recall that we have used function file called \text{fun1.m} which defines the function \(e^x - \sin x\). Try the following script:
fplot('fun1(x)',[-10,1]);
title(' The function defined by fun1.m ')
set(gca,'XTICK',[-10,-8,-6,-4,-2,0,1])
set(gca,'YTICK',[-1,0,1,2])

Actually, you don’t know where to put the ticks the first time you plot the function. They are usually added later to make the graph cleaner and nicer.

You can draw several graphs in one window. This can be done in two ways. One way is to draw the graphs on top of others. This is done by a control command which tells MATLAB whether to erase the existing figure before draw the new graph. The control command is called hold. If you tell MATLAB hold on, MATLAB will not erase the existing graph when drawing the new one. Otherwise, hold off will do the opposite. The default (i.e. if you tell nothing to MATLAB after start it up) is hold off. Try now:

    hold on;
    fplot('fun1(x)',[-10,1]);

Another way is to use the command subplot to divide the window to several sub-windows. In each sub-window, you can do anything except sub-divide it again. Try

    subplot(221), fplot('sin(x)',[-pi,pi]);
    subplot(222), fplot('sin(x)',[0,2*pi]);
    subplot(223), fplot('fun1(x)',[-10,1]);
    subplot(224), fplot('fun1(x)',[-4,-3]);

Identify the correspondence of the window and its identifier number, e.g. 223 means the left-lower corner one. This can be used, for example, to put graphs side-by-side for comparison or simply save some papers. Use help more to find more details.

Part —I Analytical Part.

1. Prove that \( f(x) = 2 + x - \tan^{-1}(x) \) has the property \(|f'(x)| < 1\). Prove that \( f(x) \) does not have a fixed point. Show whether \( f(x) \) is contractive.
2. Write two different equivalent fixed point problems for \( f(x) = 2x^2 + 6e^{-x} - 4 = 0 \).

3. Let \( z = x + iy \), a complex variable.
   - Write down the 2-dimensional root-finding problem for \( f(z) = -1 + z^2 + e^z = 0 \), by separating the real and imaginary parts of the equation.
   - Plot \( f(z) \) in the complex plane.
   - Confirm that \( z = 0 \) is a root for \( f(z) \).

Part III Experimental Part:

1. When we judge a numerical method, we usually talk about its robustness (i.e. whether it can work for a large set of problems) and its efficiency (i.e. how fast it gives you the answer). The efficiency of a method depends part on the order of convergence of the method. For the methods of finding a root of a function or a system of functions, the order of convergence is the largest real number \( q > 0 \) such that the following error estimate is true:

   \[
   |e_{n+1}| \leq C|e_n|^q
   \]

   where \( C \) is some positive constant and \( e_n = x_n - r \) with \( x_n \) be the value after \( n \) iterations. Take the logarithm of the above error inequality, we have

   \[
   \log |e_{n+1}| \leq \log C + q \log |e_n|
   \]

   If this had been an equal sign, we would have been able to say that \( \log |e_{n+1}| \) has a linear relation with \( \log |e_n| \) by a factor of \( q \). In fact, this is almost true for large \( n \) (e.g. Problem 3 on Page 96 which you are asked to do in Part I of this assignment). Therefore, when we test an algorithm for its order of convergence, we can just simply find this relative factor. Graphically, this is just the slope when plotting \( \log |e_{n+1}| \) against \( \log |e_n| \) for large enough \( n \). For example, try

   \[
   \begin{align*}
   e(1) &= 1.71e-01; \\
   e(2) &= 2.11e-02; \\
   e(3) &= 3.68e-04; \\
   e(4) &= 1.13e-07; \\
   e(5) &= 1.07e-14; \\
   e(6) &= 2.22e-16;
   \end{align*}
   \]

   \( \loglog(e(1:5), e(2:6), '\-', e(1:5), e(2:6), 'o') \)
What is the value of $q$? Can you trust $e(6)$? Why? Note that the above loglog plot draws each data point with the circle symbol ‘o’ and also connects the points by solid lines. You should produce similar graphs for the problems in the next section.

2. Use Newton Raphson to obtain an approximation to a root for $f(z) = -1 + z^2 + e^z = 0$, where $z = x + iy$.

   (a) First of all, write down the system of equations for the real and imaginary parts of this problem.

   (b) Compute the Jacobian.

   (c) Create a code to obtain the root.

   (d) Hand in a plot of the log of $|z_{n+1} - z_n|$ versus $n$. Show that you are obtaining quadratic convergence. Also plot $f(z_n)$ as a function of $n$, demonstrating that the iterates are approaching the root.