

EEL 6586: HW#2

Assignment is due Friday, Feb 8, 2008 in class. Late homework loses $e^{\# \text{ of days late}} - 1$ percentage points.

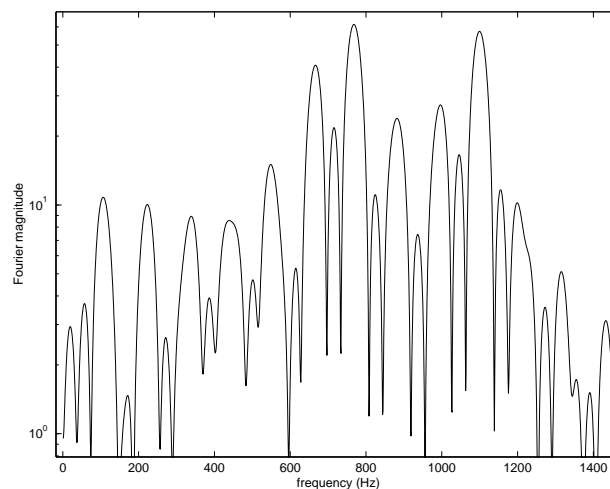
PART A: Textbook problems (Use Matlab only to optionally check your work)

A1 Compute $r[k]$, of the following discrete-time signal:

$$x[n] = \sin(\omega n)$$

Use the definition of autocorrelation in your response (i.e., don't use data samples).

A2 A train of impulses is fed through a simple all-pole model to produce a stationary voiced phoneme. The resulting sequence is windowed by a rectangular window and its Fourier magnitude is plotted as shown below. The sampling rate is 10KHz. Answer the following questions.



- Estimate the frequency of the first two formants (F1 and F2) in Hz.
- What phoneme does this recording most resemble? Explain.
- Estimate the pitch of the voice in Hz.

A3 An infinite train of impulses is created with the following relation

$$e(n) = \sum_k \delta(n + kP)$$

Assume that the sampling frequency is 10kHz.

1. Determine the value of P to create a pitch frequency of 100Hz.
2. The infinite train of impulses is fed through an all-pole model of

$$H(z) = 1/(1 + .9z^{-1} + .81z^{-2})$$

What is the dominant formant frequency in the signal?

3. Is this formant frequency higher or lower than typical first formant frequencies for humans?
4. How will the formant frequency change if pre-emphasis is applied to the signal $(s(n) - 0.95s(n-1))$?

A4 Assume that an infinite impulse train

$$\sum_k \delta(n + kP)$$

is filtered by a vocal-tract model given by $H(z) = 1/(1 + .9z^{-1} + .81z^{-2})$ to produce a speech signal $s(n)$.

1. Derive the difference equation for $s(n)$.
2. Compute the autocorrelation function $r(0)$ for the speech signal $s(n)$.
3. Compute the autocorrelation function $r(1)$ for the speech signal $s(n)$.
4. Compute the single LPC coefficient ($p = 1$) for this system.
5. How does this coefficient compare to the first coefficient when $p = 2$? Explain.

A5 Assume that white noise excitation $w(n)$ is filtered by an *all-pole* vocal-tract model $H(z) = 1/(1 + .25z^{-2})$ to produce a speech signal $s(n)$. $w(n)$ is defined:

$$E\{w(n)w(m)\} = \begin{cases} 1 & m = n \\ 0 & m \neq n \end{cases}$$

In this problem you will use LPC to derive an all-pole approximation to $H(z)$.

1. Derive the difference equation for $s(n)$.
2. Compute the autocorrelation function $r(0)$ for the speech signal $s(n)$.
3. Compute the autocorrelation function $r(1)$ and $r(2)$ for the speech signal $s(n)$.
4. Compute the first two LPC coefficients ($p = 2$).
5. (5 points) Derive $\hat{H}(z)$, the all-pole approximation to $H(z)$. Does your answer make sense?

PART B: Computer Analysis of Speech In this part you will write a program for automatic formant determination. You will run your program on your own recorded phoneme from HW#1 and the voiced phonemes at <http://www.cnel.ufl.edu/hybrid/courses/EEL6586/phonemes>

- B1 For this exercise, determine an approximate order to use for the LPC computation. Explain your reasoning.
- B2 Run the Matlab LPC algorithm with your approximate order from B1 on your own recorded phoneme from HW#1. Use the entire phoneme (no windows). What are the LPC values that you recover? Show a plot of the magnitude of the Fourier transform of $H(z)$, superimposed on the magnitude of the FFT of the original speech. Try different values of the order. Hand in a few plots with different orders and explain the differences among them. Show at least 3 different orders using reasonable values.
- B3 Inverse filter the original recorded phoneme. Using the same orders you used in B2, show plots of the resulting residue signal (zoom in on a few periods).
- B4 Write a procedure to automatically estimate the numerical values of the first two formants of the given phoneme. Explain the principles behind your algorithm in detail. Optimize the order and run it on your own recorded phoneme. How do the results compare to the formant locations you estimated in HW#1?

- B5 Run your algorithm to automatically determine the formants for those given in the directory <http://www.cnel.ufl.edu/hybrid/courses/EEL6586/phonemes>. You must use the same algorithm parameters for each phoneme (no individual tweaking). List the formants for each phoneme in a table. Comment on how well you think your algorithm is working. Do the formant frequencies correspond to the typical values for those phonemes?

As always, hand in all of your Matlab code as the appendix of your homework.