Lecture "Digital Signal Processing"

Prof. Dr. Dietrich Klakow, Summer Term 2021

Assignment 7

Submission deadline: 7 June 2021, 23:59

Submission Instructions:

You have one week to solve the assignments.
The code should be well structured and commented. Do not use any Matlab-Toolbox or Python external libraries if it is not mentioned that you can use them.

- You are required to hand in your solutions in a group of two students.
- There are two parts in this assignment: theoretical part and practical part.
- The practical part that is solved with Python should be submitted as an ipynb file (Jupyter Notebook or Google Colab), where every function is written in a separate block.
- The theoretical part can either be written by hand and scanned, or typed with LaTeX in the text / markdown area of ipynb file.
- Submission of both parts should be done via Microsoft Teams by one of the group members.
- Submission should be named as: Ex07_matriculationnumber1_matriculationnumber2.zip

The submission should contain the following files:

- file "README" that contains an information on all team members: name, matriculation number, and email address.
- code files
- file "answers.pdf" which contains answers to the questions appearing in the exercise sheet. *Note: If you use ipynb file, you don't have to submit "answers.pdf". You can embed your scanned copy or write your answers in the text / markdown area.*

1 (1.5P) Wiener - Khinchin Theorem

Show that the Fourier transform of the spatial autocorrelation function of a real-valued random field $f(x, y)$ is equal to the power spectrum $|\hat{F}(u, v)|^2$ of the field. *Note that* $\hat{F}(u, v)$ *is the Fourier transform of the field* $f(x, y)$.

2 Wiener Filter Derivation

2.1 (2P) Subtask

Based on the model given in the lecture, we know that in time domain:

$$x[k] = s[k] + n[k] \tag{1}$$

where $x[k]$ is the noisy signal, $s[k]$ is the clean signal, and $n[k]$ is the noise. And the filtered signal $\tilde{s}[k]$ can be obtained from:

$$\tilde{s}[k] = x[k] * h[k] \tag{2}$$

where $h[k]$ is the filter. In Fourier domain, derive the Wiener filter $\hat{H}[\omega_k]$ in terms of $\hat{R}_{xx}[\omega_k]$, which is the power spectrum of $x[k]$, and $\hat{R}_{sx}[\omega_k]$, which is the cross-power spectrum of $s[k]$ and $x[k]$, based on the minimization of the expected value of $\hat{J}[\omega_k]^2$, where $\hat{J}[\omega_k]$ is the Fourier transform of the error function.

2.2 (1P) Subtask

Rewrite the solution you have obtained in section 2.1 in terms of $\hat{R}_{nn}[\omega_k]$ and $\hat{R}_{ss}[\omega_k]$, which are the power spectrum of $n[k]$ and $s[k]$, respectively, based on the assumption that the clean signal $s[k]$ and noise $n[k]$ are uncorrelated.

2.3 (0.5P) Subtask

Assuming that the Wiener filter $\hat{H}[\omega_k]$ in Fourier domain has been obtained and we know $\hat{X}[\omega_k]$, which is the Fourier transform of the noisy signal $x[k]$, show how to obtain the filtered signal $\tilde{s}[k]$ in time domain.

3 Implementation of Wiener Filter

Allowed Python libraries: numpy, math, and matplotlib.pyplot. Matlab libraries would be the equivalent ones.

3.1 (1.5P) Wiener Filter Coefficients

Write a function that takes three inputs namely an input signal x, a stationery zero-mean process, and an odd filter length M, and generates output of the Wiener filter coefficients h. *Hint: You may need to use the autocorrelation function that you have written in the Assignment 4, and M = 2N + 1. Note that h contains symmetric coefficients.*

3.2 (3.5P) Filtered Signal Reconstruction

1. Generate a sinusoidal wave of amplitude equals to 3 Volts, with sample rate of 20 samples/second of a total duration 50 seconds.

2. Generate random noise that is sampled from a standard normal distribution.

3. Generate a noisy signal from above, based on the assumption that the noise is additive.

4. Plot each of the three above against time (i.e. total of 3 plots) with properly labelled axes and titles. Properly labelled axis includes the title and its unit, e.g. time (second).

5. Generate the Fourier spectrum (i.e. magnitude) of the three above (item number 1, 2, 3) and plot them against frequency. The plots should be properly labelled and titled. Ensure that in the plots, the Fourier spectrum has been shifted to the centre such that the symmetric positive and negative frequencies contribution can easily be seen.

6. Compute the Wiener filter coefficients using the function that you have written in section 3.1 (use M = 101), and obtain the reconstructed / filtered signal in time as well as in frequency domain. *Hint: You may need to do a convolution that can be performed efficiently in frequency domain.*

7. Plot the three results obtained in item number 6 (i.e. Wiener filter coefficients, filtered signal in time domain, and filtered signal in frequency domain) separately with properly labelled axes and titles. The plot in frequency domain should also follow the requirement of item number 5. How do you interpret your result?