The DFT and the Fast Fourier Transform (FFT)

1. Introduction

In these notes, we briefly describe the *Fast Fourier Transform (FFT)*, as a computationally efficient implementation of the *Discrete Fourier Transform (DFT)*.

2. The Fast Fourier Transform (FFT)

A. Introduction

Here we consider the amount of computation required for computing the Discrete Fourier Transform (DFT). If we naively apply the definition of the DFT,

$$X(k) = \sum_{n=0}^{N-1} x[n]e^{-\mathbf{j}2\pi nk/N},\ 0 \le k \le N-1, \tag{1}$$

we note that each $X(k)$ requires a number of operations proportional to N; since we need to compute N different $X(k)$, it would appear that the DFT algorithm requires on the order of N^2 operations, which we denote as $O(N^2)$. There exists, however, a computationally efficient algorithm for computing the DFT which is known as the *Fast Fourier Transform* (*FFT*).[1] For sequences of length,

$$N = 2^m,\ m \in \{1, 2, 3, \dots\}, \tag{2}$$

— that is, sequence lengths that are powers of two — the computational cost of computing the DFT using the FFT algorithm reduces from $O(N^2)$ to $O(N\log_2 N)$. It is important to note that while computationally different, the FFT algorithm is functionally equivalent to the DFT; it simply achieves computational savings by exploiting symmetries in the definition of the DFT.

B. Comparison of computational costs

The table below illustrates the computational costs associated with the DFT and the FFT algorithms in terms of the number of real-valued multiplications and additions for different values of N. Note that while for small values of N, the computational savings of the FFT are relatively modest, for larger values of N, the computational savings become enormous. It is difficult to overstate the importance of the FFT algorithm in the development of modern DSP applications; without it, many of the techniques that have been developed in the DSP field would not be computationally tractable for long discrete-time sequences.

N	*DFT (real multiplies)*	*DFT (real additions)*	*FFT (real multiplies)*	*FFT (real additions)*
	$4N^2$	$2N(2N-1)$	$2N\log_2 N$	$3N\log_2 N$
2	16	12	4	6
4	64	56	16	24
8	256	240	48	72
16	1,024	992	128	192
32	4,096	4,032	320	480
64	16,384	16,256	768	1,152
128	65,536	65,280	1,792	2,688
256	262,144	261,632	4,096	6,144
512	1,048,576	1,047,552	9,216	13,824
1,024	4,194,304	4,192,256	20,480	30,720

1. *In fact, the Fast Fourier Transform refers to a set of computationally efficient techniques for computing the DFT, not just a single algorithm.*

C. How does it work?

Thus far, we have said little about how the FFT achieves its computational savings over the DFT. While a complete discussion of this topic is beyond the scope of this course, below we give the general idea behind the FFT algorithm. Conceptually, the FFT algorithm begins by deconstructing a DFT of length N into the sum of two shorter DFTs of length $N/2$. Then, it further deconstructs each of these DFTs into the sum of two DFTs of length $N/4$. This process continues until the DFT computation has been reduced to the computation of $N/2$ DFTs of length two each. Note that a DFT of length two is easy to compute:

$$X(0) = \sum_{n=0}^{1} x[n]e^{-\mathbf{j}2\pi n(0)/2} = x[0] + x[1] \tag{3}$$

$$X(1) = \sum_{n=0}^{1} x[n]e^{-\mathbf{j}2\pi n(1)/2} = x[0] - x[1] \tag{4}$$

In constructing the N-length DFT, the DFTs of length two are recombined to form DFTs of length four, then the DFTs of length four are recombined to form DFTs of length eight, and so on and so on. In this building up process for the N-length DFT, some of the shorter DFTs must be multiplied by complex exponentials to yield the correct result (i.e. the DFT as defined).

3. Conclusion

These notes conclude our discussion of frequency transforms. Now that we have some understanding of analyzing the frequency content of signals (and the importance of this task), we will transition our discussion from merely *analyzing* the frequency content of signals to *modifying* the frequency content of signals through *discrete-time systems*. Throughout that discussion, our knowledge and understanding of frequency transforms and analysis will play a crucial role.