Assignment 4 in Signals and Transforms

Discrete Fourier Transform

Introduction

In this assignment we are going to have a closer look at the discrete Fourier transform (DFT), some of its properties, and how it can be used to approximate the Fourier transform of a signal. After successful completion of this assignment, you are able to

- calculate the spectrum approximations for sinusoidal signals using the discrete Fourier transform;
- explain zero-padding, smearing, and leakage and their effects on the spectrum approximation;
- explain the difference between graphical and spectral resolution.

To prepare for this assignment, you should at least have read the following chapters of the lecture notes:

- Chapter 3: Continuous Time Fourier Series and Transform
- Chapter 6: Sampling and Reconstruction
- Chapter 7: Discrete Time Signals and Systems
- Chapter 8: Discrete Time Fourier Analysis

Assessment

The assignment is assessed based on a written report that must answer all the questions and points asked in a coherent way. In particular, *do not repeat the questions* but structure your report in sections using the task numbers as titles. Include the final figures you are asked to generate but not the intermediate ones and explain the figures and results as requested. Recall that all figures must be referenced in the text, they must be properly scaled such that they are legible, and they must have a caption and be properly labeled (including units, where applicable). The report must not exceed *three (3)* pages using a font size of at least 11 pt, including all figures, references, etc. Recall that the assignment is graded anonymously. Hence, do not include your name in the report but do include a proper title. Please refer to the Study Guide for further instructions.

The total number of points that can be achieved in this assignment is 20 points. 15 points are given for correctly implementing the tasks and answering all the questions/providing the requested explanations and interpretations (the exact number of points per task is indicated for each task). Another 5 points are given for the structure of the report. To pass the assignment, you need to get at least 75 % of the points (i.e., 15 points), see the study guide and course homepage for further details on assessment and grading.

Additional reporting instructions: You only have to include a total of four (4) figures in your report: One for the signals $x_1[k]$ and $x_2[k]$ in Task 1, two containing all the results for Tasks 2, and one for the results of Tasks 3.

Preparations

This assignment, as well as all other assignments, are preferably implemented in Python (except for the theoretical tasks, of course). You are free to use any other scientific computation tools such as Matlab or Julia, as long as you are able to implement the tasks. However, support is only provided for Python. See the preparation instructions for Assignment 1 if you need further information.

While solving the tasks, it might be handy to have the Python (and especially NumPy, matplotlib, and SciPy) documentations ready. You can find these at:

- NumPy: `https://numpy.org/doc/stable/`
- matplotlib: `https://matplotlib.org/stable/contents.html`
- SciPy: `https://docs.scipy.org/doc/scipy/reference/`

In addition to the usual import, you will also need SciPy's `fft()` function:

```
from scipy.fft import fft
```

If this throws you an error, use the legacy `fft()` function instead:

```
from scipy.fftpack import fft
```

Tasks

1. Consider two continuous time cosine signals $x_1(t)$ and $x_2(t)$ with amplitude $A_1 = A_2 = 1$, no phase shift, and natural frequencies $f_1 = 21\,\text{Hz}$ and $f_2 = 22\,\text{Hz}$, respectively.

   a) (2p) What are the expressions for the spectra of the two signals?

   b) (2p) Assume that the two signals are sampled using the sampling frequency $f_s = 128\,\text{Hz}$ to obtain the two discrete-time signals $x_1[k] \triangleq x_1(kT_s)$ and $x_2[k] \triangleq x_2(kT_s)$. What are the expressions for the two signals? Are the signals $x_1[k]$ and $x_2[k]$ periodic discrete time signals? If so, what are their periods and how do the periods relate to the continuous-time periods?

      Generate $K = 20$ samples of the signals $x_1[k]$ and $x_2[k]$ and plot the two discrete time signals in the interval $0 \leq k < K$ into the same figure. Recall that discrete time signals are plotted using stems (i.e., undefined between instants k) and make sure to properly style your stems (see the `ax.stem()` documentation, in particular how you change the colors of the markers, the stems, and the baseline), and label and style your figure. Also note that fractional x-ticks do not make sense for discrete time signals; you can use `ax.set_xticks()` to change that.

2. Use the DFT to calculate an approximation of the spectra $X_1(f)$ and $X_2(f)$.

   a) (4p) First calculate the DFT $X[l]$ given by

   $$X[l] = \sum_{k=0}^{K-1} x[k] e^{-\mathrm{j}\frac{2\pi kl}{L}}$$

   with $L = K$. To do this, use the SciPy function `fft()`:

   ```
   X = fft(x)
   ```

   This will give you a vector of the DFT for $X[0], X[1], \ldots, X[L-1]$, where the lth frequency bin corresponds to the frequency

   $$f_l = \frac{l}{L} f_s.$$

   To create a vector for the frequencies, use

   ```
   fl = np.arange(0, L)/L*fs
   ```

   Next, scale the DFT values to compensate for the scaling introduced by the DFT. This is achieved by multiplying $X[l]$ with the scaling factor $1/K$ to obtain

   $$X(f_l) \approx \frac{1}{K} X[l].$$

   Finally, plot the magnitudes of $X_1(f_l)$ and $X_2(f_l)$ (as functions of the frequency f_l) in two separate figures (use `np.abs()` to get the magnitudes from the complex-valued DFTs). When plotting, recall that the DFT yields a discrete spectrum! Furthermore, since we know that the spectra are Hermitian, it is enough to plot their first halves, that is, for frequencies $f_l \leq \frac{f_s}{2}$. (To save space, do not include these figures in your report; we will add more curves later on and it is enough to include these complete figures.)

   Compare the two figures to each other and the spectra that you would expect for $x_1(t)$ and $x_2(t)$ from Task 1. What is your best guess for the frequencies f_1 and f_2 by just looking at the spectra? Explain why this does or does not match with the true values for the frequencies.

   b) (2p) Zero-pad the DFT, that is, calculate the DFT for $L = 256$ points (see the `fft()` documentation on how to achieve this) and approximate the spectrum as in a). Plot the result in the same figures as the results above (to improve readability, it might be useful to plot the spectra in reverse order, i.e., first the spectra from this task and then the one from a)). Explain the differences. What are the most likely frequencies of the cosines?

Comparing the results from a)–b), explain the effects of zero padding in terms of the number of samples K, the number of frequency bins L, as well as the graphical and spectral resolution.

Properly label and style your two figures which now should contain all the spectra from above. Only include these two figures for these two questions in your report and explain the results based on these figures.

3. Generate a new signal $x_3[k] = x_1[k] + x_2[k]$ with $N = 20$ samples.

   a) (3p) Calculate the spectrum using the DFT as in Task 2 (once with and once without zero-padding) and plot the result in a new figure. Based on the spectrum alone, what is your best guess for how many cosines there are in the signal? What are their most likely frequencies? Explain why this does or does not match with the knowledge you have about the signal. Explain whether and why zero-padding does or does not improve the situation.

   b) (2p) Based on your answers in a), think about what actually might improve the situation. Try your solution, plot the result in the same plot as the results from a), and explain them.