Introduction to Signals and Systems, Part III: Lecture Summary

In this lecture, we reviewed major concepts covered thus far in class, including the representation of signals in the time domain and the frequency domain, and applications of the frequency-domain representation, including filtering by frequency and amplitude modulation. Thus far, I have shown you one approach to filtering that involves the following three basic steps:

1. Convert a discrete-time signal to the frequency domain through the Fourier transform (FFT).
2. Zero undesired frequency components of the signal in the frequency domain.
3. Convert the modified frequency representation back to the time domain to view/hear the filtered signal.

While this approach is perfectly fine for some applications, it does not lend itself to implementation in a real-time or on-line manner; that is, as described thus far, this approach requires that we know the entire signal before we can filter it. In many cases, however, we want to be able to filter a signal in real time as the signal is coming in. For example, the output of an mp3 player or audio system adjusts almost instantaneously in response to listener changes in the equalizer knobs or slider bars.

In order to realize real-time filters and other similar systems, we need to introduce the concept of *difference equations*, which are the discrete-time version of continuous-time differential equations. In short, *difference equations are the mathematical representation of discrete-time systems*.

Before beginning a more general discussion of difference equations, let us start with some simple examples. Equation (1) below is a simple example of a difference equation:

$$y[n] = x[n]^2 + x[n-1] \tag{1}$$

In equation (1), $x[n]$ refers to the system input x at time index n, $n \in \{\ldots, -2, -1, 0, 1, 2, \ldots\}$, and $y[n]$ refers to the system output y at time index n; similarly, $x[n-1]$ refers to the system input at time index $n-1$. Note that while x and y are continuous-valued — that is they can assume any real value — the time index n can only assume integer values.

So what does equation (1) do? We can think of equation (1) as a system that takes a input sequence and transforms it according to the given difference equation; as input is coming in, the difference equation generates an output. Consider for example, the input sequence $x[n]$ plotted in Figure 1(a), which has non-zero values only for time indexes $n \in \{1, 2, 3, 4, 5, 6, 7\}$. The output sequence $y[n]$ corresponding to difference equation (1) is plotted in Figure 1(b). Below, we show the first few computations for determining $y[n]$:

$$y[0] = x[0]^2 + x[-1] = 0 + 0 = 0 \tag{2}$$

$$y[1] = x[1]^2 + x[0] = 1 + 0 = 1 \tag{3}$$

$$y[2] = x[2]^2 + x[1] = 4^2 + 1 = 17 \tag{4}$$

$$y[3] = x[3]^2 + x[2] = 3^3 + 4 = 13 \tag{5}$$

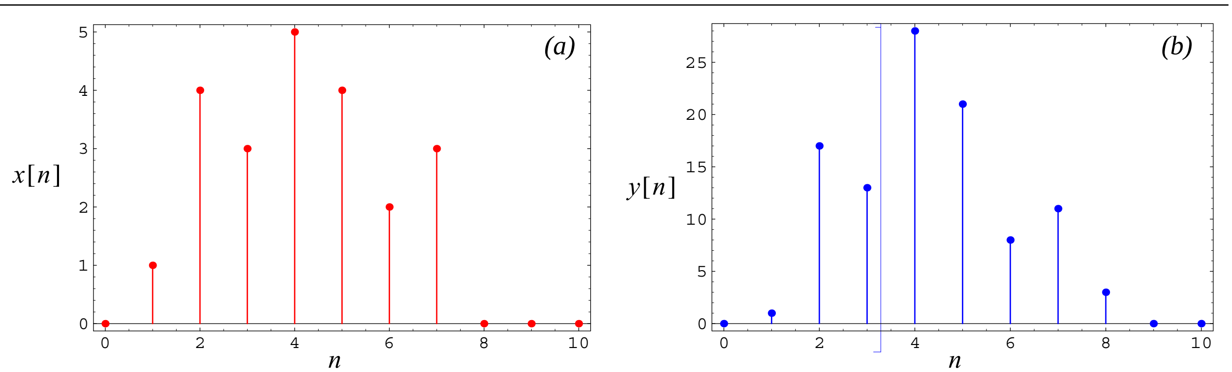

Figure 1

Equation (1) is an example of a *non-recursive* difference equation; that is, the right-hand side of the difference equation is dependent only on input values, and not on previous output values. Equation (6) below is an example of a *recursive* difference equation:[1]

$$y[n] = -1/2y[n-1] - x[n] \tag{6}$$

Note that the output at time index n is not only dependent on input values, but also on previous output values. For this equation, let us now try to compute the output sequence for the input sequence plotted in Figure 1(a):

$$y[0] = -1/2y[-1] - x[0] \tag{7}$$

As we can see from equation (7) above, in order to compute $y[0]$ we need to know the value of $y[-1]$; the value of $y[-1]$ is independent of the difference equation itself and must be specified as an *initial condition*. For this example, let us make the following assumption:

$$y[-1] \equiv 0 . \tag{8}$$

We are now ready to proceed with our computation:

$$y[0] = -1/2y[-1] - x[0] = 0 - 0 = 0 \tag{9}$$

$$y[1] = -1/2y[0] - x[1] = 0 - 1 = -1 \tag{10}$$

$$y[2] = -1/2y[1] - x[2] = (-1/2)(-1) - 4 = -3.5 \tag{11}$$

$$y[3] = -1/2y[2] - x[3] = (-1/2)(-3.5) - 3 = -1.25 \tag{12}$$

The complete input and output sequences for difference equation (6) are plotted in Figure 2 below.

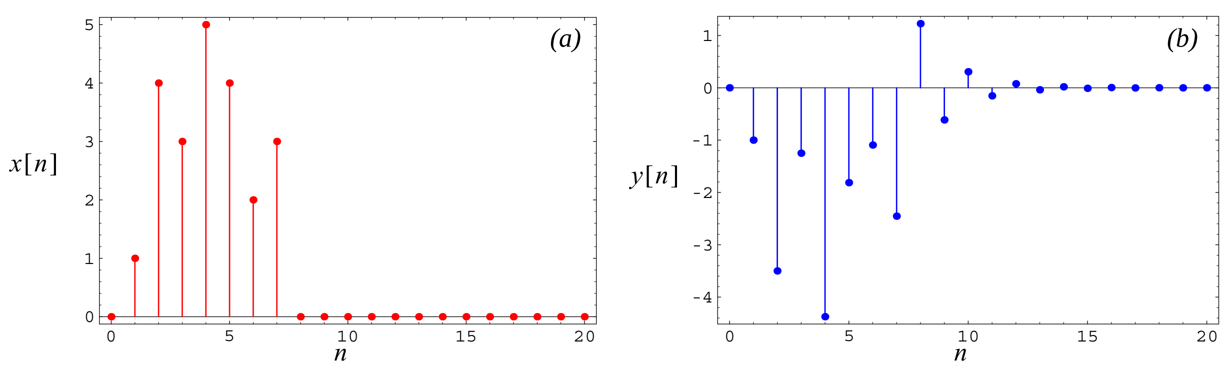

Figure 2

As we can see from the above two examples, it is straightforward to compute the output $y[n]$ for any difference equation and input $x[n]$[2]. In general, causal[3] difference equations can assume the following form:

$$y[n] = f(y[n-1], \ldots, y[n-N], x[n], \ldots, x[n-M]) \tag{13}$$

where $f(\cdot)$ is some arbitrary multi-variable function. In this class, however, we will restrict ourselves mostly to the study of a class of difference equations known as *Linear, Time-Invariant (LTI)* systems. We will talk more about the

1. *Non-recursive difference equations are also known as FIR (Finite Impulse Response) systems; recursive difference equations are also known as IIR (Infinite Impulse Response) systems. The precise meaning of these terms will become clearer when we formally define what is meant by the "impulse response" of a system.*
2. *For recursive difference equations, we also need to know the initial conditions of the system, as we saw in the second example; more on this later.*
3. *The term "causal" refers to systems where the current output is dependent only on present or past values of the input and/or output; more on this later.*

precise meaning of the terms *linear* and *time-invariant* later; for now, it suffices to say that such systems, or difference equations, assume the following general form:

$$y[n] = \sum_{l=1}^{N} a_l y[n-l] + \sum_{k=0}^{M} b_k x[n-k] \tag{14}$$

If we expand the sums in equation (14), we can write:

$$y[n] = a_1 y[n-1] + a_2 y[n-2] + \dots + a_N y[n-N] + b_0 x[n] + b_1 x[n-1] + \dots + b_M x[n-M] \tag{15}$$

In equations (14) and (15), the coefficients a_l, $l \in \{1, 2, \dots, N\}$, and b_k, $k \in \{0, 1, \dots, M\}$, are assumed to be fixed scalar numbers; that is they don't change over time (hence the term *time-invariant*). Note that difference equation (1) is not an LTI system (because of the $x[n]^2$ term), while difference equation (6) is an LTI system.

So why do we restrict ourselves to these seemingly limited systems (i.e. difference equations)? The most straightforward answer is that a great deal of theoretical results have been developed for LTI systems that are not easily extended to more general classes of difference equations. Moreover, as we shall see in this class, a lot of sophisticated things can be achieved in discrete-time systems, even if we restrict ourselves to only LTI systems.

So far, we have seen how we can simulate difference equations; that is, how we can compute the output of a difference equation for a specific input sequence $x[n]$ and (in the case of recursive difference equations) a set of initial conditions. In this course, however, we will go much further, learning to analyze the properties and predict the output of LTI systems without having to resort to explicit simulation for specific input sequences. Some important concepts and analysis techniques we will develop along the way are (1) *frequency response* and (2) *stability* of a system. We will study these concepts more closely next time.