DTU Health Technology, Hearing Systems section
Building 352, DTU, 2800 Lyngby
Phone: 45 25 39 30, Fax: 45 88 05 77 www.hea.healthtech.dtu.dk/

Course 22051: Signals and Linear Systems in Discrete Time

Cloudy...

Bastian Epp: 45 25 39 53 (office), e-mail: bepp@dtu.dk

Filtering back and forth

1 Poles and zeros - once again

We found that poles and zeros really can do cool stuff. Let's see how they impact the impulse response of the system.

1.1 In-N-out zero/pole

Create a system by placing two zeros ($\zeta_{1,2}$) and two poles ($\rho_{1,2}$) at

$$\zeta_{1,2} = 0.95 \cdot e^{\pm j\pi/4}$$

and

$$\rho_{1,2} = 0.95 \cdot e^{\pm j3\pi/4}$$

- Calculate the filter coefficients B and A from the location of the poles/zeros

  Plot the pole-zero plot, the frequency response and the impulse response.

- Now move the zeros to a new location:

  $$\zeta_{1,2} = 1.05 \cdot e^{\pm j\pi/4}$$

  Replot the figures and focus on the relation between frequency response and impulse response.

- Move the zeros back to their original location and move the poles to a new location:

  $$\rho_{1,2} = 1.05 \cdot e^{\pm j3\pi/4}$$

  Replot the figures and evaluate again the frequency response and the impulse repsonse.

  `useful commands: freqz, poly, impz`

2 Making more of less

The following example comes from a book on DSP. However, there is some error in the book: The filter specs in the book don't match the solution. To improve the book for future generations, let's do a solution that does meet the filter criteria specified. Analytical calculations are helpful up to a certain point - feel free to do them and to include them. But you might as well make use of the tools built-in into Matlab to do the numerics.

Here you will be designing a lowpass filter that meets certain specifications. The approach is similar for different filter types and what you get out is something you can apply and that has properties depending on how you generated it - that's why it is neccessary to understand what is behind different routines.

2.1 A specific example

Design a digital lowpass filter using bilinear transform with prewarping meeting the following specifications:

- A gain of unity at $\omega = 0$
- A gain of no less than -2 dB ($G_p = 0.785$) over the passband $0 \leq \omega \leq 10$
- A gain no greater than -11 dB ($G_s = 0.2818$) over the stopband $\omega \geq 15$
- The highest frequency to be processed $\omega_h = 35$ rad/s

  `useful commands: butter, buttord`

Show the amplitude and phase response of the resulting filter together with the filter specifications in the title of the plot. Write down the transfer function of the resulting filter and specify the filter type. You are pre-warping your frequencies here - what are the values after pre-warping?

2.2 The issue of phase shifts and order

Now we have a filter meeting the required specifications. Now assume your filter coefficients are implemented, compiled and the source codes are lost. You have a filter of the order coming out of its design and there is no way to change the properties of the filter - it is already in production and now you are to use the product you have in your hand.

- When passing a pure tone signal through the filter, it will be scaled and shifted in phase. Assume you are asked to process a signal through your filter - while the magnitude of the transfer function is less important, you definitely need zero phase shift. Use the filter designed above to process a sinusoidal signal such that the resulting phase shift is zero. Plot the original and the processed signals on top of each other.
- Now assume the opposite: You need to have a signal processed where the filter order needs to be about twice/three times what your designed filter has, but the phase is less important. What could you do in order to have a filter with approximately twice or three times the order? Use the filter you just designed to solve this problem. Plot the corresponding frequency transfer functions on top of each other.