Digital Signal Processing
Home Assignment 2

Course Instructor: Prof. Dr. Nikolay Shilov
Teaching Assistant: Imre Delgado

Innopolis University
Spring 2022, Second Block

Groups:
B19-**DS**-01
B19-**AAI**-01
B19-**RO**-01

May 1, 2022

Instructions

- This Home Assignment consists in two tasks. They are all to be solved in `python` in a `jupyter` notebook (no `.py` files). The only libraries here to be used are `matplotlib`, `numpy` and `math` **and just those**.
- After solving the tasks a report is to be elaborated where the student provides a summary of the solution process, explaining the procedure, the parameters chosen, the results and displaying the visualizations for every task.
- The student will upload to the moodle the `jupyter` notebook (no links to the drive, no `.py` files), the report and any other files he may had used. (Put it altogether in a zip folder if needed).
- The **deadline** is set to be May 8^{th} (inclusive), the live grading session will start on May 18^{th}, the students can choose the timeslot that better suits them in the second spreadsheet (Sheet 2).

Filter Implementation

Moving Average Filter

The general form of a Finite Impulse Response Filter (FIR) is described by,

$$y[n] = \sum_{k=0}^{M-1} h[k]\, x[n-k]$$

if the filter impulses (the $h[k]$'s) are set to constant coefficients divided by the tap number M, the equation becomes the well-known **Moving Average** Filter given by

$$y[n] = \frac{1}{M} \sum_{k=0}^{M-1} x[n-k]$$

for instance we have

$$\begin{aligned}
M = 2: \quad y[n] &= \frac{1}{2}x[n] + \frac{1}{2}x[n-1] \\
M = 3: \quad y[n] &= \frac{1}{3}x[n] + \frac{1}{3}x[n-1] + \frac{1}{3}x[n-2] \\
M = 4: \quad y[n] &= \frac{1}{4}x[n] + \frac{1}{4}x[n-1] + \frac{1}{4}x[n-2] + \frac{1}{4}x[n-3]
\end{aligned}$$

an so on. Consider the moving average filter for $M = 3$,

- compute its **system function** H, plot its poles and zeros in the complex plane.
- compute the magnitude $|H(z)|$, the argument $\arg(H(z))$ of your system function, and plot both in the frequency domain.
- implement the moving average filter for any given M. Sample 30 points of the signal

  $$x(t) = \cos(\frac{1}{2}\pi t)$$

  and test your implementation for $M = 3, 6, 9$.

The Window Method

Recall that oscillations tend to appear when we abruptly truncate any signal. Windows help to considerably reduce these spurious oscillations providing a cleaner output after truncation. Implement the following window functions for any number of points M,

- the Bartlett window
- the Blackman window
- the Hamming window
- the Hann window

test your implementation versus the one in the `signal.windows` library by displaying them vis-à-vis, then visualize the frequency response, i.e. apply the Fourier transform to your window, then shift the spectrum to the center and take its absolute value. For better visualizations apply logarithm to your final computation and plot it in the frequency domain.