MIT 6.02 **DRAFT** *Lecture Notes*
(Last update: March 17, 2012)
Comments, questions or bug reports?
Please contact {verghese, hari} at mit.edu

CHAPTER 10
Models for Physical Communication Channels

To preview what this chapter is about, it will be helpful first to look back briefly at the territory we have covered. The previous chapters have made the case for a digital (versus analog) communication paradigm, and have exposed us to communication at the level of *bits* or, more generally, at the level of the discrete symbols that encode messages.

We showed, in Chapters 2 and 3, how to obtain compressed or non-redundant representations of a discrete set of messages through source coding, which produced codewords that reflected the inherent information content or entropy of the source. In Chapter 4 we examined how the source transmitter might map a bit sequence to clocked *signals* that are suited for transmission on a physical channel (for example, as voltage levels).

Chapter 5 introduced the binary symmetric channel (BSC) abstraction for bit errors on a channel, with some associated probability of corrupting an individual bit on passage through the channel, independently of what happens to every other bit in the sequence. That chapter, together with Chapters 6, 7, and 8, showed how to re-introduce redundancy, but in a controlled way using parity bits. This resulted in error-correction codes, or channel codes, that provide some level of protection against the bit-errors introduced by the BSC.

Chapter 9 considered the challenges of "demapping" back from the received noise-corrupted signal to the underlying bit stream, assuming that the channel introduced no deterministic distortion, only additive white noise on the discrete-time samples of the received signal. A key idea from Chapter 9 was showing how Gaussian noise experienced by analog signals led to the BSC bit-flip probability for the discrete version of the channel.

The present chapter begins the process—continued through several subsequent chapters—of representing, modeling, analyzing, and exploiting the characteristics of the physical transmission channel. This is the channel seen between the signal transmitted from the source and the signal captured at the receiver. Referring back to the "single-link view" in the upper half of Figure 4-8 in Chapter 4, our intent is to study in more detail the portion of the communication channel represented by the connection between "Mapper + Xmit samples" at the source side, and "Recv samples + Demapper" at the receiver side.

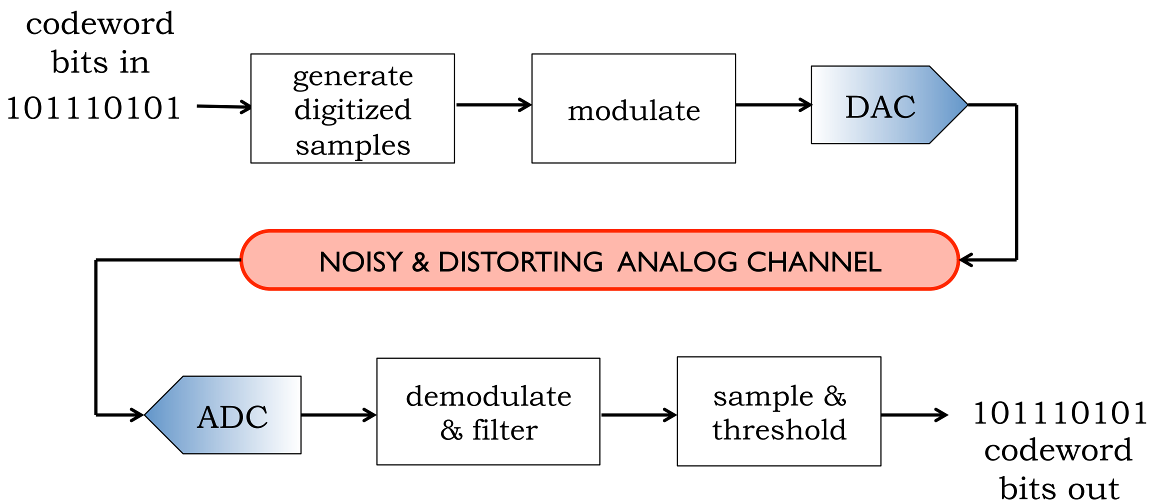


Figure 10-1: Elements of a communication channel between the channel coding step at the transmitter and channel decoding at the receiver.

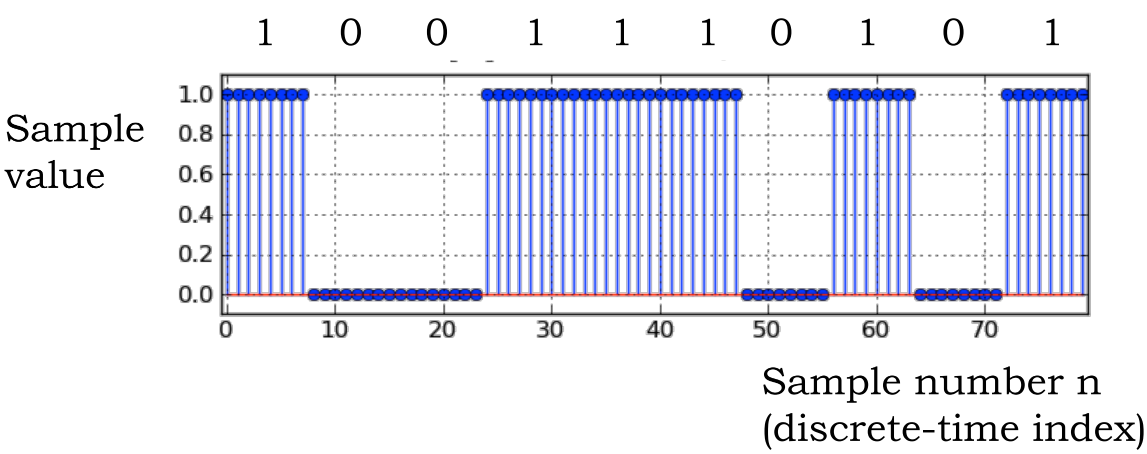


Figure 10-2: Digitized samples of the baseband signal.

10.1 Getting the Message Across

10.1.1 The Baseband Signal

In Figure 10-1 we see an expanded version of what might come between the channel coding operation at the transmitter and the channel decoding operation at the receiver (as described in the upper portion of Figure 4-8). At the source, the first stage is to convert the input *bit* stream to a digitized and *discrete-time (DT) signal*, represented by *samples* produced at a certain sample rate f_s samples/s. We denote this signal by $x[n]$, where n is the integer-valued discrete-time index, ranging in the most general case from $-\infty$ to $+\infty$.

In the simplest case, which we will continue to use for illustration, each bit is represented by a signal level held for a certain number of samples, for instance a voltage level of $V_0 = 0$ held for 8 samples to indicate a 0 bit, and a voltage level of $V_1 = 1$ held for 8 samples to indicate a 1 bit, as in Figure 10-2. The sample clock in this example operates at 8 times the rate of the bit clock, so the bit rate is $f_s/8$ bits/s. Such a signal is usually referred to as the **baseband signal**.

■ 10.1.2 Modulation

The DT baseband signal shown in Figure 10-2 is typically not ready to be transmitted on the physical transmission channel. For one thing, physical channels typically operate in continuous-time (CT) analog fashion, so at the very least one needs a digital-to-analog converter (DAC) to produce a continuous-time signal that can be applied to the channel. The DAC is usually a simple *zero-order hold*, which maintains or holds the most recent sample value for a time interval of $1/f_s$. With such a DAC conversion, the DT "rectangular-wave" in Figure 10-2 becomes a CT rectangular wave, each bit now corresponding to a signal value that is held for $8/f_s$ seconds.

Conversion to an analog CT signal will not suffice in general, because the physical channel is usually not well suited to the transmission of rectangular waves of this sort. For instance, a speech signal from a microphone may, after appropriate coding for digital transmission, result in 64 kilobits of data per second (a consequence of sampling the microphone waveform at 8 kHz and 8-bit resolution), but a rectangular wave switching between two levels at this rate is not adapted to direct radio transmission. The reasons include the fact that efficient projection of wave energy requires antennas of dimension comparable with the wavelength of the signal, typically a quarter wavelength in the case of a tower antenna. At 32 kHz, corresponding to the waveform associated with alternating 1's and 0's in the coded microphone output, and with the electromagnetic waves propagating at 3×10^8 meters/s (the speed of light), a quarter-wavelength antenna would be a rather unwieldy $3 \times 10^8/(4 \times 32 \times 10^3) = 2344$ meters long!

Even if we could arrange for such direct transmission of the baseband signal (after digital-to-analog conversion), there would be issues related to the required transmitter power, the attenuation caused by the atmosphere at this frequency, interference between this transmission and everyone else's, and so on. Regulatory organizations such as the U.S. Federal Communications Commission (FCC), and equivalent bodies in other countries, impose constraints on transmissions, which further restrict what sort of signal can be applied to a physical channel.

In order to match the baseband signal to the physical and regulatory specifications of a transmission channel, one typically has to go through a **modulation** process. This process converts the digitized samples to a form better suited for transmission on the available channel. Consider, for example, the case of direct transmission of digital information on an acoustic channel, from the speaker on your computer to the microphone on your computer (or another computer within "hearing" distance). The speaker does not respond effectively to the piecewise-constant voltages that arise from our baseband signal. It is instead designed to respond to *oscillatory* voltages at frequencies in the appropriate range, producing and projecting a wave of oscillatory acoustic pressure. Excitation by a sinusoidal wave produces a pure acoustic tone. With a speaker aperture dimension of about 5 cm (0.05 meters), and a sound speed of around 340 meters/s, we anticipate effective projection of tones with frequencies in the low kilohertz range, which is indeed in (the high end of) the audible range.

A simple way to accomplish the desired modulation in the acoustic wave example above is to apply—at the output of the digital-to-analog converter, which feeds the loudspeaker—a voltage $V_0 \cos(2\pi f_c t)$ for some duration of time to signal a 0 bit, and a voltage of the form $V_1 \cos(2\pi f_c t)$ for the same duration of time to signal a 1 bit.[1] Here $\cos(2\pi f_c t)$ is referred to as the *carrier signal* and f_c is the *carrier frequency*, chosen to be appropriately matched to the channel characteristics. This particular way of imprinting the baseband signal on a carrier by varying its amplitude is referred to as *amplitude modulation* (AM), which we will study in more detail in Chapter 14. The choice $V_0 = 0$ and $V_1 = 1$ is also referred to as *on-off keying*, with a burst of pure tone ("on") signaling a 1 bit, and an interval of silence ("off") signaling a 0.

One could also choose $V_0 = -1$ and $V_1 = +1$, which would result in a sinusoidal voltage that switches phase by $\pi/2$ each time the bit stream goes from 0 to 1 or from 1 to 0. This approach may be referred to as *polar keying* (particularly when it is thought of as an instance of amplitude modulation), but is more commonly termed *binary phase-shift keying* (BPSK). Yet another modulation possibility for this acoustic example is *frequency modulation* (FM), where a tone burst of a particular frequency in the neighborhood of f_c is used to signal a 0 bit, and a tone burst at another frequency to signal a 1 bit. All these schemes are applicable to radio frequency (RF) transmissions as well, not just acoustic transmissions, and are in fact commonly used in practice for RF communication.

■ 10.1.3 Demodulation

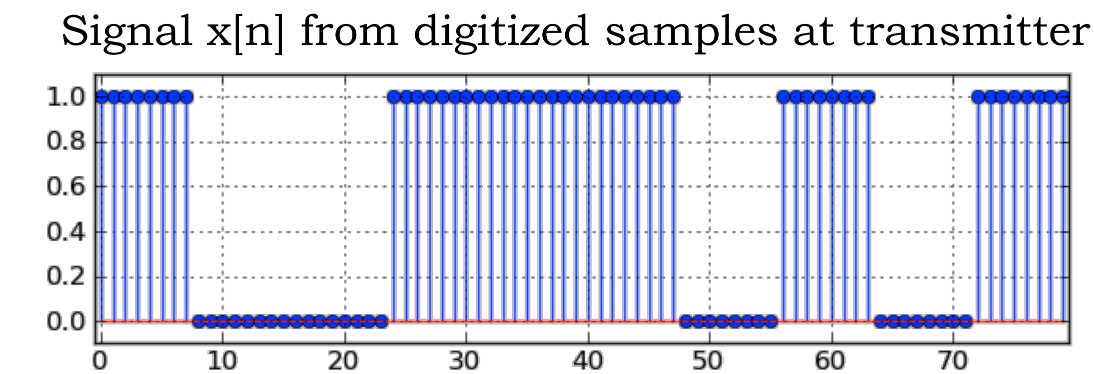
We shall have more to say about demodulation later, so for now it suffices to think of it as a process that is inverse to modulation, aimed at extracting the baseband signal from the received signal. While part of this process could act directly on the received CT analog signal, the block diagram in Figure 10-1 shows it all happening in DT, following conversion of the received signal using an analog-to-digital converter (ADC). The block diagram also indicates that a *filtering* step may be involved, to separate the channel noise as much as possible from the signal component of the received signal, as well as to compensate for deterministic distortion introduced by the channel. These ideas will be explored further in later chapters.

■ 10.1.4 The Baseband Channel

The result of the demodulation step and any associated filtering is a DT signal $y[n]$, comprising samples arriving at the rate f_s used for transmission at the source. We assume issues of clock synchronization are taken care of separately. We also neglect the effects of any signal attenuation, as this can be simply compensated for at the receiver by choosing an appropriate amplifier gain.

In the ideal case of no distortion, no noise on the channel, and insignificant propagation delay, $y[n]$ would exactly equal the modulating baseband signal $x[n]$ used at the source, for all n. If there is a fixed and known propagation delay on the channel, it can be convenient to simply set the clock at the receiver that much later than the clock at the sender. If this is done, then again we would find that in the absence of distortion and random noise, we get $y[n] = x[n]$ for all n.

[1] A zero-order-hold DAC will produce only an approximation of a pure sinusoid, but if the sample rate f_s is sufficiently high, the speaker may not sense the difference.



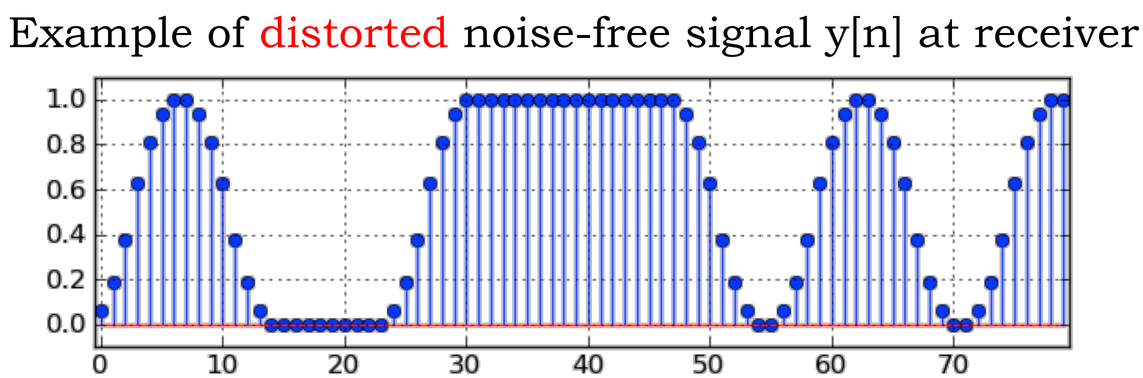


Figure 10-3: Channel distortion example. The distortion is deterministic.

More realistically, the channel does distort the baseband signal, so the output DT signal may look (in the noise-free case) as the lower waveform in Figure 10-3. Our objective in what follows is to develop and analyze an important class of models, namely *linear and time-invariant (LTI)* models, that are quite effective in accounting for such distortion, in a vast variety of settings. The models would be used to represent the end-to-end behavior of what might be called the *baseband channel*, whose input is $x[n]$ and output is $y[n]$, as in Figure 10-3.

■ 10.2 Linear Time-Invariant (LTI) Models

■ 10.2.1 Baseband Channel Model

Our baseband channel model, as represented in the block diagram in Figure 10-4 takes the DT sequence or signal $x[.]$ as input and produces the sequence or signal $y[.]$ as output. We will often use the notation $x[.]$—or even simply just x—to indicate the entire DT signal or function. Another way to point to the entire signal, though more cumbersome, is by referring to "$x[n]$ for $-\infty < n < \infty$"; this often gets abbreviated to just "the signal $x[n]$", at the risk of being misinterpreted as referring to just the value at a single time n.

Figure 10-4 shows $x[n]$ at the input and $y[n]$ at the output, but that is only to indicate that this is a snapshot of the system at time n, so indeed we see $x[n]$ at the input and $y[n]$ at

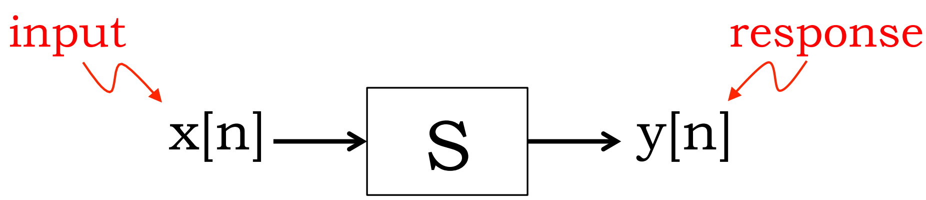


Figure 10-4: Input and output of baseband channel.

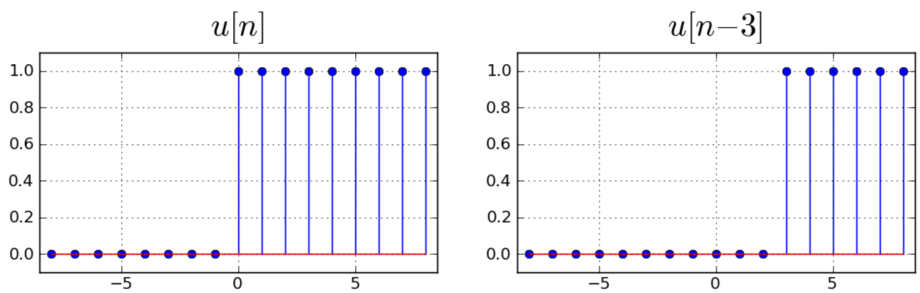


Figure 10-5: A unit step. In the picture on the left the unit step is unshifted, switching from 0 to 1 at index (time) 0. On the right, the unit step is shifted forward in time by 3 units (shifting forward in time means that we use the $-$ sign in the argument because we want the switch to occur with $n-3=0$).

the output of the system. What the diagram should **not** be interpreted as indicating is that the value of the output signal $y[.]$ at time n depends exclusively on the value of the input signal *at that same time* n. In general, the value of the output $y[.]$ at time n, namely $y[n]$, could depend on the values of the input $x[.]$ at *all* times. We are most often interested in **causal** models, however, and those are characterized by $y[n]$ only depending on *past and present* values of $x[.]$, i.e., $x[k]$ for $k \leq n$.

■ 10.2.2 Unit Sample Response $h[n]$ and Unit Step Response $s[n]$

There are two particular signals that will be very useful in our description and study of LTI channel models. The **unit step** signal or function $u[n]$ is defined as

$$\begin{aligned} u[n] &= 1 \text{ if } n \geq 0 \\ u[n] &= 0 \text{ otherwise} \end{aligned} \tag{10.1}$$

It takes the value 0 for negative values of its argument, and 1 everywhere else, as shown in Figure 10-5. Thus $u[1-n]$, for example, is 0 for $n > 1$ and 1 elsewhere.

The **unit sample** signal or function $\delta[n]$, also called the **unit impulse** function, is defined as

$$\begin{aligned} \delta[n] &= 1 \text{ if } n = 0 \\ \delta[n] &= 0 \text{ otherwise.} \end{aligned} \tag{10.2}$$

It takes the value 1 when its argument is 0, and 0 everywhere else, as shown in Figure 10-6.

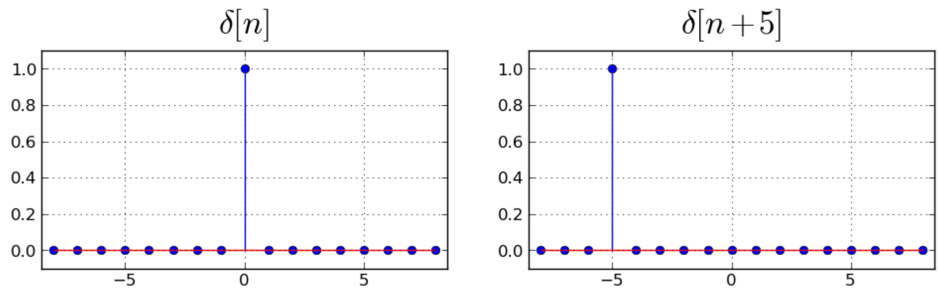


Figure 10-6: A unit sample. In the picture on the left the unit sample is unshifted, with the spike occurring at index (time) 0. On the right, the unit sample is shifted backward in time by 5 units (shifting backward in time means that we use the + sign in the argument because we want the switch to occur with $n + 5 = 0$).

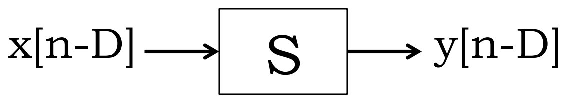


Figure 10-7: Time-invariance: if for all possible sequences $x[.]$ and integers D, the relationship between input and output is as shown above, then S is said to be "time-invariant" (TI).

Thus $\delta[n-3]$ is 1 where $n = 3$ and 0 everywhere else. One can also deduce easily that

$$\delta[n] = u[n] - u[n-1] , \tag{10.3}$$

where addition and subtraction of signals such as $u[n]$ and $u[n-1]$ are defined "point-wise", i.e., by adding or subtracting the values at each time instant. Similarly, the multiplication of a signal by a scalar constant is defined as pointwise multiplication by this scalar, so for instance $2u[n-3]$ has the value 0 for $n < 3$, and the value 2 everywhere else.

The response $y[n]$ of the system in Figure 10-4 when its input $x[n]$ is the unit sample signal $\delta[n]$ is referred to as the **unit sample response**, or sometimes the **unit impulse response**. We denote the output in this special case by $h[n]$. Similarly, the response to the unit step signal $u[n]$ is referred to as the **unit step response**, and denoted by $s[n]$.

A particularly valuable use of the unit step function, as we shall see, is in representing a rectangular-wave signal as an alternating sum of delayed (and possibly scaled) unit step functions. An example is shown in Figure 10-9. We shall return to this decomposition later.

■ 10.2.3 Time-Invariance

Consider a DT system with input signal $x[.]$ and output signal $y[.]$, so $x[n]$ and $y[n]$ are the values seen at the input and output at time n. The system is said to be *time-invariant* if shifting the input signal $x[.]$ in time by an arbitrary positive or negative integer D to get a new input signal $x_D[n] = x[n-D]$ produces a corresponding output signal $y_D[n]$ that is just $y[n-D]$, i.e., is the result of simply applying the same shift D to the response $y[.]$ that was obtained with the original unshifted input signal. The shift corresponds to delaying the signal by D if $D > 0$, and advancing it by $|D|$ if $D < 0$. In particular, for a TI system, a shifted unit sample function at the input generates an identically shifted unit sample response at the output. Figure 10-7 illustrates time-invariance.

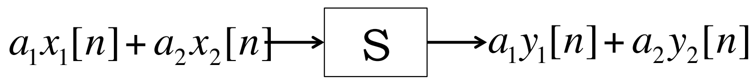


Figure 10-8: Linearity: if the input is the weighted sum of several signals, the response is the corresponding superposition (i.e., weighted sum) of the response to those signals.

The key to recognizing time-invariance in a given system description is to ask whether the rules or equations by which the input values $x[.]$ are combined, to create the output $y[n]$, involve knowing the value of n itself (or something equivalent to knowing n), or just time *differences* from the time n. If only the time differences from n are needed, the system is time-invariant. In this case, the same behavior occurs whether the system is run yesterday or today, in the following sense: if yesterday's inputs are applied today instead, then the output today is what we would have obtained yesterday, just occurring a day later.

Another operational way to recognize time-invariance is to ask whether shifting the pair of signals $x[.]$ and $y[.]$ by the arbitrary but identical amount D results in new signals $x_D[.]$ and $y_D[.]$ that still satisfy the equations defining the system. More generally, a set of signals that jointly satisfies the equations defining a system, such as $x[.]$ and $y[.]$ in our input-output example, is referred to as a *behavior* of the system. And what time-invariance requires is that time-shifting any behavior of the system by an arbitrary amount D results in a set of signals that is still a behavior of the system.

Consider a few examples. A system defined by the relation

$$y[n] = 0.5y[n-1] + 3x[n] + x[n-1] \quad \text{for all } n \tag{10.4}$$

is time-invariant, because to construct $y[.]$ at any time instant n, we only need values of $y[.]$ and $x[.]$ at the same time step and one time step back, no matter what n is — so we don't need to know n itself. To see this more concretely, note that the above relation holds for all n, so we can write

$$y[n-D] = 0.5y[(n-D)-1] + 3x[n-D] + x[(n-D)-1] \quad \text{for all } n$$

or

$$y_D[n] = 0.5y_D[n-1] + 3x_D[n] + x_D[n-1] \quad \text{for all } n\ .$$

In other words, the time-shifted input and output signals, $x_D[.]$ and $y_D[.]$ respectively, also satisfy the equation that defines the system.

The system defined by

$$y[n] = n^3 x[n] \quad \text{for all } n \tag{10.5}$$

is *not* time-invariant, because the value of n is crucial to defining the output at time n. A little more subtle is the system defined by

$$y[n] = x[0] + x[n] \quad \text{for all } n\ . \tag{10.6}$$

This again is *not* time-invariant, because the signal value at the absolute time 0 is needed, rather than a signal value that is offset from n by an amount that doesn't depend on n. We

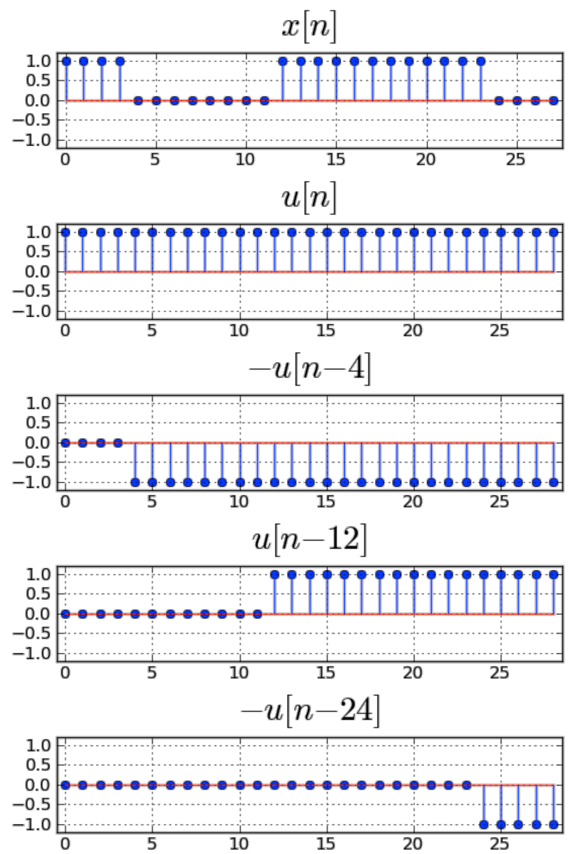


Figure 10-9: A rectangular-wave signal can be represented as an alternating sum of delayed (and possibly scaled) unit step functions. In this example, $x[n] = u[n] - u[n-4] + u[n-12] - u[n-24]$.

have $y_D[n] = x[0] + x_D[n]$ rather than what would be needed for time-invariance, namely $y_D[n] = x_D[0] + x_D[n]$.

■ 10.2.4 Linearity

Before defining the concept of linearity, it is useful to recall two operations on signals or time-functions that were defined in connection with Equation (10.3) above, namely (i) addition of signals, and (ii) scalar multiplication of a signal by a constant. These operations were defined as pointwise (i.e., occurring at each time-step), and were natural definitions. (They are completely analogous to vector addition and scalar multiplication of vectors, the only difference being that instead of the finite array of numbers that we think of for a vector, we have an infinite array, corresponding to a signal that can take values for all integer n.)

With these operations in hand, one can talk of *weighted linear combinations* of signals. Thus, if $x_1[.]$ and $x_2[.]$ are two possible input signals to a system, for instance the signals associated with experiments numbered 1 and 2, then we can consider an experiment 3 in which the input $x_3[.]$ a weighted linear combination of the inputs in the other two experiments:

$$x_3[n] = a_1 x_1[n] + a_2 x_2[n] \quad \text{for all } n \,,$$

where a_1 and a_2 are scalar constants.

The system is termed *linear* if the response to this weighted linear combination of the two signals is the *same weighted combination of the responses* to the two signals, for all possible choices of $x_1[.]$, $x_2[.]$ a_1 and a_2, i.e., if

$$y_3[n] = a_1 y_1[n] + a_2 y_2[n] \quad \text{for all } n \; ,$$

where $y_i[.]$ denotes the response of the system to input $x_i[.]$ for $i = 1, 2, 3$.

This relationship is shown in Figure 10-8. If this property holds, we say that the results of any two experiments can be *superposed* to yield the results of other experiments; a linear system is said to have the **superposition property**. (In terms of the notion of behaviors defined earlier, what linearity requires is that weighted linear combinations, or superpositions, of behaviors are again behaviors of the system.)

We can revisit the examples introduced in Equations (10.4), (10.5), (10.6) to apply this definition, and recognize that all three systems are linear. The following are examples of systems that are *not* linear:

$$y[n] = x[n] + 3 \; ;$$

$$y[n] = x[n] + x^2[n-1] \; ;$$

$$y[n] = \cos\Big(\frac{x^2[n]}{x^2[n]+1}\Big) \; .$$

All three examples here are time-invariant.

10.2.5 Linear, Time-Invariant (LTI) Models

Models that are both linear and time-invariant, or *LTI models*, are hugely important in engineering and other domains. We will mention some of the reasons in the next chapter. We will develop insights into their behavior and tools for their analysis, and then return to apply what we have learned, to better understand signal transmission on physical channels.

In the context of audio communication using a computer's speaker and microphone, transmissions are done using bursts at the loudspeaker of a computer, and receptions by detecting the response at a microphone. The input $x[n]$ in this case is a baseband signal of the form in Figure 10-3, but alternating regularly between high and low values. This was converted through a modulation process into the tone bursts that you heard. The signal received at the microphone is then demodulated to reconstruct an estimate $y[n]$ of the baseband input.

With the microphone in a fixed position, responses have some consistency from one transition to the next (between tone and no-tone), despite the presence of random fluctuations riding on top of things. The deterministic or repeatable part of the response $y[n]$ does show distortion, i.e., deviation from $x[n]$, though more "real-world" than what is shown in the synthetic example in Figure 10-3. However, when the microphone is very close to the speaker, the distortion is low.

There were features of the system response in this communication system to suggest that it may not be unreasonable to model the baseband acoustic channel as LTI. Time-invariance (at least over the time-horizon of the demo!) is suggested by the repeatability of the transient responses to the various transitions. Linearity is suggested by the fact that the downward transients caused by negative (i.e., downward) steps at the input look like reflections of the upward transients caused by positive (i.e., upward) steps of the same magnitude at the input, and is also suggested by the appropriate scaling of the response when the input is scaled.