The Discrete Fourier Transform

Lab 9, DSP

Objective

The students shall know how to use Matlab's `fft()` and `ifft()` functions for frequency analysis of discrete signals.

Theoretical aspects

The Fourier series coefficients of a discrete periodical signal can be computed from one period of the signal with:

Exercises

1. Generate a 100 samples long signal `x` defined as $x[n] = 0.7\cos(2\pi f_1 n) + 1.2\sin(2\pi f_2 n)$, with $f_1 = 0.05$ and $f_1 = 0.1$.

    a. Plot the signal in the top half of a figure (use `subplot()`).
    b. Compute the Fourier series coefficients with `fft()` and plot their magnitude in the lower half of the figure.
    c. Repeat for a signal length of 93 samples. Why do additional frequency components appear in the spectrum?

2. Generate a 39 samples long **triangular** signal `x` defined as:

    - first 10 samples are zeros
    - next, `x` increases linearly from`x(10) = 0` up to `x(19) = 4`, then decreases linearly to `x(29) = 0`.
    - last 10 samples are 0

    a. Plot the signal in the top half of a figure, and the magnitude of the Fourier series coefficients in the lower half.

b. What is the amplitude of the **third harmonic component** in the signal's spectrum?
c. Concatenate 50 zeros at the end of the signal and redo the exercise. What do you observe?

3. Take the Fourier series coefficients of the triangular signal before, and keep only the coefficients of the DC + first two sinusoidal components. Generate the signal from the Fourier coefficients with `ifft()` and plot it.

4. Generate two 10-long random signals `x` and `y`.

   a. Perform **linear convolution** with `conv()`.
   b. Perform **circular convolution** via the frequency domain, using `fft()` and `ifft()`.
   c. Perform **linear convolution** via the frequency domain using the `fft` in N points, with N larger then 19.
   d. Which method of linear convolution is is faster, `conv()` or via `fft()`? Use long signals (e.g. length 40000).

Final questions

1. TBD