Digital Signal Processing: Course Review

Professor Deepa Kundur

University of Toronto

Course Review

References:

Sections:
1.1, 1.2, 1.3, 1.4
2.1, 2.2, 2.3, 2.4, 2.5
3.1, 3.2, 3.3, 3.4
4.1, 4.2, 4.3, 4.4, 5.1, 5.2, 5.4, 5.5
7.1, 7.2, 8.1
11.2, 11.3, 11.4 of
John G. Proakis and Dimitris G. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 4th edition, 2007,

and supplementary audio, image and video processing notes.

Chapter 1: Introduction

Sampling Theorem

If the highest frequency contained in an analog signal $x_a(t)$ is $F_{max} = B$ and the signal is sampled at a rate

$$F_s > 2F_{max} = 2B$$

then $x_a(t)$ can be exactly recovered from its sample values using the interpolation function

$$g(t) = \frac{\sin(2\pi Bt)}{2\pi Bt}$$

Note: $F_N = 2B = 2F_{max}$ is called the Nyquist rate.

Sampling Theorem

$$\text{Sampling Period} = T = \frac{1}{F_s} = \frac{1}{\text{Sampling Frequency}}$$

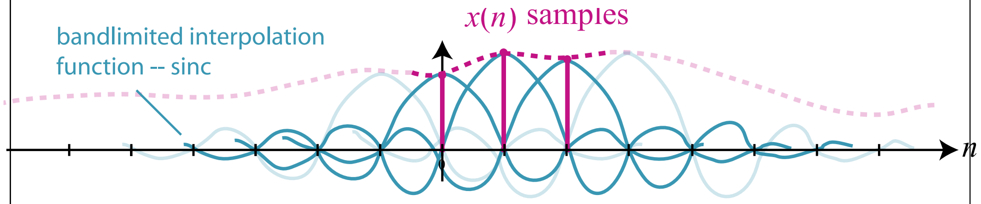
Therefore, given the interpolation relation, $x_a(t)$ can be written as

$$x_a(t) = \sum_{n=-\infty}^{\infty} x_a(nT) g(t - nT)$$

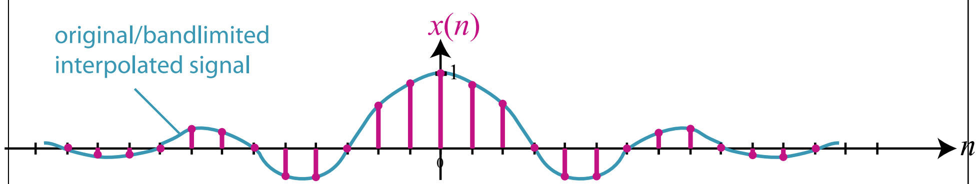
$$x_a(t) = \sum_{n=-\infty}^{\infty} x(n) \; g(t - nT)$$

where $x_a(nT) = x(n)$; called bandlimited interpolation.

Bandlimited Interpolation

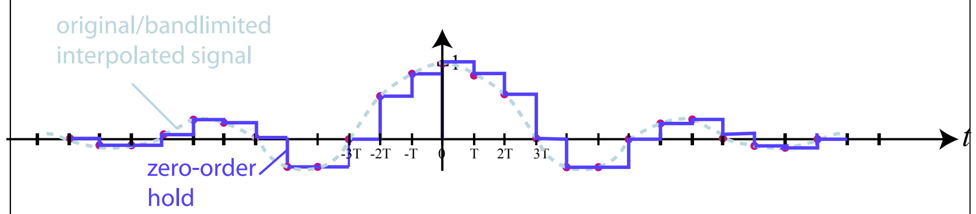


Digital-to-Analog Conversion



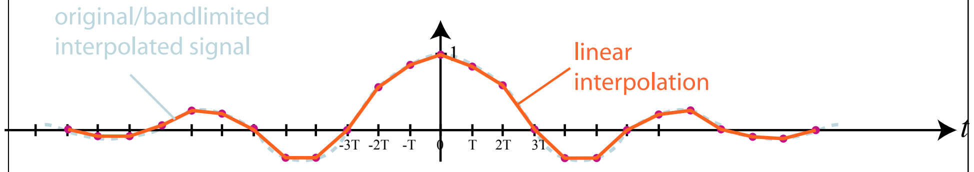
- Common interpolation approaches: bandlimited interpolation, zero-order hold, linear interpolation, higher-order interpolation techniques, e.g., using splines
- In practice, "cheap" interpolation along with a smoothing filter is employed.

Digital-to-Analog Conversion



- Common interpolation approaches: bandlimited interpolation, zero-order hold, linear interpolation, higher-order interpolation techniques, e.g., using splines
- In practice, "cheap" interpolation along with a smoothing filter is employed.

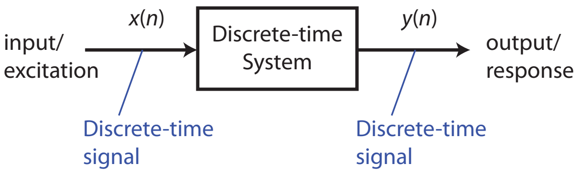
Digital-to-Analog Conversion



- Common interpolation approaches: bandlimited interpolation, zero-order hold, linear interpolation, higher-order interpolation techniques, e.g., using splines
- In practice, "cheap" interpolation along with a smoothing filter is employed.

Chapter 2: Discrete-Time Signals and Systems

Terminology: Input-Output Description



- Input-output description (exact structure of system is unknown or ignored):

$$y(n) = \mathcal{T}[x(n)]$$

- "black box" representation:

$$x(n) \xrightarrow{\mathcal{T}} y(n)$$

Classification of Discrete-Time Systems

Common System Properties:

static	vs.	dynamic
time-invariant	vs.	time-variant
linear	vs.	nonlinear
causal	vs.	non-causal
stable	vs.	unstable systems
⋮		⋮

The Convolution Sum

Let the response of a linear time-invariant (LTI) system to the unit sample input $\delta(n)$ be $h(n)$.

$$\begin{aligned}
\delta(n) &\xrightarrow{\mathcal{T}} h(n) \\
\delta(n-k) &\xrightarrow{\mathcal{T}} h(n-k) \\
\alpha\, \delta(n-k) &\xrightarrow{\mathcal{T}} \alpha\, h(n-k) \\
x(k)\, \delta(n-k) &\xrightarrow{\mathcal{T}} x(k)\, h(n-k) \\
\sum_{k=-\infty}^{\infty} x(k)\delta(n-k) &\xrightarrow{\mathcal{T}} \sum_{k=-\infty}^{\infty} x(k)h(n-k) \\
x(n) &\xrightarrow{\mathcal{T}} y(n)
\end{aligned}$$

The Convolution Sum

Therefore,

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

for any LTI system.

Finite vs. Infinite Impulse Response

Implementation: Two classes

Finite impulse response (FIR):

$$y(n) = \sum_{k=0}^{M-1} h(k)x(n-k) \quad \Big\} \text{ nonrecursive systems}$$

Infinite impulse response (IIR):

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) \quad \Big\} \text{ recursive systems}$$

System Realization

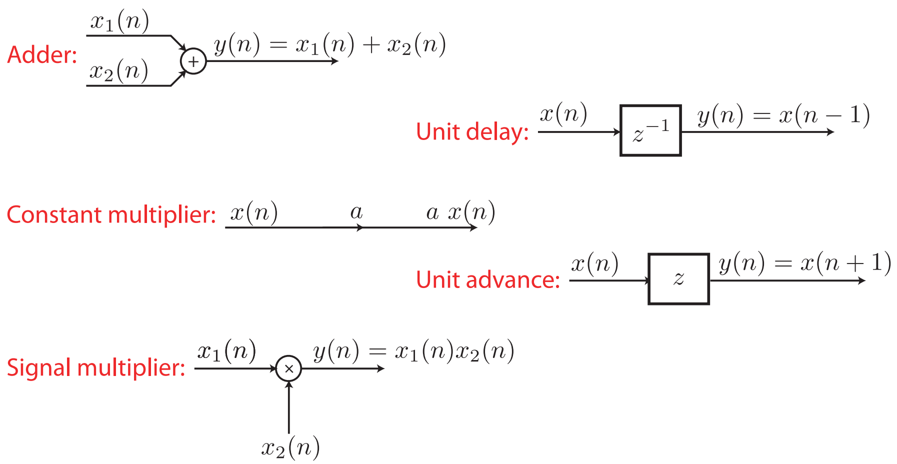
General expression for Nth-order LCCDE:

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k) \qquad a_0 \triangleq 1$$

Initial conditions: $y(-1), y(-2), y(-3), \ldots, y(-N)$.

Need: (1) constant scale, (2) addition, (3) delay elements.

Building Block Elements



Direct Form I vs. Direct Form II Realizations

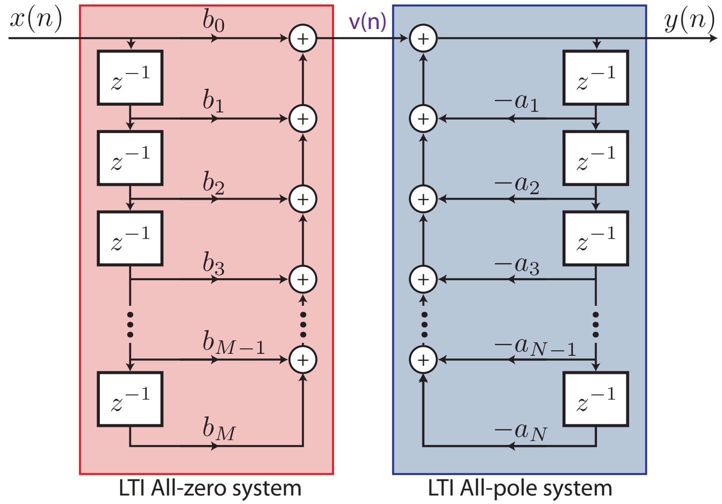
$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k)$$

is equivalent to the cascade of the following systems:

$$\underbrace{v(n)}_{\text{output 1}} = \sum_{k=0}^{M} b_k \underbrace{x(n-k)}_{\text{input 1}} \qquad \underline{\text{nonrecursive}}$$

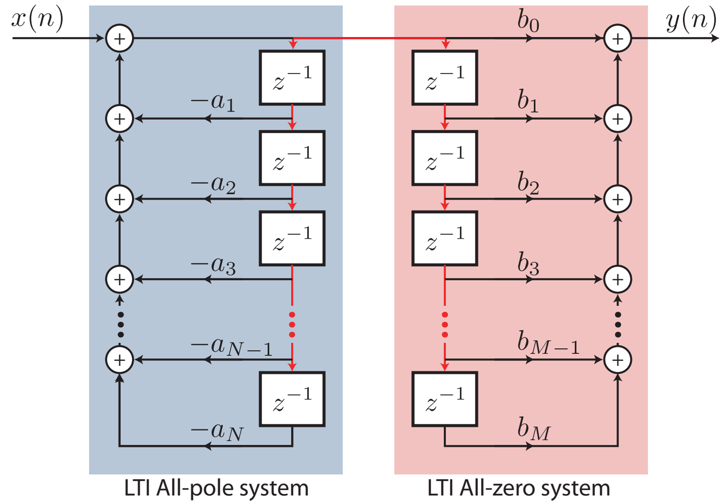
$$\underbrace{y(n)}_{\text{output 2}} = -\sum_{k=1}^{N} a_k y(n-k) + \underbrace{v(n)}_{\text{input 2}} \qquad \underline{\text{recursive}}$$

Direct Form I IIR Filter Implementation



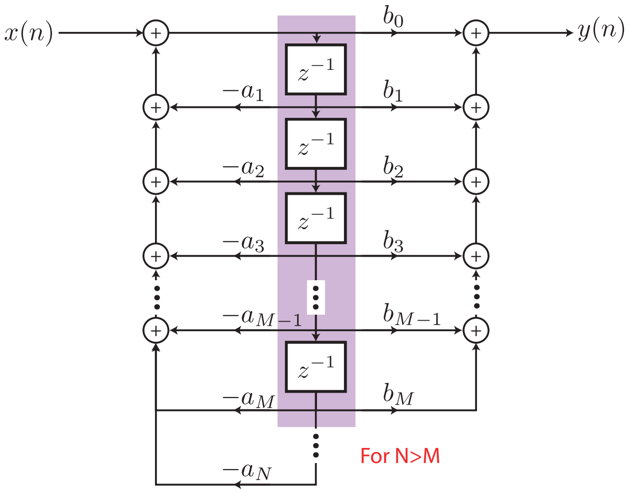
Requires: $M+N+1$ multiplications, $M+N$ additions, $M+N$ memory locations

Direct Form II IIR Filter Implementation



Requires: $M+N+1$ multiplications, $M+N$ additions, $M+N$ memory locations

Direct Form II IIR Filter Implementation



Requires: $M + N + 1$ multiplications, $M + N$ additions, $\max(M, N)$ memory locations

Chapter 3: The z-Transform and Its Applications

The Direct z-Transform

- Direct z-Transform:

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

- Notation:

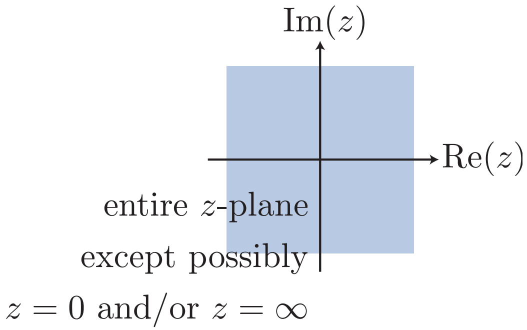
$$X(z) \equiv \mathcal{Z}\{x(n)\}$$

$$x(n) \stackrel{\mathcal{Z}}{\longleftrightarrow} X(z)$$

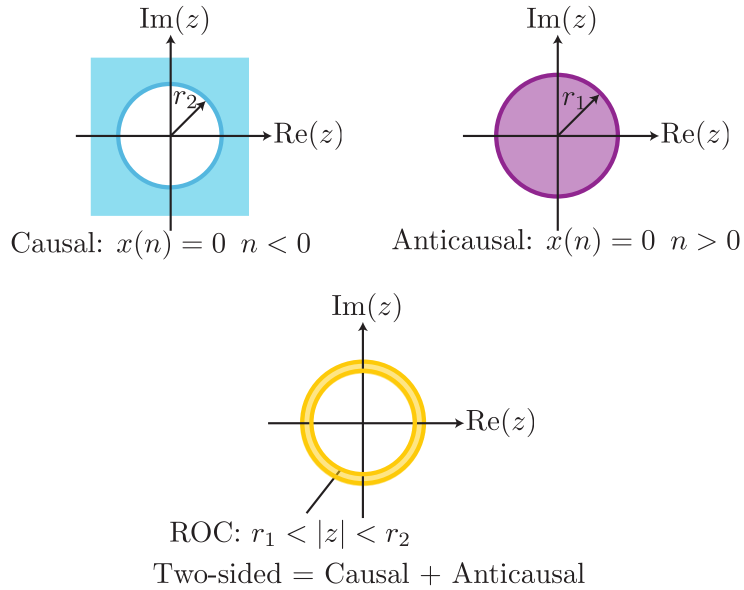
Region of Convergence

- the region of convergence (ROC) of $X(z)$ is the set of all values of z for which $X(z)$ attains a finite value
- The z-Transform is, therefore, uniquely characterized by:
  1. expression for $X(z)$
  2. ROC of $X(z)$

ROC Families: Finite Duration Signals



ROC Families: Infinite Duration Signals



z-Transform Properties

Property	Time Domain	z-Domain	ROC
Notation:	$x(n)$	$X(z)$	ROC: $r_2 < \|z\| < r_1$
	$x_1(n)$	$X_1(z)$	ROC_1
	$x_2(n)$	$X_2(z)$	ROC_2
Linearity:	$a_1 x_1(n) + a_2 x_2(n)$	$a_1 X_1(z) + a_2 X_2(z)$	At least $\text{ROC}_1 \cap \text{ROC}_2$
Time shifting:	$x(n-k)$	$z^{-k} X(z)$	At least ROC, except $z = 0$ (if $k > 0$) and $z = \infty$ (if $k < 0$)
z-Scaling:	$a^n x(n)$	$X(a^{-1} z)$	$\|a\| r_2 < \|z\| < \|a\| r_1$
Time reversal	$x(-n)$	$X(z^{-1})$	$\frac{1}{r_1} < \|z\| < \frac{1}{r_2}$
Conjugation:	$x^*(n)$	$X^*(z^*)$	ROC
z-Differentiation:	$n\, x(n)$	$-z \frac{dX(z)}{dz}$	$r_2 < \|z\| < r_1$
Convolution:	$x_1(n) * x_2(n)$	$X_1(z) X_2(z)$	At least $\text{ROC}_1 \cap \text{ROC}_2$

among others . . .

Convolution using the z-Transform

Basic Steps:

1. Compute z-Transform of each of the signals to convolve (time domain $\rightarrow$ z-domain):

$$\begin{aligned} X_1(z) &= \mathcal{Z}\{x_1(n)\} \\ X_2(z) &= \mathcal{Z}\{x_2(n)\} \end{aligned}$$

2. Multiply the two z-Transforms (in z-domain):

$$X(z) = X_1(z) X_2(z)$$

3. Find the inverse z-Transformof the product (z-domain $\rightarrow$ time domain):

$$x(n) = \mathcal{Z}^{-1}\{X(z)\}$$

Common Transform Pairs

	Signal, $x(n)$	z-Transform, $X(z)$	ROC
1	$\delta(n)$	1	All z
2	$u(n)$	$\frac{1}{1-z^{-1}}$	$\|z\| > 1$
3	$a^n u(n)$	$\frac{1}{1-az^{-1}}$	$\|z\| > \|a\|$
4	$na^n u(n)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\|z\| > \|a\|$
5	$-a^n u(-n-1)$	$\frac{1}{1-az^{-1}}$	$\|z\| < \|a\|$
6	$-na^n u(-n-1)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\|z\| < \|a\|$
7	$\cos(\omega_0 n)u(n)$	$\frac{1-z^{-1}\cos\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$	$\|z\| > 1$
8	$\sin(\omega_0 n)u(n)$	$\frac{z^{-1}\sin\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$	$\|z\| > 1$
9	$a^n\cos(\omega_0 n)u(n)$	$\frac{1-az^{-1}\cos\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$	$\|z\| > \|a\|$
10	$a^n\sin(\omega_0 n)u(n)$	$\frac{1-az^{-1}\sin\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$	$\|z\| > \|a\|$

Common Transform Pairs

	Signal, $x(n)$	z-Transform, $X(z)$	ROC
1	$\delta(n)$	1	All z
2	$u(n)$	$\frac{1}{1-z^{-1}}$	$\|z\| > 1$
3	$a^n u(n)$	$\frac{1}{1-az^{-1}}$	$\|z\| > \|a\|$
4	$na^n u(n)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\|z\| > \|a\|$
5	$-a^n u(-n-1)$	$\frac{1}{1-az^{-1}}$	$\|z\| < \|a\|$
6	$-na^n u(-n-1)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\|z\| < \|a\|$
7	$\cos(\omega_0 n)u(n)$	$\frac{1-z^{-1}\cos\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$	$\|z\| > 1$
8	$\sin(\omega_0 n)u(n)$	$\frac{z^{-1}\sin\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$	$\|z\| > 1$
9	$a^n\cos(\omega_0 n)u(n)$	$\frac{1-az^{-1}\cos\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$	$\|z\| > \|a\|$
10	$a^n\sin(\omega_0 n)u(n)$	$\frac{1-az^{-1}\sin\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$	$\|z\| > \|a\|$

Common Transform Pairs

- z-Transform expressions that are a fraction of polynomials in z^{-1} (or z) are called rational.
- z-Transforms that are rational represent an important class of signals and systems.

The System Function

$$h(n) \overset{\mathcal{Z}}{\longleftrightarrow} H(z)$$
$$\text{time-domain} \overset{\mathcal{Z}}{\longleftrightarrow} z\text{-domain}$$
$$\text{impulse response} \overset{\mathcal{Z}}{\longleftrightarrow} \text{system function}$$

$$y(n) = x(n) * h(n) \overset{\mathcal{Z}}{\longleftrightarrow} Y(z) = X(z) \cdot H(z)$$

Therefore,

$$H(z) = \frac{Y(z)}{X(z)}$$

The System Function of LCCDEs

$$
\begin{aligned}
y(n) &= -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \\
\mathcal{Z}\{y(n)\} &= \mathcal{Z}\{-\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k)\} \\
\mathcal{Z}\{y(n)\} &= -\sum_{k=1}^{N} a_k \mathcal{Z}\{y(n-k)\} + \sum_{k=0}^{M} b_k \mathcal{Z}\{x(n-k)\} \\
Y(z) &= -\sum_{k=1}^{N} a_k z^{-k} Y(z) + \sum_{k=0}^{M} b_k z^{-k} X(z)
\end{aligned}
$$

The System Function of LCCDEs

$$
\begin{aligned}
Y(z) + \sum_{k=1}^{N} a_k z^{-k} Y(z) &= \sum_{k=0}^{M} b_k z^{-k} X(z) \\
Y(z)\left[1 + \sum_{k=1}^{N} a_k z^{-k}\right] &= X(z) \sum_{k=0}^{M} b_k z^{-k} \\
H(z) = \frac{Y(z)}{X(z)} &= \frac{\sum_{k=0}^{M} b_k z^{-k}}{\left[1 + \sum_{k=1}^{N} a_k z^{-k}\right]} \\
\text{LCCDE} &\longleftrightarrow \text{Rational System Function}
\end{aligned}
$$

Many signals of practical interest have a rational z-Transform.

Inversion of the z-Transform

Three popular methods:

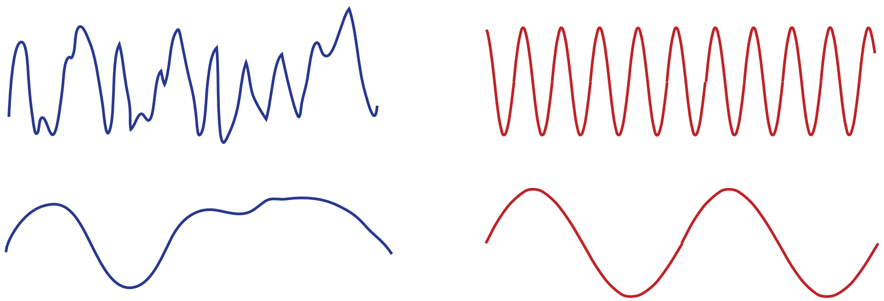
1. Contour integration:
$$x(n) = \frac{1}{2\pi j} \oint_C X(z) z^{n-1} dz$$
2. Expansion into a power series in z or z^{-1}:
$$X(z) = \sum_{k=-\infty}^{\infty} x(k) z^{-k}$$
and obtaining $x(k)$ for all k by inspection.
3. Partial-fraction expansion and table look-up.

Chapter 4: Frequency Analysis of Signals

CTFT: Intuition

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\Omega)e^{j\Omega t}d\Omega$$

- We may consider $x(t)$ as a linear combination of $e^{j\Omega t}$ for $\Omega \in \mathbb{R}$.
- The larger $|X(\Omega)|$, the more $x(t)$ will look like a sinusoid with Ω.



CTFT: Duality

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\Omega)e^{j\Omega t}d\Omega$$
$$X(\Omega) = \int_{-\infty}^{\infty} x(t)e^{-j\Omega t}dt$$

$$\text{Shape A} \overset{\mathcal{F}}{\longleftrightarrow} \text{Shape B}$$
$$\text{Shape B} \overset{\mathcal{F}}{\longleftrightarrow} \text{Shape A}$$
$$\text{Operation A} \overset{\mathcal{F}}{\longleftrightarrow} \text{Operation B}$$
$$\text{Operation B} \overset{\mathcal{F}}{\longleftrightarrow} \text{Operation A}$$

CTFT: Magnitude and Phase

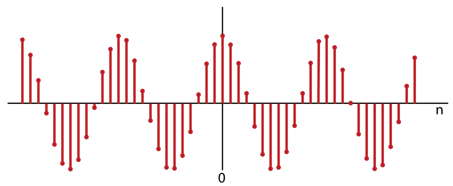
$$\begin{aligned} x(t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\Omega)e^{j\Omega t}d\Omega \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} |X(\Omega)|e^{j\angle X(\Omega)}e^{j\Omega t}d\Omega \\ &= \int_{\infty}^{\infty} |X(\Omega)|e^{j(\Omega t + \angle X(\Omega))}df \end{aligned}$$

- $|X(\Omega)|$ dictates the relative presence of the sinusoid of frequency Ω in $x(t)$.
- $\angle X(\Omega)$ dictates the relative alignment of the sinusoid of frequency Ω in $x(t)$.

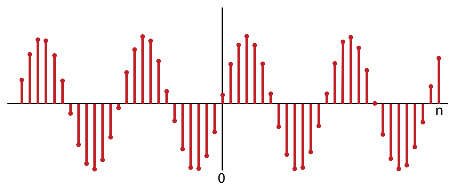
Complex Sinusoids: Discrete-Time

$e^{j\omega n} = \cos(\omega n) + j\sin(\omega n) \quad \equiv$ dst-time complex sinusoid

$\cos(2\pi f n)$



$\sin(2\pi f n)$



Classification of Fourier Pairs

	CTS-TIME	DST-TIME
PERIODIC	Continuous-Time Fourier Series (CTFS)	Discrete-Time Fourier Series (DTFS)
APERIODIC	Continuous-Time Fourier Transform (CTFT)	Discrete-Time Fourier Transform (DTFT)

Duality

	CTS-TIME	DST-TIME
PER	$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t}$ $c_k = \frac{1}{T_p}\int_{T_p} x(t) e^{-j2\pi k F_0 t} dt$	$x(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N}$ $c_k = \frac{1}{N}\sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}$
APER	$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\Omega) e^{j\Omega t} d\Omega$ $X(\Omega) = \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt$	$x(n) = \frac{1}{2\pi}\int_{2\pi} X(\omega) e^{j\omega n} d\omega$ $X(\omega) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n}$

$$\text{periodic} \overset{\mathcal{F}}{\longleftrightarrow} \text{discrete}$$
$$\text{discrete} \overset{\mathcal{F}}{\longleftrightarrow} \text{periodic}$$
$$\text{aperiodic} \overset{\mathcal{F}}{\longleftrightarrow} \text{continuous}$$
$$\text{continuous} \overset{\mathcal{F}}{\longleftrightarrow} \text{aperiodic}$$

Discrete-Time Fourier Series (DTFS)

For discrete-time periodic signals with period N:

- Synthesis equation:

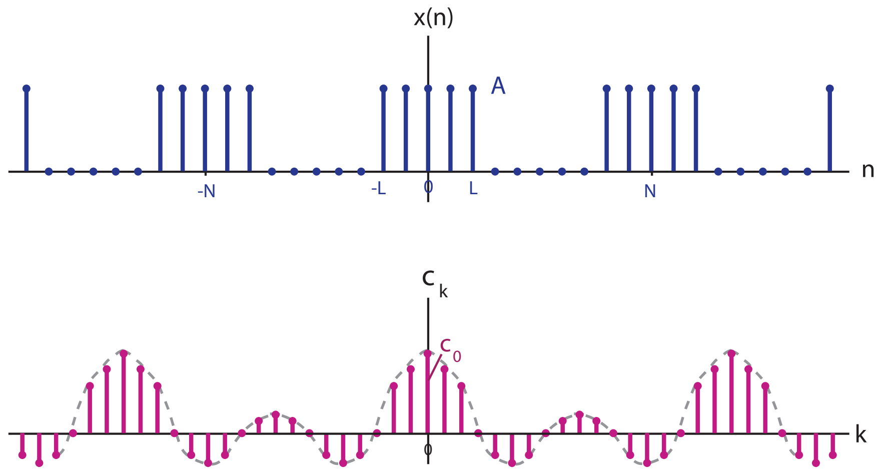
$$x(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N}$$

- Analysis equation:

$$c_k = \frac{1}{N}\sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}$$

Convergence conditions: None due to finite sums.

DTFS: Example Pair



Discrete-Time Fourier Transform (DTFT)

For discrete-time aperiodic signals:

- Synthesis equation:

$$x(n) = \frac{1}{2\pi} \int_{2\pi} X(\omega) e^{j\omega n} d\omega$$

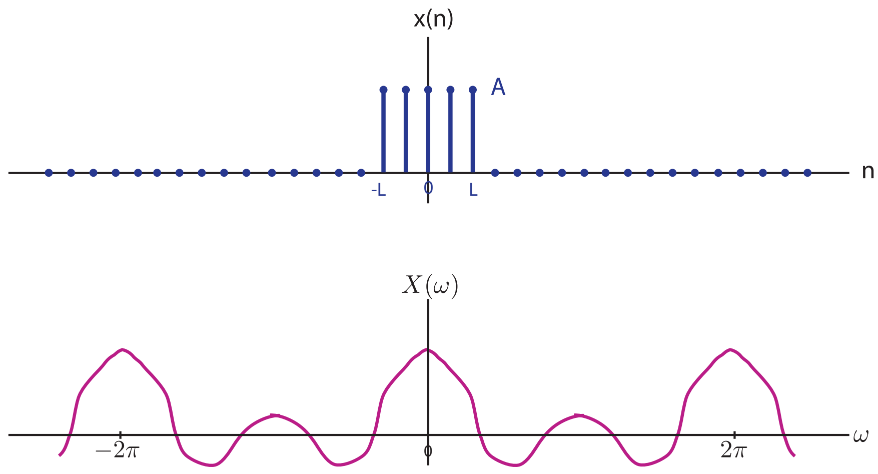
- Analysis equation:

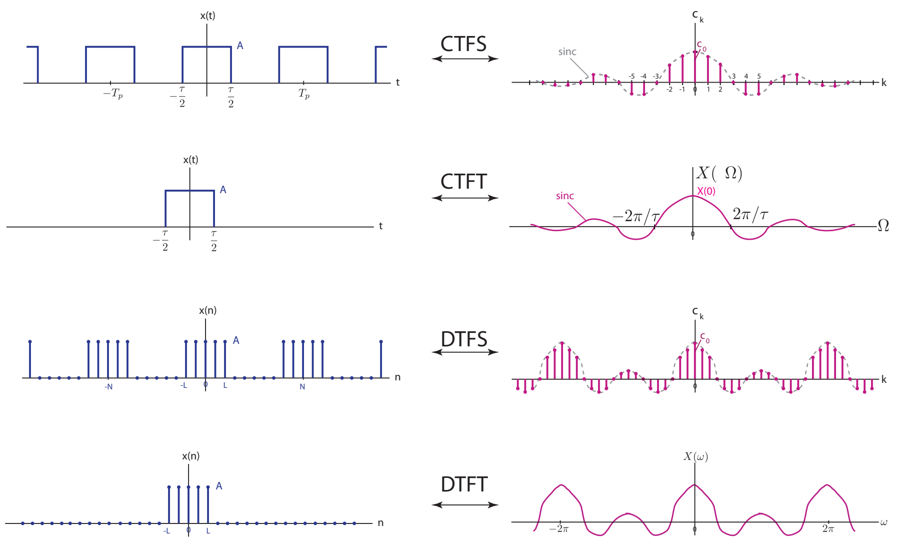
$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n}$$

Convergence conditions:

$$\sum_{n=-\infty}^{\infty} |x(n)| < \infty$$

DTFT: Example Pair





DTFT Theorems and Properties

Property	Time Domain	Frequency Domain
Notation:	$x(n)$	$X(\omega)$
	$x_1(n)$	$X_1(\omega)$
	$x_2(n)$	$X_1(\omega)$
Linearity:	$a_1x_1(n) + a_2x_2(n)$	$a_1X_1(\omega) + a_2X_2(\omega)$
Time shifting:	$x(n-k)$	$e^{-j\omega k}X(\omega)$
Time reversal	$x(-n)$	$X(-\omega)$
Convolution:	$x_1(n) * x_2(n)$	$X_1(\omega)X_2(\omega)$
Correlation:	$r_{x_1x_2}(l) = x_1(l) * x_2(-l)$	$S_{x_1x_2}(\omega) = X_1(\omega)X_2(-\omega)$
		$= X_1(\omega)X_2^*(\omega)$ [if $x_2(n)$ real]
Wiener-Khintchine:	$r_{xx}(l) = x(l) * x(-l)$	$S_{xx}(\omega) = \lvert X(\omega)\rvert^2$

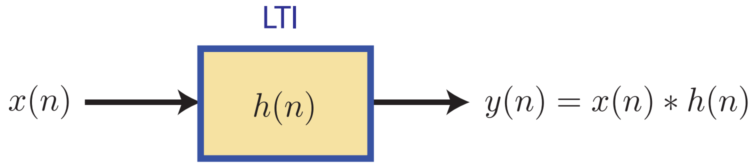
among others ...

DTFT Symmetry Properties

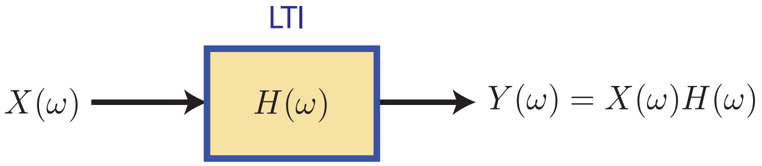
Time Sequence	DTFT
$x(n)$	$X(\omega)$
$x^*(n)$	$X^*(-\omega)$
$x^*(-n)$	$X^*(\omega)$
$x(-n)$	$X(-\omega)$
$x_R(n)$	$X_e(\omega) = \frac{1}{2}[X(\omega) + X^*(-\omega)]$
$jx_I(n)$	$X_o(\omega) = \frac{1}{2}[X(\omega) - X^*(-\omega)]$
$x(n)$ real	$X(\omega) = X^*(-\omega)$ $X_R(\omega) = X_R(-\omega)$ $X_I(\omega) = -X_I(-\omega)$ $\lvert X(\omega)\rvert = \lvert X(-\omega)\rvert$ $\angle X(\omega) = -\angle X(-\omega)$
$x_e'(n) = \frac{1}{2}[x(n) + x^*(-n)]$	$X_R(\omega)$
$x_o'(n) = \frac{1}{2}[x(n) - x^*(-n)]$	$jX_I(\omega)$

Chapter 5: Frequency Domain Analysis of LTI Systems

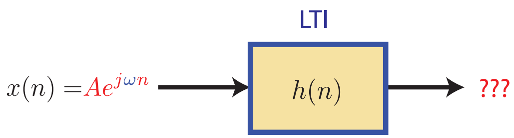
Linear Time-Invariant (LTI) Systems

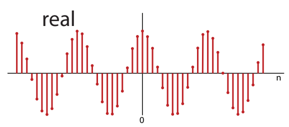


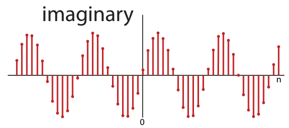
$h(n) \longleftrightarrow H(\omega)$



Linear Time-Invariant (LTI) Systems

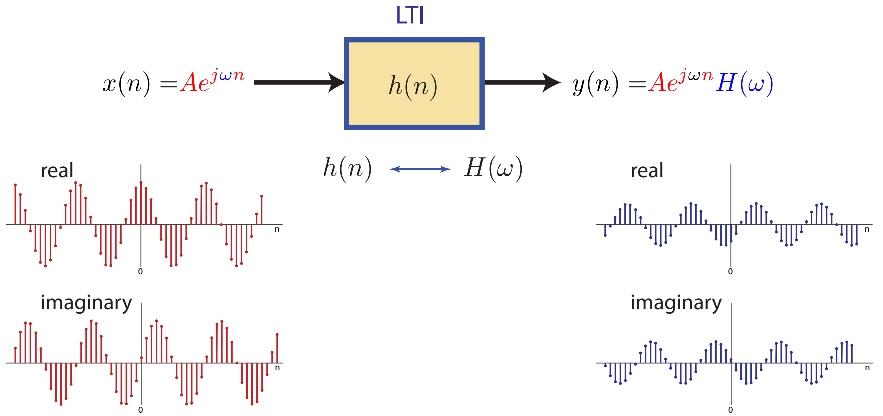






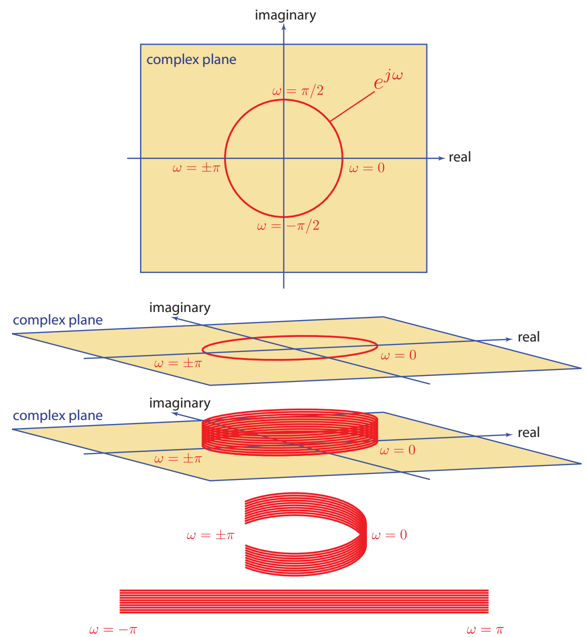
$h(n) \longleftrightarrow H(\omega)$

Linear Time-Invariant (LTI) Systems



Frequency Response of LTI Systems

z-Domain		ω-Domain
$H(z)$	$\overset{z=e^{j\omega}}{\Longrightarrow}$	$H(\omega)$
system function	$\overset{z=e^{j\omega}}{\Longrightarrow}$	frequency response
$Y(z) = X(z)H(z)$	$\overset{z=e^{j\omega}}{\Longrightarrow}$	$Y(\omega) = X(\omega)H(\omega)$



LTI Systems as Frequency-Selective Filters

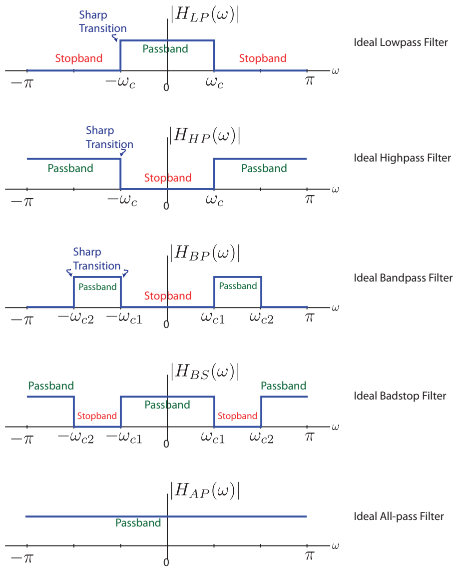
- Filter: device that discriminates, according to some attribute of the input, what passes through it
- For LTI systems, given $Y(\omega) = X(\omega)H(\omega)$
  - $H(\omega)$ acts as a kind of weighting function or spectral shaping function of the different frequency components of the signal

LTI system $\Longleftrightarrow$ Filter

Ideal Filters

Classification:

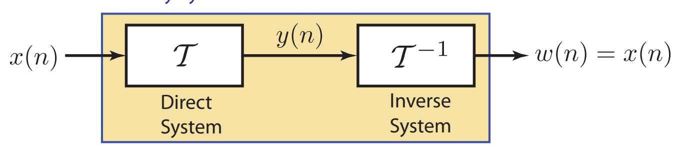
- lowpass
- highpass
- bandpass
- bandstop
- all-pass



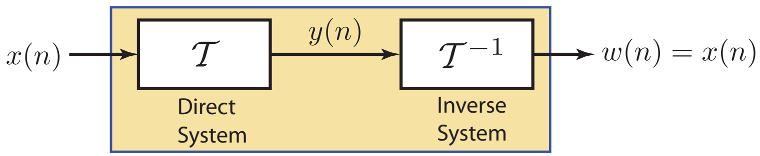
Invertibility of Systems

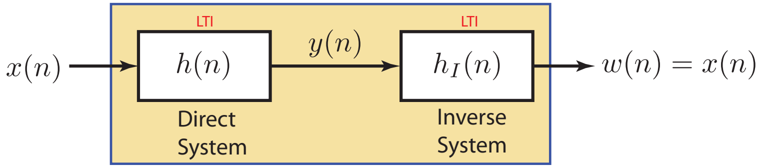
- Invertible system: there is a one-to-one correspondence between its input and output signals
- the one-to-one nature allows the process of reversing the transformation between input and output; suppose

$$y(n) = \mathcal{T}[x(n)] \quad \text{where } \mathcal{T} \text{ is one-to-one}$$
$$w(n) = \mathcal{T}^{-1}[y(n)] = \mathcal{T}^{-1}[\mathcal{T}[x(n)]] = x(n)$$



Invertibility of LTI Systems





Invertibility of LTI Systems

- Therefore,

$$h(n) * h_I(n) = \delta(n)$$

- For a given $h(n)$, how do we find $h_I(n)$?
- Consider the $z-$domain

$$H(z)H_I(z) = 1$$
$$H_I(z) = \frac{1}{H(z)}$$

Invertibility of Rational LTI Systems

- Suppose, $H(z)$ is rational:

$$
\begin{aligned}
H(z) &= \frac{A(z)}{B(z)} \\
H_I(z) &= \frac{B(z)}{A(z)} \\
\text{poles of } H(z) &= \text{zeros of } H_I(z) \\
\text{zeros of } H(z) &= \text{poles of } H_I(z)
\end{aligned}
$$

Chapter 7: The Discrete Fourier Transform

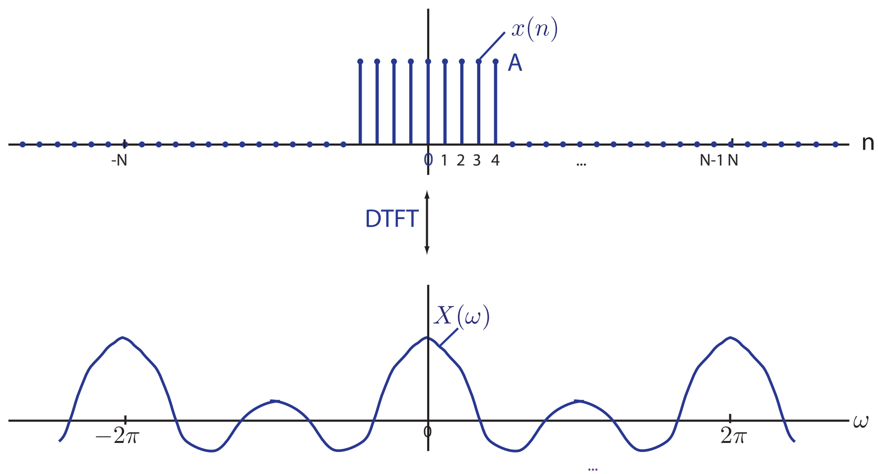
Intuition

aperiodic + dst in time	$\overset{\text{DTFT}}{\longleftrightarrow}$	cts + periodic in freq
$\downarrow$ periodic repetition		$\downarrow$ sampling
periodic + dst in time	$\overset{\text{DTFS}}{\longleftrightarrow}$	dst + periodic in freq
one period of dst-time samples $n = 0, 1, \ldots, N-1$	$\overset{\text{DFT}}{\longleftrightarrow}$	one period of dst-freq samples $k = 0, 1, \ldots, N-1$

Therefore, we define the Discrete Fourier Transform (DFT) as being a computable transform that approximates the DTFT.

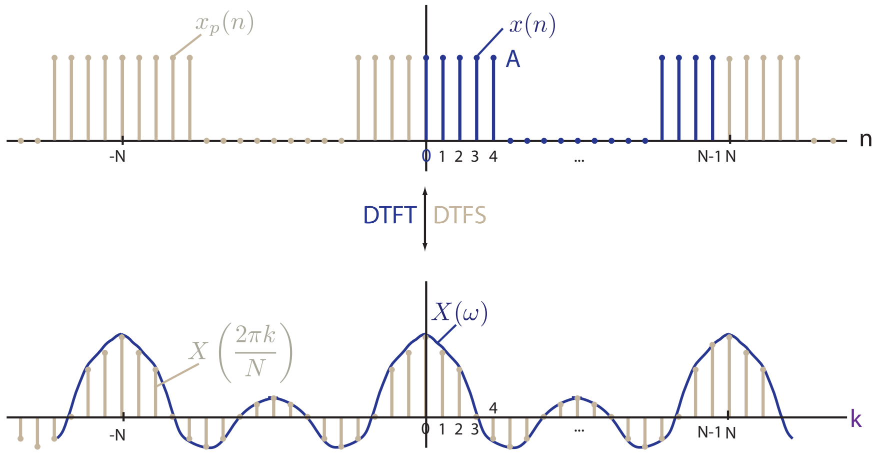
Intuition

Example



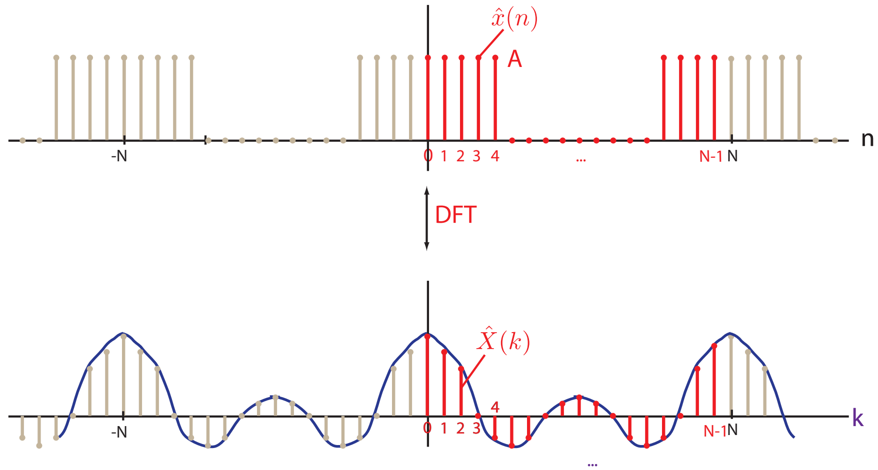
Intuition

Example



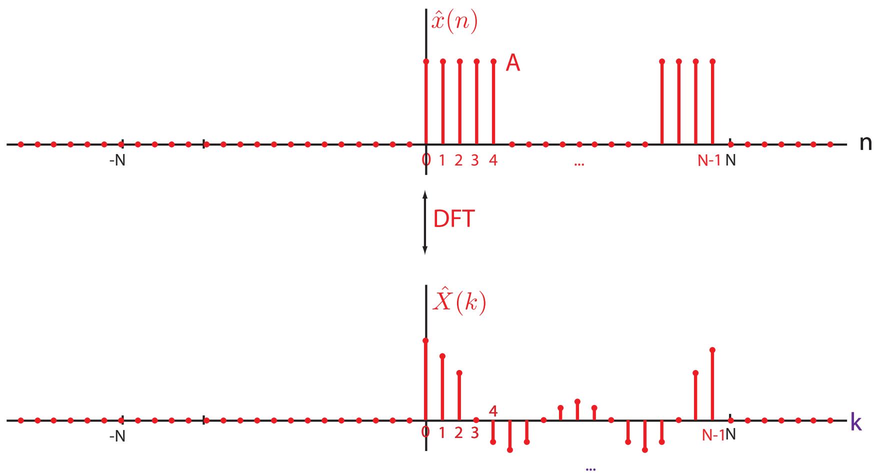
Intuition

Example



Intuition

Example



DTFT, DTFS and DFT

$$x(n) \text{ for all } n \quad \stackrel{\text{DTFT}}{\longleftrightarrow} \quad X(\omega) \text{ for all } \omega$$

$\downarrow$ periodic repetition $\qquad\qquad$ $\downarrow$ sampling

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n + lN) \text{ for all } n \quad \stackrel{\text{DTFS}}{\longleftrightarrow} \quad X(k) = X(\omega)|_{\omega=\frac{2\pi}{N}k} \text{ for all } k$$

$$\hat{x}(n) \quad \stackrel{\text{DFT}}{\longleftrightarrow} \quad \hat{X}(k)$$

where

$$\hat{x}(n) = \begin{cases} x_p(n) & \text{for } n = 0, \ldots, N-1 \\ 0 & \text{otherwise} \end{cases}$$

and

$$\hat{X}(k) = \begin{cases} X(k) & \text{for } k = 0, \ldots, N-1 \\ 0 & \text{otherwise} \end{cases}$$

The Discrete Fourier Transform Pair

- DFT and inverse-DFT (IDFT):

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi k \frac{n}{N}}, \quad k = 0, 1, \ldots, N-1$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi k \frac{n}{N}}, \quad n = 0, 1, \ldots, N-1$$

Note: we drop the $\hat{\cdot}$ notation from now on.

Frequency Domain Sampling

- Recall, sampling in time results in a periodic repetition in frequency.

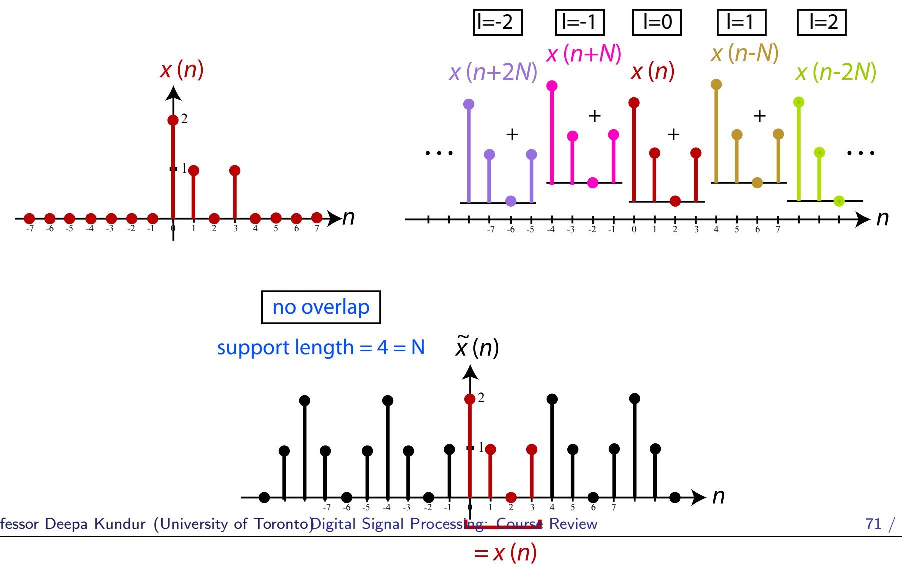
$$x(n) = x_a(t)|_{t=nT} \quad \stackrel{\mathcal{F}}{\longleftrightarrow} \quad X(\omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_a(\omega + \frac{2\pi}{T}k)$$

- Similarly, sampling in frequency results in periodic repetition in time.

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n + lN) \quad \stackrel{\mathcal{F}}{\longleftrightarrow} \quad X(k) = X(\omega)|_{\omega = \frac{2\pi}{N}k}$$

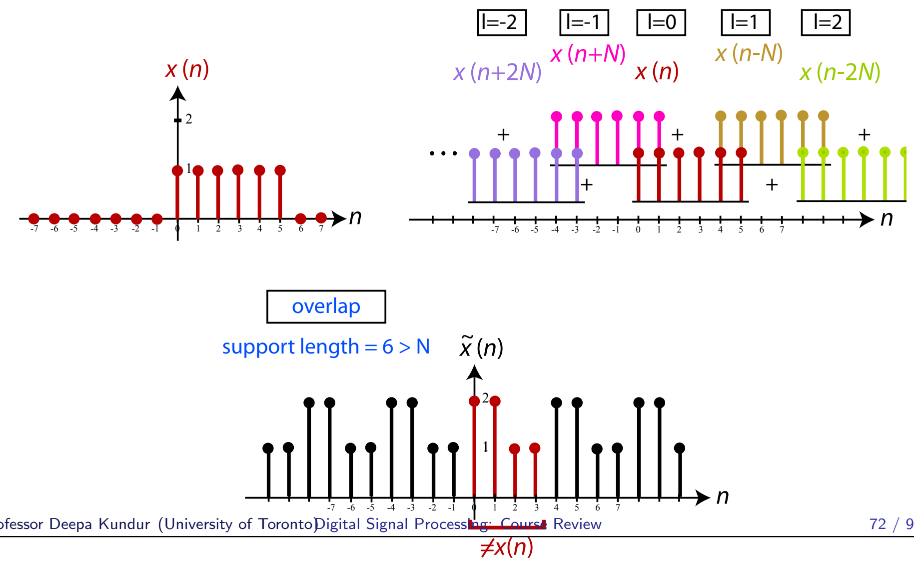
Frequency Domain Sampling and Reconstruction

$N = 4$



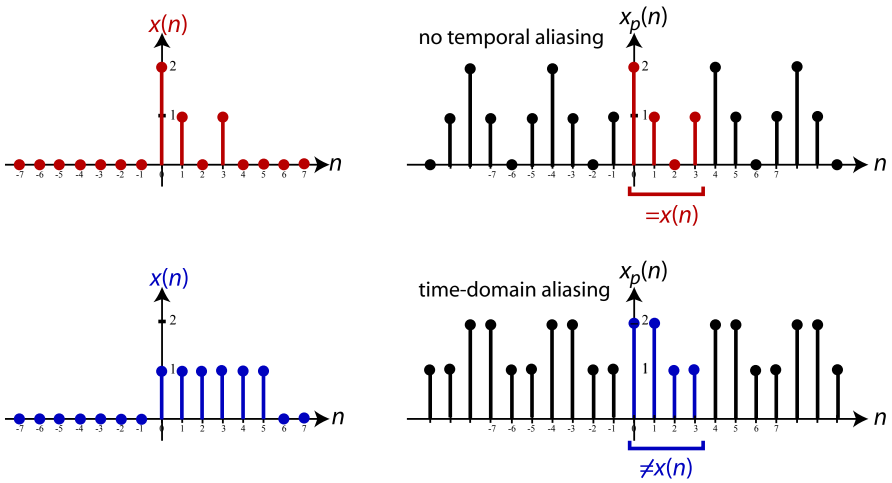
Frequency Domain Sampling and Reconstruction

$N = 4$



Frequency Domain Sampling and Reconstruction

$N = 4$



Important DFT Properties

Property	Time Domain	Frequency Domain
Notation:	$x(n)$	$X(k)$
Periodicity:	$x(n) = x(n+N)$	$X(k) = X(k+N)$
Linearity:	$a_1 x_1(n) + a_2 x_2(n)$	$a_1 X_1(k) + a_2 X_2(k)$
Time reversal	$x(N-n)$	$X(N-k)$
Circular time shift:	$x((n-l))_N$	$X(k)e^{-j2\pi kl/N}$
Circular frequency shift:	$x(n)e^{j2\pi ln/N}$	$X((k-l))_N$
Complex conjugate:	$x^*(n)$	$X^*(N-k)$
Circular convolution:	$x_1(n) \circledast x_2(n)$	$X_1(k)X_2(k)$
Multiplication:	$x_1(n)x_2(n)$	$\frac{1}{N} X_1(k) \circledast X_2(k)$
Parseval's theorem:	$\sum_{n=0}^{N-1} x(n)y^*(n)$	$\frac{1}{N}\sum_{k=0}^{N-1} X(k)Y^*(k)$

Chapter 8: The Fast Fourier Transform

Radix-2 FFT

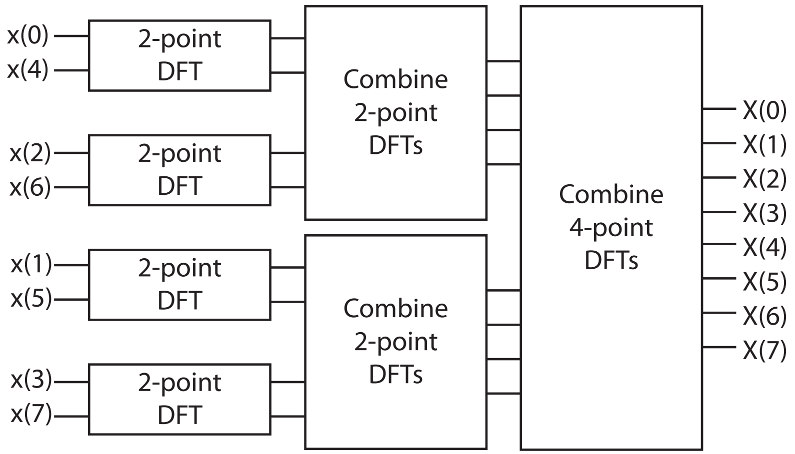
Two strategies:

- Decimation in time (our focus in the lecture)
- Decimation in frequency

- Note: We assume that N is a power of two; i.e., $N = 2^r$.

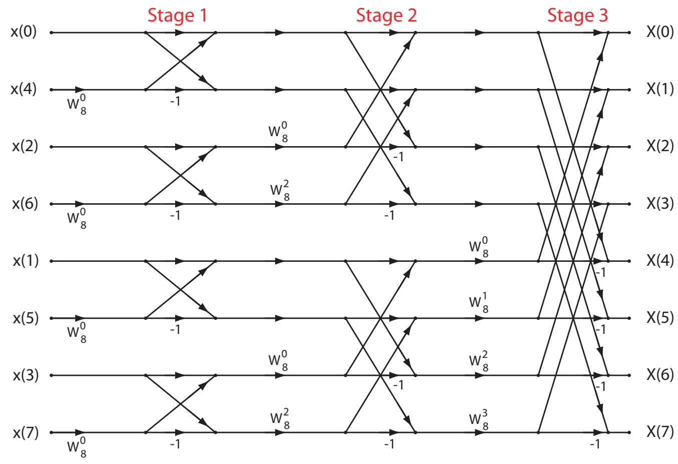
Radix-2 FFT: Decimation-in-time

For $N = 8$.

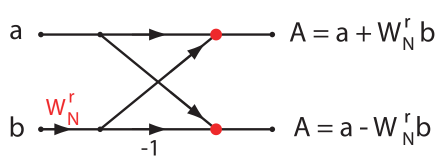


Radix-2 FFT: Decimation-in-time

For $N = 8$.



FFT Complexity



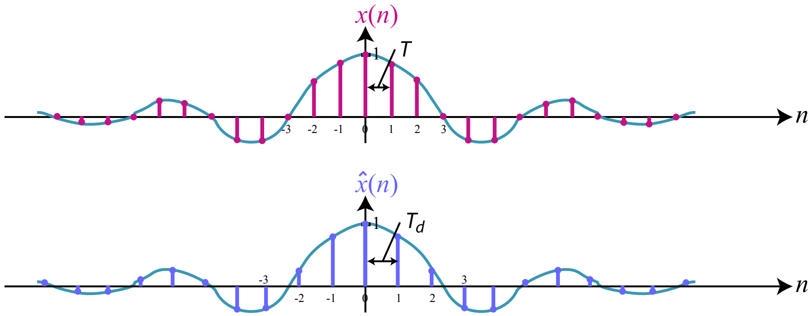
- Each butterfly requires:
  - one complex multiplication
  - two complex additions
- In total, there are:
  - $\frac{N}{2}$ butterflies per stage
  - $\log N$ stages
- Order of the overall DFT computation is: $O(N \log N)$.

Chapter 11: Multirate Digital Signal Processing

Sampling Rate Conversion

- **Goal:** Given a discrete-time signal $x(n)$ sampled at period T from an underlying continuous-time signal $x_a(t)$, determine a new sequence $\hat{x}(n)$ that is a sampled version of $x_a(t)$ at a different sampling rate T_d.

$$x(n) = x_a(nT) \qquad \hat{x}(n) = x_a(nT_d)$$

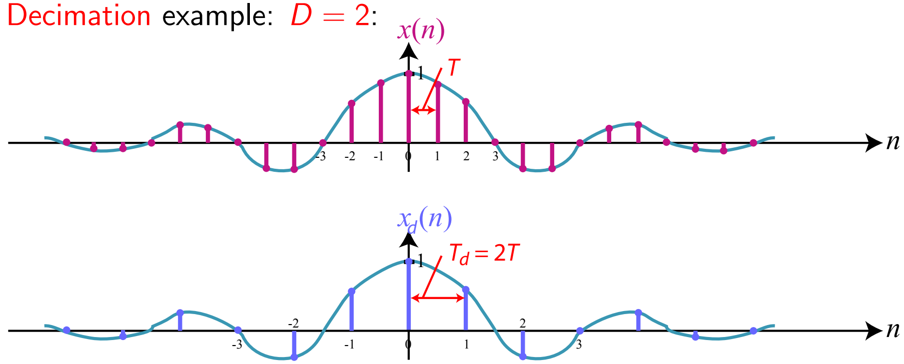


Sampling of Discrete-Time Signals

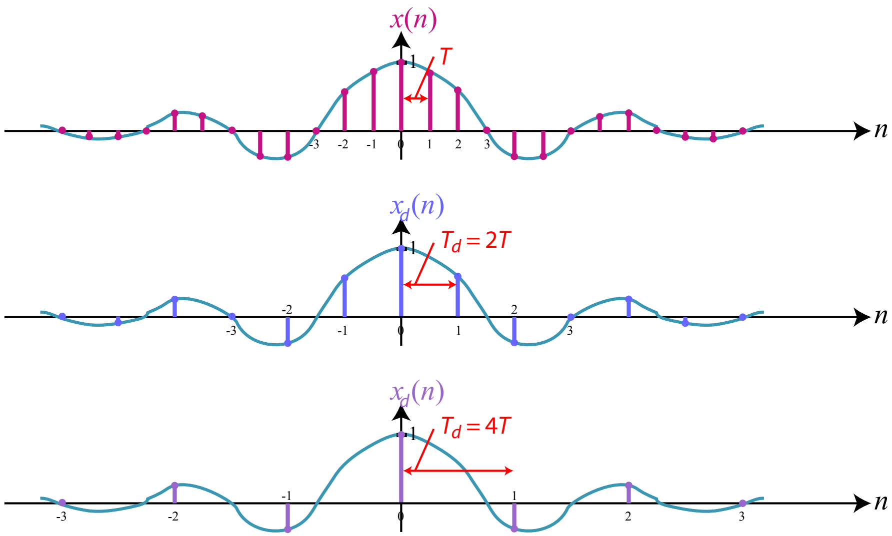
Suppose a discrete-time signal $x(n)$ is sampled by taking every Dth sample as follows:

$$x_d(n) = x(nD), \qquad \text{for all } n$$

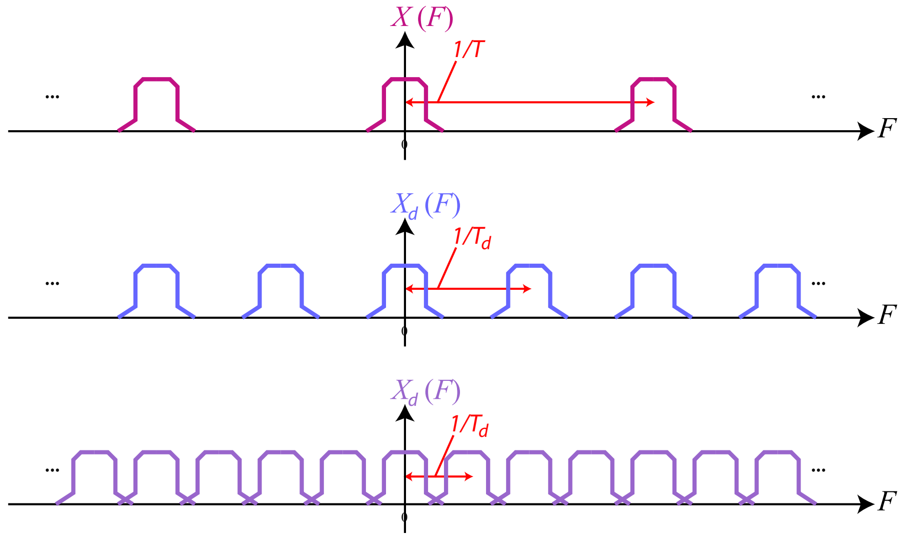
Decimation example: $D = 2$:



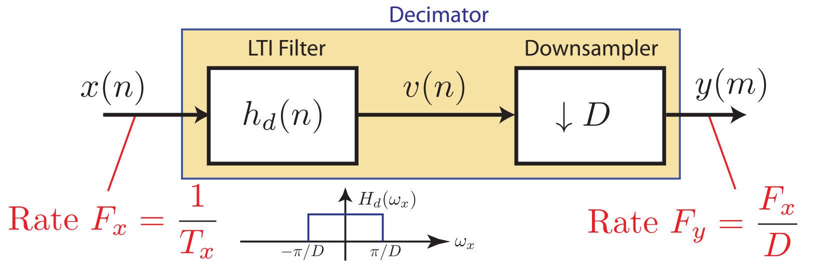
Decimation example: $D = 2, 4$:



Decimation example: $D = 2, 4$:



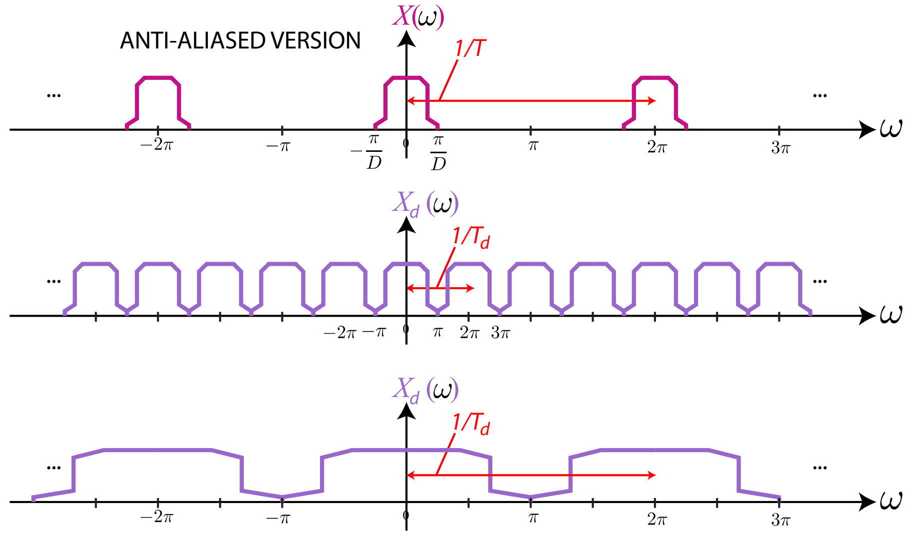
Downsampling with Anti-Alaising Filter



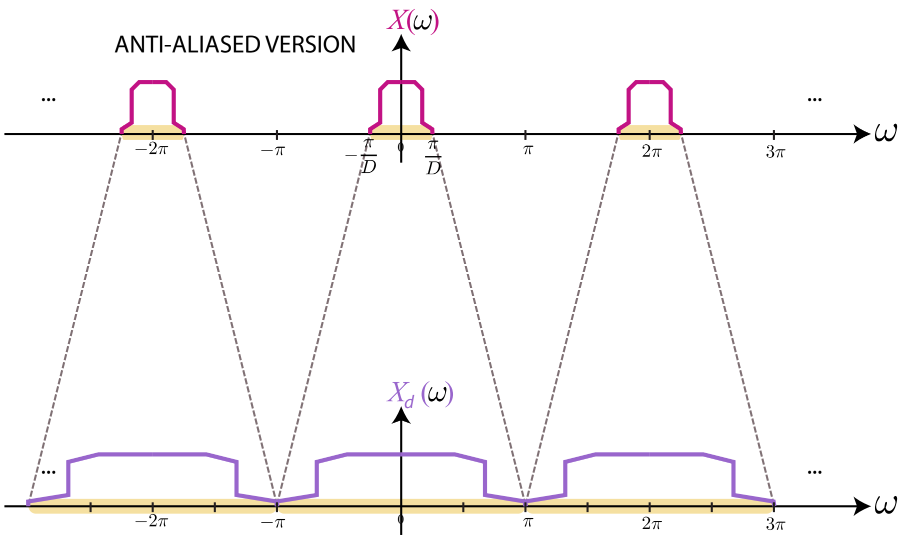
- The anti-aliasing filter $H_d(\omega)$ should have effective continuous-time frequency cutoff of $F_0 = \frac{1}{2DT}$ Hz, which is equivalent to a normalized cutoff of:

$$f_0 = \frac{F_0}{F_s} = \frac{1}{2DT} \cdot \frac{1}{F_s} = \frac{1}{2D} \quad \text{or} \quad \omega_0 = 2\pi \frac{1}{2D} = \frac{\pi}{D}$$

$-\pi/D \leq \omega \leq \pi/D$ is expanded into $-\pi \leq \omega \leq \pi$



$-\pi/D \leq \omega \leq \pi/D$ is expanded into $-\pi \leq \omega \leq \pi$



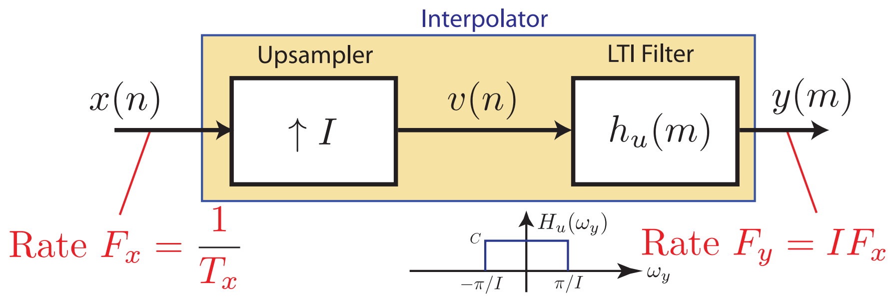
Interpolation of Discrete-time Signals

To achieve this, consider a two-stage process:

- Stage 1: Upsample to appropriately compress the spectrum.
- Stage 2: Then filter with an appropriate lowpass filter.

- We will consider upsampling by a factor of I.
  - Note: we change here the interpolation factor from D to I to distinguish our results from decimation.

Interpolation by a Factor I



- Interpolation only increases the visible resolution of the signal. No new information is gained.

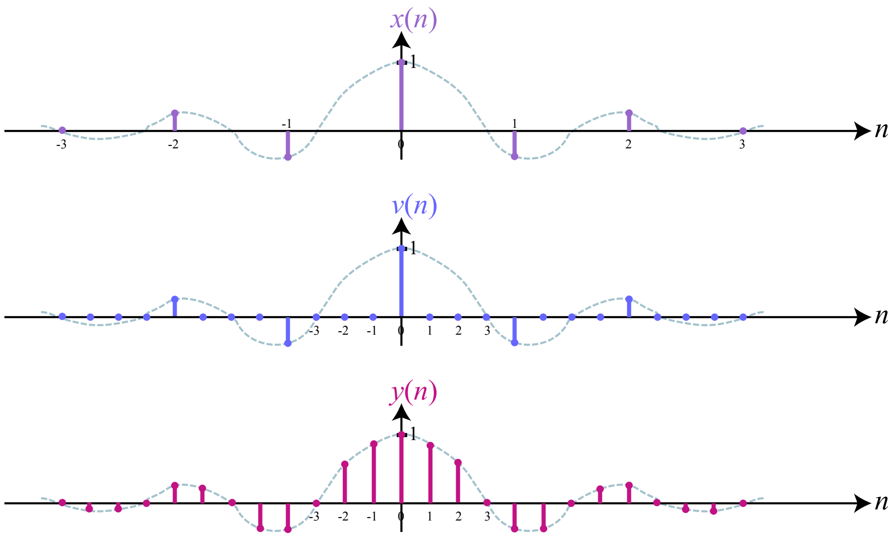
Interpolation of Discrete-time Signals

- Upsampling (without filtering) can be represented as:

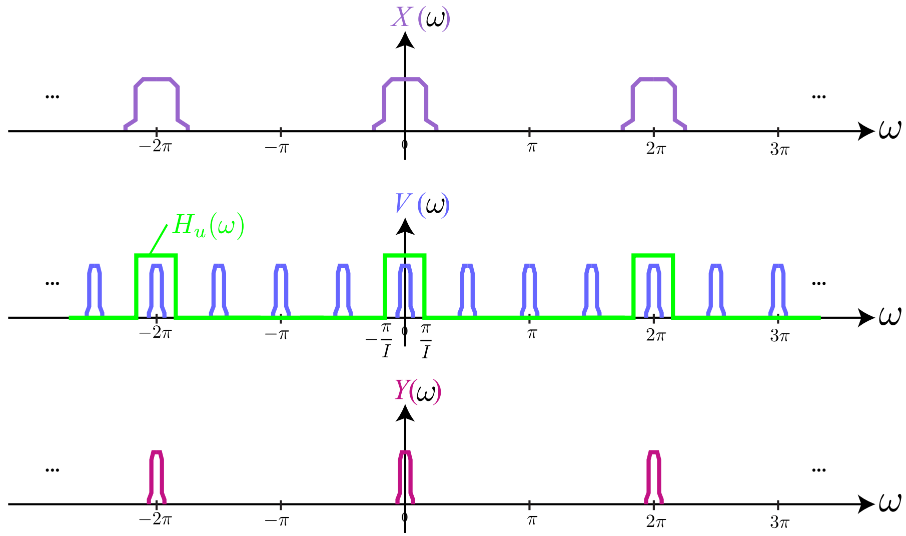
$$v(m) = \begin{cases} x(m/I) & m = 0, \pm I, \pm 2I, \ldots \\ 0 & \text{otherwise} \end{cases}$$

$$V(\omega) = X(\omega I)$$

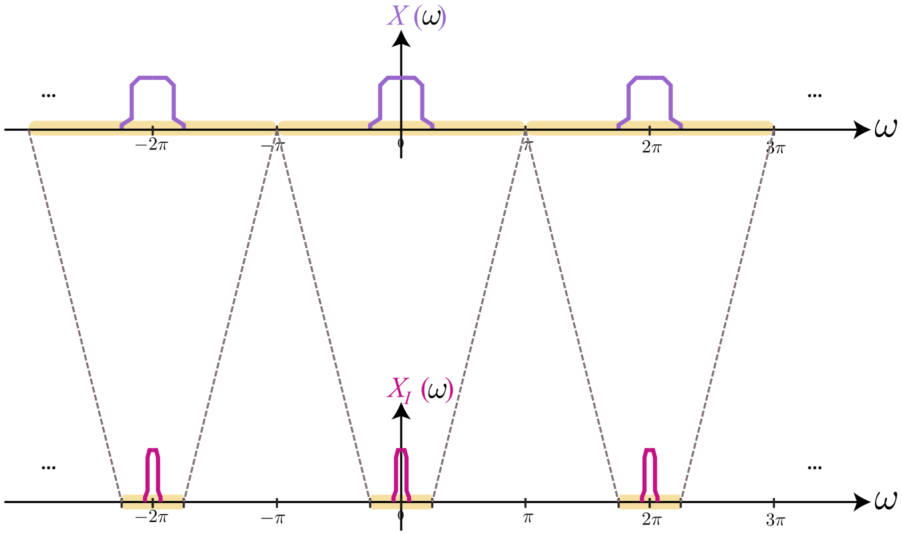
Interpolation example: $I = 4$: upsampling + lowpass filtering



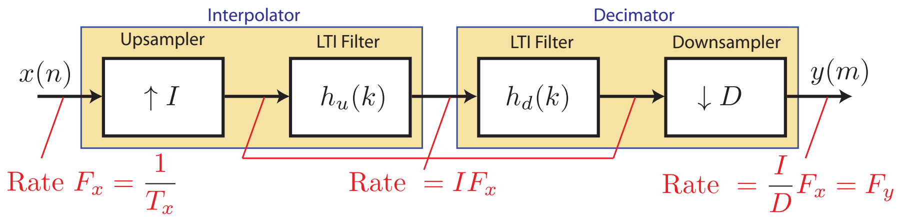
Interpolation example: $I = 4$: upsampling + lowpass filtering



$-\pi \le \omega \le \pi$ is compressed into $-\pi/I \le \omega \le \pi/I$

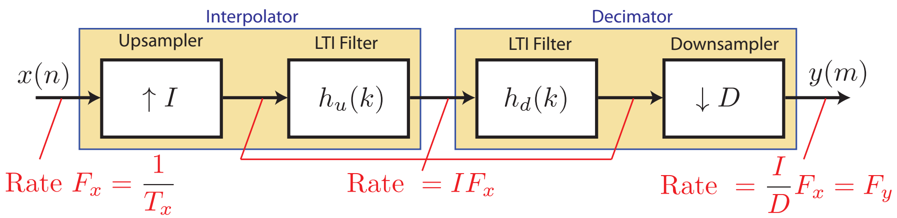


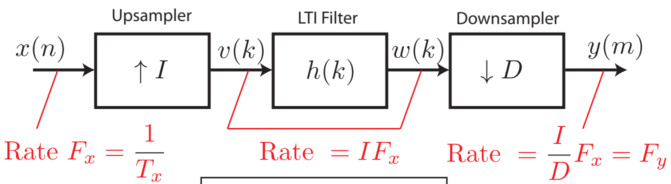
Sampling Rate Conversion by I/D



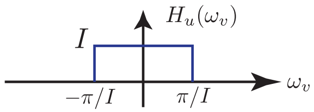
- $x(n)$: original samples at sampling rate F_x
- $y(n)$: new samples at sampling rate F_y

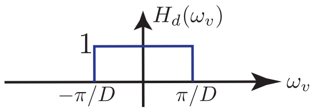
Sampling Rate Conversion by I/D





$h(k) = h_u(k) * h_d(k)$





$$H(\omega) = H_u(\omega)H_d(\omega) = \begin{cases} I & 0 \le |\omega| \le \min(\pi/D, \pi/I) \\ 0 & \text{otherwise} \end{cases}$$

Sampling Rate Conversion by I/D