Politechnika Łódzka
Instytut Elektroniki

SIGNAL PROCESSING

Laboratory #6:

Signal filtering in Python

M. Kociński, P. Strumiłło

Medical Electronics Division
Institute of Electronics

PURPOSE:

Getting acquainted with basic properties of *Finite Impulse Response* (FIR) filters and *Infinite Impulse Response* (IIR) filters, their design in Python environment and their practical applications.

TASKS:

<u>Introduction:</u>

Difference equation of a digital causal filter is given by:

$$\sum_{k=0}^{M} b(k)x(n-k) = \sum_{k=0}^{N} a(k)y(n-k)$$

which is equivalent to a difference equation:

$$a[0]y[n] = -a[1]y[n-1] - a[2]y[n-2] - \ldots - a[N]y[n-N] + b[0]x[n] + \ldots + b[M]x[n-M]$$

where:

$a[k]$ – autoregressive filter coefficients,
$b[k]$ – moving average filter coefficients.

For the FIR filters all the autoregressive coefficients are zero except $a[0]$=1.
Justify the names for $a[k]$ and b[k] coefficients.

<u>TASK 1:</u>
Write difference equations and block diagrams of the following filters (by default we assume $a[0]=1$, if $a[0]\neq 1$ we need to divide all coefficients values by $a[0] \neq 0$):

- $b[0]$ =4, $b[1]$=2,
- $b[0]$ =2, $a[1]$=0.5
- $b[0]$ =0.5, $a[3]$=1
- $b[0]$ =1, $b[1]$=2, $a[1]$=0.5, $a[3]$=1

I. FIR filters
<u>Pros:</u>
- available simple, linear methods for designing filters,
- linear phase can be easily achieved (symmetry of coefficients),
- are always stable,
- feature short transient states (finite impulse response),

<u>Cons</u>:
- a high number of filter coefficients are required (i.e. high filter order) to obtain steep frequency characteristics.

<u>TASK 2:</u>
A FIR filter is given *b*[0] =0.5, b[1]=0.5. Write a script showing that impulse response of this filter is equal to its moving average coefficients.
hint: define signal *x*=[1, 0, 0, ….,0] of length *N*=100 sample at a frequency rate *fs*=11400 Hz, and use the *lfilter* function from *scipy.signal* package to obtain the impulse response.

<u>TASK 3:</u>
Compute frequency characteristic of the filter defined in Task 2 by using the two following methods:

1) by computing the Fourier transform of the impulse response obtained in Task 2; plot the amplitude and phase spectrum of the filter
2) by using the *freqz* command from the *scipy.signal* package
   What kind of filter frequency characteristic you have obtained?
   Is the phase linear? What is the practical consequence of phase linearity?
3) From within the Python environment play the *voice_noise.wav* wave file (see lab. 3). From the lecture signal database load the wave file *voice_noise.wav* into the Python workspace (the signal is sampled with a rate of f_s=11.4 kHz. Filter the voice signal using the filter from Task 2. Store the filtering result in file *voice_filtered.wav*. Finally, play the *voice_filtered.wav* wave file

<u>TASK 4:</u>
Repeat Task 2 for the FIR filter with coefficients *b*[0] =0.5, b[1]=-0.5

<u>TASK 5:</u>
Compute an impulse response of a filter which is a cascade connection of the filters studied in Task 2 and Task 4. What is the frequency characteristic of this combined filter.

II. IIR filters

<u>Pros:</u>
- steep frequency characteristics can be achieved for low filter orders.

<u>Cons</u>:
- difficulty in keeping the filter chase linear,
- can be unstable.

<u>Task 6:</u>
There is an IIR filter given:

$$y(n) = x(n) - \alpha y(n-1)$$

where: α - is a parameter.
Write a function plotting impulse response of the filter for different values of parameter α.
What is the impulse response for parameter's values:

$\alpha = 0.9$, $\alpha = -0.9$, $\alpha = 1.1$, $\alpha = -1.1$, $\alpha = 1$

Comment the obtain results. What is the range of values for parameter α giving a stable filter?

TASK 7:
Load into the Pylab environment an audio signal *male_voice.wav* sampled at a rate of f_s=11.4 kHz. Use the following IIR filter coefficients *b*[0]=1, *a*[0]=1, *a*[1000]=-0.7 to filter this signal. Save the result in a wave file. Play the file and comment the obtained audio effect.

TASK 8:
Go to scipy.signal reference guide available from the webpage:
http://docs.scipy.org/doc/scipy/reference/signal.html

and find the *Filter design* section. Then select the *firwin* function for designing finite impulse response filter for specified frequency characteristics, e.g. the command *b=firwin*(40,0.7) will compute *M*=40 autoregressive filter coefficients defining a low-pass filter with a cut-off frequency at *fc*=0.7*(*fs*/2). Use *freqz* command to plot the frequency characteristic of this filter.

Follow the examples in the *scipy.signal.firwin* section and define: high-pass, band-pass and band-stop filters. Plot frequency characteristics of these filters.

□ 1/12/2012