New Mexico State University
Klipsch School of Electrical Engineering

EE589 - Digital Speech Processing
Fall 2005 - Exam #2

Name: ______________________

"The attached solution is due entirely to my own, individual efforts. I have not discussed this exam with any other student nor have I consulted with anyone other than (possibly) the instructor of this course in creating these solutions."

Signature: ______________________ Date: ______________________

Prob. 1	/ 20 points
Prob. 2	/ 25 points
Prob. 3	/ 25 points
Prob. 4	/ 30 points
Total	/ 100 points

Overview

The goal of this exam/project is to explore several aspects of speech signal enhancement and modification. You are free to use any and all resources available to you to prepare solutions to this exam. These resources include textbooks, class notes, computers, MATLAB, and the instructor of this course. These resources do *not* include the Internet and other students.

Required Files

All required files for this exam can be found under the Related Links page of the EE589 web site.

Important Dates and Items to be Submitted

Your printed solution (answers, plots, and codes) to this exam is due on or before **5:00pm Friday, Dec. 9, 2005**. In addition, all codes necessary to generate your solution should be submitted as an email to `pdeleon@nmsu.edu` with an exam2.zip file attached.

Lab Hours, Office Hours, and Appointments

During the period of Dec. 5 – 9, Prof. De Leon will be available on and off from 8:00am – 5:00pm and students are encouraged to come by and discuss the problems. Students are also free to call or email.

Prob. 1

The following problems can be found on p. 138 – 139 of Chapter 4 and p. 172 of Chapter 5 in *Speech Communications* by Douglas O'Shaughnessy. Please reference the specific page numbers from this text which are used in your solution.

(a) Chapter 4, Problem 3

(b) Chapter 4, Problem 5

(c) Chapter 4, Problem 7

(d) Chapter 5, Problem 5

Prob. 2

In this problem we will develop a short code to perform eigenfiltering of a noisy speech signal. Let $x[n]$ be the signal from your `Name.wav` and let $b[n]$ be an additive, white Gaussian noise (AWGN). Using the `add_noise.m` tool, create a "noisy" speech signal, $y[n]$ with a 0 dB SNR.

(a) Code the `welch2.m` tool (see attached) for computing average periodograms. Plot the average periodograms of both the clean speech, $S_x(\omega)$ and noisy speech $S_y(\omega)$.

(b) Plot the first 15 lags of the unbiased correlation sequence, $\mathbf{r}$ of $x[n]$.

(c) Using $\mathbf{r}$ to build the Toeplitz matrix, $\mathbf{R}$ (16×16), plot the elements of the eigenvector, $\mathbf{w}$ corresponding to the maximum eigenvalue. Scale the coefficients (if necessary) so that the filter's noise gain is 1. The filter $\mathbf{w}$ is the *eigenfilter* for $x[n]$.

(d) Plot the magnitude response (in dB) of the eigenfilter as a function of f in Hertz. Based on the plots in (a) and (d), why does this filter make sense?

(e) Apply the eigenfilter to $y[n]$ and plot the periodogram of the enhanced speech, $x_e[n]$. Listen to the noisy and enhanced speech signals and comment.

Prob. 3

In this problem we will develop a short code to perform a time-scale modification of a speech signal. To begin, code the `olaistft.m` tool (see attached).

(a) Using your `Digits.wav`, plot the signal (use `plotcsig2`) and the spectrogram (STFT).

(b) Time-scale the signal by a factor of 2 by repeating each frame of the STFT from (a). Synthesize the time-scale expanded signal from the modified STFT using the overlap-add method. Plot the synthesized signal and its spectrogram.

(c) Listen to the modified signal. The articulation rate shold be S-L-O-W (with little distortion) while maintaining the pitch. Comment on the plots in (a) and (b).

Prob. 4

In this problem we will develop a short code to perform noise reduction of a speech signal using the Wiener filter; see p. 674, Figure 13.2 for the block diagram. Please use the file `BPFnoisyspeech.wav` (available on the class web page) and note that the first two seconds of the signal is pure noise. To begin, code the `lseistft.m` tool (see attached).

(a) Plot the spectrogram and averaged periodogram of the noise portion of the signal.

(b) Plot the spectrogram and averaged periodogram of the noisy speech portion of the signal.

(c) Write a code that implements the block diagram and use the LSE-based method for synthesizing the enhanced signal. You may find it easier to modify (filter) all frames first and then synthesize. Note that if you use `welch2.m` to estimate the power spectrum of the noise, you will need to divide the estimate by NU where N is the window length and U is the window gain since this term is omitted from the Wiener filter formula. Plot the spectrogram and averaged periodogram of the enhanced signal.

(d) Plot the magnitude response of the Wiener filter, $H_s(pL, \omega)$.

(e) Comment on your results from (a) – (d).