ELEC 631 Digital Filters: II

Review of Prerequisite Material
Part I

Victoria, BC, Canada
Email: aantoniou@ieee.org

September 10, 2010

Part I:

- What is a digital filter?
- Characterization

Review of the Prerequisite Material

Part I:

- What is a digital filter?
- Characterization
- Network Representation
- Convolution Summation

Review of the Prerequisite Material

Part I:

- What is a digital filter?
- Characterization
- Network Representation
- Convolution Summation
- The z Transform

Part II:

- The Transfer Function
- Time-Domain Analysis

Part II:

- The Transfer Function
- Time-Domain Analysis
- Frequency-Domain Analysis
- Amplitude and Delay Distortion

Review of the Prerequisite Material *Cont'd*

Part III:

- Spectral Interrelations
- The Sampling Theorem
- Processing of Continuous-Time Signals by Digital Filters

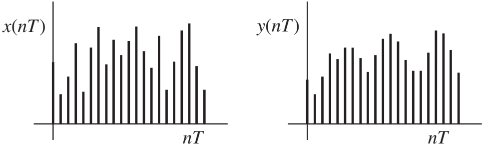
Review of the Prerequisite Material *Cont'd*

Part III:

- Spectral Interrelations
- The Sampling Theorem
- Processing of Continuous-Time Signals by Digital Filters
- Introduction to the Design Process
- The Realization Process

What is a digital filter?

- In its most general form, a digital filter is a system that will receive an input (excitation) in the form of a discrete-time signal and produce an output (response) again in the form of a discrete-time signal.



What is a digital filter? *Cont'd*

- There are many types of discrete-time systems, for example, digital control systems, encoders, and decoders.

What is a digital filter? *Cont'd*

- There are many types of discrete-time systems, for example, digital control systems, encoders, and decoders.
- What differentiates digital filters from other discrete-time systems is the nature of the processing involved.

What is a digital filter? *Cont'd*

- There are many types of discrete-time systems, for example, digital control systems, encoders, and decoders.
- What differentiates digital filters from other discrete-time systems is the nature of the processing involved.
- In digital filters, there is a fundamental requirement that the spectrum of the output signal be related to that of the input by some rule of correspondence.

- For example, if the input signal of a digital filter has a spectrum over the frequency range 0 to $\omega_{\max}$ rad/s, the output of the digital filter may consist of all the components of the input signal with frequencies in some range

$$\omega_{p1} \leq \omega \leq \omega_{p2}$$

where $\omega_{p2} < \omega_{\max}$.

What is a digital filter? *Cont'd*

- For example, if the input signal of a digital filter has a spectrum over the frequency range 0 to $\omega_{\max}$ rad/s, the output of the digital filter may consist of all the components of the input signal with frequencies in some range

$$\omega_{p1} \leq \omega \leq \omega_{p2}$$

  where $\omega_{p2} < \omega_{\max}$.

- Such a digital filter is said to be a *bandpass* filter.

What is a digital filter? *Cont'd*

- For example, if the input signal of a digital filter has a spectrum over the frequency range 0 to $\omega_{\max}$ rad/s, the output of the digital filter may consist of all the components of the input signal with frequencies in some range

$$\omega_{p1} \leq \omega \leq \omega_{p2}$$

  where $\omega_{p2} < \omega_{\max}$.

- Such a digital filter is said to be a *bandpass* filter.
- The frequencies ω_{p1} and ω_{p2} are called the *lower and upper passband edges.*

What is a digital filter? *Cont'd*

- For example, if the input signal of a digital filter has a spectrum over the frequency range 0 to $\omega_{\max}$ rad/s, the output of the digital filter may consist of all the components of the input signal with frequencies in some range

$$\omega_{p1} \leq \omega \leq \omega_{p2}$$

  where $\omega_{p2} < \omega_{\max}$.

- Such a digital filter is said to be a *bandpass* filter.
- The frequencies ω_{p1} and ω_{p2} are called the *lower and upper passband edges.*
- The frequency range ω_{p1} to ω_{p2} is said to be the *passband* of the filter.

Characterization

- There are two types of digital filters:
  - Nonrecursive
  - Recursive

Characterization *Cont'd*

- A linear, time-invariant, causal, nonrecursive filter is characterized by a difference equation of the form

$$y(nT) = \sum_{i=0}^{N} a_i x(nT - iT)$$

where coefficients a_i for 0, 1, ..., N are constants and integer N is the *order* of the filter.

Characterization *Cont'd*

- A linear, time-invariant, causal, nonrecursive filter is characterized by a difference equation of the form

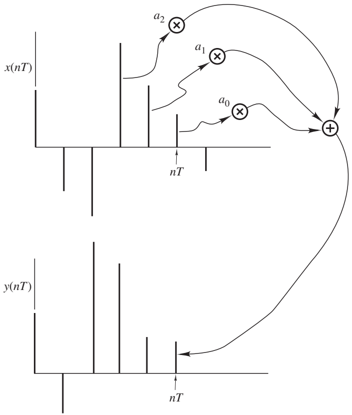
$$y(nT) = \sum_{i=0}^{N} a_i x(nT - iT)$$

  where coefficients a_i for 0, 1, ..., N are constants and integer N is the *order* of the filter.

- In effect, the output values is a linear combination of the input values.

Characterization *Cont'd*

- The operation of a second-order nonrecursive filter is illustrated in the figure.



- An arbitrary linear, time-invariant, causal, recursive digital filter can be represented by the Nth-order difference equation

$$y(nT) = \sum_{i=0}^{N} a_i x(nT - iT) - \sum_{i=1}^{N} b_i y(nT - iT)$$

where a_i and b_i are constant coefficients and integer N is the filter order.

Characterization *Cont'd*

- An arbitrary linear, time-invariant, causal, recursive digital filter can be represented by the Nth-order difference equation

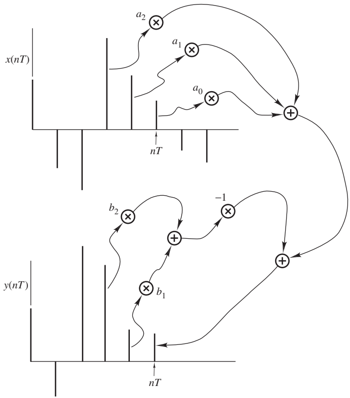
$$y(nT) = \sum_{i=0}^{N} a_i x(nT - iT) - \sum_{i=1}^{N} b_i y(nT - iT)$$

where a_i and b_i are constant coefficients and integer N is the filter order.

- Evidently, the output at instant nT depends on previous values of the output and, in effect, the above difference equation is a recursive equation.

Characterization *Cont'd*

- The operation of a second-order recursive filter is illustrated in the figure.



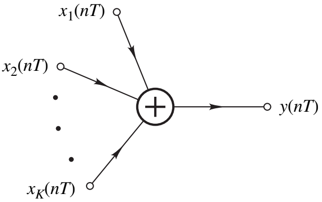
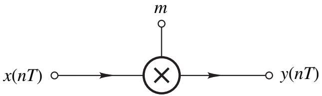
Network Representation

The basic elements of digital filters are

- the unit delay,
- the adder, and
- the multiplier

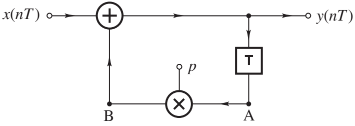
Network Representation *Cont'd*

Table 4.1 Elements of discrete-time systems

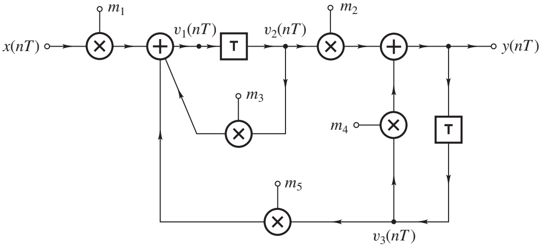
Element	Symbol	Equation
Unit delay		$y(nT) = x(nT-T)$
Adder		$y(nT) = \sum_{i=1}^{K} x_i(nT)$
Multiplier		$y(nT) = mx(nT)$

Network Representation *Cont'd*

- Digital filters can be represented by networks which are collections of interconnected unit delays, adders, and multipliers:

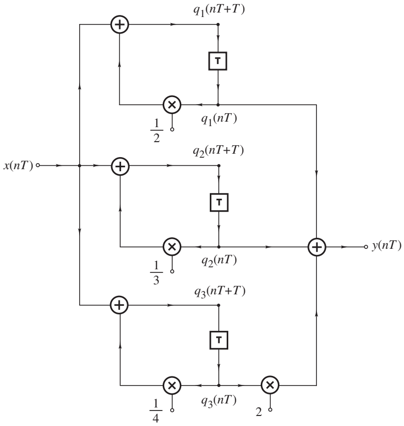


(a)



(c)

- A third-order digital filter made up of three first-order filters connected in parallel:



Network Representation *Cont'd*

The network of a digital filter can be analyzed to find the difference equation of the filter.

Several analysis methods are available:

- By constructing a system of simultaneous difference equations and then solving the system for the response of the filter as a function of the excitation.

The network of a digital filter can be analyzed to find the difference equation of the filter.

Several analysis methods are available:

- By constructing a system of simultaneous difference equations and then solving the system for the response of the filter as a function of the excitation.
- By using signal flow graphs.

The network of a digital filter can be analyzed to find the difference equation of the filter.

Several analysis methods are available:

- By constructing a system of simultaneous difference equations and then solving the system for the response of the filter as a function of the excitation.
- By using signal flow graphs.
- By applying state-space techniques.

- A third-order recursive filter can be represented by a signal flow graph as shown.

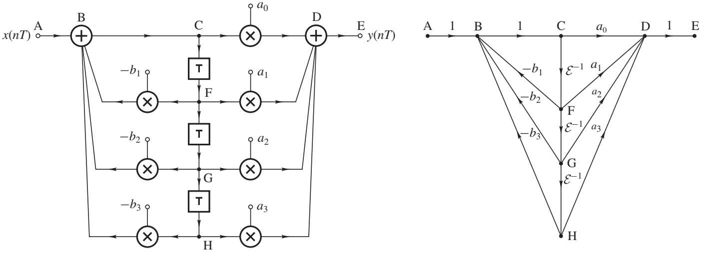


Fig. 6*a*　　Fig. 6*b*

- A third-order recursive filter can be represented by a signal flow graph as shown.
- The difference equation of the filter can be deduced by applying a technique known as *node elimination*.

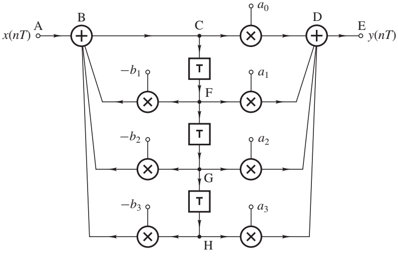


Fig. 6*a*

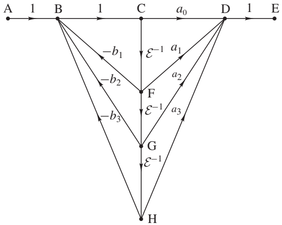


Fig. 6*b*

Network Representation *Cont'd*

- Eliminate node H in Fig. 6a, combine parallel branches in Fig. 8a, and eliminate node G in Fig. 8b:

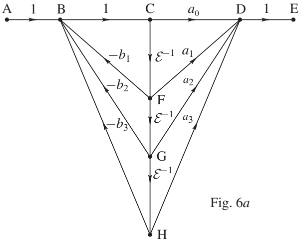


Fig. 6*a*

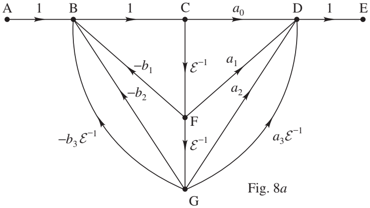


Fig. 8*a*

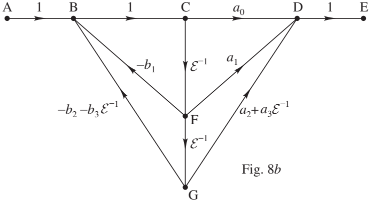


Fig. 8*b*

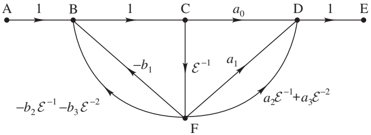


Fig. 8*c*

Network Representation *Cont'd*

- Combine parallel branches in Fig. 8c, eliminate node F in Fig. 8d and node C in Fig. 8e:

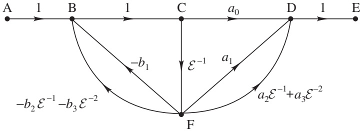


Fig. 8*c*

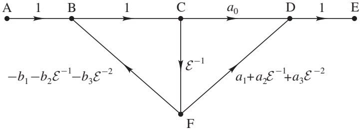


Fig. 8*d*

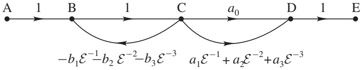


Fig. 8*e*

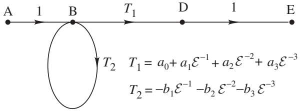


Fig. 8*f*

Network Representation *Cont'd*

- Eliminate self-loop and node D in Fig. 8f and node B in Fig. 8g:

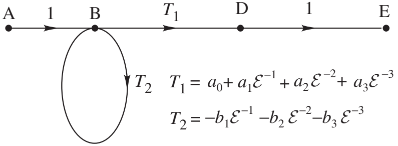


Fig. 8*f*

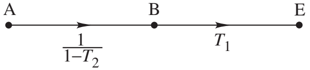


Fig. 8*g*

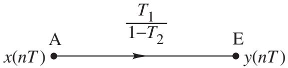


Fig. 8*h*

Network Representation *Cont'd*

- The difference equation can be obtained as

$$\begin{aligned}y(nT) = {} & a_0x(nT) + a_1x(nT - T) + a_2x(nT - 2T) + a_3x(nT - 3T)\\ & -b_1y(nT - T) - b_2y(nT - 2T) - b_3y(nT - 3T)\end{aligned}$$

Convolution Summation

- The impulse response of a digital filter, designated as $h(nT)$, is the response produced by the unit-impulse which is defined as

$$\delta(nT) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$

Convolution Summation

- The impulse response of a digital filter, designated as $h(nT)$, is the response produced by the unit-impulse which is defined as

$$\delta(nT) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$

- If the impulse response of a linear and time-invariant digital filter is known, then the response, $y(nT)$, produced by an arbitrary excitation, $x(nT)$, can be deduced as

$$y(nT) = \sum_{k=-\infty}^{\infty} x(kT)h(nT - kT) = \sum_{k=-\infty}^{\infty} h(kT)x(nT - kT)$$

Convolution Summation

- The impulse response of a digital filter, designated as $h(nT)$, is the response produced by the unit-impulse which is defined as

$$\delta(nT) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$

- If the impulse response of a linear and time-invariant digital filter is known, then the response, $y(nT)$, produced by an arbitrary excitation, $x(nT)$, can be deduced as

$$y(nT) = \sum_{k=-\infty}^{\infty} x(kT)h(nT - kT) = \sum_{k=-\infty}^{\infty} h(kT)x(nT - kT)$$

- This most important relationship is said to be the *convolution summation*.

Convolution Summation *Cont'd*

- If the filter is causal, $h(nT) = 0$ for $n < 0$ and so

$$y(nT) = \sum_{k=-\infty}^{n} x(kT)h(nT - kT) = \sum_{k=0}^{\infty} h(kT)x(nT - kT)$$

If, in addition, $x(nT) = 0$ for $n < 0$ we have

$$y(nT) = \sum_{k=0}^{n} x(kT)h(nT - kT) = \sum_{k=0}^{n} h(kT)x(nT - kT)$$

Convolution Summation *Cont'd*

- If the impulse response of a digital filter is of finite duration such that $h(nT) = 0$ for $n > N$,

$$y(nT) = \sum_{k=0}^{N} h(kT)x(nT - kT)$$

Convolution Summation *Cont'd*

- If the impulse response of a digital filter is of finite duration such that $h(nT) = 0$ for $n > N$,

$$y(nT) = \sum_{k=0}^{N} h(kT)x(nT - kT)$$

- This is essentially an N-order difference equation representing a nonrecursive filter, i.e.,, *if the impulse response is of finite duration, the filter is a nonrecursive one.*

  For this reason, nonrecursive filters are often referred to as finite-duration impulse response (FIR) filters.

Convolution Summation *Cont'd*

- If the impulse response of a digital filter is of finite duration such that $h(nT) = 0$ for $n > N$,

$$y(nT) = \sum_{k=0}^{N} h(kT)x(nT - kT)$$

- This is essentially an N-order difference equation representing a nonrecursive filter, i.e.,, *if the impulse response is of finite duration, the filter is a nonrecursive one.*

  For this reason, nonrecursive filters are often referred to as finite-duration impulse response (FIR) filters.

- On the other hand, the impulse response of recursive filters is of infinite duration and, consequently, they are also referred to as *infinite-duration impulse response* (IIR) filters.

- Earlier on, the difference equation for an Nth-order, linear, time-invariant, causal, nonrecursive filter was specified as

$$y(nT) = \sum_{i=0}^{N} a_i x(nT - iT)$$

Convolution Summation *Cont'd*

- Earlier on, the difference equation for an Nth-order, linear, time-invariant, causal, nonrecursive filter was specified as

$$y(nT) = \sum_{i=0}^{N} a_i x(nT - iT)$$

- From the convolution summation

$$y(nT) = \sum_{i=0}^{N} h(iT) x(nT - iT)$$

Convolution Summation *Cont'd*

- Earlier on, the difference equation for an Nth-order, linear, time-invariant, causal, nonrecursive filter was specified as

$$y(nT) = \sum_{i=0}^{N} a_i x(nT - iT)$$

- From the convolution summation

$$y(nT) = \sum_{i=0}^{N} h(iT)x(nT - iT)$$

- Therefore, *the coefficients of the difference equation are numerically equal to the values of the impulse response*, i.e.,

$$a_i = h(iT) \quad \text{for} \quad 0 \leq i \leq N$$

- A digital filter is said to be *stable* if and only if any bounded excitation results in a bounded response, i.e., if

$$|x(nT)| < \infty \quad \text{for all } n$$

then

$$|y(nT)| < \infty \quad \text{for all } n$$

Convolution Summation *Cont'd*

- A digital filter is said to be *stable* if and only if any bounded excitation results in a bounded response, i.e., if

$$|x(nT)| < \infty \quad \text{for all } n$$

then

$$|y(nT)| < \infty \quad \text{for all } n$$

- Through the use of the convolution summation, a necessary and sufficient condition for stability can be obtained as

$$\sum_{k=-\infty}^{\infty} |h(kT)| < \infty$$

that is, *the impulse response of a stable digital filter is absolutely summable*; conversely, *if the impulse response is absolutely summable, then the filter is stable*.

The z Transform

- The analysis of linear, time-invariant digital filters is almost invariably carried out by using the *z transform*.

The z Transform

- The analysis of linear, time-invariant digital filters is almost invariably carried out by using the *z transform*.
- The principal reason for this is that upon the application of the z transform, the *difference equations* characterizing digital filters *are transformed into algebraic equations* which are usually much easier to manipulate and solve.

The Z Transform

- Consider a bounded discrete-time signal $x(nT)$ that satisfies the conditions

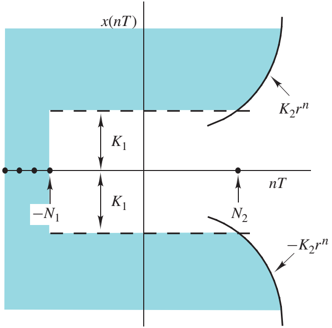
$$\begin{array}{lll} \text{(i)} & x(nT) = 0 & \text{for} \quad n < -N_1 \\ \text{(ii)} & |x(nT)| \leq K_1 & \text{for} \quad -N_1 \leq n < N_2 \\ \text{(iii)} & |x(nT)| \leq K_2 r^n & \text{for} \quad n \geq N_2 \end{array}$$

where N_1 and N_2 are positive integers and r is a positive constant.

The Z Transform *Cont'd*

- ...

$$
\begin{array}{lll}
\text{(i)} & x(nT) = 0 & \text{for} \quad n < -N_1 \\
\text{(ii)} & |x(nT)| \leq K_1 & \text{for} \quad -N_1 \leq n < N_2 \\
\text{(iii)} & |x(nT)| \leq K_2 r^n & \text{for} \quad n \geq N_2
\end{array}
$$



(*a*)

- The z transform of a discrete-time signal $x(nT)$ is defined as

$$X(z) = \sum_{n=-\infty}^{\infty} x(nT) z^{-n}$$

where $z = re^{j\phi}$ is a complex variable.

- The z transform of a discrete-time signal $x(nT)$ is defined as

$$X(z) = \sum_{n=-\infty}^{\infty} x(nT) z^{-n}$$

where $z = re^{j\phi}$ is a complex variable.

- The z transform is defined for all z for which $X(z)$ converges and is often represented by the simplified notation

$$X(z) = \mathcal{Z}x(nT)$$

The z Transform *Cont'd*

- Although the z transform of a signal $x(nT)$ is an infinite series, in practice it can usually be represented in terms of a rational function as

$$\begin{aligned} X(z) &= \sum_{n=-\infty}^{\infty} x(nT)z^{-n} \\ &= \frac{N(z)}{D(z)} = \frac{\sum_{i=0}^{M} a_i z^{M-i}}{z^N + \sum_{i=1}^{N} b_i z^{N-i}} = H_0 \frac{\prod_{i=1}^{M}(z - z_i)}{\prod_{i=1}^{N}(z - p_i)} \end{aligned}$$

where z_i and p_i are the zeros and poles of the z transform and H_0 is a multiplier constant.

The z Transform *Cont'd*

- Although the z transform of a signal $x(nT)$ is an infinite series, in practice it can usually be represented in terms of a rational function as

$$X(z) = \sum_{n=-\infty}^{\infty} x(nT)z^{-n}$$

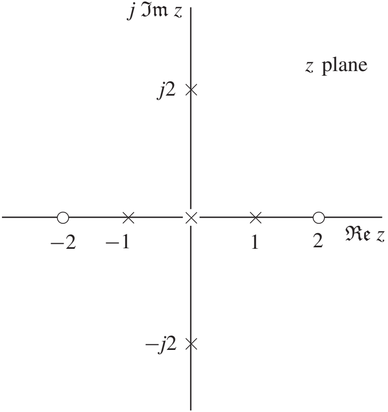
$$= \frac{N(z)}{D(z)} = \frac{\sum_{i=0}^{M} a_i z^{M-i}}{z^N + \sum_{i=1}^{N} b_i z^{N-i}} = H_0 \frac{\prod_{i=1}^{M}(z - z_i)}{\prod_{i=1}^{N}(z - p_i)}$$

where z_i and p_i are the zeros and poles of the z transform and H_0 is a multiplier constant.

- In effect, z transforms can be represented by *zero-pole plots*.

The following z transform has the zero-pole plot shown.

$$X(z) = \frac{(z^2-4)}{z(z^2-1)(z^2+4)} = \frac{(z-2)(z+2)}{z(z-1)(z+1)(z-j2)(z+j2)}$$



Theorem 3.1 Absolute Convergence

If

$$
\begin{array}{lrl}
\text{(i)} & x(nT) = 0 & \text{for} \quad n < -N_1 \\
\text{(ii)} & |x(nT)| \leq K_1 & \text{for} \quad -N_1 \leq n < N_2 \\
\text{(iii)} & |x(nT)| \leq K_2 r^n & \text{for} \quad n \geq N_2
\end{array}
$$

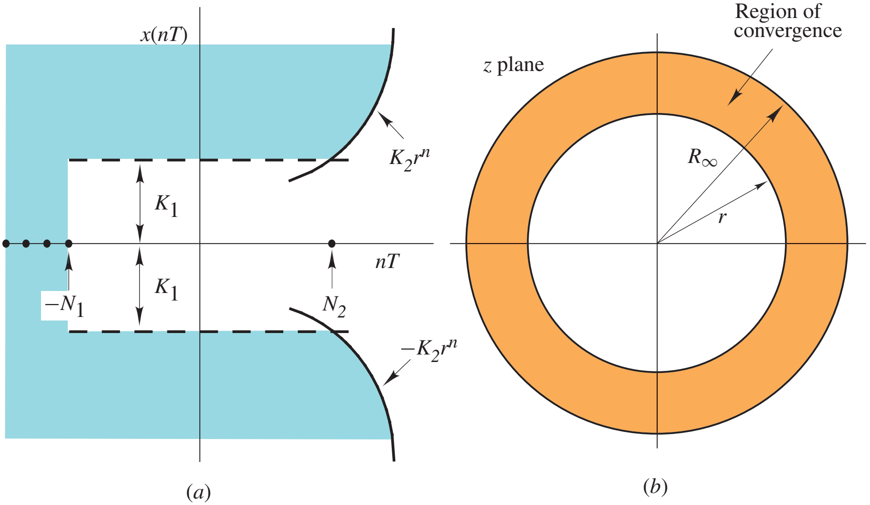
where N_1 and N_2 are positive constants and r is the smallest positive constant that will satisfy condition (iii), then the z transform of $x(nT)$, i.e.,

$$X(z) = \sum_{n=-\infty}^{\infty} x(nT) z^{-n}$$

exists and converges absolutely if and only if

$$r < |z| < R_\infty \quad \text{with} \quad R_\infty \to \infty$$

Absolute Convergence *Cont'd*



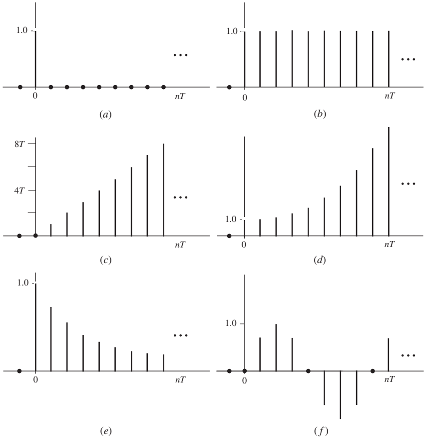
The z Transform – Properties

Property	*Relation*
Linearity	$\mathcal{Z}[ax(nT)+by(nT)] = aX(z)+bY(z)$
Translation	$\mathcal{Z}x(nT+mT) = z^m X(z)$
Complex Scale Change	$\mathcal{Z}[w^{-n}x(nT)] = X(wz)$
Complex Differentiation	$\mathcal{Z}[nTx(nT)] = -Tz\frac{dX(z)}{dz}$
Real Convolution	$\mathcal{Z}\sum_{k=-\infty}^{\infty} h(kT)x(nT-kT) = H(z)X(z)$
	$\mathcal{Z}\sum_{k=-\infty}^{\infty} h(nT-kT)x(kT) = H(z)X(z)$
Complex Convolution	$Y(z) = \mathcal{Z}[h(nT)x(nT)] = \frac{1}{2\pi j}\oint_{\Gamma_1} H(v)X\left(\frac{z}{v}\right)v^{-1}dv$
	$Y(z) = \mathcal{Z}[h(nT)x(nT)] = \frac{1}{2\pi j}\oint_{\Gamma_2} H\left(\frac{z}{v}\right)X(v)v^{-1}dv$

The z Transform *Cont'd*

Function	Definition
Unit impulse	$\delta(nT) = \begin{cases} 1 & \text{for } n = 0 \\ 0 & \text{for } n \neq 0 \end{cases}$
Unit step	$u(nT) = \begin{cases} 1 & \text{for } n \geq 0 \\ 0 & \text{for } n < 0 \end{cases}$
Unit ramp	$r(nT) = \begin{cases} nT & \text{for } n \geq 0 \\ 0 & \text{for } n < 0 \end{cases}$
Exponential	$u(nT)e^{\alpha nT}, \ (\alpha > 0)$
Exponential	$u(nT)e^{\alpha nT}, \ (\alpha < 0)$
Sinusoid	$u(nT)\sin \omega nT$

The z Transform *Cont'd*



(a) Unit impulse, (b) unit step, (c) unit ramp, (d) increasing exponential (e) decreasing exponential, (c) sinusoid.

Standard Z transforms

$x(nT)$	$X(z)$
$\delta(nT)$	1
$u(nT)$	$\dfrac{z}{z-1}$
$u(nT-kT)K$	$\dfrac{Kz^{-(k-1)}}{z-1}$
$u(nT)Kw^n$	$\dfrac{Kz}{z-w}$
$u(nT-kT)Kw^{n-1}$	$\dfrac{K(z/w)^{-(k-1)}}{z-w}$
$u(nT)e^{-\alpha nT}$	$\dfrac{z}{z-e^{-\alpha T}}$
$r(nT)$	$\dfrac{Tz}{(z-1)^2}$

Standard Z transforms *Cont'd*

$x(nT)$	$X(z)$
$r(nT)e^{-\alpha nT}$	$\dfrac{Te^{-\alpha T}z}{(z-e^{-\alpha T})^2}$
$u(nT)\sin\omega nT$	$\dfrac{z\sin\omega T}{z^2-2z\cos\omega T+1}$
$u(nT)\cos\omega nT$	$\dfrac{z(z-\cos\omega T)}{z^2-2z\cos\omega T+1}$
$u(nT)e^{-\alpha nT}\sin\omega nT$	$\dfrac{ze^{-\alpha T}\sin\omega T}{z^2-2ze^{-\alpha T}\cos\omega T+e^{-2\alpha T}}$
$u(nT)e^{-\alpha nT}\cos\omega nT$	$\dfrac{z(z-e^{-\alpha T}\cos\omega T)}{z^2-2ze^{-\alpha T}\cos\omega T+e^{-2\alpha T}}$

The z Transform *Cont'd*

- The Laurent series of a function $X(z)$ about a point $z = a$ assumes the form

$$X(z) = \sum_{n=-\infty}^{\infty} a_n (z-a)^{-n}$$

The z Transform *Cont'd*

- The Laurent series of a function $X(z)$ about a point $z = a$ assumes the form

$$X(z) = \sum_{n=-\infty}^{\infty} a_n (z-a)^{-n}$$

- The z transform is given by

$$X(z) = \sum_{n=-\infty}^{\infty} x(nT) z^{-n}$$

The z Transform *Cont'd*

- The Laurent series of a function $X(z)$ about a point $z = a$ assumes the form

$$X(z) = \sum_{n=-\infty}^{\infty} a_n (z-a)^{-n}$$

- The z transform is given by

$$X(z) = \sum_{n=-\infty}^{\infty} x(nT) z^{-n}$$

- If we compare the above two series for $X(z)$, we conclude that the z transform is a Laurent series of $X(z)$ about the origin, i.e., $a = 0$, with

$$a_n = x(nT)$$

- Since the z transform is a specific Laurent series, it follows that *it inherits all the properties* of the Laurent series, for example, convergence properties.

Inverse Z Transform

- The absolute-convergence theorem states that the z transform, $X(z)$, of a discrete-time signal $x(nT)$ satisfying the conditions

$$\begin{array}{ll} \text{(i)} & x(nT) = 0 \quad \text{for} \quad n < -N_1 \\ \text{(ii)} & |x(nT)| \leq K_1 \quad \text{for} \quad -N_1 \leq n < N_2 \\ \text{(iii)} & |x(nT)| \leq K_2 r^n \quad \text{for} \quad n \geq N_2 \end{array}$$

exists and converges absolutely if and only if

$$r < |z| < R \quad \text{with} \quad R \to \infty$$

Inverse Z Transform

- The Laurent theorem states that a function $X(z)$ has as many distinct Laurent series about the origin as there are annuli of convergence.

Inverse Z Transform

- The Laurent theorem states that a function $X(z)$ has as many distinct Laurent series about the origin as there are annuli of convergence.

- One of these series converges in the outer annulus (i.e., the largest one) which is defined as

$$R_0 < |z| < R \quad \text{with} \quad R \to \infty$$

where R_0 is the radius of a circle passing through the most distant pole of $X(z)$ from the origin.

Summarizing:

- From the absolute convergence theorem, the z transform converges in the annulus

$$r < |z| < R \quad \text{with} \quad R \to \infty$$

Inverse Z Transform *Cont'd*

Summarizing:

- From the absolute convergence theorem, the z transform converges in the annulus

$$r < |z| < R \quad \text{with} \quad R \to \infty$$

- From the Laurent theorem, there is a unique Laurent series of $X(z)$ that converges in the outer annulus of convergence

$$R_0 < |z| < R \quad \text{with} \quad R \to \infty$$

Summarizing:

- From the absolute convergence theorem, the z transform converges in the annulus

$$r < |z| < R \quad \text{with} \quad R \to \infty$$

- From the Laurent theorem, there is a unique Laurent series of $X(z)$ that converges in the outer annulus of convergence

$$R_0 < |z| < R \quad \text{with} \quad R \to \infty$$

- Therefore, *the z transform of $x(nT)$ is the unique Laurent series that converges in the outer annulus* and, furthermore, $r = R_0$.

Inverse Z Transform *Cont'd*

- We conclude that signal $x(nT)$ can be obtained from its z transform $X(z)$ by finding the coefficients of the Laurent series of $X(z)$ that converges in the outer annulus.

Inverse Z Transform *Cont'd*

- We conclude that signal $x(nT)$ can be obtained from its z transform $X(z)$ by finding the coefficients of the Laurent series of $X(z)$ that converges in the outer annulus.
- From the Laurent theorem, we have

$$x(nT) = \frac{1}{2\pi j} \oint_{\Gamma} X(z) z^{n-1}\, dz$$

where contour Γ encloses all the poles of $X(z)z^{n-1}$.

Inverse Z Transform *Cont'd*

- We conclude that signal $x(nT)$ can be obtained from its z transform $X(z)$ by finding the coefficients of the Laurent series of $X(z)$ that converges in the outer annulus.

- From the Laurent theorem, we have

$$x(nT) = \frac{1}{2\pi j} \oint_{\Gamma} X(z) z^{n-1}\, dz$$

  where contour Γ encloses all the poles of $X(z)z^{n-1}$.

- In DSP, this contour integral is said to be the *inverse z transform* of $X(z)$.

Notation

- Like the Fourier transform and its inverse, the z transform and its inverse are often represented in terms of operator notation as

$$X(z) = \mathcal{Z}x(nT) \quad \text{and} \quad x(nT) = \mathcal{Z}^{-1}X(z)$$

respectively.

- If

$$X(z)z^{n-1} = X_0(z) = \frac{N(z)}{\prod_{i=1}^{M}(z-p_i)^{m_i}}$$

where M and m_i are positive integers, then by using the *residue theorem* of complex analysis, we have

$$x(nT) = \sum_{i=1}^{M} \text{res}_{\ z=p_i} X_0(z)$$

where

$$\text{res}_{\ z=p_i} X_0(z) = \frac{1}{(m_i-1)!} \lim_{z\to p_i} \frac{d^{m_i-1}}{dz^{m_i-1}}[(z-p_i)^{m_i} X_0(z)]$$

for a pole of order m_i, and

$$\text{res}_{\ z=p_i} X_0(z) = \lim_{z\to p_i}[(z-p_i)X_0(z)]$$

for a simple pole.

The Inverse z Transform *Cont'd*

From the properties of the Laurent series it follows that several inversion techniques are available for obtaining the inverse z transform, as follows:

- using the binomial theorem,
- using the convolution theorem,
- performing long division,
- using the initial-value theorem, or
- expanding $X(z)$ into partial fractions.

This slide concludes Part I of the review.
Thank you for your attention.