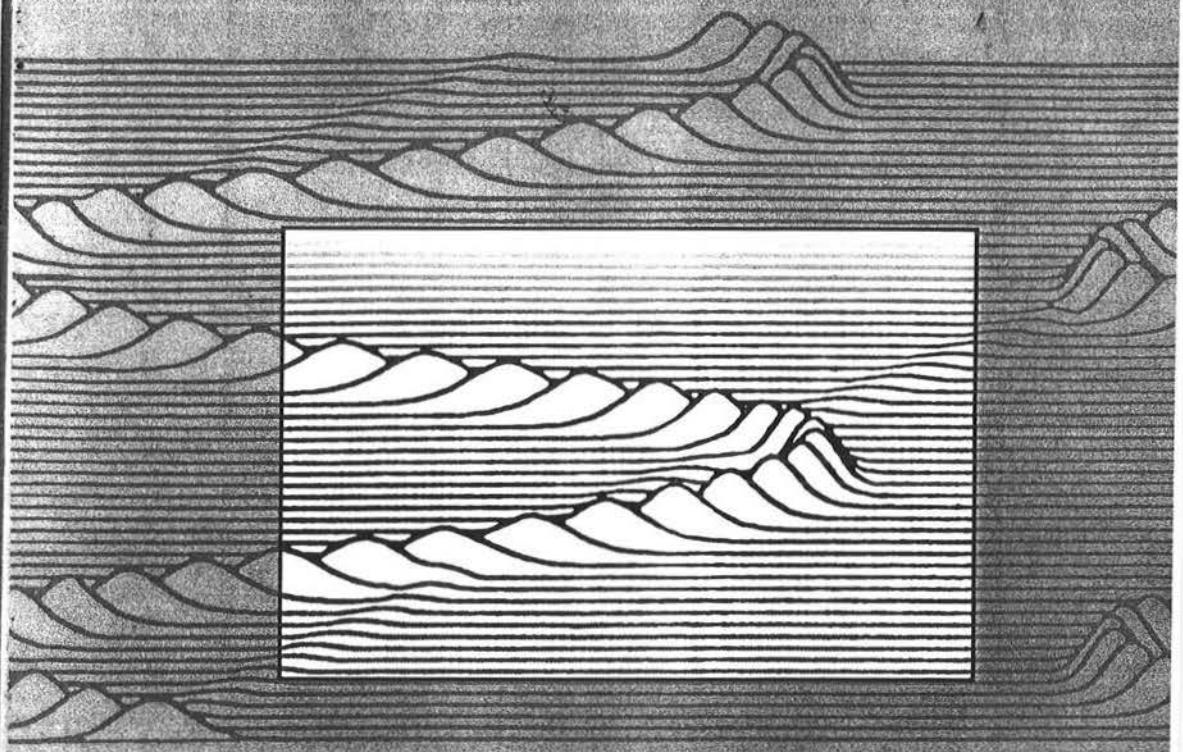
DISCRETE-TIME SIGNAL PROCESSING



Alan V. Oppenheim · Ronald W. Schafer

PRENTICE HALL SIGNAL PROCESSING SERIES
ALAN V. OPPENHEIM, SERIES EDITOR

Library of Congress Cataloging-in-Publication Data

Oppenheim, Alan V.
Discrete-time signal processing / Alan V. Oppenheim, Ronald W. Schafer.
p. cm.—(Prentice Hall signal processing series)
Bibliography: p.
Includes index.
ISBN 0-13-216292-X
1. Signal processing—Mathematics. 2. Discrete-time systems.
I. Schafer, Ronald W. II. Title. III. Series.
TK5102.5.02452 1989
621.38'043—dc 19
88-25562
CIP

Editorial/production supervision: Barbara G. Flanagan
Interior design: Roger Brower
Cover design: Vivian Berman
Manufacturing buyer: Mary Noonan

Published by Prentice-Hall, Inc.
A Division of Simon & Schuster
Englewood Cliffs, New Jersey 07632

Printed in the United States of America
10 9 8 7 6 5

ISBN 0-13-216292-X

Prentice-Hall International (UK) Limited, *London*
Prentice-Hall of Australia Pty. Limited, *Sydney*
Prentice-Hall Canada Inc., *Toronto*
Prentice-Hall Hispanoamericana, S.A., *Mexico*
Prentice-Hall of India Private Limited, *New Delhi*
Prentice-Hall of Japan, Inc., *Tokyo*
Simon & Schuster Asia Pte. Ltd., *Singapore*
Editora Prentice-Hall do Brasil, Ltda., *Rio de Janeiro*

2.2 DISCRETE-TIME SYSTEMS

A discrete-time system is defined mathematically as a transformation or operator that maps an input sequence with values $x[n]$ into an output sequence with values $y[n]$. This can be denoted as

$$y[n] = T\{x[n]\} \tag{2.21}$$

and is indicated pictorially in Fig. 2.6. Equation (2.21) represents a rule or formula for computing the output sequence values from the input sequence values. It should be emphasized that the value of the output sequence at each value of the index n may be a function of $x[n]$ at all values of n. The following examples illustrate some simple and useful systems.

Example 2.1 The Ideal Delay System

The ideal delay system is defined by the equation

$$y[n] = x[n - n_d], \qquad -\infty < n < \infty, \tag{2.22}$$

where n_d is a fixed positive integer called the delay of the system. In words, the ideal delay system simply shifts the input sequence to the right by n_d samples to form the output. If in Eq. (2.22) n_d is a fixed negative integer, then the system would shift the input to the left by n_d samples corresponding to a time advance.

In Example 2.1 only one sample of the input sequence is involved in determining a certain output sample. In the following example, this is not the case.

Example 2.2 Moving Average

The general moving average system is defined by the equation

$$\begin{aligned} y[n] &= \frac{1}{M_1 + M_2 + 1} \sum_{k=-M_1}^{M_2} x[n-k] \\ &= \frac{1}{M_1 + M_2 + 1} \{x[n + M_1] + x[n + M_1 - 1] + \cdots + x[n] \\ &\quad + x[n-1] + \cdots + x[n - M_2]\}. \end{aligned} \tag{2.23}$$

This system computes the nth sample of the output sequence as the average of $(M_1 + M_2 + 1)$ samples of the input sequence around the nth sample. Figure 2.7 shows an input sequence plotted as a function of a dummy index k and the samples involved in the computation of the output sample $y[n]$ for $n = 7$, $M_1 = 0$, and $M_2 = 5$. The output sample $y[7]$ is equal to $\frac{1}{6}$ times the sum of all the samples between the vertical dotted lines. To compute $y[8]$, both dotted lines would move one sample to the right.

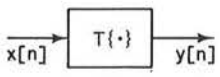


Figure 2.6 Representation of a discrete-time system, i.e., a transformation that maps an input sequence $x[n]$ into a unique output sequence $y[n]$.

$\omega_0 n$ for several different ω_0 increases from zero -d), the sequence apidly. As ω_0 increases 'ts d–a), the oscillations

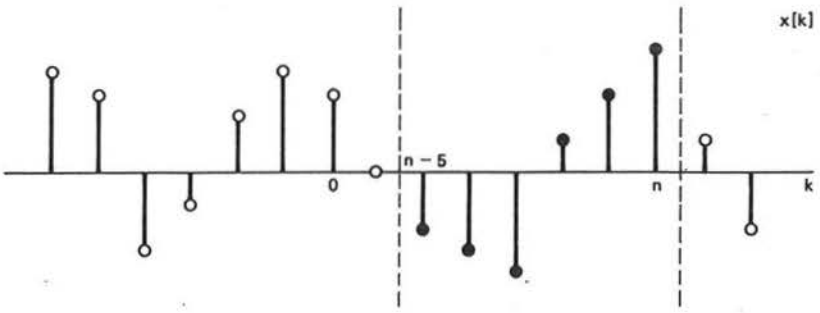


Figure 2.7 Sequence values involved in computing a causal moving average.

Classes of systems are defined by placing constraints on the properties of the transformation $T\{\cdot\}$. Doing so often leads to very general mathematical representations, as we will see. Of particular importance to consider are the system constraints and properties, discussed in Sections 2.2.1–2.2.5.

2.2.1 Memoryless Systems

A system is referred to as memoryless if the output $y[n]$ at every value of n depends only on the input $x[n]$ at the same value of n.

Example 2.3

An example of a memoryless system is one for which $x[n]$ and $y[n]$ are related by

$$y[n] = (x[n])^2 \quad \text{for each value of } n. \tag{2.24}$$

The system in Example 2.1 is not memoryless unless $n_d = 0$. In particular it is referred to as having "memory" whether n_d is positive (time delay) or negative (time advance). The system in Example 2.2 is not memoryless unless $M_1 = M_2 = 0$.

2.2.2 Linear Systems

The class of *linear systems* is defined by the principle of superposition. If $y_1[n]$ and $y_2[n]$ are the responses of a system when $x_1[n]$ and $x_2[n]$ are the respective inputs, then the system is linear if and only if

$$T\{x_1[n] + x_2[n]\} = T\{x_1[n]\} + T\{x_2[n]\} = y_1[n] + y_2[n] \tag{2.25a}$$

and

$$T\{ax[n]\} = aT\{x[n]\} = ay[n]. \tag{2.25b}$$

where a is an arbitrary constant. The first property is called the *additivity property* and the second is called the *homogeneity or scaling property*. These two properties can be combined into the principle of superposition, stated as

$$T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\} \tag{2.26}$$

for arbitrary constants a and b. This can be generalized to the superposition of many inputs. Specifically, if

$$x[n] = \sum_k a_k x_k[n], \tag{2.27a}$$

then

$$y[n] = \sum_k a_k y_k[n], \tag{2.27b}$$

where $y_k[n]$ is the system response to the input $x_k[n]$.

By using the definition of the principle of superposition, we can easily show that the systems of Examples 2.1 and 2.2 are linear systems. (See Problem 2.2.) An example of a nonlinear system is the system in Example 2.3.

Example 2.4

The system referred to as an accumulator and defined by the relation

$$y[n] = \sum_{k=-\infty}^{n} x[k] \tag{2.28}$$

is a linear system. The output at each sample instant n is equal to the accumulated sum of the value at n and all previous values of the input. Comparing Eq. (2.28) and Eq. (2.8), we note that if the input $x[n]$ is a unit impulse $\delta[n]$, then the output $y[n]$ is a unit step $u[n]$.

2.2.3 Time-Invariant Systems

A time-invariant system (equivalently often referred to as a shift-invariant system) is one for which a time shift or delay of the input sequence causes a corresponding shift in the output sequence. Specifically, suppose that a system transforms the input sequence with values $x[n]$ into the output sequence with values $y[n]$. The system is said to be time-invariant if for all n_0 the input sequence with values $x_1[n] = x[n - n_0]$ produces the output sequence with values $y_1[n] = y[n - n_0]$.

All of the systems in Examples 2.1–2.4 are time-invariant. The following example illustrates a system that is not time-invariant.

Example 2.5

The system defined by the relation

$$y[n] = x[Mn], \qquad -\infty < n < \infty, \tag{2.29}$$

with M a positive integer, is called a *compressor*. Specifically, it discards $(M - 1)$ samples out of M; i.e., it creates the output sequence by selecting every Mth sample. It should be clear simply from the description of its operation that this system is not time-invariant. We can see this mathematically by considering the response $y_1[n]$ to the input $x_1[n]$ defined as $x_1[n] = x[n - n_0]$. Then

$$y_1[n] = x_1[Mn] = x[Mn - n_0]. \tag{2.30}$$

But

$$y[n - n_0] = x[M(n - n_0)] \neq y_1[n]. \tag{2.31}$$

With $x_1[n]$ equal to $x[n]$ shifted by n_0, $y_1[n]$ is not $y[n]$ shifted by n_0, so the downsampler is not time-invariant, unless of course $M = 1$.

2.2.4 Causality

A system is causal if for every choice of n_0 the output sequence value at index $n = n_0$ depends only on the input sequence values for $n \leq n_0$. This implies that if $x_1[n] = x_2[n]$ for $n \leq n_0$, then $y_1[n] = y_2[n]$ for $n \leq n_0$. That is, the system is *nonanticipative*. The system of Example 2.1 is causal for $n_d \geq 0$ and is noncausal for $n_d < 0$. The system of Example 2.2 is causal if $-M_1 \geq 0$ and $M_2 \geq 0$; otherwise it is noncausal. The system of Example 2.3 is causal, as is the accumulator of Example 2.4. However, the system of Example 2.5 is noncausal if $M > 1$ since $y[1] = x[M]$. Another noncausal system is given in the following example.

Example 2.6

The *forward difference system* is defined by the relation

$$y[n] = x[n+1] - x[n]. \tag{2.32}$$

Obviously $y[n]$ depends on $x[n+1]$; therefore the system is noncausal. However, the *backward difference system*, defined by

$$y[n] = x[n] - x[n-1], \tag{2.33}$$

is causal.

2.2.5 Stability

A system is stable in the bounded-input bounded-output (BIBO) sense if and only if every bounded input sequence produces a bounded output sequence. The input $x[n]$ is bounded if there exists a fixed positive finite value B_x such that

$$|x[n]| \leq B_x < \infty \qquad \text{for all } n. \tag{2.34}$$

Stability requires that for every bounded input there exists a fixed positive finite value B_y such that

$$|y[n]| \leq B_y < \infty \qquad \text{for all } n. \tag{2.35}$$

Examples 2.1, 2.2, 2.3, and 2.5 are stable systems. The accumulator of Example 2.4 is unstable. To demonstrate this we need only specify one bounded input for which the output is unbounded. Thus, consider $x[n] = u[n]$, which is clearly bounded. For this input to the accumulator, the output is

$$\begin{aligned} y[n] &= \sum_{k=-\infty}^{n} u[k] \\ &= \begin{cases} 0, & n < 0, \\ (n+1), & n \geq 0. \end{cases} \end{aligned} \tag{2.36}$$

Although this response is finite if n is finite, it is nevertheless unbounded, i.e., there is no fixed positive finite value B_y such that $(n+1) \leq B_y < \infty$ for all n.

It is important to emphasize that the properties we have defined in this section are properties of *systems*, not of the inputs to the system. That is, we may be able to

find inputs for which the properties hold, but the existence of the property for some inputs does not mean that the system has the property. For the system to have the property, it must hold for all inputs. For example, an unstable system may have some bounded inputs for which the output is bounded, but for the system to have the property of stability, it must be true that for *all* bounded inputs the output is bounded. If, as we were able to do with the accumulator system of Example 2.4, we can find just one input for which the system property does not hold, then we have shown that the system does *not* have that property.

2.3 LINEAR TIME-INVARIANT SYSTEMS

A particularly important class of systems consists of those that are linear and time-invariant. These two properties in combination lead to especially convenient representations for these systems. Most important, this class of systems has significant signal processing applications. The class of linear systems is defined by the principle of superposition in Eq. (2.26). If the linearity property is combined with the representation of a general sequence as a linear combination of delayed impulses as in Eq. (2.6), it follows that a linear system can be completely characterized by its impulse response. Specifically, let $h_k[n]$ be the response of the system to $\delta[n-k]$, an impulse occurring at $n = k$. Then from Eq. (2.6),

$$y[n] = T\left\{\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]\right\}. \tag{2.37}$$

From the principle of superposition in Eq. (2.26), we can write

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]T\{\delta[n-k]\} = \sum_{k=-\infty}^{\infty} x[k]h_k[n]. \tag{2.38}$$

According to Eq. (2.38), the system response to any input can be expressed in terms of the response of the system to $\delta[n-k]$. If only linearity is imposed, $h_k[n]$ will depend on both n and k, in which case the computational usefulness of Eq. (2.38) is limited. We obtain a more useful result if we impose the additional constraint of time invariance.

The property of time invariance implies that if $h[n]$ is the response to $\delta[n]$, then the response to $\delta[n-k]$ is $h[n-k]$. With this additional constraint, Eq. (2.38) becomes

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]. \tag{2.39}$$

As a consequence of Eq. (2.39), a linear time-invariant system (which we will sometimes abbreviate as LTI) is completely characterized by its impulse response, $h[n]$, in the sense that, given $h[n]$, it is possible to use Eq. (2.39) to compute the output $y[n]$ due to *any* input $x[n]$.

Equation (2.39) is commonly called the *convolution sum.* If $y[n]$ is a sequence whose values are related to the values of two sequences $h[n]$ and $x[n]$ as in Eq. (2.39), we say that $y[n]$ is the convolution of $x[n]$ with $h[n]$ and represent this by the notation

$$y[n] = x[n] * h[n]. \tag{2.40}$$

The operation of discrete-time convolution takes two sequences $x[n]$ and $h[n]$ and produces a third sequence $y[n]$. Equation (2.39) expresses each sample of the output sequence in terms all of the samples of the input and the impulse response sequences.

The derivation of Eq. (2.39) suggests the interpretation that the input sample at $n = k$, represented as $x[k]\delta[n-k]$, is transformed by the system into an output sequence $x[k]h[n-k]$, $-\infty < n < \infty$ and that these sequences for each k are superimposed to form the overall output sequence. This interpretation is illustrated in Fig. 2.8, which shows an impulse response, a simple input sequence having three nonzero samples, the individual outputs due to each sample, and the composite output due to all the samples in the input sequences. Specifically, $x[n]$ can be decomposed as the sum of the three sequences $x[-2]\delta[n+2]$, $x[0]\delta[n]$, and $x[3]\delta[n-3]$ representing the three nonzero values in the sequence $x[n]$. The sequences $x[-2]h[n+2]$, $x[0]h[n]$, and $x[3]h[n-3]$ are the system responses to $x[-2]\delta[n+2]$, $x[0]\delta[n]$, and $x[3]\delta[n-3]$, respectively. The response to $x[n]$ is then the sum of these three individual responses.

Although the convolution-sum expression is analogous to the convolution integral of continuous-time linear system theory, the convolution sum should not be thought of as an approximation to the convolution integral. The convolution integral plays mainly a theoretical role in continuous-time linear system theory; we will see that the convolution sum, in addition to its theoretical importance, often serves as an explicit realization of a discrete-time linear system. Thus it is important to gain some insight into the properties of the convolution sum in actual calculations.

The preceding interpretation of Eq. (2.39) emphasizes that the convolution sum is a direct result of linearity and time invariance. However, a slightly different way of looking at Eq. (2.39) leads to a particularly useful computational interpretation. When viewed as a formula for computing a single value of the output sequence, Eq. (2.39) dictates that $y[n]$ (i.e., the nth value of the output) is obtained by multiplying the input sequence (expressed as a function of k) by the sequence whose values are $h[n-k]$, $-\infty < k < \infty$, and then, for any fixed value of n, summing all the values of the products $x[k]h[n-k]$, with k a counting index in the summation process. The operation of convolving two sequences thus involves doing the computation for all values of n, thus generating the complete output sequence $y[n]$, $-\infty < n < \infty$. The key to carrying out the computations of Eq. (2.39) to obtain $y[n]$ is understanding how to form the sequence $h[n-k]$, $-\infty < k < \infty$, for all values of n that are of interest. To this end, it is useful to note that

$$h[n-k] = h[-(k-n)]. \tag{2.41}$$

The interpretation of Eq. (2.41) is best done with an example.

$n]$ is a sequence
] as in Eq. (2.39),
sent this by the

(2.40)

$n]$ and $h[n]$ and
ple of the output
ponse sequences.
: input sample at
into an output
for each k are
ion is illustrated
nce having three
d the composite
lly, $x[n]$ can be
], $x[0]\delta[n]$, and
uence $x[n]$. The
tem responses to
ponse to $x[n]$ is

the convolution
um should not be
volution integral
eory; we will see
often serves as an
ant to gain some
tions.
convolution sum
y different way of
al interpretation.
ut sequence, Eq.
d by multiplying
whose values are
g all the values of
ion process. The
mputation for all
], $-\infty < n < \infty$.
n $y[n]$ is under-
alues of n that are

(2.41)

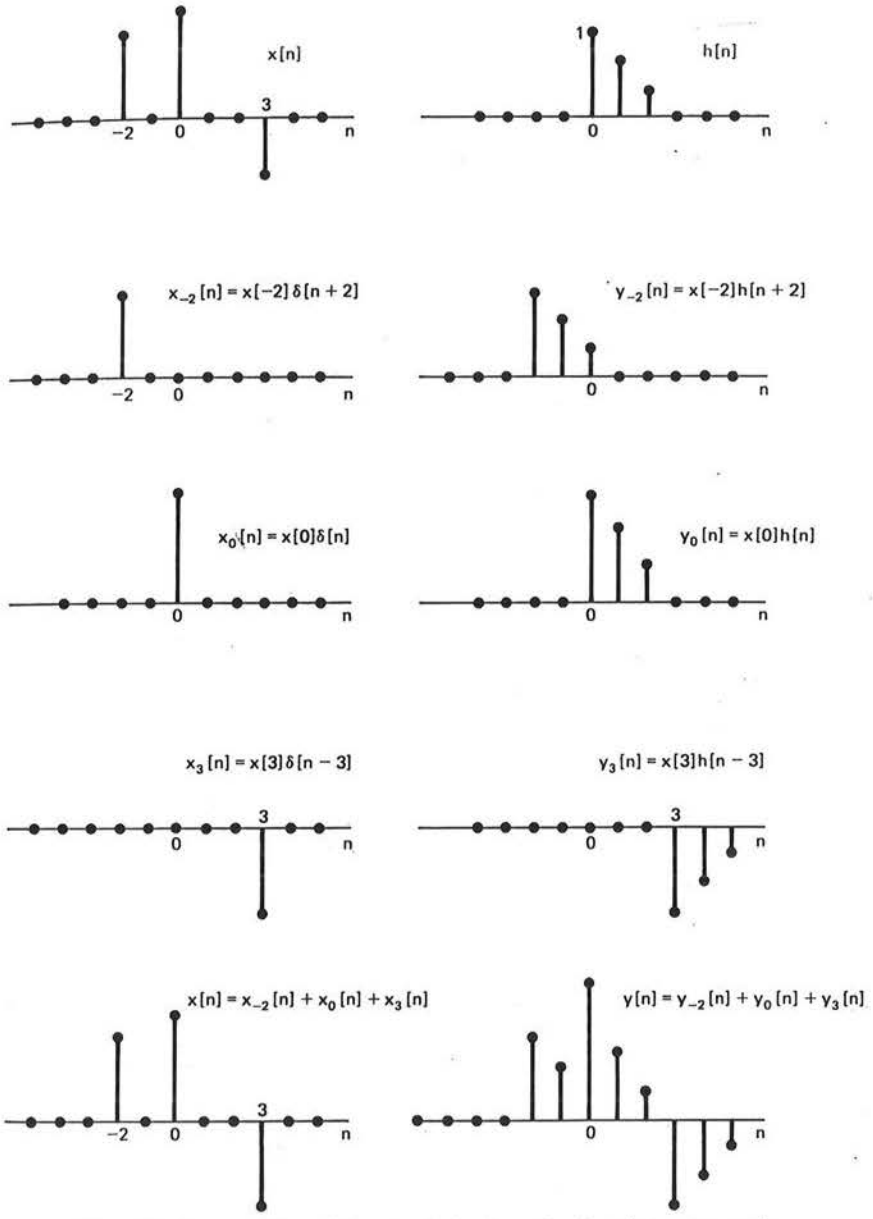


Figure 2.8 Representation of the output of a linear time-invariant system as the superposition of responses to individual samples of the input.

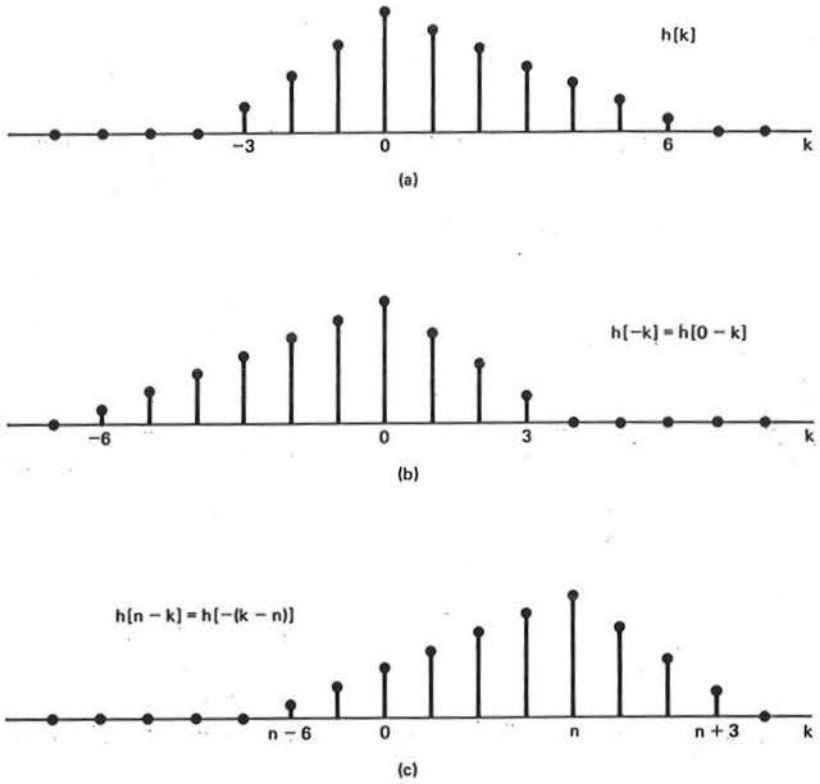


Figure 2.9 Forming the sequence $h[n-k]$. (a) The sequence $h[k]$ as a function of k. (b) The sequence $h[-k]$ as a function of k. (c) The sequence $h[n-k]$ as a function of k for $n = 4$.

Example 2.7

Suppose $h[k]$ is the sequence shown in Fig. 2.9(a). First consider $h[-k]$ plotted against k; $h[-k]$ is simply $h[k]$ reflected or "flipped" about $k = 0$, as shown in Fig. 2.9(b). Replacing k by $k - n$, where n is a fixed integer, leads to a shift of the origin of the sequence $h[-k]$ to $k = n$, as shown in Fig. 2.9(c) for the case $n = 4$.

Generalizing Example 2.7, it should be clear that in general the sequence $h[n-k]$, $-\infty < k < \infty$, is obtained by

1. reflecting $h[k]$ about the origin to obtain $h[-k]$;
2. shifting the origin of the reflected sequence to $k = n$.

To implement discrete-time convolution, the two sequences $x[k]$ and $h[n-k]$ are multiplied together for $-\infty < k < \infty$ and the products are summed to compute the output sample $y[n]$. To obtain another output sample, the origin of the sequence

$h[-k]$ is shifted to the new sample position and the process is repeated. This computational procedure applies whether the computations are carried out numerically on sampled data or analytically with sequences for which the sample values have simple formulas. The following example illustrates discrete-time convolution for the latter case.

Example 2.8

Consider a system with impulse response

$$h[n] = u[n] - u[n-N] = \begin{cases} 1, & 0 \le n \le N-1, \\ 0, & \text{otherwise.} \end{cases}$$

The input is

$$x[n] = a^n u[n].$$

To find the output at a particular index n, we must form the sums over all k of the product $x[k]h[n-k]$. In this case, we can find formulas for $y[n]$ for different sets of values of n. For example, Fig. 2.10(a) shows the sequences $x[k]$ and $h[n-k]$ plotted for n a negative integer. Clearly, all negative values of n give a similar picture; i.e., the nonzero portions of the sequences $x[k]$ and $h[n-k]$ do not overlap, so

$$y[n] = 0, \qquad n < 0.$$

Figure 2.10(b) illustrates the two sequences when $0 \le n$ and $n - N + 1 \le 0$. These two conditions can be combined into the single condition $0 \le n \le N-1$. By considering Fig. 2.10(b), we see that since

$$x[k]h[n-k] = a^k,$$

then

$$y[n] = \sum_{k=0}^{n} a^k \qquad \text{for } 0 \le n \le N-1. \tag{2.42}$$

The limits on the sum are determined directly from Fig. 2.10(b). Equation (2.42) shows that $y[n]$ is the sum of $n+1$ terms of a geometric series where the ratio of terms is a. This sum can be expressed in closed form using the general formula

$$\sum_{k=N_1}^{N_2} \alpha^k = \frac{\alpha^{N_1} - \alpha^{N_2+1}}{1-\alpha}, \qquad N_2 \ge N_1. \tag{2.43}$$

Applying this to Eq. (2.42), we obtain

$$y[n] = \frac{1-a^{n+1}}{1-a}, \qquad 0 \le n \le N-1. \tag{2.44}$$

Finally, Fig. 2.10(c) shows the two sequences when $0 < n - N + 1$ or $N - 1 < n$. As before,

$$x[k]h[n-k] = a^k, \qquad n - N + 1 < k \le n,$$

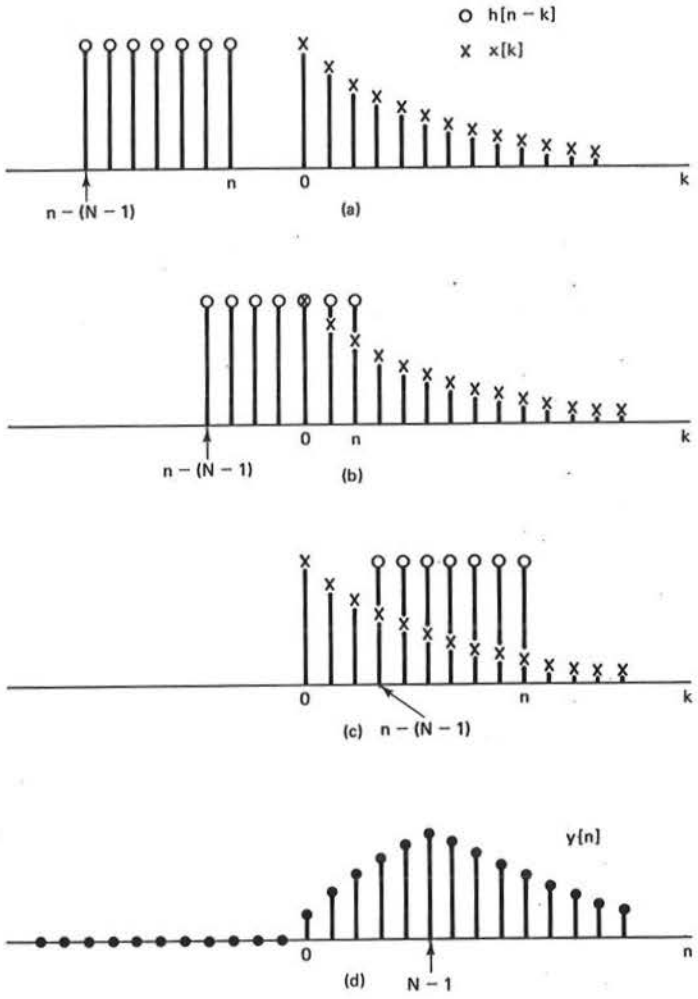


Figure 2.10 Sequences involved in computing a discrete convolution. (a)-(c) The sequences $x[k]$ and $h[n-k]$ as a function of k for different values of n. (Only nonzero samples are shown.) (d) Corresponding output sequence as a function of n.

but now the lower limit on the sum is $n - N + 1$, as seen in Fig. 2.10(c). Thus

$$y[n] = \sum_{k=n-N+1}^{n} a^k \qquad \text{for } N-1 < n. \tag{2.45}$$

Using Eq. (2.43) we obtain

$$y[n] = \frac{a^{n-N+1} - a^{n+1}}{1-a}$$

or

$$y[n] = a^{n-N+1}\left(\frac{1-a^N}{1-a}\right). \tag{2.46}$$

Thus, because of the piecewise-exponential nature of both the input and the unit sample response, we have been able to obtain the following closed-form expression for $y[n]$ as a function of the index n:

$$y[n] = \begin{cases} 0, & n < 0, \\ \dfrac{1-a^{n+1}}{1-a}, & 0 \le n \le N-1, \\ a^{n-N+1}\left(\dfrac{1-a^N}{1-a}\right), & N-1 < n. \end{cases} \tag{2.47}$$

This sequence is shown in Fig. 2.10(d).

Example 2.8 illustrates how the convolution sum can be computed analytically when the input and the impulse response are given by simple formulas. In such cases, the sums may have a compact form that may be derived using the formula for the sum of a geometric series or other "closed-form" formulas.† When no simple form is available, the convolution sum can still be evaluated numerically using the technique illustrated in Example 2.8 whenever the sums are finite, which will be the case if either the input sequence or the impulse response is of finite length, i.e., has a finite number of nonzero samples. We cannot overemphasize that neatly drawn and labeled sketches such as those in Fig. 2.10 are invaluable to ensure that the limits on the sum are correctly determined.

2.4 PROPERTIES OF LINEAR TIME-INVARIANT SYSTEMS

Since all linear time-invariant systems are described by the convolution sum of Eq. (2.39), the properties of this class of systems are defined by the properties of discrete-time convolution. Therefore the impulse response is a complete characterization of the properties of a specific linear time-invariant system.

Some general properties of the class of linear time-invariant systems can be found by considering properties of the convolution operation. For example, the convolution operation is commutative:

$$x[n] * h[n] = h[n] * x[n]. \tag{2.48}$$

This can be shown by applying a substitution of variables to Eq. (2.39). Specifically, with $m = n - k$,

$$y[n] = \sum_{m=-\infty}^{\infty} x[n-m]h[m], \tag{2.49}$$

† Such results are discussed, for example, in Grossman (1982). A very useful book of sums is Jolley (1961).

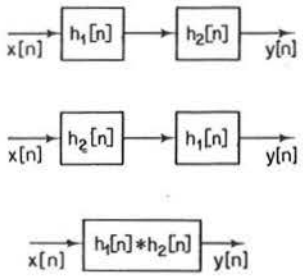


Figure 2.11 Three linear time-invariant systems with identical impulse responses.

so the roles of $x[n]$ and $h[n]$ in the summation are interchanged. That is, the order of the sequences in a convolution is unimportant, and hence the system output is the same if the roles of the input and impulse response are reversed. A linear time-invariant system with input $x[n]$ and impulse response $h[n]$ will have the same output as a linear time-invariant system with input $h[n]$ and impulse response $x[n]$. The convolution operation also distributes over addition, i.e.,

$$x[n] * (h_1[n] + h_2[n]) = x[n] * h_1[n] + x[n] * h_2[n].$$

This follows in a straightforward way from Eq. (2.39).

In a *cascade connection* of systems, the output of the first system is the input to the second, the output of the second is the input to the third, etc. The output of the last system is the overall output. Two linear time-invariant systems in cascade correspond to a linear time-invariant system with an impulse response that is the convolution of the impulse responses of the two systems. This is illustrated in Fig. 2.11. In the upper block diagram, the output of the first system will be $h_1[n]$ if $x[n] = \delta[n]$. Thus the output of the second system (and the overall impulse response of the system) will be

$$h[n] = h_1[n] * h_2[n]. \tag{2.50}$$

As a consequence of the commutative property of convolution, it follows that the impulse response of a cascade combination of linear time-invariant systems is independent of the order in which they are cascaded. This result is summarized in Fig. 2.11, where the three systems all have the same impulse response.

In a *parallel connection*, the systems have the same input and their outputs are summed to produce an overall output. It follows from the distributive property of convolution that two linear time-invariant systems in parallel are equivalent to a single system whose impulse response is the sum of the individual impulse responses, i.e.,

$$h[n] = h_1[n] + h_2[n]. \tag{2.51}$$

This is depicted in Fig. 2.12.

The constraints of linearity and time invariance define a class of systems with very special properties. Stability and causality represent additional properties, and it is often important to know whether a linear time-invariant system is stable and whether it is causal. Recall from Section 2.2.5 that a stable system is one for which

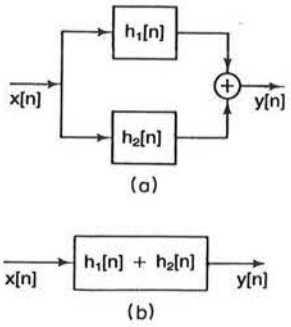


Figure 2.12 (a) Parallel combination of linear time-invariant systems. (b) An equivalent system.

every bounded input produces a bounded ouput. Linear time-invariant systems are stable if and only if the impulse response is absolutely summable, i.e., if

$$S = \sum_{k=-\infty}^{\infty} |h[k]| < \infty. \tag{2.52}$$

This can be shown as follows. From Eq. (2.49),

$$|y[n]| = \left| \sum_{k=-\infty}^{\infty} h[k]x[n-k] \right| \leq \sum_{k=-\infty}^{\infty} |h[k]||x[n-k]|. \tag{2.53}$$

If $x[n]$ is bounded so that

$$|x[n]| \leq B_x,$$

then

$$|y[n]| \leq B_x \sum_{k=-\infty}^{\infty} |h[k]|. \tag{2.54}$$

Thus $y[n]$ is bounded if Eq. (2.52) holds; i.e., Eq. (2.52) is a sufficient condition for stability. To show that it is also a necessary condition, we must show that if $S = \infty$, then a bounded input can be found that will cause an unbounded output. Such an input is the sequence with values

$$x[n] = \begin{cases} \dfrac{h^*[-n]}{|h[-n]|}, & h[n] \neq 0, \\ 0, & h[n] = 0, \end{cases} \tag{2.55}$$

where $h^*[n]$ is the complex conjugate of $h[n]$. The sequence $x[n]$ is clearly bounded by unity. However, the value of the output at $n = 0$ is

$$y[0] = \sum_{k=-\infty}^{\infty} x[-k]h[k] = \sum_{k=-\infty}^{\infty} \frac{|h[k]|^2}{|h[k]|} = S. \tag{2.56}$$

Thus if $S = \infty$, it is possible for a bounded input sequence to produce an unbounded output sequence.

The class of causal systems was defined in Section 2.2.4 as those systems for which the output $y[n_0]$ depends only on the input samples $x[n]$, for $n \leq n_0$. It follows from Eq. (2.39) or Eq. (2.49) that this definition implies the condition

$$h[n] = 0, \qquad n < 0, \tag{2.57}$$

for causality of linear time-invariant systems. (See Problem 2.12.) For this reason it is sometimes convenient to refer to a sequence that is zero for $n < 0$ as a *causal sequence*, meaning that it could be the impulse response of a causal system.

To illustrate how the properties of linear time-invariant systems are reflected in the impulse response, let us reconsider some of the systems defined in Examples 2.1–2.6. First note that only the systems of Examples 2.1, 2.2, 2.4, and 2.6 are linear and time-invariant. Although the impulse response of nonlinear or time-varying systems can be found, it is generally of limited interest since the convolution-sum formula and Eqs. (2.52) and (2.57), expressing stability and causality, do not apply to such systems.

First, let us find the impulse responses of the systems in Examples 2.1, 2.2, 2.4, and 2.6. We can do this by simply computing the response of each system to $\delta[n]$ using the defining relationship for the system. The resulting impulse responses are as follows:

Ideal Delay (Example 2.1)

$$h[n] = \delta[n - n_d], \qquad n_d \text{ a positive fixed integer} \tag{2.58}$$

Moving Average (Example 2.2)

$$\begin{aligned} h[n] &= \frac{1}{M_1 + M_2 + 1} \sum_{k=-M_1}^{M_2} \delta[n-k] \\ &= \begin{cases} \dfrac{1}{M_1 + M_2 + 1}, & -M_1 \leq n \leq M_2 \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{2.59}$$

Accumulator (Example 2.4)

$$\begin{aligned} h[n] &= \sum_{k=-\infty}^{n} \delta[k] \\ &= \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \\ &= u[n] \end{aligned} \tag{2.60}$$

Forward Difference (Example 2.6)

$$h[n] = \delta[n+1] - \delta[n] \tag{2.61}$$

Backward Difference (Example 2.6)

$$h[n] = \delta[n] - \delta[n-1] \tag{2.62}$$

Next let us test the stability of each system by computing the sum

$$S = \sum_{n=-\infty}^{\infty} |h[n]|.$$

For the ideal delay, moving average, forward difference, and backward difference examples, it is clear that $S < \infty$ since the impulse response has only a finite number of nonzero samples. Such systems are called *finite-duration impulse response* (FIR) systems. Clearly FIR systems will always be stable as long as each of the impulse response values is finite in magnitude. The accumulator, however, is unstable because

$$S = \sum_{n=0}^{\infty} u[n] = \infty.$$

In Section 2.2.5 we also demonstrated the instability of the accumulator by giving an example of a bounded input (the unit step) for which the output is unbounded.

The impulse response of the accumulator is infinite in duration. This is an example of the class of systems referred to as *infinite-duration impulse response* (IIR) systems. An example of an IIR system that is stable is one having the impulse response $h[n] = a^n u[n]$ with $|a| < 1$. In this case

$$S = \sum_{n=0}^{\infty} |a|^n. \tag{2.63}$$

If $|a| < 1$, the formula for the sum of the terms of an infinite geometric series gives

$$S = \frac{1}{1-|a|} < \infty. \tag{2.64}$$

If, on the other hand, $|a| \geq 1$, the sum is infinite and the system is unstable.

To test causality of the linear time-invariant systems in Examples 2.1, 2.2, 2.4, and 2.6, we can check to see if $h[n] = 0$ for $n < 0$. As discussed in Section 2.2.4, the ideal delay ($n_d \geq 0$ in Eq. 2.22) is causal. If $n_d < 0$, the system is noncausal. For the moving average, causality requires that $-M_1 \geq 0$ and $M_2 \geq 0$. The accumulator and backward difference systems are causal, and the forward difference system is noncausal.

The property of convolution can be used to describe and analyze interconnections of linear time-invariant systems. Consider the system of Fig. 2.13(a), which consists of a forward difference system cascaded with an ideal delay of one sample. According to the commutative property of convolution, the order in which systems are cascaded does not matter as long as they are linear and time-invariant. Therefore we obtain the same result when we compute the forward difference of a sequence and delay the result (Fig. 2.13a) as when we delay the sequence first and then compute the forward difference (Fig. 2.13b). Also, it follows from Eq. (2.50) that the

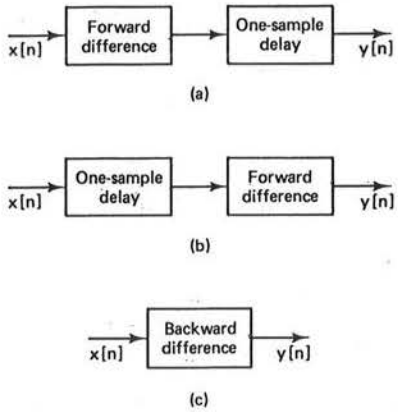


Figure 2.13 Equivalent systems found by using the commutative property of convolution.

overall impulse response of each cascade system is the convolution of the individual impulse responses. Therefore

$$\begin{aligned} h[n] &= (\delta[n+1] - \delta[n]) * \delta[n-1] \\ &= \delta[n-1] * (\delta[n+1] - \delta[n]) \\ &= \delta[n] - \delta[n-1]. \end{aligned} \tag{2.65}$$

Thus $h[n]$ is identical to the impulse response of the backward difference system, i.e., the cascaded systems of Figs. 2.13(a) and 2.13(b) can be replaced by a backward difference system, as shown in Fig. 2.13(c).

Note that the noncausal forward difference systems in Figs. 2.13(a) and (b) have been converted to causal systems by cascading them with a delay. In general, any noncausal FIR system can be made causal by cascading it with a sufficiently long delay.

Another example of cascaded systems introduces the concept of an *inverse system.* Consider the cascade of systems in Fig. 2.14. The impulse response of the cascade system is

$$\begin{aligned} h[n] &= u[n] * (\delta[n] - \delta[n-1]) \\ &= u[n] - u[n-1] \\ &= \delta[n]. \end{aligned} \tag{2.66}$$

That is, the cascade combination of an accumulator followed by a backward difference (or vice versa) yields a system whose overall impulse response is the impulse. Thus the output of the cascade system will always be equal to the input since

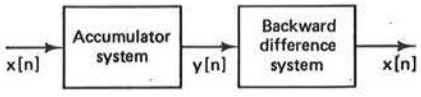


Figure 2.14 An accumulator in cascade with a backward difference. Since the backward difference is the inverse system for the accumulator, the cascade combination is equivalent to the identity system.

$x[n] * \delta[n] = x[n]$. In this case, the backward difference system compensates exactly for (or inverts) the effect of the accumulator, i.e., the backward difference system is the *inverse system* for the accumulator. From the commutative property of convolution, the accumulator is likewise the inverse system for the backward difference system. Also note that this example provides a system interpretation of Eqs. (2.8) and (2.10). In general, if a linear time-invariant system has impulse response $h[n]$, then its inverse system, if it exists, has impulse response $h_i[n]$ defined by the relation

$$h[n] * h_i[n] = h_i[n] * h[n] = \delta[n]. \tag{2.67}$$

Inverse systems are useful in many situations where it is necessary to compensate for the effects of a linear system. In general, it is difficult to solve Eq. (2.67) directly for $h_i[n]$ given $h[n]$. However, in Chapter 5 we will see that the Fourier transform and more generally the z-transform provide a straightforward method of finding an inverse system.

2.5 LINEAR CONSTANT-COEFFICIENT DIFFERENCE EQUATIONS

An important subclass of linear time-invariant systems consists of those systems for which the input $x[n]$ and the output $y[n]$ satisfy an Nth-order linear constant-coefficient difference equation of the form

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k]. \tag{2.68}$$

The properties discussed in Section 2.4 and some of the analysis techniques introduced there can be used to find difference equation representations for some of the linear time-invariant systems that we have defined.

Example 2.9

An example of the class of linear constant-coefficient difference equations is the accumulator (Example 2.4). To show that its input and output satisfy such a difference equation, consider Fig. 2.14 again. If $y[n]$ represents the output of the accumulator, then we have seen that the backward difference of $y[n]$ is equal to $x[n]$ since the backward difference system is the inverse of the accumulator; i.e.,

$$y[n] - y[n-1] = x[n]. \tag{2.69}$$

Thus we have shown that in addition to the defining relationship of Eq. (2.28), which is in the form of a convolution, the input and output also satisfy a linear constant-coefficient difference equation of the form of Eq. (2.68), where $N = 1$, $a_0 = 1$, $a_1 = -1$, $M = 0$, and $b_0 = 1$.

The difference equation in Eq. (2.69) as a representation of the accumulator can be better understood by rewriting it as

$$y[n] = y[n-1] + x[n]. \tag{2.70}$$