Solved examples on topics presented in Lecture 5

Aliasing in DTFT spectrum: See also the corresponding `Matlab script`

- Compare, for sampling frequency $f_s = 100$ Hz, a spectrum of two harmonic functions: $x(t) = \cos(2\pi 30t)$ (no aliasing) and $x(t) = \cos(2\pi 70t)$ (the sampling theorem does not hold, aliasing).

Down-sampling, decimation: See the corresponding `Matlab script`

Up-sampling, interpolation: See the corresponding `Matlab script`

Interpolation, speech signal: See the corresponding `Matlab script`

DFT implementation: See also the corresponding `Matlab script`

- Create a function to compute DFT by definition formula. Compare its output with Matlab function `fft()`.

DFT computation and properties of the DFT spectrum:

- Compute DFT of a signal given by

$$x[n] = 0.5\delta[n] + 1.5\delta[n-1] + 2.5\delta[n-3]. \tag{1}$$

- When nothing else is specified, the DFT should be computed using the whole *duration* of the signal, in our case $N = 4, n = 0 \dots 3$. The DFT output (for frequency indices $k = 0 \dots 3$) is obtained by substitution in the definition formula, i.e.,:

$$\begin{aligned}
X[k] &= \sum_{n=0}^{N-1} x[n]e^{-j2\pi nk/N}, \ \ 0 \le k \le N-1 \\
X[0] &= \sum_{n=0}^{3} x[n]e^{-j2\pi n0/4} \\
X[0] &= 0.5 \cdot e^0 + 1.5 \cdot e^0 + 2.5 \cdot e^0 = 4.5 \\
X[1] &= \sum_{n=0}^{3} x[n]e^{-j2\pi n1/4} \\
X[1] &= 0.5 \cdot e^{-j2\pi 0\cdot 1/4} + 1.5 \cdot e^{-j2\pi 1\cdot 1/4} + 2.5 \cdot e^{-j2\pi 3\cdot 1/4} = 0.5 + 1.5(-j) + 2.5(+j) = 0.5 + j \\
X[2] &= \sum_{n=0}^{3} x[n]e^{-j2\pi n2/4} \\
X[2] &= 0.5 \cdot e^{-j2\pi 0\cdot 2/4} + 1.5 \cdot e^{-j2\pi 1\cdot 2/4} + 2.5 \cdot e^{-j2\pi 3\cdot 2/4} = 0.5 + 1.5(-1) + 2.5(-1) = -3.5 \\
X[3] &= \sum_{n=0}^{3} x[n]e^{-j2\pi n3/4} \\
X[3] &= 0.5 \cdot e^{-j2\pi 0\cdot 3/4} + 1.5 \cdot e^{-j2\pi 1\cdot 3/4} + 2.5 \cdot e^{-j2\pi 3\cdot 3/4} = 0.5 + 1.5(+j) + 2.5(-j) = 0.5 - j
\end{aligned}$$

- Using the formula for spectral resolution, the samples with indices $k \in \{0, 1, 2, 3\}$ correspond to frequencies $\omega_k \in \{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$ according to

$$\omega_k = \frac{2\pi k}{N} = \frac{\pi k}{2}, k = 0 \cdots 3. \tag{2}$$

- Substitution of negative indices $k \in \{-4, -3, -2, -1\}$, results into negative frequencies $\omega_k \in \{-2\pi, \frac{-3\pi}{2}, -\pi, \frac{-\pi}{2}\}$.
- DFT is a sampled variant of DTFT, it preserves its properties concerning the symmetry (for real-valued input sequences) and periodicity. DTFT has even real part and odd imaginary part (even magnitude and odd phase) and it is periodical with period 2π. DFT has even real part and odd imaginary part and it is periodical with period N (which corresponds according to (2) to $\omega_N = 2\pi$, see Lecture 6 for details).
- In our case, when $N = 4$, then $X[0] = X[-4] = X[4]$, $X[1] = X^*[-1] = X^*[3]$ a $X[2] = X[-2] = X[6]$, where $(\cdot)^*$ is complex conjugation. Elements $X[0]$ and $X[N/2]$ are always real-valued.
- `Matlab`: The computation can be verified using Matlab via command `fft(x,N)` (fft - Fast Fourier transform), where `x` is the input sequence and `N` is the considered length of the input/output.
- When plotting the magnitude spectrum, the following commands should be used:
```
x = [0.5 1.5 0 2.5];
N = length(x);
k = 0:(N-1);
X = fft(x,N);
stem(k,abs(X));
```
- On the independent variable axis can also be the digital frequency ω or analog frequency f, when the sampling frequency f_s is known. In our case, considering sampling frequency 16 kHz
```
w = (0:(N-1))/N*(2*pi);
stem(w,abs(X));
f_s = 16000;
f = (0:(N-1))/N*(f_s);
stem(f,abs(X));
```
- Note that DFT is usually evaluated in the interval $k = 0 \ldots N-1$ for $\omega \in< 0, 2\pi)$. In contrast, DTFT is usually plotted in the interval $\omega \in< -\pi, \pi)$. However, due to periodicity and symmetry, both intervals contain the same information and important is only half of the interval $k = 0 \ldots N/2$, $\omega \in< 0, \pi >$, that is $f \in< 0, f_s/2 >$.