11 Sea Spectrum and Marine Vehicle Pitch Response

1. Make a plot of the spectrum for about one hundred frequencies from zero to 4 rad/s, with modal frequency \( \omega_m = 1 \) rad/s, and significant wave height \( H_{1/3} = 0.90 \)m. 

   *See the top graph in Figure 2 for the wave spectrum.*

2. Confirm that the area under the spectrum is equal to \( E_S \), by making a numerical integration. You can then take this \( E_S \) to double-check \( H_{1/3} \).

   *The area under the curve \( E \) is 0.0502; it was supposed to be 0.0506. Note in the MATLAB code that I randomize the actual frequencies used; this is to avoid related frequencies (as seen in the first problem above). The corresponding significant height is 0.896m; pretty close to the desired value of 0.90m.*

3. Make ten minutes worth of this wave-like data, using a sampling period of 0.1 seconds, and show a plot, with the original \( H_{1/3} \) maximum and minimum levels indicated.

   *See the middle graph in Figure 2 for a time-domain realization of this spectrum.*

4. Compute the parameter \( E_Y \) from the area under \( Y(\omega) \), and so estimate the “significant height” of the pitch motion.

   *The significant height of the pitch motion is 0.26 radians, or about fifteen degrees. The lower graph in the figure shows the multiplication in frequency space.*

%-------------------------------------------------------------------------
% Bretschneider Sea Spectra and Vehicle Pitch Response
%
clear all;

dfreq = .04; % frequency resolution for creating waves, rad/s
maxfreq = 3.99 ; % highest wave frequency made, rad/s
dt = .1 ; % sampling period, s
tff = 600 ; % final time, s
wmodal = 1 ; % modal frequency (used for Bretschneider construction)
E = .05 ; % spectra parameter for Bretschneider:
          % E = sig_height^2 / 16

% no user parameters below this point

% time = 0:dt:tff;  % time vector
n = length(time) ; % number of samples
freq = dfreq:dfreq:maxfreq;  % imposed frequencies
      % (near zero to maxfreq)
freq = freq + (.5-rand(1,length(freq)))*dfreq;
    \% add random components in freq vector

disp(sprintf('Imposed dfreq/available resolution: \%g (best if below one)',
    dfreq/(2*pi/dt/length(time))));

if (pi/dt < max(freq)),
    disp('Must have higher sampling rate to avoid aliasing! -- ABORT.');
    break;
end;

\% make up a Bretschneider spectrum and then get out the amplitudes
\% for the example case
B = wmodal^4*1.25;
H = 4*sqrt(E);
A = 4*B*E;
sBret = A./freq.^5.*exp(-B./freq.^4);

figure(1);clf;hold off;
subplot('Position', [.2 .2 .5 .2]);
plot(freq,sBret,'LineWidth',2);
xlabel('rad/s');
title('Bretschneider Spectrum, H_{1/3} = 0.90m');
print -deps bret_veh1.eps

disp(sprintf('E: \%g vs. \%g.', E, sum(sBret)*dfreq));

amp = sqrt(2*sBret*dfreq);
\% Note that sBret is invariant with dfreq, but amp definitely changes
\% with dfreq, according to PNA def. of spectrum given in the problem.

\% make up the time series, with random phase
phase = 2*pi*randn(length(freq),1);
x = zeros(size(time));
for i = 1:length(freq)
    x = x+amp(i)*cos(freq(i)*time+phase(i));
end

figure(2);clf;hold off;
subplot(211);
plot(time,x); xlabel('Time, seconds')
hold on;
plot([0 tff],[1 1]*H/2,'r--',[0 tff],[-1 1]*H/2,'r--');
legend('Simulation','+/-' H_{1/3}/2');
print -deps bret_veh2.eps

% Here is the transfer function, and the pointwise frequency multiply of
% F*F' with S
i = sqrt(-1);
F = (.4*i*freq + .3) ./ (-freq.^2 + i*freq*1 + 3);
FF = F.*conj(F);

figure(3);clf;hold off;
subplot('Position', [.2 .2 .5 .5]);
plot(freq,FF,freq,sBret,'--',freq,FF.*sBret*20,':','LineWidth',2);
legend('FF*','S','20 x S x FF*',1);
xlabel('rad/s');
print -deps bret_veh3.eps

% Compute the area under the curve and from it get the significant
% response height
FFS = FF.*sBret;
EFFS = sum(FFS)*dfreq;
FFSsig = 4*sqrt(EFFS);
disp(sprintf('Significant response height: %g.', FFSsig));

%---------------------------------------------------------------