**Homework #1 Software Tools for Random Signal Analysis**
**(due Sep. 4, 2002)**

1) For all software experiments, be sure to think hard about what the data and plots are telling you and how it relates to the developed theory. Always strive to increase your intuition!

2) All adaptive signal processing tools in the workbook are to be written as functions. Individual "main" codes which implement each homework problem should call these functions. For more information on main codes and function codes, please see Chapters 2 and 3 of DSP Software Toolkit available at

`http://www.ece.nmsu.edu/~pdeleon/Teaching/EE594/DSP_Software_Toolkit.pdf`

When a tool is implemented for the first time, the code should be attached to the end of the homework assignment. All main codes should always be attached to the **end** of the homework assignment. Answers to homework assignments should not be embedded in the code. If you are a novice MATLAB user, you may wish to examine the codes in the posted solutions.

3) Unless otherwise indicated, frequency-domain plots (power spectral densities, periodograms, etc…) should be in units of dB unless otherwise specified and with a frequency variable $f$ in Hertz if a sampling rate $f_s$ is given or $\omega$ in radians/sample if no sample rate is given. Clearly label all plots with variable names and units and title the plot with problem number and other additional information. Plots should have a grid and be zoomed in or out so that relevant features are at maximum resolution. If in doubt, implement the FR_PLOT.M tool from the DSP Software Toolkit.

4) Should you require access to audio recording equipment, the DSP lab (Goddard Annex 132) will be open during office hours.

5) Code the following tools in your DSP toolkit:

CORRELATION.M, PERIODOGRAM.M, PERIODOGRAM_PLOT.M, AR_SYNTHESIZER.M, AR_COEFFICIENT_ESTIMATOR.M

---

**Software Problems**

1. Record your speech signal using the utterance "0-1-2-3-4-5-6-7-8-9" at a rate of $f_s$ = 8kHz with 16 bit resolution. Call the file "speech.wav." Remove any DC offset so the signal has a zero mean. A simple MATLAB code segment for this would be

```
u = u – mean(u);
```

Normalize the signal to unit variance. Note that this normalization may cause large sample amplitudes which will distort the speech signal on playback since the D/A accepts sample values in the range –1 to +1. A simple MATLAB code segment for power normalization would be

```
u = u ./ sqrt(cov(u));
```

a) Measure the mean (should be near zero) and variance (should be near unity) of your speech signal.
b) Plot the first 81 (10ms) correlations [$r(0)$, …, $r(80)$] of your speech signal.
c) Plot the first 801 (100ms) correlations [$r(0)$, …, $r(800)$] of your speech signal.
d) Compare your correlations with other students'. Comment.
e) Plot the periodogram (512 points) of the speech signal. Describe what you see.

2. Synthesize a length 2048 AR process with the coefficients $\mathbf{a} = \begin{bmatrix} 1 & -1 & \frac{1}{4} \end{bmatrix}^T$ and unit-variance input noise.

a) Plot the first 32 correlations [$r(0), ..., r(31)$] of the AR process. You should ensemble average 25 correlation vectors to arrive at a "good" correlation vector.
b) Compute the eigenvalues (use MATLAB's eig function) of the 256 x 256 correlation matrix corresponding to $r(0), ..., r(255)$ (use MATLAB's toeplitz function to build the matrix from the correlation vector). Sort the eigenvalues from highest to lowest and plot.

3. Using the AR model from Problem 2 do the following:
a) Plot the periodogram (512 points) in normal units. You should ensemble average 25 periodograms to arrive at a "good" periodogram.
b) Plot the magnitude-squared response (in normal units) of the all-pole filter $\mathbf{a} = \begin{bmatrix} 1 & -1 & \frac{1}{4} \end{bmatrix}^T$. Compare the magnitude-squared response [which according to (3.21) should be similar to the periodogram given the white noise input] with the periodogram.
c) Compare the periodogram to the eigenvalue distribution in Problem 2b. Your should observe that the plots are similar. Therefore note that for relatively large correlation matrices, the eigenvalues of the correlation matrix can be approximated as equally spaced samples of the periodogram (PSD estimate). Note that we did the eigenvalue sorting to help you visualize the result better but note that there is no "order" to eigenvalues. This fundamental result from Szegö is worth remembering and one you'll often see used in signal processing research. See R. M. Gray, "On the asymptotic eigenvalue distribution of toeplitz matrices," *IEEE Transactions on Information* Theory, vol. IT-18, no. 6, Nov. 1972 for a good tutorial on this result.

4. Linear Predictive Coding is a method for speech which attempts to model the speech signal with an AR model. The model can then be used to compress the speech signal or synthesis a speech signal. In the compression application, the compressed file consists of the AR parameters and some additional "side" information. Typical data rates for compressed speech are on the order of 8kbps (as opposed to 128kbps for uncompressed speech signal). Since speech is not stationary, the AR parameters and side information are computed for short segments of speech and updated regularly.

(a) Estimate the parameters of a 14th-order AR model of your speech signal from Problem 1.
(b) Plot the magnitude response of the all-pole filter (AR model) and compare to the periodogram in Problem 1d.

---

**Text Problems**

Chapter 2 #1, 5, 8, 9