ACM 11: Homework 4

Assigned Wednesday October 22, 2008. Due Wednesday October 29 at noon. 50 points. Instructions are identical to Homework 3. Please display your email address so we can email grades. Specifically, here are the instructions:

Create a directory entitled `Firstname_Lastname_4`, where you replace `Firstname` and `Lastname` with your first and last names, respectively. Save your solution for this problem set in this directory; when finished, follow the directions on the course site to submit the directory by FTP. The main program – the one that the TAs will execute – should be named `main.m`; give the auxiliary functions referred to in the assignment reasonable names. If you need to submit revisions of your solution, follow the instructions provided on the course site.

Begin `main.m` with a title comment containing your name and email address. This information should also be displayed in the command window when your script runs. Output the answers to all questions to the command window, properly labeled, including the problem number; if you output a number, indicate what has been measured, and if you provide an explanation, indicate what phenomenon you are explaining. All values in the command windows should be the result of deliberate display statements (e.g. `disp` or `fprintf`) and not MATLAB default evaluations! This documentation is required! Put simply, your output should be easy to read and understandable without reference to your code. Your grade will be almost entirely determined by the **output** of `main.m`, not the code itself.

Each graph that is requested, or which you found useful while solving a problem (e.g. one which supports an explanation), should be created in a new figure (using the `figure` command), properly labeled, and referred to in your output by figure number. Be sure to put `clear`, `clc`, and `close all` as the first executable instructions in `main.m`.

1. Touch-tone phones.

   Please read the first section in chapter 8 of Cleve Moler's online MATLAB textbook, available at `http://www.mathworks.com/moler/fourier.pdf`, about touch-tone dialing. Then download his touchtone data file, availabe at `http://www.mathworks.com/moler/ncm/touchtone.mat`.

   (a) Write a function that takes the row k and column j of a number dialed on a keypad, and returns the number (or symbol, e.g. #) dialed, assuming a standard phone keypad as in figure 8.1 in Moler. Return the number as a string. 5 points.

   (b) Extract the data from `touchtone.mat`. The file has some data we are not interested in; to get the interesting data, extract the time signal from the `sig` field in `y`, and the sampling rate from the `fs` field in `y`. As is evident from plotting the signal, there were 11 numbers dialed, with pauses in between; you can also hear the signal using MATLAB's `sound` command (give this command both the signal and the sampling rate). Plot the signal, and using the "data cursor" button in the figure window, find 11 windows such that only one touch-tone number is in each window. Window the data into 11 chunks, and store each chunk as the element of a cell array. Figure 1 shows an example of 11 chunks of data, each plotted in a different color. 10 points.

   (c) For each chunk of data (corresponding to one touch-tone number), analyze its spectral properties with either `fft`, or `pwelch` (requires signal processing toolbox). To use `pwelch`, the command is

   ```
   [Power,Frequency] = pwelch( yourSignal ,[],[],[],Fs);
   ```

   where `Fs` is the sampling rate. If you use `fft`, you will need to convert the frequency to the appropraite units yourself. Determine which two frequencies dominate, and round them to the nearest touch-tone frequency, using the `fr` and `fc` frequencies listed in Moler's book. Use the function from part (a) to determine what phone number was dialed.

Other than the chunking, everything must be automated. In addition to printing out the correct phone number, also write a function that can take in a new signal (we will generate a new test signal with similar properties to the one in `touchtone.mat`). Your function must accept two inputs: the signal (as either a row vector or a column vector), and the sampling rate. If no sampling rate is provided, the default should be the sampling rate from `touchtone.mat`. The output should be a string containing the number dialed. 35 points.

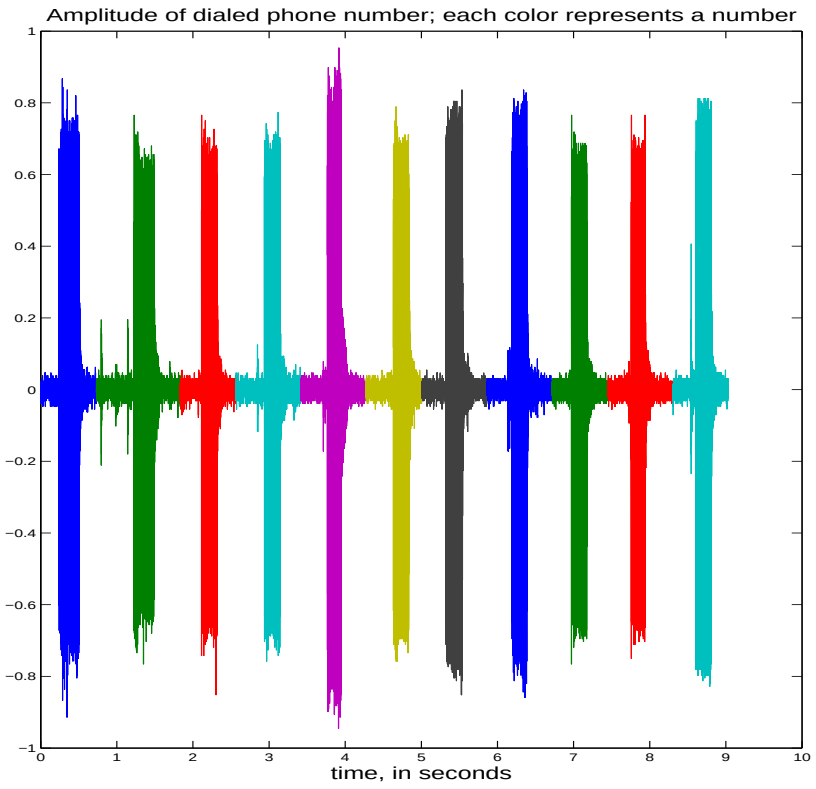

Figure 1: After manually breaking the data into 11 chunks (shown in different colors). A plot is not required for the homework.