**51:040/51:141 Biological Systems Analysis I**

**Lab 4**

**Instructions:** Create a new MATLAB .m for each part. Each .m file should include comments that indicate problem number and names of the group members. Choose descriptive variable names and use comments as needed to describe your solutions. Each group should hand in a copy of your .m files and results for the problems below to the TA in room G255 by 4 PM Friday on 10/29/04. No late submissions will be accepted.

**Objectives:** Approximating signals using the Fourier series.

**Background:** The Fourier series for a periodic pulse train discussed in class and in homework assignment #5. We will use MATLAB to compute and plot the Fourier series coefficients for a periodic shifted pulse train:

\[ x(t) = \sum_{k=-\infty}^{\infty} p(t - kP - T), \]

where \( P \) is the period of the signal, \( p(t) = q(t + a) - q(t - a) \), \( 2a \) is the pulse width, and \( T \) is the shift parameter. This is a shifted version of the signal shown in the textbook figure 4.1.

We can use the results from homework #5 to show that the Fourier series coefficients for this signal are:

\[
C_m = \begin{cases} 
\frac{2a}{P} & m = 0, \\
\frac{2 \sin(m \omega_0 a)}{m \omega_0 P} e^{-jm \omega_0 T} & m \neq 0,
\end{cases}
\]

where \( \omega_0 \) is the fundamental frequency for this signal.

1. Create a function called `coefficient` in MATLAB file `coefficient.m` to compute the Fourier series coefficients for the signal discussed above.

   Your function should start like this:

   ```matlab
   function cm = coefficient(m, P, a, T)
   %
   % x = coefficient(m, P, a, T)
   % return the m^th Fourier Series coefficient for
   % a shifted pulse train
   %
   % m is the term number of Cm
   % P is the period
   % a is 1/2 the pulse width
   % T is the amount of time shift for the pulse center
   %
   % See Chen example 4.2.1 for details
   %
   ```
Test your function by plotting the coefficients $C_m$ for the case shown in the text on page 119 ($P = 4, a = 1, T = 0$) and for the case covered in homework #5 ($P = 4, a = 1, T = 1$). An example MATLAB script `plot_cm.m` is attached—you can use this to plot the magnitude and phase of the $C_m$ values.

Experiment with the period, pulse width, and time shift parameters and use your observations to answer the following questions:

(a) Are the magnitudes of $C_m$ coefficients odd or even?
(b) Are the phases of $C_m$ coefficients odd or even?
(c) What happens to the magnitude and phase plots of $C_m$ when comparing shifted and unshifted pulse trains? (Hint: try $P = 4, a = 0.125$, and $T = 0$ versus $T = 0.25$.)

2. Modify the `synthesize.m` example from class so that it will reconstruct the shifted pulse train. Save this as a new MATLAB function `synthesize2`. You will need to add one more calling argument (the shift value) and change the calculation of the terms in the Fourier series reconstruction so that your coefficient function is called to compute the $C_m$ values. Your `synthesize2` function should start like this:

```matlab
function x = synthesize2(t, M, P, a, T)
  % x = synthesize(t, M, P, a, T)
  % use Fourier Series synthesis equation to
  % create an approximation to a shifted pulse train
  % M is the 1/2 the number of terms to use, M >= 1
  % P is the period
  % a is 1/2 the pulse width
  % T is the time shift for the pulse center
  %
  % See Chen example 4.2.1 for details
```

Hand in a plot of your reconstructed pulse train for the parameters $t = -10$ to 10 seconds, $P = 4, a = 1, T = 1$ for $M = 5$ and $M = 20$. 

2
The `synthesize.m` class example for Fourier series reconstruction

```matlab
function x = synthesize(t, M, P, a)
    %
    % x = synthesize(t, M, P, a)
    % use Fourier Series synthesis equation to
    % create an approximation to a pulse train
    %
    % M is the 1/2 the number of terms to use, M >= 1
    % P is the period
    % a is 1/2 the pulse width
    %
    % See Chen example 4.2.1 for details
    %
    if M < 1
        error('Must have M > 0!')
    end

    % compute w0
    w0 = (2.*pi)/P;

    % start with the C0 term
    x = 2*a/P * exp(j*0*w0*t);

    % now add each positive and negative Cm term
    for m = 1:M
        % positive m
        x = x + 2*sin(m*w0*a)/(m*w0*P) * exp(j*m*w0*t);
        %
        % negative m
        x = x + 2*sin(-m*w0*a)/(-m*w0*P) * exp(-j*m*w0*t);
        % note, the above two lines could be replaced by
        %
        % x = x + 4*sin(m*w0*a)/(m*w0*P) * cos(m*w0*t);
    end

    % function returns x when the loop is finished
```
%% MATLAB script to plot the cm coefficients

% N is the number of coefficients, we will actually
% plot Cm from m = -N to +N.
N = 50;

if N == 0
    error "Number of coefficients N must be bigger than 0!"
end

P = 4;
a = 1;
T = 1;

cm = zeros(2 * N + 1, 1);
mvalues = zeros(2 * N + 1, 1);

for m = -N:N
    index = m + N + 1;
    mvalues(index) = m;
    cm(index) = coefficient(m, P, a, T);
end

% plot Cm magnitude and phase
% phase plot is in radians, multiply by
% 180./pi to plot it degrees
subplot(2, 1, 1), stem(mvalues, abs(cm));
xlabel('m')
ylabel('|C_m|')
title(sprintf('Lab4 Problem1 P=%.1f, a=%.1f, T=%.1f', P, a, T))

subplot(2, 1, 2), stem(mvalues, angle(cm));
axis([-N,N, -4, 4])
xlabel('m')
ylabel('∠ C_m')