1 Lecture Outline

Reading: Chapter 9 DFT/FFT Algorithms

- Physical vs. Computation Resolution (Section 9.3)

2 Physical versus Computation Resolution

There are two parameters controlling the DFT: L the length of the signal and N the number of frequency evaluation points. The bin width (called computational frequency resolution)

$$\Delta\omega_{\text{bin}} = 2\pi/N \tag{1}$$

represents the spacing between DFT frequencies. The physical frequency resolution

$$\Delta\omega = c2\pi/L \tag{2}$$

refers to the minimum resolvable frequency separation between two sinusoidal components. Here c depends on the window used i.e. $c = 1$ for the rectangular window and $c = 2$ for the Hamming window.

Example: We take the DTFT for a length-256 point signal composed of three sinusoids; frequency spacing between sinusoids is 0.314 rads/sample. The three sinusoids have a normalized frequency

$$\begin{aligned} f_1/f_s &= 0.2 \\ f_2/f_s &= 0.25 \\ f_3/f_s &= 0.3 \end{aligned} \tag{3}$$

Figure 1: Orfanidis p. 483 Figure 9.3.1

A few observations:

1. If the length of the signal is not large enough to provide sufficient physical resolution, then there is no point increasing N (length of the DFT) since this only puts more points on the graph Figure 9.3.1 (a) versus (b).

2. We see that the peaks of the sinusoids coincide exactly with one of the DFT frequencies. This of course will happen whenever the sinusoid frequency is an exact multiple of $2\pi/N$ (rads/sample) or fs/N (Hz), i.e. $\omega_0 = k_0 2\pi/N$. Typically, this scenario is unlikely and the DFT will miss the exact peaks. See Figure 9.3.2.

3. It may happen that peaks in the DFT spectrum do not correspond to the correct frequencies. This phenomenon called biasing, is caused by a lack of adequate physical resolution especially when the sinusoidal frequencies are too close to each other and the sidelobes from the window interact strongly.

We can estimate the frequency of a sinusoid even if it does not coincide with a DFT frequency, i.e. not a multiple of $2\pi/N$. In this case, we have inherently rounded k_0 and the frequency estimate is given by

$$\omega_0 = k_0 2\pi/N \tag{4}$$

or in Hz as

$$f_0 = k_0 f_s/N. \tag{5}$$

For each jump in k_0 by 1, the corresponding frequency jumps by $2\pi/N$. Due to the rounding of k_0 (a jump of up to ± 0.5) the frequency estimate can therefore be in error by a maximum of $f_s/(2N)$.