Chapter 2

Discrete-Time Signals

In this chapter we will introduce more formally the concept of a discrete-time signal and we will establish an associated basic taxonomy which we will use in the remainder of the course. Historically, discrete-time signals have often been introduced as the discretized version of continuous-time signals, i.e., as the *sampled* values of analog quantities such as the voltage at the output of an analog circuit; accordingly, many of the derivations proceeded within the framework of an underlying continuous-time reality. In truth, the discretization of analog signals is only part of the story, and a rather minor one nowadays. Digital signal processing, especially in the context of communication systems, is much more concerned with the *synthesis* of discrete-time signals rather than with sampling. That is why we introduce discrete-time signals from an abstract, self-contained point of view.

2.1 Continuous and Discrete-Time Signals

Almost all the signals we described were defined over a continuous space. For example we described a speech signal as a pressure-intensity function over time. Therefore a speech signal $s(t)$ is defined over $t \in \mathbb{R}$, the real line. This gives the speech signal a continuous-time representation. However, most computers deal with discrete-time signals and therefore a fundamental question is how we can represent a continuous system in discrete-time.

$$s[k] = s(kT_s), \qquad k \in \mathbb{Z} \tag{2.1}$$

By sampling a continuous-time signal one can produce a discrete-time signal. However a basic question that arises is how much fidelity such a representation has with respect to the original signal. This turns out to be a fundamental question in signal representation.

For signals with some properties it is possible to have a completely faithful representation using a discrete-time signal. In some other cases, the representation can be as faithful as one wants by appropriately choosing the sampling period T_s. This is a question we will re-visit later.

2.2 Informal Description of the Sampling Theorem

Given a signal

$$x(t), \qquad t \in \mathbb{R}$$

a discrete-time signal can be obtained by sampling it at regular intervals of T_s seconds, i.e.,

$$x[n] = x(nT_s), \qquad n \in \mathbb{Z}, \qquad T_s \in \mathbb{R}.$$

If T_s is sufficiently small, then it is possible to reconstruct the original signal $x(t)$ from its samples $\{x(kT_s)\}_{k=-\infty}^{k=\infty}$ through an interpolation function.
Take a sinusoidal continuous-time signal

$$x(t) = A\cos(2\pi f_0 t + \theta_0). \tag{2.2}$$

If we sample this at times nT_s, we produce

$$x[n] = x(nT_s) = A\cos(2\pi f_0 nT_s + \theta_0). \tag{2.3}$$

Now suppose $f_0' = f_0 + \frac{1}{T_s}$, and that we have a signal

$$x'(t) = A\cos(2\pi f_0' t + \theta_0) \tag{2.4}$$

Sampling $x'(t)$ at the same times nT_s gives

$$\begin{aligned} x'[n] &= A\cos(2\pi f_0' nT_s + \theta_0) && (2.5)\\ &= A\cos(2\pi (f_0 + \frac{1}{T_s}) nT_s + \theta_0) && (2.6)\\ &= A\cos(2\pi f_0 nT_s + \theta_0) && (2.7)\\ &= x[n] && (2.8) \end{aligned}$$

Therefore, if we sample at intervals of T_s seconds, we can only distinguish frequencies between 0 and $\frac{1}{2T_s}$, all others produce the same samples as a sinusoid of lower frequency(or negative frequency)

Example 2.1 *Sampling of sinusoidal signals:*
Let

$$x_1(t) = \cos(2\pi 10t),$$

$$x_2(t) = \cos(2\pi 50t)$$

and let $T_s = \frac{1}{40}$ *, i.e.,* $F_s = 40Hz$ *is the sampling frequency. Then*

$$x_1[n] = \cos(2\pi\frac{10}{40}n) = \cos(\frac{\pi}{2}n)$$

$$x_2[n] = \cos(2\pi\frac{50}{40}n) = \cos(\frac{5\pi}{2}n) = \cos(\frac{\pi}{2}n + 2\pi n) = \cos(\frac{\pi}{2}n)$$

Hence the sampled version of $x_1(t)$ *and* $x_2(t)$ *at a sampling rate of* $40Hz$ *are indistinguishable.*

One can see that if we know the largest frequency that occurs, we can determine the sampling frequency needed to faithfully and uniquely represent such a signal.
In general, if

$$x_a(t) = \sin(2\pi f_0 t) \tag{2.9}$$

and T_s is the sampling period, then

$$x[n] = \sin(2\pi f_0 n T_s) = \sin(2\pi[f_0 T_s + r]n) \tag{2.10}$$

Hence $f_0' = f_0 + \frac{r}{T_s}$, $\quad r \in \mathbb{Z}$, are indistinguishable.
For real signals, for every frequency f_0, there is a mirror image at $-f_0$.

Example 2.2 *Negative and positive frequency:*

$$\sin(-2\pi f_0 t) = -\sin(2\pi f_0 t)$$

$$\cos(-2\pi f_0 t) = \cos(2\pi f_0 t)$$

We need to accommodate *both* the positive and negative mirrors for distinguishablity.

Hence, we can say that for a sampling period T_s, only frequencies

$$-\frac{1}{2T_s} \leq f \leq \frac{1}{2T_s} \tag{2.11}$$

are distinguishable. Therefore, if the maximum frequency is F_{max}, we need $F_s = 2F_{max}$ for any hope of reconstruction.

Example 2.3 *Sampling*
Let the signals $x(t)$ and $x'(t)$ be given by

$$x(t) = \sin(2\pi\epsilon t) + \sin(2\pi 75t) + \sin(2\pi 150t),$$

$$x'(t) = \sin(2\pi 75t).$$

If we sample these signals with a sampling frequency of $F_s = (150 + \epsilon)$ Hz we get

$$\begin{aligned} x[n] &= \sin(2\pi\epsilon\frac{n}{150+\epsilon}) + \sin(2\pi\frac{75}{150+\epsilon}n) + \sin(2\pi\frac{150}{150+\epsilon}n) \\ &= \sin(2\pi\epsilon\frac{n}{150+\epsilon}) + \sin(2\pi\frac{75}{150+\epsilon}n) + \sin(2\pi\frac{-\epsilon}{150+\epsilon}n) \\ &= \sin(2\pi\frac{75}{150+\epsilon}n) \end{aligned}$$

and

$$x'[n] = \sin(2\pi\frac{75}{150+\epsilon}n).$$

The sampled signals are indistinguishable!
However, if $F_s = (300 + \epsilon)Hz$ then

$$x[n] = \sin(2\pi\epsilon\frac{\epsilon}{300+\epsilon}n) + \sin(2\pi\frac{75}{300+\epsilon}n) + \sin(2\pi\frac{150}{300+\epsilon}n)$$

and

$$x'[n] = \sin(2\pi\frac{75}{300+\epsilon}n)$$

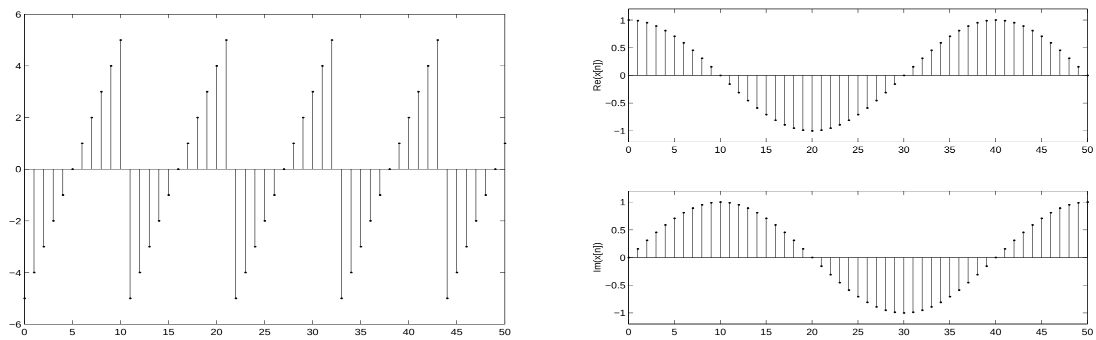
are completely distinguishable.

Remark: Distinguishability is a necessary but not sufficient condition for being able to reconstruct the original signals.

The following is a deep result in signal representation and has impact in a lot of areas.

Theorem 2.1 *If the signal $x(t)$ satisfies the regularity condition that it is band-limited and f_{max}, is the largest frequency (i.e. bandwidth), then samples of $x(t)$, $\{x(kT_s)\}$ for $\frac{1}{T_s} > 2f_{max}$ will be sufficient to reconstruct the signal $x(t)$.*

This is a result we will return to and prove formally, but for now it is important to understand that the discrete-time sequences that we will work with for the next few weeks can be connected to physical, continuous-time signals under some mild conditions.



(a) (b)

Figure 2.1: Examples of signals. (a) triangular wave; (b) complex exponential .

2.3 Discrete-time sequences

A sequence is a set of numbers denoted as

$$x[n], \qquad n \in \mathbb{Z},$$

i.e., defined over the set of integers.

A discrete-time sequence can arise from sampling a continuous-time sequence, but it can also arise in its own. For example

$$x[n] = (n \mod 11) - 5, \tag{2.12}$$

which is the "triangular" waveform plotted in Figure 2.1-(a), or

$$x[n] = e^{j\frac{\pi}{20}n} \tag{2.13}$$

which is a complex exponential of period 40 samples and which is plotted in Figure 2.1-(b). Two example of a sequence drawn from the real world are

$$x[n] = \text{The average stock market index in year } n,$$

and

$$x[n] = \text{Number of hits to a web-page in the } n^{th}\text{hour},$$

which are inherently discrete-time and therefore need not be represented as a physical modeling of a continuous-time signal. Therefore, we will deal with discrete-time signals in their own merit and connect them to continuous-time entities much later in the class.

2.3.1 Basic Signals

The following sequences are fundamental building blocks in the theory of signal processing:

- The discrete-time impulse (Figure 2.2-(a))

$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0. \end{cases}$$

- The discrete-time unit step (Figure 2.2-(b))

$$u[n] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0. \end{cases}$$

  which can be represented as

$$u[n] = \sum_{k=-\infty}^{n} \delta[k] = \sum_{k=0}^{\infty} \delta[n-k].$$

- The discrete-time exponential decay (Figure 2.2-(c))

$$x[n] = a^n u[n], \quad a \in \mathbb{C}, |a| < 1.$$

- The discrete-time sinusoidal oscillations (Figure 2.2-(d))

$$x[n] = \sin(\omega_0 n + \phi)$$

$$x[n] = \cos(\omega_0 n + \phi).$$

- The discrete-time complex exponential (Figure 2.1-(b))

$$x[n] = e^{j(\omega_0 n + \phi)}.$$

Example 2.4 *Combining basic sequences*

$$y[n] = \begin{cases} A\alpha^n & n \geq 0 \\ 0 & n < 0 \end{cases}$$

takes $y[n] = x[n]u[n]$, where $x[n] = A\alpha^n$ and $u[n]$ is discrete-time unit step. (Figure 2.2-(d))

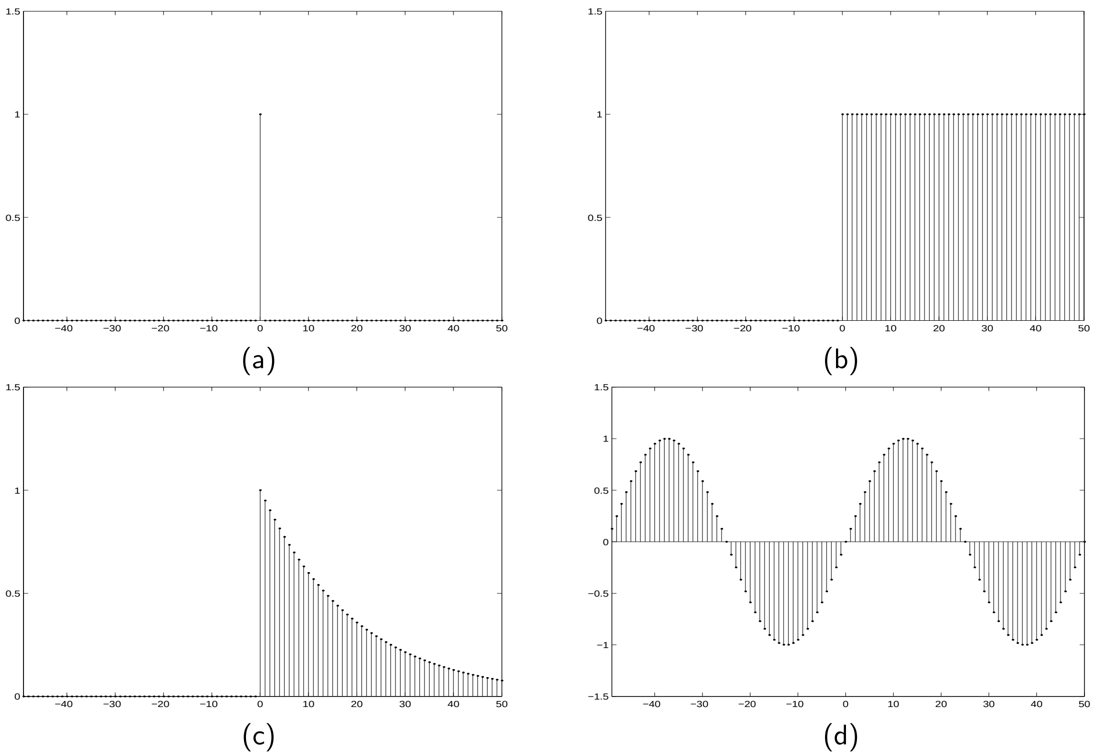


Figure 2.2: Basic signals.

Definition 2.1 *A sequence $\{x[n]\}$ is said to have period N if*

$$x[n] = x[n+N], \qquad \text{for all } n \in \mathbb{Z}.$$

For a complex exponential $x[n] = e^{j\omega_0 n}$ if it is to have a period of N, we need

$$e^{j\omega_0 n} = e^{j\omega_0 (n+N)}, \quad \forall n,$$

i.e.,

$$\omega_0 N = 2\pi r, r \in \mathbb{Z}$$

or $\omega_0 = \frac{2\pi r}{N}$ which means that since $N \in \mathbb{Z}$, we need ω_0 to be rational for a complex exponential to be periodic in a discrete sense.

Example 2.5 *Let* $x_1[n] = \cos(\frac{\pi n}{4})$*. Since for all* n*,* $x_1[n] = x_1[n+8]$*, we have a period of* $N = 8$ *for the sequence.*
However, $x_2[n] = \cos(\frac{3\pi n}{8})$ *gives* $x_2[n+8] = \cos[\frac{3\pi n}{8} + 3\pi] = -x_2[n]$ *and hence does* not *have a period of* 8*. In fact it has a period of* $N = 16$*. Therefore, even though we think of a higher "'frequency"' for* $x_2[n]$ *in comparison to* $x_1[n]$*, it has a larger period. This is due to the limitations imposed by integer time index* n *for discrete-time signals.*

2.3.2 Digital Frequency

With respect to the last two examples a note on the concept of "frequency" is in order. In the analog world the usual unit of measure for frequency is the Hertz, which has a physical dimension of s^{-1}. In the discrete-time world, where the index n represents dimensionless time, "digital" frequency is expressed in radians which is itself an dimensionless quantity[1]. The best way to appreciate this is to consider an algorithm to generate successive samples of a discrete time sinusoid at a digital frequency ω_0:

$\omega \leftarrow 0$;	*initialization*
$\phi \leftarrow$ initial phase value;	
repeat	
$x \leftarrow \sin(\omega + \phi)$;	*compute next value*
$\omega \leftarrow \omega + \omega_0$;	*update phase*
until done	

At each iteration[2] , the argument of the trigonometric function is incremented by ω_0 and a new output sample is produced. With this in mind, it is easy to see that the highest frequency manageable by a discrete-time system is 2π; for any frequency larger than this, the inner 2π-periodicity of the trigonometric functions "maps back" the output values to a frequency between 0 and 2π. In formulas:

$$\sin(n(\omega + 2k\pi) + \phi) = \sin(n\omega + \phi) \tag{2.14}$$

[1]An angle measure in radians is dimensionless since it is defined in terms of the ratio of two lengths, the radius and the arc subtended by the measured angle on an arbitrary circle.

[2]Here is the same algorithm written as a C function, if it helps:

```
extern double omega0;
extern double phi;
static double omega = 0;
double GetNextValue()
{
    omega += omega0;
    return sin(omega + phi);
}
```

for all values of $k \in \mathbb{Z}$. This 2π-equivalence of digital frequencies is a pervasive concept in digital signal processing and it has many important consequences which we will study in detail throughout the course.

2.3.3 Elementary Operators

Elementary operations on sequences are defined as follows:

- **Shift.** The shifted version of the sequence $x[n]$ by an integer k is

  $$y[n] = x[n-k].$$

  If k is positive, the signal has been *delayed*; if k is negative, it has been *advanced.*

- **Scaling.** The scaled version of the sequence $x[n]$ by a factor $\alpha \in \mathbb{C}$ is

  $$y[n] = \alpha x[n].$$

- **Sum.** The sum of two sequences $x[n]$ and $w[n]$ is their term-by-term sum,

  $$y[n] = x[n] + w[n].$$

- **Product.** The product of two sequences $x[n]$ and $w[n]$ is their term-by-term product,

  $$y[n] = x[n]w[n].$$

- **Moving average**

  $$y[n] = \frac{1}{M_1 + M_2 + 1} \sum_{k=-M_1}^{M_2} X[n-k].$$

- **Integration.** The discrete-time equivalent of integration is the running sum:

  $$y[n] = \sum_{k=-\infty}^{n} x[k].$$

- **Differentiation.** A discrete-time approximation to differentiation is the first-order difference[3]:

$$y[n] = x[n] - x[n-1].$$

With respect to section 2.3.1, note how the unit step can be obtained by applying the integration operator to the discrete-time impulse; conversely, the impulse can be obtained by applying the differentiation operator to the unit step.

Definition 2.2 *Let $\mathcal{L}(\cdot)$ be an operation. The operation is linear if*

$$\mathcal{L}(\alpha x_1[n] + \beta x_2[n]) = \alpha\mathcal{L}(x_1[n]) + \beta\mathcal{L}(x_1[n]), \tag{2.15}$$

for any sequences $x_1[n]$, $x_2[n]$ and scalars α and β.

All the operations defined above are linear operations.

Example 2.6 *The shift operation is linear. Let $\mathcal{L}(x[n]) = x[n-k]$. Now, if $y_1[n] = x_1[n-k]$ and $y_2[n] = x_2[n-k]$, then $y[n] = \mathcal{L}(\alpha x_1[n] + \beta x_2[n]) = \alpha x_1[n-k] + \beta x_2[n-k])$ which can be expressed as $y[n] = \alpha y_1[n] + \beta y_2[n]$.*

Example 2.7 *Suppose $y[n] = x^2[n]$. If $y_1[n] = x_1[n]^2$, and $y_2[n] = x_2[n]^2$, then $y[n] = \mathcal{L}(x_1[n]+x_2[n]) = (x_1[n]+x_2[n])^2 = x_1[n]^2+x_2[n]^2+2x_1[n]x_2[n]$ which can not be expressed as $y_1[n] + y_2[n]$. So this operation is not linear.*

2.3.4 The Reproducing Formula

The signal reproducing formula is a simple application of the basic signal and signal properties we have just seen and it states that:

$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]. \tag{2.16}$$

In words, any signal can be expressed as a linear combination of suitably weighed shifted impulses. In this case, the weights are simply the signal values. While apparently self-evident, this formula will reappear in a multitude of reincarnations in the rest of the course. You are encouraged to spend a few minutes thinking about how it actually works.

[3]We will see later, when we study filters, that the "correct" approximation to differentiation is given by a filter $H(e^{j\omega}) = j\omega$. For most application, however, the first-order difference will suffice.

2.3.5 Energy and power

We define the *energy* of a discrete-time signal as

$$E_x = ||x||_2^2 = \sum_{n=-\infty}^{\infty} |x[n]|^2 \tag{2.17}$$

(where the squared-norm notation will be clearer after the next chapter.) This definition is consistent with the idea that, if the values of the sequence represent a time-varying voltage, the above sum would express the total energy (in joules) dissipated over a 1Ω-resistor. Obviously, the energy is finite only if the above sum converges, i.e., if the sequence $x[n]$ is *square-summable*. A signal with this property is sometimes referred to as a *finite-energy signal*. For a simple example of the converse, note that a periodic signal which is not identically zero is *not* square-summable.

We define the *power* of a signal as the usual ratio of energy over time, taking the limit over the number of samples considered:

$$P_x = \lim_{N\to\infty} \frac{1}{2N} \sum_{-N}^{N-1} |x[n]|^2. \tag{2.18}$$

Clearly, signals whose energy is finite have zero total power (i.e. their energy dilutes to zero over an infinite time duration). Note however that many signals whose energy is infinite do have finite power and, in particular, so do periodic signals (such as sinusoids and combinations thereof). Due to their periodic nature, however, the above limit is undetermined; we therefore *define* their power to be simply the *average energy over a period*. Assuming that the period is N samples, we have:

$$P_x = \frac{1}{N} \sum_{n=0}^{N-1} |x[n]|^2. \tag{2.19}$$

2.4 Classes of Discrete-Time Signals

The examples of discrete-time signals in (2.12) and (2.13) are two-sided, infinite sequences. Of course, in the practice of signal processing, it is impossible to deal with infinite sequences: for a processing algorithm to compute in a finite amount of time and use a finite amount of storage, the input data must be of finite length; even for algorithms that operate on the fly, i.e. algorithms that produce an output sample for each new input sample, an implicit finiteness is imposed by the necessarily limited life span of the processing device or, in the extreme limit, of the supervising engineer. This limitation was eminently clear in our attempt to plot the sequences in Figures 2.1-(a), (b): we were content with showing a representative portion of the sequences, and we relied on their analytical description to describe their behavior outside of the observation window we chose for the plot. When the discrete-time signal admits no closed-form representation, as is basically always the case when dealing with real-world signals, its finite time support arises naturally because of the finite time we spend measuring said signal: every piece of music has a beginning and an end, and so does any phone conversation. In the case of the sequence representing the Dow Jones index, for instance, we sort of cheated since the index does not even exist for years before 1884, and its value tomorrow is certainly not known – so that's not really a sequence. But, more importantly and more often, the finiteness of a discrete-time signal is arbitrarily imposed since we are interested in concentrating our processing efforts on a small portion of an otherwise much longer signal; in a speech recognition system, for instance, the practice is to cut up a speech signal into small segments and try to identify the phonemes associated to each one of them[4]. A special case is that of periodic signals; even though these are bona-fide infinite sequences, it is clear that all information about them is contained in just one period. By describing graphically or otherwise this period, we are in fact providing a complete description of the sequence. In order to capture these particular cases, we will divide signals into three main families.

2.4.1 Finite-Length Signals

As we just mentioned, finite-length discrete-time signals of length N are just a collection of N complex values. To introduce a point that will reappear throughout these notes, a finite-length signal of length N is entirely equivalent to a vector in $\mathbb{C}^N$. This equivalence is of immense import since all the tools of linear algebra become readily available for describing and manipulating finite-length signals. We can represent an N-point finite-length signal using the standard vector notation

$$\mathbf{x} = [x_0 \quad x_1 \quad \ldots \quad x_{N-1}]^T;$$

note the transpose operator, which declares $\mathbf{x}$ as a *column* vector; this is the customary practice in the case of complex-valued vectors. Alternatively, we can (and often will) use a notation that mimics that which we use for proper sequences:

$$x[n], \quad n = 0, \ldots, N-1;$$

here we *must* remember that, although we use the notation $x[n]$, $x[n]$ is *not defined* for values outside its support, i.e. for $n < 0$ or for $n \geq N$. Note that we can always obtain a finite-length signal from an infinite sequence by simply dropping the sequence values

[4]Note that, in the end, phonemes are pasted together into words and words into sentences; therefore, for a complete speech recognition system, long-range dependencies become important again.

outside the indices of interest. Vector and sequence notations are equivalent and will be used interchangeably according to convenience; in general, the vector notation is useful when we want to stress the algorithmic or geometric nature of certain signal processing operations. The sequence notation is useful in stressing the algebraic structure of signal processing.

Finite-length signals are extremely convenient entities: their energy is always finite as long as the elements in the signals are finite; as a consequence, no stability issues arise in processing. From the computational point of view, they are not only a necessity but often the cornerstone of very efficient algorithmic design (as we will see for instance in the case of the FFT); one could say that all "practical" signal processing lives in $\mathbb{C}^N$. It would be extremely awkward, however, to develop the whole theory of signal processing only in terms of finite-length signals; the asymptotic behavior of algorithms and transformations for infinite sequences is extremely valuable as well since a stability result proven for a general sequence will hold for all finite-length signals too. Furthermore the notational flexibility which infinite sequences derive from their function-like definition is extremely practical from the point of view of the notation. We can immediately recognize and understand the expression $x[n-k]$ as a k-point shift of a sequence $x[n]$; but, in the case of finite-support signals, how are we to define such a shift? We would have to explicitly take into account the finiteness of the signal and the associated "border effects", i.e. the behavior of operations at the edges of the signal. This is why, in most derivations which involve finite-length signal, these signals will be *embedded* into a proper sequences, as we will see momentarily.

2.4.2 Infinite, Aperiodic Signals

The most general type of discrete-time signal is represented by a generic infinite complex sequence. Although, as we said, they lie beyond our processing and storage capabilities, they are invaluably useful as a generalization in the limit. As such, they must be handled with some care when it comes to their properties. We will see shortly that two of the most important properties of infinite sequences concern their summability: this can take the form of either *absolute summability* (stronger condition) or *square summability* (weaker condition corresponding to finite energy).

2.4.3 Periodic Signals and Periodic Extensions

A periodic sequence with period N is one for which

$$\tilde{x}[n] = \tilde{x}[n+kN], \quad k \in \mathbb{Z}. \tag{2.20}$$

The tilde notation $\tilde{x}[n]$ will be used whenever we need to explicitly stress a periodic behavior. Clearly a N-periodic sequence is completely defined by its N values over a

$$\begin{array}{rcll} \tilde{x}[n] & = & \ldots x_{N-2}, x_{N-1}, & \overbrace{x_0, x_1, x_2, \ldots, x_{N-2}, x_{N-1},}^{\mathbf{x}} \quad x_0, x_1, \ldots \\ & & & \updownarrow \rightarrow n = 0 \\ \tilde{x}[n-1] & = & \ldots x_{N-3}, x_{N-2}, & \underbrace{x_{N-1}, x_0, x_1, x_2, \ldots, x_{N-2},}_{\mathbf{x}'} \quad x_{N-1}, x_0, x_1, \ldots \end{array}$$

Figure 2.3: Equivalence between a right shift by one of a periodized signal and the circular shift of the original signal. $\mathbf{x}$ and $\mathbf{x}'$ are the length-N original signal and its right circular shift by one, respectively.

period; that is, a periodic sequence "carries no more information" that a finite-length signal of length N. In this sense, periodic sequences are a bridge between finite-length signals and infinite sequences. We are therefore ready to discover the first way to embed a finite-length signal $x[n]$, $n = 0, \ldots, N-1$ into a sequence which is by taking its periodized version:

$$\tilde{x}[n] = x[n \mod N], \quad n \in \mathbb{Z}; \tag{2.21}$$

this is called the *periodic extension* of the finite length signal $x[n]$. This type of extension is the "natural" one in many contexts, for reasons which will be apparent later when we study the frequency-domain representation of discrete-time signals. Note that now an arbitrary shift of the periodic sequence correspond to the periodization of a *circular shift* of the original finite-length signal. A circular shift by $k \in \mathbb{Z}$ is easily visualized by imagining a shift register; if we are shifting towards the right ($k > 0$), the values which pop out of the rightmost end of the shift register are pushed back in at the other end[5]. The relationship between circular shift of a finite-length signal and the linear shift of its periodic extension is depicted in Figure 2.3. Finally, the energy of a periodic extension becomes infinite, while its power is simply the energy of the finite-length original signal scaled by $1/N$.

Example 2.8 *What is the period of the following sequence?*

$$\tilde{x}[n] = 2 + \sin(\frac{2\pi}{3}n) + \cos(\frac{4\pi}{5}n)$$

[5] For example, if $\mathbf{x} = [1\,2\,3\,4\,5]$, a right circular shift by 2 yields $\mathbf{x} = [4\,5\,1\,2\,3]$.

To answer this, we need to look for N such that for all n, $\tilde{x}[n+N] = \tilde{x}[n]$. This means that we need to find N for which

$$2 + \sin\left(\frac{2\pi}{3}(n+N)\right) + \cos\left(\frac{4\pi}{5}(n+N)\right) = 2 + \sin\left(\frac{2\pi}{3}n\right) + \cos\left(\frac{4\pi}{5}n\right). \quad (2.22)$$

We have that $\sin\left(\frac{2\pi}{3}(n+N_1)\right) = \sin\left(\frac{2\pi}{3}n\right)$ *for* $N_1 = 3$ *and* $\sin\left(\frac{4\pi}{5}(n+N_2)\right) = \sin\left(\frac{4\pi}{5}n\right)$ *for* $N_2 = 5$. *If we take N equal to the least common multiple of N_1 and N_2 we satisfy (2.22). Hence $N = 15$.*

2.4.4 Finite-Support Signals

An infinite discrete-time sequence $\bar{x}[n]$ is said to have *finite support* if its values are zero for all indices outside of an interval; that is, there exist N and $M \in \mathbb{Z}$ such that

$$\bar{x}[n] = 0 \ \text{ for } n < M \text{ and } n > M + N - 1.$$

Note that, although $\bar{x}[n]$ is an infinite sequence, the knowledge of M and of the N nonzero values of the sequence completely specify the entire signal. This suggest another approach to embedding a finite-length signal $x[n]$, $n = 0, \ldots, N-1$ into a sequence, i.e.

$$\bar{x}[n] = \begin{cases} x[n] & \text{if } 0 \le n < N-1 \\ 0 & \text{otherwise} \end{cases} \qquad n \in \mathbb{Z} \quad (2.23)$$

where we have chosen $M = 0$ (but any other choice of M would do as well). Note that here, in contrast to the the periodic extension of $x[n]$, we are actually adding arbitrary information in the form of the the zero values outside of the support interval. This is not without consequences, as we will see in the following chapters. In general we will use the bar notation $\bar{x}[n]$ for sequences defined as the finite support extension of a finite-length signal. Note that now the shift of the finite-support extension gives rise to a zero-padded shift of the signal locations between M and $M + N - 1$; the dynamics of the shift are shown in Figure 2.4.

2.5 Summary

The main points introduced by this chapter have been:

- The formal definition for the concept of discrete-time signal.
- A gallery of prototypical signals and fundamental signal operators.
- A discussion of digital frequency and its 2π-periodic nature.

$$
\begin{array}{rcll}
\bar{x}[n] & = & \ldots, 0, 0, & \overbrace{x_0, x_1, x_2, \ldots, x_{N-2}, x_{N-1},}^{\mathbf{x}} \quad 0, 0, 0, 0, \ldots \\
 & & & \updownarrow \rightarrow n = 0 \\
\bar{x}[n-1] & = & \ldots 0, 0, & \underbrace{0, x_0, x_1, x_2, \ldots, x_{N-3}, x_{N-2},}_{\mathbf{x}'} \quad x_{N-1}, 0, 0, \ldots
\end{array}
$$

Figure 2.4: Relationship between the right shift by one of a finite-support extension and the zero padded shift of the original signal. $\mathbf{x}$ and $\mathbf{x}'$ are the length-N original signal and its zero-padded shift by one, respectively.

- The definitions of energy and power.
- A classification of signals into finite-length, infinite-length, periodic and finite-support signals, with their respective properties.

The fundamental discrete-time signal types, along with their properties, are summarized in Table 2.1.

Signal Type	Notation	Energy	Power
Finite-Length	$x[n], \quad n = 0, 1, \ldots, N-1$ $\mathbf{x}, \quad \mathbf{x} \in \mathbb{C}^N$	$\sum_{n=0}^{N-1} \lvert x[n] \rvert^2$	undef.
Infinite-Length	$x[n], \quad n \in \mathbb{Z}$	eq. (2.17)	eq. (2.18)
N-Periodic	$\tilde{x}[n], \quad n \in \mathbb{Z}$, $\tilde{x}[n] = \tilde{x}[n+kN]$	∞	eq. (2.19)
Finite-Support	$\bar{x}[n], \quad n \in \mathbb{Z}$ $\bar{x}[n] \neq 0$ for $M \leq n \leq M+N-1$	$\sum_{n=M}^{M+N-1} \lvert x[n] \rvert^2$	0

Table 2.1: Basic discrete-time signal types.

Chapter 3

Representation of Discrete-Time Sequences (DFS, DFT)

Fourier theory has a long history, from J. Fourier's early work on the transmission of heat to recent results on non-harmonic Fourier series and related topics. Fourier theory is a branch of harmonic analysis, and in that sense, a topic in pure and applied mathematics. At the same time, because of its usefulness in practical applications, Fourier analysis is a key tool in several engineering branches, and in signal processing in particular.

Why is Fourier analysis so important? To understand this, it is useful to take a little philosophical detour. Interesting signals are time-varying quantities: you can imagine for instance the voltage level at the output of a microphone or the measured level of the tide at a particular location; in all cases, the variation of a signal over time implies that a transfer of energy is happening someplace, and this is what ultimately we want to study. Now, a time-varying value which only *increases* over time is not only a physical impossibility but a recipe for disaster for whatever system is supposed to deal with it: fuses will blow, wires will melt and so on. Oscillations, on the other hand, are nature's and man's way to keep things in motion without trespassing all physical bounds; from Maxwell's wave equation to the mechanics of the vocal cords, from the motion of an engine to the ebb and flow of tide, oscillatory behavior is the recurring theme. Sinusoidal oscillations, as it stands, are the purest form of such a constrained motion and, in a nutshell, Fourier's immense contribution was to show that (at least mathematically) one could express any given phenomenon as the combined output of a number of sinusoidal "generators".

Sinusoids have another remarkable property which justifies their ubiquitous presence. Indeed, *any linear transformation of a sinusoid is a sinusoid at the same frequency*: we express this by saying that sinusoidal oscillations are eigenfunctions of linear systems. This is a formidable tool for the analysis and design of signal processing structures, as we will see in much detail in the context of linear systems.

The purpose of the present chapter is to review key results on Fourier series and Fourier transforms in the context of discrete-time signal processing. As it turns out, and as we hinted at in the previous chapter, the Fourier transform of a signal is a change of basis in its appropriate Hilbert space. While this notion constitutes an extremely useful unifying framework, we will also point out the peculiarities of its specialization within the different classes of signals. In particular, for finite-length signals we will highlight the eminently algebraic nature of the transform, which will lead to efficient computational procedures; for infinite sequences, we will analyze some of its interesting mathematical subtleties.

A periodic sequence has the property that

$$\tilde{x}[n+N] = \tilde{x}[n] \quad \forall n,$$

and N is a period. Therefore a periodic sequence is completely specified by N values. Without loss of generality, we can take these N values from $x[0], \ldots, x[N-1]$.
One representation of this sequence is in the "time domain", but one can imagine completely equivalent representations in other forms (bases). We will consider other representations of such signals in this class such as

- Fourier Transform (through complex exponentials)
- Z-Transform
- Time-frequency representation (like wavelets)

A way to think of these different representations is that each of them has some canonical properties suited to particular scenarios. The most important representation, historically as well as in applications, is the so-called Fourier representation.

3.1 Preliminaries

3.1.1 Terminology

The Fourier transform of a signal is an alternative representation of the data in the signal. While a signal lives in the *time domain*[1], its Fourier representation lives in the *frequency domain*. We can move back and forth at will from one domain to the other using the direct and inverse Fourier operators, since these operators are invertible.

In this chapter we will study two types of Fourier transforms which apply to two of the main classes of signals we have seen so far:

[1] *Discrete*-time, of course.

- the Discrete Fourier Transform (DFT), which maps length-N signals into a set of N discrete frequency components
- the Discrete Fourier Series (DFS), which maps N-periodic sequences into a set of N discrete frequency components.

The frequency representation of a signal (given by a set of coefficients in the case of the DFT and DFS) is called the *spectrum*.

3.1.2 Complex Oscillations? Negative Frequencies?

In the introduction, we hinted at the fact that Fourier analysis allows us to decompose a physical phenomenon into oscillatory components. It may seem odd, however, that we chose to use complex oscillation for the analysis of real-world signals. It may seem even more odd that these oscillations can have a negative frequency and that, as we will soon see in the context of the DTFT, the spectrum extends over to the negative axis.

The starting point in answering these legitimate questions is to recall that the use of complex exponentials is essentially a matter of convenience. One could develop a complete theory of frequency analysis for real signals using only the basic trigonometric functions. You may actually have seen this in the context of Fourier series; yet the notational overhead is undoubtedly heavy since it involves two separate sets of coefficients for the sine and cosine basis functions, plus a distinct term for the zero-order coefficient. The use of complex exponentials elegantly unifies these separate series into a single complex-valued sequence. Yet, one may ask again, what does it mean for the spectrum of a musical sound to be complex? Simply put, the complex nature of the spectrum is a compact way of representing two concurrent pieces of information which uniquely define each spectral component: its *frequency* and its *phase*. This couple of values is a two-element vector in $\mathbb{R}^2$ but, since $\mathbb{R}^2$ is isomorphic to $\mathbb{C}$, we use complex numbers for their mathematical convenience.

What about negative frequencies, then? Again, first of all consider a basic complex exponential sequence such as $x[n] = e^{j\omega n}$. We can visualize its evolution over discrete-time as a series of points on the unit circle in the complex plane. At each step, the angle increases by ω, defining a counterclockwise circular motion. It is easy to see that a complex exponential sequence of frequency $-\omega$ is just the same series of points which moves *clockwise* instead; this is illustrated in detail in Figure 3.1. We will show that if we decompose a *real* signal into complex exponentials, for any given frequency value, the phases of the positive and negative components are always opposite in sign; as the two oscillations move in opposite directions along the unit circle, their complex part will always cancel out exactly, thus returning a purely real signal[2].

[2]To anticipate a question which may appear later, the fact that modulation "makes negative frequencies

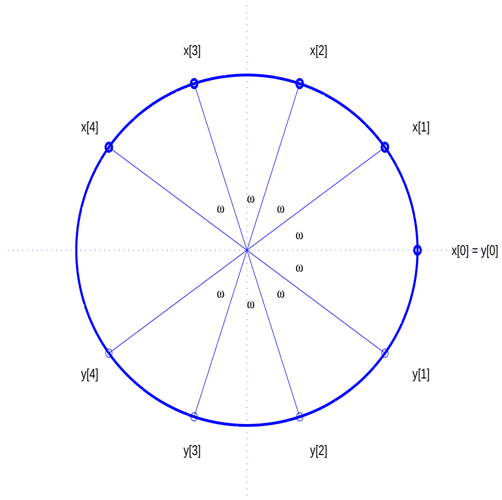


Figure 3.1: Complex exponentials as a series of points on the unit circle; $x[n] = e^{j\omega n}$ and $y[n] = e^{-j\omega n}$ for $\omega = \pi/5$.

The final step in developing a comfortable feeling for complex oscillations comes from the realization that, in the *synthesis* of discrete-time signals (and especially in the case of communication systems) it is actually more convenient to work with complex-valued signals themselves. While in the end the transmitted signal of a device like an ADSL box is a real signal, the internal representation of the underlying sequences is complex, and therefore complex oscillations become a necessity.

3.1.3 Complex Exponentials

The basic ingredient of all Fourier representations (transforms) is the complex exponential which we have seen before

$$x[n] = Ae^{j\omega n} = A\cos(\omega n) + jA\sin(\omega n). \tag{3.1}$$

appear in the positive spectrum" is really a consequence of the very mundane formula:

$$\cos\alpha\cos\beta = \frac{1}{2}[\cos(\alpha+\beta) + \cos(\alpha-\beta)].$$

A natural question to ask is why we would use a complex oscillating signal, when most signals we encounter are real. The simplest answer is in terms of notational convenience. It is possible to develop representation using only real sinusoids, but in order to account for the phase as well as frequency, it becomes more cumbersome.
A real sinusoid can always be represented using complex sinusoids as follows:

$$\sin(\omega n) = \frac{e^{j\omega n} - e^{-j\omega n}}{2j}$$

$$\cos(\omega n) = \frac{e^{j\omega n} + e^{-j\omega n}}{2}$$

Moreover, a representation using complex sinusoids is inherently more general.

3.2 Representation of Periodic Sequences: The Discrete-Time Fourier Series (DFS)

Since we want to represent periodic signals of period N using complex exponentials, we need to find a set of complex exponentials which contain a whole number of periods over N. Let us examine

$$w_k[n] = e^{j\omega_k n}.$$

Since we want $w_k[n]$ to contain a whole number of periods over N samples, we need to have ω_k such that

$$w_k[0] = w_k[N],$$

i.e.

$$w_k[N] = 1 = e^{j\omega_k N}.$$

Clearly this equation has N possible solutions,

$$\omega_k = \frac{2\pi}{N}k, \qquad k = 0, \cdots, N-1.$$

Therefore, if we define

$$W_N = e^{-j\frac{2\pi}{N}}$$

then the family of sequences with the property of having complete periods over N samples, is

$$w_k[n] = W_N^{-nk}, \qquad n = 0, \ldots, N-1, \qquad k = 0, \ldots, N-1.$$

That is we have defined a family of N sequences which have a complete period over N samples.
Now suppose that we want to represent the periodic signals using the family of sequences $\{w_k[n]\}_{k=0}^{N-1}$. Given a periodic sequence $\tilde{x}[n]$, we want to solve for $\tilde{X}_k, k = 0, \cdots, N-1$, in

$$\tilde{x}[n] = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}_k w_k[n] = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}_k e^{j\frac{2\pi}{N}nk}.$$

The factor $1/N$ has been added for notational convenience further along in the analysis.

We can write this as

$$\tilde{x}[0] = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}_k \tag{3.2}$$

$$\tilde{x}[1] = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}_k e^{j\frac{2\pi}{N}k} \tag{3.3}$$

$$\vdots$$

$$\tilde{x}[N-1] = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}_k e^{j\frac{2\pi(N-1)}{N}k} \tag{3.4}$$

We want to express the above equations in matrix form. Therefore, we introduce

$$\mathbf{w}_k^{(k)} = \begin{bmatrix} 1 \; W_N^{-k} \; W_N^{-2k} \; \ldots \; W_N^{-(N-1)k} \end{bmatrix}^T \tag{3.5}$$

and the matrix Λ defined as

$$\Lambda^H = \begin{bmatrix} \left(\mathbf{w}^{(0)}\right)^T \\ \left(\mathbf{w}^{(1)}\right)^T \\ \vdots \\ \left(\mathbf{w}^{(N-1)}\right)^T \end{bmatrix}.$$

In matrix form, we have

$$\begin{bmatrix} \tilde{x}[0] \\ \tilde{x}[1] \\ \vdots \\ \tilde{x}[N-1] \end{bmatrix} = \frac{1}{N} \underbrace{\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & e^{j\frac{2\pi}{N}1} & \cdots & e^{j\frac{2\pi}{N}(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{j\frac{2\pi(N-1)}{N}} & \cdots & e^{j\frac{2\pi(N-1)^2}{N}} \end{bmatrix}}_{\Lambda^H} \begin{bmatrix} \tilde{X}[0] \\ \tilde{X}[1] \\ \vdots \\ \tilde{X}[N-1] \end{bmatrix}. \tag{3.6}$$

We have N unknowns and N equations, and hence if the equations are linearly independent we can expect a solution.
Now, note that

$$\left(\mathbf{w}^{(k)}\right)^H \cdot \mathbf{w}^{(m)} = \begin{bmatrix}1 \; e^{-j\frac{2\pi}{N}k} \; \ldots \; e^{-j\frac{2\pi}{N}k(N-1)}\end{bmatrix} \begin{bmatrix} 1 \\ e^{j\frac{2\pi}{N}m} \\ \vdots \\ e^{j\frac{2\pi}{N}m(N-1)} \end{bmatrix}$$
$$= \sum_{i=0}^{N-1} e^{j\frac{2\pi}{N}(m-k)i}$$
$$= \begin{cases} N & k = m \\ 0 & k \neq m \end{cases}. \tag{3.7}$$

This follows from the fact that for $k \neq m$

$$\sum_{i=0}^{N-1} e^{j\frac{2\pi}{N}(m-k)i} = \frac{1 - e^{j\frac{2\pi}{N}(m-k)N}}{1 - e^{j\frac{2\pi}{N}(m-k)}} = 0.$$

Therefore, the rows of the matrix Λ in (3.6) are orthogonal (Not orthonormal, but that can be fixed by normalizing by $\sqrt{N}$.)

Hence (3.7) shows that

$$\langle \mathbf{w}^{(k)}, \mathbf{w}^{(m)} \rangle = \begin{cases} N & k = m \\ 0 & k \neq m \end{cases}$$

making $\{\mathbf{w}^{(0)}, \cdots, \mathbf{w}^{(N-1)}\}$ an orthognal set of vectors in $\mathbb{C}^N$.

Theorem 3.1 $\Lambda \; \Lambda^H = NI$

Proof

$$\begin{bmatrix} \mathbf{w}^{(0)} \\ \vdots \\ \mathbf{w}^{(N-1)} \end{bmatrix} \begin{bmatrix} \left(\mathbf{w}^{(0)}\right)^H & \cdots & \left(\mathbf{w}^{(N-1)}\right)^H \end{bmatrix} = [a_{p,q}]_{N\times N} = NI,$$

where

$$a_{p,q} = \langle \mathbf{w}^{(p)}, \mathbf{w}^{(q)} \rangle.$$

□

Hence, looking at equation (3.6), we see that

$$\begin{bmatrix} \tilde{X}[0] \\ \tilde{X}[1] \\ \vdots \\ \tilde{X}[N-1] \end{bmatrix} = \Lambda \begin{bmatrix} \tilde{x}[0] \\ \tilde{x}[1] \\ \vdots \\ \tilde{x}[N-1] \end{bmatrix} \tag{3.8}$$

$$= \begin{bmatrix} 1 & \cdots & 1 \\ e^{-j\frac{2\pi}{N}0} & \cdots & e^{-j\frac{2\pi}{N}(N-1)} \\ \vdots & \ddots & \vdots \\ e^{-j\frac{2\pi(N-1)}{N}0} & \cdots & e^{-j\frac{2\pi(N-1)^2}{N}} \end{bmatrix} \begin{bmatrix} \tilde{x}[0] \\ \tilde{x}[1] \\ \vdots \\ \tilde{x}[N-1] \end{bmatrix}. \tag{3.9}$$

Therefore,

$$\tilde{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}kn}. \tag{3.10}$$

Hence, we have the following representation of the periodic sequence $\{\tilde{x}[n]\}$ through $\{\tilde{X}[k]\}_{k=0}^{N-1}$ and the class of periodic exponentials $\{w_k[n]\}_{k=0}^{N-1}$ as

$$\tilde{x}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] w_k[n], \tag{3.11}$$

where

$$\tilde{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n] w_k[n]^H. \tag{3.12}$$

Thus $\{\tilde{X}[k]\}_{k=0}^{N-1}$ can be thought of as the weights on the complex exponentials to represent a periodic sequence.

Notes:

1. *Any* periodic sequence can therefore be represented as a weighted sum of complex exponentials. This is an akin to decomposing a periodic sequence into elementary periodic functions.

2. Since $\langle \mathbf{w}^{(k)}, \mathbf{w}^{(m)} \rangle = \delta_{k-m} N$, the vectors $\mathbf{w}^{(k)}, \quad k = 0, \cdots N-1$ are orthogonal and form a *basis* of $\mathbb{C}^N$. The representation for periodic sequences is a consequence of this property. We implicitly used this in in inverting Λ in 3.6 to find the weights $\{\tilde{X}[k]\}_{k=0}^{N-1}$.

3. Extending this thought process, one can envisage expanding $\{\tilde{x}[n]\}$ in *any* basis of $\mathbb{C}^n$, which is also periodically extended.

In summary, we have the following Discrete-time Fourier Series (DFS) representation of discrete-time periodic sequences. The *synthesis formula*,

$$\tilde{x}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] e^{j\frac{2\pi}{N}kn}. \tag{3.13}$$

The *analysis formula*:

$$\tilde{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}kn}. \tag{3.14}$$

This set of equations describe how to synthesize $\{\tilde{x}[n]\}$ given the Discrete-time Fourier Series (DFS) coefficients $\{\tilde{X}[k]\}$ and how to analyze $\{\tilde{x}[n]\}$ to produce DFS coefficients $\{\tilde{X}[k]\}$.

Example 3.1 *Suppose* $x[n] = \sin(\frac{2\pi}{N}n)$
This has a period of N. *Since*

$$\begin{aligned} \sin(\frac{2\pi}{N}n) &= \frac{1}{2j}e^{j\frac{2\pi}{N}n} - \frac{1}{2j}e^{-j\frac{2\pi}{N}n} \\ &= \frac{1}{2j}e^{j\frac{2\pi}{N}n} - \frac{1}{2j}e^{j\frac{2\pi}{N}(N-1)n} \end{aligned}$$

Hence we see that

$$\tilde{X}[0] = \frac{1}{2j}, \quad \tilde{X}[N-1] = -\frac{1}{2j}, \quad \tilde{X}[k] = 0, k = 1, \cdots N-2.$$

In general the synthesis and analysis equations can be written in matrix form as:

$$\tilde{\mathbf{x}} = \frac{1}{N}\Lambda^H \tilde{\mathbf{X}} \tag{3.15}$$

$$\tilde{\mathbf{X}} = \Lambda \tilde{\mathbf{x}} \tag{3.16}$$

where

$$\tilde{\mathbf{x}} = [\tilde{x}[0], \cdots, \tilde{x}[N-1]]^T$$

$$\tilde{\mathbf{X}} = [\tilde{X}[0], \cdots, \tilde{X}[N-1]]^T$$

Note that we can find the energy in one period of the sequence in terms of its Fourier series coefficients as

$$||\tilde{\mathbf{x}}||_2^2 = \sum_{n=0}^{N-1} |\tilde{x}[n]|^2 = \frac{1}{N^2}\tilde{\mathbf{X}}^H \Lambda \cdot \Lambda^H \tilde{\mathbf{X}} = \frac{1}{N}\sum_{k=0}^{N-1} |\tilde{X}[n]|^2 = \frac{1}{N}||\tilde{\mathbf{X}}||_2^2.$$

Therefore, the energy in one period of the periodic signal is N-times the energy in the Fourier series coefficients.

$$N||\tilde{\mathbf{x}}||_2^2 = ||\tilde{\mathbf{X}}||_2^2$$

This is called *Parseval's relationship.*

3.2.1 Interpretation of the Fourier series

We have expressed the periodic sequence $\tilde{x}[n]$ as a weighted sum of N sinusoids,

$$\tilde{x}[n] = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}[k] e^{j\frac{2\pi}{N}kn} = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}[k] w_k[n].$$

The magnitude and the phase weighting of each sinusoid $w_k[n] = e^{j\frac{2\pi}{N}kn}$ is given by $\tilde{X}[k]$, which is the Fourier series coefficient. Therefore $\tilde{X}[k]$ shows "how much" of an oscillatory behavior at freqency $\frac{2\pi}{N}k$ is contained in the periodic signal $\tilde{x}[n]$. The coefficients $\{\tilde{X}[k]\}$ can therefore be interpreted as the *spectrum* of the signal. Parseval's relationship shows that up to a scaling factor of N, the energy contained in the spectrum of the signal is the same as the energy in the signal itself, i.e. , $N\sum_{n=0}^{N-1}|\tilde{x}[n]|^2 = \sum_{n=0}^{N-1}|\tilde{X}[n]|^2$.
One can view $\{\tilde{X}[k]\}$ as just a *different* representation of $\{\tilde{x}[n]\}$.

Example 3.2 (Discrete Fourier Series) *Consider the periodic discrete signal $\tilde{x}[n]$ of period 10, defined on $0 \le n \le 9$ as*

$$\tilde{x}[n] = \begin{cases} 1 & 0 \le n \le 4 \\ -1 & 5 \le n \le 9 \end{cases}$$

(a) Find $\tilde{X}[k]$, the discrete Fourier series of $\tilde{x}[n]$.

(b) Compare $\tilde{X}[3]$ and $\tilde{X}[-33]$. What is the period of $\tilde{X}[n]$?

(c) Define $\tilde{y}[n] = \tilde{x}[n-5]$. What is the discrete Fourier series of $\tilde{y}[n]$?

(d) Let $\tilde{z}[n] = \tilde{X}[n]$. Find the discrete Fourier series of $\tilde{z}[n]$ and compare it to $\tilde{x}[n]$.

The answers to these questions are:

(a)

$$
\begin{aligned}
\tilde{X}[k] &= \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{10}nk} \\
&= \sum_{n=0}^{4} e^{-j\frac{2\pi}{10}nk} - \sum_{n=5}^{9} e^{-j\frac{2\pi}{10}nk} \\
&= \sum_{n=0}^{4} e^{-j\frac{2\pi}{10}nk} - e^{-j\pi k} \sum_{n'=0}^{4} e^{-j\frac{2\pi}{10}n'k} \\
&= \left(1 - e^{-j\pi k}\right) \frac{1 - e^{-j\frac{2\pi}{10}5k}}{1 - e^{-j\frac{2\pi}{10}k}} \\
&= \begin{cases} 0, & \textit{for } k \textit{ even} \\ \frac{4}{1-e^{-j\frac{2\pi}{10}k}}, & \textit{for } k \textit{ odd.} \end{cases}
\end{aligned}
$$

(b) We can immediately say that the period of $\tilde{X}[k]$ is $N = 10$. For any sequence of period N, it's DFS is of period N. This is a basic property of the DFS. We can easily derive this property as follows: Let $\tilde{y}[n]$ be a (any) sequence of period M. Then

$$
\begin{aligned}
\tilde{Y}[k+M] &= \sum_{n=0}^{M-1} \tilde{y}[n] e^{-j\frac{2\pi}{M}n(k+M)} \\
&= \sum_{n=0}^{M-1} \tilde{y}[n] e^{-j\frac{2\pi}{M}nk} e^{-j2\pi n} \\
&= \sum_{n=0}^{M-1} \tilde{y}[n] e^{-j\frac{2\pi}{M}nk} \\
&= \tilde{Y}[k],
\end{aligned}
$$

so $\tilde{Y}[k]$, the DFS of $\tilde{y}[n]$ has period M.
Since $\tilde{X}[k]$ has period 10, $\tilde{X}[-33] = \tilde{X}[7] = \tilde{X}[N-3]$.
For any DFS of a real periodic sequence

$$
\tilde{X}[N-k] = \tilde{X}[k]^*, \qquad k = 0, \ldots, N-1. \tag{3.17}
$$

This gives $\tilde{X}[-33] = \tilde{X}[3]^*$.

Equation (3.17) can easily be derived:

$$\begin{aligned}\tilde{X}[N-k]^* &= \left(\sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}n(N-k)}\right)^* \\ &= \sum_{n=0}^{N-1} \tilde{x}[n]^* e^{-j\frac{2\pi}{N}nk} \\ &= \tilde{X}[k],\end{aligned}$$

where the last equality follows from the assumption that the input sequence is real.

(c) For $\tilde{y}[n] = \tilde{x}[n-5]$ we use another basic property of the DFS:

$$\tilde{x}[n-n_0] \overset{DFS}{\leftrightarrow} e^{-j\frac{2\pi}{N}n_0 k}\tilde{X}[k]. \qquad (3.18)$$

Again, we can easily derive this ourselves:

$$\begin{aligned}\tilde{Y}[k] &= \sum_{n=0}^{N-1} \tilde{x}[n-n_0] e^{-j\frac{2\pi}{N}nk} \\ &= \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}(n+n_0)k} \\ &= e^{-j\frac{2\pi}{N}n_0 k}\tilde{X}[k].\end{aligned}$$

Using (3.18) gives us

$$\tilde{Y}[k] = e^{-j\frac{2\pi}{10}5k}\tilde{X}[k] = -\tilde{X}[k].$$

(d) We are asked to compute

$$\tilde{Z}[n] = \sum_{k=0}^{N-1} \tilde{X}[k] e^{-j\frac{2\pi}{N}nk}.$$

Remember that all sequences are periodic. If we consider $\tilde{Z}[-n]/N$, we get

$$\tilde{Z}[-n]/N = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}[k] e^{j\frac{2\pi}{N}nk},$$

which is exactly the reconstruction formula for $\tilde{x}[n]$, so $\tilde{Z}[n] = \tilde{x}[-n]/N$.

3.3 The Discrete Fourier Transform (DFT)

We will now develop a similar Fourier representation for a finite-length signal. The basic idea is to consider a signal of length N, such that

$$x[n] = 0, \qquad n \geq N, n < 0$$

and represent it as a sum of N *finite-length* complex exponentials. To do this we borrow the idea from the discrete Fourier series representation. We can think of the a-periodic finite length sequence $\{x[n]\}_{n=0}^{n=N-1}$ as a *single-period* of a periodic sequence $\tilde{x}[n]$ by constructing $\tilde{x}[n]$ as ,

$$\tilde{x}[n] = x[n], \qquad 0 \leq n \leq N-1$$

$$\tilde{x}[n] = \tilde{x}[n+N], \qquad \forall n$$

Therefore using the Fourier-series representation we have

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k] e^{j\frac{2\pi}{N}kn} \tag{3.19}$$

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}. \tag{3.20}$$

In matrix form, the *Discrete Fourier Transform (DFT)* can be written as

$$\mathbf{x} = \frac{1}{N}\Lambda^H \cdot \mathbf{X} \tag{3.21}$$

$$\mathbf{X} = \Lambda \cdot \mathbf{x} \tag{3.22}$$

where $\mathbf{x} = \left[x[0]x[1]\ldots x[N-1]\right]^T$ and $\mathbf{X} = \left[X[0]X[1]\ldots X[N-1]\right]^T$.

Therefore, another interpretation of the DFT is in terms of basis representation. Given any vector $\mathbf{x} \in \mathbb{C}$, we can think of

$$\mathbf{x} = \sum_{n=0}^{N-1} x[n]\mathbf{e}^n$$

where $\mathbf{e}^n \in \mathbb{C}^N$ is the "unit vector"

$$\mathbf{e}^n = \begin{bmatrix} e_0^{(n)} \\ e_1^{(n)} \\ \vdots \\ e_{N-1}^{(n)} \end{bmatrix} \in \mathbb{C}^N$$

with components

$$e_i^{(n)} = \begin{cases} 1 & i = n \\ 0 & i \neq n. \end{cases}$$

By looking at the DFT, we see that

$$\mathbf{x} = \frac{1}{N}\Lambda^H \cdot \mathbf{X}$$

where

$$\Lambda^H = \begin{bmatrix} \left(\mathbf{w}^{(0)}\right)^T \\ \left(\mathbf{w}^{(1)}\right)^T \\ \vdots \\ \left(\mathbf{w}^{(N-1)}\right)^T \end{bmatrix} = \begin{bmatrix} 1 & \cdots & 1 \\ e^{j\frac{2\pi}{N}0} & \cdots & e^{j\frac{2\pi}{N}(N-1)} \\ \vdots & \ddots & \vdots \\ e^{j\frac{2\pi(N-1)}{N}0} & \cdots & e^{j\frac{2\pi(N-1)^2}{N}} \end{bmatrix}$$

and

$$\mathbf{w}^{(k)} = \begin{bmatrix} 1 \\ e^{j\frac{2\pi}{N}k} \\ \vdots \\ e^{j\frac{2\pi(N-1)}{N}k} \end{bmatrix} \in \mathbb{C}^{N\times 1}$$

is an expansion of $\mathbf{x}$ in the orthogonal *basis*,

$$\left\{ \left(\mathbf{w}^{(0)}\right)^T, \cdots, \left(\mathbf{w}^{(N-1)}\right)^T \right\}$$

with coefficients $\frac{1}{N}\tilde{X}[k]$. This gives another interpretation of the DFT as an expansion of a sequence in another "basis" set, and therefore giving it an alternate representation. This viewpoint is actually quite useful and general and to be able to utilize it we make a detour in the next chapter to understand vector spaces.

Example 3.3 (Discrete Fourier Transform) *Derive the DFT for a general sinusoidal sequence,* $\tilde{x}[n] = \sin(\frac{2\pi L}{N}n + \theta)$, $n = 0, \ldots, N$.

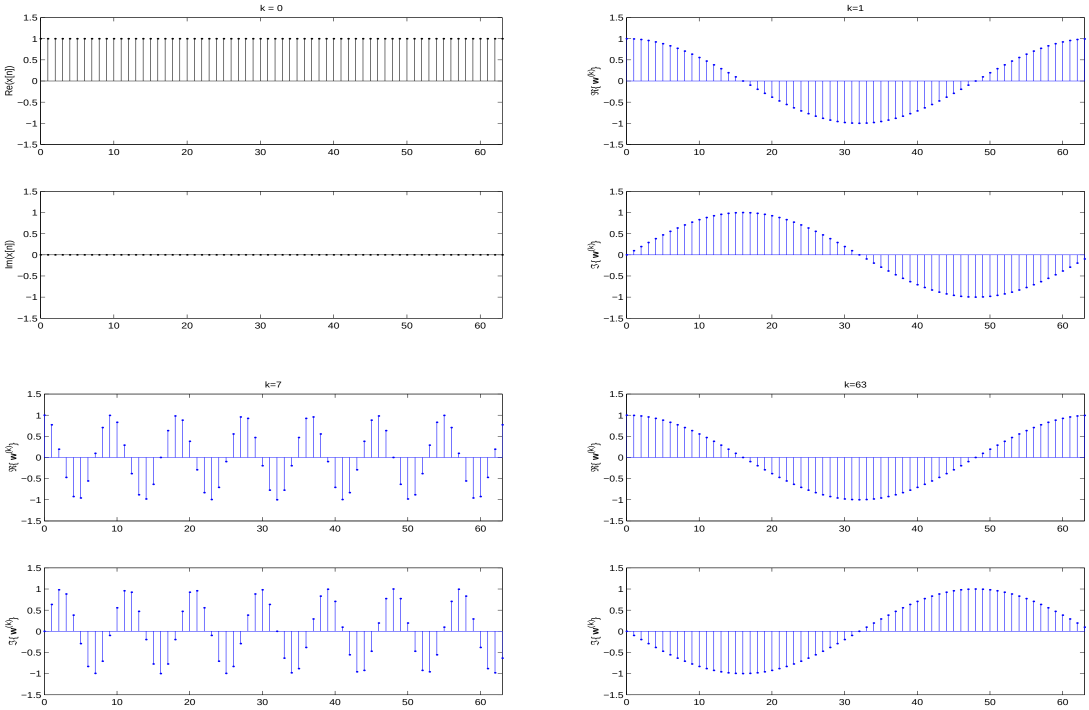


Figure 3.2: Some DFT basis vectors $\mathbf{w}^{(k)}$ for $N = 64$; $k = 0, 1, 7$ and 63.

For L a positive integer we have

$$\begin{aligned} X[k] &= \sum_{n=0}^{N-1} \sin(\frac{2\pi L}{N} n + \theta) e^{-j\frac{2\pi}{N} nk} \\ &= \sum_{n=0}^{N-1} \frac{e^{j(\frac{2\pi L}{N} n + \theta)} - e^{-j(\frac{2\pi L}{N} n + \theta)}}{2j} e^{-j\frac{2\pi}{N} nk} \\ &= \frac{e^{j(-\pi/2+\theta)}}{2} \sum_{n=0}^{N-1} e^{-j(\frac{2\pi}{N} n(k-L)} + \frac{e^{j(\pi/2-\theta)}}{2} \sum_{n=0}^{N-1} e^{-j(\frac{2\pi}{N} n(k+L)} \\ &= \begin{cases} \frac{N}{2} e^{j(-\pi/2+\theta)} & k = L \\ \frac{N}{2} e^{j(\pi/2-\theta)} & k = N - L. \end{cases} \end{aligned}$$

3.4 Properties of the DFS

Symmetries & Structure. The DFS of a ***real*** sequence $\tilde{x}[n] \in \mathbb{R}$ possesses the following symmetries:

$$\tilde{X}[k] = \tilde{X}^*[-k] \qquad \textit{the transform is conjugate-symmetric} \tag{3.23}$$
$$|\tilde{X}[k]| = |\tilde{X}[-k]| \qquad \textit{the magnitude is symmetric} \tag{3.24}$$
$$\measuredangle\tilde{X}[k] = -\measuredangle\tilde{X}[-k] \qquad \textit{the phase is antisymmetric} \tag{3.25}$$
$$\mathrm{Re}\{\tilde{X}[k]\} = \mathrm{Re}\{\tilde{X}[-k]\} \qquad \textit{the real part is symmetric} \tag{3.26}$$
$$\mathrm{Im}\{\tilde{X}[k]\} = -\mathrm{Im}\{\tilde{X}[-k]\} \qquad \textit{the imaginary is antisymmetric} \tag{3.27}$$

Finally, if $x[n]$ is real and symmetric (using the symmetry definition in (3.34)), then the DFS is real:

$$\tilde{x}[k] = \tilde{x}[-k] \Longleftrightarrow \tilde{X}[k] \in \mathbb{R} \tag{3.28}$$

while, for real antisymmetric signals we have that the DFS is purely imaginary.

Linearity & Shifts. The DFS is a linear operator, since it is a matrix-vector product. A shift in the discrete-time domain leads to multiplication by a phase term in the frequency domain:

$$\tilde{x}[n - n_0] \stackrel{\text{DFS}}{\longleftrightarrow} W_N^{kn_0}\tilde{X}[k] \tag{3.29}$$

while multiplication of the signal by a complex exponential of frequency a multiple of $2\pi/N$ leads to a shift in frequency:

$$W_N^{-nL}\tilde{x}[n] \stackrel{\text{DFS}}{\longleftrightarrow} \tilde{X}[k - L]. \tag{3.30}$$

Energy Conservation. We have already seen the conservation of energy property in the context of basis expansion. Here, we will simply recall Parseval's Theorem, which states:

$$\sum_{n=0}^{N-1} |\tilde{x}[n]|^2 = \frac{1}{N}\sum_{k=0}^{N-1} |\tilde{X}[k]|^2. \tag{3.31}$$

3.5 Properties of the DFT

The properties of the DFT are obviously the same as those for the DFS, given the formal equivalence of the transforms. The only detail is how to interpret shifts, index reversal

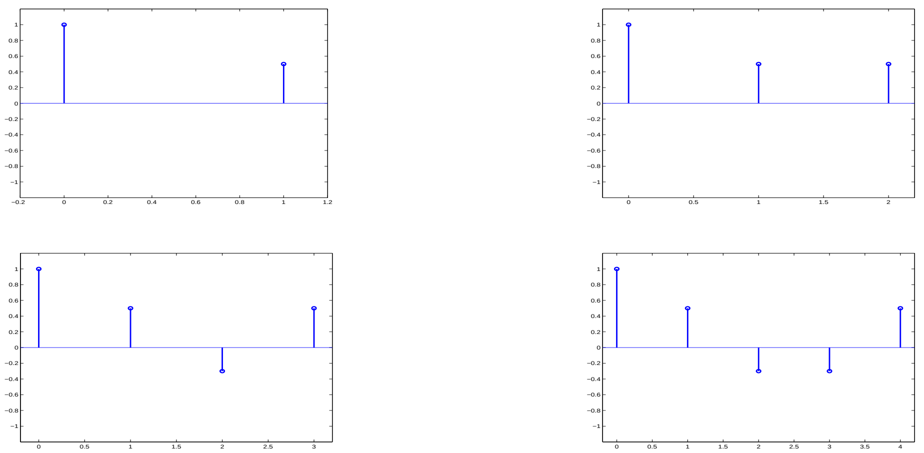


Figure 3.3: Examples of finite-length symmetric signals for $N = 2, 3, 4, 5$.

and symmetries for finite, length-N vectors; this is easily solved by considering the fact that the DFT subsumes an N-periodic structure and therefore the underlying model for the signal is that of periodic extension. We can therefore consider the periodized version of the signal, operate the shift and then take the values from 0 to $N-1$. Explicitly, shifts and index reversal of a length-N vector are carried out modulo N; the reversal of a signal $x[n] = \mathbf{x} = [x[0]\ x[1]\ \ldots\ x[N-1]]$ is:

$$x[-n \mod N] = [x[0]\ x[N-1]\ x[N-2]\ \ldots\ x[2]\ x[1]] \tag{3.32}$$

whereas its shift by k is the circular shift:

$$x[(n-k) \mod N] = [x[k]\ x[k-1]\ \ldots\ x[0]\ x[N-1]\ x[N-2]\ \ldots\ x[k+1]]. \tag{3.33}$$

This implies that, when we say a length-N signal $x[k]$ is *symmetric*, we have in fact:

$$x[k] = x[N-k], \quad k = 1, 2, \ldots, \lfloor (N-1)/2 \rfloor; \tag{3.34}$$

note that the index k starts off at one in the above definition and ends at the *floor* of $(N-1)/2$; this means that X[0] is always unconstrained and so is $x[N/2]$ for even-length signals. Figure 3.3 shows some examples of symmetric length-N signals for different values of N. Of course the same definition can be used for antisymmetric signals with just a change of sign.

Symmetries & Structure. The DFT of a ***real*** sequence $x[n] \in \mathbb{R}$ possesses the following symmetries:

$$
\begin{aligned}
X[k] &= X^*[-k \mod N] && \textit{the transform is conjugate-symmetric} && (3.35)\\
|X[k]| &= |X[-k \mod N]| && \textit{the magnitude is symmetric} && (3.36)\\
\measuredangle X[k] &= -\measuredangle X[-k \mod N] && \textit{the phase is antisymmetric} && (3.37)\\
\mathrm{Re}\{X[k]\} &= \mathrm{Re}\{X[-k \mod N]\} && \textit{the real part is symmetric} && (3.38)\\
\mathrm{Im}\{X[k]\} &= -\mathrm{Im}\{X[-k \mod N]\} && \textit{the imaginary is antisymmetric} && (3.39)
\end{aligned}
$$

Finally, if $x[n]$ is real and symmetric (using the symmetry definition in (3.34), then the DFT is real:

$$x[k] = x[N-k], \quad k = 1, 2, \dots, \lfloor (N-1)/2 \rfloor \Longleftrightarrow X[k] \in \mathbb{R} \tag{3.40}$$

while, for real antisymmetric signals we have that the DFT is purely imaginary.

Linearity & Shifts. The DFT is obviously a linear operator. A circular shift in the discrete-time domain leads to multiplication by a phase term in the frequency domain:

$$x[(n - n_0) \mod N] \overset{\text{DFT}}{\longleftrightarrow} W_N^{kn_0} X[k] \tag{3.41}$$

while the finite-length equivalent of the Modulation theorem states:

$$W_N^{-nL} x[n] \overset{\text{DFT}}{\longleftrightarrow} X[(k - L) \mod N]. \tag{3.42}$$

Energy Conservation. See (3.31).

3.6 Summary

This chapter introduced the concept of Fourier Transform for digital signals. The main points have been:

- A review of complex exponentials, finding a set of orthogonal complex exponentials;
- The DFS for periodic sequences;
- The DFT as a change of basis in $\mathbb{C}^N$, both in matrix and explicit form;
- Symmetries and structures of the two transforms.

Here is a tables of common DFT transforms:

Some DFT Pairs for Length-N Signals:	$(n, k = 0, 1, \ldots, N-1)$
$x[n] = \delta[n-k]$	$X[k] = e^{-j\frac{2\pi}{N}k}$
$x[n] = 1$	$X[k] = N\delta[k]$
$x[n] = e^{j\frac{2\pi}{N}L}$	$X[k] = N\delta[k-L]$
$x[n] = \cos(\frac{2\pi}{N}Ln + \phi)$	$X[k] = (N/2)[e^{j\phi}\delta[k-L] + e^{-j\phi}\delta[k-N+L)]]$
$x[n] = \sin(\frac{2\pi}{N}Ln + \phi)$	$X[k] = (-jN/2)[e^{j\phi}\delta[k-L] - e^{-j\phi}\delta[k-N+L]]$
$x[n] = \begin{cases} 1 & \text{for } n \leq M-1 \\ 0 & \text{for } M \leq n \leq N-1 \end{cases}$	$X[k] = \dfrac{\sin((\pi/N)Mk)}{\sin((\pi/N)k)} e^{-j\frac{\pi}{N}(M-1)k}$

3.7 Problems

Problem 3.1 *Derive the formula for the DFT of the length-N signal*

$$x[n] = \cos((2\pi/N)Ln + \phi).$$

Problem 3.2 *Consider a length-64 signal $x[n]$ which is the sum of the three sinusoidal signals plotted in Figure 3.4. Compute the DFT coefficients $X[k], k = 0, 1, \ldots, 63$ using the results from Problem 3.1.*

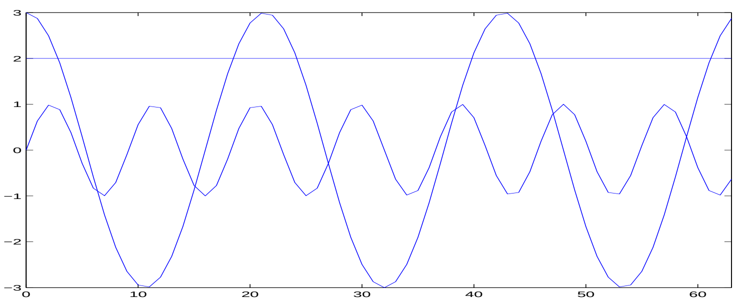


Figure 3.4: Three sinusoidal signals.

Problem 3.3 *The DFT and inverse DFT (IDFT) formulas are similar, but not identical. Consider a length-N signal* $x[n], N = 0, \ldots, N-1$*; what is the length-N signal* $y[n]$ *obtained as*

$$y[n] = DFT\{DFT\{x[n]\}\}$$

(i.e. by applying the DFT algorithm twice in a row)?

Problem 3.4 (Implementing DFT in MATLAB) *In this exercise we want to provide a simple m-file in MATLAB to compute the discrete Fourier transform of a given sequence. The inputs of the function are the input sequence* x *as a row-vector and the length of the transform* N*. It checks the length of* x *to be satisfied with* N*. Then a transformation matrix* W *will be formed and the DFT vector* X *will be produced by a matrix-vector multiplication. The magnitude of the DFT should be plotted at the end.*

Download the m-file `myDFT` *from the course website and put it into your* `work` *directory. Fill the blanks and run the function to compute and plot the DFT of* $x[n]$*,* $n = 0, \ldots, 45$*, given in Exercise 1. Read MATLAB help for the standard function "*`fft`*". Compare the output of your function to output of* `fft`*.*

Problem 3.5 (DFT with Different Lengths) *Consider the finite length sequences*

$$x[n] = y[n] = \begin{cases} 1 & 0 \le n \le 5 \\ 0 & otherwise \end{cases}.$$

(a) Use your `myDFT` *m-file to find the DFT of length* 6 *for* $x[n]$*.*

(b) Repeat part (a) to compute length 12 *DFT of* $x[n]$*.*

Let $a[n]$ *and* $b[n]$ *be two length* N *sequences. The* circular convolution *of the two sequences* $a[n]$ *and* $b[n]$ *is defined as*

$$a[n] \otimes b[n] = \sum_{m=0}^{N-1} a[m] b[(n-m) \bmod N].$$

Note that $b[(n-m) \bmod N]$ *is a circular shifted version of* $b[n]$*, i.e.*

$$b[(n-m) \bmod N] = \Big[b[N-m] \; b[N-m+1] \; \ldots \; b[N-1] \; b[0] \; \ldots \; b[N-m-1] \Big].$$

In the remaining parts of this exercise we are going to implement the circular convolution in MATLAB.

(c) Download the `rcshift` *m-file from the website and fill the blanks. At the end compute the circular shift of length 3 to the right of* `t(1:10)=sin([1:10])`*.*

(d) Download the `cir_conv` *and complete it according to its comments.*

(e) Use your `cir_conv` *m-file to compute the* 6*-point circular convolution of* $x[n]$ *and* $y[n]$*.*

(f) Compute the 12*-point circular convolution of* $x[n]$ *and* $y[n]$ *and call it in* $z[n]$*.*

(g) Compare the results of the two circular convolutions.

(h) compare the DFT of $z[t]$ *to the multiplication of DFT's of* $x[n]$ *and* $y[n]$*.*

Problem 3.6 *Compute the DFS of* $x[n] = \cos(\pi \frac{n}{3})$ *and* $y[n] = 1 + \cos(\pi \frac{n}{3})$