Signal and Communication Lab
DII University of Brescia - ITALY

Digital Audio Processing

Lab.1 - March 4, 2014

Alessio Degani, [a.degani@unibs.it]
http://www.ing.unibs.it/alessio.degani

1 Matlab basics

In order to familiarize with MATLAB® interface and commands, try to open, visualize, play and save some wave files. Explore the variuos functionality of MATLAB® commands such as *wavread*, *wavwrite*, *sound*, *soundsc*, *sin*, *plot*, ...

> ⚠ **NOTE: remember that the command "help" (or "doc") is always your best friend!**

2 Basic tone and noise synthesis

- Create and play a 10 seconds (the number of samples depends on the sample rate) white noise sequence (see *rand* command).
- Create a sound s of length d seconds with a simple sinusoid $s(t) = \sin(2\pi f_0 t)$, with $f_0 = 440$ Hz). Experiment with different sample rate f_s (eg. 8 kHz, 22.05 kHz, 44.1 kHz, ...) and different frequency. Hear the results.

A discrete (sampled) sinusoid is not directly indexed using time t. In MATLAB you have to use a discrete index $n \in [1, \ldots, N]$. The discrete index n is related to the time instant in seconds with the following equation: $t = \frac{n}{f_s}$.

3 Beats

In acoustics, a beat is an interference between two sounds of slightly different frequencies, perceived as periodic variations in volume whose rate is the difference between the two frequencies. The volume varies like in a tremolo as the sounds alternately interfere constructively and destructively. As the two tones gradually approach unison, the beating slows down and may become so slow as to be imperceptible.

- Experimets with **beats**. Synthesize a two tone sound $s(t) = \sin(2\pi f_0 t) + \sin(2\pi f_1 t)$ with $|f_1 - f_0| \leq 10Hz$ to clearly hear beats. Now increase f_1 until the beats are gone.

- **Bonus**: Generate a two tone sound with one fixed frequency while the other tone frequency exponentially increases (*exponential sweep*). In this way, you can hear the beats that gradually disappear. This kind of signal can be generated by the following formula

$$s(t) = \frac{1}{2}\left[\sin(2\pi f_0 t) + \sin(2\pi f_0 t + \pi k t^3)\right]$$

A good value for k in a 10 seconds signal is $k = 1$.

> ⚠ **<u>NOTE</u>: Pay attention to the maximum absolute value of generated sounds. A sound sequence amplitude must $\in [-1, +1]$.**

4 Digital Filters: Time domain filtering and filter design tools

- Consider the high-pass (derivative) digital FIR filter with impulse response $h[n] = \frac{1}{2}(\delta(n) - \delta(n-1))$. Calculate the filtered version of the white noise sequence using the convolution in the time domain (see command *conv*). Listen the filtered signal.

- Use the MATLAB filter design tools (commands: *fdesign.lowpass, design*; parameters: N=order, F_c=cut-off freq.) to design a 2^{nd} order low pass filter with normalized cut-off frequency $\omega = 0.5$ $(= \frac{f_s}{4})$. Use the command *filter* to apply the filter object previously designed on a white noise signal. Listen the filtered signal.

- Visualize the impulse and frequency response of the filters with the command *fvtool*.