The objective of this project is to explore how amplitude and phase distortion within a channel affects a signal being transmitted through that channel.

**Procedure**

Download the Octave file `lab01.m` from the course web site. The code from this file is shown in Appendix A. If you run this code from Octave it should produce the graphs shown in Figure 1. (You will need to have the ue_signal package loaded. Watch the Octave Installation video to learn how to install all of the UE Octave packages.)

The code simulates distortion of a triangular waveform as it passes through a communication channel. Lines 1 – 10 define some basic constants. The waveform has a period of 1 ms (note that time axis in the figure has units of ms) and a corresponding fundamental frequency of 1000 Hz. The first 11 terms in the Fourier series are used to approximate the triangular waveform. These terms correspond to frequencies of 1000 Hz, 2000 Hz, 3000 Hz, ..., 11000 Hz.

Lines 21 – 27 calculate the amplitude terms in the compact trigonometric Fourier series representation of the waveform. Lines 31 – 35 calculate the phase angle terms. Line 38 calculates the channel input waveform from its Fourier series. Line 42 plots this waveform and produces the top graph in Figure 1.

Lines 46 – 62 simulate the output of a distortionless channel. Line 51 creates an 11 element attenuation vector. Notice that all 11 values are equal to 1/2. The amplitude coefficients of the input waveform are multiplied by the attenuation values in line 53 to create the amplitude coefficients for the second
waveform. Line 50 creates an 11 element phase shift vector by multiplying the frequency vector by the constant \(-2\pi t0\) where \(t0\) is set to 0.75 ms. This results in a vector whose elements increase linearly with increasing frequency. This phase shift vector is added to the phase shift terms of the input waveform in line 54. Line 56 calculates the waveform using the modified amplitude and phase terms and line 59 creates the second graph shown in Figure 1. Since the amplitude attenuation is constant across all frequencies and the phase shift is linear we see an undistorted version of the input in the second graph. Notice that the second graph looks like the first one only shifted to the right by 0.75 ms.

Use Figure 1.8 to determine the attenuation (in dB) of a typical telephone channel at frequencies of 1000 Hz, 2000 Hz, 3000 Hz, ..., 11000 Hz. Since the Figure frequency axis only goes up to 3600 Hz you will need to estimate the attenuation at the higher frequencies. Enter the 11 estimated attenuation values into the code at line 68. The values should all be positive dB values. (Line 69 converts the dB values to linear attenuation factors. Run the program again. The third graph should now show the effects of only amplitude distortion in the channel. (The linear phase shift terms from the previous waveform are included in this waveform also.)

Use Figure 1.9 to estimate the actual channel phase shift at the 11 harmonic frequencies between 1000 Hz and 11000 Hz. Enter these values at line 85. You will again need to estimate the phase shift at the higher frequencies. (Note that the figure shows the phase shift as positive, it is actually negative.) The phase shift values should all be in radians. Run the program again. The fourth graph should now show the effects of only phase distortion in the channel. The fifth graph will show the effects of both amplitude and phase distortion.

**Report**
Discuss your results. Why do we call the second waveform an undistorted version of the first one? Which type of distortion (amplitude or phase) seems to have the biggest impact on the waveform? Fit a line to the graph in Figure 1.9 and use the slope of the line to determine the approximate channel delay for this particular channel. Figure 1.8 and Figure 1.9 are the attenuation and phase shift for approximately 7 km of phone line, how do you think Figures 1.8 and 1.9 might change if 14 km of phone line were used. How would the waveform graphs change? Include your estimated attenuation (dB) and phase shift values in your report. Also include the final figure showing all 5 waveforms.
1. % Determine approximate output of telephone channel with triangular wave input
2. clf;
3. N = 11;               % Number of terms in FS (11)
4. T0 = 1e-3;            % Period of waveform
5. t = [0:1/100:3]*T0;   % Time interval for calculation
6. % Fund freq is 1000 Hz
7. % Form freq vector with values [1000 2000 3000 ... 11000]
8. f = [1:N]/T0;       % Frequencies in FS
9. x_a0 = 0;             % DC coefficient
10. splot = 5;            % number of subplots
11. % Complex Fourier series coefficients for triang. wave
12. % c(n) = 0                           for n even
13. % c(n) = 8/(pi*n)^2 * exp(-j*pi/2)   for n = 1, 5, 9, 13, ...
14. % c(n) = -8/(pi*n)^2 * exp(-j*pi/2)  for n = 3, 7, 11, 15, ...
15. % Compute amp. terms in the Fourier series coeff. of a tri. wave
16. % n must be a vector of the form 1:N
17. function r = x_an(n)
18. r = zeros(size(n));
19. ind = (mod(n, 2) == 1);   % Find all odd n values
20. r(ind) = 8/(pi*pi) ./ (n(ind).*n(ind));
21. ind(1:4:end) = 0;         % Negate n = 3, 7, 11, 15 terms
22. end
23. % Compute phase terms in the Fourier series coeff. of a tri. wave
24. % n must be a vector of the form 1:N
25. function r = x_pn(n)
26. r = zeros(size(n));
27. ind = (mod(n, 2) == 1);
28. r(ind) = -pi/2;
29. end
30. % Plot the input waveform, x
31. x = fs_trig2(x_a0, x_an(1:N), x_pn(1:N), T0, t);
32. subplot(1,1,plotno)
33. plot(t/1e-3, x, 'linewidth', 2)
34. grid on
35. title('Telephone Channel Input')
36. % First simulate a pure time delay of 0.75 ms and gain of 1/2
37. t0 = 0.75e-3;
38. P1 = -2*pi*f*t0;   % Phase is a linear function of frequency.
39. A1 = (1/2)*ones(1,N);
40. y1_an = @(n) A1(n).* x_an(n);
41. y1_pn = @(n) P1(n) + x_pn(n);
56. y1 = fs_trig2(x_a0, y1_an(1:N), y1_pn(1:N), T0, t);
57. plotno = plotno + 1;
58. subplot(splot,1,plotno)
59. plot(t/1e-3, y1, 'linewidth', 2)
60. grid on
61. title('Telephone Channel Output w Linear Phase Shift (t0 = ' sprintf("%f",t0*1000) 'ms)'))
62. % Simulation with just Amplitude distortion
63. % Attenuation in dB at 1000, 2000, 3000, ... 11000 Hz
64. % Estimated from Figure 1.8
65. L = [0 0 0 0 0 0 0 0 0 0];
66. att = (10).^(-L/20);
67. y2_an = @(n) att(n).*x_an(n);
68. y2_pn = @(n) P1(n) + x_pn(n);
69. y2 = fs_trig2(x_a0, y2_an(1:N), y2_pn(1:N), T0, t);
70. plotno = plotno + 1;
71. subplot(splot,1,plotno)
72. plot(t/1e-3, y2, 'linewidth', 2)
73. grid on
74. title('Telephone Channel Output w only Amp Dist (no delay)')
75. % Simulation with just Phase distortion
76. % Estimated phase from figure
77. P = -[0 0 0 0 0 0 0 0 0 0].*pi;
78. y3_an = @(n) x_an(n);
79. y3_pn = @(n) P(n) + x_pn(n);
80. y3 = fs_trig2(x_a0, y3_an(1:N), y3_pn(1:N), T0, t);
81. plotno = plotno + 1;
82. subplot(splot,1,plotno)
83. plot(t/1e-3, y3, 'linewidth', 2)
84. grid on
85. title('Telephone Channel Output w only Phase Dist')
86. % Simulation with Amplitude and Phase distortion
87. % Estimated phase from figure
88. y4_an = @(n) att(n).*x_an(n);
89. y4_pn = @(n) P(n) + x_pn(n);
90. y4 = fs_trig2(x_a0, y4_an(1:N), y4_pn(1:N), T0, t);
91. plotno = plotno + 1;
92. subplot(splot,1,plotno)
93. plot(t/1e-3, y4, 'linewidth', 2)
94. grid on
95. title('Telephone Channel Output w Amp and Phase Dist')
96. xlabel('time (ms)')