Filter Design, poles and zeros

DSP Lab 10

Table of contents

1 Objective

Students should understand how the pole-zero diagram influences the frequency response of a filter, and be able to design basic digital filters using the built-in tools of Matlab.

2 Theoretical notions

In this class, we frequently used Z transforms with polynomials as numerator and denominator, denoted as $B(z)$ (numerator) and $A(z)$ (denominator). A polynomial is completely determined by its roots and a scaling factor.

Definitions:

- the **zeros** of $X(z)$ are the roots of the numerator $B(z)$
- the **poles** of $X(z)$ are the roots of the denominator $A(z)$

The zeros are usually named $z_1, z_2, ...z_M$, and the poles $p_1, p_2, ...p_N$.

The poles and zeros are typically represented graphically in the complex plane, and in simple cases their position can indicate rapidly some properties of the filters.

In this laboratory we will study how the position of zeros and poles influences the amplitude response of a filter.

3 Exercises

1. Open the Filter Design tool in Matlab (call `fdatool` in command line). Draw two zeros and two plots and place them such as to obtain:

    - a Low-Pass filter
    - a High-Pass filter
    - a Band-Pass filter
    - a Band-Stop filter

    Visualize the frequency response, the impulse response and the filter coefficients in each case.

2. Using the controls available in the Filter Design window, design

    - a Low-Pass filter of order 3, IIR, with cutoff frequency 0.1
    - a High-Pass filter of order 3, IIR, with cutoff frequency 0.4
    - a Band-Pass filter of order 4, IIR, with frequency pass-band [0.22 0.28]
    - a Band-Stop filter of order 4, IIR, with frequency stop-band [0.22 0.28]
    - a linear-phase FIR filter of order

Experiment with the various options available. What happens in case of: - a larger filter order? - different filter types (Butterworth / Chebyshev 1 / Chebyshev 2 / Elliptic)?

3. Generate a 300-long periodic square signal, composed of 30 values of 1 followed by 30 values of 0, repeated for 5 times. Filter the signal with every filter from the previous exercise (use `filter()`). Plot a figure with 5x1 subfigures showing the original signal and the outputs of all filters.

4. Apply these filters on the sample signals `mtlb`, `chirp` and `train` (call `load mtlb`, `load chirp` and `load train`). Play the outputs at the speakers (`soundsc()`).

4 Final questions

1. What happens if we place a pole and a zero in the exact same spot? What happens? (think from a mathematical point of view)
2. Can we have multiple zeros with the same value, or multiple poles with the same value?