Table of Chapters

Chapter 1: Overview of Digital Signal Processing

Chapter Intended Learning Outcomes:

(i) Understand basic terminology in digital signal processing

(ii) Differentiate digital signal processing and analog signal processing

(iii) Describe basic digital signal processing application areas

Signal:

- Anything that conveys information, e.g.,
    - Speech
    - Electrocardiogram (ECG)
    - Radar pulse
    - DNA sequence
    - Stock price
    - Code division multiple access (CDMA) signal
    - Image
    - Video

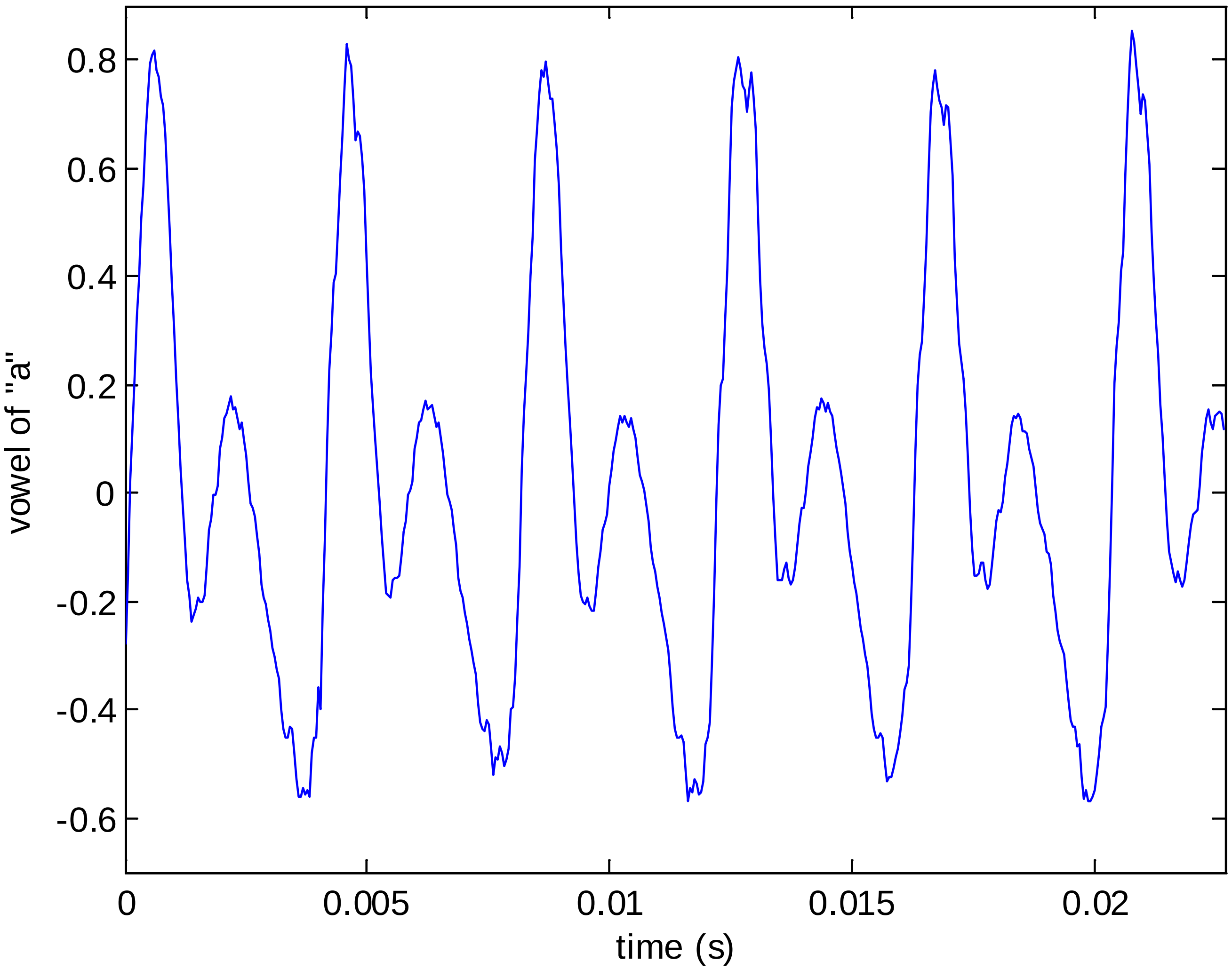


Fig.1.1: Speech

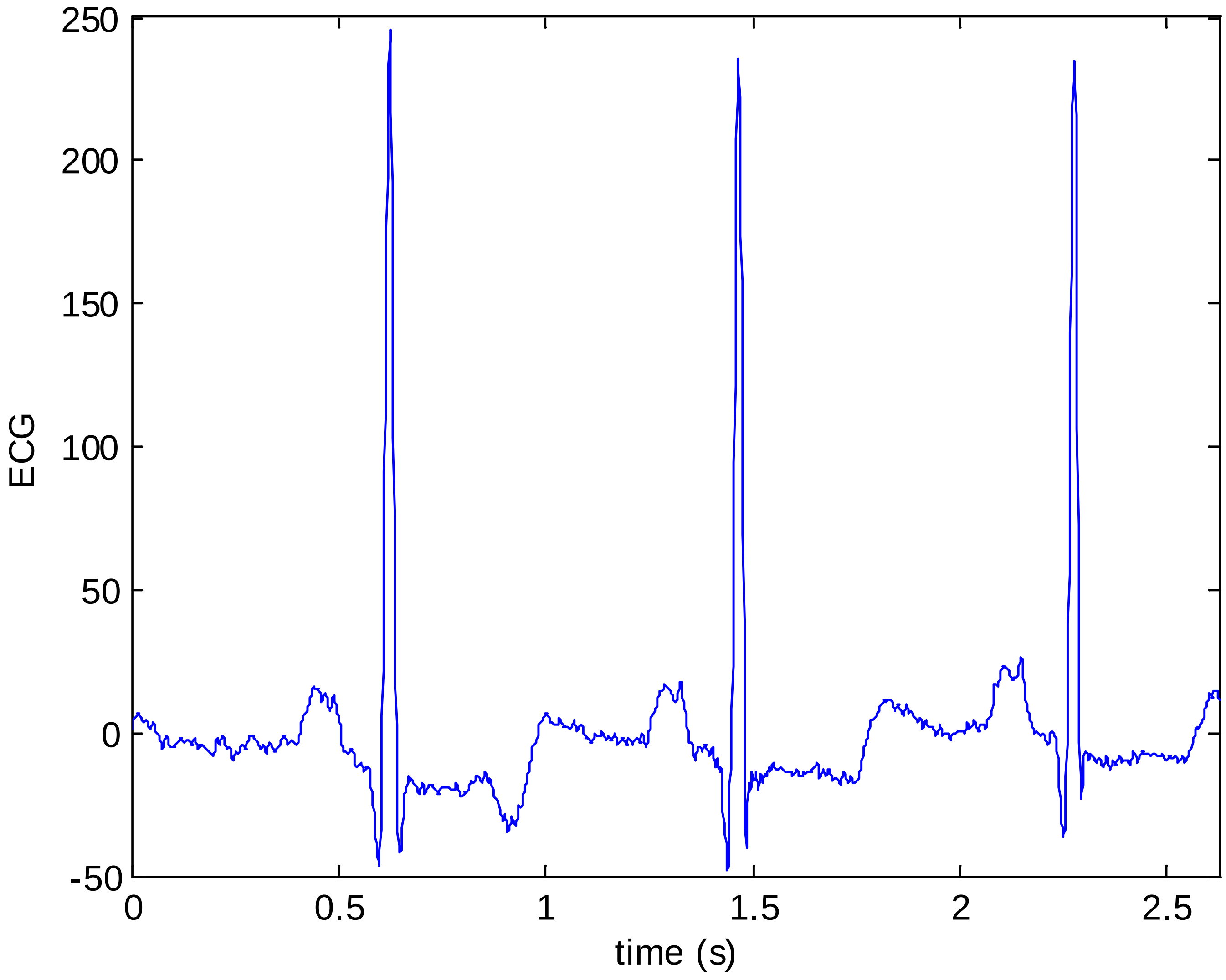


Fig.1.2: ECG

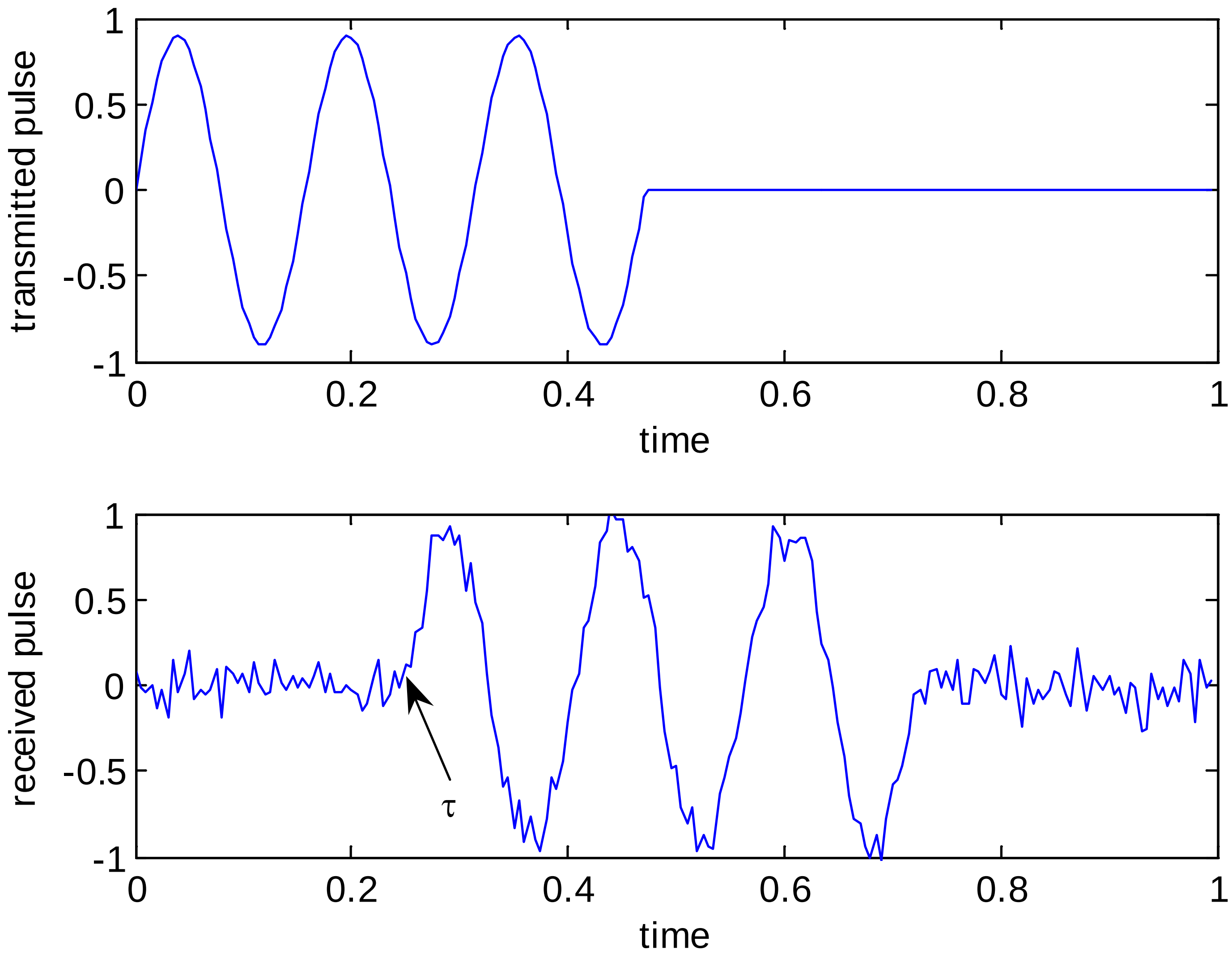


Fig.1.3: Transmitted & received radar waveforms

Radar transceiver sends a 1-D sinusoidal pulse at time 0

It then receives echo reflected by an object at a range of R

Reflected signal is noisy and has a time delay of τ which corresponds to round trip propagation time of radar pulse

Given the signal propagation speed, denoted by c, τ is simply related to R as:

$$\tau = \frac{2R}{c} \tag{1.1}$$

As a result, the radar pulse contains the object range information

- Can be a function of one, two or three independent variables, e.g., speech is 1-D signal, function of time; image is 2-D, function of space; wind is 3-D, function of latitude, longitude and elevation

- 3 types of signals that are functions of time:
  - Continuous-time (analog) $x(t)$: defined on a continuous range of time t, amplitude can be any value
  - Discrete-time $x(nT)$: defined only at discrete instants of time $t = \cdots - T, 0, T, 2T, \cdots$, amplitude can be any value
  - Digital (quantized) $x_Q(nT)$: both time and amplitude are discrete, i.e., it is defined only at $t = \cdots - T, 0, T, 2T, \cdots$ and amplitude is confined to a finite set of numbers

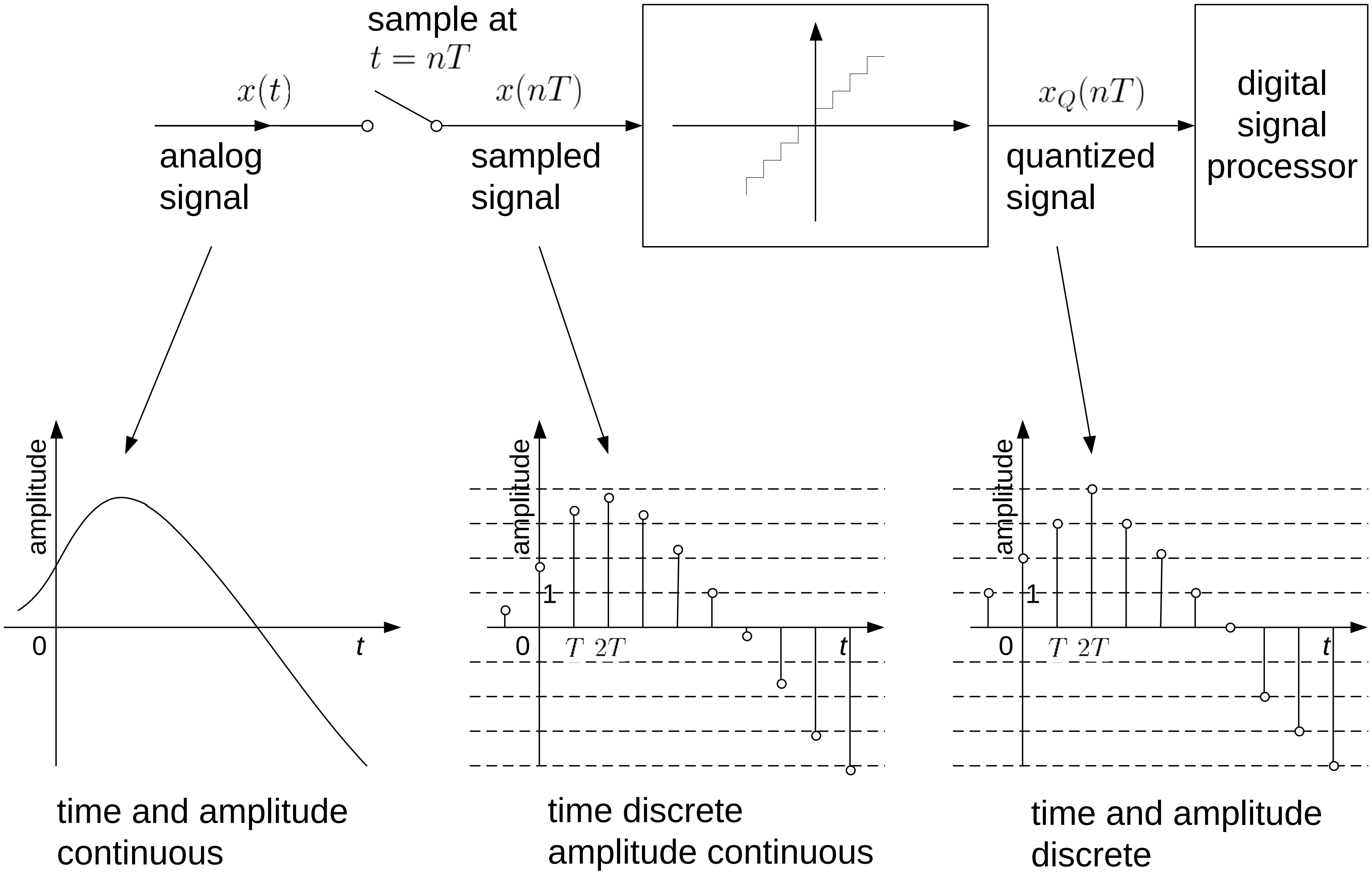


Fig. 1.4: Relationships between $x(t)$, $x(nT)$ and $x_Q(nT)$

$x(nT)$ at $n=0$ is close to 2 and $x_Q(0)=2$

$x(nT) \in (3,4)$ at $n=1$ and $x_Q(T)=3$

Using 4-bit representation, $x_Q(0)=0010$ and $x_Q(T)=0011$, and in general, the value of $x_Q(nT)$ is restricted to be an integer between -8 and 7 according to the two's complement representation.

In digital signal processing (DSP), we deal with $x_Q(nT)$ as it corresponds to computer-based processing. Throughout the course, it is assumed that discrete-time signal = digital signal, or the quantizer has infinite resolution

System:

- Mathematical model or abstraction of a physical process that relates input to output, e.g.,
  - Grading system: inputs are coursework and examination marks, output is grade
  - Squaring system: input is 5, then the output is 25
  - Amplifier: input is $\cos(\omega t)$, then output is $10\cos(\omega t)$
  - Communication system: input to mobile phone is voice, output from mobile phone is CDMA signal
  - Noise reduction system: input is a noisy speech, output is a noise-reduced speech
  - Feature extraction system: input is $\cos(\omega t)$, output is ω
- Any system that processes digital signals is called a digital system, digital filter or digital (signal) processor

Processing:

- Perform a particular function by passing a signal through system

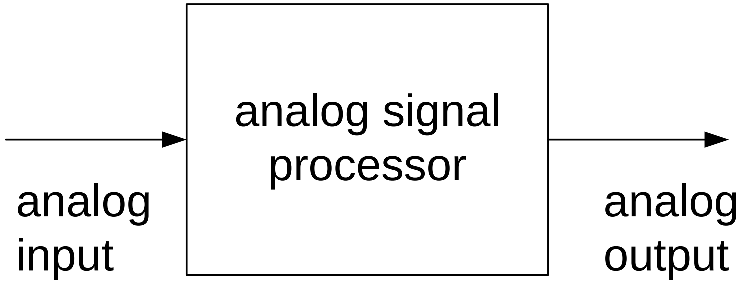


Fig.1.5: Analog processing of analog signal

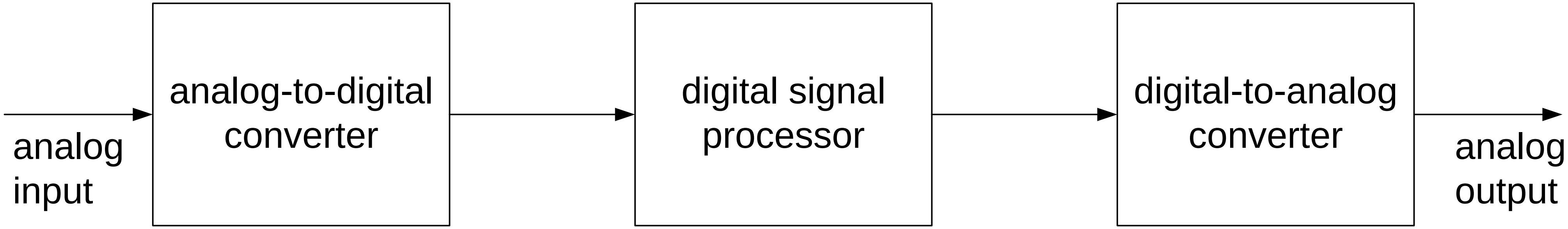


Fig.1.6: Digital processing of analog signal

Advantages of DSP over Analog Signal Processing

- Allow development with the use of PC, e.g., MATLAB
- Allow flexibility in reconfiguring the DSP operations simply by changing the program
- Reliable: processing of 0 and 1 is almost immune to noise and data are easily stored without deterioration
- Lower cost due to advancement of VLSI technology
- Security can be introduced by encrypting/scrambling
- Simple: additions and multiplications are main operations

DSP Application Areas

- Speech
  - Compression (e.g., LPC is a coding standard for compression of speech data)
  - Synthesis (computer production of speech signals, e.g., text-to-speech engine by Microsoft)
  - Recognition (e.g., automatic telephone number enquiry system)
  - Enhancement (e.g., noise reduction for a noisy speech)
- Audio
  - Compression (e.g., MP3 is a coding standard for compression of audio data)

- Generation of music by different musical instruments such as piano, cello, guitar and flute using computer
- Song with low-cost electronic piano keyboard quality
- Automatic music transcription (writing a piece of music down from a recording)

- Image and Video
  - Compression (e.g., JPEG and MPEG is are coding standards for image and video compression, respectively)
  - Recognition such as face, palm and fingerprint
  - Enhancement
  - Construction of 3-D objects from 2-D images
  - Animation, e.g., “Avatar”

- Communications: encoding and decoding of digital communication signals

- Astronomy: finding the periods of orbits

- Biomedical Engineering: medical care and diagnosis, analysis of ECG, electroencephalogram (EEG), nuclear magnetic resonance (NMR) data

- Bioinformatics: DNA sequence analysis, extracting, processing, and interpreting the information contained in genomic and proteomic data

- Finance: market risk management, trading algorithm design, investment portfolio analysis

Chapter 2: Review of Analog Signal Analysis

Chapter Intended Learning Outcomes:

(i) Review of Fourier series which is used to analyze continuous-time periodic signals

(ii) Review of Fourier transform which is used to analyze continuous-time aperiodic signals

(iii) Review of analog linear time-invariant system

Fourier series and Fourier transform are the tools for analyzing analog signals. Basically, they are used for signal conversion between time and frequency domains:

$$x(t) \leftrightarrow X(j\Omega) \tag{2.1}$$

Fourier Series

- For analysis of continuous-time periodic signals
- Express periodic signals using harmonically related sinusoids with frequencies $\cdots - \Omega_0, 0, \Omega_0, 2\Omega_0, \cdots$ where Ω_0 is called the fundamental frequency
- In the frequency domain, Ω only takes discrete values at $\cdots - \Omega_0, 0, \Omega_0, 2\Omega_0, \cdots$

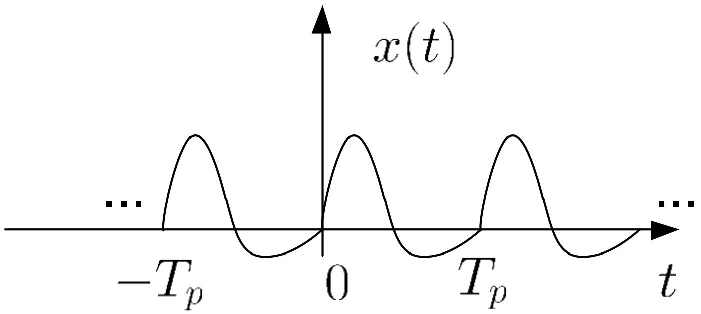
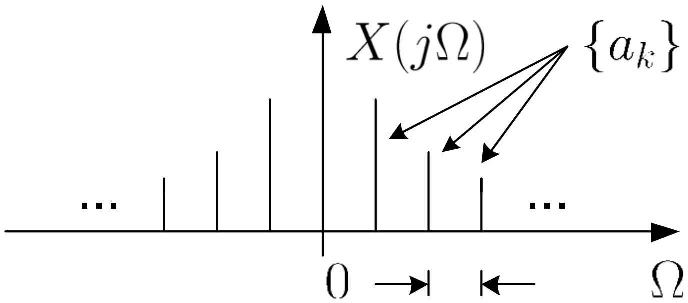
time domain	frequency domain
 $a_k = \frac{1}{T_p} \int_{-T_p/2}^{T_p/2} x(t) e^{-jk\Omega_0 t} dt \Longrightarrow$	 $\Omega_0 = \frac{2\pi}{T_p}$ $\Longleftarrow x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\Omega_0 t}$
continuous and periodic	discrete and aperiodic

Fig.2.1: Illustration of Fourier series

A continuous-time function $x(t)$ is said to be periodic if there exists $T_p > 0$ such that

$$x(t) = x(t + T_p), \qquad t \in (-\infty, \infty) \tag{2.2}$$

The smallest T_p for which (2.2) holds is called the fundamental period

The fundamental frequency is related to T_p as:

$$\Omega_0 = \frac{2\pi}{T_p} \tag{2.3}$$

Every periodic function can be expanded into a Fourier series as

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\Omega_0 t}, \qquad t \in (-\infty, \infty) \tag{2.4}$$

where

$$a_k = \frac{1}{T_p} \int_{-T_p/2}^{T_p/2} x(t) e^{-jk\Omega_0 t} dt, \qquad k = \cdots - 1, 0, 1, 2, \cdots \tag{2.5}$$

are called Fourier series coefficients

$X(j\Omega)$ is characterized by $\{a_k\}$, the Fourier series coefficients in fact correspond to the frequency representation of $x(t)$.

Generally, a_k is complex and we use magnitude and phase for its representation

$$|a_k| = \sqrt{(\Re\{a_k\})^2 + (\Im\{a_k\})^2} \tag{2.6}$$

and

$$\angle(a_k) = \tan^{-1}\left(\frac{\Im\{a_k\}}{\Re\{a_k\}}\right) \tag{2.7}$$

Example 2.1

Find the Fourier series coefficients for $x(t) = \cos(10\pi t) + \cos(20\pi t)$.

It is clear that the fundamental frequency of $x(t)$ is $\Omega_0 = 10\pi$. According to (2.3), the fundamental period is thus equal to $T_p = 2\pi/\Omega_0 = 1/5$, which is validated as follows:

$$\begin{aligned} x\left(t + \frac{1}{5}\right) &= \cos\left(10\pi\left(t + \frac{1}{5}\right)\right) + \cos\left(20\pi\left(t + \frac{1}{5}\right)\right) \\ &= \cos(10\pi t + 2\pi) + \cos(20\pi t + 4\pi) \\ &= \cos(10\pi t) + \cos(20\pi t) \end{aligned}$$

With the use of Euler formulas:

$$\cos(u) = \frac{e^{ju} + e^{-ju}}{2}$$

and

$$\sin(u) = \frac{e^{ju} - e^{-ju}}{2j}$$

we can express $x(t)$ as:

$$\begin{aligned} x(t) &= \cos(10\pi t) + \cos(20\pi t) \\ &= \frac{e^{j\Omega_0 t} + e^{-j\Omega_0 t}}{2} + \frac{e^{j2\Omega_0 t} + e^{-j2\Omega_0 t}}{2} \\ &= \frac{1}{2}e^{-j2\Omega_0 t} + \frac{1}{2}e^{-j\Omega_0 t} + \frac{1}{2}e^{j\Omega_0 t} + \frac{1}{2}e^{j2\Omega_0 t} \end{aligned}$$

By inspection and using (2.4), we have $a_{-2} = a_{-1} = a_1 = a_2 = 1/2$ while all other Fourier series coefficients are equal to zero

Example 2.2

Find the Fourier series coefficients for $x(t) = 1 + \sin(\Omega_0 t) + 2\cos(\Omega_0 t) + \cos(3\Omega_0 t + \pi/4)$.

Can we use (2.5)? Why?

With the use of Euler formulas, $x(t)$ can be written as:

$$\begin{aligned} x(t) &= 1 + \left(1 + \frac{1}{2j}\right) e^{j\Omega_0 t} + \left(1 - \frac{1}{2j}\right) e^{-j\Omega_0 t} + \frac{1}{2} e^{j\pi/4} e^{3j\Omega_0 t} + \frac{1}{2} e^{-j\pi/4} e^{-3j\Omega_0 t} \\ &= \frac{\sqrt{2}}{4}(1-j)e^{-3j\Omega_0 t} + \left(1 + j\frac{1}{2}\right) e^{-j\Omega_0 t} + 1 + \left(1 - j\frac{1}{2}\right) e^{j\Omega_0 t} \\ &\quad + \frac{\sqrt{2}}{4}(1+j)e^{3j\Omega_0 t} \end{aligned}$$

Using (2.4), we have:

$$a_k = \begin{cases} \frac{\sqrt{2}}{4}(1-j), & k=-3 \\ 1+\frac{j}{2}, & k=-1 \\ 1, & k=0 \\ 1-\frac{j}{2}, & k=1 \\ \frac{\sqrt{2}}{4}(1+j), & k=3 \\ 0, & \text{otherwise} \end{cases}$$

To plot $\{a_k\}$, we need to compute $|a_k|$ and $\angle(a_k)$ for all k, e.g.,

$$|a_{-3}| = \sqrt{\left(\frac{\sqrt{2}}{4}\right)^2 + \left(-\frac{\sqrt{2}}{4}\right)^2} = \frac{1}{2}$$

and

$$\angle(a_{-3}) = \tan^{-1}(-1) = -\frac{\pi}{4}$$

Example 2.3

Find the Fourier series coefficients for $x(t)$, which is a periodic continuous-time signal of fundamental period T and is a pulse with a width of $2T_0$ in each period. Over the specific period from $-T/2$ to $T/2$, $x(t)$ is:

$$x(t) = \begin{cases} 1, & -T_0 < t < T_0 \\ 0, & \text{otherwise} \end{cases}$$

with $T > 2T_0$. According to (2.3), the fundamental frequency is $\Omega_0 = 2\pi/T$. Using (2.5), we get:

$$a_k = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-jk\Omega_0 t} dt = \frac{1}{T} \int_{-T_0}^{T_0} e^{-jk\Omega_0 t} dt$$

For $k=0$:

$$a_0 = \frac{1}{T}\int_{-T_0}^{T_0} 1dt = \frac{2T_0}{T}$$

For $k \neq 0$:

$$a_k = \frac{1}{T}\int_{-T_0}^{T_0} e^{-jk\Omega_0 t}dt = -\frac{1}{jk\Omega_0}e^{jk\Omega_0 t}\Big|_{-T_0}^{T_0} = \frac{\sin(k\Omega_0 T_0)}{k\pi} = \frac{\sin\left(2\pi k T_0/T\right)}{k\pi}$$

The reason of separating the cases of $k=0$ and $k \neq 0$ is to facilitate the computation of a_0 , whose value is not straightforwardly obtained from the general expression which involves “0/0”. Nevertheless, using $\mathrm{L'H\hat{o}pital}$’s rule:

$$\lim_{k\to 0}\frac{\sin\left(2\pi k T_0/T\right)}{k\pi} = \lim_{k\to 0}\frac{\frac{d\sin\left(2\pi k T_0/T\right)}{dk}}{\frac{dk\pi}{dk}} = \lim_{k\to 0}\frac{2\pi T_0/T\cos((2\pi k T_0/T))}{\pi} = \frac{2T_0}{T}$$

In summary, if a signal $x(t)$ is continuous in time and periodic, we can write:

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\Omega_0 t}, \qquad t \in (-\infty, \infty) \tag{2.4}$$

The basic steps for finding the Fourier series coefficients are:

1. Determine the fundamental period T_p and fundamental frequency Ω_0

2. For all k, multiply $x(t)$ by $e^{-jk\Omega_0 t}$, then integrate with respect to t for one period, finally divide the result by T_p. Usually we separate the calculation into two cases: $k=0$ and $k \neq 0$

Fourier Transform

- For analysis of continuous-time aperiodic signals
- Defined on a continuous range of Ω

The Fourier transform of an aperiodic and continuous-time signal $x(t)$ is:

$$X(j\Omega) = \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt \qquad (2.8)$$

which is also called spectrum. The inverse transform is given by

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\Omega) e^{j\Omega t} d\Omega \qquad (2.9)$$

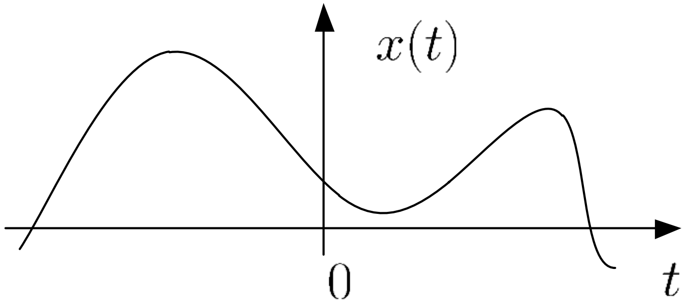
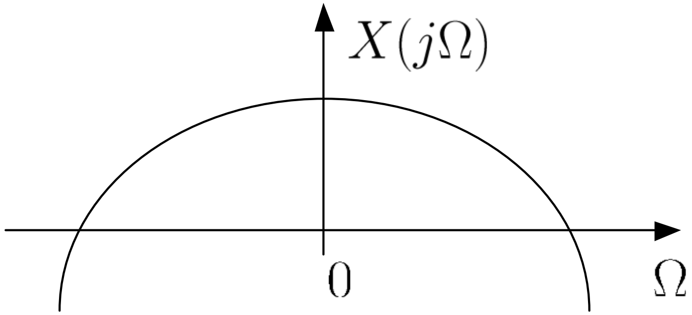
time domain	frequency domain
 $X(j\Omega)=\int\limits_{-\infty}^{\infty} x(t)e^{-j\Omega t}dt \Longrightarrow$	 $\Longleftarrow x(t)=\frac{1}{2\pi}\int\limits_{-\infty}^{\infty} X(j\Omega)e^{j\Omega t}d\Omega$
continuous and aperiodic	continuous and aperiodic

Fig.2.2: Illustration of Fourier transform

The delta function $\delta(t)$ has the following characteristics:

$$\delta(t)=0, \quad t \neq 0 \tag{2.10}$$

$$\int_{-\infty}^{\infty} \delta(t)dt = 1 \tag{2.11}$$

and

$$f(t)\delta(t-t_0) = f(t_0)\delta(t-t_0) \tag{2.12}$$

where $f(t)$ is a continuous-time signal.

(2.10) and (2.11) indicate that $\delta(t)$ has a very large value or impulse at $t=0$. That is, $\delta(t)$ is not well defined at $t=0$

(2.12) is known as the sifting property

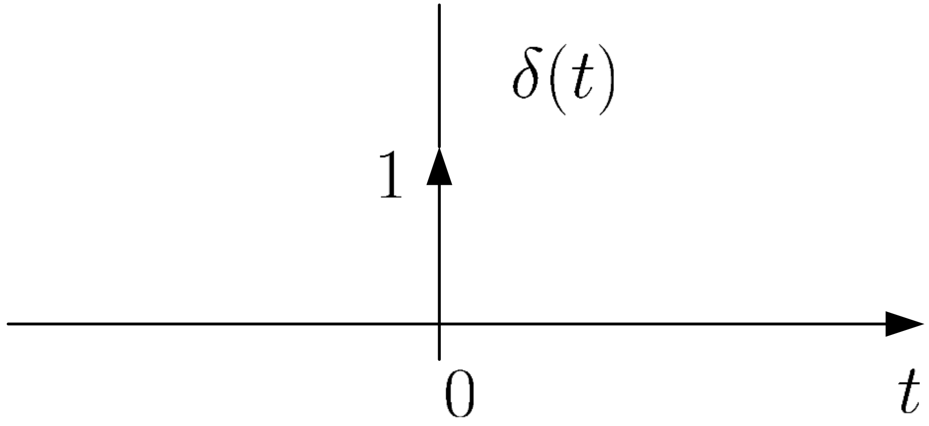


Fig.2.3: Representation of $\delta(t)$

The unit step function $u(t)$ has the form of:

$$u(t) = \begin{cases} 1, & t > 0 \\ 0, & t < 0 \end{cases} \tag{2.13}$$

As there is a sudden change from 0 to 1 at $t = 0$, $u(0)$ is not well defined

Example 2.4

Find the Fourier transform of $x(t)$ which is a rectangular pulse of the form:

$$x(t) = \begin{cases} 1, & -T_0 < t < T_0 \\ 0, & \text{otherwise} \end{cases}$$

Note that the signal is of finite length and corresponds to one period of the periodic function in Example 2.3. Applying (2.8) on $x(t)$ yields:

$$X(j\Omega) = \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt = \int_{-T_0}^{T_0} e^{-j\Omega t} dt = \frac{2\sin(\Omega T_0)}{\Omega}$$

Define the sinc function as:

$$\mathrm{sinc}(u) = \frac{\sin(\pi u)}{\pi u}$$

It is seen that $X(j\Omega)$ is a scaled sinc function because

$$X(j\Omega) = \frac{2\sin(\Omega T_0)}{\Omega} = 2T_0\mathrm{sinc}\left(\frac{\Omega T_0}{\pi}\right)$$

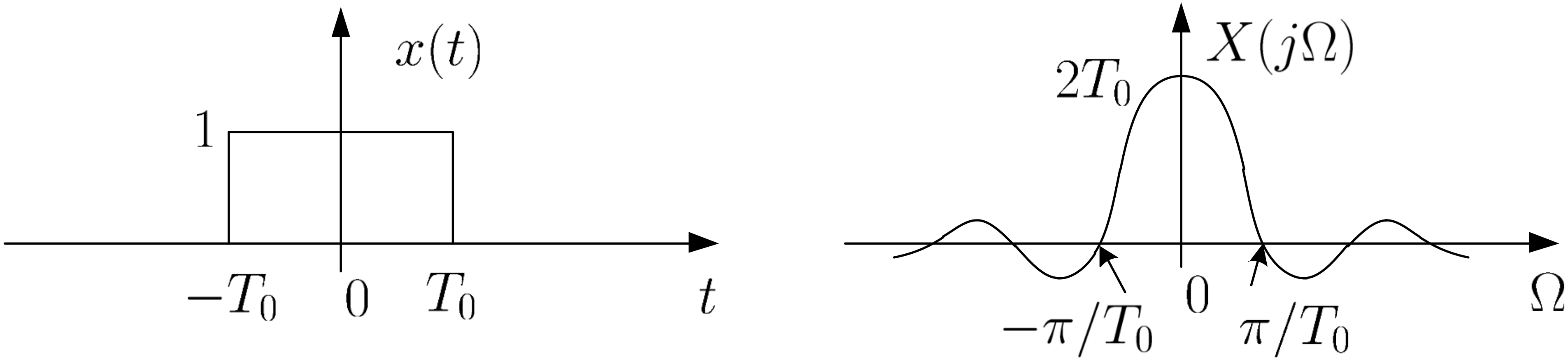


Fig.2.4: Fourier transform pair for rectangular pulse of $x(t)$

Example 2.5

Find the inverse Fourier transform of $X(j\Omega)$ which is a rectangular pulse of the form:

$$X(j\Omega) = \begin{cases} 1, & -W_0 < \Omega < W_0 \\ 0, & \text{otherwise} \end{cases}$$

Using (2.9), we get:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\Omega) e^{j\Omega t} d\Omega = \frac{1}{2\pi} \int_{-W_0}^{W_0} e^{j\Omega t} d\Omega = \frac{\sin(W_0 t)}{\pi t}$$

$$= \frac{W_0}{\pi} \text{sinc}\left(\frac{W_0 t}{\pi}\right)$$

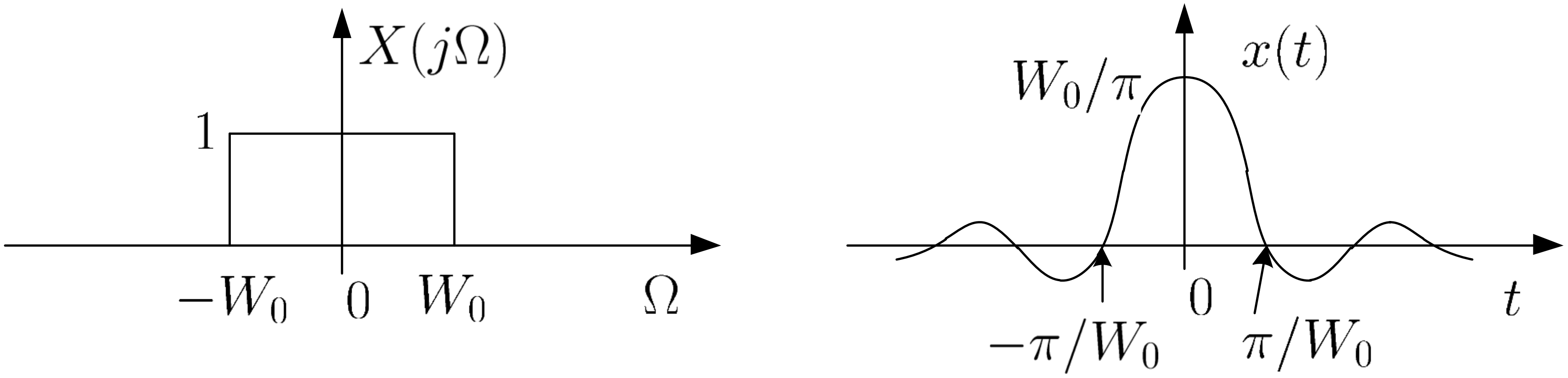


Fig.2.5: Fourier transform pair for rectangular pulse of $X(j\Omega)$

From Examples 2.4 and 2.5, we observe the duality property of Fourier transform

Can you guess why we have the duality property?

Example 2.6

Find the Fourier transform of $x(t)=e^{-at}u(t)$ with $a>0$.

Employing the property of $u(t)$ in (2.13) and (2.8), we get:

$$X(j\Omega)=\int_0^\infty e^{-at}e^{-j\Omega t}dt=-\frac{1}{a+j\Omega}e^{-(a+j\Omega)t}\bigg|_0^\infty=\frac{1}{a+j\Omega}=\frac{a-j\Omega}{a^2+\Omega^2}$$

Note that when $t\to\infty$, $e^{-at}\to 0$

$$|X(j\Omega)|=\frac{1}{\sqrt{a^2+\Omega^2}}$$

and

$$\angle(X(j\Omega))=-\tan^{-1}\left(\frac{\Omega}{a}\right)$$

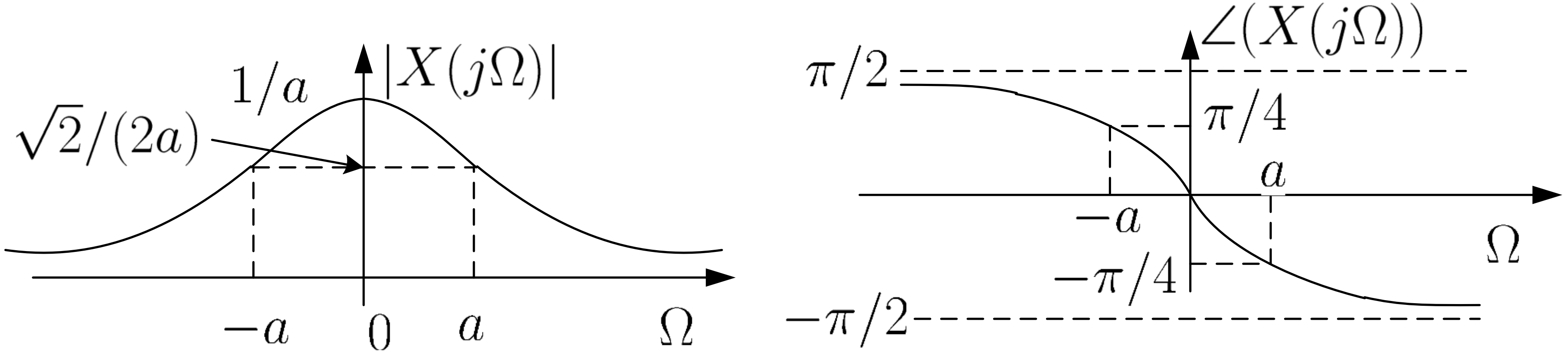


Fig.2.6: Magnitude and phase plots for $1/(a+j\Omega)$

Example 2.7

Find the Fourier transform of the delta function $x(t) = \delta(t)$.

Using (2.11) and (2.12) with $f(t) = e^{-j\Omega t}$ and $t_0 = 0$, we get:

$$X(j\Omega) = \int_{-\infty}^{\infty} \delta(t) e^{-j\Omega t} dt = \int_{-\infty}^{\infty} \delta(t) e^{-j\Omega \cdot 0} dt = e^{-j\Omega \cdot 0} \int_{-\infty}^{\infty} \delta(t) dt = e^{-j\Omega \cdot 0} = 1$$

Spectrum of $\delta(t)$ has unit amplitude at all frequencies

Based on $\delta(t)$, Fourier transform can be used to represent continuous-time periodic signals. Consider

$$X(j\Omega) = 2\pi\delta(\Omega - \Omega_0) \qquad (2.14)$$

Taking the inverse Fourier transform of $X(j\Omega)$ and employing Example 2.7, $x(t)$ is computed as:

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} 2\pi\delta(\Omega - \Omega_0)e^{j\Omega t}d\Omega = e^{j\Omega_0 t} \qquad (2.15)$$

As a result, the Fourier transform pair is:

$$e^{j\Omega_0 t} \leftrightarrow 2\pi\delta(\Omega - \Omega_0) \qquad (2.16)$$

From (2.4) and (2.16), the Fourier transform pair for a continuous-time periodic signal is:

$$\sum_{k=-\infty}^{\infty} a_k e^{jk\Omega_0 t} \leftrightarrow \sum_{k=-\infty}^{\infty} 2\pi a_k \delta(\Omega - k\Omega_0) \qquad (2.17)$$

Example 2.8

Find the Fourier transform of $x(t)=\sum_{k=-\infty}^{\infty}\delta(t-kT)$ which is called an impulse train.

Clearly, $x(t)$ is a periodic signal with a period of T. Using (2.5) and Example 2.7, the Fourier series coefficients are:

$$a_k=\frac{1}{T}\int_{-T/2}^{T/2}\delta(t)e^{-jk\Omega_0 t}dt=\frac{1}{T}$$

with $\Omega_0=2\pi/T$. According to (2.17), the Fourier transform is:

$$X(j\Omega)=\frac{2\pi}{T}\sum_{k=-\infty}^{\infty}\delta\left(\Omega-\frac{2\pi k}{T}\right)=\Omega_0\sum_{k=-\infty}^{\infty}\delta(\Omega-k\Omega_0)$$

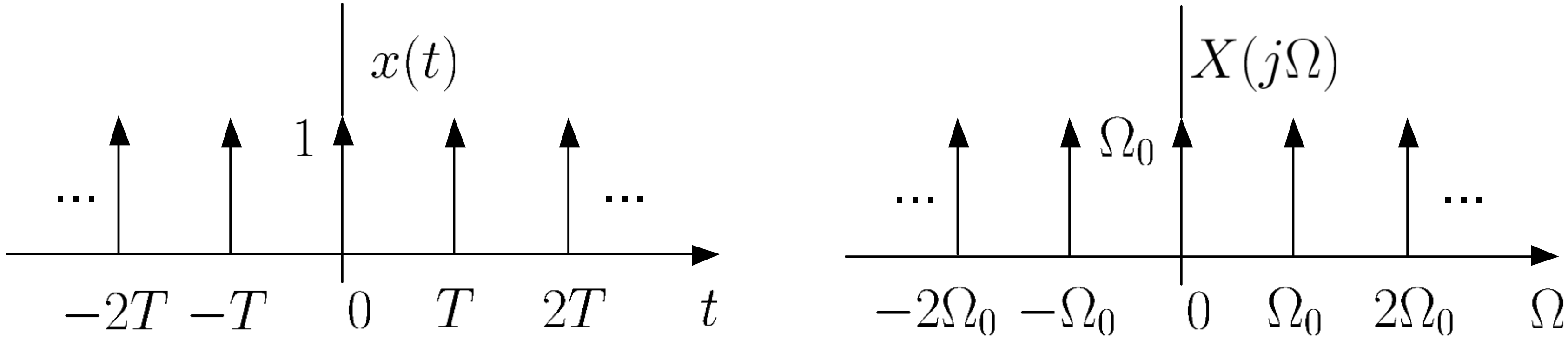


Fig.2.7: Fourier transform pair for impulse train

Fourier transform can be derived from Fourier series:

Consider $x(t)$ and $\tilde{x}(t)$:

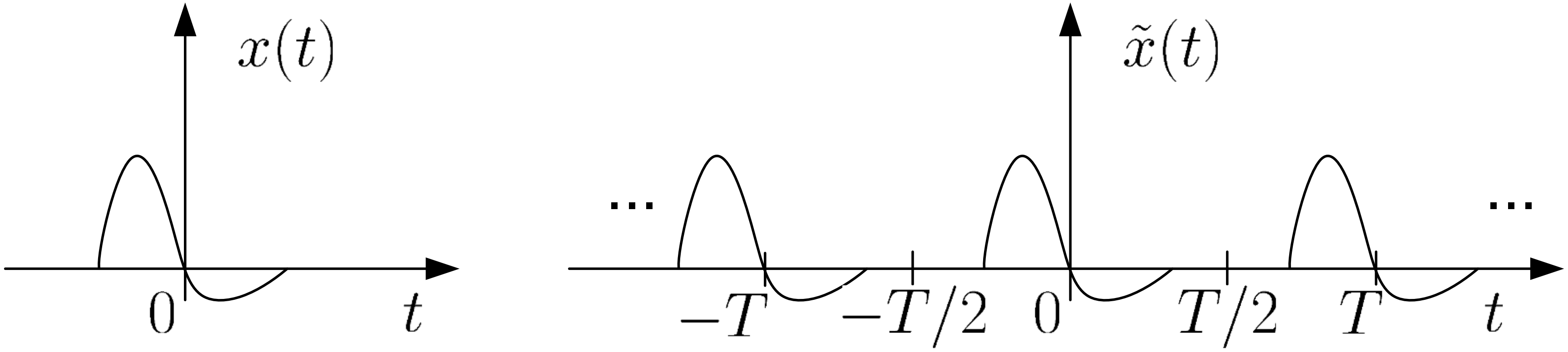


Fig.2.8: Constructing $\tilde{x}(t)$ from $x(t)$

$\tilde{x}(t)$ is constructed as a periodic version of $x(t)$, with period T

According to (2.5), the Fourier series coefficients of $\tilde{x}(t)$ are:

$$a_k = \frac{1}{T} \int_{-T/2}^{T/2} \tilde{x}(t) e^{-jk\Omega_0 t} dt \tag{2.18}$$

where $\Omega_0 = 2\pi/T$. Noting that $x(t) = \tilde{x}(t)$ for $|t| < T/2$ and $x(t) = 0$ for $|t| > T/2$, (2.18) can be expressed as:

$$a_k = \frac{1}{T} \int_{-T/2}^{T/2} x(t) e^{-jk\Omega_0 t} dt = \frac{1}{T} \int_{-\infty}^{\infty} x(t) e^{-jk\Omega_0 t} dt \tag{2.19}$$

According to (2.8), we can express a_k as:

$$a_k = \frac{1}{T} X(jk\Omega_0) \tag{2.20}$$

The Fourier series expansion for $\tilde{x}(t)$ is thus:

$$\tilde{x}(t) = \sum_{k=-\infty}^{\infty} \frac{1}{T} X(jk\Omega_0) e^{jk\Omega_0 t} = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \Omega_0 X(jk\Omega_0) e^{jk\Omega_0 t} \tag{2.21}$$

Considering $\tilde{x}(t) \to x(t)$ as $T \to \infty$ or $\Omega_0 \to 0$ and $\Omega_0 X(jk\Omega_0) e^{jk\Omega_0 t}$ as the area of a rectangle whose height is $X(jk\Omega_0) e^{jk\Omega_0 t}$ and width corresponds to the interval of $[k\Omega_0, (k+1)\Omega_0]$, we obtain

$$x(t) = \lim_{\Omega_0 \to 0} \tilde{x}(t) = \lim_{\Omega_0 \to 0} \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \Omega_0 X(jk\Omega_0) e^{jk\Omega_0 t} = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\Omega) e^{j\Omega t} d\Omega \tag{2.22}$$

Linear Time-Invariant (LTI) System

- Linearity: if $(x_1(t), y_1(t))$ and $(x_2(t), y_2(t))$ are two input-output pairs, then $ax_1(t) + bx_2(t) \rightarrow ay_1(t) + by_2(t)$

- Time-Invariance: if $x(t) \rightarrow y(t)$, then $x(t-t_0) \rightarrow y(t-t_0)$

- The input-output relationship for a LTI system is characterized by convolution:

$$y(t) = x(t) \otimes h(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau \qquad (2.23)$$

where $x(t)$, $y(t)$ and $h(t)$ are input, output and impulse response, respectively

- Convolution in time domain corresponds to multiplication in Fourier transform domain, i.e.,

$$x(t) \otimes h(t) \leftrightarrow X(j\Omega)H(j\Omega) \qquad (2.24)$$

Proof:

The Fourier transform of $x(t) \otimes h(t)$ is

$$\begin{aligned} Y(j\Omega) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x(\tau)h(t-\tau)e^{-j\Omega t}d\tau dt \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x(\tau)h(u)e^{-j\Omega\tau}e^{-j\Omega u}d\tau du, \quad u = t-\tau \\ &= \left[\int_{-\infty}^{\infty} x(\tau)e^{-j\Omega\tau}d\tau\right]\cdot\left[\int_{-\infty}^{\infty} h(u)e^{-j\Omega u}du\right] \\ &= X(j\Omega)\cdot H(j\Omega) \end{aligned} \tag{2.25}$$

This suggests that $y(t)$ can be computed from inverse Fourier transform of $X(j\Omega)H(j\Omega)$.

Chapter 3: Discrete-Time Signals and Systems

Chapter Intended Learning Outcomes:

(i) Understanding deterministic and random discrete-time signals and ability to generate them

(ii) Ability to recognize the discrete-time system properties, namely, memorylessness, stability, causality, linearity and time-invariance

(iii) Understanding discrete-time convolution and ability to perform its computation

(iv) Understanding the relationship between difference equations and discrete-time signals and systems

Discrete-Time Signal

- Discrete-time signal can be generated using a computing software such as MATLAB
- It can also be obtained from sampling continuous-time signals in real world

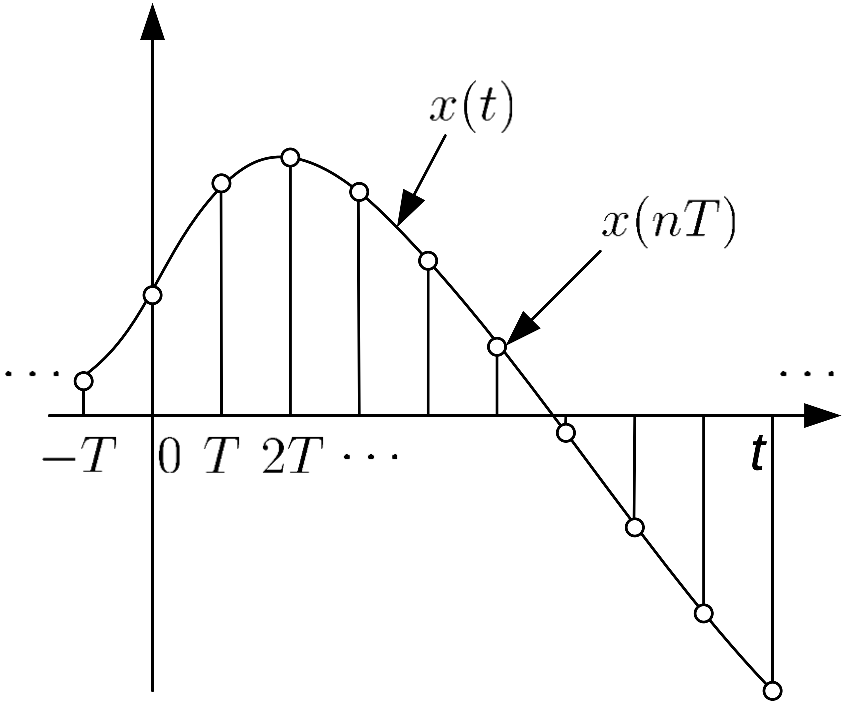


Fig.3.1:Discrete-time signal obtained from analog signal

- The discrete-time signal $x(nT)$ is equal to $x(t)$ only at the sampling interval of $t = nT$, $n = \cdots - 1, 0, 1, 2, \cdots$

$$x[n] = x(t)|_{t=nT} = x(nT), \quad n = \cdots - 1, 0, 1, 2, \cdots \tag{3.1}$$

where T is called the sampling period

- $x[n]$ is a sequence of numbers, $\cdots x[-1], x[0], x[1], x[2], \cdots$, with n being the time index

Basic Sequences

- Unit Sample (or Impulse)

$$\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \tag{3.2}$$

It is similar to the continuous-time unit impulse $\delta(t)$ which is defined in (2.10)-(2.12)

$\delta[n]$ is simpler than $\delta(t)$ because it is well defined for all n while $\delta(t)$ is not defined at $t=0$

- Unit Step

$$u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \qquad (3.3)$$

It is similar to to the continuous-time $u(t)$ of (2.13)

$u[n]$ is well defined for all n but $u(t)$ is not defined $t=0$.

$\delta[n]$ is an important function because it serves as the building block of any discrete-time signal $x[n]$:

$$\begin{aligned} x[n] &= \cdots + x[-1]\delta[n+1] + x[0]\delta[n] + x[1]\delta[n-1] + \cdots \\ &= \sum_{k=-\infty}^{\infty} x[k]\delta[n-k] \end{aligned} \qquad (3.4)$$

For example, $u[n]$ can be expressed in terms of $\delta[n]$ as:

$$u[n] = \sum_{k=0}^{\infty} \delta[n-k] \qquad (3.5)$$

Conversely, we can use $u[n]$ to represent $\delta[n]$:

$$\delta[n] = u[n] - u[n-1] \qquad (3.6)$$

Introduction to MATLAB

- MATLAB stands for ”Matrix Laboratory”
- Interactive matrix-based software for numerical and symbolic computation in scientific and engineering applications
- Its user interface is relatively simple to use, e.g., we can use the `help` command to understand the usage and syntax of each MATLAB function
- Together with the availability of numerous toolboxes, there are many useful and powerful commands for various disciplines
- MathWorks offers MATLAB to C conversion utility
- Similar packages include Maple and Mathematica

Discrete-Time Signal Generation using MATLAB

A deterministic discrete-time signal $x[n]$ satisfies a generating model with known functional form:

$$x[n] = f(\boldsymbol{\psi}, n) \tag{3.7}$$

where $f(\cdot)$ is a function of parameter vector $\boldsymbol{\psi}$ and time index n. That is, given $f(\cdot)$ and $\boldsymbol{\psi}$, $x[n]$ can be produced

e.g., the time-shifted unit sample $\delta[n-n_0]$ and unit step function $u[n-n_0]$, where the parameter is n_0

e.g., for an exponential function α^{n-n_0}, we have $\boldsymbol{\psi} = [\alpha \quad n_0]$ where α is the decay factor and n_0 is the time shift

e.g., for a sinusoid $A\cos(\omega n + \theta)$, we have $\boldsymbol{\psi} = [A \quad \omega \quad \theta]$

Example 3.1

Use MATLAB to generate a discrete-time sinusoid of the form:

$$x[n] = A\cos(\omega n + \theta), \quad n = 0, 1, \cdots, N-1$$

with $A = 1$, $\omega = 0.3$, $\theta = 1$ and $N = 21$, which has a duration of 21 samples

We can generate $x[n]$ by using the following MATLAB code:

```
N=21;                   %number of samples is 21
A=1;                    %tone amplitude is 1
w=0.3;                  %frequency is 0.3
p=1;                    %phase is 1
for n=1:N
x(n)=A*cos(w*(n-1)+p); %time index should be >0
end
```

Note that x is a vector and its index should be at least 1.

Alternatively, we can also use:

```
N=21;                      %number of samples is 21
A=1;                       %tone amplitude is 1
w=0.3;                     %frequency is 0.3
p=1;                       %phase is 1
n=0:N-1;                   %define time index vector
x=A.*cos(w.*n+p);          %first time index is also 1
```

Both give

```
x =
Columns 1 through 7
0.5403 0.2675 -0.0292 -0.3233 -0.5885 -0.8011 -0.9422
Columns 8 through 14
-0.9991 -0.9668 -0.8481 -0.6536 -0.4008 -0.1122 0.1865
Columns 15 through 21
0.4685 0.7087 0.8855 0.9833 0.9932 0.9144 0.7539
```

Which approach is better? Why?

To plot $x[n]$, we can either use the commands `stem(x)` and `plot(x)`

If the time index is not specified, the default start time is $n=1$

Nevertheless, it is easy to include the time index vector in the plotting command

e.g., Using `stem` to plot $x[n]$ with the correct time index:

```
n=0:N-1;                  %n is vector of time index
stem(n,x)                 %plot x versus n
```

Similarly, `plot(n,x)` can be employed to show $x[n]$

The MATLAB programs for this example are provided as `ex3_1.m` and `ex3_1_2.m`

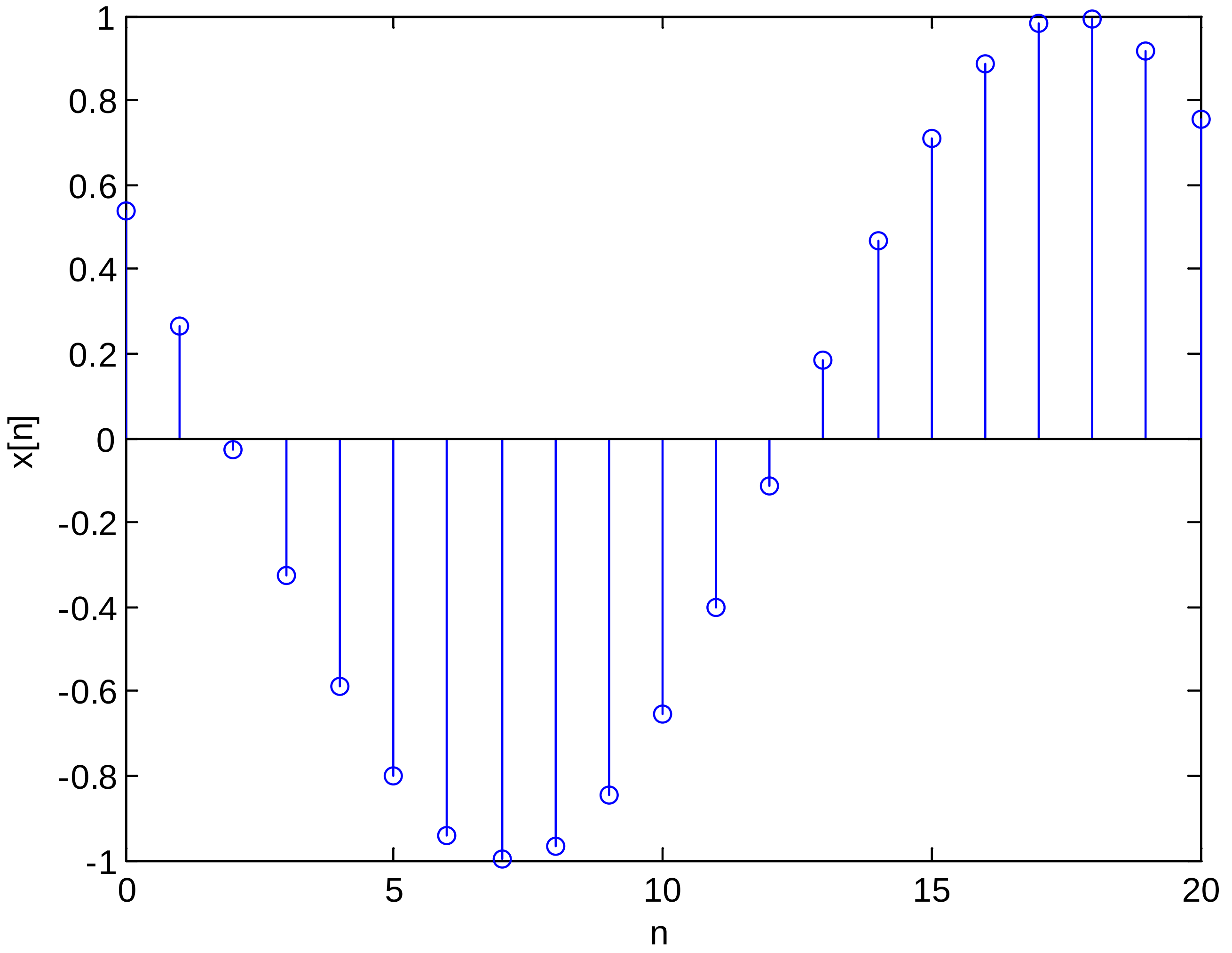


Fig.3.2: Plot of discrete-time sinusoid using stem

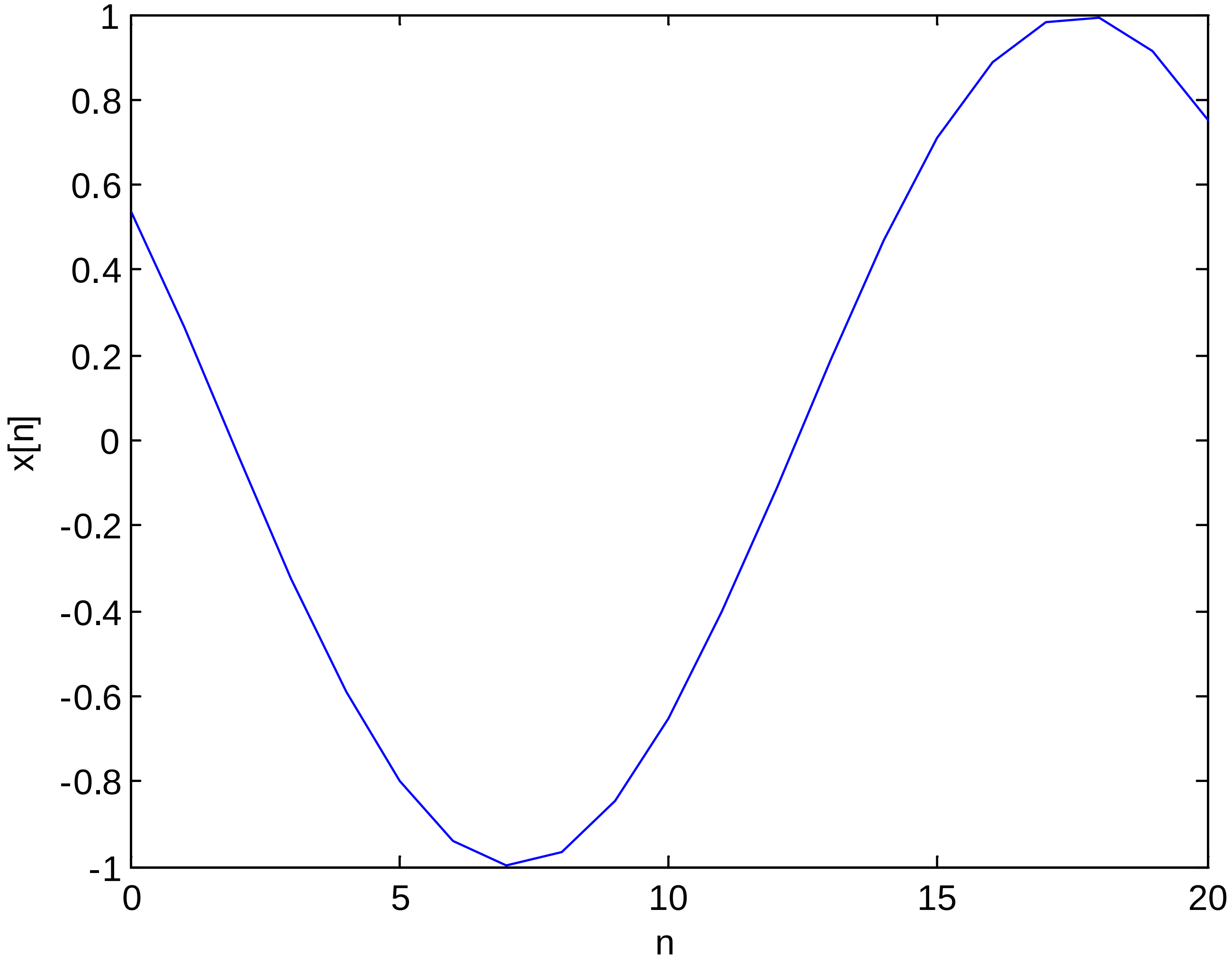


Fig.3.3: Plot of discrete-time sinusoid using plot

Apart from deterministic signal, random signal is another importance signal class. It cannot be described by mathematical expressions like deterministic signals but is characterized by its probability density function (PDF). MATLAB has commands to produce two common random signals, namely, uniform and Gaussian (normal) variables.

A uniform integer sequence $x[n]$ whose values are uniformly distributed between 0 and m, can be generated using:

$$x[n] = (a \cdot x[n-1]) \mod (m) \tag{3.8}$$

where a and m are very large positive integers, $x[n]$ is the reminder of dividing $a \cdot x[n-1]$ by m

Each admissible value of $0 \leq x[n] < m$ has the same probability of occurrence of approximately $1/m$

We also need an initial integer or seed, say, $x[-1] \in [0, m-1]$, for starting the generation of $x[n]$

(3.8) can be easily modified by properly scaling and shifting

e.g., a random number which is uniformly between –0.5 and 0.5, denoted by $y[n]$, is obtained from $x[n]$:

$$y[n] = \frac{x[n]}{m} - 0.5 \tag{3.9}$$

The MATLAB command `rand` is used to generate random numbers which are uniformly between 0 and 1

e.g., each realization of `stem(0:20,rand(1,21))` gives a distinct and random sequence, with values are bounded between 0 and 1

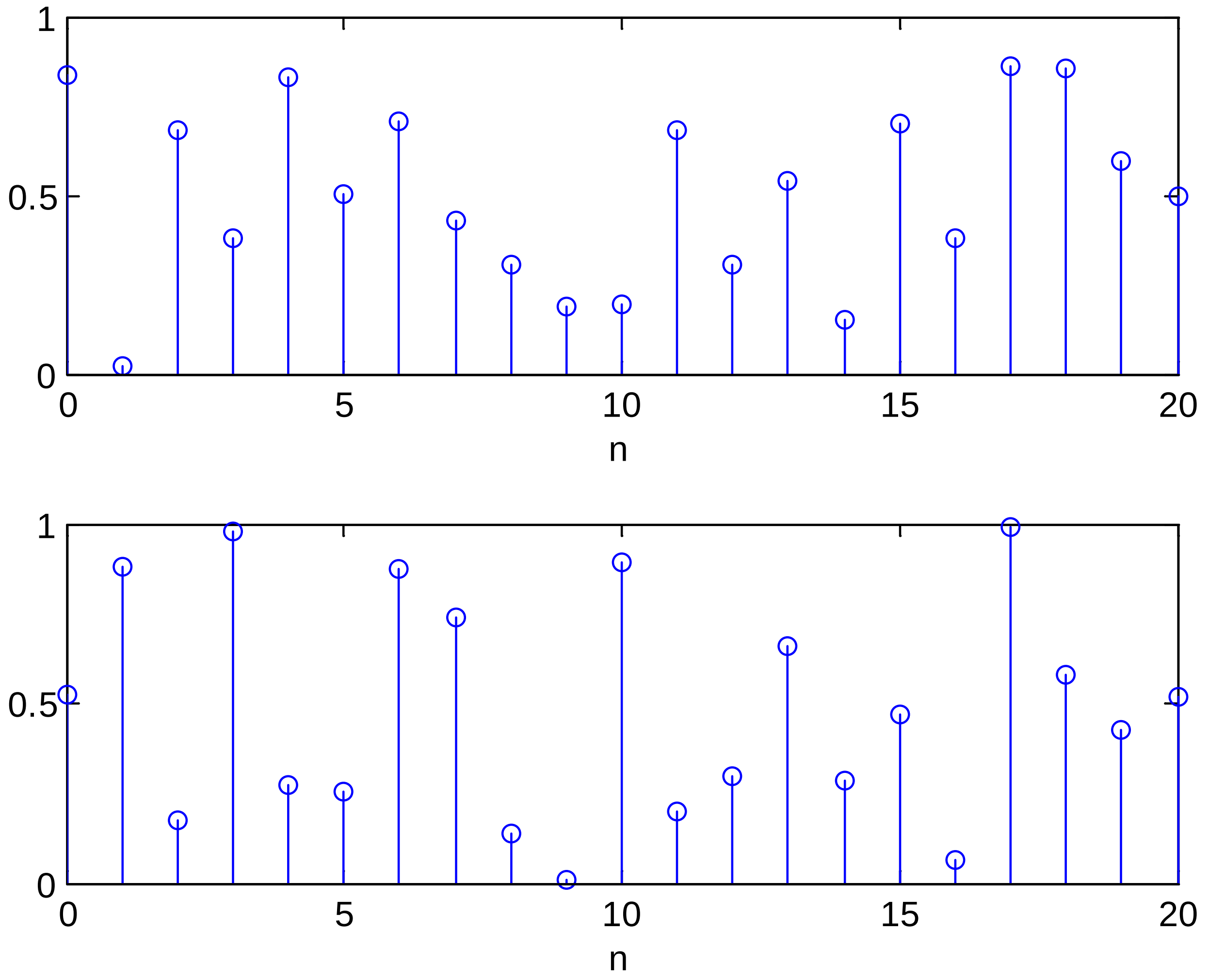


Fig.3.4: Uniform number realizations using `rand`

Example 3.2

Use MATLAB to generate a sequence of 10000 random numbers uniformly distributed between –0.5 and 0.5 based on the command `rand`. Verify its characteristics.

According to (3.9), we use `u=rand(1,10000)-0.5` to generate the sequence

To verify the uniform distribution, we use `hist(u,10)`, which bins the elements of `u` into 10 equally-spaced containers

We see all numbers are bounded between –0.5 and 0.5, and each bar which corresponds to a range of 0.1, contains approximately 1000 elements.

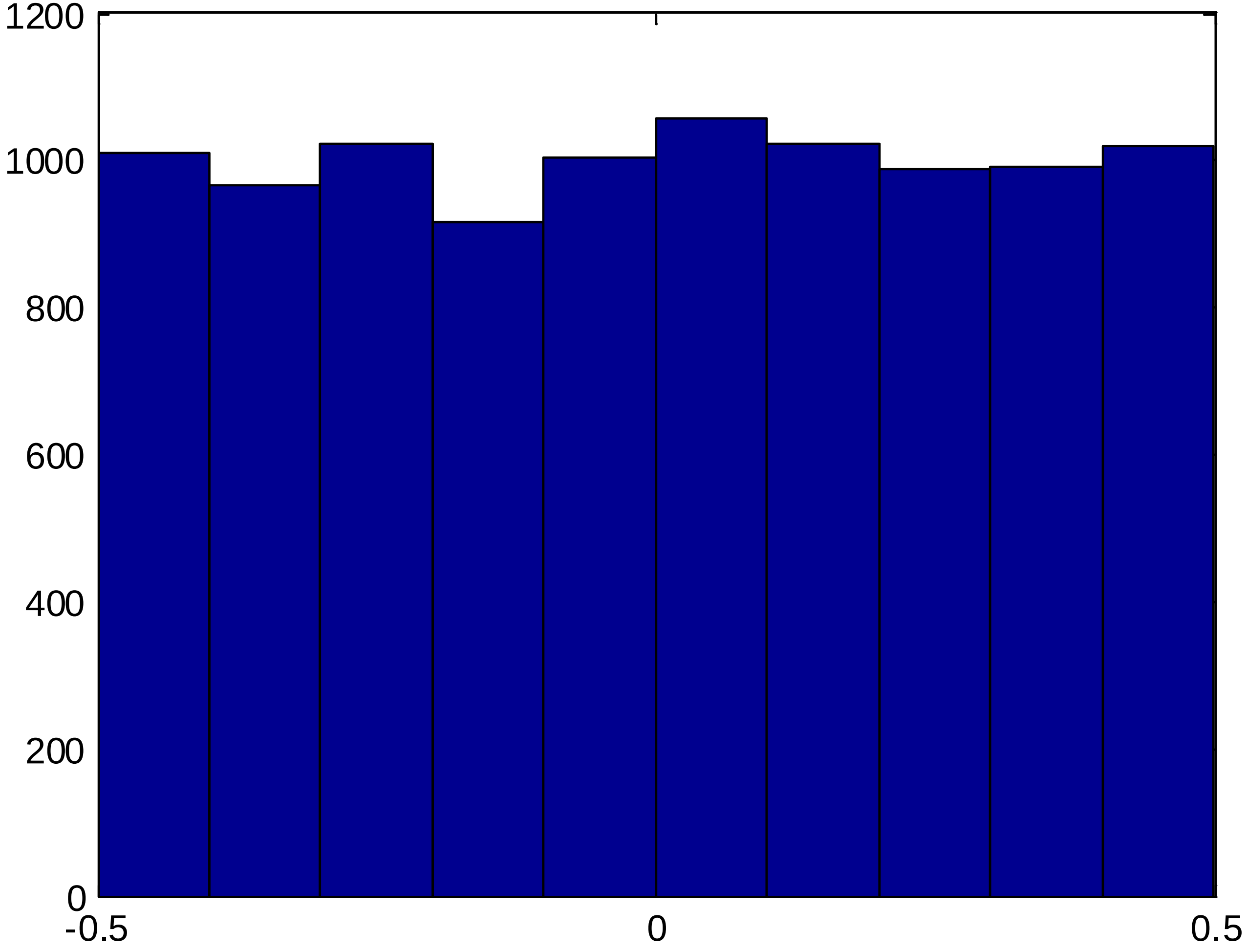


Fig.3.5: Histogram for uniform sequence

On the other hand, the PDF of u, denoted by $p(u)$, is

$$p(u) = \begin{cases} 1, & -0.5 \leq u \leq 0.5 \\ 0, & \text{otherwise} \end{cases}$$

such that $\int_{-\infty}^{\infty} p(u)du = 1$. The theoretical mean and power of u, are computed as

$$E\{u\} = \int_{-\infty}^{\infty} up(u)du = \int_{-0.5}^{0.5} udu = \frac{1}{2}u^2\Big|_{-0.5}^{0.5} = 0$$

and

$$E\{u^2\} = \int_{-\infty}^{\infty} u^2p(u)du = \int_{-0.5}^{0.5} u^2p(u)du = \frac{1}{3}u^3\Big|_{-0.5}^{0.5} = \frac{1}{12}$$

Average value and power of u in this realization are computed using `mean(u)` and `mean(u.*u)`, which give 0.002 and 0.0837, and they align with theoretical calculations

Gaussian numbers can be generated from the uniform variables

Given a pair of independent random numbers uniformly distributed between 0 and 1, (u_1, u_2) , a pair of independent Gaussian numbers (w_1, w_2), which have zero mean and unity power (or variance), can be generated from:

$$w_1 = \sqrt{-2\ln(u_1)} \cdot \cos(2\pi u_2) \tag{3.10}$$

and

$$w_2 = \sqrt{-2\ln(u_1)} \cdot \sin(2\pi u_2) \tag{3.11}$$

The MATLAB command is randn. Equations (3.10) and (3.11) are known as the Box-Mueller transformation

e.g., each realization of `stem(0:20,randn(1,21))` gives a distinct and random sequence, whose values are fluctuating around zero

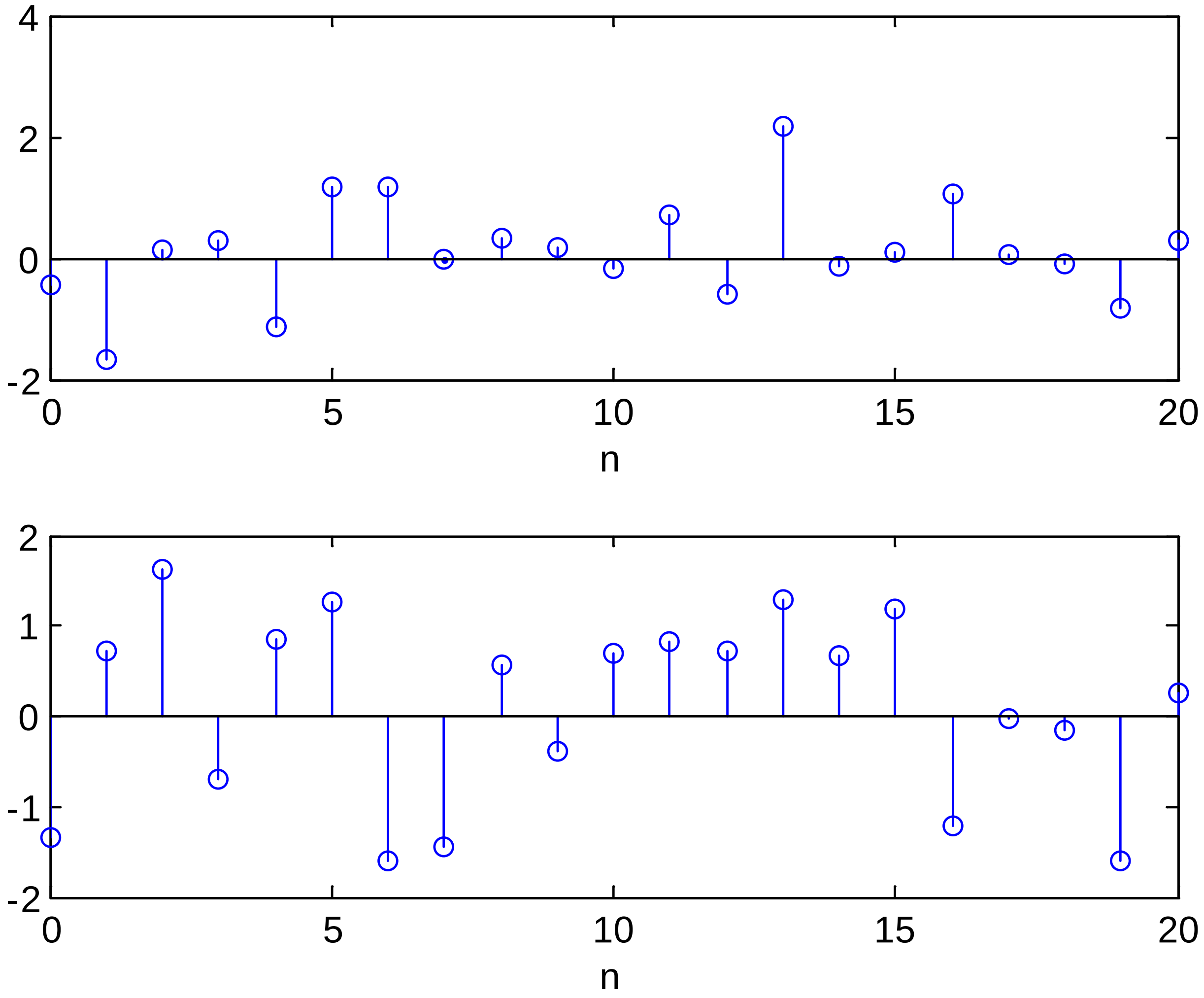


Fig.3.6: Gaussian number realizations using randn

Example 3.3

Use the MATLAB command randn to generate a zero-mean Gaussian sequence of length 10000 and unity power. Verify its characteristics.

We use `w=randn(1,10000)` to generate the sequence and `hist(w,50)` to show its distribution

The distribution aligns with Gaussian variables which is indicated by the bell shape

The empirical mean and power of `w` computed using `mean(w)` and `mean(w.*w)` are 4.1992×10^{-4} and 1.0028

The theoretical standard deviation is 1 and we see that most of the values are within –3 and 3

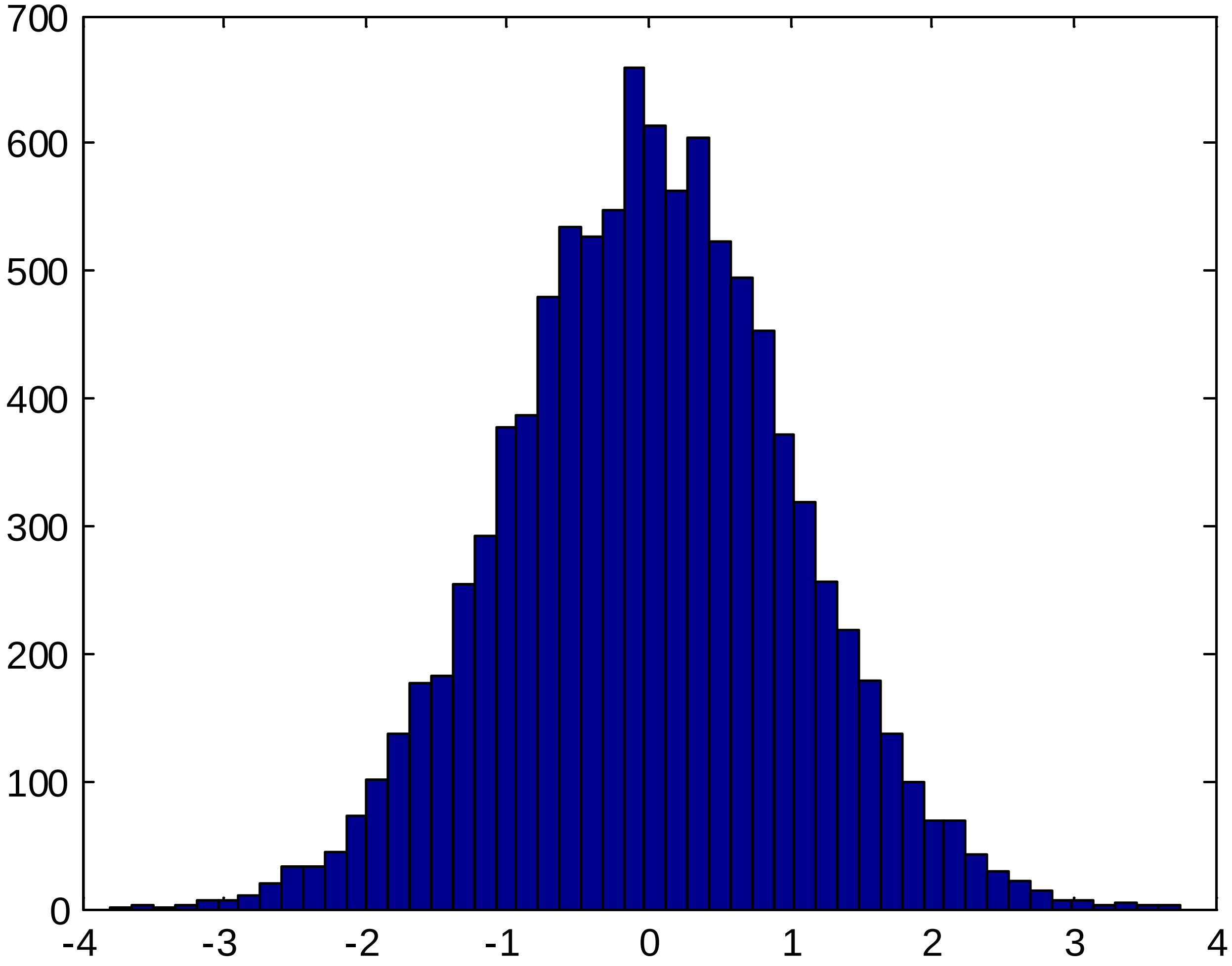


Fig.3.7: Histogram for Gaussian sequence

Discrete-Time Systems

A discrete-time system is an operator $\mathcal{T}$ which maps an input sequence $x[n]$ into an output sequence $y[n]$:

$$y[n] = \mathcal{T}\{x[n]\} \tag{3.12}$$

- **Memoryless**: $y[n]$ at time n depends only on $x[n]$ at time n

Are they memoryless systems?

$$y[n]=(x[n])^2$$
$$y[n]=x[n]+ x[n-2]$$

- **Linear**: obey principle of superposition, i.e., if

$$y_1[n] = \mathcal{T}\{x_1[n]\} \qquad \text{and} \qquad y_2[n] = \mathcal{T}\{x_2[n]\}$$

then

$$\mathcal{T}\{ax_1[n] + bx_2[n]\} = a\mathcal{T}\{x_1[n]\} + b\mathcal{T}\{x_2[n]\} = ay_1[n] + by_2[n] \tag{3.13}$$

Example 3.4

Determine whether the following system with input $x[n]$ and output $y[n]$, is linear or not:

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

A standard approach to determine the linearity of a system is given as follows. Let

$$y_i[n] = \mathcal{T}\{x_i[n]\}, \quad i = 1, 2, 3$$

with

$$x_3[n] = ax_1[n] + bx_2[n]$$

If $y_3[n] = ay_1[n] + by_2[n]$, then the system is linear. Otherwise, the system is nonlinear.

Assigning $x_3[n] = ax_1[n] + bx_2[n]$, we have:

$$\begin{aligned} y_3[n] &= \sum_{k=-\infty}^{n} x_3[k] \\ &= \sum_{k=-\infty}^{n} (ax_1[k] + bx_2[k]) \\ &= a \sum_{k=-\infty}^{n} x_1[k] + b \sum_{k=-\infty}^{n} x_2[k] \\ &= ay_1[n] + by_2[n] \end{aligned}$$

Note that the outputs for $x_1[n]$ and $x_2[n]$ are $y_1[n] = \sum_{k=-\infty}^{n} x_1[k]$ and $y_2[n] = \sum_{k=-\infty}^{n} x_2[k]$

As a result, the system is linear

Example 3.5

Determine whether the following system with input $x[n]$ and output $y[n]$, is linear or not:

$$y[n] = 3x^2[n] + 2x[n-3]$$

The system outputs for $x_1[n]$ and $x_2[n]$ are $y_1[n] = 3x_1^2[n] + 2x_1[n-3]$ and $y_2[n] = 3x_2^2[n] + 2x_2[n-3]$. Assigning $x_3[n] = ax_1[n] + bx_2[n]$, its system output is then:

$$\begin{aligned} y_3[n] &= 3x_3^2[n] + 2x_3[n-3] \\ &= 3\left(ax_1[n] + bx_2[n]\right)^2 + 2ax_1[n-3] + 2bx_2[n-3] \\ &= 3\left(a^2x_1^2[n] + b^2x_2^2[n] + 2abx_1[n]x_2[n]\right) + 2ax_1[n-3] + 2bx_2[n-3] \\ &\neq a\left(3x_1^2[n] + 2x_1[n-3]\right) + b\left(3x_2^2[n] + 2x_2[n-3]\right) \\ &= ay_1[n] + by_2[n] \end{aligned}$$

As a result, the system is nonlinear

- Time-Invariant: a time-shift of input causes a corresponding shift in output, i.e., if

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

then

$$y[n-n_0] = \mathcal{T}\{x[n-n_0]\} \tag{3.14}$$

Example 3.6

Determine whether the following system with input $x[n]$ and output $y[n]$, is time-invariant or not:

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

A standard approach to determine the time-invariance of a system is given as follows.

Let

$$y_1[n] = \mathcal{T}\{x_1[n]\}$$

with $x_1[n] = x[n-n_0]$

If $y_1[n] = y[n-n_0]$, then the system is time-invariant. Otherwise, the system is time-variant.

From the given input-output relationship, $y[n-n_0]$ is:

$$y[n-n_0] = \sum_{k=-\infty}^{n-n_0} x[k]$$

Let $x_1[n] = x[n-n_0]$, its system output is:

$$\begin{aligned} y_1[n] &= \sum_{k=-\infty}^{n} x_1[k] = \sum_{k=-\infty}^{n} x[k-n_0] = \sum_{l=-\infty}^{n-n_0} x[l], \quad l = k - n_0 \\ &= y[n-n_0] \end{aligned}$$

As a result, the system is time-invariant.

Example 3.7

Determine whether the following system with input $x[n]$ and output $y[n]$, is time-invariant or not:

$$y[n] = 3x[3n]$$

From the given input-output relationship, $y[n-n_0]$ is of the form:

$$y[n-n_0] = 3x[3(n-n_0)] = 3x[3n-3n_0]$$

Let $x_1[n] = x[n-n_0]$, its system output is:

$$y_1[n] = 3x_1[3n] = 3x[3n-n_0] \neq y[n-n_0]$$

As a result, the system is time-variant.

- Causal: output $y[n]$ at time n depends on input $x[n]$ up to time n

  For linear time-invariant (LTI) systems, there is an alternative definition. A LTI system is causal if its impulse response $h[n]$ satisfies:

$$h[n] = 0, \quad n < 0 \qquad (3.15)$$

Are they causal systems?

$$\mathbf{y[\mathit{n}]=x[\mathit{n}]+x[n+1]}$$

$$\mathbf{y[\mathit{n}]=x[\mathit{n}]+x[\mathit{n}-2]}$$

- Stable: a bounded input $x[n]$ ($|x[n]| < \infty$) produces a bounded output $y[n]$ ($|y[n]| < \infty$)

For LTI system, stability also corresponds to

$$\sum_{n=-\infty}^{\infty} |h[n]| < \infty \tag{3.16}$$

Are they stable systems?

y[*n*]=x[*n*]+x[n+1]

y[*n*]=1/x[*n*]

Convolution

The input-output relationship for a LTI system is characterized by convolution:

$$y[n] = x[n] \otimes h[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m] \tag{3.17}$$

which is similar to (2.23)

(3.17) is simpler as it only needs additions and multiplications

$h[n]$ specifies the functionality of the system

- Commutative

$$\begin{aligned} x[n] \otimes h[n] &= h[n] \otimes x[n] \\ &= \sum_{m=-\infty}^{\infty} x[m]h[n-m] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] \end{aligned} \qquad (3.18)$$

and

$$\begin{aligned} y[n] &= x[n] \otimes h_1[n] \otimes h_2[n] \\ &= x[n] \otimes h_2[n] \otimes h_1[n] \\ &= x[n] \otimes (h_1[n] \otimes h_2[n]) \end{aligned} \qquad (3.19)$$

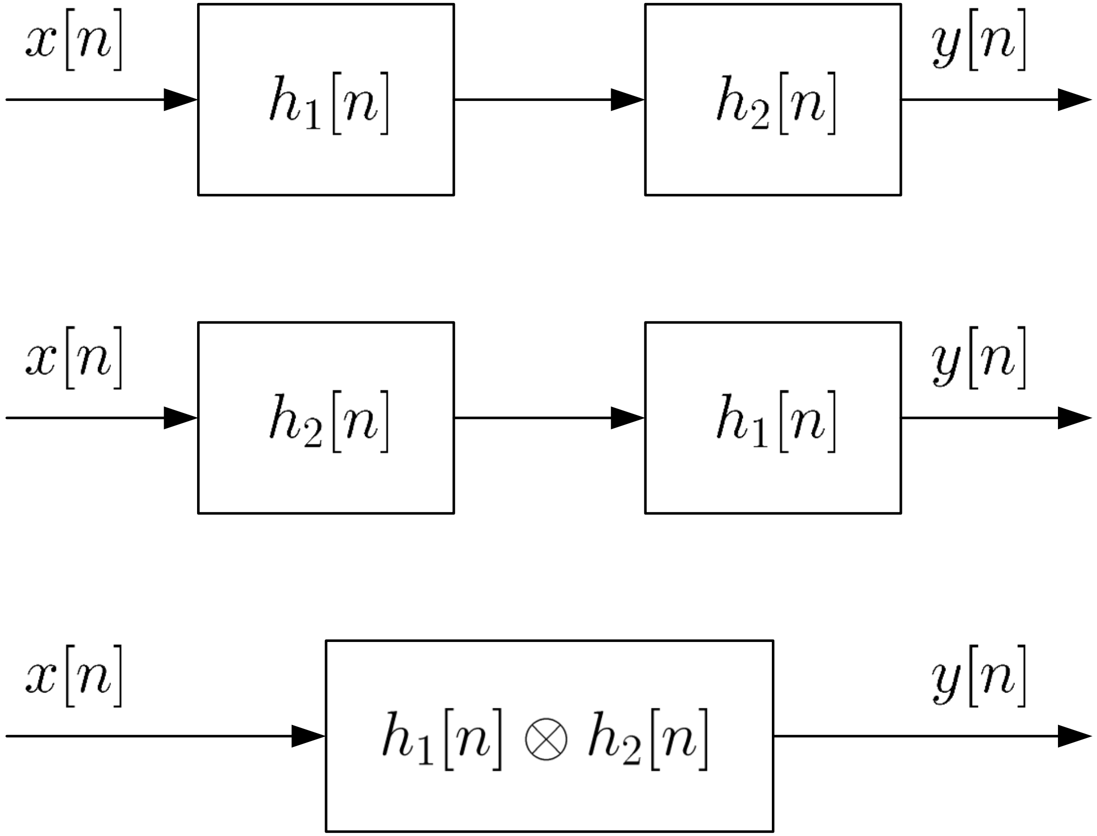


Fig.3.8: Commutative property of convolution

- Linearity

$$y[n] = x[n] \otimes (h_1[n] + h_2[n]) = x[n] \otimes h_1[n] + x[n] \otimes h_2[n] \qquad (3.20)$$

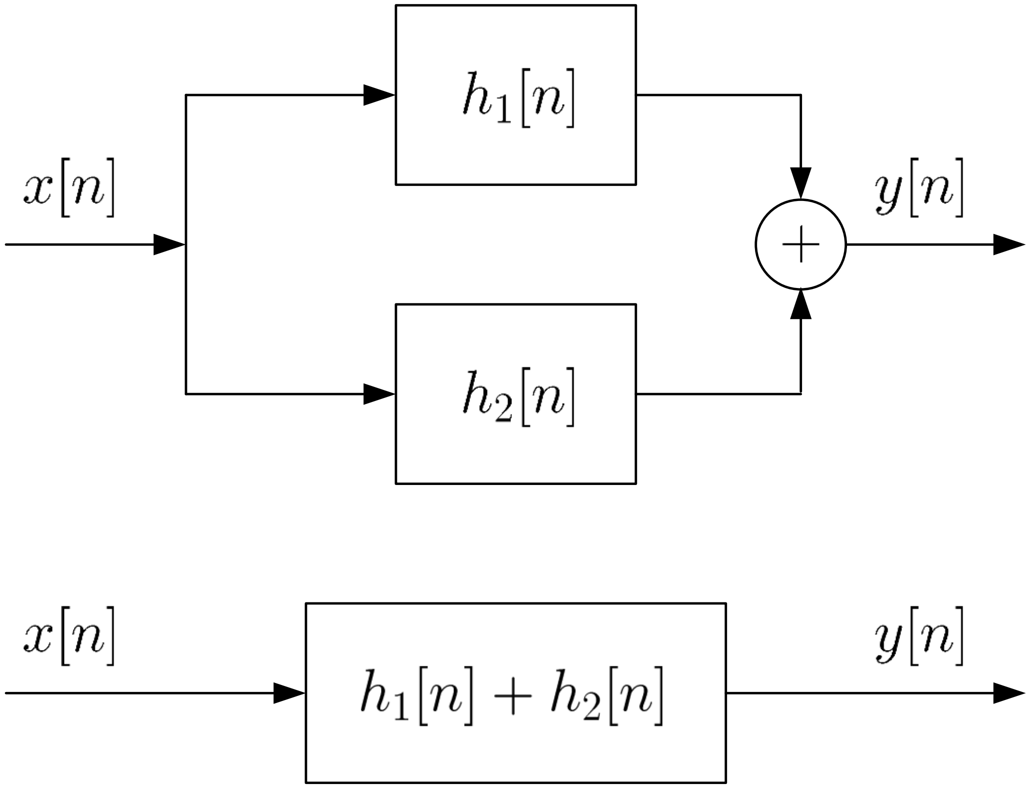


Fig.3.9: Linear property of convolution

Example 3.8

Compute the output $y[n]$ if the input is $x[n] = u[n]$ and the LTI system impulse response is $h[n] = \delta[n] + 0.5\delta[n-1]$. Determine the stability and causality of system.

Using (3.17), we have:

$$
\begin{aligned}
y[n] &= x[n] \otimes h[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m] \\
&= \sum_{m=-\infty}^{\infty} u[m]\left(\delta[n-m] + 0.5\delta[n-1-m]\right) \\
&= \sum_{m=0}^{\infty} \left(\delta[n-m] + 0.5\delta[n-1-m]\right) \\
&= \sum_{m=0}^{\infty} \delta[n-m] + 0.5\sum_{m=0}^{\infty} \delta[n-1-m] = u[n] + 0.5u[n-1]
\end{aligned}
$$

Alternatively, we can first establish the general relationship between $y[n]$ and $x[n]$ with the specific $h[n]$ and (3.4):

$$\begin{aligned} y[n] &= x[n] \otimes h[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m] \\ &= \sum_{m=-\infty}^{\infty} x[m]\left(\delta[n-m] + 0.5\delta[n-1-m]\right) \\ &= \sum_{m=-\infty}^{\infty} x[m]\delta[n-m] + 0.5 \sum_{m=-\infty}^{\infty} x[m]\delta[n-1-m] \\ &= x[n] + 0.5x[n-1] \end{aligned}$$

Substituting $x[n] = u[n]$ yields the same $y[n]$.

Since $\sum_{n=-\infty}^{\infty} |h[n]| = \sum_{n=0}^{1} |h[n]| = 1.5 < \infty$ and $h[n] = 0$ for $n < 0$ the system is stable and causal

Example 3.9

Compute the output $y[n]$ if the input is $x[n]=a^n u[n]$ and the LTI system impulse response is $h[n]=u[n]-u[n-10]$. Determine the stability and causality of system.

Using (3.17), we have:

$$\begin{aligned}
y[n] &= \sum_{m=-\infty}^{\infty} x[m]h[n-m] \\
&= \sum_{m=-\infty}^{\infty} a^m u[m]\left(u[n-m]-u[n-10-m]\right) \\
&= \sum_{m=0}^{\infty} a^m \left(u[n-m]-u[n-10-m]\right) \\
&= \sum_{m=0}^{\infty} a^m u[n-m] - \sum_{m=0}^{\infty} a^m u[n-10-m]
\end{aligned}$$

Let $y_1[n]=\sum_{m=0}^{\infty} a^m u[n-m]$ and $y_2[n]=\sum_{m=0}^{\infty} a^m u[n-10-m]$ such that $y[n]=y_1[n]-y_2[n]$. By employing a change of variable, $y_1[n]$ is expressed as

$$\begin{aligned} y_1[n] &= \sum_{m=0}^{\infty} a^m u[n-m] = \sum_{k=n}^{-\infty} a^{n-k} u[k], \quad k=n-m \\ &= \sum_{k=-\infty}^{n} a^{n-k} u[k] \end{aligned}$$

Since $u[k]=0$ for $k<0$, $y_1[n]=0$ for $n<0$. For $n \geq 0$, $y_1[n]$ is:

$$y_1[n]=\sum_{k=0}^{n} a^{n-k} u[k] = 1+a+\cdots+a^n = \frac{1-a^{n+1}}{1-a}$$

That is,

$$y_1[n]=\frac{1-a^{n+1}}{1-a} u[n]$$

Similarly, $y_2[n]$ is:

$$\begin{aligned} y_2[n] &= \sum_{m=0}^{\infty} a^m u[n-10-m] \\ &= \sum_{k=-\infty}^{n-10} a^{n-10-k} u[k], \quad k = n-10-m \end{aligned}$$

Since $u[k]=0$ for $k<0$, $y_2[n]=0$ for $n<10$. For $n \geq 10$, $y_2[n]$ is:

$$y_2[n] = \sum_{k=0}^{n-10} a^{n-10-k} = 1 + a + \cdots + a^{n-10} = \frac{1-a^{n-9}}{1-a}$$

That is,

$$y_2[n] = \frac{1-a^{n-9}}{1-a} u[n-10]$$

Combining the results, we have:

$$y[n] = \frac{1-a^{n+1}}{1-a}u[n] - \frac{1-a^{n-9}}{1-a}u[n-10]$$

or

$$y[n] = \begin{cases} 0, & n < 0 \\ \dfrac{1-a^{n+1}}{1-a}, & 0 \leq n < 10 \\ \dfrac{a^{n-9}\left(1-a^{10}\right)}{1-a}, & 10 \leq n \end{cases}$$

Since $\sum_{n=-\infty}^{\infty} |h[n]| = \sum_{n=0}^{9} |h[n]| = 10 < \infty$, the system is stable. Moreover, the system is causal because $h[n] = 0$ for $n < 0$.

Example 3.10

Determine $y[n] = x[n] \otimes h[n]$ where $x[n]$ and $h[n]$ are

$$x[n] = \begin{cases} n^2 + 1, & 0 \le n \le 3 \\ 0, & \text{otherwise} \end{cases}$$

and

$$h[n] = \begin{cases} n + 1, & 0 \le n \le 3 \\ 0, & \text{otherwise} \end{cases}$$

Here, the lengths of both $x[n]$ and $h[n]$ are finite. More precisely, $x[0] = 1$, $x[1] = 2$, $x[2] = 5$, $x[3] = 10$, $h[0] = 1$, $h[1] = 2$, $h[2] = 3$ and $h[3] = 4$ while all other $x[n]$ and $h[n]$ have zero values.

We still use (3.17) but now it reduces to a finite summation:

$$\begin{aligned} y[n] &= \sum_{m=-\infty}^{\infty} x[m]h[n-m] \\ &= x[0]h[n] + x[1]h[n-1] + x[2]h[n-2] + x[3]h[n-3] \end{aligned}$$

By considering the non-zero values of $h[n]$, we obtain:

$$y[n] = \begin{cases} 1, & n=0 \\ 4, & n=1 \\ 12, & n=2 \\ 30, & n=3 \\ 43, & n=4 \\ 50, & n=5 \\ 40, & n=6 \\ 0, & \text{otherwise} \end{cases}$$

Alternatively, for finite-length discrete-time signals, we can use the MATLAB command conv to compute the convolution of finite-length sequences:

```
n=0:3;
x=n.^2+1;
h=n+1;
y=conv(x,h)
```

The results are

```
y = 1     4     12     30     43     50     40
```

As the default starting time indices in both h and x are 1, we need to determine the appropriate time index for y

The correct index can be obtained by computing one value of $y[n]$ using (3.17). For simplicity, we may compute $y[0]$:

$$
\begin{aligned}
y[0] &= \sum_{m=-\infty}^{\infty} x[m]h[-m] \\
&= \cdots + x[-1]h[1] + x[0]h[0] + x[1]h[-1] + \cdots \\
&= x[0]h[0] \\
&= 1
\end{aligned}
$$

In general, if the lengths of $x[n]$ and $h[n]$ are M and N, respectively, the length of $y[n] = x[n] \otimes h[n]$ is $(M + N - 1)$.

Linear Constant Coefficient Difference Equations

For a LTI system, its input $x[n]$ and output $y[n]$ are related via a Nth-order linear constant coefficient difference equation:

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k] \qquad (3.21)$$

which is useful to check whether a system is both linear and time-invariant or not

Example 3.11

Determine if the following input-output relationships correspond to LTI systems:

(a) $y[n] = 0.1y[n-1] + x[n] + x[n-1]$

(b) $y[n] = x[n+1] + x[n]$

(c) $y[n] = 1/x[n]$

We see that (a) corresponds to a LTI system with $N = M = 1$, $a_0 = 1$, $a_1 = -0.1$ and $b_0 = b_1 = 1$

For (b), we reorganize the equation as:

$$y[n] = x[n+1] + x[n] \Rightarrow y[n-1] = x[n] + x[n-1]$$

which agrees with (3.21) when $N = M = 1$, $a_0 = 0$ and $a_1 = b_0 = b_1 = 1$. Hence (b) also corresponds to a LTI system

For (c), it does not correspond to a LTI system because $x[n]$ and $y[n]$ are not linear in the equation

Note that if a system cannot be fitted into (3.21), there are three possibilities: linear and time-variant; nonlinear and time-invariant; or nonlinear and time-variant

Example 3.12

Compute the impulse response $h[n]$ for a LTI system which is characterized by the following difference equation:

$$y[n] = x[n] - x[n-1]$$

Expanding (3.17) as

$$\begin{aligned} y[n] &= \sum_{m=-\infty}^{\infty} h[m]x[n-m] \\ &= \cdots + h[-1]x[n+1] + h[0]x[n] + h[1]x[n-1] + \cdots \end{aligned}$$

we can easily deduce that only $h[0]$ and $h[1]$ are nonzero. That is, the impulse response is:

$$h[n] = \delta[n] - \delta[n-1]$$

The difference equation is also useful to generate the system output and input.

Assuming that $a_0 \neq 0$, $y[n]$ is computed as:

$$y[n] = \frac{1}{a_0}\left(-\sum_{k=1}^{N} a_k y[n-k] + \sum_{k=0}^{M} b_k x[n-k]\right) \qquad (3.22)$$

Assuming that $b_0 \neq 0$, $x[n]$ can be obtained from:

$$x[n] = \frac{1}{b_0}\left(\sum_{k=0}^{N} a_k y[n-k] - \sum_{k=1}^{M} b_k x[n-k]\right) \qquad (3.23)$$

Example 3.13

Given a LTI system with difference equation of $y[n]=0.5y[n-1]+x[n]+x[n-1]$, compute the system output $y[n]$ for $0 \leq n \leq 50$ with an input of $x[n]=u[n]$. It is assumed that $y[-1]=0$.

The MATLAB code is:

```
N=50;              %data length is N+1
y(1)=1;            %compute y[0], only x[n] is nonzero
for n=2:N+1
y(n)=0.5*y(n-1)+2; %compute y[1],y[2],…,y[50]
                   %x[n]=x[n-1]=1 for n>=1
end
n=[0:N];           %set time axis
stem(n,y);
```

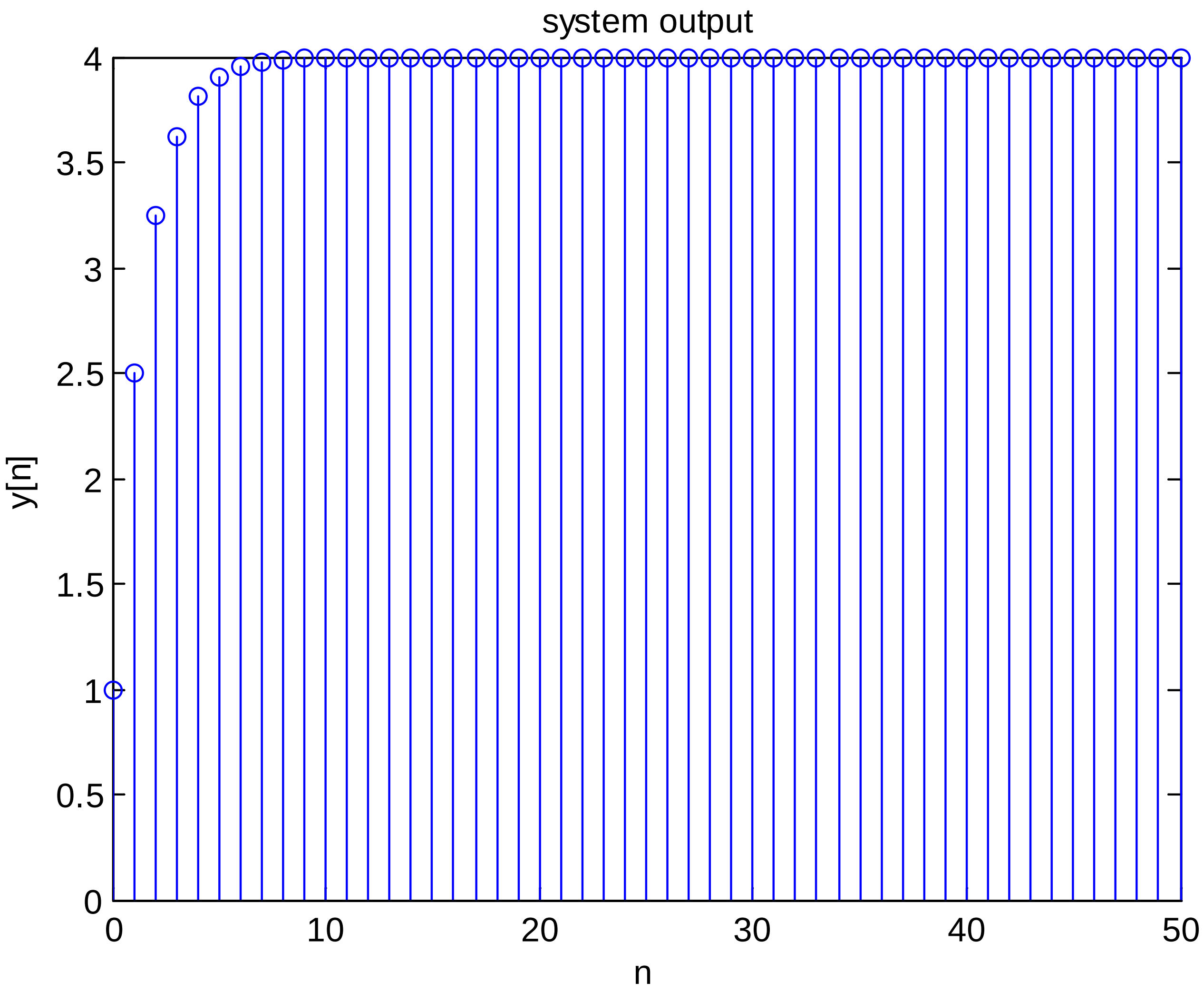


Fig.3.10: Output generation with difference equation

Alternatively, we can use the MATLAB command filter by rewriting the equation as:

$$y[n] - 0.5y[n-1] = x[n] + x[n-1]$$

The corresponding MATLAB code is:

```
x=ones(1,51);                  %define input
a=[1,-0.5];                    %define vector of a_k
b=[1,1];                       %define vector of b_k
y=filter(b,a,x);               %produce output
stem(0:length(y)-1,y)
```

The x is the input which has a value of 1 for $0 \le n \le 50$, while a and b are vectors which contain $\{a_k\}$ and $\{b_k\}$, respectively.

The MATLAB programs for this example are provided as ex3_13.m and ex3_13_2.m.

Chapter 4: Sampling and Reconstruction of Analog Signals

Chapter Intended Learning Outcomes:

(i) Ability to convert an analog signal to a discrete-time sequence via sampling

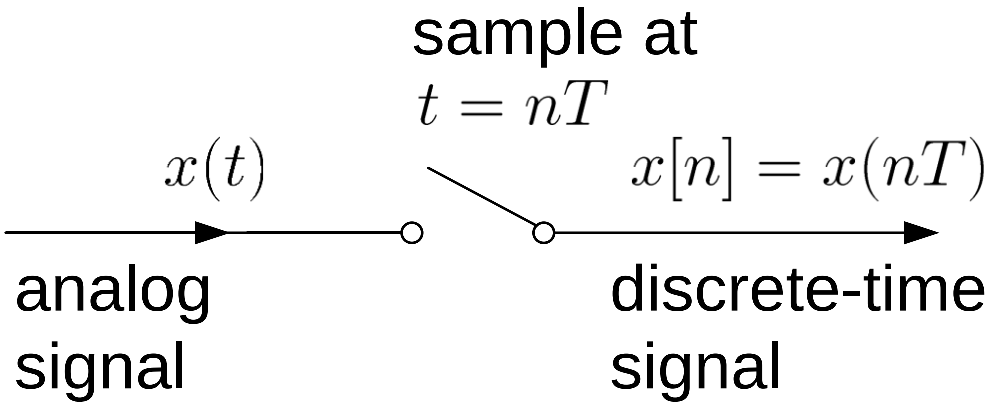
(ii) Ability to construct an analog signal from a discrete-time sequence

(iii) Understanding the conditions when a sampled signal can uniquely represent its analog counterpart

Sampling

- Process of converting a continuous-time signal $x(t)$ into a discrete-time sequence $x[n]$

- $x[n]$ is obtained by extracting $x(t)$ every T s where T is known as the sampling period or interval



sample at $t = nT$

$x(t)$

$x[n] = x(nT)$

analog signal

discrete-time signal

Fig.4.1: Conversion of analog signal to discrete-time sequence

- Relationship between $x(t)$ and $x[n]$ is:

$$x[n] = x(t)|_{t=nT} = x(nT), \quad n = \cdots -1, 0, 1, 2, \cdots \qquad (4.1)$$

- Conceptually, conversion of $x(t)$ to $x[n]$ is achieved by a continuous-time to discrete-time (CD) converter:

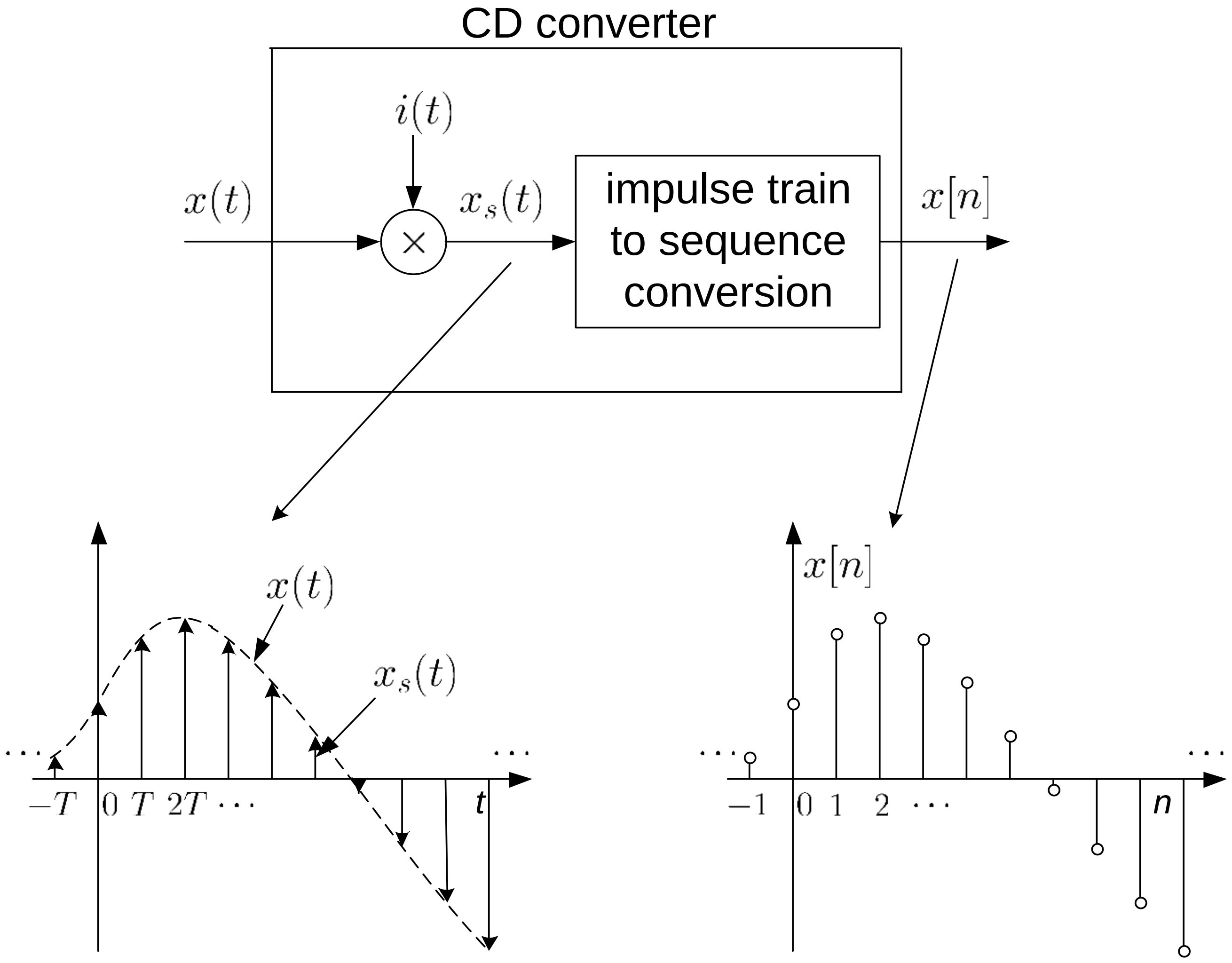

CD converter
$i(t)$
$x(t)$
$x_s(t)$
impulse train to sequence conversion
$x[n]$
$x(t)$
$x_s(t)$
$\cdots$
$-T$ 0 T $2T$ $\cdots$
t
$x[n]$
-1 0 1 2 $\cdots$
n

Fig.4.2: Block diagram of CD converter

- A fundamental question is whether $x[n]$ can uniquely represent $x(t)$ or if we can use $x[n]$ to reconstruct $x(t)$

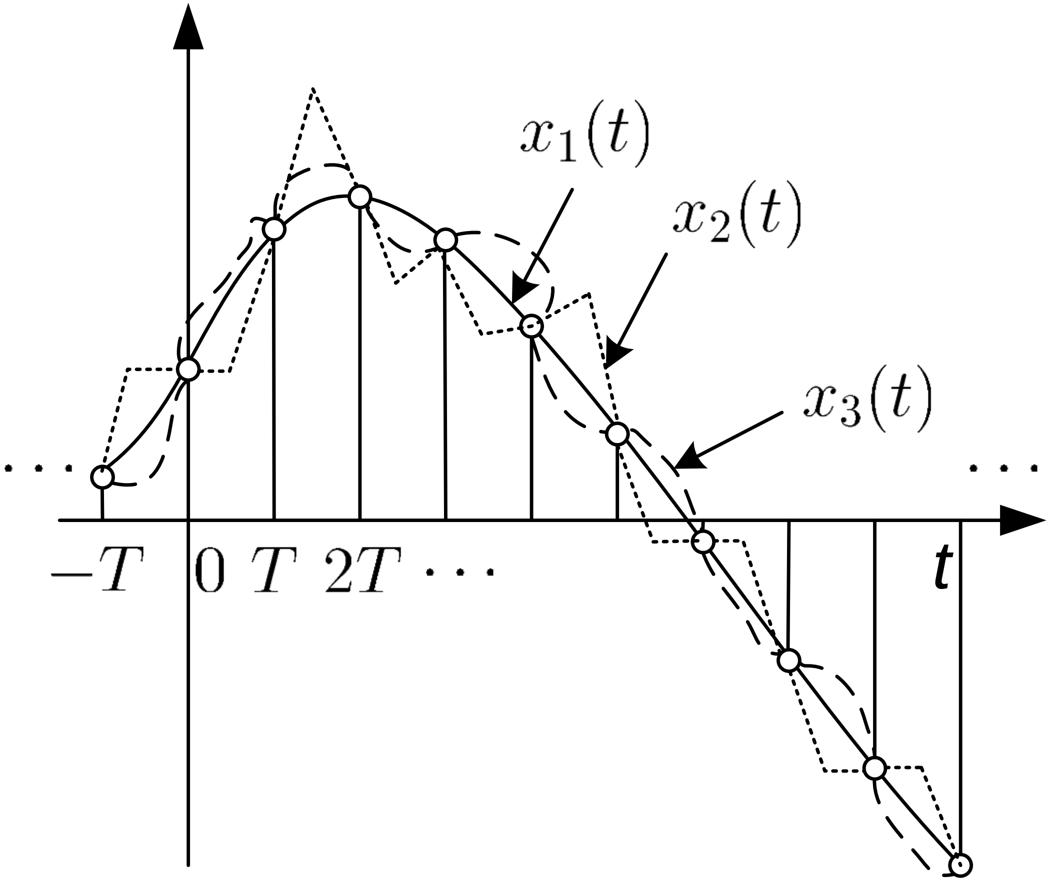


Fig.4.3: Different analog signals map to same sequence

But, the answer is yes when:

(1) $x(t)$ is bandlimited such that its Fourier transform $X(j\Omega)=0$ for $|\Omega|\geq\Omega_b$ where Ω_b is called the bandwidth

(2) Sampling period T is sufficiently small

<u>Example 4.1</u>

The continuous-time signal $x(t)=\cos(200\pi t)$ is used as the input for a CD converter with the sampling period $1/300$ s. Determine the resultant discrete-time signal $x[n]$.

According to (4.1), $x[n]$ is

$$x[n]=x(nT)=\cos(200n\pi T)=\cos\left(\frac{2\pi n}{3}\right), \quad n=\cdots-1,0,1,2,\cdots$$

The frequency in $x(t)$ is $200\pi \ \mathrm{rads}^{-1}$ while that of $x[n]$ is $2\pi/3$

Frequency Domain Representation of Sampled Signal

In the time domain, $x_s(t)$ is obtained by multiplying $x(t)$ by the impulse train $i(t)=\sum_{k=-\infty}^{\infty}\delta(t-kT)$:

$$x_s(t)=x(t)\sum_{k=-\infty}^{\infty}\delta(t-kT)=\sum_{k=-\infty}^{\infty}x[k]\delta(t-kT) \qquad (4.2)$$

with the use of the sifting property of (2.12)

Let the sampling frequency in radian be $\Omega_s=2\pi/T$ (or $F_s=1/T=\Omega_s/(2\pi)$ in Hz). From Example 2.8:

$$I(j\Omega)=\Omega_s\sum_{k=-\infty}^{\infty}\delta(\Omega-k\Omega_s) \qquad (4.3)$$

Using multiplication property of Fourier transform:

$$x_1(t) \cdot x_2(t) \leftrightarrow \frac{1}{2\pi} X_1(j\Omega) \otimes X_2(j\Omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X_1(j\tau) X_2(j(\Omega - \tau)) d\tau \quad (4.4)$$

where the convolution operation corresponds to continuous-time signals

Using (4.2)-(4.4) and properties of $\delta(t)$, $X_s(j\Omega)$ is:

$$
\begin{aligned}
X_s(j\Omega) &= \frac{1}{2\pi}\int\limits_{-\infty}^{\infty} I(j\tau)X(j(\Omega-\tau))d\tau \\
&= \frac{1}{2\pi}\int\limits_{-\infty}^{\infty}\left(\Omega_s\sum_{k=-\infty}^{\infty}\delta(\tau-k\Omega_s)\right)X(j(\Omega-\tau))d\tau \\
&= \frac{1}{T}\sum_{k=-\infty}^{\infty}\left(\int\limits_{-\infty}^{\infty}X(j(\Omega-\tau))\delta(\tau-k\Omega_s)d\tau\right) \\
&= \frac{1}{T}\sum_{k=-\infty}^{\infty}X(j(\Omega-k\Omega_s))\left(\int\limits_{-\infty}^{\infty}\delta(\tau-k\Omega_s)d\tau\right) \\
&= \frac{1}{T}\sum_{k=-\infty}^{\infty}X(j(\Omega-k\Omega_s)) \qquad (4.5)
\end{aligned}
$$

which is the sum of infinite copies of $X(j\Omega)$ scaled by $1/T$

When Ω_s is chosen sufficiently large such that all copies of $X(j\Omega)/T$ do not overlap, that is, $\Omega_s - \Omega_b > \Omega_b$ or $\Omega_s > 2\Omega_b$, we can get $X(j\Omega)$ from $X_s(j\Omega)$

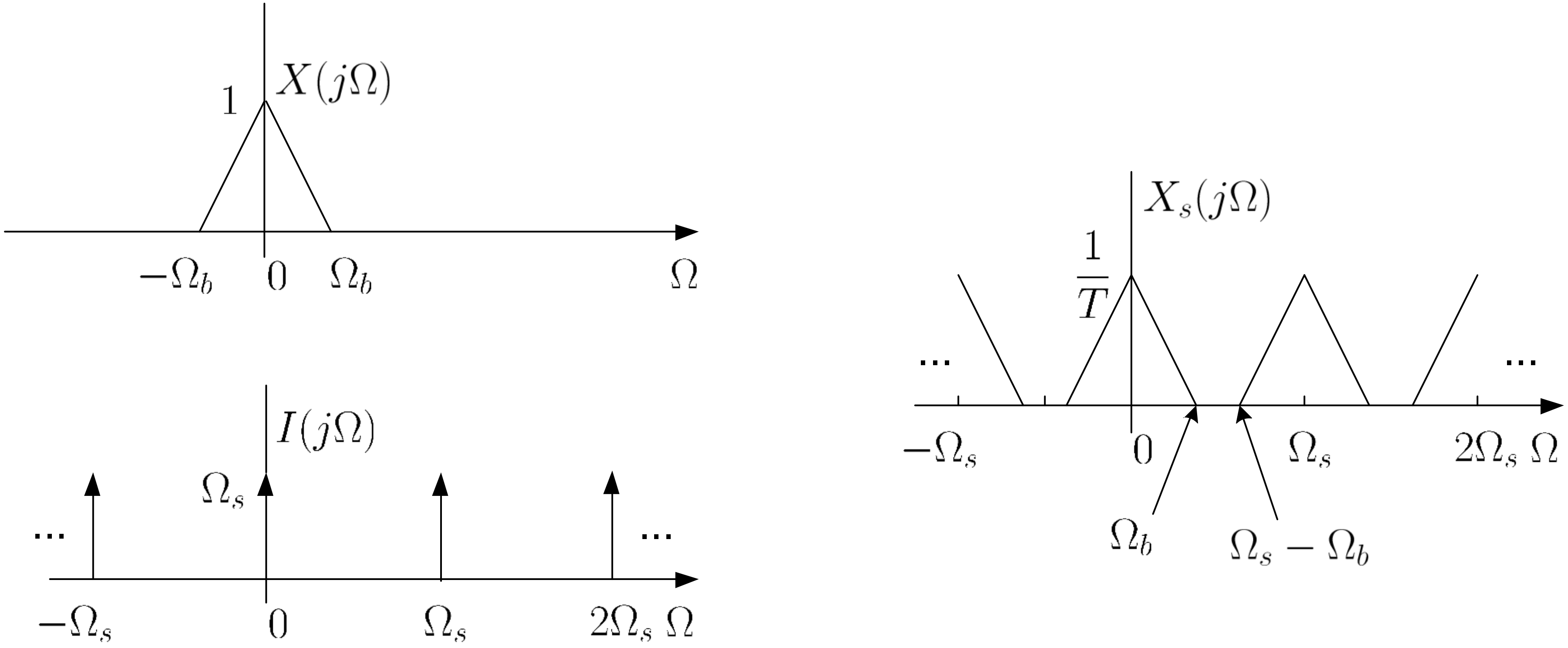


Fig.4.4: $X_s(j\Omega) = X(j\Omega) \otimes I(j\Omega)$ for sufficiently large Ω_s

When Ω_s is not chosen sufficiently large such that $\Omega_s < 2\Omega_b$, copies of $X(j\Omega)/T$ overlap, we cannot get $X(j\Omega)$ from $X_s(j\Omega)$, which is referred to aliasing

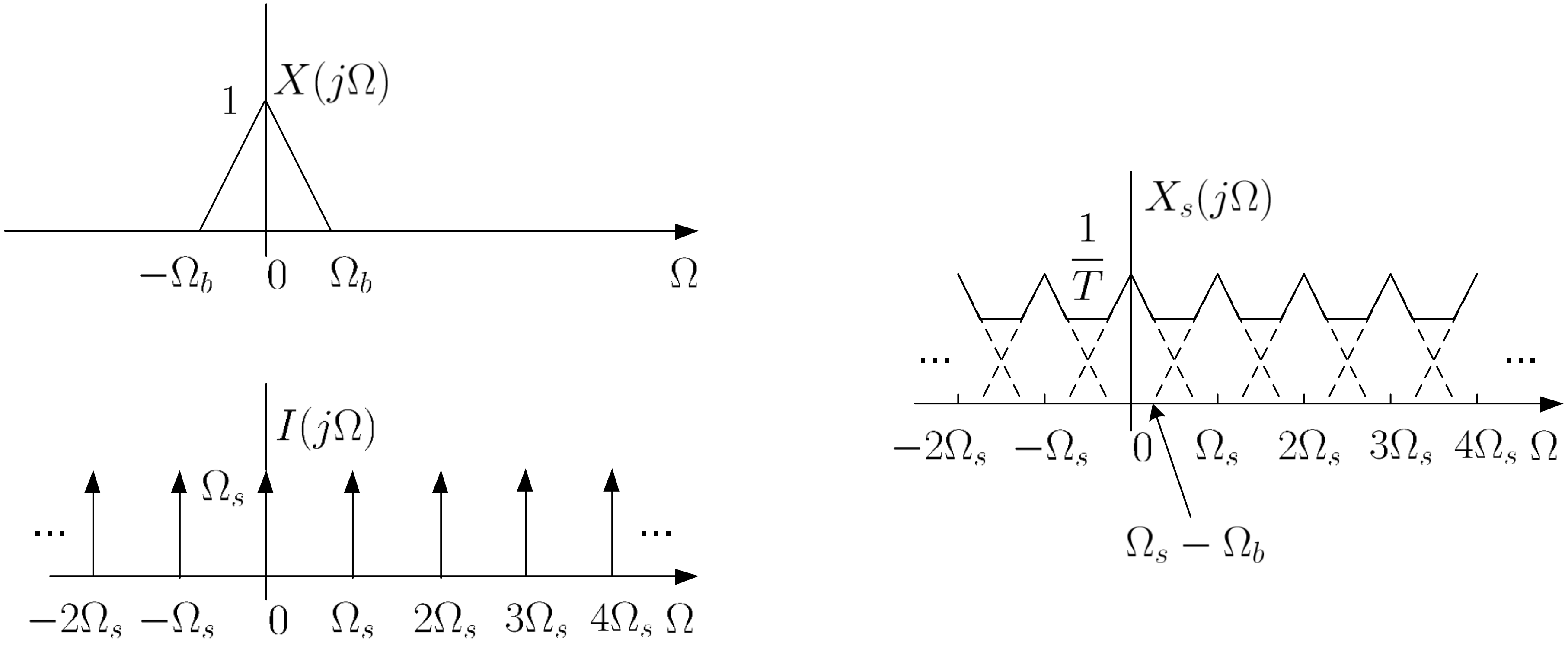


Fig.4.5: $X_s(j\Omega) = X(j\Omega) \otimes I(j\Omega)$ when Ω_s is not large enough

Nyquist Sampling Theorem (1928)

Let $x(t)$ be a bandlimited continuous-time signal with

$$X(j\Omega) = 0, \quad |\Omega| \geq \Omega_b \tag{4.6}$$

Then $x(t)$ is uniquely determined by its samples $x[n] = x(nT)$, $n = \cdots - 1, 0, 1, 2, \cdots$, if

$$\Omega_s = \frac{2\pi}{T} > 2\Omega_b \tag{4.7}$$

The bandwidth Ω_b is also known as the Nyquist frequency while $2\Omega_b$ is called the Nyquist rate and Ω_s must exceed it in order to avoid aliasing

Application	$f_b = \Omega_b/(2\pi)$	$f_s = \Omega_s/(2\pi)$
Biomedical	< 500 Hz	1 kHz
Telephone speech	< 4 kHz	8 kHz
Music	< 20 kHz	44.1 kHz
Ultrasonic	< 100 kHz	250 kHz
Radar	< 100 MHz	200 MHz

Table 4.1: Typical bandwidths and sampling frequencies in signal processing applications

Example 4.2

Determine the Nyquist frequency and Nyquist rate for the continuous-time signal $x(t)$ which has the form of:

$$x(t) = 1 + \sin(2000\pi t) + \cos(4000\pi t)$$

The frequencies are 0, 2000π and 4000π . The Nyquist frequency is $4000\pi \text{ rads}^{-1}$ and the Nyquist rate is $8000\pi \text{ rads}^{-1}$

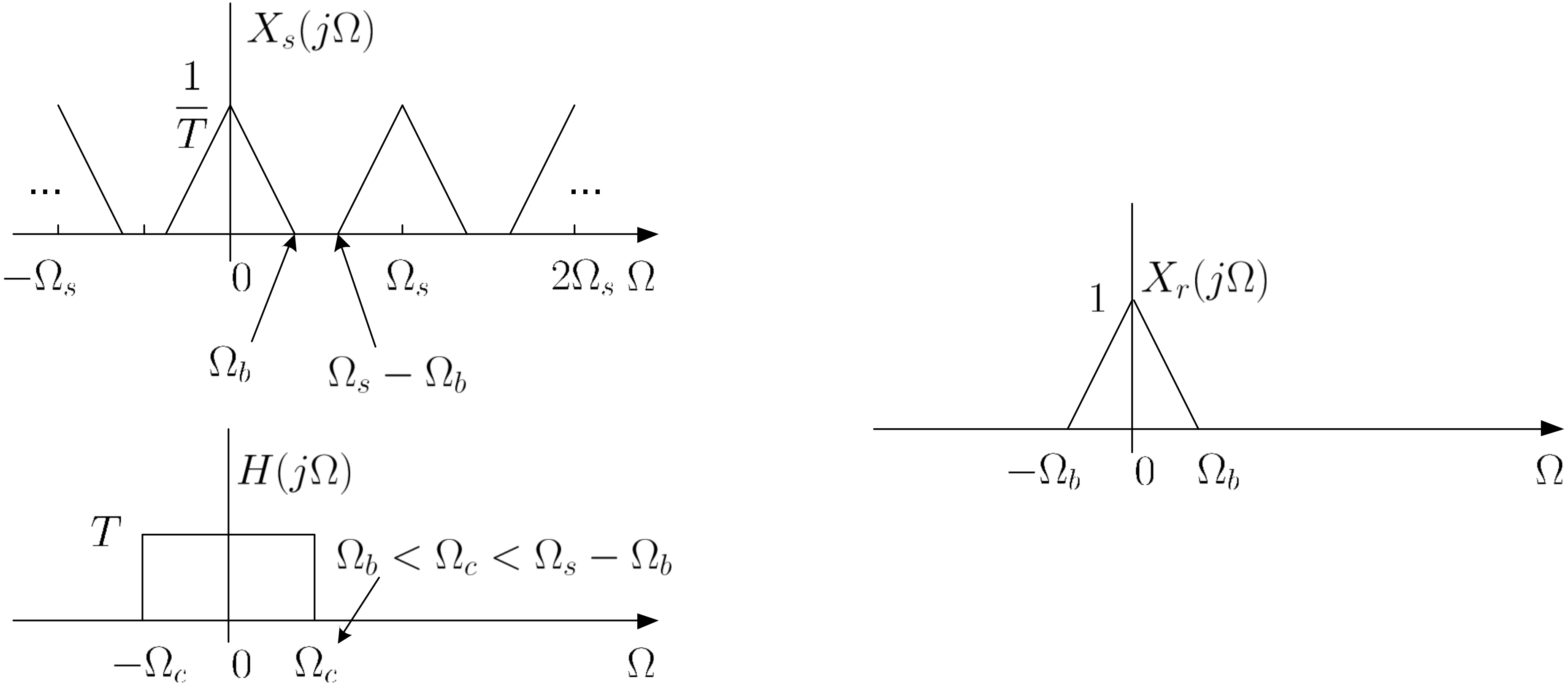


Fig.4.6: Multiplying $X_s(j\Omega)$ and $H(j\Omega)$ to recover $X(j\Omega)$

In frequency domain, we multiply $X_s(j\Omega)$ by $H(j\Omega)$ with amplitude T and bandwidth Ω_c with $\Omega_b < \Omega_c < \Omega_s - \Omega_b$, to obtain $X_r(j\Omega)$, and it corresponds to $x_r(t) = x_s(t) \otimes h(t)$

Reconstruction

- Process of transforming $x[n]$ back to $x(t)$

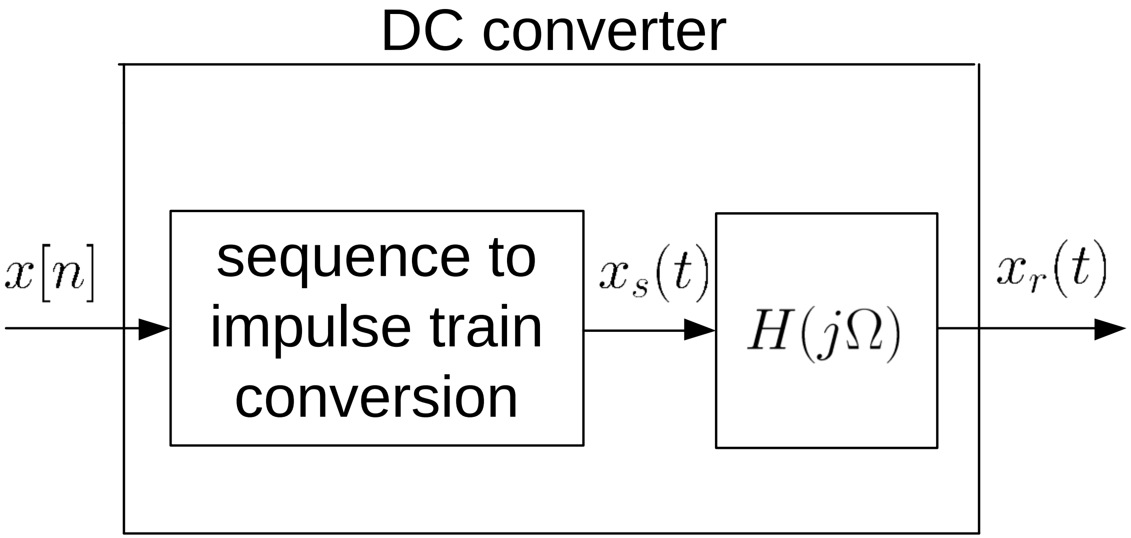


Fig.4.7: Block diagram of DC converter

From Fig.4.6, $H(j\Omega)$ is

$$H(j\Omega) = \begin{cases} T, & -\Omega_c < \Omega < \Omega_c \\ 0, & \text{otherwise} \end{cases} \qquad (4.8)$$

where $\Omega_b < \Omega_c < \Omega_s - \Omega_b$

For simplicity, we set Ω_c as the average of Ω_b and $(\Omega_s - \Omega_b)$:

$$\Omega_c = \frac{\Omega_s}{2} = \frac{\pi}{T} \tag{4.9}$$

To get $h(t)$, we take inverse Fourier transform of $H(j\Omega)$ and use Example 2.5:

$$\begin{aligned} h(t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} H(j\Omega)e^{j\Omega t}d\Omega = \frac{1}{2\pi}\int_{-\pi/T}^{\pi/T} Te^{j\Omega t}d\Omega = \frac{T\sin(\pi t/T)}{\pi t} \\ &= \text{sinc}\left(\frac{t}{T}\right) \end{aligned} \tag{4.10}$$

where $\text{sinc}(u) = \sin(\pi u)/(\pi u)$

Using (2.23)-(2.24), (4.2) and (2.11)-(2.12), $x_r(t)$ is:

$$\begin{aligned} x_r(t) &= x_s(t) \otimes h(t) \\ &= \left(\sum_{k=-\infty}^{\infty} x[k]\delta(t-kT) \right) \otimes h(t) \\ &= \int_{-\infty}^{\infty} \sum_{k=-\infty}^{\infty} x[k]\delta(\tau - kT)h(t-\tau)d\tau \\ &= \sum_{k=-\infty}^{\infty} x[k]h(t-kT) \\ &= \sum_{k=-\infty}^{\infty} x[k]\mathrm{sinc}\left(\frac{t-kT}{T}\right) \qquad (4.11) \end{aligned}$$

which holds for all real values of t

The interpolation formula can be verified at $t = nT$:

$$x_r(nT) = \sum_{k=-\infty}^{\infty} x[k]\mathrm{sinc}\,(n-k) \tag{4.12}$$

It is easy to see that

$$\mathrm{sinc}\,(n-k) = \frac{\sin((n-k)\pi)}{(n-k)\pi} = 0, \quad n \neq k \tag{4.13}$$

For $n = k$, we use $\mathrm{L'H\hat{o}pital}$'s rule to obtain:

$$\mathrm{sinc}(0) = \lim_{m\to 0} \frac{\sin\,(m\pi)}{m\pi} = \lim_{m\to 0} \frac{\frac{d\sin\,(m\pi)}{dm}}{\frac{dm\pi}{dm}} = \lim_{m\to 0} \frac{\pi\cos(m\pi)}{\pi} = 1 \tag{4.14}$$

Substituting (4.13)-(4.14) into (4.12) yields:

$$x_r(nT) = x[n] = x(nT) \tag{4.15}$$

which aligns with $x_r(t) = x(t)$

Example 4.3

Given a discrete-time sequence $x[n] = x(nT)$. Generate its time-delayed version $y[n]$ which has the form of

$$y[n] = x(nT - \Delta)$$

where $\Delta \neq mT > 0$ and m is a positive integer. Applying (4.11) with $t = nT - \Delta$:

$$y[n] = x(nT - \Delta) = \sum_{k=-\infty}^{\infty} x[k]\mathrm{sinc}\left(\frac{nT - kT - \Delta}{T}\right)$$

By employing a change of variable of $l = n - k$:

$$y[n] = \sum_{l=-\infty}^{\infty} x[n-l]\mathrm{sinc}\left(\frac{lT - \Delta}{T}\right)$$

Is it practical to get *y*[n]?

Note that when $\Delta = mT$, the time-shifted signal is simply obtained by shifting the sequence $x[n]$ by m samples:

$$y[n] = x(nT - mT) = x[n-m]$$

Sampling and Reconstruction in Digital Signal Processing

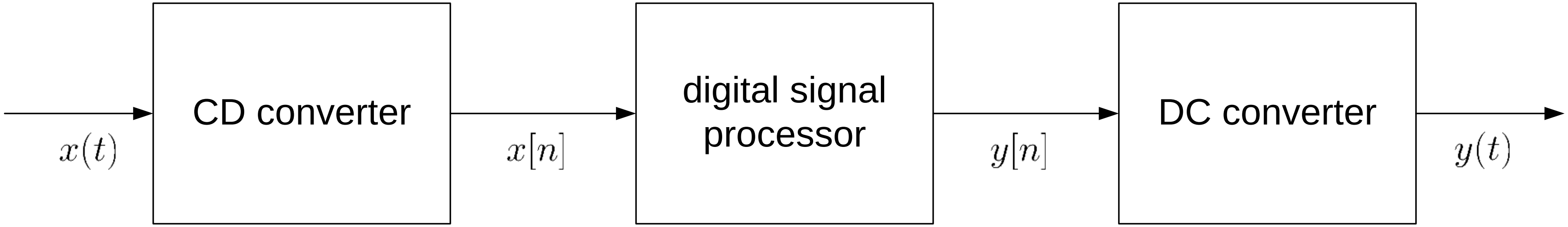


Fig.4.8: Ideal digital processing of analog signal

- CD converter produces a sequence $x[n]$ from $x(t)$
- $x[n]$ is processed in discrete-time domain to give $y[n]$
- DC converter creates $y(t)$ from $y[n]$ according to (4.11):

$$y(t) = \sum_{k=-\infty}^{\infty} y[k] \mathrm{sinc}\left(\frac{t-kT}{T}\right) \qquad (4.16)$$

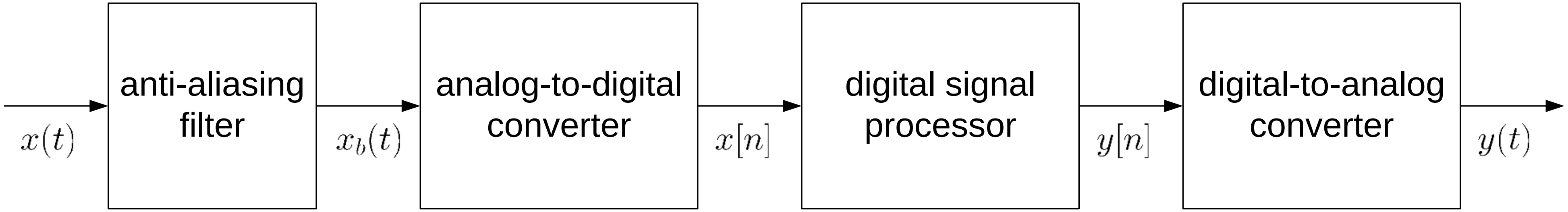


Fig.4.9: Practical digital processing of analog signal

- $x(t)$ may not be precisely bandlimited $\Rightarrow$ a lowpass filter or anti-aliasing filter is needed to process $x(t)$
- Ideal CD converter is approximated by AD converter
  - Not practical to generate $\delta(t)$
  - AD converter introduces quantization error
- Ideal DC converter is approximated by DA converter because ideal reconstruction of (4.16) is impossible
  - Not practical to perform infinite summation
  - Not practical to have future data
- $x[n]$ and $y[n]$ are quantized signals

Example 4.4

Suppose a continuous-time signal $x(t) = \cos(\Omega_0 t)$ is sampled at a sampling frequency of 1000Hz to produce $x[n]$:

$$x[n] = \cos\left(\frac{\pi n}{4}\right)$$

Determine 2 possible positive values of Ω_0, say, Ω_1 and Ω_2. Discuss if $\cos(\Omega_1 t)$ or $\cos(\Omega_2 t)$ will be obtained when passing $x[n]$ through the DC converter.

According to (4.1) with $T = 1/1000$ s:

$$\cos\left(\frac{\pi n}{4}\right) = x[n] = x(nT) = \cos\left(\frac{\Omega_0 n}{1000}\right)$$

Ω_1 is easily computed as:

$$\frac{\pi n}{4} = \frac{\Omega_1 n}{1000} \Rightarrow \Omega_1 = \frac{1000\pi}{4} = 250\pi$$

Ω_2 can be obtained by noting the periodicity of a sinusoid:

$$\cos\left(\frac{\pi n}{4}\right) = \cos\left(\frac{\pi n}{4} + 2n\pi\right) = \cos\left(\frac{9\pi n}{4}\right) = \cos\left(\frac{\Omega_2 n}{1000}\right)$$

As a result, we have:

$$\frac{9\pi n}{4} = \frac{\Omega_2 n}{1000} \Rightarrow \Omega_2 = \frac{9000\pi}{4} = 2250\pi$$

This is illustrated using the MATLAB code:

```
O1=250*pi;           %first frequency
O2=2250*pi;          %second frequency
Ts=1/100000;%successive sample separation is 0.01T
t=0:Ts:0.02;%observation interval
x1=cos(O1.*t);       %tone from first frequency
x2=cos(O2.*t);        %tone from second frequency
```

There are 2001 samples in 0.02s and interpolating the successive points based on plot yields good approximations

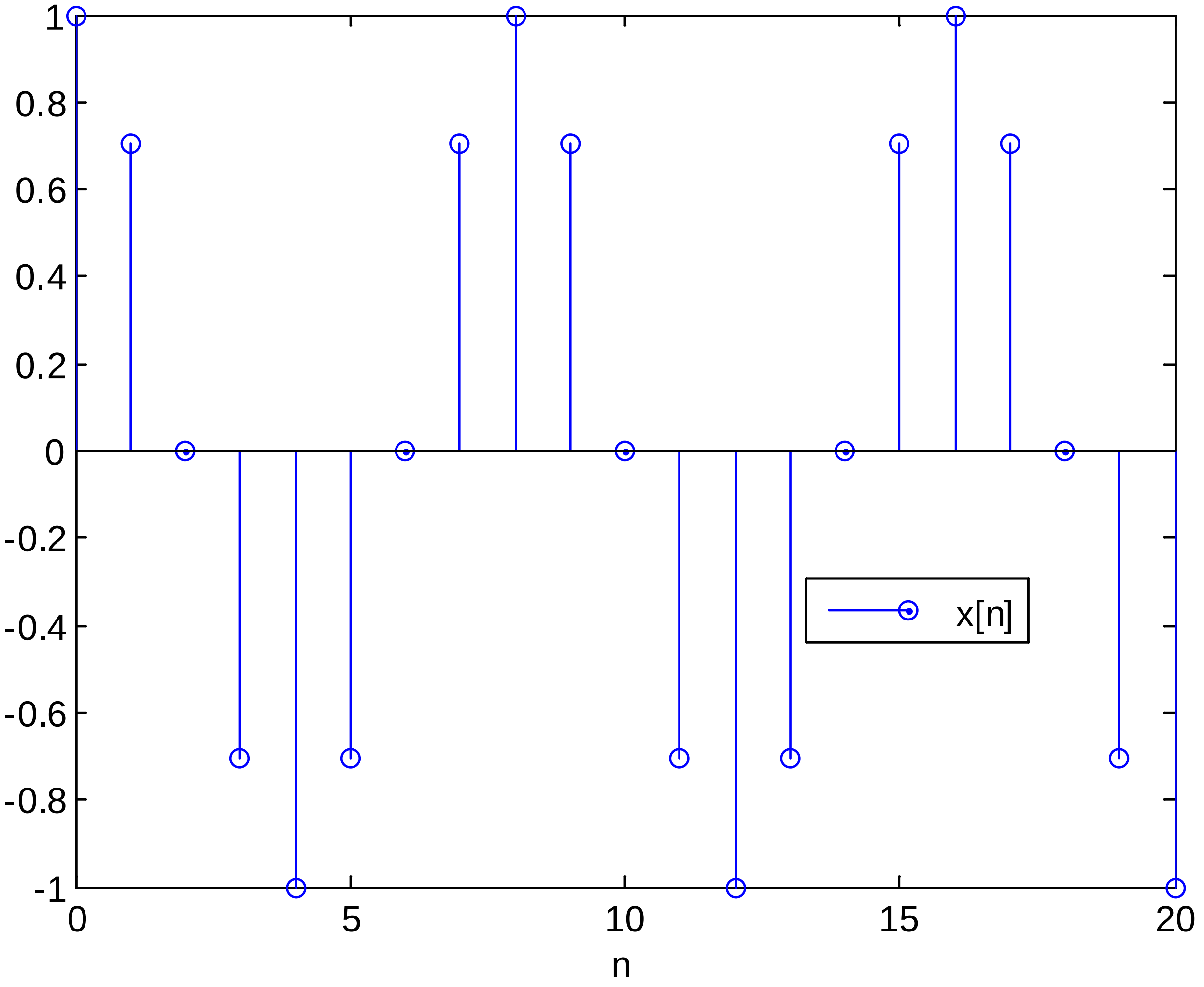


Fig.4.10: Discrete-time sinusoid

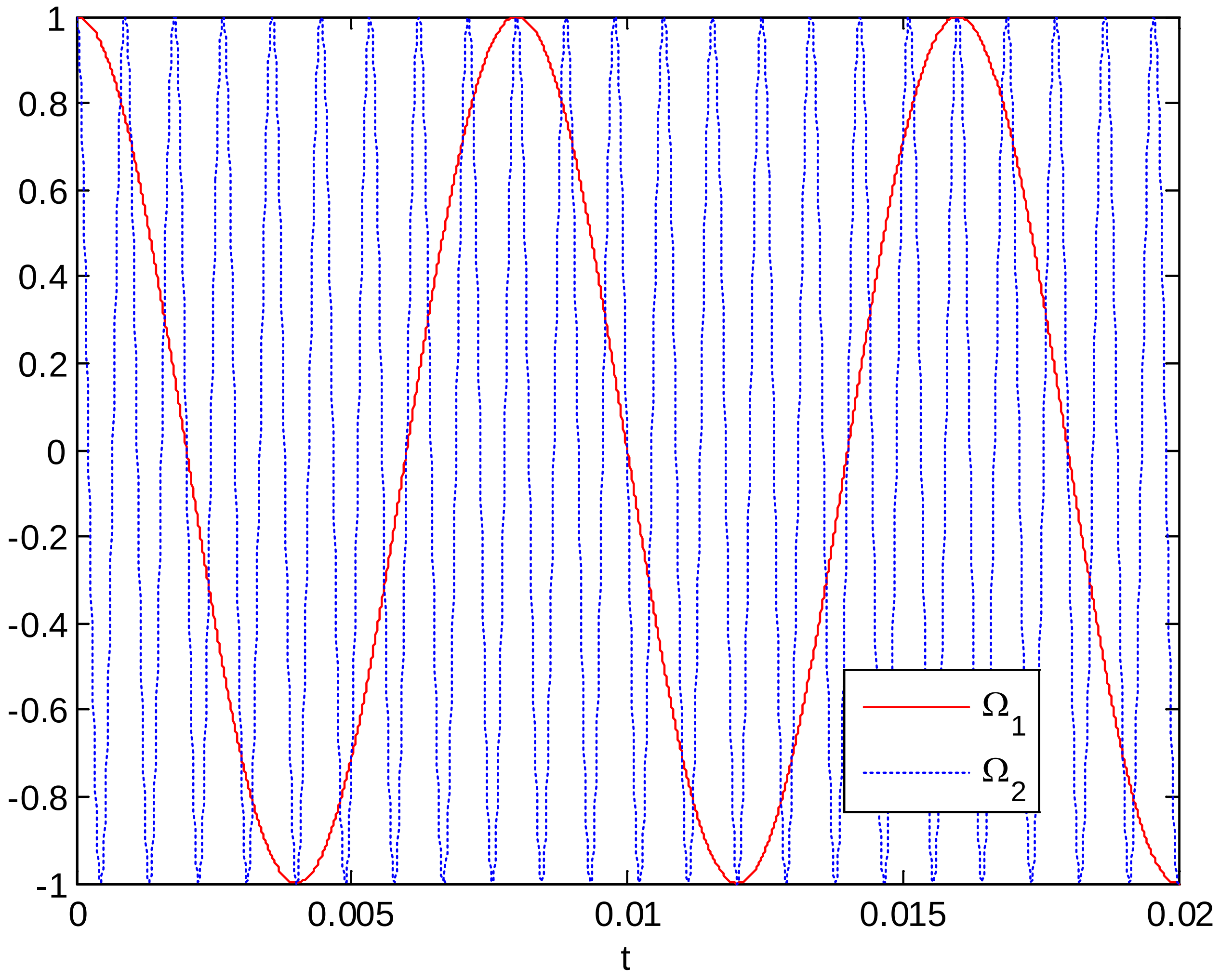


Fig.4.11: Continuous-time sinusoids

Passing $x[n]$ through the DC converter only produces $\cos(\Omega_1 t)$ but not $\cos(\Omega_2 t)$

The Nyquist frequency of $\cos(\Omega_2 t)$ is $2250\pi \text{ rads}^{-1}$ and hence the sampling frequency without aliasing is $\Omega_s > 4500\pi$

Given $F_s = 1000$ Hz or $\Omega_s = 2000\pi \quad \text{rads}^{-1}$, $\cos(\Omega_2 t)$ does not correspond to $x[n]$

We can recover $x_r(t) = \cos(\Omega_1 t)$ because the Nyquist frequency and Nyquist rate for $\cos(\Omega_1 t)$ are $250\pi \text{ rads}^{-1}$ and $500\pi \text{ rads}^{-1}$

Based on (4.11), $x_r(t) = \cos(\Omega_1 t)$ is:

$$x_r(t) = \sum_{k=-\infty}^{\infty} x[k]\text{sinc}\left(\frac{t-kT}{T}\right) \approx \sum_{k=-10}^{30} x[k]\text{sinc}\left(\frac{t-kT}{T}\right)$$

with $T = 1/1000$ s

The MATLAB code for reconstructing $\cos(\Omega_1 t)$ is:

```
n=-10:30;             %add 20 past and future samples
x=cos(pi.*n./4);
T=1/1000;             %sampling interval is 1/1000
for l=1:2000          %observed interval is [0,0.02]
t=(l-1)*T/100;%successive sample separation is 0.01T
h=sinc((t-n.*T)./T);
xr(l)=x*h.'; %approximate interpolation of (4.11)
end
```

We compute 2000 samples of $x_r(t)$ in $t \in [0, 0.02]$s

The value of each $x_r(t)$ at time `t` is approximated as `x*h.'` where the sinc vector is updated for each computation

The MATLAB program is provided as `ex4_4.m`

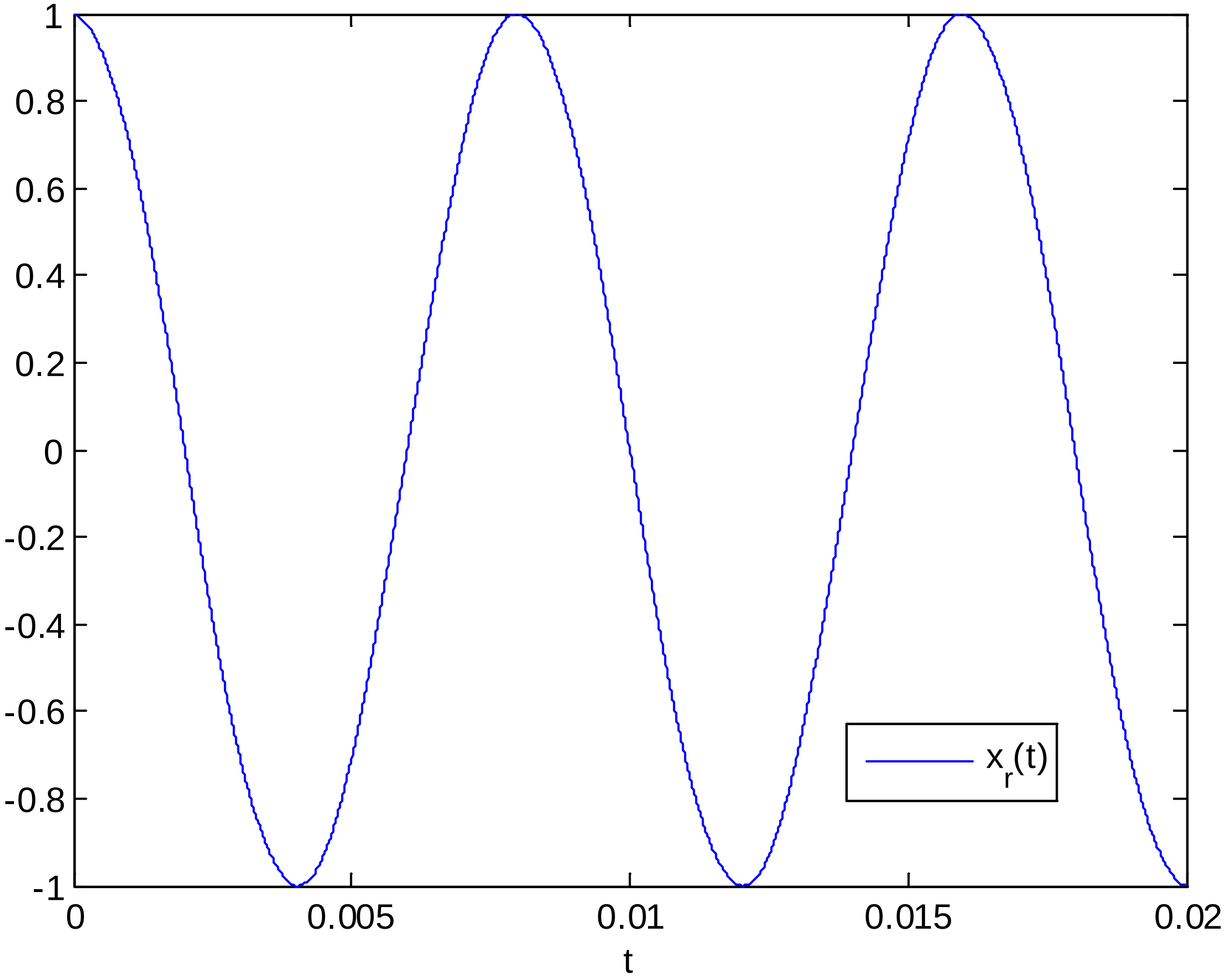


Fig.4.12: Reconstructed continuous-time sinusoid

Example 4.5

Play the sound for a discrete-time tone using MATLAB. The frequency of the corresponding analog signal is 440 Hz which corresponds to the A note in the American Standard pitch. The sampling frequency is 8000 Hz and the signal has a duration of 0.5 s.

The MATLAB code is

```
A=sin(2*pi*440*(0:1/8000:0.5));%discrete-time A
sound(A,8000);                 %DA conversion and play
```

Note that sampling frequency in Hz is assumed for sound. The frequencies of notes B, C#, D, E and F# are 493.88 Hz, 554.37 Hz, 587.33 Hz, 659.26 Hz and 739.99 Hz, respectively. You can easily produce a piece of music with notes: A, A, E, E, F#, F#, E, E, D, D, C#, C#, B, B, A, A.

Chapter 5: z Transform

Chapter Intended Learning Outcomes:

(i) Understanding the relationship between z transform and the Fourier transform for discrete-time signals

(ii) Understanding the characteristics and properties of z transform

(iii) Ability to compute z transform and inverse z transform

(iv) Ability to apply z transform for analyzing linear time-invariant (LTI) systems

Definition

The z transform of $x[n]$, denoted by $X(z)$, is defined as:

$$X(z) = \sum_{n=-\infty}^{\infty} x[n] z^{-n} \tag{5.1}$$

where z is a continuous complex variable.

Is *X*(*z*) real-valued or complex-valued?

Relationship with Fourier Transform

Employing (4.2), we construct the continuous-time sampled signal $x_s(t)$ with a sampling interval of T from $x[n]$:

$$x_s(t) = \sum_{n=-\infty}^{\infty} x[n] \delta(t - nT) \tag{5.2}$$

Taking Fourier transform of $x_s(t)$ with using properties of $\delta(t)$:

$$X_s(j\Omega) = \int_{-\infty}^{\infty} x_s(t)e^{-j\Omega t}dt = \int_{-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x[n]\delta(t-nT)e^{-j\Omega t}dt$$

$$= \sum_{n=-\infty}^{\infty} x[n]e^{-j\Omega nT} \qquad (5.3)$$

Defining $\omega = \Omega T$ as the discrete-time frequency parameter and writing $X_s(j\Omega)$ as $X(e^{j\omega})$, (5.3) becomes

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \qquad (5.4)$$

which is known as discrete-time Fourier transform (DTFT) or Fourier transform of discrete-time signals

$X(e^{j\omega})$ is periodic with period 2π:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = \sum_{n=-\infty}^{\infty} x[n]e^{-j(\omega+2k\pi)n} = X(e^{j(\omega+2k\pi)}) \quad (5.5)$$

where k is any integer. Since z is a continuous complex variable, we can write

$$z = re^{j\omega} \quad (5.6)$$

where $r = |z| > 0$ is magnitude and $\omega = \angle(z)$ is angle of z. Employing (5.6), the z transform is:

$$X(z)\big|_{z=re^{j\omega}} = X(re^{j\omega}) = \sum_{n=-\infty}^{\infty} \left(x[n]r^{-n}\right) e^{-j\omega n} \quad (5.7)$$

which is equal to the DTFT of $x[n]r^{-n}$. When $r = 1$ or $z = e^{j\omega}$, (5.7) and (5.4) are identical:

$$X(z)|_{z=e^{j\omega}} = X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \tag{5.8}$$

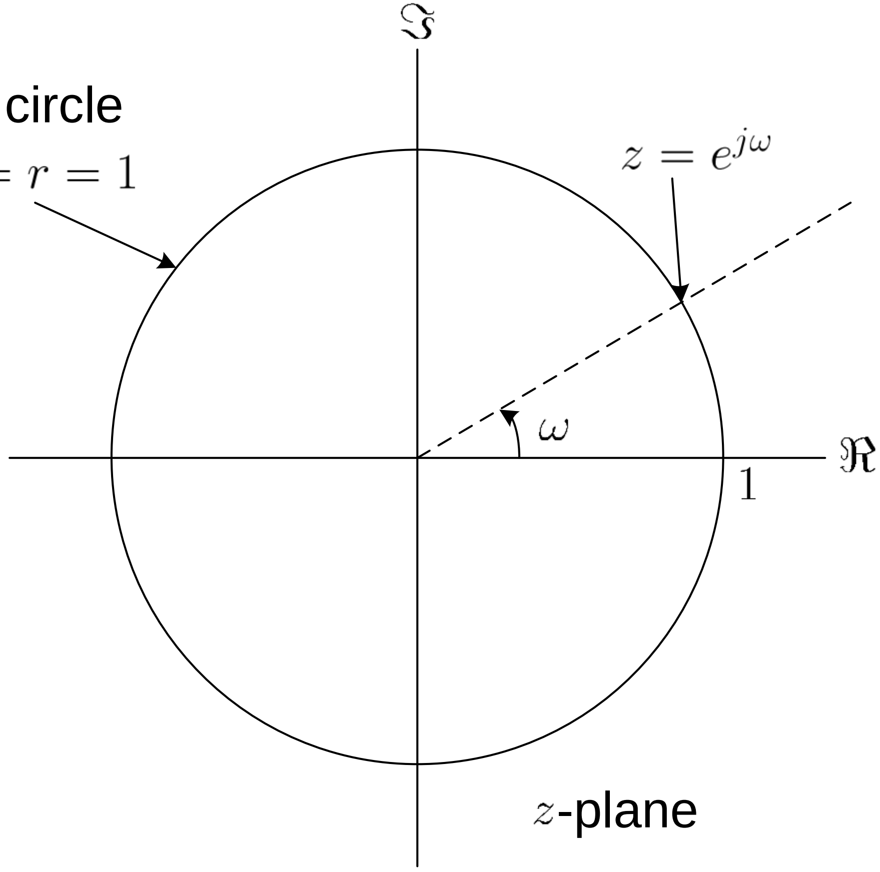


Fig.5.1: Relationship between $X(z)$ and $X(e^{j\omega})$ on the z-plane

Region of Convergence (ROC)

ROC indicates when z transform of a sequence converges

Generally there exists some z such that

$$|X(z)| = \left| \sum_{n=-\infty}^{\infty} x[n] z^{-n} \right| \rightarrow \infty \qquad (5.9)$$

where the z transform does not converge

The set of values of z for which $X(z)$ converges or

$$|X(z)| = \left| \sum_{n=-\infty}^{\infty} x[n] z^{-n} \right| \leq \sum_{n=-\infty}^{\infty} \left| x[n] z^{-n} \right| < \infty \qquad (5.10)$$

is called the ROC, which must be specified along with $X(z)$ in order for the z transform to be complete

Assuming that $x[n]$ is of infinite length, we decompose $X(z)$:

$$X(z) = X_-(z) + X_+(z) \tag{5.11}$$

where

$$X_-(z) = \sum_{n=-\infty}^{-1} x[n]z^{-n} = \sum_{m=1}^{\infty} x[-m]z^{m} \tag{5.12}$$

and

$$X_+(z) = \sum_{n=0}^{\infty} x[n]z^{-n} \tag{5.13}$$

Let $f_n(z) = x[n]z^{-n}$, $X_+(z)$ is expanded as:

$$\begin{aligned} X_+(z) &= x[0]z^{-0} + x[1]z^{-1} + \cdots + x[n]z^{-n} + \cdots \\ &= f_0(z) + f_1(z) + \cdots + f_n(z) + \cdots \end{aligned} \tag{5.14}$$

According to the ratio test, convergence of $X_+(z)$ requires

$$\lim_{n\to\infty}\left|\frac{f_{n+1}(z)}{f_n(z)}\right| < 1 \qquad (5.15)$$

Let $\lim_{n\to\infty} |x[n+1]/x[n]| = R_+ > 0$. $X_+(z)$ converges if

$$\begin{aligned}
&\lim_{n\to\infty}\left|\frac{x[n+1]z^{-n-1}}{x[n]z^{-n}}\right| = \lim_{n\to\infty}\left|\frac{x[n+1]}{x[n]}\right| |z^{-1}| < 1 \\
&\Rightarrow |z| > \lim_{n\to\infty}\left|\frac{x[n+1]}{x[n]}\right| = R_+
\end{aligned} \qquad (5.16)$$

That is, the ROC for $X_+(z)$ is $|z| > R_+$.

Let $\lim_{m\to\infty} |x[-m]/x[-m-1]| = R_- > 0$. $X_-(z)$ converges if

$$\lim_{m\to\infty}\left|\frac{x[-m-1]z^{m+1}}{x[-m]z^m}\right| = \lim_{m\to\infty}\left|\frac{x[-m-1]}{x[-m]}\right||z| < 1$$

$$\Rightarrow |z| < \lim_{m\to\infty}\left|\frac{x[-m]}{x[-m-1]}\right| = R_- \qquad (5.17)$$

As a result, the ROC for $X_-(z)$ is $|z| < R_-$

Combining the results, the ROC for $X(z)$ is $R_+ < |z| < R_-$:

- ROC is a ring when $R_+ < R_-$
- No ROC if $R_- < R_+$ and $X(z)$ does not exist

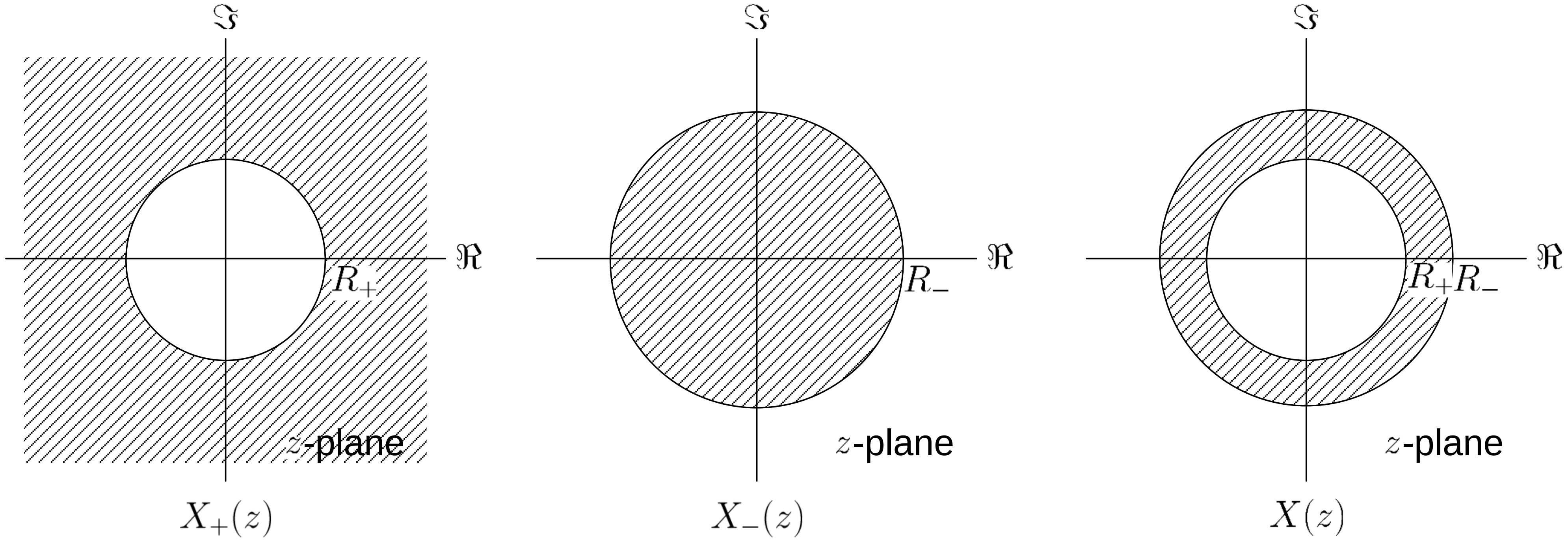


Fig.5.2: ROCs for $X_+(z)$, $X_-(z)$ and $X(z)$

Poles and Zeros

Values of z for which $X(z) = 0$ are the zeros of $X(z)$

Values of z for which $X(z) = \infty$ are the poles of $X(z)$

In many real-world applications, $X(z)$ is represented as a rational function:

$$X(z)=\frac{P(z)}{Q(z)}=\frac{\sum_{k=0}^{M} b_k z^k}{\sum_{k=0}^{N} a_k z^k} \tag{5.18}$$

Factorizing $P(z)$ and $Q(z)$, (5.18) can be written as

$$X(z)=\frac{b_0(z-d_1)(z-d_2)\cdots(z-d_M)}{a_0(z-c_1)(z-c_2)\cdots(z-c_N)} \tag{5.19}$$

How many poles and zeros in (5.18)? What are they?

Example 5.1

Determine the z transform of $x[n]=a^n u[n]$ where $u[n]$ is the unit step function. Then determine the condition when the DTFT of $x[n]$ exists.

Using (5.1) and (3.3), we have

$$X(z)=\sum_{n=-\infty}^{\infty} a^n u[n] z^{-n} = \sum_{n=0}^{\infty} \left(az^{-1}\right)^n$$

According to (5.10), $X(z)$ converges if

$$\sum_{n=0}^{\infty} \left|az^{-1}\right|^n < \infty$$

Applying the ratio test, the convergence condition is

$$\left|az^{-1}\right| < 1 \Leftrightarrow |z| > |a|$$

Note that we cannot write $|z| > a$ because a may be complex

For $|z| > |a|$, $X(z)$ is computed as

$$X(z) = \sum_{n=0}^{\infty} \left(az^{-1}\right)^n = \frac{1 - \left(az^{-1}\right)^\infty}{1 - az^{-1}} = \frac{1}{1 - az^{-1}} = \frac{z}{z - a}$$

Together with the ROC, the z transform of $x[n] = a^n u[n]$ is:

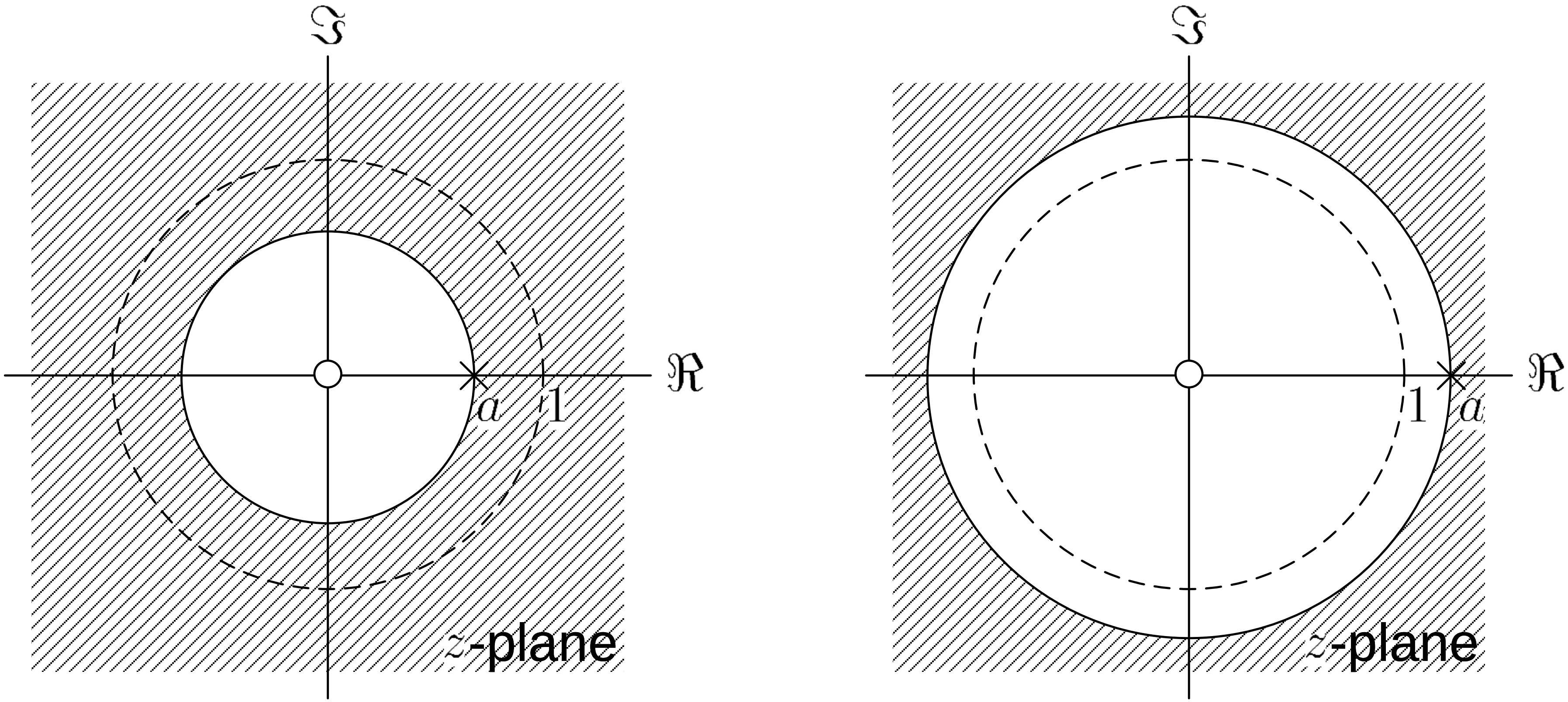
$$X(z) = \frac{z}{z - a}, \quad |z| > |a|$$

It is clear that $X(z)$ has a zero at $z = 0$ and a pole at $z = a$. Using (5.8), we substitute $z = e^{j\omega}$ to obtain

$$X(e^{j\omega}) = \frac{e^{j\omega}}{e^{j\omega} - a}, \quad |e^{j\omega}| = 1 > |a|$$

As a result, the existence condition for DTFT of $x[n]$ is $|a| < 1$.

Otherwise, its DTFT does not exist. In general, the DTFT $X(e^{j\omega})$ exists if its ROC includes the unit circle. If $|z| > |a|$ includes $|z| = 1$, $|a| < 1$ is required.



$|a| < 1$ $|a| > 1$

Fig.5.3: ROCs for $|a| < 1$ and $|a| > 1$ when $x[n] = a^n u[n]$

Example 5.2

Determine the z transform of $x[n] = -a^n u[-n-1]$. Then determine the condition when the DTFT of $x[n]$ exists.

Using (5.1) and (3.3), we have

$$X(z) = \sum_{n=-\infty}^{-1} -a^n z^{-n} = -\sum_{m=1}^{\infty} a^{-m} z^m = -\sum_{m=1}^{\infty} \left(a^{-1}z\right)^m$$

Similar to Example 5.1, $X(z)$ converges if $\left|a^{-1}z\right| < 1$ or $|z| < |a|$, which aligns with the ROC for $X_-(z)$ in (5.17). This gives

$$X(z) = -\sum_{m=1}^{\infty} \left(a^{-1}z\right)^m = -\frac{a^{-1}z\left(1-\left(a^{-1}z\right)^{\infty}\right)}{1-a^{-1}z} = -\frac{a^{-1}z}{1-a^{-1}z} = \frac{z}{z-a}$$

Together with ROC, the z transform of $x[n] = -a^n u[-n-1]$ is:

$$X(z) = \frac{z}{z-a}, \quad |z| < |a|$$

Using (5.8), we substitute $z = e^{j\omega}$ to obtain

$$X(e^{j\omega}) = \frac{e^{j\omega}}{e^{j\omega} - a}, \quad |e^{j\omega}| = 1 < |a|$$

As a result, the existence condition for DTFT of $x[n]$ is $|a| > 1$.

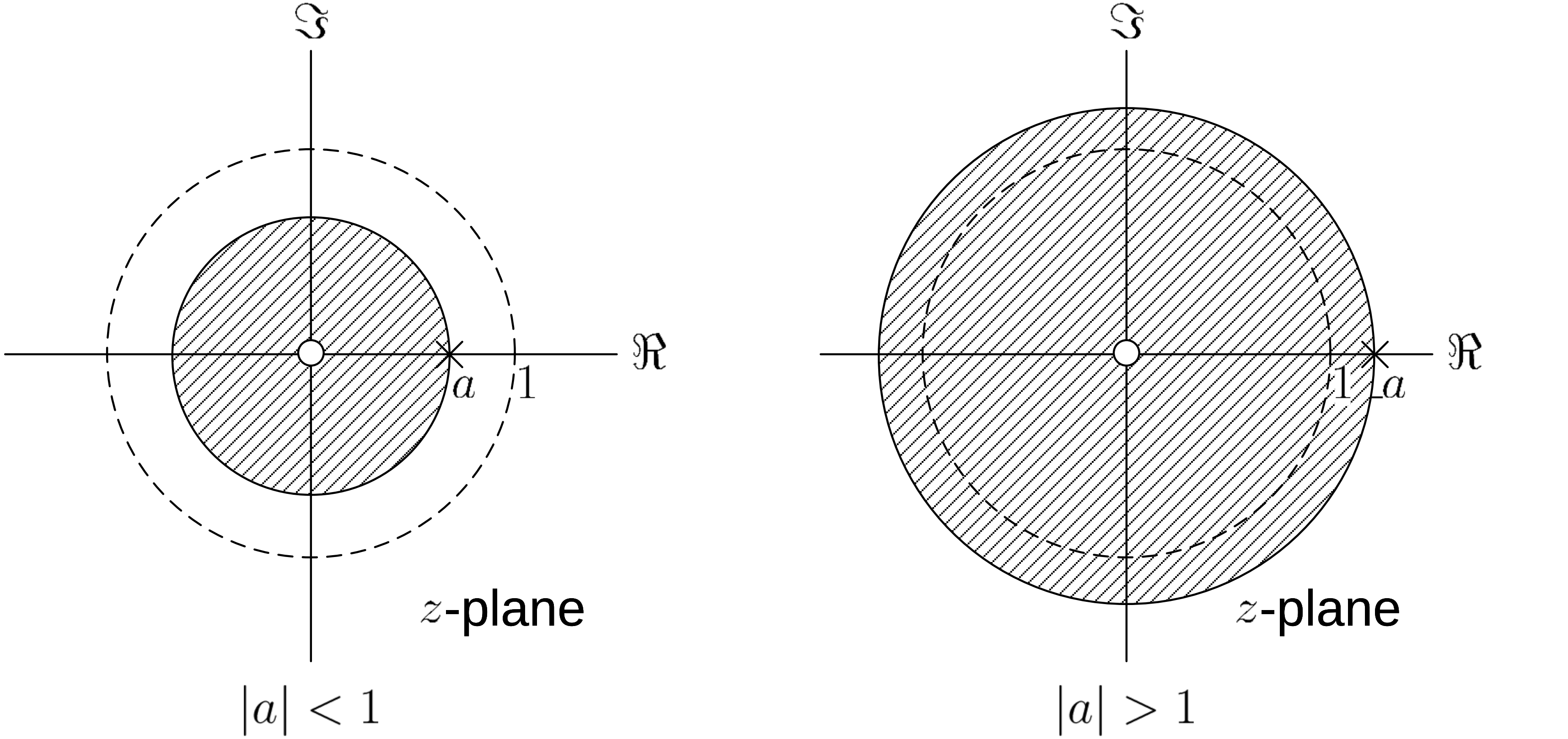


Fig.5.4: ROCs for $|a| < 1$ and $|a| > 1$ when $x[n] = -a^n u[-n-1]$

Example 5.3

Determine the z transform of $x[n] = a^n u[n] + b^n u[-n-1]$ where $|a| < |b|$.

Employing the results in Examples 5.1 and 5.2, we have

$$\begin{aligned} X(z) &= \frac{1}{1-az^{-1}} + \left(-\frac{1}{1-bz^{-1}}\right), \quad |z| > |a| \quad \text{and} \quad |z| < |b| \\ &= \frac{(a-b)z^{-1}}{\left(1-az^{-1}\right)\left(1-bz^{-1}\right)} \\ &= \frac{(a-b)z}{(z-a)(z-b)}, \quad |a| < |z| < |b| \end{aligned}$$

Note that its ROC agrees with Fig.5.2.

What are the pole(s) and zero(s) of *X*(*z*)?

Example 5.4

Determine the z transform of $x[n]=\delta[n+1]$.

Using (5.1) and (3.2), we have

$$X(z)=\sum_{n=-\infty}^{\infty}\delta[n+1]z^{-n}=z$$

Example 5.5

Determine the z transform of $x[n]$ which has the form of:

$$x[n]=\begin{cases} a^n, & 0\leq n\leq N-1 \\ 0, & \text{otherwise}\end{cases}$$

Using (5.1), we have

$$X(z)=\sum_{n=0}^{N-1}\left(az^{-1}\right)^n=\frac{1-\left(az^{-1}\right)^N}{1-az^{-1}}=\frac{1}{z^{N-1}}\frac{z^N-a^N}{z-a}$$

What are the ROCs in Examples 5.4 and 5.5?

Finite-Duration and Infinite-Duration Sequences

Finite-duration sequence: values of $x[n]$ are nonzero only for a finite time interval

Otherwise, $x[n]$ is called an infinite-duration sequence:

- Right-sided: if $x[n]=0$ for $n<N_+<\infty$ where N_+ is an integer (e.g., $x[n]=a^n u[n]$ with $N_+=0$; $x[n]=a^n u[n-10]$ with $N_+=10$; $x[n]=a^n u[n+10]$ with $N_+=-10$)
- Left-sided: if $x[n]=0$ for $n>N_->-\infty$ where N_- is an integer (e.g., $x[n]=-a^n u[-n-1]$ with $N_-=-1$)
- Two-sided: neither right-sided nor left-sided (e.g., Example 5.3)

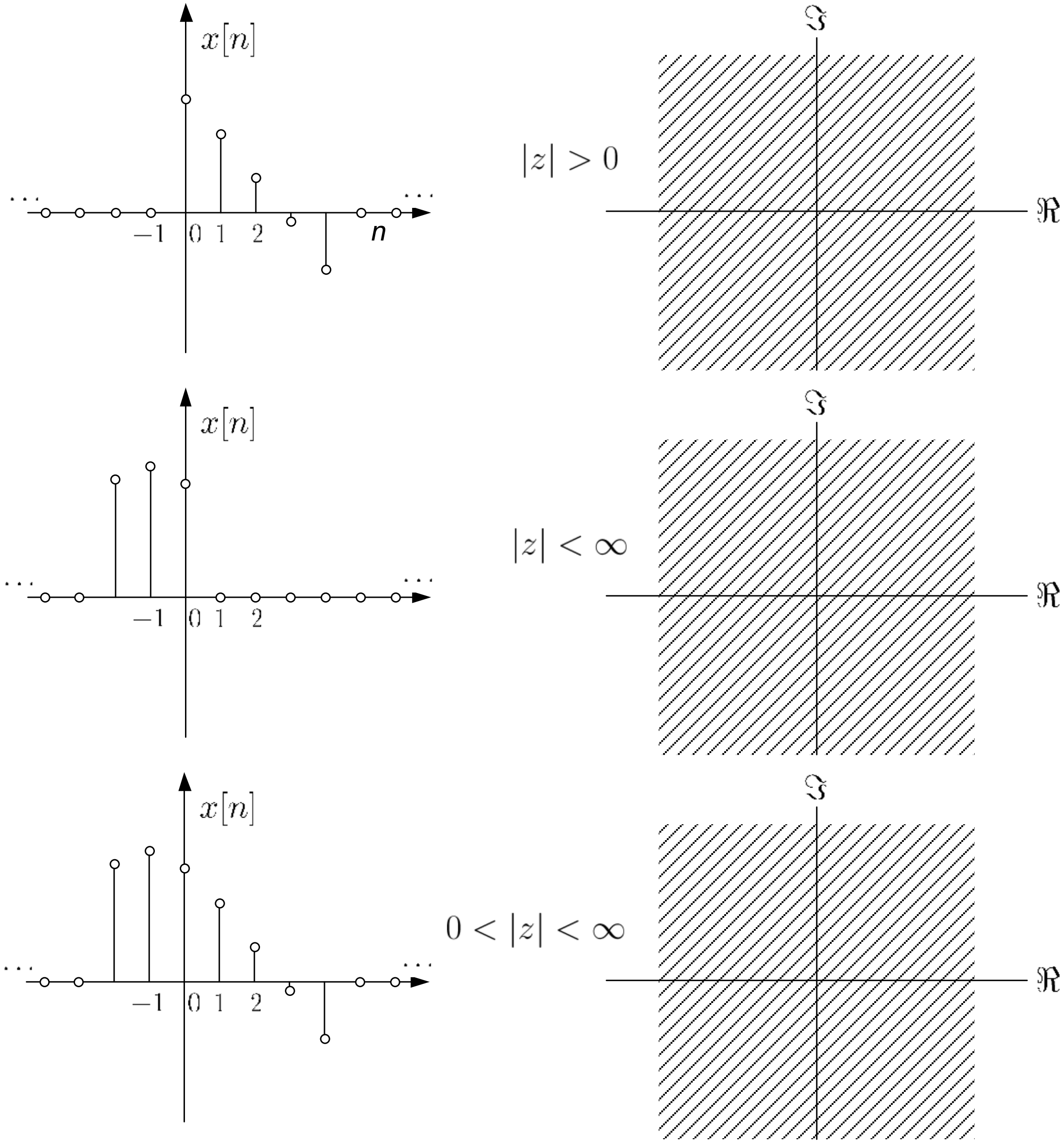


Fig.5.5: Finite-duration sequences

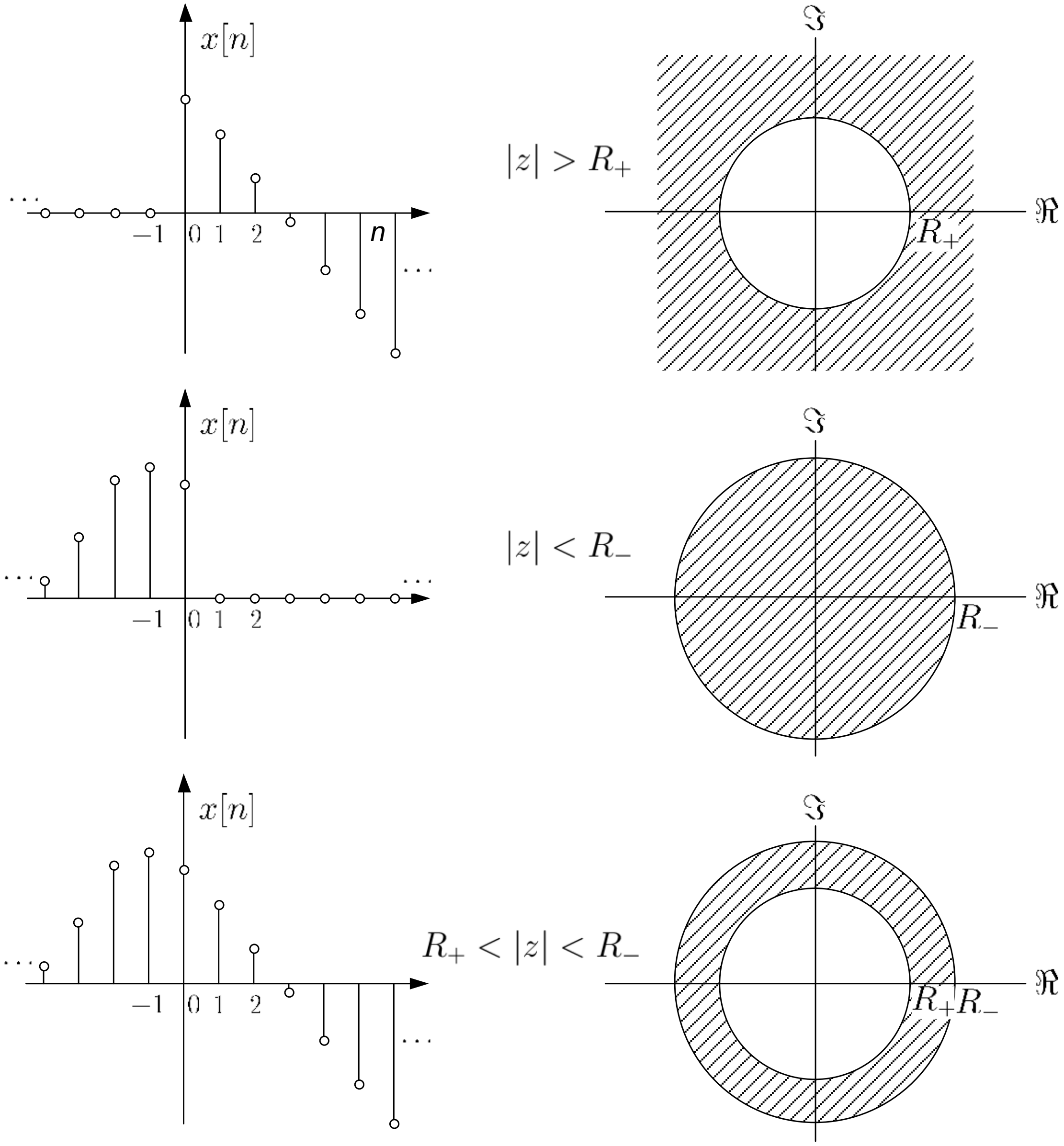


Figure 5.6: Infinite-duration sequences

Sequence	Transform	ROC
$\delta[n]$	1	All z
$\delta[n-m]$	z^{-m}	$\|z\|>0$, $m>0$; $\|z\|<\infty$, $m<0$
$a^n u[n]$	$\frac{1}{1-az^{-1}}$	$\|z\|>\|a\|$
$-a^n u[-n-1]$	$\frac{1}{1-az^{-1}}$	$\|z\|<\|a\|$
$na^n u[n]$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\|z\|>\|a\|$
$-na^n u[-n-1]$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\|z\|<\|a\|$
$a^n \cos(bn)u[n]$	$\frac{1-a\cos(b)z^{-1}}{1-2a\cos(b)z^{-1}+a^2z^{-2}}$	$\|z\|>\|a\|$
$a^n \sin(bn)u[n]$	$\frac{a\sin(b)z^{-1}}{1-2a\cos(b)z^{-1}+a^2z^{-2}}$	$\|z\|>\|a\|$

Table 5.1: z transforms for common sequences

Eight ROC properties are:

P1. There are four possible shapes for ROC, namely, the entire region except possibly $z=0$ and/or $z=\infty$, a ring, or inside or outside a circle in the z-plane centered at the origin (e.g., Figures 5.5 and 5.6)

P2. The DTFT of a sequence $x[n]$ exists if and only if the ROC of the z transform of $x[n]$ includes the unit circle (e.g., Examples 5.1 and 5.2)

P3: The ROC cannot contain any poles (e.g., Examples 5.1 to 5.5)

P4: When $x[n]$ is a finite-duration sequence, the ROC is the entire z-plane except possibly $z=0$ and/or $z=\infty$ (e.g., Examples 5.4 and 5.5)

P5: When $x[n]$ is a right-sided sequence, the ROC is of the form $|z| > |p_{\max}|$ where $p_{\max}$ is the pole with the largest magnitude in $X(z)$ (e.g., Example 5.1)

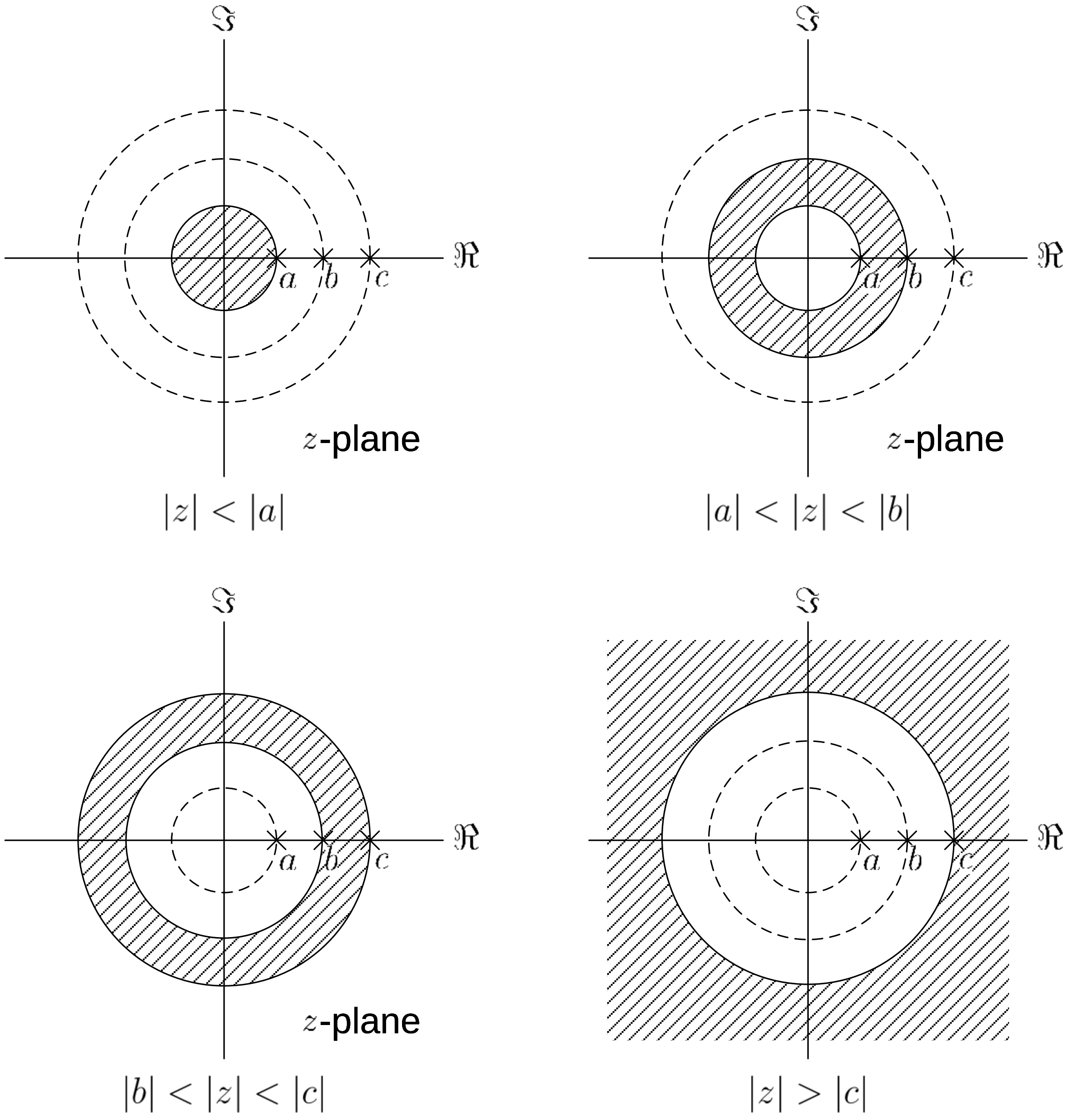
P6: When $x[n]$ is a left-sided sequence, the ROC is of the form $|z| < |p_{\min}|$ where $p_{\min}$ is the pole with the smallest magnitude in $X(z)$ (e.g., Example 5.2)

P7: When $x[n]$ is a two-sided sequence, the ROC is of the form $|p_a| < |z| < |p_b|$ where p_a and p_b are two poles with the successive magnitudes in $X(z)$ such that $|p_a| < |p_b|$ (e.g., Example 5.3)

P8: The ROC must be a connected region

Example 5.6

A z transform $X(z)$ contains three poles, namely, a, b and c with $|a| < |b| < |c|$. Determine all possible ROCs.



$|z| < |a|$

$|a| < |z| < |b|$

$|b| < |z| < |c|$

$|z| > |c|$

Fig.5.7: ROC possibilities for three poles

What are other possible ROCs?

Inverse z Transform

Inverse z transform corresponds to finding $x[n]$ given $X(z)$ and its ROC

The z transform and inverse z transform are one-to-one mapping provided that the ROC is given:

$$x[n] \leftrightarrow X(z) \quad (5.20)$$

There are 4 commonly used techniques to evaluate the inverse z transform. They are

1. Inspection
2. Partial Fraction Expansion
3. Power Series Expansion
4. Cauchy Integral Theorem

Inspection

When we are familiar with certain transform pairs, we can do the inverse z transform by inspection

Example 5.7

Determine the inverse z transform of $X(z)$ which is expressed as:

$$X(z) = \frac{z}{2z-1}, \quad |z| > 0.5$$

We first rewrite $X(z)$ as:

$$X(z) = \frac{0.5}{1-0.5z^{-1}}$$

Making use of the following transform pair in Table 5.1:

$$a^n u[n] \leftrightarrow \frac{1}{1-az^{-1}}, \quad |z| > |a|$$

and putting $a = 0.5$, we have:

$$\frac{0.5}{1-0.5z^{-1}} \leftrightarrow 0.5(0.5)^n u[n]$$

By inspection, the inverse z transform is:

$$x[n] = (0.5)^{n+1} u[n]$$

Partial Fraction Expansion

It is useful when $X(z)$ is a rational function in z^{-1}:

$$X(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} \tag{5.21}$$

For pole and zero determination, it is advantageous to multiply z^{M+N} to both numerator and denominator:

$$X(z) = \frac{z^N \sum_{k=0}^{M} b_k z^{M-k}}{z^M \sum_{k=0}^{N} a_k z^{N-k}} \tag{5.22}$$

- When $M > N$, there are $(M - N)$ pole(s) at $z = 0$
- When $M < N$, there are $(N - M)$ zero(s) at $z = 0$

To obtain the partial fraction expansion from (5.21), the first step is to determine the N nonzero poles, $c_1, c_2, \cdots, c_N$

There are 4 cases to be considered:

Case 1: $M < N$ and all poles are of first order

For first-order poles, all $\{c_k\}$ are distinct. $X(z)$ is:

$$X(z) = \sum_{k=1}^{N} \frac{A_k}{1 - c_k z^{-1}} \qquad (5.23)$$

For each first-order term of $A_k / \left(1 - c_k z^{-1}\right)$, its inverse z transform can be easily obtained by inspection

Multiplying both sides by $\left(1-c_k z^{-1}\right)$ and evaluating for $z=c_k$

$$A_k = \left(1-c_k z^{-1}\right) X(z)\Big|_{z=c_k} \tag{5.24}$$

An illustration for computing A_1 with $N=2>M$ is:

$$\begin{aligned} X(z) &= \frac{A_1}{1-c_1 z^{-1}} + \frac{A_2}{1-c_2 z^{-1}} \\ \Rightarrow \left(1-c_1 z^{-1}\right) X(z) &= A_1 + \frac{A_2\left(1-c_1 z^{-1}\right)}{1-c_2 z^{-1}} \end{aligned} \tag{5.25}$$

Substituting $z=c_1$, we get A_1

In summary, three steps are:

- Find poles
- Find $\{A_k\}$
- Perform inverse z transform for the fractions by inspection

Example 5.8

Find the pole and zero locations of $H(z)$:

$$H(z) = -\frac{1+0.1z^{-1}}{1-2.05z^{-1}+z^{-2}}$$

Then determine the inverse z transform of $H(z)$.

We first multiply z^2 to both numerator and denominator polynomials to obtain:

$$H(z) = -\frac{z\left(z+0.1\right)}{z^2-2.05z+1}$$

Apparently, there are two zeros at $z=0$ and $z=-0.1$. On the other hand, by solving the quadratic equation at the denominator polynomial, the poles are determined as $z=0.8$ and $z=1.25$.

According to (5.23), we have:

$$H(z) = \frac{A_1}{1-0.8z^{-1}} + \frac{A_2}{1-1.25z^{-1}}$$

Employing (5.24), A_1 is calculated as:

$$A_1 = \left(1-0.8z^{-1}\right) H(z)\Big|_{z=0.8} = -\left.\frac{1+0.1z^{-1}}{1-1.25z^{-1}}\right|_{z=0.8} = 2$$

Similarly, A_2 is found to be -3. As a result, the partial fraction expansion for $H(z)$ is

$$H(z) = \frac{2}{1-0.8z^{-1}} - \frac{3}{1-1.25z^{-1}}$$

As the ROC is not specified, we investigate all possible scenarios, namely, $|z| > 1.25$, $0.8 < |z| < 1.25$, and $|z| < 0.8$.

For $|z| > 1.25$, we notice that

$$(0.8)^n u[n] \leftrightarrow \frac{1}{1-0.8z^{-1}}, \quad |z| > 0.8$$

and

$$(1.25)^n u[n] \leftrightarrow \frac{1}{1-1.25z^{-1}}, \quad |z| > 1.25$$

where both ROCs agree with $|z| > 1.25$. Combining the results, the inverse z transform $h[n]$ is:

$$h[n] = \left(2(0.8)^n - 3(1.25)^n\right) u[n]$$

which is a right-sided sequence and aligns with P5.

For $0.8 < |z| < 1.25$, we make use of

$$(0.8)^n u[n] \leftrightarrow \frac{1}{1-0.8z^{-1}}, \quad |z| > 0.8$$

and

$$-(1.25)^n u[-n-1] \leftrightarrow \frac{1}{1-1.25z^{-1}}, \quad |z| < 1.25$$

where both ROCs agree with $0.8 < |z| < 1.25$. This implies:

$$h[n] = 2(0.8)^n u[n] + 3(1.25)^n u[-n-1]$$

which is a two-sided sequence and aligns with P7.

Finally, for $|z| < 0.8$:

$$-(0.8)^n u[-n-1] \leftrightarrow \frac{1}{1-0.8z^{-1}}, \quad |z| < 0.8$$

and

$$-(1.25)^n u[-n-1] \leftrightarrow \frac{1}{1-1.25z^{-1}}, \quad |z| < 1.25$$

where both ROCs agree with $|z| < 0.8$. As a result, we have:

$$h[n] = \left(-2(0.8)^n + 3(1.25)^n\right) u[-n-1]$$

which is a left-sided sequence and aligns with P6.

Suppose $h[n]$ is the impulse response of a discrete-time LTI system. Recall (3.15) and (3.16):

$$h[n] = 0, \quad n < 0$$

and

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

The three possible impulse responses:

- $h[n] = (2(0.8)^n - (1.25)^n)\, u[n]$ is the impulse response of a causal but unstable system
- $h[n] = 2(0.8)^n u[n] + (1.25)^n u[-n-1]$ corresponds to a noncausal but stable system
- $h[n] = (-2(0.8)^n + (1.25)^n)\, u[-n-1]$ is noncausal and unstable

Which of the h[n] has/have DTFT?

Case 2: $M \geq N$ and all poles are of first order

In this case, $X(z)$ can be expressed as:

$$X(z) = \sum_{l=0}^{M-N} B_l z^{-l} + \sum_{k=1}^{N} \frac{A_k}{1 - c_k z^{-1}} \qquad (5.26)$$

- B_l are obtained by long division of the numerator by the denominator, with the division process terminating when the remainder is of lower degree than the denominator
- A_k can be obtained using (5.24).

Example 5.9

Determine $x[n]$ which has z transform of the form:

$$X(z) = \frac{4 - 2z^{-1} + z^{-2}}{1 - 1.5z^{-1} + 0.5z^{-2}}, \quad |z| > 1$$

The poles are easily determined as $z = 0.5$ and $z = 1$

According to (5.26) with $M = N = 2$:

$$X(z) = B_0 + \frac{A_1}{1 - 0.5z^{-1}} + \frac{A_2}{1 - z^{-1}}$$

The value of B_0 is found by dividing the numerator polynomial by the denominator polynomial as follows:

$$\begin{array}{rl} & \quad 2 \\ 0.5z^{-2} - 1.5z^{-1} + 1 \big) \!\!\!\! & \overline{z^{-2} - 2z^{-1} + 4} \\ & \underline{z^{-2} - 3z^{-1} + 2} \\ & \qquad\quad z^{-1} + 2 \end{array}$$

That is, $B_0 = 2$. Thus $X(z)$ is expressed as

$$X(z) = 2 + \frac{2 + z^{-1}}{\left(1 - 0.5z^{-1}\right)\left(1 - z^{-1}\right)} = 2 + \frac{A_1}{1 - 0.5z^{-1}} + \frac{A_2}{1 - z^{-1}}$$

According to (5.24), A_1 and A_2 are calculated as

$$A_1 = \left.\frac{4-2z^{-1}+z^{-2}}{1-z^{-1}}\right|_{z=0.5} = -4$$

and

$$A_2 = \left.\frac{4-2z^{-1}+z^{-2}}{1-0.5z^{-1}}\right|_{z=1} = 6$$

With $|z|>1$:

$$\delta[n] \leftrightarrow 1$$

$$(0.5)^n u[n] \leftrightarrow \frac{1}{1-0.5z^{-1}}, \quad |z|>0.5$$

and

$$u[n] \leftrightarrow \frac{1}{1-z^{-1}}, \quad |z|>1$$

the inverse z transform $x[n]$ is:

$$x[n] = 2\delta[n] - 4(0.5)^n u[n] + 6u[n]$$

Case 3: $M < N$ with multiple-order pole(s)

If $X(z)$ has a s-order pole at $z = c_i$ with $s \geq 2$, this means that there are s repeated poles with the same value of c_i. $X(z)$ is:

$$X(z) = \sum_{k=1, k\neq i}^{N} \frac{A_k}{1 - c_k z^{-1}} + \sum_{m=1}^{s} \frac{C_m}{\left(1 - c_i z^{-1}\right)^m} \qquad (5.27)$$

- When there are two or more multiple-order poles, we include a component like the second term for each corresponding pole
- A_k can be computed according to (5.24)
- C_m can be calculated from:

$$C_m = \frac{1}{(s-m)!(-c_i)^{s-m}} \cdot \frac{d^{s-m}}{dw^{s-m}} \left[(1 - c_i w)^s X(w^{-1}) \right] \Big|_{w=c_i^{-1}} \qquad (5.28)$$

Example 5.10

Determine the partial fraction expansion for $X(z)$:

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

It is clear that $X(z)$ corresponds to Case 3 with $N=3>M$ and one second-order pole at $z=1$. Hence $X(z)$ is:

$$X(z) = \frac{A_1}{1+z^{-1}} + \frac{C_1}{1-z^{-1}} + \frac{C_2}{\left(1-z^{-1}\right)^2}$$

Employing (5.24), A_1 is:

$$A_1 = \left.\frac{4}{\left(1-z^{-1}\right)^2}\right|_{z=-1} = 1$$

Applying (5.28), C_1 is:

$$\begin{aligned} C_1 &= \frac{1}{(2-1)!(-1)^{2-1}} \cdot \frac{d}{dw}\left[(1-1\cdot w)^2 \frac{4}{(1+w)(1-w)^2}\right]\Bigg|_{w=1} \\ &= -\frac{d}{dw}\frac{4}{1+w}\bigg|_{w=1} \\ &= \frac{4}{(1+w)^2}\bigg|_{w=1} \\ &= 1 \end{aligned}$$

and

$$\begin{aligned} C_2 &= \frac{1}{(2-2)!(-1)^{2-2}} \cdot \left[(1-1\cdot w)^2 \frac{4}{(1+w)(1-w)^2}\right]\Bigg|_{w=1} \\ &= \frac{4}{1+w}\bigg|_{w=1} \\ &= 2 \end{aligned}$$

Therefore, the partial fraction expansion for $X(z)$ is

$$X(z)=\frac{1}{1+z^{-1}}+\frac{1}{1-z^{-1}}+\frac{2}{\left(1-z^{-1}\right)^2}$$

Case 4: $M \geq N$ with multiple-order pole(s)

This is the most general case and the partial fraction expansion of $X(z)$ is

$$X(z)=\sum_{l=0}^{M-N} B_l z^{-l}+\sum_{k=1,k\neq i}^{N}\frac{A_k}{1-c_k z^{-1}}+\sum_{m=1}^{s}\frac{C_m}{\left(1-c_i z^{-1}\right)^m} \quad (5.29)$$

assuming that there is only one multiple-order pole of order $s \geq 2$ at $z=c_i$. It is easily extended to the scenarios when there are two or more multiple-order poles as in Case 3. The A_k , B_l and C_m can be calculated as in Cases 1, 2 and 3

Power Series Expansion

When $X(z)$ is expanded as power series according to (5.1):

$$X(z)=\sum_{n=-\infty}^{\infty}x[n]z^{-n}=\cdots+x[-1]z^{1}+x[0]+x[1]z^{-1}+x[2]z^{-2}+\cdots \quad (5.30)$$

any particular value of $x[n]$ can be determined by finding the coefficient of the appropriate power of z^{-1}

Example 5.11

Determine $x[n]$ which has z transform of the form:

$$X(z)=2z^{2}\left(1-0.5z^{-1}\right)\left(1+z^{-1}\right)\left(1-z^{-1}\right), \quad 0<|z|<\infty$$

Expanding $X(z)$ yields

$$X(z)=2z^{2}-z-2+z^{-1}$$

From (5.30), $x[n]$ is deduced as:

$$x[n]=2\delta[n+2]-\delta[n+1]-2\delta[n]+\delta[n-1]$$

Example 5.12

Determine $x[n]$ whose z transform is given as:

$$X(z) = \log(1 + az^{-1}), \quad |z| > |a|$$

With the use of the power series expansion for $\log(1+\lambda)$:

$$\log(1+\lambda) = \sum_{n=1}^{\infty} \frac{(-1)^{n+1}\lambda^n}{n}, \quad |\lambda| < 1$$

$X(z)$ with $|az^{-1}| < 1$ can be expressed as

$$\log(1 + az^{-1}) = \sum_{n=1}^{\infty} \frac{(-1)^{n+1}a^n z^{-n}}{n}$$

From (5.30), $x[n]$ is deduced as:

$$x[n] = \frac{(-1)^{n+1}a^n}{n} u[n-1]$$

Example 5.13

Determine $x[n]$ whose z transform has the form of:

$$X(z) = \frac{1}{1 - az^{-1}}, \quad |z| > |a|$$

With the use of

$$\frac{1}{1-\lambda} = 1 + \lambda + \lambda^2 + \cdots, \quad |\lambda| < 1$$

Carrying out long division in $X(z)$ with $|az^{-1}| < 1$:

$$X(z) = 1 + az^{-1} + \left(az^{-1}\right)^2 + \cdots$$

From (5.30), $x[n]$ is deduced as:

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

which agrees with Example 5.1 and Table 5.1

Example 5.14

Determine $x[n]$ whose z transform has the form of:

$$X(z) = \frac{1}{1 - az^{-1}}, \quad |z| < |a|$$

We first express $X(z)$ as:

$$X(z) = \frac{-a^{-1}z}{-a^{-1}z} \cdot \frac{1}{1 - az^{-1}} = \frac{-a^{-1}z}{1 - a^{-1}z}$$

Carrying out long division in $X(z)$ with $|a^{-1}z| < 1$:

$$X(z) = -a^{-1}z\left(1 + a^{-1}z + \left(a^{-1}z\right)^2 + \cdots\right)$$

From (5.30), $x[n]$ is deduced as:

$$x[n] = -a^n u[-n-1]$$

which agrees with Example 5.2 and Table 5.1

Properties of z Transform

1. Linearity

Let $(x_1[n], X_1(z))$ and $(x_2[n], X_2(z))$ be two z transform pairs with ROCs $\mathcal{R}_{x_1}$ and $\mathcal{R}_{x_2}$, respectively, we have

$$a x_1[n] + b x_2[n] \leftrightarrow a X_1(z) + b X_2(z) \qquad (5.31)$$

Its ROC is denoted by $\mathcal{R}$, which includes $\mathcal{R}_{x_1} \cap \mathcal{R}_{x_2}$ where $\cap$ is the intersection operator. That is, $\mathcal{R}$ contains at least the intersection of $\mathcal{R}_{x_1}$ and $\mathcal{R}_{x_2}$.

Example 5.15

Determine the z transform of $y[n]$ which is expressed as:

$$y[n] = x_1[n] + x_2[n]$$

where $x_1[n] = (0.2)^n u[n]$ and $x_2[n] = (-0.3)^n u[n]$. By inspection, the z transforms of $x_1[n]$ and $x_2[n]$ are:

$$x_1[n] = (0.2)^n u[n] \leftrightarrow \frac{1}{1-0.2z^{-1}}, \quad |z| > 0.2$$

and

$$x_2[n] = (-0.3)^n u[n] \leftrightarrow \frac{1}{1+0.3z^{-1}}, \quad |z| > 0.3$$

According to the linearity property, the z transform of $y[n]$ is

$$Y(z) = \frac{1}{1-0.2z^{-1}} + \frac{1}{1+0.3z^{-1}}, \quad |z| > 0.3$$

Why the ROC is |z|>0.3 instead of |z|>0.2?

2. Time Shifting

A time-shift of n_0 in $x[n]$ causes a multiplication of z^{-n_0} in $X(z)$

$$x[n-n_0] \leftrightarrow z^{-n_0} X(z) \tag{5.32}$$

The ROC for $x[n-n_0]$ is basically identical to that of $X(z)$ except for the possible addition or deletion of $z=0$ or $z=\infty$

Example 5.16

Find the z transform of $x[n]$ which has the form of:

$$x[n] = a^{n-1}u[n-1]$$

Employing the time-shifting property with $n_0 = 1$ and:

$$a^n u[n] \leftrightarrow \frac{1}{1-az^{-1}}, \quad |z| > |a|$$

we easily obtain

$$a^{n-1}u[n-1] \leftrightarrow z^{-1} \cdot \frac{1}{1-az^{-1}} = \frac{z^{-1}}{1-az^{-1}}, \quad |z| > |a|$$

Note that using (5.1) with $|z| > |a|$ also produces the same result but this approach is less efficient:

$$X(z) = \sum_{n=1}^{\infty} a^{n-1}z^{-n} = a^{-1}\sum_{n=1}^{\infty} \left(az^{-1}\right)^n = a^{-1}\frac{az^{-1}\left[1-\left(az^{-1}\right)^{\infty}\right]}{1-az^{-1}} = \frac{z^{-1}}{1-az^{-1}}$$

3. Multiplication by an Exponential Sequence (Modulation)

If we multiply $x[n]$ by z_0^n in the time domain, the variable z will be changed to z/z_0 in the z transform domain. That is:

$$z_0^n x[n] \leftrightarrow X(z/z_0) \tag{5.33}$$

If the ROC for $x[n]$ is $R_+ < |z| < R_-$, the ROC for $z_0^n x[n]$ is $|z_0|R_+ < |z| < |z_0|R_-$

Example 5.17

With the use of the following z transform pair:

$$u[n] \leftrightarrow \frac{1}{1-z^{-1}}, \quad |z| > 1$$

Find the z transform of $x[n]$ which has the form of:

$$x[n] = a^n \cos(bn) u[n]$$

Noting that $\cos(bn) = (e^{jbn} + e^{-jbn})/2$, $x[n]$ can be written as:

$$x[n] = \frac{1}{2}\left(ae^{jb}\right)^n u[n] + \frac{1}{2}\left(ae^{-jb}\right)^n u[n]$$

By means of the modulation property of (5.33) with the substitution of $z_0 = ae^{jb}$ and $z_0 = ae^{-jb}$, we obtain:

$$\frac{1}{2}\left(ae^{jb}\right)^n u[n] \leftrightarrow \frac{1}{2}\frac{1}{1-(z/(ae^{jb}))^{-1}} = \frac{1}{2}\frac{1}{1-ae^{jb}z^{-1}}, \quad |z| > |a|$$

and

$$\frac{1}{2}\left(ae^{-jb}\right)^n u[n] \leftrightarrow \frac{1}{2}\frac{1}{1-(z/(ae^{-jb}))^{-1}} = \frac{1}{2}\frac{1}{1-ae^{-jb}z^{-1}}, \quad |z| > |a|$$

By means of the linearity property, it follows that

$$X(z) = \frac{1}{2}\frac{1}{1-ae^{jb}z^{-1}} + \frac{1}{2}\frac{1}{1-ae^{-jb}z^{-1}} = \frac{1-a\cos(b)z^{-1}}{1-2a\cos(b)z^{-1}+a^2z^{-2}}, |z| > |a|$$

which agrees with Table 5.1.

4. Differentiation

Differentiating $X(z)$ with respect to z corresponds to multiplying $x[n]$ by n in the time domain:

$$nx[n] \leftrightarrow -z\frac{dX(z)}{dz} \tag{5.34}$$

The ROC for $nx[n]$ is basically identical to that of $X(z)$ except for the possible addition or deletion of $z=0$ or $z=\infty$

<u>Example 5.18</u>

Determine the z transform of $x[n]=na^n u[n]$.

Since

$$a^n u[n] \leftrightarrow \frac{1}{1-az^{-1}}, \quad |z|>|a|$$

and

$$\frac{d}{dz}\left(\frac{1}{1-az^{-1}}\right) = \frac{d\left(1-az^{-1}\right)^{-1}}{d\left(1-az^{-1}\right)} \cdot \frac{d\left(1-az^{-1}\right)}{dz} = -\frac{az^{-2}}{\left(1-az^{-1}\right)^2}$$

By means of the differentiation property, we have

$$na^n u[n] \leftrightarrow -z \cdot -\frac{az^{-2}}{\left(1-az^{-1}\right)^2} = \frac{az^{-1}}{\left(1-az^{-1}\right)^2}, \quad |z| > |a|$$

which agrees with Table 5.1.

5. Conjugation

The z transform pair for $x^*[n]$ is:

$$x^*[n] \leftrightarrow X^*(z^*) \tag{5.35}$$

The ROC for $x^*[n]$ is identical to that of $x[n]$

6. Time Reversal

The z transform pair for $x[-n]$ is:

$$x[-n] \leftrightarrow X(z^{-1}) \tag{5.36}$$

If the ROC for $x[n]$ is $R_+ < |z| < R_-$, the ROC for $x[-n]$ is $1/R_- < |z| < 1/R_+$

Example 5.19

Determine the z transform of $x[n] = -na^{-n}u[-n]$

Using Example 5.18:

$$na^n u[n] \leftrightarrow \frac{az^{-1}}{(1-az^{-1})^2}, \quad |z| > |a|$$

and from the time reversal property:

$$X(z) = \frac{az}{(1-az)^2} = \frac{a^{-1}z^{-1}}{(1-a^{-1}z^{-1})^2}, \quad |z| < |a^{-1}|$$

7. Convolution

Let $(x_1[n], X_1(z))$ and $(x_2[n], X_2(z))$ be two z transform pairs with ROCs $\mathcal{R}_{x_1}$ and $\mathcal{R}_{x_2}$, respectively. Then we have:

$$x_1[n] \otimes x_2[n] \leftrightarrow X_1(z)X_2(z) \tag{5.37}$$

and its ROC includes $\mathcal{R}_{x_1} \cap \mathcal{R}_{x_2}$.

The proof is given as follows.

Let

$$y[n] = x_1[n] \otimes x_2[n] = \sum_{k=-\infty}^{\infty} x_1[k]x_2[n-k] \tag{5.38}$$

With the use of the time shifting property, $Y(z)$ is:

$$
\begin{aligned}
Y(z) &= \sum_{n=-\infty}^{\infty} \left[\sum_{k=-\infty}^{\infty} x_1[k] x_2[n-k] \right] z^{-n} \\
&= \sum_{k=-\infty}^{\infty} x_1[k] \left[\sum_{n=-\infty}^{\infty} x_2[n-k] z^{-n} \right] \\
&= \sum_{k=-\infty}^{\infty} x_1[k] X_2(z) z^{-k} \\
&= X_1(z) X_2(z) \qquad (5.39)
\end{aligned}
$$

Transfer Function of Linear Time-Invariant System

A LTI system can be characterized by the transfer function, which is a z transform expression

Starting with:

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k] \tag{5.40}$$

Applying z transform on (5.40) with the use of the linearity and time shifting properties, we have

$$Y(z)\sum_{k=0}^{N} a_k z^{-k} = X(z)\sum_{k=0}^{M} b_k z^{-k} \tag{5.41}$$

The transfer function, denoted by $H(z)$, is defined as:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\displaystyle\sum_{k=0}^{M} b_k z^{-k}}{\displaystyle\sum_{k=0}^{N} a_k z^{-k}} \tag{5.42}$$

The system impulse response $h[n]$ is given by the inverse z transform of $H(z)$ with an appropriate ROC, that is, $h[n] \leftrightarrow H(z)$, such that $y[n] = x[n] \otimes h[n]$. This suggests that we can first take the z transforms for $x[n]$ and $h[n]$, then multiply $X(z)$ by $H(z)$, and finally perform the inverse z transform of $X(z)H(z)$.

Example 5.20

Determine the transfer function for a LTI system whose input $x[n]$ and output $y[n]$ are related by:

$$y[n] = 0.1y[n-1] + x[n] + x[n-1]$$

Applying z transform on the difference equation with the use of the linearity and time shifting properties, $H(z)$ is:

$$Y(z)\left(1 - 0.1z^{-1}\right) = X(z)\left(1 + z^{-1}\right) \Rightarrow H(z) = \frac{Y(z)}{X(z)} = \frac{1 + z^{-1}}{1 - 0.1z^{-1}}$$

Note that there are two ROC possibilities, namely, $|z| > 0.1$ and $|z| < 0.1$ and we cannot uniquely determine $h[n]$

Example 5.21

Find the difference equation of a LTI system whose transfer function is given by

$$H(z) = \frac{\left(1+z^{-1}\right)\left(1-2z^{-1}\right)}{\left(1-0.5z^{-1}\right)\left(1+2z^{-1}\right)}$$

Let $H(z) = Y(z)/X(z)$. Performing cross-multiplication and inverse z transform, we obtain:

$$\begin{aligned}
&\left(1-0.5z^{-1}\right)\left(1+2z^{-1}\right)Y(z) = \left(1+z^{-1}\right)\left(1-2z^{-1}\right)X(z) \\
&\Rightarrow \left(1+1.5z^{-1}-z^{-2}\right)Y(z) = \left(1-z^{-1}-2z^{-2}\right)X(z) \\
&\Rightarrow y[n]+1.5y[n-1]-y[n-2] = x[n]-x[n-1]-2x[n-2]
\end{aligned}$$

Examples 5.20 and 5.21 imply the equivalence between the difference equation and transfer function

Example 5.22

Compute the impulse response $h[n]$ for a LTI system which is characterized by the following difference equation:

$$y[n] = x[n] - x[n-1]$$

Applying z transform on the difference equation with the use of the linearity and time shifting properties, $H(z)$ is:

$$Y(z) = X(z)\left(1 - z^{-1}\right) \Rightarrow H(z) = \frac{Y(z)}{X(z)} = 1 - z^{-1}$$

There is only one ROC possibility, namely, $|z| > 0$. Taking the inverse z transform on $H(z)$, we get:

$$h[n] = \delta[n] - \delta[n-1]$$

which agrees with Example 3.12

Example 5.23

Determine the output $y[n]$ if the input is $x[n] = u[n]$ and the LTI system impulse response is $h[n] = \delta[n] + 0.5\delta[n-1]$

The z transforms for $x[n]$ and $h[n]$ are

$$X(z) = \frac{1}{1 - z^{-1}}, \quad |z| > 1$$

and

$$H(z) = 1 + 0.5z^{-1} \quad |z| > 0$$

As a result, we have:

$$Y(z) = X(z)H(z) = \frac{1}{1 - z^{-1}} + 0.5\frac{z^{-1}}{1 - z^{-1}}, \quad |z| > 1$$

Taking the inverse z transform of $Y(z)$ with the use of the time shifting property yields:

$$y[n] = u[n] + 0.5u[n-1]$$

Chapter 6: Discrete-Time Fourier Transform (DTFT)

Chapter Intended Learning Outcomes:

(i) Understanding the characteristics and properties of DTFT

(ii) Ability to perform discrete-time signal conversion between the time and frequency domains using DTFT and inverse DTFT

Definition

DTFT is a frequency analysis tool for aperiodic discrete-time signals

The DTFT of $x[n]$, $X(e^{j\omega})$, has been derived in (5.4):

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \tag{6.1}$$

The derivation is based on taking the Fourier transform of $x_s(t)$ of (5.2)

As in Fourier transform, $X(e^{j\omega})$ is also called spectrum and is a continuous function of the frequency parameter ω

To convert $X(e^{j\omega})$ to $x[n]$, we use inverse DTFT:

$$x[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n}d\omega \qquad (6.2)$$

Proof: Putting (6.1) into (6.2) and using (4.13)-(4.14):

$$\begin{aligned}
\frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n}d\omega &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\left[\sum_{m=-\infty}^{\infty} x[m]e^{-j\omega m}\right]e^{j\omega n}d\omega \\
&= \frac{1}{2\pi}\sum_{m=-\infty}^{\infty} x[m]\int_{-\pi}^{\pi} e^{-j\omega m}e^{j\omega n}d\omega \\
&= \frac{1}{2\pi}\sum_{m=-\infty}^{\infty} x[m]\frac{2\sin((n-m)\pi)}{n-m} \\
&= x[n] \qquad (6.3)
\end{aligned}$$

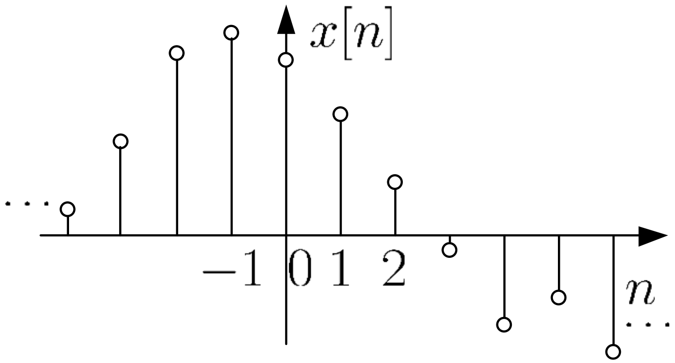
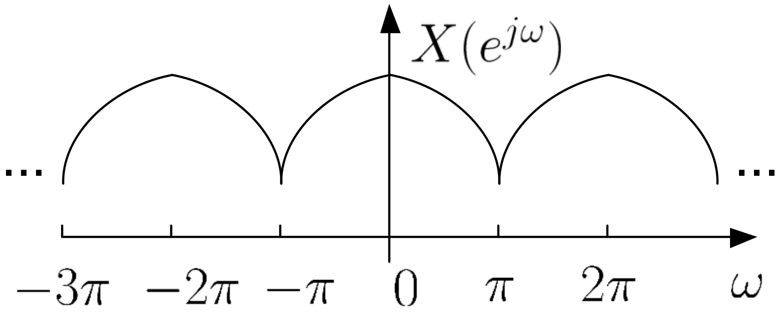
time domain	frequency domain
 $X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \Longrightarrow$	 $\Longleftarrow x[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(e^{j\omega})e^{j\omega n}d\omega$
discrete and aperiodic	continuous and periodic

Fig.6.1: Illustration of DTFT

$X(e^{j\omega})$ is continuous and periodic with a period of 2π

$X(e^{j\omega})$ is generally complex, we can illustrate $X(e^{j\omega})$ using the magnitude and phase spectra, i.e., $|X(e^{j\omega})|$ and $\angle(X(e^{j\omega}))$:

$$|X(e^{j\omega})| = \sqrt{(\Re\{X(e^{j\omega})\})^2 + (\Im\{X(e^{j\omega})\})^2} \tag{6.4}$$

and

$$\angle(X(e^{j\omega})) = \tan^{-1}\left(\frac{\Im\{X(e^{j\omega})\}}{\Re\{X(e^{j\omega})\}}\right) \tag{6.5}$$

where both are continuous in frequency and periodic.

Convergence of DTFT

The DTFT of a sequence $x[n]$ converges if

$$\left|X(e^{j\omega})\right| = \left|\sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}\right| \leq \sum_{n=-\infty}^{\infty} |x[n]| \cdot \left|e^{-j\omega n}\right| = \sum_{n=-\infty}^{\infty} |x[n]| < \infty \quad (6.6)$$

Recall (5.10) and assume the z transform of $x[n]$ converges for region of convergence (ROC) of $R_+ < |z| < R_-$:

$$|X(z)| = \left|\sum_{n=-\infty}^{\infty} x[n]z^{-n}\right| \leq \sum_{n=-\infty}^{\infty} \left|x[n]z^{-n}\right| < \infty, \quad R_+ < |z| < R_- \quad (6.7)$$

When ROC includes the unit circle:

$$\sum_{n=-\infty}^{\infty} \left|x[n]z^{-n}\right|\Bigg|_{|z|=1} = \sum_{n=-\infty}^{\infty} |x[n]| < \infty \quad (6.8)$$

which leads to the convergence condition for $X(e^{j\omega})$. This also proves the P2 property of the z transform.

Let $h[n]$ be the impulse response of a linear time-invariant (LTI) system, the following three statements are equivalent:

S1. ROC for the z transform of $h[n]$ includes unit circle

S2. The system is stable so that $\sum_{n=-\infty}^{\infty} |h[n]| < \infty$

S3. The DTFT of $h[n]$, i.e., $H(e^{j\omega})$, converges

Note that $H(e^{j\omega})$ is also known as system frequency response

Example 6.1

Determine the DTFT of $x[n] = u[n]$.

Using (6.1), the DTFT of $x[n]$ is computed as:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = \sum_{n=0}^{\infty} e^{-j\omega n}$$

Since

$$\sum_{n=0}^{\infty} \left|e^{-j\omega n}\right| = \sum_{n=0}^{\infty} 1 = \infty$$

$X(e^{j\omega})$ does not exist.

Alternatively, employing the stability condition:

$$\sum_{n=-\infty}^{\infty} |u[n]| = \sum_{n=0}^{\infty} 1 = \infty$$

which also indicates that the DTFT does not converge

Furthermore, the z transform of $x[n]$ is:

$$u[n] \leftrightarrow \frac{1}{1-z^{-1}}, \quad |z|>1$$

Because $|z|>1$ does not include the unit circle, there is no DTFT for $x[n]$.

Example 6.2

Find the DTFT of $x[n]=u[n]-u[n-N]$. Plot the magnitude and phase spectra for $N=10$.

Using (6.1), we have

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = \sum_{n=0}^{N-1} e^{-j\omega n} = \frac{e^{-j\omega \cdot 0}\left(1-e^{-j\omega N}\right)}{1-e^{-j\omega}} = \frac{1-e^{-j\omega N}}{1-e^{-j\omega}}$$

Alternatively, we can first use z transform because

$$X(e^{j\omega}) = X(z)\big|_{z=e^{j\omega}}$$

The z transform of $x[n]$ is evaluated as

$$X(z) = \sum_{n=0}^{N-1} z^{-n} = \frac{1-z^{-N}}{1-z^{-1}}, \quad |z| > 0$$

As the ROC includes the unit circle, its DTFT exists and the same result is obtained by the substitution of $z = e^{j\omega}$.

There are two advantages of z transform over DTFT:

- z transform is a generalization of DTFT and it encompasses a broader class of signals since DTFT does not converge for all sequences
- notation convenience of writing z instead of $e^{j\omega}$.

To plot the magnitude and phase spectra, we express $X(e^{j\omega})$:

$$X(e^{j\omega}) = \frac{1-e^{-j\omega N}}{1-e^{-j\omega}} = \frac{e^{-j\omega N/2}}{e^{-j\omega/2}} \cdot \frac{e^{j\omega N/2}-e^{-j\omega N/2}}{e^{j\omega/2}-e^{-j\omega/2}} = e^{-j\omega(N-1)/2} \cdot \frac{\sin(\omega N/2)}{\sin(\omega/2)}$$

In doing so, $|X(e^{j\omega})|$ and $\angle(X(e^{j\omega}))$ can be written in closed-forms as:

$$|X(e^{j\omega})| = \left|\frac{\sin(\omega N/2)}{\sin(\omega/2)}\right|$$

and

$$\angle(X(e^{j\omega})) = -\frac{\omega(N-1)}{2} + \angle\left(\frac{\sin(\omega N/2)}{\sin(\omega/2)}\right)$$

Note that we generally employ (6.4) and (6.5) for magnitude and phase computation

In using MATLAB to plot $|X(e^{j\omega})|$ and $\angle(X(e^{j\omega}))$, we utilize the command `sinc` so that there is no need to separately handle the "0/0" cases due to the sine functions

Recall the definition of sinc function:

$$\text{sinc}(u) = \frac{\sin(\pi u)}{\pi u}$$

As a result, we have:

$$\begin{aligned}\frac{\sin(\omega N/2)}{\sin(\omega/2)} &= \frac{\sin(\omega \cdot N\pi/(2\pi))}{\omega \cdot N\pi/(2\pi)} \cdot \frac{\omega \cdot N\pi}{2\pi} \cdot \frac{\omega\pi/(2\pi)}{\sin(\omega\pi/(2\pi))} \cdot \frac{2\pi}{\omega\pi} \\ &= N \cdot \frac{\text{sinc}(\omega N/(2\pi))}{\text{sinc}(\omega/(2\pi))}\end{aligned}$$

The key MATLAB code for $N = 10$ is

```
N=10;                   %N=10
w=0:0.01*pi:2*pi;       %successive frequency point
                        %separation is 0.01pi
dtft=N.*sinc(w.*N./2./pi)./(sinc(w./2./pi)).*exp(-
j.*w.*(N-1)./2);        %define DTFT function
subplot(2,1,1)
Mag=abs(dtft);          %compute magnitude
plot(w./pi,Mag);        %plot magnitude
subplot(2,1,2)
Pha=angle(dtft);        %compute phase
plot(w./pi,Pha);        %plot phase
```

Analogous to Example 4.4, there are 201 uniformly-spaced points to approximate the continuous functions $|X(e^{j\omega})|$ and $\angle(X(e^{j\omega}))$.

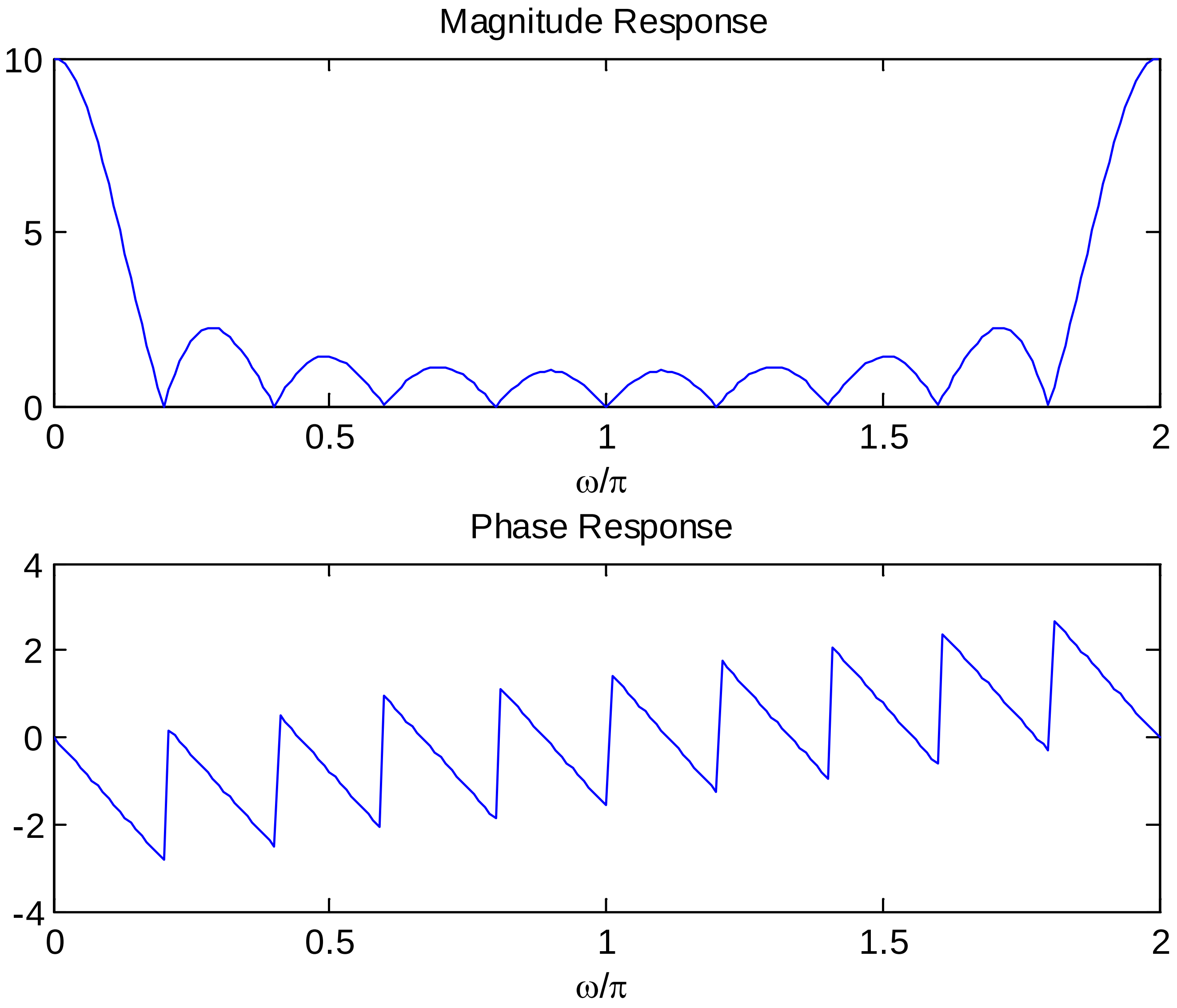


Fig.6.2: DTFT plots using abs and angle

Alternatively, we can use the command `freqz`:

$$X(z) = \frac{1 - z^{-N}}{1 - z^{-1}}$$

which is ratio of two polynomials in z^{-1}

The corresponding MATLAB code is:

```
N=10;                      %N=10
a=[1,-1];                  %vector for denominator
b=[1,zeros(1,N-1),-1];     %vector for numerator
freqz(b,a)                 %plot magnitude & phase (dB)
```

Note that it is also possible to use $X(z) = \sum_{n=0}^{N-1} z^{-n}$ and in this case we have `b=ones(N,1)` and `a=1`.

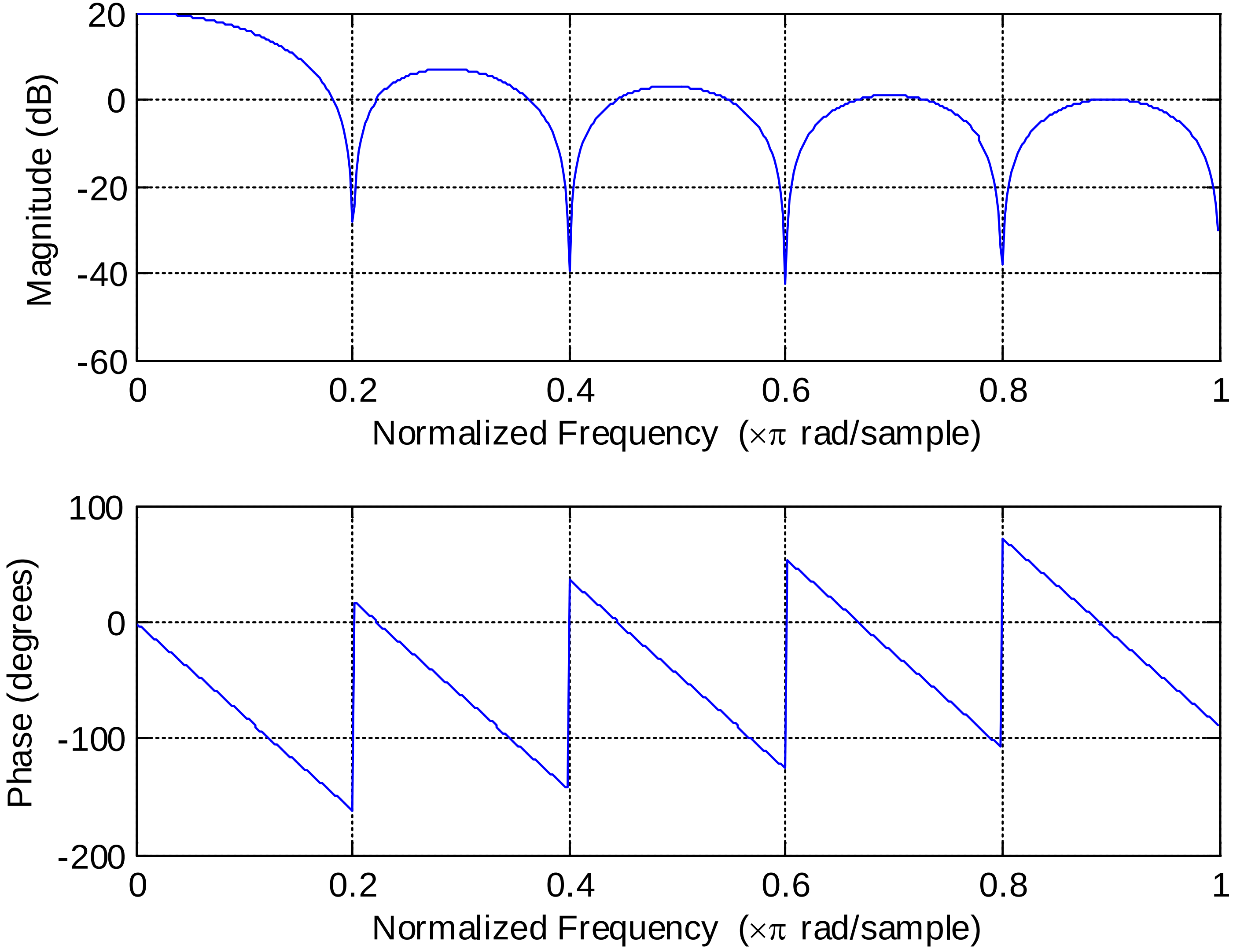


Fig.6.3: DTFT plots using `freqz`

The results in Figs. 6.2 and 6.3 are identical, although their presentations are different:

- $|X(e^{j\omega})| = 10$ at $\omega = 0$ in Fig. 6.2 while that of Fig. 6.3 is 20 dB. It is easy to verify that 10 corresponds to $20\log_{10} 10 = 20$ dB

- units of phase spectra in Figs. 6.2 and 6.3 are radian and degree, respectively. To make the phase values in both plots identical, we also need to take care of the 2π phase ambiguity.

The MATLAB programs for this example are provided as `ex6_2.m` and `ex6_2_2.m`.

Example 6.3

Find the inverse DTFT of $X(e^{j\omega})$ which is a rectangular pulse within $\omega \in [-\pi, \pi]$:

$$X(e^{j\omega}) = \begin{cases} 1, & -w_0 < \omega < w_0 \\ 0, & \text{otherwise} \end{cases}$$

where $0 < w_0 < \pi$.

Using (6.2), we get:

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega = \frac{1}{2\pi} \int_{-w_0}^{w_0} e^{j\omega n} d\omega = \frac{\sin(w_0 n)}{\pi n} = \frac{w_0}{\pi} \mathrm{sinc}\left(\frac{w_0 n}{\pi}\right)$$

That is, $x[n]$ is an infinite-duration sequence whose values are drawn from a scaled sinc function.

Example 6.4

Determine the inverse DTFT of $X(e^{j\omega})$ which has the form of:

$$X(e^{j\omega}) = \frac{e^{-j2\omega}}{1+0.7e^{-j\omega}}$$

With the use of $z = e^{j\omega}$, the corresponding z transform is

$$X(z) = \frac{z^{-2}}{1+0.7z^{-1}}, \quad |z| > 0.7$$

Note that ROC should include the unit circle as DTFT exists

Employing the time shifting property, we get

$$x[n] = (-0.7)^{n-2}u[n-2]$$

Properties of DTFT

Since DTFT is closely related to z transform, its properties follow those of z transform. Note that ROC is not involved because it should include unit circle in order for DTFT exists

1. Linearity

If $(x_1[n], X_1(e^{j\omega}))$ and $(x_2[n], X_2(e^{j\omega}))$ are two DTFT pairs, then:

$$ax_1[n] + bx_2[n] \leftrightarrow aX_1(e^{j\omega}) + bX_2(e^{j\omega}) \qquad (6.9)$$

2. Time Shifting

A shift of n_0 in $x[n]$ causes a multiplication of $e^{-j\omega n_0}$ in $X(e^{j\omega})$:

$$x[n-n_0] \leftrightarrow e^{-j\omega n_0}X(e^{j\omega}) \qquad (6.10)$$

3. Multiplication by an Exponential Sequence

Multiplying $x[n]$ by $e^{j\omega_0 n}$ in time domain corresponds to a shift of ω_0 in the frequency domain:

$$e^{j\omega_0 n}x[n] \leftrightarrow X(e^{j(\omega-\omega_0)}) \tag{6.11}$$

which agrees with (5.33) by putting $z = e^{j\omega}$ and $z_0 = e^{j\omega_0}$

4. Differentiation

Differentiating $X(e^{j\omega})$ with respect to ω corresponds to multiplying $x[n]$ by n:

$$nx[n] \leftrightarrow j\frac{dX(e^{j\omega})}{d\omega} \tag{6.12}$$

Note the RHS is obtained from (5.34) by putting $z = e^{j\omega}$:

$$-e^{j\omega}\frac{dX(e^{j\omega})}{de^{j\omega}} = -e^{j\omega}\frac{dX(e^{j\omega})}{d\omega}\cdot\left(\frac{de^{j\omega}}{d\omega}\right)^{-1} = j\frac{dX(e^{j\omega})}{d\omega} \qquad (6.13)$$

5. Conjugation

The DTFT pair for $x^*[n]$ is given as:

$$x^*[n] \leftrightarrow X^*(e^{-j\omega}) \qquad (6.14)$$

6. Time Reversal

The DTFT pair for $x[-n]$ is given as:

$$x[-n] \leftrightarrow X(e^{-j\omega}) \qquad (6.15)$$

7. Convolution

If $(x_1[n], X_1(e^{j\omega}))$ and $(x_2[n], X_2(e^{j\omega}))$ are two DTFT pairs, then:

$$x_1[n] \otimes x_2[n] \leftrightarrow X_1(e^{j\omega})X_2(e^{j\omega}) \qquad (6.16)$$

In particular, for a LTI system with input $x[n]$, output $y[n]$ and impulse response $h[n]$, we have:

$$y[n] = x[n] \otimes h[n] \leftrightarrow Y(e^{j\omega}) = X(e^{j\omega})H(e^{j\omega}) \qquad (6.17)$$

which is analogous to (2.24) for continuous-time LTI systems

8. Multiplication

Multiplication in the time domain corresponds to convolution in the frequency domain:

$$x_1[n] \cdot x_2[n] \leftrightarrow X_1(e^{j\omega}) \tilde{\otimes} X_2(e^{j\omega}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X_1(e^{j\tau}) X_2(e^{j(\omega-\tau)}) d\tau \quad (6.18)$$

where $\tilde{\otimes}$ denotes convolution within one period

9. Parseval's Relation

The Parseval's relation addresses the energy of a sequence:

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| X(e^{j\omega}) \right|^2 d\omega \quad (6.19)$$

With the use of (6.2), the proof is:

$$\begin{aligned}
\sum_{n=-\infty}^{\infty} |x[n]|^2 &= \sum_{n=-\infty}^{\infty} x[n]x^*[n] \\
&= \sum_{n=-\infty}^{\infty} x[n] \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega \right)^* \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} X^*(e^{j\omega}) \left(\sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n} \right) d\omega \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| X(e^{j\omega}) \right|^2 d\omega \qquad (6.20)
\end{aligned}$$

Chapter 7: Discrete Fourier Series & Discrete Fourier Transform

Chapter Intended Learning Outcomes

(i) Understanding the relationships between the z transform, discrete-time Fourier transform (DTFT), discrete Fourier series (DFS) and discrete Fourier transform (DFT)

(ii) Understanding the characteristics and properties of DFS and DFT

(iii) Ability to perform discrete-time signal conversion between the time and frequency domains using DFS and DFT and their inverse transforms

Discrete Fourier Series

DTFT may not be practical for analyzing $x[n]$ because $X(e^{j\omega})$ is a function of the continuous frequency variable ω and we cannot use a digital computer to calculate a continuum of functional values

DFS is a frequency analysis tool for periodic infinite-duration discrete-time signals which is practical because it is discrete in frequency

The DFS is derived from the Fourier series as follows.

Let $\tilde{x}[n]$ be a periodic sequence with fundamental period N where N is a positive integer. Analogous to (2.2), we have:

$$\tilde{x}[n] = \tilde{x}[n + rN] \qquad (7.1)$$

for any integer value of r.

Let $x(t)$ be the continuous-time counterpart of $\tilde{x}[n]$. According to Fourier series expansion, $x(t)$ is:

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\Omega_0 t} = \sum_{k=-\infty}^{\infty} a_k e^{\frac{j2\pi kt}{T_p}} \qquad (7.2)$$

which has frequency components at $\Omega = 0, \pm\Omega_0, \pm 2\Omega_0, \cdots$. Substituting $x(t) = \tilde{x}[n]$, $T_p = N$ and $t = n$:

$$\tilde{x}[n] = \sum_{k=-\infty}^{\infty} a_k e^{\frac{j2\pi kn}{N}} \qquad (7.3)$$

Note that (7.3) is valid for discrete-time signals as only the sample points of $x(t)$ are considered.

It is seen that $\tilde{x}[n]$ has frequency components at $\omega = 0, \pm 2\pi/N, \pm(2\pi/N)(2), \cdots$, and the respective complex exponentials are $e^{j(2\pi/N(0))}, e^{\pm j(2\pi/N(1))}, e^{\pm j(2\pi/N(2))}, \cdots$.

Nevertheless, there are only N distinct frequencies in $\tilde{x}[n]$ due to the periodicity of $e^{j2\pi k/N}$.

Without loss of generality, we select the following N distinct complex exponentials, $e^{j(2\pi/N(0))}, e^{j(2\pi/N(1))}, \cdots, e^{j(2\pi/N(N-1))}$, and thus the infinite summation in (7.3) is reduced to:

$$\tilde{x}[n] = \sum_{k=0}^{N-1} a_k e^{\frac{j2\pi kn}{N}} \qquad (7.4)$$

Defining $\tilde{X}[k] = N a_k$, $k = 0, 1, \cdots, N-1$, as the DFS coefficients, the inverse DFS formula is given as:

$$\tilde{x}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] e^{\frac{j2\pi kn}{N}} \qquad (7.5)$$

The formula for converting $\tilde{x}[n]$ to $\tilde{X}[k]$ is derived as follows.

Multiplying both sides of (7.5) by $e^{-j(2\pi/N)rn}$ and summing from $n=0$ to $n=N-1$:

$$\begin{aligned}
\sum_{n=0}^{N-1} \tilde{x}[n] e^{\frac{-j2\pi rn}{N}} &= \sum_{n=0}^{N-1} \left(\frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] e^{\frac{j2\pi kn}{N}} \right) e^{\frac{-j2\pi rn}{N}} \\
&= \sum_{n=0}^{N-1} \frac{1}{N} \left(\sum_{k=0}^{N-1} \tilde{X}[k] e^{\frac{j2\pi(k-r)n}{N}} \right) \\
&= \sum_{k=0}^{N-1} \tilde{X}[k] \left[\frac{1}{N} \sum_{n=0}^{N-1} e^{\frac{j2\pi(k-r)n}{N}} \right] \qquad (7.6)
\end{aligned}$$

Using the orthogonality identity of complex exponentials:

$$\frac{1}{N} \sum_{n=0}^{N-1} e^{\frac{j2\pi(k-r)n}{N}} = \begin{cases} 1, \ k-r=mN, \quad m \quad \text{is} \quad \text{an} \quad \text{integer} \\ 0, \ \text{otherwise} \end{cases} \qquad (7.7)$$

(7.6) is reduced to

$$\sum_{n=0}^{N-1} \tilde{x}[n]e^{-\frac{j2\pi rn}{N}} = \tilde{X}[r] \tag{7.8}$$

which is also periodic with period N.

Let

$$W_N = e^{-\frac{j2\pi}{N}} \tag{7.9}$$

The DFS analysis and synthesis pair can be written as:

$$\tilde{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n]W_N^{kn} \tag{7.10}$$

and

$$\tilde{x}[n] = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}[k]W_N^{-kn} \tag{7.11}$$

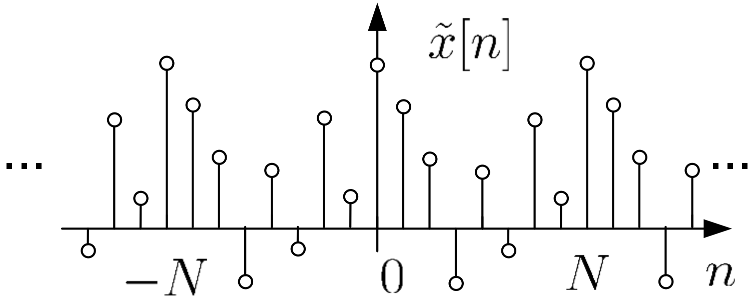
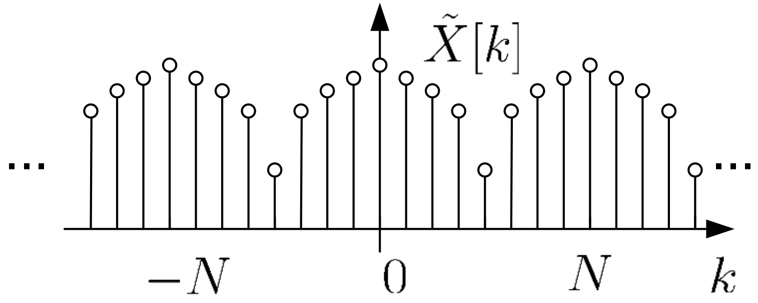
time domain	frequency domain
	
$\tilde{X}[k]=\sum_{n=0}^{N-1}\tilde{x}[n]W_N^{kn} \Longrightarrow$	
	$\Longleftarrow \tilde{x}[n]=\frac{1}{N}\sum_{k=0}^{N-1}\tilde{X}[k]W_N^{-kn}$
discrete and periodic	discrete and periodic

Fig.7.1: Illustration of DFS

Example 7.1

Find the DFS coefficients of the periodic sequence $\tilde{x}[n]$ with a period of $N=5$. Plot the magnitudes and phases of $\tilde{X}[k]$. Within one period, $\tilde{x}[n]$ has the form of:

$$\tilde{x}[n] = \begin{cases} 1, & n = 0,1,2 \\ 0, & n = 3,4 \end{cases}$$

Using (7.10), we have

$$\begin{aligned}
\tilde{X}[k] &= \sum_{n=0}^{N-1} \tilde{x}[n] W_N^{kn} \\
&= W_5^0 + W_5^k + W_5^{2k} \\
&= 1 + e^{-\frac{j2\pi k}{5}} + e^{-\frac{j4\pi k}{5}} \\
&= e^{-\frac{j2\pi k}{5}} \left(e^{\frac{j2\pi k}{5}} + 1 + e^{-\frac{j2\pi k}{5}} \right) \\
&= e^{-\frac{j2\pi k}{5}} \left[1 + 2\cos\left(\frac{2\pi k}{5} \right) \right]
\end{aligned}$$

Similar to Example 6.2, we get:

$$|\tilde{X}[k]| = \left|1 + 2\cos\left(\frac{2\pi k}{5}\right)\right|$$

and

$$\angle(\tilde{X}[k]) = -\frac{2\pi k}{5} + \angle\left(1 + 2\cos\left(\frac{2\pi k}{5}\right)\right)$$

The key MATLAB code for plotting DFS coefficients is

```
N=5;
x=[1 1 1 0 0];
k=-N:2*N;                  %plot for 3 periods
Xm=abs(1+2.*cos(2*pi.*k/N));%magnitude computation
Xa=angle(exp(-2*j*pi.*k/5).*(1+2.*cos(2*pi.*k/N)));
                           %phase computation
```

The MATLAB program is provided as `ex7_1.m`.

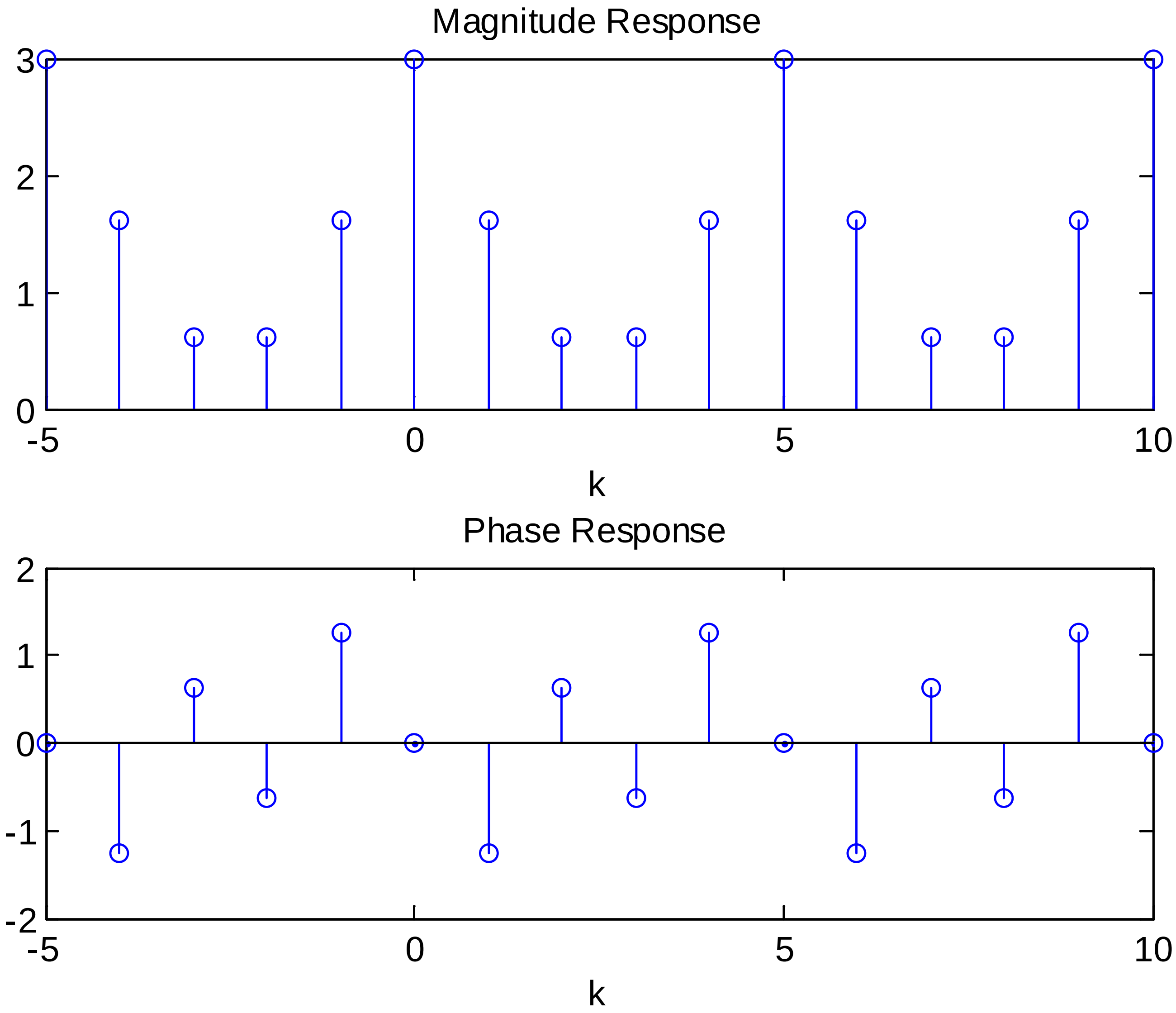


Fig.7.2: DFS plots

Relationship with DTFT

Let $x[n]$ be a finite-duration sequence which is extracted from a periodic sequence $\tilde{x}[n]$ of period N:

$$x[n] = \begin{cases} \tilde{x}[n], & 0 \leq n \leq N-1 \\ 0, & \text{otherwise} \end{cases} \tag{7.12}$$

Recall (6.1), the DTFT of $x[n]$ is:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \tag{7.13}$$

With the use of (7.12), (7.13) becomes

$$X(e^{j\omega}) = \sum_{n=0}^{N-1} x[n]e^{-j\omega n} = \sum_{n=0}^{N-1} \tilde{x}[n]e^{-j\omega n} \tag{7.14}$$

Comparing the DFS and DTFT in (7.8) and (7.14), we have:

$$\tilde{X}[k] = X(e^{j\omega})|_{\omega=\frac{2\pi k}{N}} \tag{7.15}$$

That is, $\tilde{X}[k]$ is equal to $X(e^{j\omega})$ sampled at N distinct frequencies between $\omega \in [0, 2\pi]$ with a uniform frequency spacing of $2\pi/N$.

Samples of $X(e^{j\omega})$ or DTFT of a finite-duration sequence $x[n]$ can be computed using the DFS of an infinite-duration periodic sequence $\tilde{x}[n]$, which is a periodic extension of $x[n]$.

Relationship with z Transform

$X(e^{j\omega})$ is also related to z transform of $x[n]$ according to (5.8):

$$X(e^{j\omega}) = X(z)|_{z=e^{j\omega}} \qquad (7.16)$$

Combining (7.15) and (7.16), $\tilde{X}[k]$ is related to $X(z)$ as:

$$\tilde{X}[k] = X(z)|_{z=e^{\frac{j2\pi k}{N}}} = X(e^{\frac{j2\pi k}{N}}) \qquad (7.17)$$

That is, $\tilde{X}[k]$ is equal to $X(z)$ evaluated at N equally-spaced points on the unit circle, namely, $1, e^{j2\pi/N}, \cdots, e^{j2(N-1)\pi/N}$.

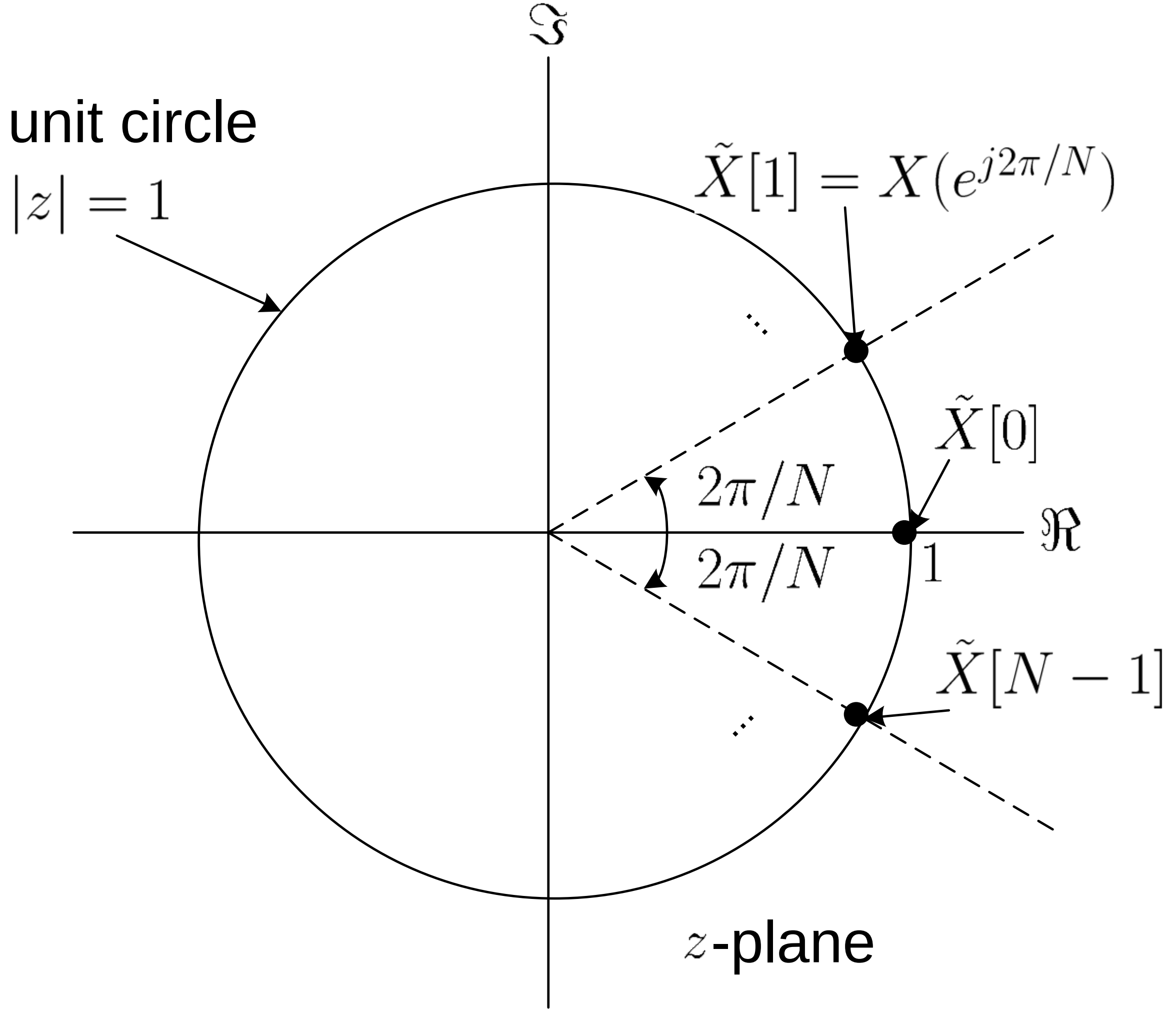


Fig.7.3: Relationship between $\tilde{X}[k]$, $X(e^{j\omega})$ and $X(z)$

Example 7.2

Determine the DTFT of a finite-duration sequence $x[n]$:

$$x[n] = \begin{cases} 1, & n = 0,1,2 \\ 0, & \text{otherwise} \end{cases}$$

Then compare the results with those in Example 7.1.

Using (6.1), the DTFT of $x[n]$ is computed as:

$$\begin{aligned} X(e^{j\omega}) &= \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \\ &= 1 + e^{-j\omega} + e^{-j2\omega} \\ &= e^{-j\omega}\left(e^{j\omega} + 1 + e^{-j\omega}\right) \\ &= e^{-j\omega}\left[1 + 2\cos(\omega)\right] \end{aligned}$$

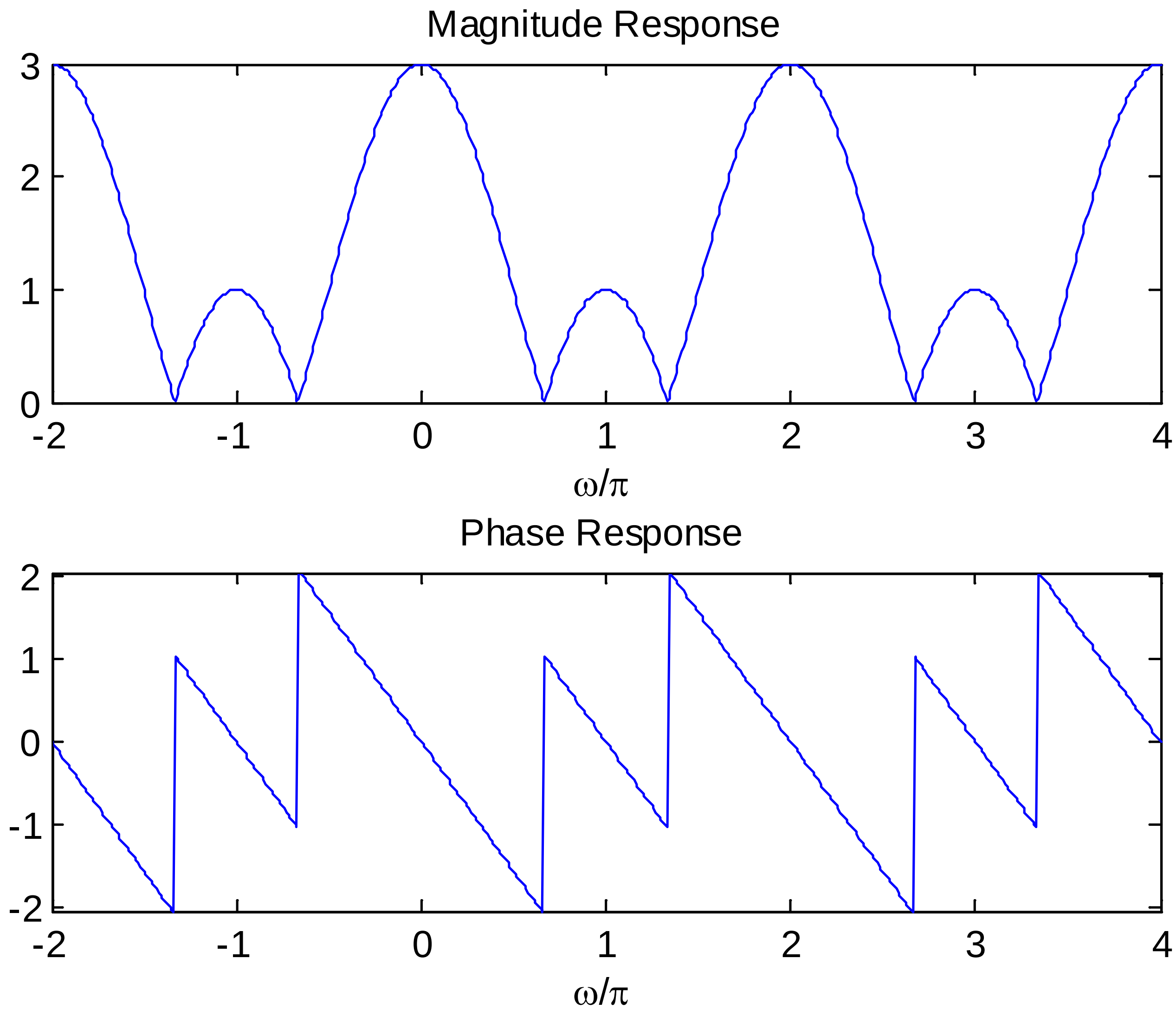


Fig.7.4: DTFT plots

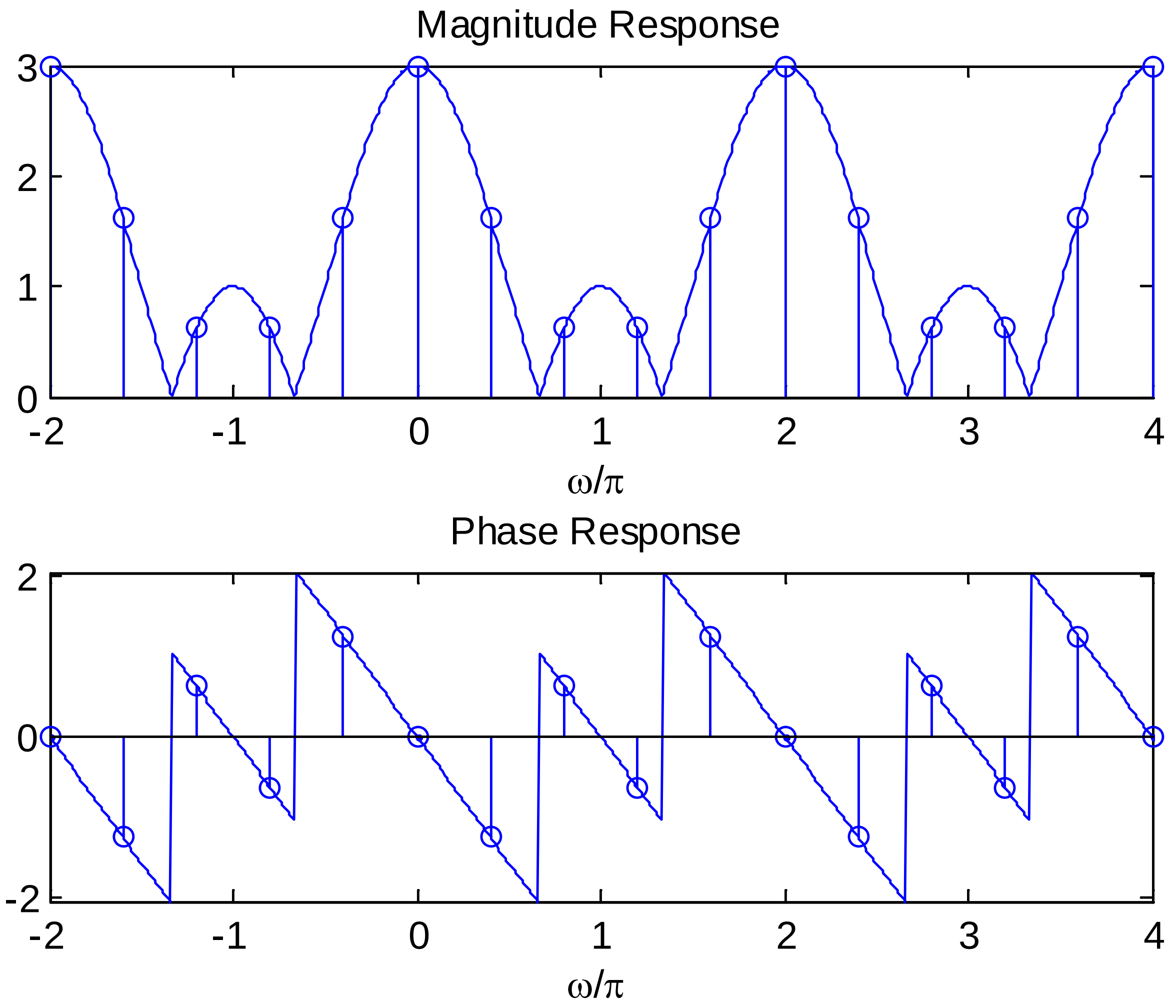


Fig.7.5: DFS and DTFT plots with $N = 5$

Suppose $\tilde{x}[n]$ in Example 7.1 is modified as:

$$\tilde{x}[n] = \begin{cases} 1, & n = 0, 1, 2 \\ 0, & n = 3, 4, \cdots, 9 \end{cases}$$

Via appending 5 zeros in each period, now we have $N = 10$.

What is the period of the DFS?

What is its relationship with that of Example 7.2?

How about if infinite zeros are appended?

The MATLAB programs are provided as `ex7_2.m`, `ex7_2_2.m` and `ex7_2_3.m`.

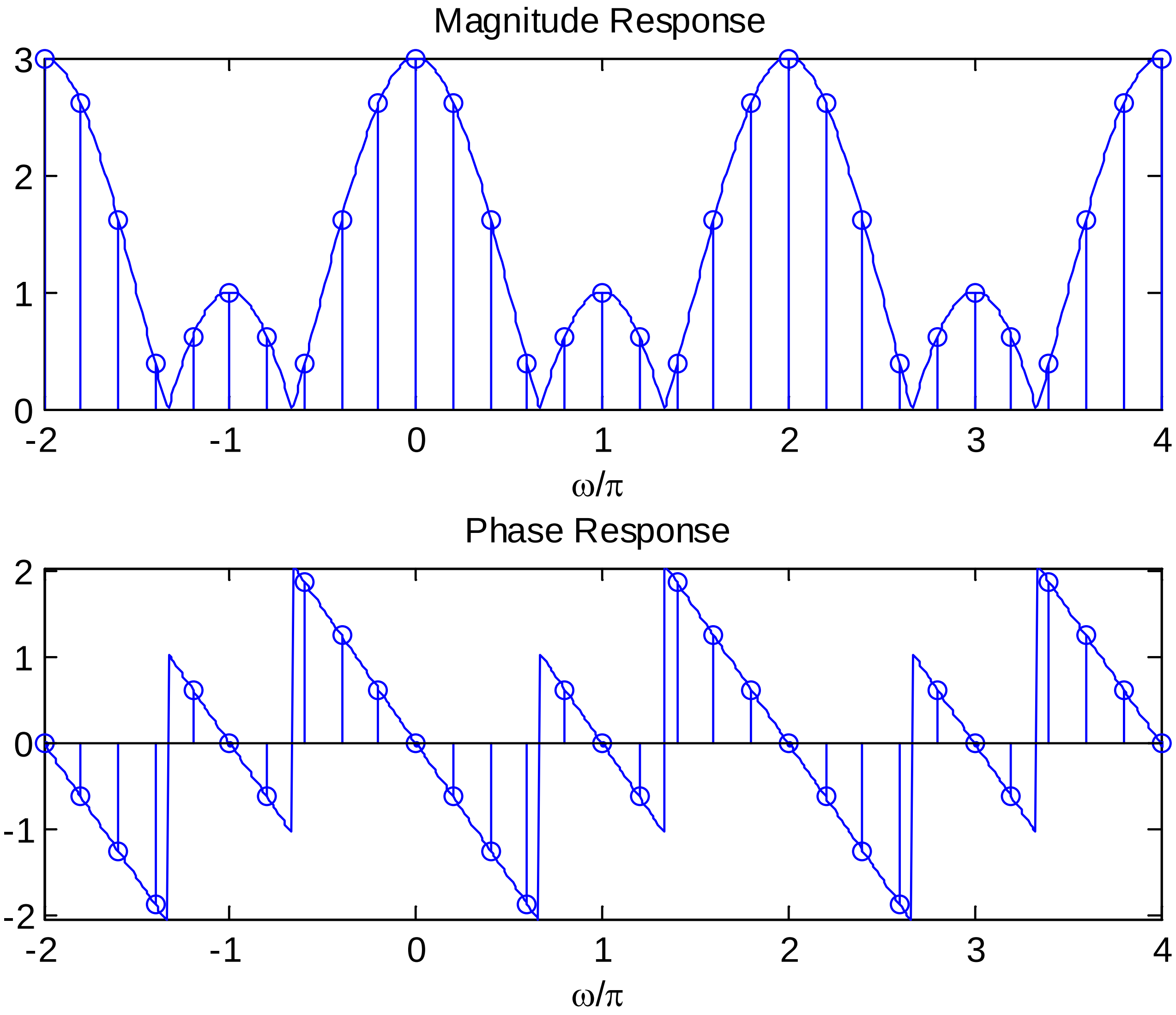


Fig.7.6: DFS and DTFT plots with $N = 10$

Properties of DFS

1. Periodicity

If $\tilde{x}[n]$ is a periodic sequence with period N, its DFS $\tilde{X}[k]$ is also periodic with period N:

$$\tilde{x}[n] = \tilde{x}[n + rN] \leftrightarrow \tilde{X}[k] = \tilde{X}[k + rN] \tag{7.18}$$

where r is any integer. The proof is obtained with the use of (7.10) and $W_N^{rN} = e^{-j2\pi r} = 1$ as follows:

$$\begin{aligned}\tilde{X}[k + rN] &= \sum_{n=0}^{N-1} \tilde{x}[n] W_N^{(k+rN)n} = \sum_{n=0}^{N-1} \tilde{x}[n] W_N^{nk} W_N^{n(rN)} \\ &= \sum_{n=0}^{N-1} \tilde{x}[n] W_N^{nk} = \tilde{X}[k]\end{aligned} \tag{7.19}$$

2. Linearity

Let $(\tilde{x}_1[n], \tilde{X}_1[k])$ and $(\tilde{x}_2[n], \tilde{X}_2[k])$ be two DFS pairs with the same period of N. We have:

$$a\tilde{x}_1[n] + b\tilde{x}_2[n] \leftrightarrow a\tilde{X}_1[k] + b\tilde{X}_2[k] \tag{7.20}$$

3. Shift of Sequence

If $\tilde{x}[n] \leftrightarrow \tilde{X}[k]$, then

$$\tilde{x}[n-m] \leftrightarrow W_N^{km}\tilde{X}[k] \tag{7.21}$$

and

$$W_N^{-nl}\tilde{x}[n] \leftrightarrow \tilde{X}[k-l] \tag{7.22}$$

where N is the period while m and l are any integers. Note that (7.21) follows (6.10) by putting $\omega = 2\pi k/N$ and (7.22) follows (6.11) via the substitution of $\omega_0 = 2\pi l/N$.

4. Duality

If $\tilde{x}[n] \leftrightarrow \tilde{X}[k]$, then

$$\tilde{X}[n] \leftrightarrow N\tilde{x}[-k] \tag{7.23}$$

5. Symmetry

If $\tilde{x}[n] \leftrightarrow \tilde{X}[k]$, then

$$\tilde{x}^*[n] \leftrightarrow \tilde{X}^*[-k] \tag{7.24}$$

and

$$\tilde{x}^*[-n] \leftrightarrow \tilde{X}^*[k] \tag{7.25}$$

Note that (7.24) corresponds to the DTFT conjugation property in (6.14) while (7.25) is similar to the time reversal property in (6.15).

6. Periodic Convolution

Let $(\tilde{x}_1[n], \tilde{X}_1[k])$ and $(\tilde{x}_2[n], \tilde{X}_2[k])$ be two DFS pairs with the same period of N. We have

$$\tilde{x}_1[n]\tilde{\otimes}\tilde{x}_2[n] = \sum_{m=0}^{N-1} \tilde{x}_1[m]\tilde{x}_2[n-m] \leftrightarrow \tilde{X}_1[k]\tilde{X}_2[k] \tag{7.26}$$

Analogous to (6.18), $\tilde{\otimes}$ denotes discrete-time convolution within one period.

With the use of (7.11) and (7.21), the proof is given as follows:

$$
\begin{aligned}
\sum_{n=0}^{N-1}\left[\sum_{m=0}^{N-1}\tilde{x}_1[m]\tilde{x}_2[n-m]\right]W_N^{nk} &= \sum_{m=0}^{N-1}\tilde{x}_1[m]\left[\sum_{n=0}^{N-1}\tilde{x}_2[n-m]W_N^{nk}\right] \\
&= \sum_{m=0}^{N-1}\tilde{x}_1[m]\tilde{X}_2[k]W_N^{mk} \\
&= \tilde{X}_2[k]\left[\sum_{m=0}^{N-1}\tilde{x}_1[m]W_N^{mk}\right] \\
&= \tilde{X}_1[k]\tilde{X}_2[k] \qquad (7.27)
\end{aligned}
$$

To compute $\tilde{x}[n]\tilde{\otimes}\tilde{y}[n]$ where both $\tilde{x}[n]$ and $\tilde{y}[n]$ are of period N, we indeed only need the samples with $n = 0, 1, \cdots, N-1$.

Let $\tilde{z}[n] = \tilde{x}[n]\tilde{\otimes}\tilde{y}[n]$. Expanding (7.26), we have:

$$\tilde{z}[n] = \tilde{x}[0]\tilde{y}[n] + \cdots + \tilde{x}[N-2]\tilde{y}[n-(N-2)] + \tilde{x}[N-1]\tilde{y}[n-(N-1)] \quad (7.28)$$

For $n = 0$:

$$\begin{aligned} \tilde{z}[0] &= \tilde{x}[0]\tilde{y}[0] + \cdots + \tilde{x}[N-2]\tilde{y}[0-(N-2)] + \tilde{x}[N-1]\tilde{y}[0-(N-1)] \\ &= \tilde{x}[0]\tilde{y}[0] + \cdots + \tilde{x}[N-2]\tilde{y}[0-(N-2)+N] + \tilde{x}[N-1]\tilde{y}[0-(N-1)+N] \\ &= \tilde{x}[0]\tilde{y}[0] + \cdots + \tilde{x}[N-2]\tilde{y}[2] + \tilde{x}[N-1]\tilde{y}[1] \end{aligned} \quad (7.29)$$

For $n = 1$:

$$\begin{aligned} \tilde{z}[1] &= \tilde{x}[0]\tilde{y}[1] + \cdots + \tilde{x}[N-2]\tilde{y}[1-(N-2)] + \tilde{x}[N-1]\tilde{y}[1-(N-1)] \\ &= \tilde{x}[0]\tilde{y}[1] + \cdots + \tilde{x}[N-2]\tilde{y}[1-(N-2)+N] + \tilde{x}[N-1]\tilde{y}[1-(N-1)+N] \\ &= \tilde{x}[0]\tilde{y}[1] + \cdots + \tilde{x}[N-2]\tilde{y}[3] + \tilde{x}[N-1]\tilde{y}[2] \end{aligned} \quad (7.30)$$

A period of $\tilde{z}[n]$ can be computed in matrix form as:

$$\begin{bmatrix} \tilde{z}[0] \\ \tilde{z}[1] \\ \vdots \\ \tilde{z}[N-2] \\ \tilde{z}[N-1] \end{bmatrix} = \begin{bmatrix} \tilde{y}[0] & \tilde{y}[N-1] & \cdots & \tilde{y}[2] & \tilde{y}[1] \\ \tilde{y}[1] & \tilde{y}[0] & \cdots & \tilde{y}[3] & \tilde{y}[2] \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \tilde{y}[N-2] & \tilde{y}[N-3] & \cdots & \tilde{y}[0] & \tilde{y}[N-1] \\ \tilde{y}[N-1] & \tilde{y}[N-2] & \cdots & \tilde{y}[1] & \tilde{y}[0] \end{bmatrix} \begin{bmatrix} \tilde{x}[0] \\ \tilde{x}[1] \\ \vdots \\ \tilde{x}[N-2] \\ \tilde{x}[N-1] \end{bmatrix} \quad (7.31)$$

Example 7.3

Given two periodic sequences $\tilde{x}[n]$ and $\tilde{y}[n]$ with period 4:

$$\left[\, \tilde{x}[0] \; \tilde{x}[1] \; \tilde{x}[2] \; \tilde{x}[3] \,\right] = \left[\, 4 \;\; -3 \; 2 \;\; -1 \,\right]$$

and

$$\left[\, \tilde{y}[0] \; \tilde{y}[1] \; \tilde{y}[2] \; \tilde{y}[3] \,\right] = \left[\, 1 \; 2 \; 3 \; 4 \,\right]$$

Compute $\tilde{z}[n] = \tilde{x}[n] \tilde{\otimes} \tilde{y}[n]$.

Using (7.31), $\tilde{z}[n]$ is computed as:

$$\begin{bmatrix} \tilde{z}[0] \\ \tilde{z}[1] \\ \tilde{z}[2] \\ \tilde{z}[3] \end{bmatrix} = \begin{bmatrix} \tilde{y}[0] & \tilde{y}[3] & \tilde{y}[2] & \tilde{y}[1] \\ \tilde{y}[1] & \tilde{y}[0] & \tilde{y}[3] & \tilde{y}[2] \\ \tilde{y}[2] & \tilde{y}[1] & \tilde{y}[0] & \tilde{y}[3] \\ \tilde{y}[3] & \tilde{y}[2] & \tilde{y}[1] & \tilde{y}[0] \end{bmatrix} \begin{bmatrix} \tilde{x}[0] \\ \tilde{x}[1] \\ \tilde{x}[2] \\ \tilde{x}[3] \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 4 & 3 & 2 \\ 2 & 1 & 4 & 3 \\ 3 & 2 & 1 & 4 \\ 4 & 3 & 2 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ -3 \\ 2 \\ -1 \end{bmatrix} = \begin{bmatrix} -4 \\ 10 \\ 4 \\ 10 \end{bmatrix}$$

The square matrix can be determined using the MATLAB command `toeplitz([1,2,3,4],[1,4,3,2])`. That is, we only need to know its first row and first column.

Periodic convolution can be utilized to compute convolution of finite-duration sequences in (3.17) as follows.

Let $x[n]$ and $y[n]$ be finite-duration sequences with lengths M and N, respectively, and $z[n] = x[n] \otimes y[n]$ which has a length of $(M + N - 1)$

We append $(N - 1)$ and $(M - 1)$ zeros at the ends of $x[n]$ and $y[n]$ for constructing periodic $\tilde{x}[n]$ and $\tilde{y}[n]$ where both are of period $(M + N - 1)$

$z[n]$ is then obtained from one period of $\tilde{x}[n]\tilde{\otimes}\tilde{y}[n]$.

<u>Example 7.4</u>

Compute the convolution of $x[n]$ and $y[n]$ with the use of periodic convolution. The lengths of $x[n]$ and $y[n]$ are 2 and 3 with $x[0] = 2$, $x[1] = 3$, $y[0] = 1$, $y[1] = -4$ and $y[2] = 5$.

The length of $x[n] \otimes y[n]$ is 4. As a result, we append two zeros and one zero in $x[n]$ and $y[n]$, respectively. According to (7.31), the MATLAB code is:

```
toeplitz([1,-4,5,0],[1,0,5,-4])*[2;3;0;0]
```

which gives

```
2     -5     -2     15
```

Note that the command conv([2,3],[1,-4,5]) also produces the same result.

Discrete Fourier Transform

DFT is used for analyzing discrete-time finite-duration signals in the frequency domain

Let $x[n]$ be a finite-duration sequence of length N such that $x[n]=0$ outside $0 \le n \le N-1$. The DFT pair of $x[n]$ is:

$$X[k] = \begin{cases} \sum_{n=0}^{N-1} x[n] W_N^{kn}, & 0 \le k \le N-1 \\ 0, & \text{otherwise} \end{cases} \tag{7.32}$$

and

$$x[n] = \begin{cases} \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-kn}, & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases} \tag{7.33}$$

If we extend $x[n]$ to a periodic sequence $\tilde{x}[n]$ with period N, the DFS pair for $\tilde{x}[n]$ is given by (7.10)-(7.11). Comparing (7.32) and (7.10), $X[k] = \tilde{X}[k]$ for $0 \leq k \leq N-1$. As a result, DFT and DFS are equivalent within the interval of $[0, N-1]$

Example 7.5

Find the DFT coefficients of a finite-duration sequence $x[n]$ which has the form of

$$x[n] = \begin{cases} 1, & n = 0,1,2 \\ 0, & \text{otherwise} \end{cases}$$

Using (7.32) and Example 7.1 with $N=3$, we have:

$$\begin{aligned} X[k] &= \sum_{n=0}^{2} x[n] W_N^{kn} = W_3^0 + W_3^k + W_3^{2k} \\ &= e^{-\frac{j2\pi k}{3}} \left[1 + 2\cos\left(\frac{2\pi k}{3}\right)\right] = \begin{cases} 3, & k = 0 \\ 0, & k = 1,2 \end{cases} \end{aligned}$$

Together with $X[k]$ whose index is outside the interval of $0 \leq k \leq 2$, we finally have:

$$X[k] = \begin{cases} 3, & k = 0 \\ 0, & \text{otherwise} \end{cases}$$

If the length of $x[n]$ is considered as $N = 5$ such that $x[3] = x[4] = 0$, then we obtain:

$$\begin{aligned} X[k] &= \sum_{n=0}^{N-1} x[n] W_N^{kn} = W_5^0 + W_5^k + W_5^{2k} \\ &= \begin{cases} e^{-\frac{j2\pi k}{5}} \left[1 + 2\cos\left(\frac{2\pi k}{5}\right) \right], & k = 0, 1, \cdots, 4 \\ 0, & \text{otherwise} \end{cases} \end{aligned}$$

The MATLAB command for DFT computation is `fft`. The MATLAB code to produce magnitudes and phases of $X[k]$ is:

```
N=5;
x=[1 1 1 0 0];                  %append 2 zeros
subplot(2,1,1);
stem([0:N-1],abs(fft(x))); %plot magnitude response
title('Magnitude Response');
subplot(2,1,2);
stem([0:N-1],angle(fft(x)));%plot phase response
title('Phase Response');
```

According to Example 7.2 and the relationship between DFT and DFS, the DFT will approach the DTFT when we append infinite zeros at the end of $x[n]$

The MATLAB program is provided as `ex7_5.m`.

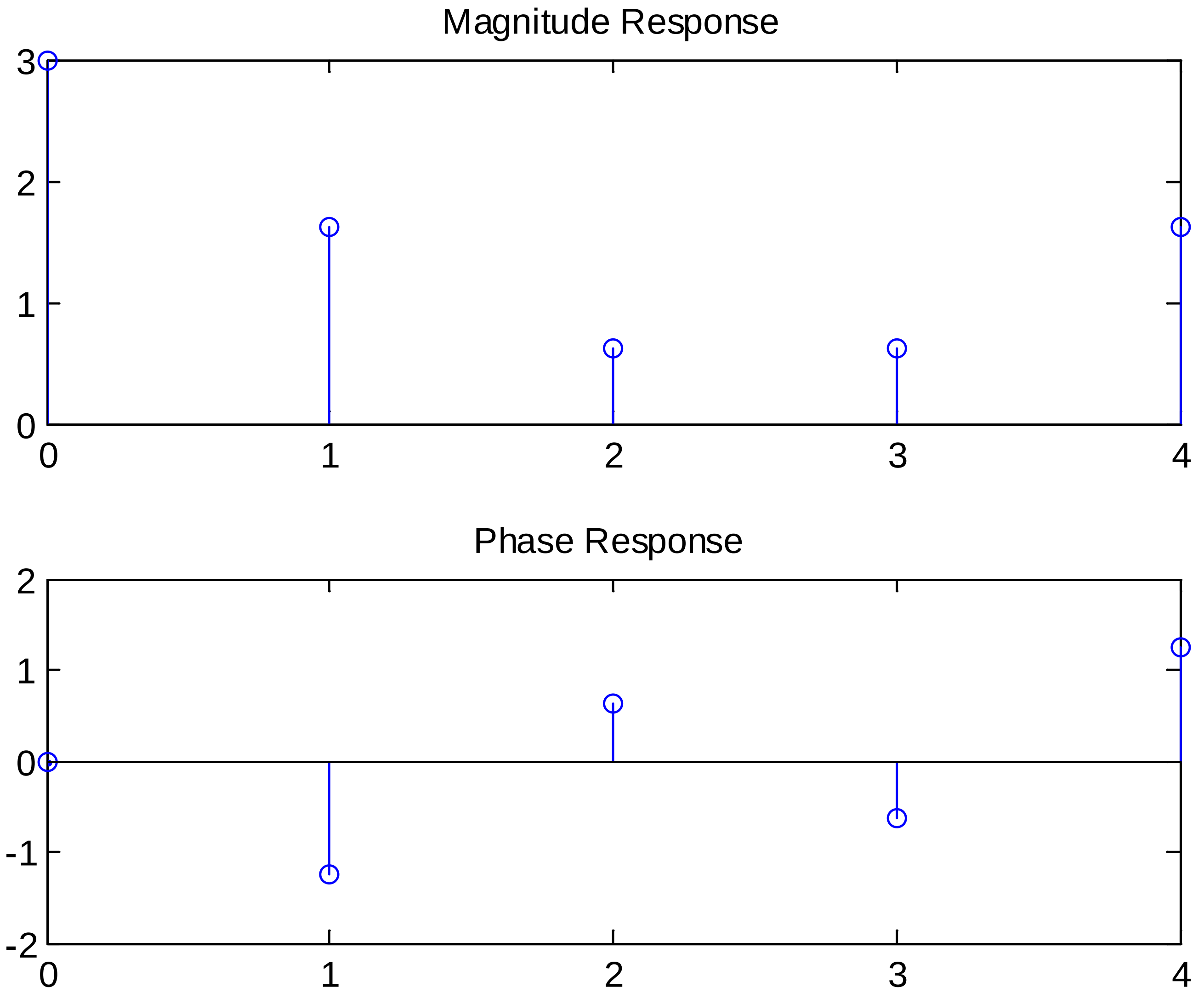


Fig.7.7: DFT plots with $N = 5$

Example 7.6

Given a discrete-time finite-duration sinusoid:

$$x[n] = 2\cos(0.7\pi n + 1), \quad n = 0, 1, \cdots, 20$$

Estimate the tone frequency using DFT.

Consider the continuous-time case first. According to (2.16), Fourier transform pair for a complex tone of frequency Ω_0 is:

$$e^{j\Omega_0 t} \leftrightarrow 2\pi\delta(\Omega - \Omega_0)$$

That is, Ω_0 can be found by locating the peak of the Fourier transform. Moreover, a real-valued tone $\cos(\Omega_0 t)$ is:

$$\cos(\Omega_0 t) = \frac{e^{j\Omega_0 t} + e^{-j\Omega_0 t}}{2}$$

From the Fourier transform of $\cos(\Omega_0 t)$, Ω_0 and $-\Omega_0$ are located from the two impulses.

Analogously, there will be two peaks which correspond to frequencies 0.7π and -0.7π in the DFT for $x[n]$.

The MATLAB code is

```
N=21;             %number of samples is 21
A=2;              %tone amplitude is 2
w=0.7*pi;         %frequency is 0.7*pi
p=1;              %phase is 1
n=0:N-1;          %define a vector of size N
x=A*cos(w*n+p);   %generate tone
X=fft(x);         %compute DFT
subplot(2,1,1);
stem(n,abs(X));   %plot magnitude response
subplot(2,1,2);
stem(n,angle(X)); %plot phase response
```

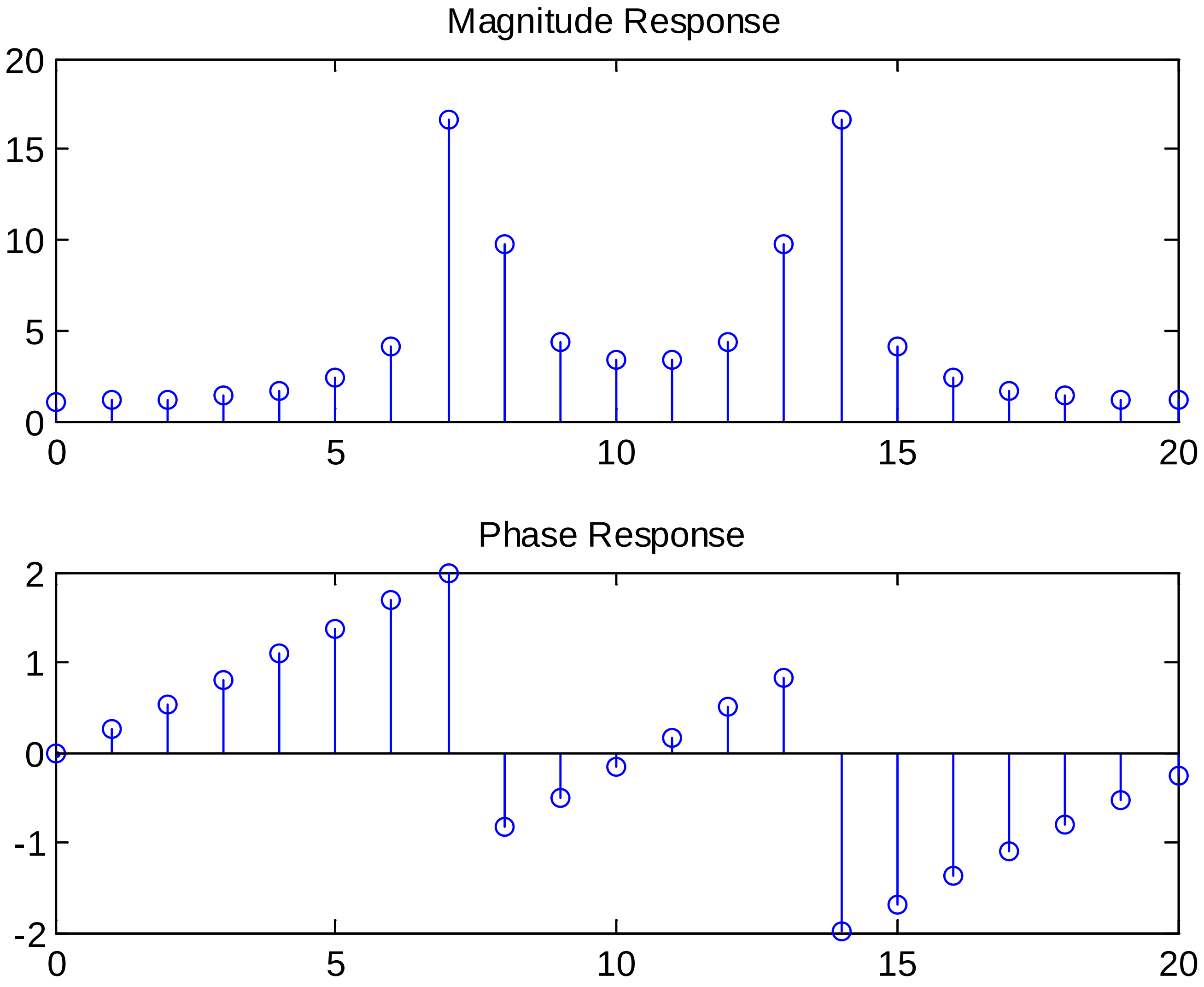


Fig.7.8: DFT plots for a real tone

```
X =
 1.0806              1.0674+0.2939i   1.0243+0.6130i
 0.9382+0.9931i      0.7756+1.5027i   0.4409+2.3159i
-0.4524+4.1068i     -6.7461+15.1792i  6.5451-7.2043i
 3.8608-2.1316i      3.3521-0.5718i   3.3521+0.5718i
 3.8608+2.1316i      6.5451+7.2043i  -6.7461-15.1792i
-0.4524-4.1068i      0.4409-2.3159i   0.7756-1.5027i
 0.9382-0.9931i      1.0243-0.6130i   1.0674-0.2939i
```

Interestingly, we observe that $\Re\{X[k]\} = \Re\{X[N-k]\}$ and $\Im\{X[k]\} = -\Im\{X[N-k]\}$. In fact, all real-valued sequences possess these properties so that we only have to compute around half of the DFT coefficients.

As the DFT coefficients are complex-valued, we search the frequency according to the magnitude plot.

There are two peaks, one at $k = 7$ and the other at $k = 14$ which correspond to $\omega = 0.7\pi$ and $\omega = -0.7\pi$, respectively.

From Example 7.2, it is clear that the index k refers to $\omega = 2\pi k/N$. As a result, an estimate of ω_0 is:

$$\hat{\omega}_0 = \frac{2\pi \cdot 7}{21} \approx 0.6667\pi$$

To improve the accuracy, we append a large number of zeros to $x[n]$. The MATLAB code for $x[n]$ is now modified as:

```
x=[A*cos(w.*n+p) zeros(1,1980)];
```

where 1980 zeros are appended.

The MATLAB code is provided as `ex7_6.m` and `ex7_6_2.m`.

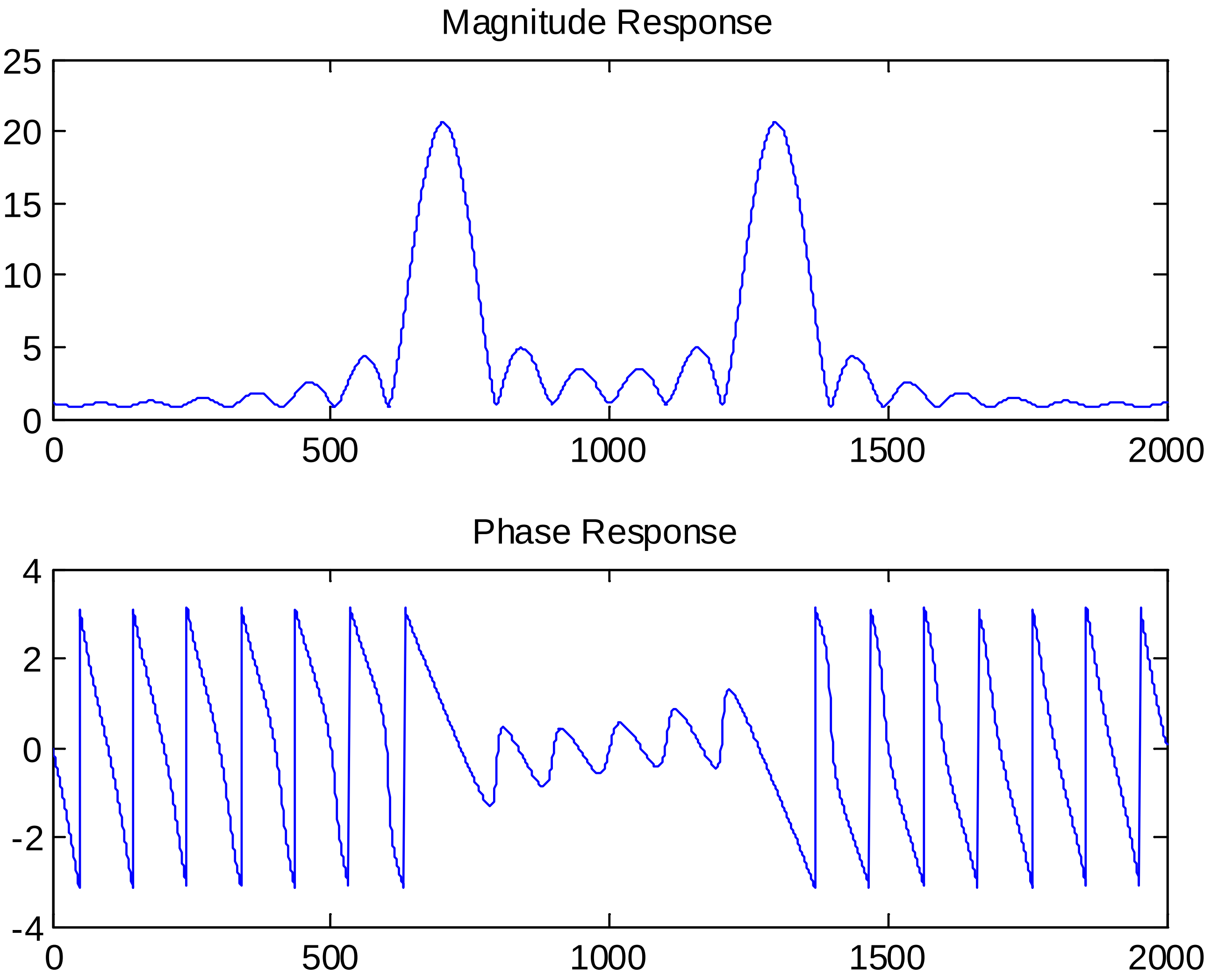


Fig.7.9: DFT plots for a real tone with zero padding

The peak index is found to be $k = 702$ with $N = 2001$. Thus

$$\hat{\omega}_0 = \frac{2\pi \cdot 702}{2001} \approx 0.7016\pi$$

<u>Example 7.7</u>

Find the inverse DFT coefficients for $X[k]$ which has a length of $N = 5$ and has the form of

$$X[k] = \begin{cases} 1, \ n = 0, 1, 2 \\ 0, \ n = 3, 4 \end{cases}$$

Plot $x[n]$.

Using (7.33) and Example 7.5, we have:

$$
\begin{aligned}
x[n] &= \frac{1}{N}\sum_{n=0}^{N-1} X[k] W_N^{-kn} = \frac{1}{5}\left(W_5^0 + W_5^{-n} + W_5^{-2n}\right) \\
&= \begin{cases} \frac{1}{5} e^{\frac{j2\pi n}{5}} \left[1 + 2\cos\left(\frac{2\pi n}{5}\right)\right], & n = 0, 1, \cdots, 4 \\ 0, & \text{otherwise} \end{cases}
\end{aligned}
$$

The main MATLAB code is:

```
N=5;
X=[1 1 1 0 0];
subplot(2,1,1);
stem([0:N-1],abs(ifft(X)));
subplot(2,1,2);
stem([0:N-1],angle(ifft(X)));
```

The MATLAB program is provided as `ex7_7.m`.

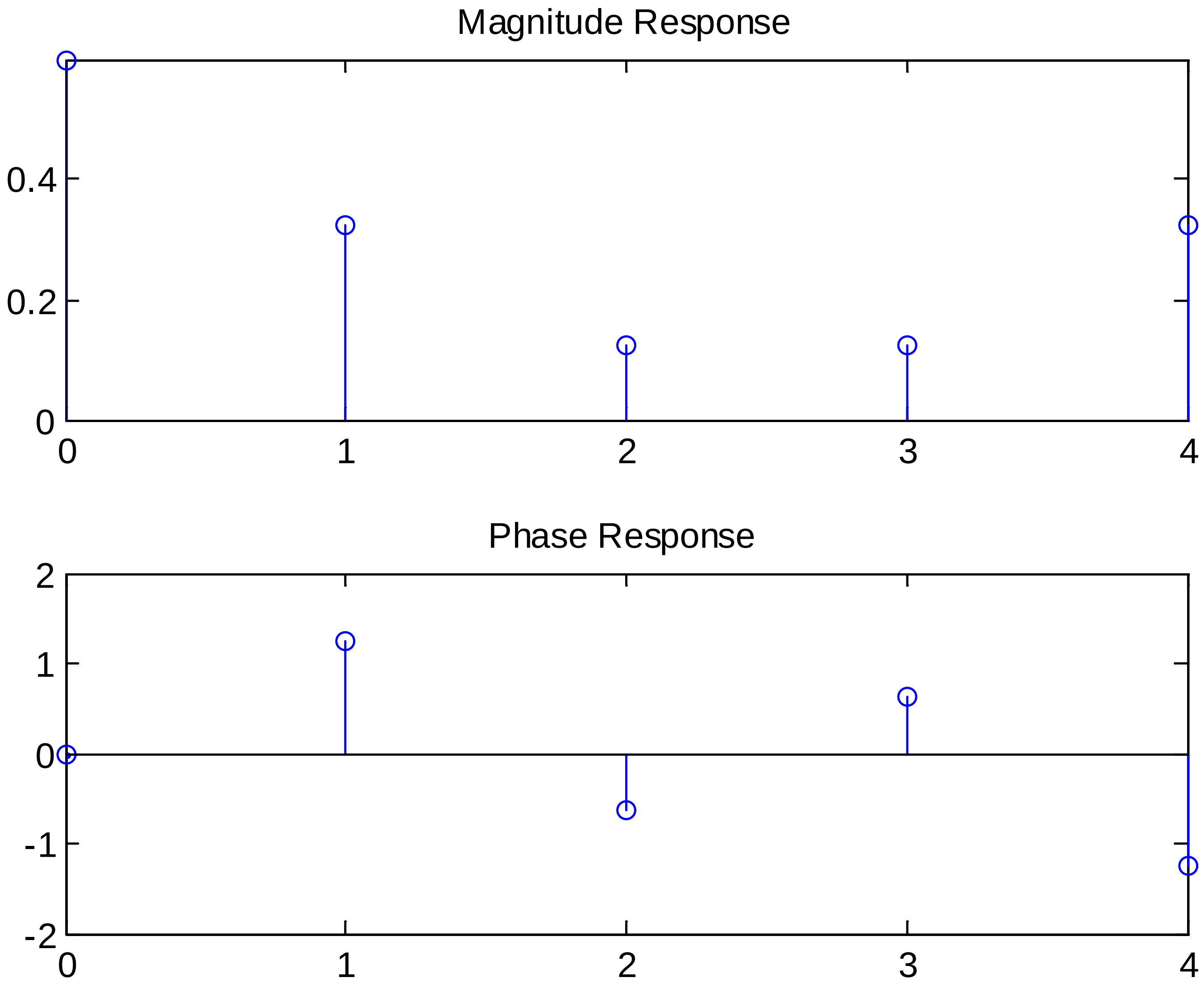


Fig.7.10: Inverse DFT plots

Properties of DFT

Since DFT pair is equal to DFS pair within $[0, N-1]$, their properties will be identical if we take care of the values of $x[n]$ and $X[k]$ when the indices are outside the interval

1. Linearity

Let $(x_1[n], X_1[k])$ and $(x_2[n], X_2[k])$ be two DFT pairs with the same duration of N. We have:

$$ax_1[n] + bx_2[n] \leftrightarrow aX_1[k] + bX_2[k] \tag{7.34}$$

Note that if $x_1[n]$ and $x_2[n]$ are of different lengths, we can properly append zero(s) to the shorter sequence to make them with the same duration.

2. Circular Shift of Sequence

If $x[n] \leftrightarrow X[k]$, then

$$x[(n-m) \mod (N)] \leftrightarrow W_N^{km} X[k] \tag{7.35}$$

Note that in order to make sure that the resultant time index is within the interval of $[0, N-1]$, we need circular shift, which is defined as

$$(n-m) \mod (N) = n - m + r \cdot N \tag{7.36}$$

where the integer r is chosen such that

$$0 \leq n - m + r \cdot N \leq N - 1 \tag{7.37}$$

Example 7.8

Determine $x_1[n] = x[(n-2) \mod (4)]$ where $x[n]$ is of length 4 and has the form of:

$$x[n] = \begin{cases} 1, \ n = 0 \\ 3, \ n = 1 \\ 2, \ n = 2 \\ 4, \ n = 3 \end{cases}$$

According to (7.36)-(7.37) with $N = 4$, $x_1[n]$ is determined as:

$$\begin{aligned} x_1[0] &= x[(0-2) \mod (4)] = x[2] = 2, \quad r = 1 \\ x_1[1] &= x[(1-2) \mod (4)] = x[3] = 4, \quad r = 1 \\ x_1[2] &= x[(2-2) \mod (4)] = x[0] = 1, \quad r = 0 \\ x_1[3] &= x[(3-2) \mod (4)] = x[1] = 3, \quad r = 0 \end{aligned}$$

3. Duality

If $x[n] \leftrightarrow X[k]$, then

$$X[n] \leftrightarrow Nx[(-k) \mod (N)] \tag{7.38}$$

4. Symmetry

If $x[n] \leftrightarrow X[k]$, then

$$x^*[n] \leftrightarrow X^*[(-k) \mod (N)] \tag{7.39}$$

and

$$x^*[(-n) \mod (N)] \leftrightarrow X^*[k] \tag{7.40}$$

5. Circular Convolution

Let $(x_1[n], X_1[k])$ and $(x_2[n], X_2[k])$ be two DFT pairs with the same duration of N. We have

$$x_1[n] \otimes_N x_2[n] = \sum_{m=0}^{N-1} x_1[m] x_2[(n-m) \mod (N)] \leftrightarrow X_1[k] X_2[k] \quad (7.41)$$

where $\otimes_N$ is the circular convolution operator.

Fast Fourier Transform

FFT is a fast algorithm for DFT and inverse DFT computation.

Recall (7.32):

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn}, \quad 0 \le k \le N-1 \qquad (7.42)$$

Each $X[k]$ involves N and $(N-1)$ complex multiplications and additions, respectively.

Computing all DFT coefficients requires N^2 complex multiplications and $N(N-1)$ complex additions.

Assuming that $N = 2^v$, the corresponding computational requirements for FFT are $0.5N\log_2(N)$ complex multiplications and $N\log_2(N)$ complex additions.

N	Direct Computation		FFT	
	Multiplication N^2	Addition $N(N-1)$	Multiplication $0.5N\log_2(N)$	Addition $N\log_2(N)$
2	4	2	1	2
8	64	56	12	24
32	1024	922	80	160
64	4096	4022	192	384
2^{10}	1048576	1047552	5120	10240
2^{20}	$\sim 10^{12}$	$\sim 10^{12}$	$\sim 10^{7}$	$\sim 2\times 10^{7}$

Table 7.1: Complexities of direct DFT computation and FFT

Basically, FFT makes use of two ideas in its development:

- Decompose the DFT computation of a sequence into successively smaller DFTs

- Utilize two properties of $W_N^k = e^{-j2\pi k/N}$:

  - complex conjugate symmetry property:

$$W_N^{k(N-n)} = W_N^{-kn} = \left(W_N^{kn}\right)^* \tag{7.43}$$

  - periodicity in n and k:

$$W_N^{kn} = W_N^{k(n+N)} = W_N^{n(k+N)} \tag{7.44}$$

Decimation-in-Time Algorithm

The basic idea is to compute (7.42) according to

$$X[k] = \sum_{n=\text{even}}^{N-1} x[n]W_N^{kn} + \sum_{n=\text{odd}}^{N-1} x[n]W_N^{kn} \qquad (7.45)$$

Substituting $n = 2r$ and $n = 2r + 1$ for the first and second summation terms:

$$\begin{aligned} X[k] &= \sum_{r=0}^{N/2-1} x[2r]W_N^{2rk} + \sum_{r=0}^{N/2-1} x[2r+1]W_N^{(2r+1)k} \\ &= \sum_{r=0}^{N/2-1} x[2r]\left(W_N^2\right)^{rk} + W_N^k \sum_{r=0}^{N/2-1} x[2r+1]\left(W_N^2\right)^{rk} \qquad (7.46) \end{aligned}$$

Using $W_N^2 = W_{N/2}$ since $W_N^2 = e^{-j2\pi/N\cdot 2} = e^{-j2\pi/(N/2)}$, we have:

$$\begin{aligned} X[k] &= \sum_{r=0}^{N/2-1} x[2r] W_{N/2}^{rk} + W_N^k \sum_{r=0}^{N/2-1} x[2r+1] W_{N/2}^{rk} \\ &= G[k] + W_N^k \cdot H[k], \quad k = 0, 1, \cdots, N-1 \end{aligned} \qquad (7.47)$$

where $G[k]$ and $H[k]$ are the DFTs of the even-index and odd-index elements of $x[n]$, respectively. That is, $X[k]$ can be constructed from two $N/2$-point DFTs, namely, $G[k]$ and $H[k]$.

Further simplifications can be achieved by writing the N equations as 2 groups of $N/2$ equations as follows:

$$X[k] = G[k] + W_N^k \cdot H[k], \quad k = 0, 1, \cdots, N/2-1 \qquad (7.48)$$

and

$$\begin{aligned} X[k+N/2] &= \sum_{r=0}^{N/2-1} x[2r]W_{N/2}^{r(k+N/2)} + W_N^{k+N/2} \sum_{r=0}^{N/2-1} x[2r+1]W_{N/2}^{r(k+N/2)} \\ &= \sum_{r=0}^{N/2-1} x[2r]W_{N/2}^{rk} - W_N^k \sum_{r=0}^{N/2-1} x[2r+1]W_{N/2}^{rk} \\ &= G[k] - W_N^k \cdot H[k], \quad k = 0, 1, \cdots, N/2-1 \end{aligned} \tag{7.49}$$

with the use of $W_{N/2}^{N/2} = 1$ and $W_N^{N/2} = -1$. Equations (7.48) and (7.49) are known as the butterfly merging equations.

Noting that $N/2$ multiplications are also needed to calculate $W_N^k H[k]$, the number of multiplications is reduced from N^2 to $2(N/2)^2 + N/2 = N(N+1)/2$.

The decomposition step of (7.48)-(7.49) is repeated v times until 1-point DFT is reached.

Decimation-in-Frequency Algorithm

The basic idea is to decompose the frequency-domain sequence $X[k]$ into successively smaller subsequences.

Recall (7.42) and employing $W_N^{2r(n+N/2)} = W_N^{2nr} \cdot W_N^{rN} = W_N^{2nr}$ and $W_N^2 = W_{N/2}$, the even-index DFT coefficients are:

$$
\begin{aligned}
X[2r] &= \sum_{n=0}^{N-1} x[n] W_N^{n(2r)} = \sum_{n=0}^{N/2-1} x[n] W_N^{2nr} + \sum_{n=N/2}^{N-1} x[n] W_N^{2nr} \\
&= \sum_{n=0}^{N/2-1} x[n] W_N^{2nr} + \sum_{n=0}^{N/2-1} x[n+N/2] W_N^{2r(n+N/2)} \\
&= \sum_{n=0}^{N/2-1} (x[n] + x[n+N/2]) \cdot W_{N/2}^{nr}, \quad r = 0, 1, \cdots, N/2-1
\end{aligned}
\qquad (7.50)
$$

Using $W_N^{Nr}=1$ and $W_N^{N/2}=-1$, the odd-index coefficients are:

$$\begin{aligned} X[2r+1] &= \sum_{n=0}^{N/2-1} x[n]W_N^{n(2r+1)} + \sum_{n=N/2}^{N-1} x[n]W_N^{n(2r+1)} \\ &= \sum_{n=0}^{N/2-1} x[n]W_N^n W_{N/2}^{nr} + \sum_{n=0}^{N/2-1} x[n+N/2]W_N^{(n+N/2)(2r+1)} \\ &= \sum_{n=0}^{N/2-1} x[n]W_N^n W_{N/2}^{nr} + W_N^{N/2(2r+1)} \sum_{n=0}^{N/2-1} x[n+N/2]W_N^{n(2r+1)} \\ &= \sum_{n=0}^{N/2-1} \left(x[n]-x[n+N/2]\right) W_N^n \cdot W_{N/2}^{nr}, \quad r=0,1,\cdots,N/2-1 \end{aligned} \quad (7.51)$$

$X[2r]$ and $X[2r+1]$ are equal to $N/2$ -point DFTs of $(x[n]+x[n+N/2])$ and $(x[n]-x[n+N/2])\,W_N^n$, respectively. The decomposition step of (7.50)-(7.51) is repeated v times until 1-point DFT is reached.

Chapter 8: Responses of Digital Filters

Chapter Intended Learning Outcomes:

(i) Understanding the relationships between impulse response, frequency response, difference equation and transfer function in characterizing a LTI system

(ii) Ability to identify infinite impulse response (IIR) and finite impulse response (FIR) filters. Note that a digital filter is system which processes discrete-time signals

(iii) Ability to compute system frequency response

LTI System Characterization

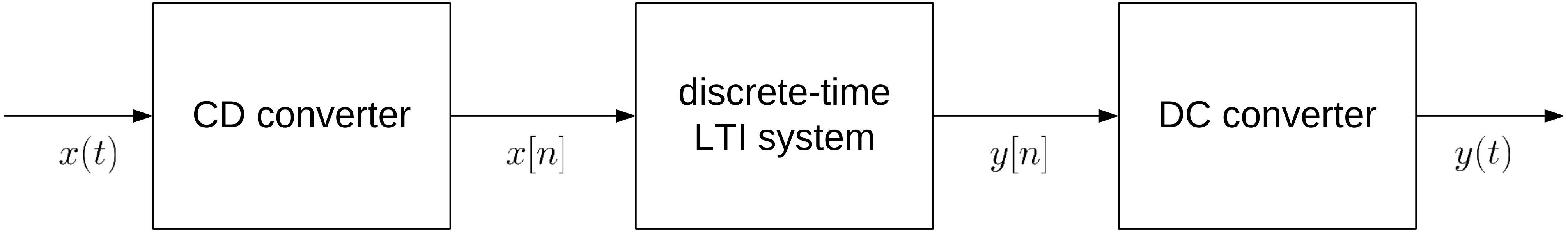


Fig.8.1: Processing of analog signal with LTI filter

The LTI system can be characterized as

- Impulse Response

Let $h[n]$ be the impulse response of the LTI filter. Recall from (3.17), it characterizes the system via the convolution:

$$y[n] = x[n] \otimes h[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m] \qquad (8.1)$$

$h[n]$ is the time-domain response of the LTI filter

- Frequency Response

We have from (6.17), which is the discrete-time Fourier transform (DTFT) of (8.1):

$$Y(e^{j\omega}) = X(e^{j\omega})H(e^{j\omega}) \tag{8.2}$$

$H(e^{j\omega})$ is the frequency-domain response of the LTI filter

- Difference Equation

A LTI system satisfies a difference equation of the form:

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k] \tag{8.3}$$

- Transfer Function

Taking the z transform on both sides of (8.3):

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} \qquad (8.4)$$

Which of them can uniquely characterize a system? Which of them cannot?

Example 8.1

Given the difference equation with input $x[n]$ and output $y[n]$:

$$y[n] = ay[n-1] + x[n]$$

Find all possible ways to compute $y[n]$ given $x[n]$.

In fact, there are two ways to compute $y[n]$.

A straightforward and practical way is to implement a causal system by using the difference equation recursively with a given initial condition of $y[-1]$:

$$\begin{aligned} y[0] &= ay[-1] + x[0] \\ y[1] &= ay[0] + x[1] \\ y[2] &= ay[1] + x[2] \\ \cdots & \quad \cdots \quad \cdots \end{aligned}$$

On the other hand, it is possible to implement a noncausal system via reorganizing the difference equation as:

$$y[n] = ay[n-1] + x[n] \Rightarrow y[n-1] = \frac{1}{a}(y[n] - x[n])$$

In doing so, we need an initial future value of $y[-1]$ and future inputs, and the recursive implementation is:

$$\begin{aligned} y[-2] &= \frac{1}{a}(y[-1] - x[-1]) \\ y[-3] &= \frac{1}{a}(y[-2] - x[-2]) \\ y[-4] &= \frac{1}{a}(y[-3] - x[-3]) \\ \cdots & \quad \cdots \quad \cdots \end{aligned}$$

As a result, generally the difference equation cannot uniquely characterize the system.

Nevertheless, if causality is assumed, then the difference equation corresponds to a unique LTI system.

Alternatively, we can also study the computation of $y[n]$ using system transfer function $H(z)$:

$$Y(z) = az^{-1}Y(z) + X(z) \Rightarrow H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 - az^{-1}}$$

Since the ROC is not specified, there are two possible cases, namely, $|z| > |a|$ and $|z| < |a|$.

For $|z| > |a|$ or $|az^{-1}| < 1$, using inverse z transform:

$$\begin{aligned}\frac{Y(z)}{X(z)} &= \frac{1}{1-az^{-1}} = 1 + az^{-1} + a^2z^{-2} + \cdots \\ &\Rightarrow Y(z) = X(z)\left(1 + az^{-1} + a^2z^{-2} + \cdots\right) \\ &\Rightarrow y[n] = x[n] + ax[n-1] + a^2x[n-2] + \cdots = \sum_{k=0}^{\infty} a^k x[n-k]\end{aligned}$$

which corresponds to a causal system.

The same result can also be produced by first finding the system impulse response $h[n]$ via inverse z transform of $H(z)$:

$$h[n] = a^n u[n] \leftrightarrow H(z) = \frac{1}{1-az^{-1}}, \quad |z| > |a|$$

and then computing $x[n] \otimes h[n]$.

For $|z| < |a|$ or $|a^{-1}z| < 1$, using inverse z transform:

$$\begin{aligned}
\frac{Y(z)}{X(z)} &= \frac{1}{1-az^{-1}} = \frac{-a^{-1}z}{1-a^{-1}z} = -a^{-1}z\left(1 + a^{-1}z + a^{-2}z^2 + \cdots\right) \\
&\Rightarrow Y(z) = -X(z)\left(a^{-1}z + a^{-2}z^2 + a^{-3}z^3 + \cdots\right) \\
&\Rightarrow y[n] = -a^{-1}x[n+1] - a^{-2}x[n+2] - a^{-3}x[n+3] + \cdots \\
&= \sum_{k=-\infty}^{-1} -a^k x[n-k]
\end{aligned}$$

which corresponds to a noncausal system.

Example 8.2

Discuss all possibilities for the LTI system whose input $x[n]$ and output $y[n]$ are related by:

$$y[n] - 5.4y[n-1] + 2y[n-2] = x[n]$$

Taking z transform on the difference equation yields:

$$H(z)=\frac{1}{1-5.4z^{-1}+2z^{-2}}=\frac{1}{(1-0.4z^{-1})(1-5z^{-1})}=\frac{z^2}{(z-0.4)(z-5)}$$

There are 3 possible choices:

$|z|>5$: The ROC is outside the circle with radius characterized by the largest-magnitude pole. The system can be causal but is not stable since the ROC does not include the unit circle

$0.4<|z|<5$: The system is stable because the ROC includes the unit circle but not causal

$|z|<0.4$: The system is unstable and noncausal

Impulse Response of Digital Filters

When $H(z)$ is a rational function of z^{-1} with only first-order poles, we have from (5.26):

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} = \sum_{l=0}^{M-N} B_l z^{-l} + \sum_{k=1}^{N} \frac{A_k}{1 - c_k z^{-1}} \qquad (8.5)$$

where the first component is present only if $M \geq N$.

If the system is causal, then the ROC must be of the form $|z| > |p_{\max}|$ where $p_{\max}$ is the largest-magnitude pole.

According to this ROC, the impulse response $h[n]$ is:

$$h[n] = \sum_{l=0}^{M-N} B_l \delta[n-l] + \sum_{k=1}^{N} A_k (c_k)^n u[n] \qquad (8.6)$$

There are two possible cases for (8.6) which correspond to IIR and FIR filters:

- IIR Filter

If $N \geq 1$ or there is at least one pole, the system is referred to as an IIR filter because $h[n]$ is of infinite duration.

- FIR Filter

If $N = 0$ or there is no pole, the system is referred to as a FIR filter because $h[n]$ is of finite duration.

Notice that the definitions of IIR and FIR systems also hold for noncausal systems.

Example 8.3

Determine if the following difference equations correspond to IIR or FIR systems. All systems are assumed causal.

(a) $y[n] = 0.1y[n-1] + x[n]$

(b) $y[n] = x[n] + 2x[n-1] + 3x[n-2]$

Taking the z transform on (a) yields

$$Y(z) = az^{-1}Y(z) + X(z) \Rightarrow H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1-0.1z^{-1}}$$

which has one pole. For causal system:

$$h[n] = (0.1)^n u[n] \leftrightarrow H(z) = \frac{1}{1 - 0.1z^{-1}}, \quad |z| > 0.1$$

which is a right-sided sequence and corresponds to an IIR system as $h[n]$ is of infinite duration.

Similarly, we have for (b):

$$Y(z) = X(z) + 2z^{-1}X(z) + 3z^{-2}X(z) \Rightarrow H(z) = \frac{Y(z)}{X(z)} = 1 + 2z^{-1} + 3z^{-2}$$

which does not have any nonzero pole. Hence

$$h[n] = \delta[n] + 2\delta[n-1] + 3\delta[n-2]$$

which corresponds to a FIR system as $h[n]$ is of finite duration.

Frequency Response of Digital Filters

The frequency response of a LTI system whose impulse response $h[n]$ is obtained by taking the DTFT of $h[n]$:

$$H(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h[n]e^{-j\omega n} \tag{8.7}$$

According to (5.8), $H(e^{j\omega})$ is also obtained if $H(z)$ is available:

$$H(e^{j\omega}) = H(z)\big|_{z=e^{j\omega}} = \frac{\displaystyle\sum_{k=0}^{M} b_k e^{-j\omega k}}{\displaystyle\sum_{k=0}^{N} a_k e^{-\omega k}} \tag{8.8}$$

assuming that the ROC of $H(z)$ includes the unit circle.

Example 8.4

Plot the frequency response of the system with impulse response $h[n]$ of the form:

$$h[n] = \begin{cases} \mathrm{sinc}(0.1(n-50)), & 0 \leq n \leq 100 \\ 0, & \text{otherwise} \end{cases}$$

where

$$\mathrm{sinc}(u) = \frac{\sin(\pi u)}{\pi u}$$

Following Example 7.6, we append a large number of zeros at the end of $h[n]$ prior to performing discrete Fourier transform (DFT) to produce more DTFT samples.

Is it a low-pass filter? Why?

The MATLAB program is provided as `ex8_4.m`.

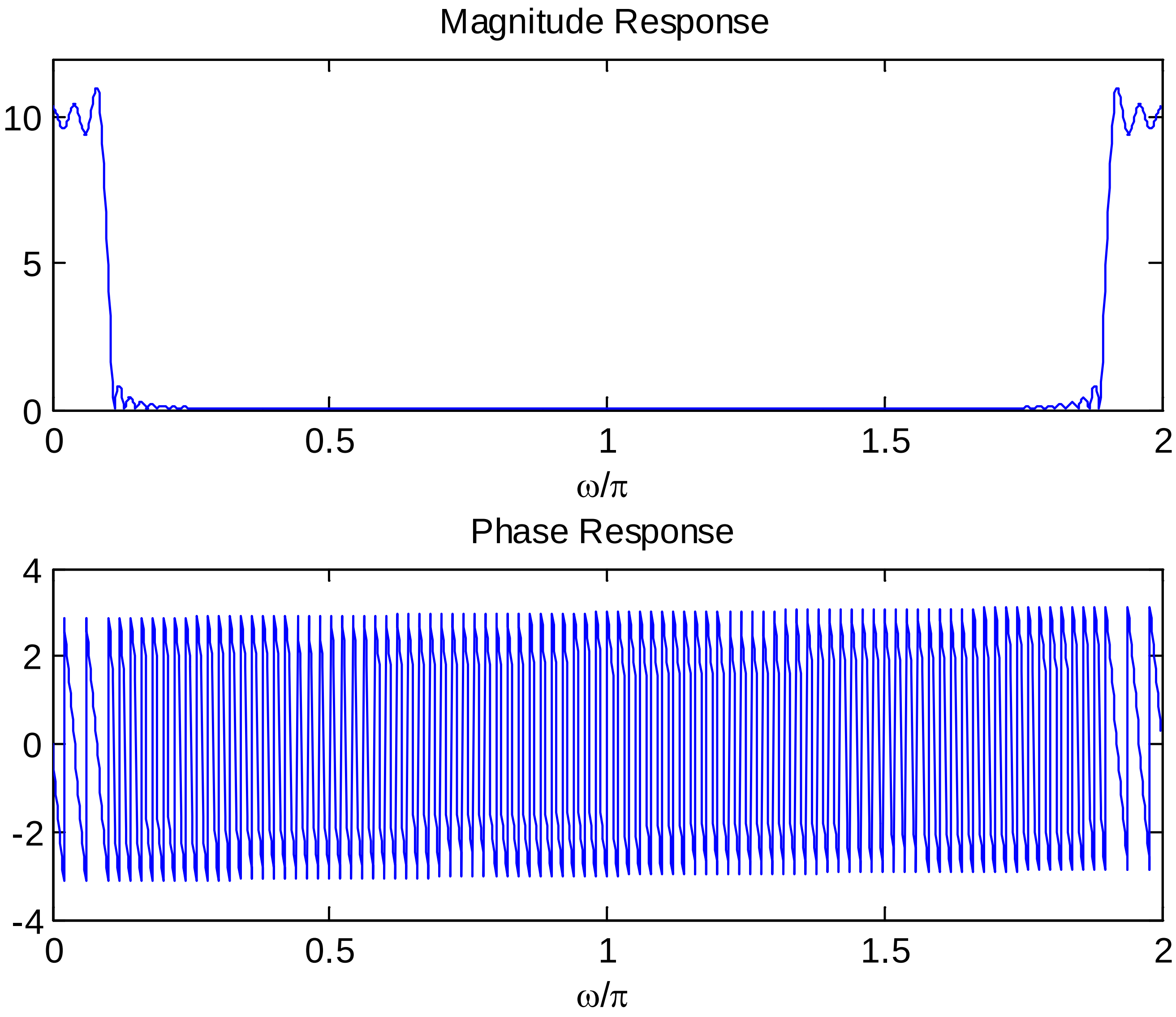


Fig.8.2: Frequency response for sinc $h[n]$

Example 8.5

Plot the frequency response of the system with transfer function $H(z)$ of the form:

$$H(z) = \frac{1 + 2z^{-1} + 3z^{-2}}{2 + 3z^{-1} + 4z^{-2}}$$

It is assumed that the ROC of $H(z)$ includes the unit circle.

The MATLAB code is

```
b=[1,2,3];
a=[2,3,4];
freqz(b,a);
```

Is it a lowpass filter? Why?

The MATLAB program is provided as `ex8_5.m`.

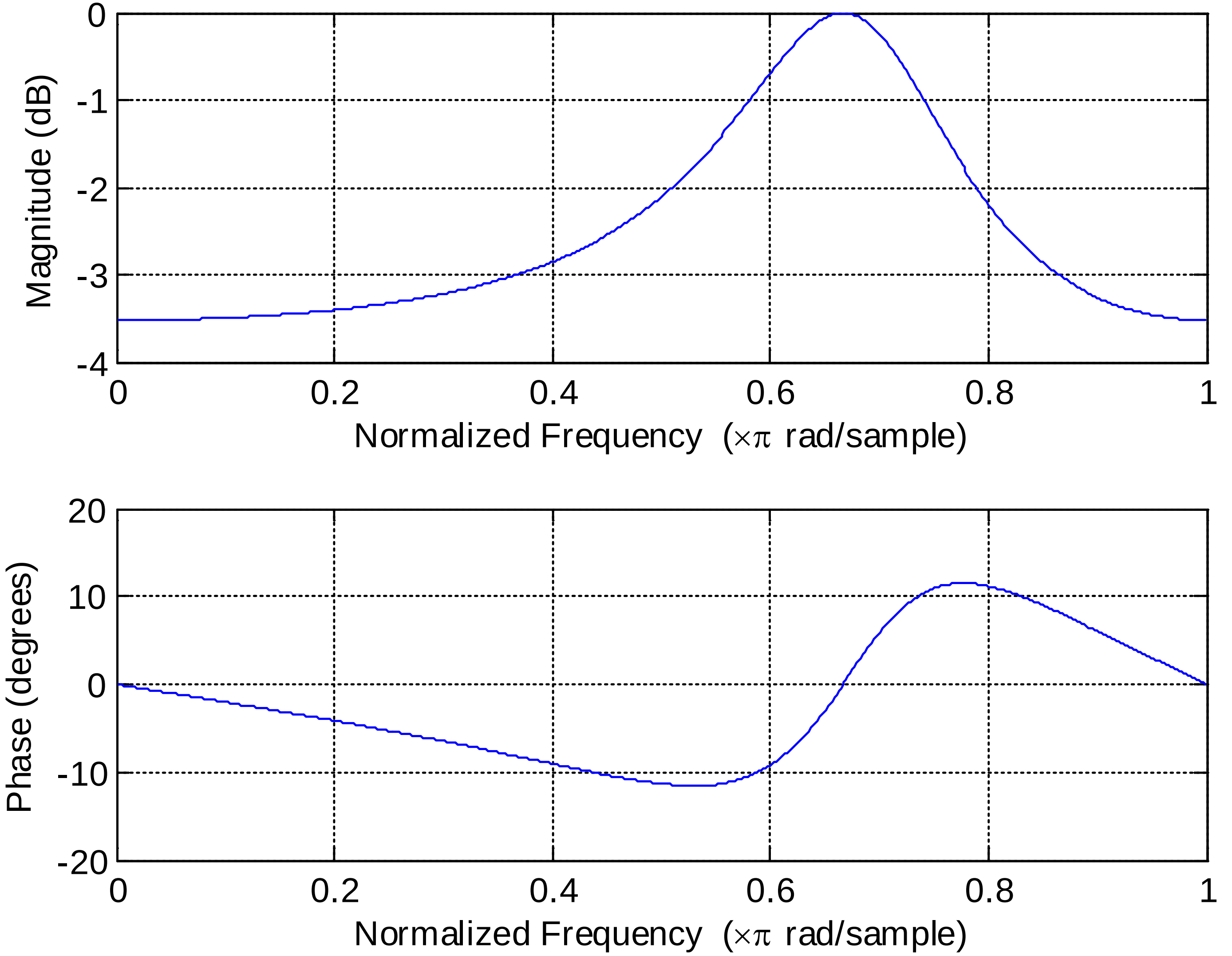


Fig.8.3: Frequency response for second-order system

Chapter 9: Realization of Digital Filters

Chapter Intended Learning Outcomes:

(i) Ability to implement finite impulse response (FIR) and infinite impulse response (IIR) filters using different structures in terms of block diagram and signal flow graph

(ii) Ability to determine the system transfer function and difference equation given the corresponding block diagram or signal flow graph representation.

Filter Implementation

When causality is assumed, a LTI filter can be uniquely characterized by its transfer function $H(z)$:

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} \tag{9.1}$$

or the corresponding difference equation:

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k] \tag{9.2}$$

where $x[n]$ and $y[n]$ are the system input and output

Assuming $a_0 \neq 0$, the output is:

$$y[n] = -\sum_{k=1}^{N} \frac{a_k}{a_0} y[n-k] + \sum_{k=0}^{M} \frac{b_k}{a_0} x[n-k] \qquad (9.3)$$

Assigning $a_k/a_0 \rightarrow a_k$ and $b_k/b_0 \rightarrow b_k$ yields

$$y[n] = -\sum_{k=1}^{N} a_k y[n-k] + \sum_{k=0}^{M} b_k x[n-k] \qquad (9.4)$$

Computing $y[n]$ involves $y[n-1], y[n-2], \cdots, y[n-N]$, and $x[n], x[n-1], \cdots, x[n-M]$. That is, we need

- Delay elements or storage
- Multipliers
- Adders (subtraction is considered as addition)

How many storage elements are needed?
How many multipliers are needed?
How many adders are needed?

Computations of $y[n]$ can be arranged in different ways to give the same difference equation, which leads to different structures for realization of discrete-time LTI systems

4 basic forms of implementations, namely, direct form, canonic form, cascade form and parallel form will be described

An implementation can be represented using either a block diagram or a signal flow graph

Block Diagram Representation

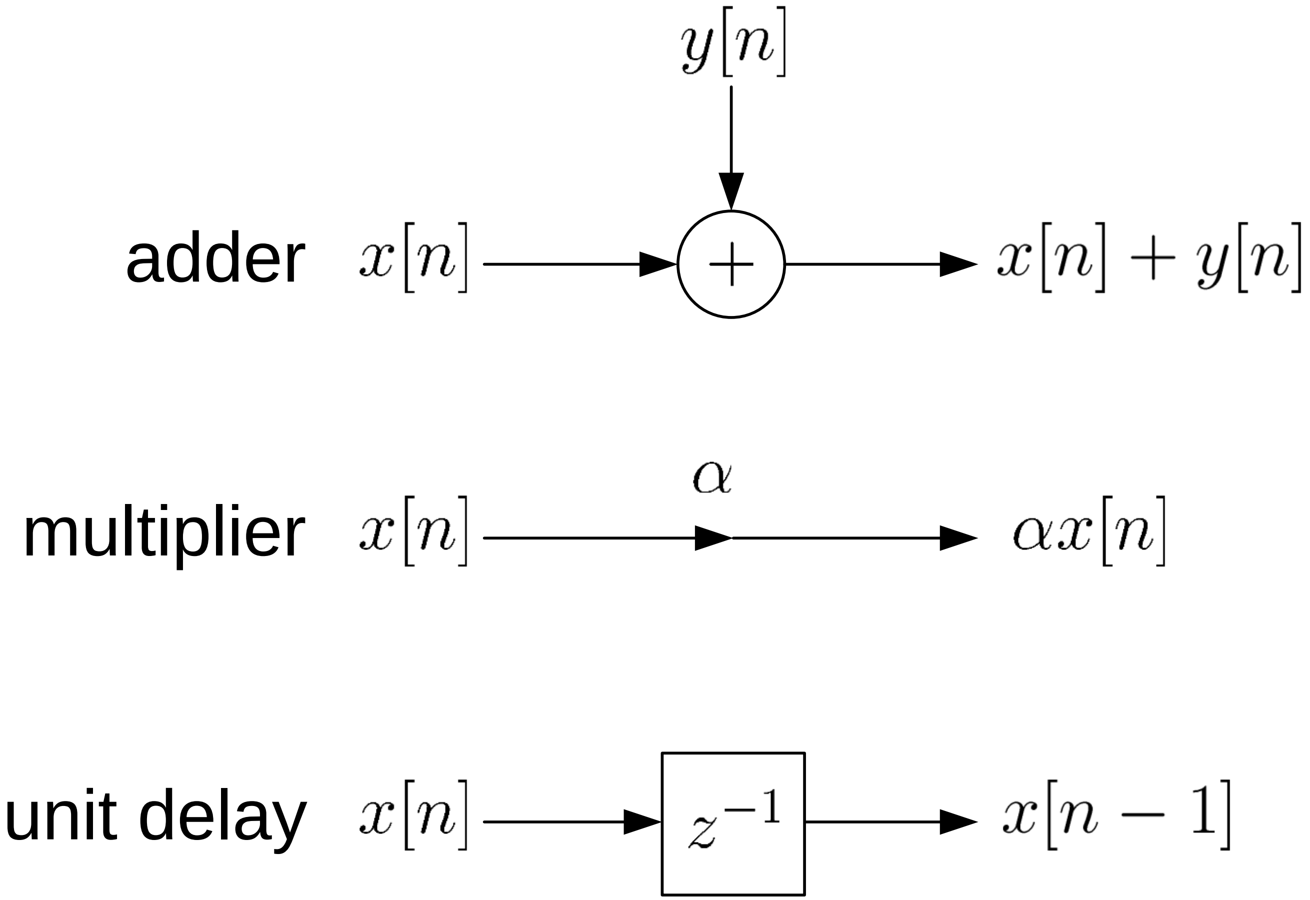


Fig.9.1: Basic operations in block diagram

Although an adder can generally deal with more than two sequences, here we consider two signals in order to align with practical implementation in microprocessors.

When $|\alpha| > 1$, it corresponds to signal amplification while the signal is attenuated for $|\alpha| < 1$. Note that a multiplier usually has the highest implementation or computational cost and thus it is desired to reduce the number of multipliers in different systems, if possible.

The transfer function z^{-1} corresponds to a unit delay. It can be implemented by providing a storage register for each unit delay in digital implementation. If the required number of samples of delay is $D > 1$, then the corresponding system function is z^{-D}.

Signal Flow Graph Representation

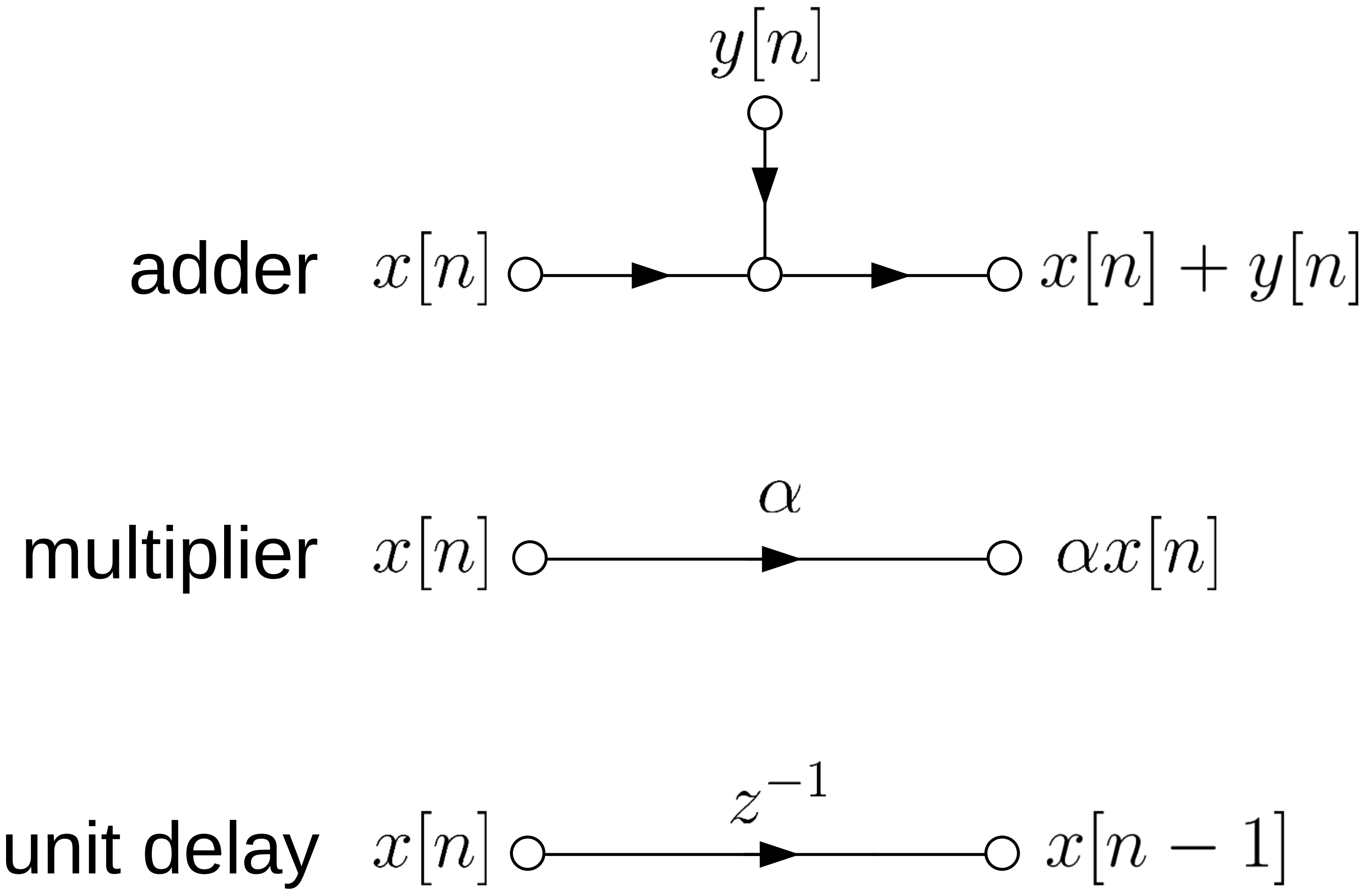


Figure 9.2: Basic operations in signal flow graph

Its basic elements are branches with directions, and nodes. That is, a signal flow graph is a set of directed branches that connect at nodes.

Signal at a node of a flow graph is equal to the sum of the signals from all branches connecting to the node.

Signal out of a branch is equal to the branch gain times the signal into the branch.

Branch gain can refer to a scalar or a transfer function of z^{-1} corresponding to multiplication or unit delay operation, respectively.

When the branch gain is unity, it is left unlabeled.

A signal flow graph provides an alternative but equivalent graphical representation to a block diagram structure

Example 9.1

Draw the block diagram and signal flow graph representations of a LTI system whose input $x[n]$ and output $y[n]$ satisfy the following difference equation:

$$y[n] = -a_1 y[n-1] - a_2 y[n-2] + b_0 x[n]$$

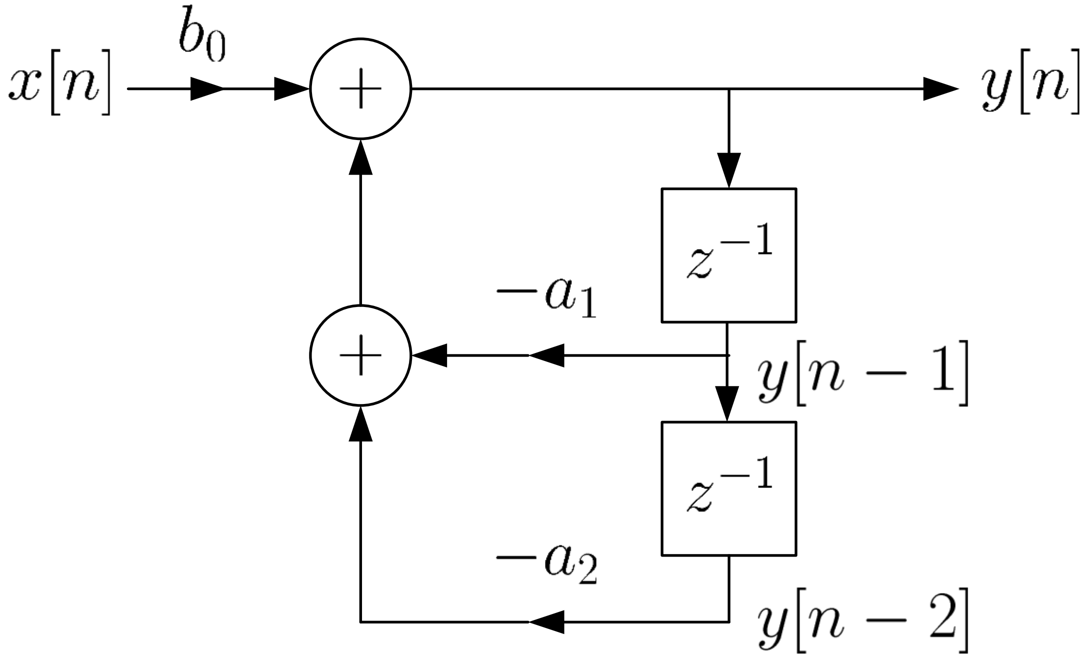


Fig.9.3: Illustration of block diagram

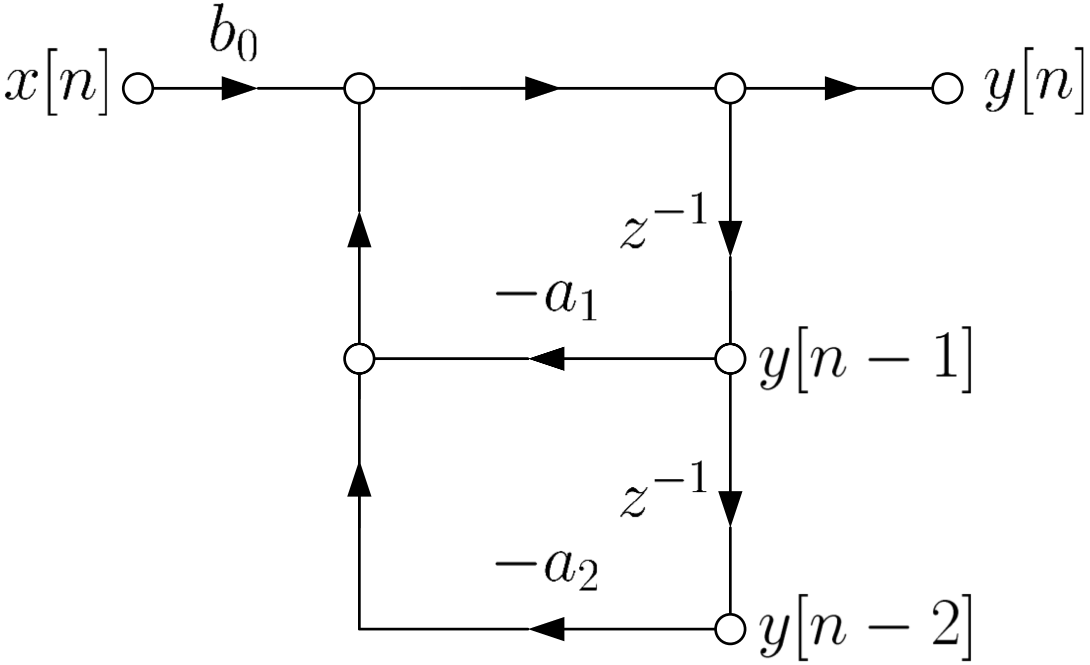


Fig.9.4: Illustration of signal flow graph

2 adders, 3 multipliers and 2 delay elements are required to implement the system.

Structures for FIR Filter

For FIR filter, its transfer function does not contain pole. That is, setting $a_0 = 1$ and $a_1 = a_2 = \cdots = a_N = 0$ in (9.1) yields a FIR system:

$$H(z) = \sum_{k=0}^{M} b_k z^{-k} \tag{9.5}$$

while the corresponding difference equation is:

$$y[n] = \sum_{k=0}^{M} b_k x[n-k] \tag{9.6}$$

1. Direct Form

Comparing (9.6) with the convolution formula of (3.18), $\{b_k\}$ correspond to the system impulse response $h[n]$:

$$h[n] = \begin{cases} b_n, & 0 \leq n \leq M \\ 0, & \text{otherwise} \end{cases} \qquad (9.7)$$

(9.6) can also be written as:

$$y[n] = \sum_{k=0}^{M} h[k]x[n-k] = h[0]x[n] + h[1]x[n-1] + \cdots h[M]x[n-M] \qquad (9.8)$$

The direct form follows straightforwardly from the difference equation.

The implementation needs M memory locations for storing M previous inputs of $x[n]$, $(M+1)$ multiplications and M additions for computing each output value of $y[n]$.

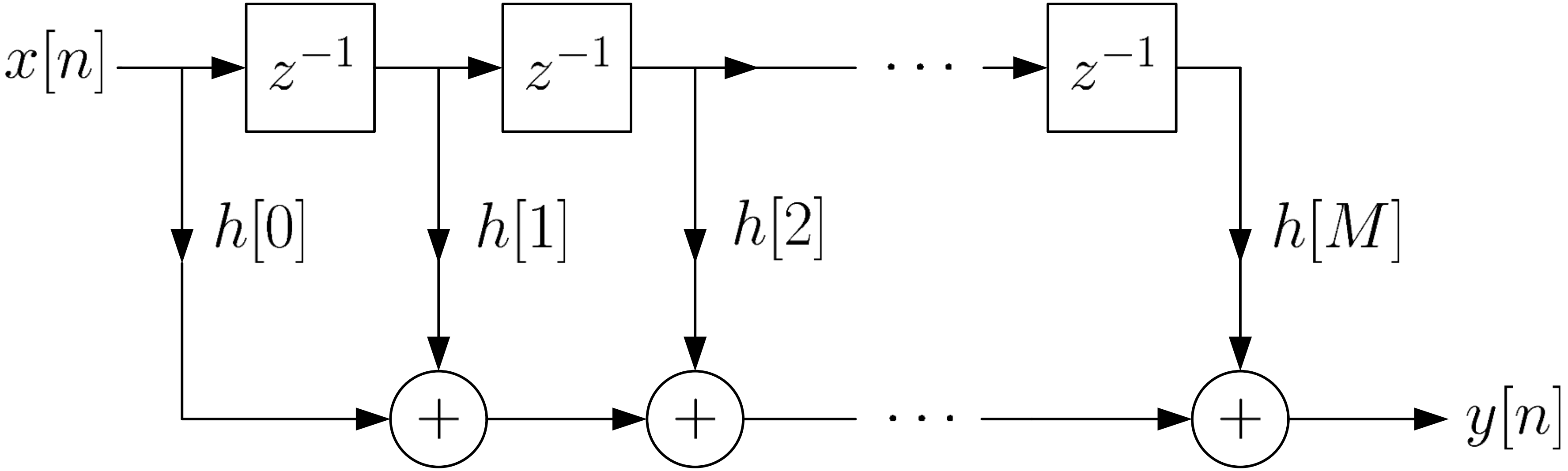


Fig.9.5: Direct form of FIR filter

2. Cascade Form

Expressing (9.5) as products of second-order polynomial system functions via factorization:

$$H(z) = \sum_{k=0}^{M} b_k z^{-k} = \prod_{k=1}^{M_c} \left(\beta_{0k} + \beta_{1k} z^{-1} + \beta_{2k} z^{-2}\right) \qquad (9.9)$$

where $M_c = \lfloor (M+1)/2 \rfloor$ is the largest integer contained in $(M+1)/2$. Note that when M is odd, one of the $\{\beta_{2k}\}$ will be zero. Assuming that M is even, this implementation needs M storage elements, $3M/2$ multiplications and M additions, for computing each output value of $y[n]$.

Why second-order polynomial instead of first-order polynomial?

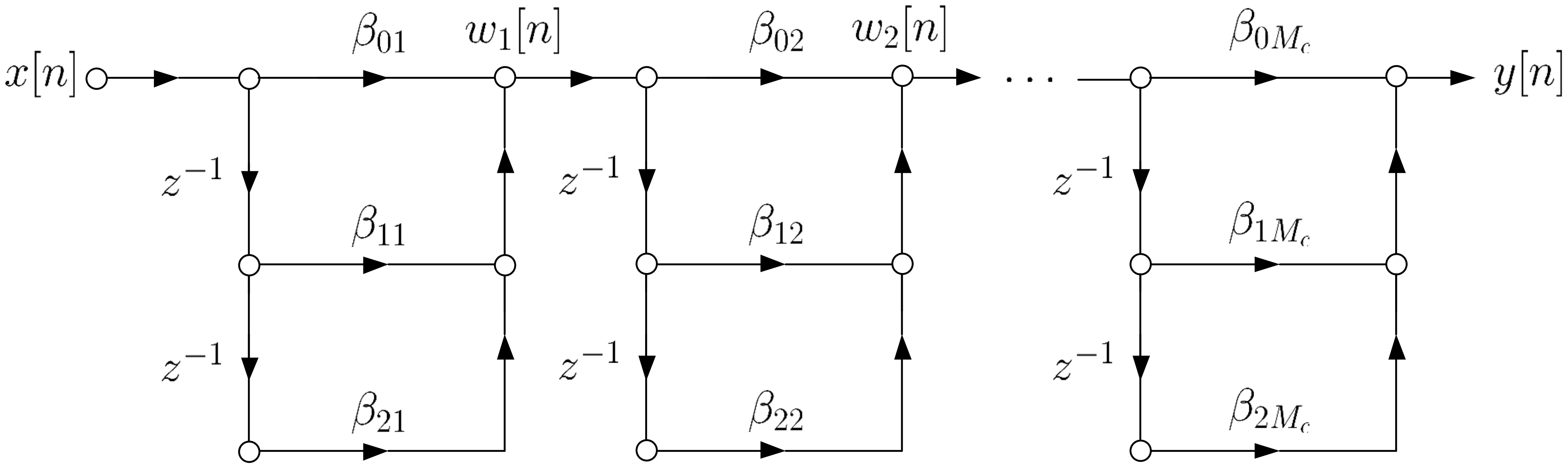


Fig.9.6: Cascade form of FIR filter

$$w_1[n] = \beta_{01}x[n] + \beta_{11}x[n-1] + \beta_{21}x[n-2] \tag{9.10}$$

and

$$w_2[n] = \beta_{02}w_1[n] + \beta_{12}w_1[n-1] + \beta_{22}w_1[n-2] \tag{9.11}$$

Taking z transform on (9.10) and (9.11):

$$\begin{aligned} W_1(z) &= \beta_{01}X(z) + \beta_{11}z^{-1}X(z) + \beta_{21}z^{-2}X(z) \\ &= \left(\beta_{01} + \beta_{11}z^{-1} + \beta_{21}z^{-2}\right) X(z) \end{aligned} \quad (9.12)$$

and

$$\begin{aligned} W_2(z) &= \beta_{02}W_1(z) + \beta_{12}z^{-1}W_1(z) + \beta_{22}z^{-2}W_1(z) \\ &= \left(\beta_{02} + \beta_{12}z^{-1} + \beta_{22}z^{-2}\right) W_1(z) \end{aligned} \quad (9.13)$$

Substituting (9.13) into (9.12) yields:

$$W_2(z) = \left(\beta_{01} + \beta_{11}z^{-1} + \beta_{21}z^{-2}\right)\left(\beta_{02} + \beta_{12}z^{-1} + \beta_{22}z^{-2}\right) X(z) \quad (9.14)$$

Extending (9.14) to $y[n]$, we finally get:

$$\begin{aligned} Y(z) &= X(z)\left(\beta_{01} + \beta_{11}z^{-1} + \beta_{21}z^{-2}\right)\cdots\left(\beta_{0M_c} + \beta_{1M_c}z^{-1} + \beta_{2M_c}z^{-2}\right) \\ &= X(z)\prod_{k=1}^{M_c}\left(\beta_{0k} + \beta_{1k}z^{-1} + \beta_{2k}z^{-2}\right) \end{aligned} \quad (9.15)$$

To save the computational complexity, we express (9.9) as:

$$H(z)=G\prod_{k=1}^{M_c}\left(1+\beta'_{1k}z^{-1}+\beta'_{2k}z^{-2}\right) \qquad (9.16)$$

where $G=\beta_{01}\beta_{02}\cdots\beta_{0M_c}$, $\beta'_{1k}=\beta_{1k}/\beta_{0k}$ and $\beta'_{2k}=\beta_{2k}/\beta_{0k}$, $k=1,2,\cdots,M_c$. That is, all $\{\beta_{0k}\}$ are normalized to 1.

Assuming that M is even, (9.16) needs M delay elements, $(M+1)$ multiplications and M additions, for computing each output value of $y[n]$

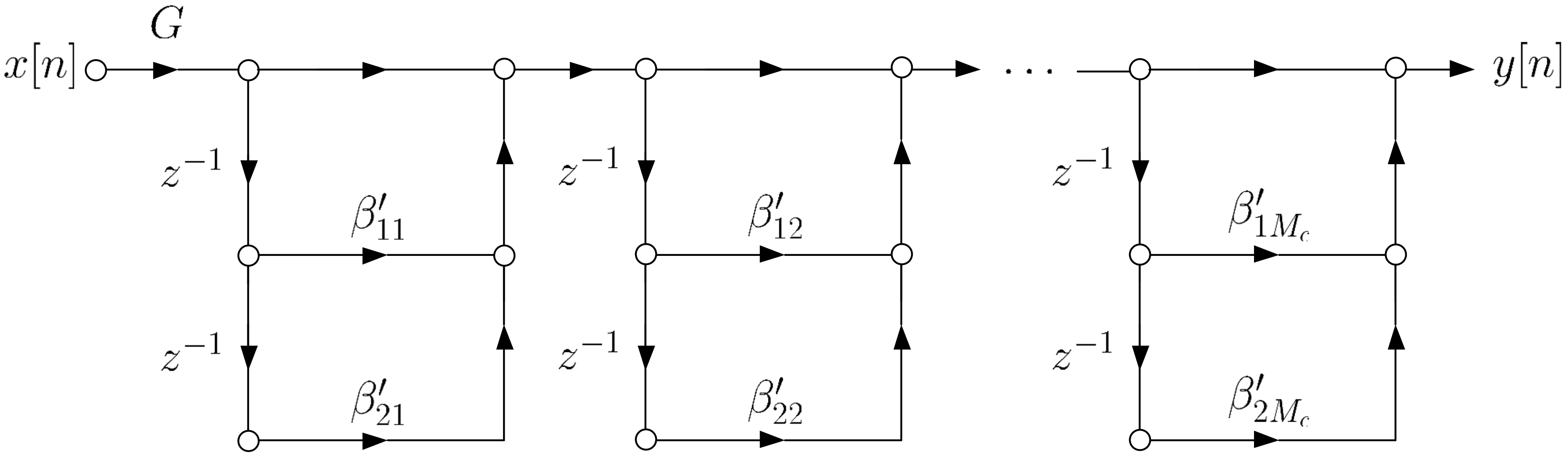


Fig.9.7: Cascade form of FIR filter with smaller complexity

Example 9.2

Draw the signal flow graph using the cascade form for the LTI system whose transfer function is:

$$H(z) = \sum_{k=0}^{4} z^{-k} = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4}$$

To factorize $H(z)$, we use the MATLAB command `roots([1 1 1 1 1])` to solve for the roots:

```
 0.3090 + 0.9511i
 0.3090 - 0.9511i
-0.8090 + 0.5878i
-0.8090 - 0.5878i
```

Hence $H(z)$ can be factorized as

$$\begin{aligned} H(z) = & \left(1-[0.309+j0.9511]z^{-1}\right)\left(1-[0.309-j0.9511]z^{-1}\right) \times \\ & \left(1-[-0.809+j0.5878]z^{-1}\right)\left(1-[-0.809-j0.5878]z^{-1}\right) \end{aligned}$$

Although it can be realized with first-order sections, complex coefficients are needed, which implies higher computational cost. To guarantee real-valued coefficients, we group the sections of complex conjugates together:

$$\begin{aligned} H(z) &= \left(1 - 0.618z^{-1} + z^{-2}\right)\left(1 + 1.618z^{-1} + z^{-2}\right) \\ &= \left(1 + 1.618z^{-1} + z^{-2}\right)\left(1 - 0.618z^{-1} + z^{-2}\right) \end{aligned}$$

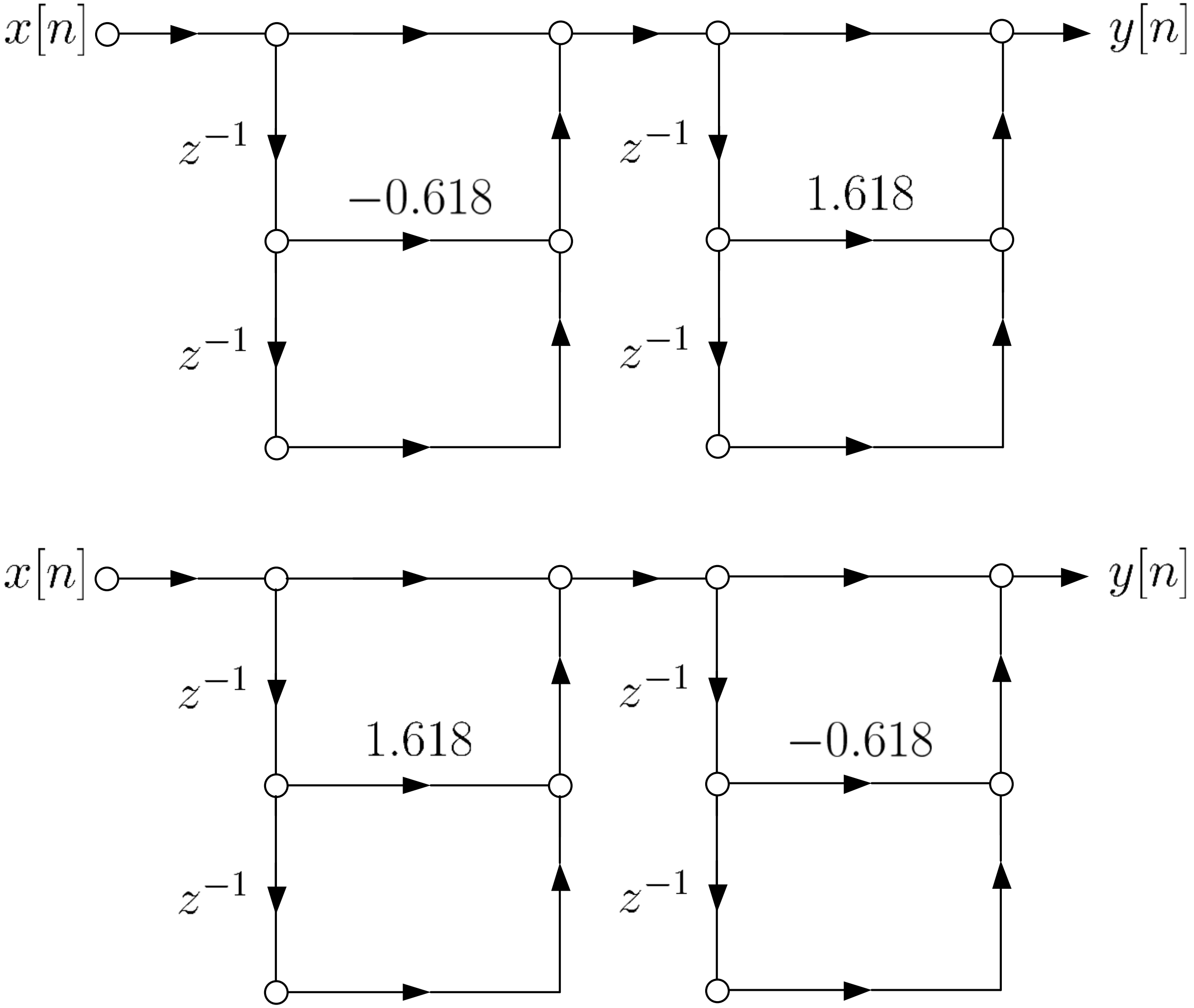


Fig.9.8: Two possible cascade forms for FIR filter

Structures for IIR Filter

When there is at least one pole in $H(z)$, it corresponds to an IIR filter. The corresponding transfer function is thus:

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} \tag{9.17}$$

That is, IIR filter is the general form of any discrete-time LTI system.

1. Direct Form

It realizes (9.4) in an explicit manner via decomposing it into a pair of difference equations:

$$v[n] = \sum_{k=0}^{M} b_k x[n-k] \tag{9.18}$$

and

$$y[n] = -\sum_{k=1}^{N} a_k y[n-k] + v[n] \tag{9.19}$$

The direct form can also be obtained by decomposing $H(z)$ into two transfer functions as:

$$H(z) = H_1(z) \cdot H_2(z) \tag{9.20}$$

where

$$H_1(z) = \sum_{k=0}^{M} b_k z^{-k} \tag{9.21}$$

and

$$H_2(z) = \frac{1}{1 + \sum_{k=1}^{N} a_k z^{-k}} \tag{9.22}$$

In the z transform domain, we have:

$$V(z) = X(z)H_1(z) = X(z) \sum_{k=0}^{M} b_k z^{-k} \tag{9.23}$$

and

$$Y(z) = V(z)H_2(z) = \frac{V(z)}{1 + \sum_{k=1}^{N} a_k z^{-k}} \tag{9.24}$$

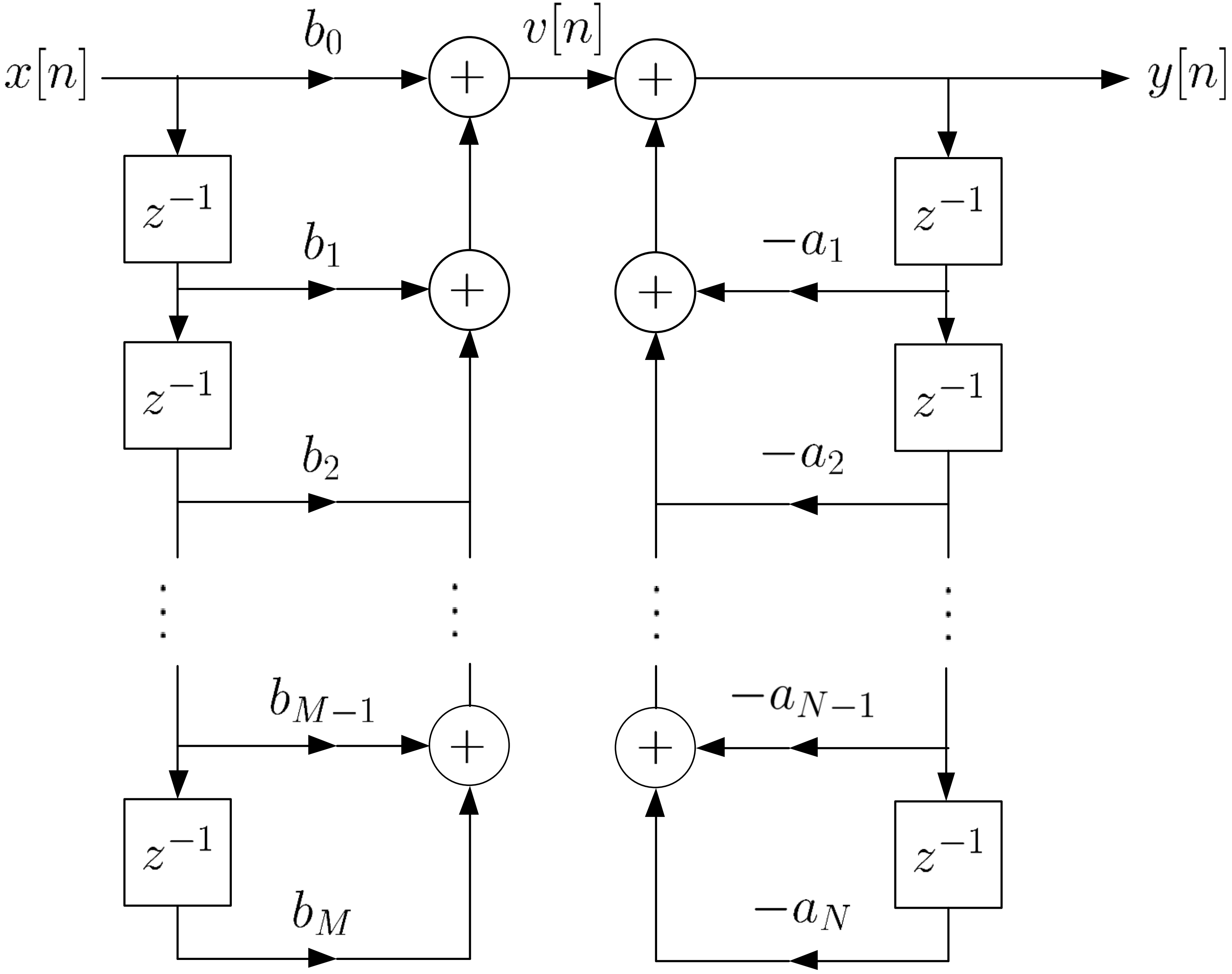


Fig.9.9: Direct form of IIR filter

The direct form implementation needs $(M+N)$ memory locations, $(M+N+1)$ multiplications and $(M+N)$ additions, for computing each output value of $y[n]$.

2. Canonic Form

On the other hand, we can first pass $x[n]$ through the filter $H_2(z)$ to produce an intermediate signal $w[n]$. The $w[n]$ is then passed through the system $H_1(z)$ to give $y[n]$:

$$W(z) = X(z)H_2(z) = \frac{X(z)}{1 + \sum_{k=1}^{N} a_k z^{-k}} \tag{9.25}$$

and

$$Y(z) = W(z)H_1(z) = W(z)\sum_{k=0}^{M} b_k z^{-k} \tag{9.26}$$

Applying inverse z transform, we get:

$$w[n] = -\sum_{k=1}^{N} a_k w[n-k] + x[n] \tag{9.27}$$

and

$$y[n] = \sum_{k=0}^{M} b_k w[n-k] \tag{9.28}$$

which can be considered as an alternative direct form.

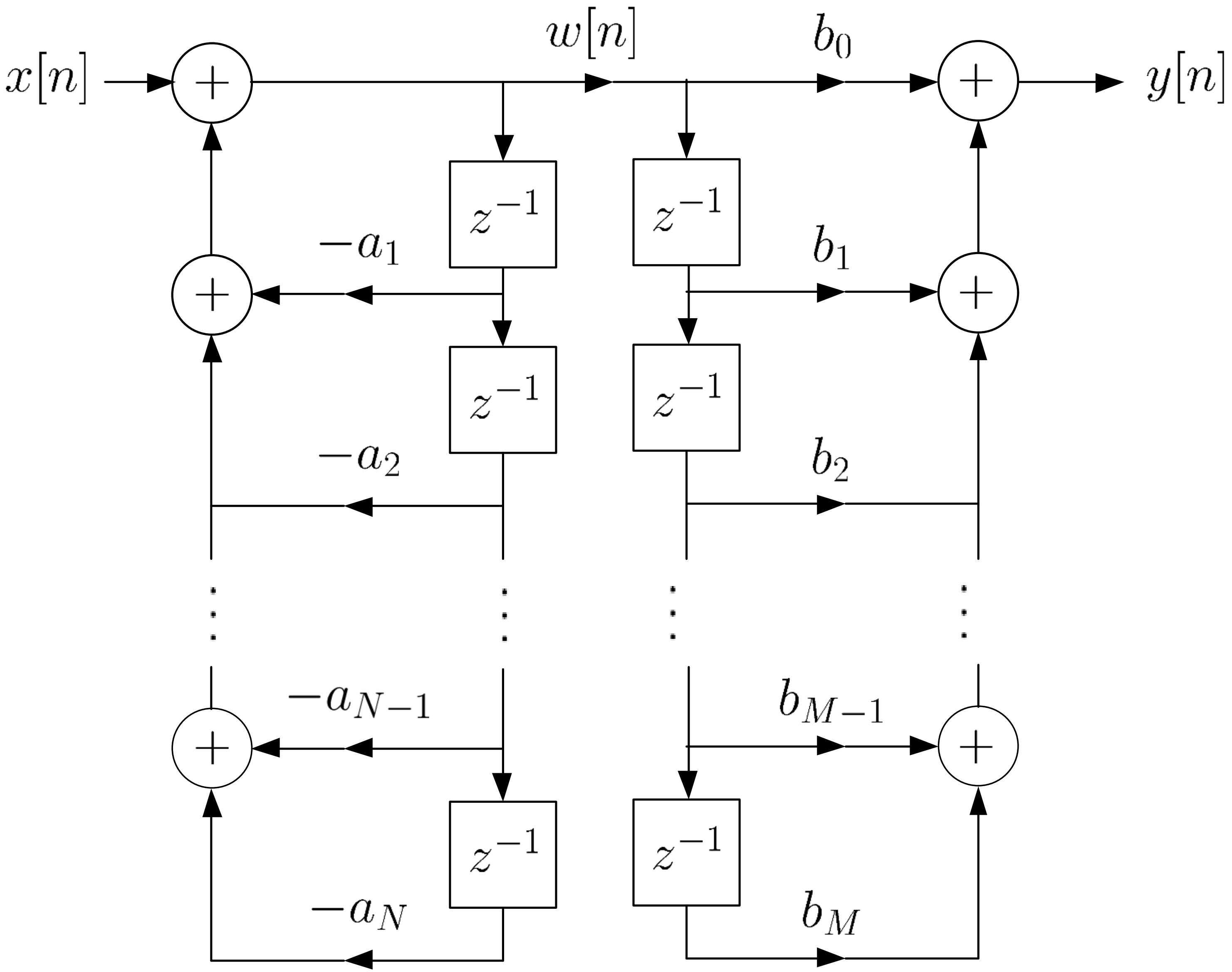


Fig.9.10: Alternative direct form of IIR filter

Assume $M = N$. Since the same signals $w[n], w[n-1], \cdots, w[n-N]$, are stored in the two chains of storage elements, they can be combined to reduce the memory requirement.

In general, the minimum number of delay elements required is $\max(M, N)$.

It is called canonic form because this implementation involves the minimum number of storages.

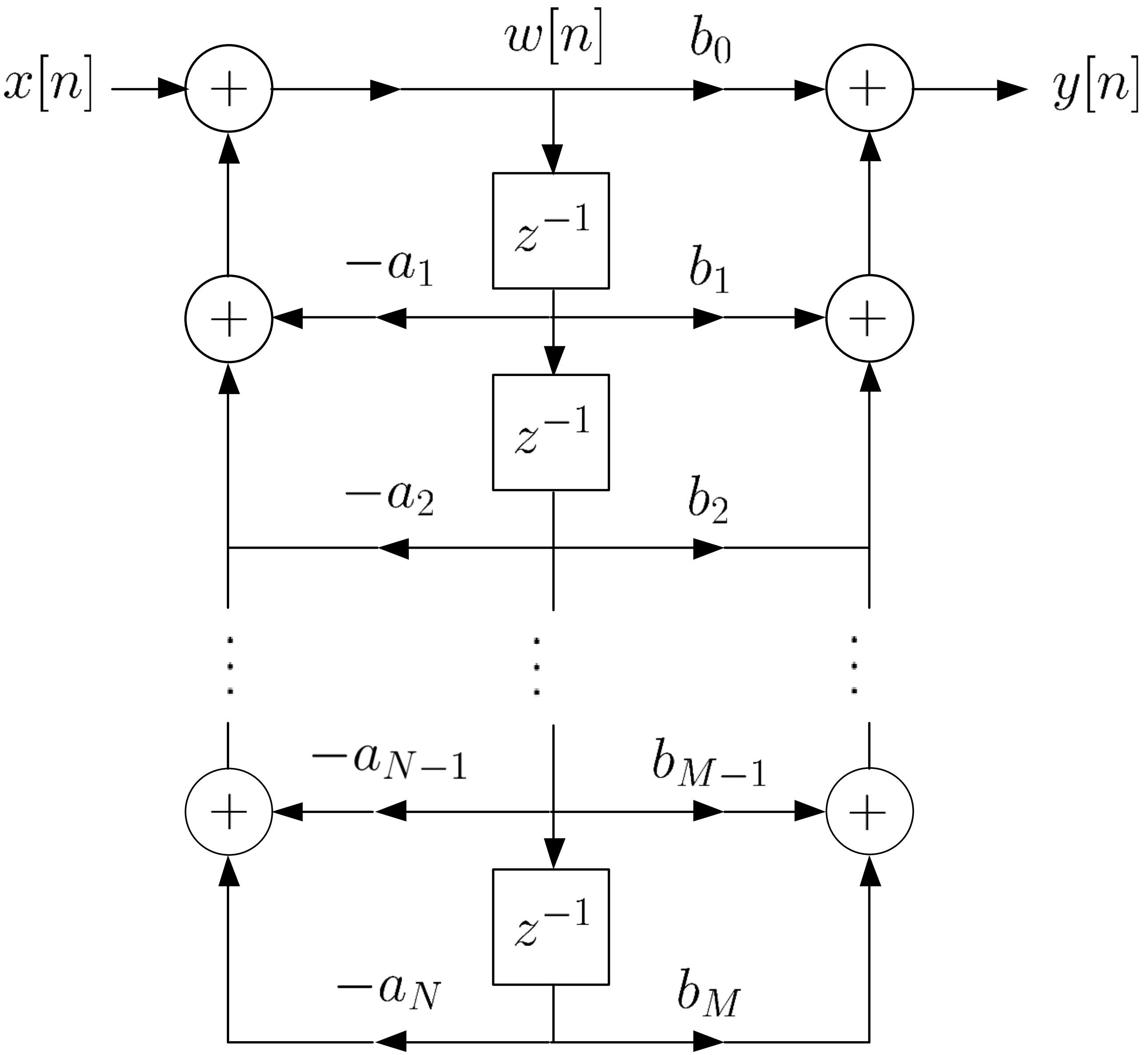


Fig.9.11: Canonic form of IIR filter

Example 9.3

Draw the block diagrams using the direct and canonic forms for the LTI system whose transfer function is:

$$H(z) = \frac{1 - 3z^{-1} + 2z^{-2}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$

According to (9.18)-(9.19):

$$\begin{gathered} v[n] = x[n] - 3x[n-1] + 2x[n-2] \\ y[n] = -0.3y[n-1] + 0.1y[n-2] + v[n] \end{gathered}$$

Based on (9.27)-(9.28):

$$\begin{gathered} w[n] = -0.3w[n-1] + 0.1w[n-2] + x[n] \\ y[n] = w[n] - 3w[n-1] + 2w[n-2] \end{gathered}$$

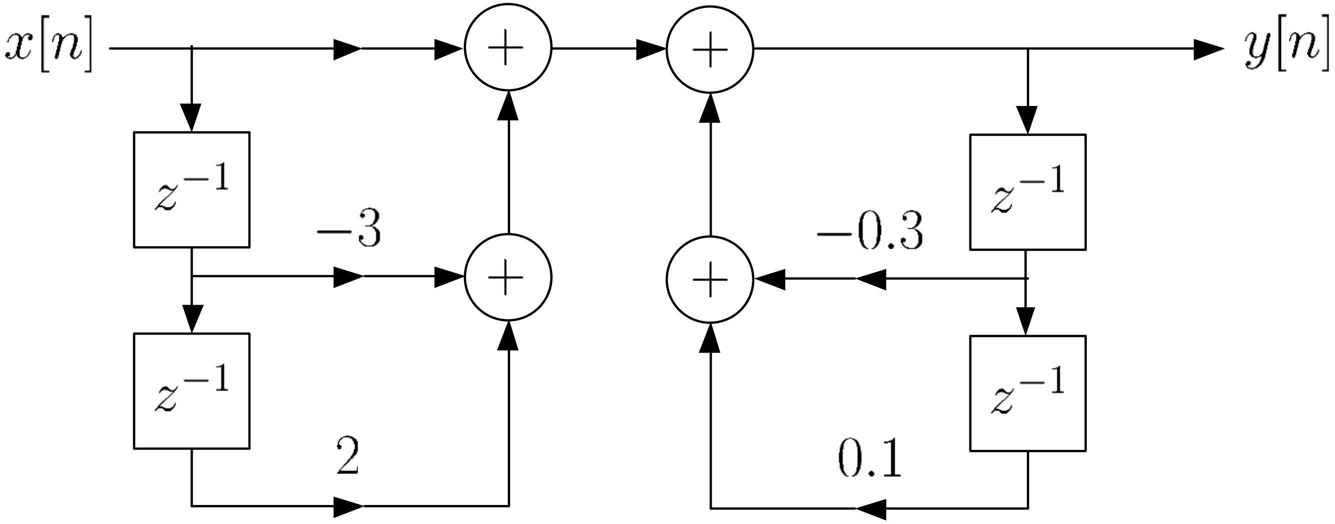


Fig.9.12: Direct form of second-order IIR filter

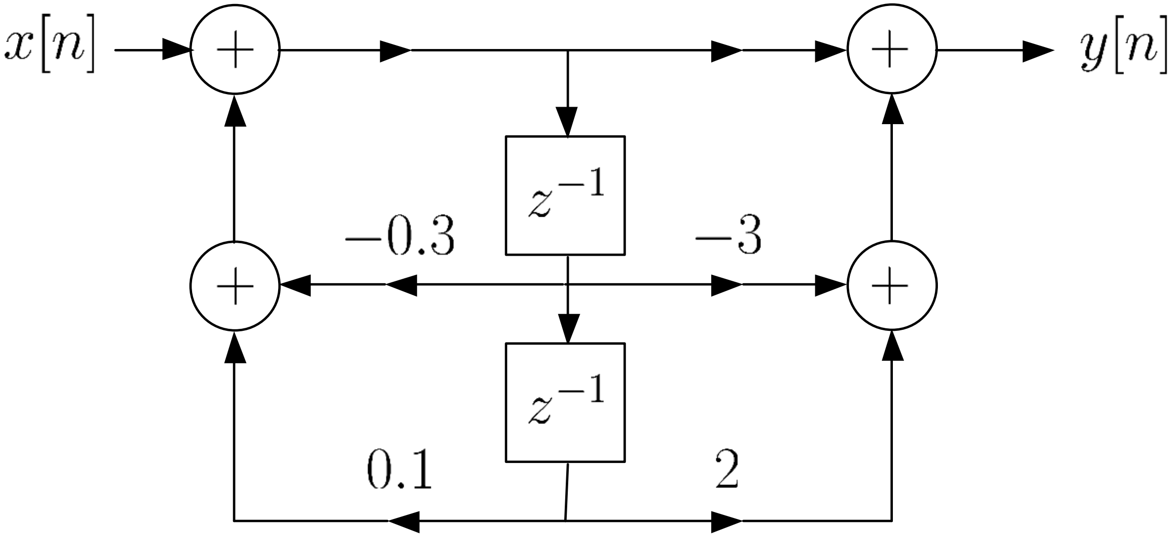


Fig.9.13: Canonic form of second-order IIR filter

3. Cascade Form

We factorize the numerator and denominator polynomials in terms of second-order polynomial system functions as:

$$H(z)=\frac{\displaystyle\sum_{k=0}^{M} b_k z^{-k}}{1+\displaystyle\sum_{k=1}^{N} a_k z^{-k}}=\prod_{k=1}^{N_c}\frac{\beta_{0k}+\beta_{1k}z^{-1}+\beta_{2k}z^{-2}}{1+\alpha_{1k}z^{-1}+\alpha_{2k}z^{-2}} \tag{9.29}$$

Without loss of generality, it is assumed that $N \geq M$ so that $N_c=\lfloor (N+1)/2 \rfloor$. Note that when M or N is odd, one of the $\{\beta_{2k}\}$ or $\{\alpha_{2k}\}$ will be zero.

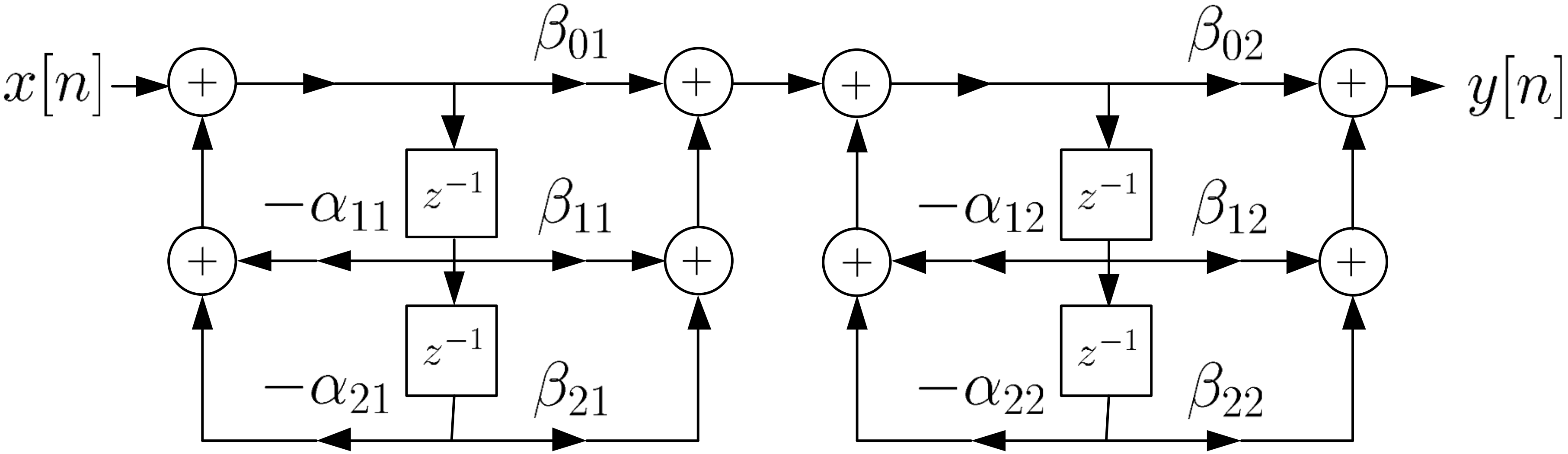
Each second-order subsystem

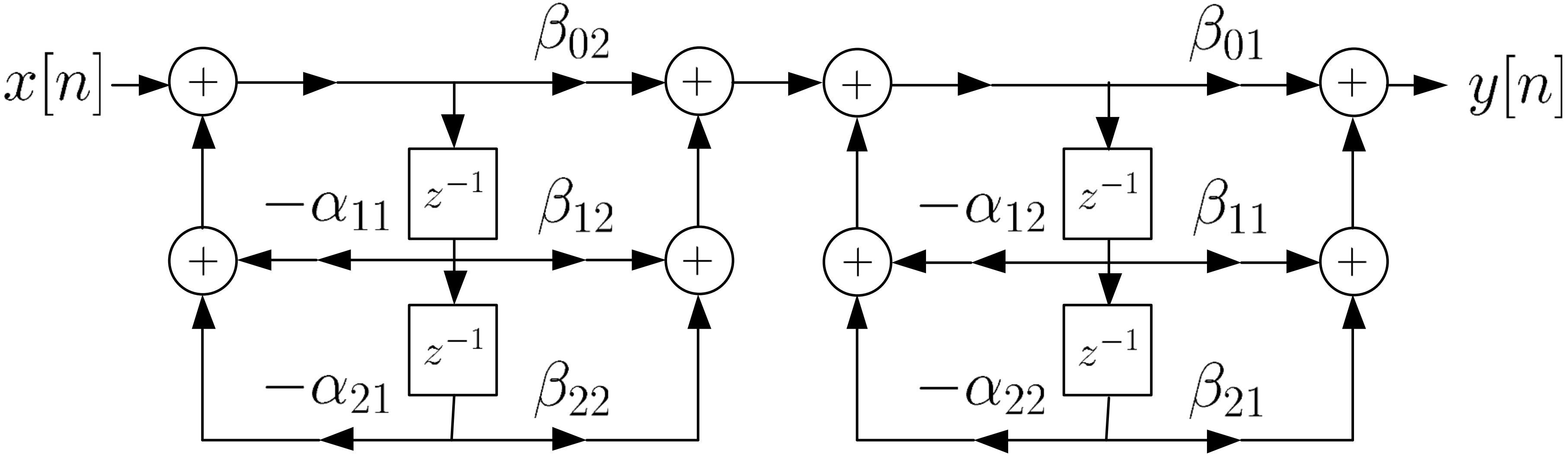
$$\frac{\beta_{0k} + \beta_{1k}z^{-1} + \beta_{2k}z^{-2}}{1 + \alpha_{1k}z^{-1} + \alpha_{2k}z^{-2}} \tag{9.30}$$

can be realized in either the direct or canonic form. Nevertheless, the canonic form is preferred because it requires the minimum number of delay elements.

In IIR filter implementation, we can group the numerator and denominator of (9.30) in different ways, leading to different pole and zero combinations in each of the second-order sections.

For example, there are four possible cascade realizations for a fourth-order IIR filter with $M = N = 4$.


$x[n]$
β_{01}
β_{02}
$y[n]$
z^{-1}
$-\alpha_{11}$
β_{11}
$-\alpha_{12}$
β_{12}
$-\alpha_{21}$
β_{21}
$-\alpha_{22}$
β_{22}


$x[n]$
β_{02}
β_{01}
$y[n]$
z^{-1}
$-\alpha_{11}$
β_{12}
$-\alpha_{12}$
β_{11}
$-\alpha_{21}$
β_{22}
$-\alpha_{22}$
β_{21}

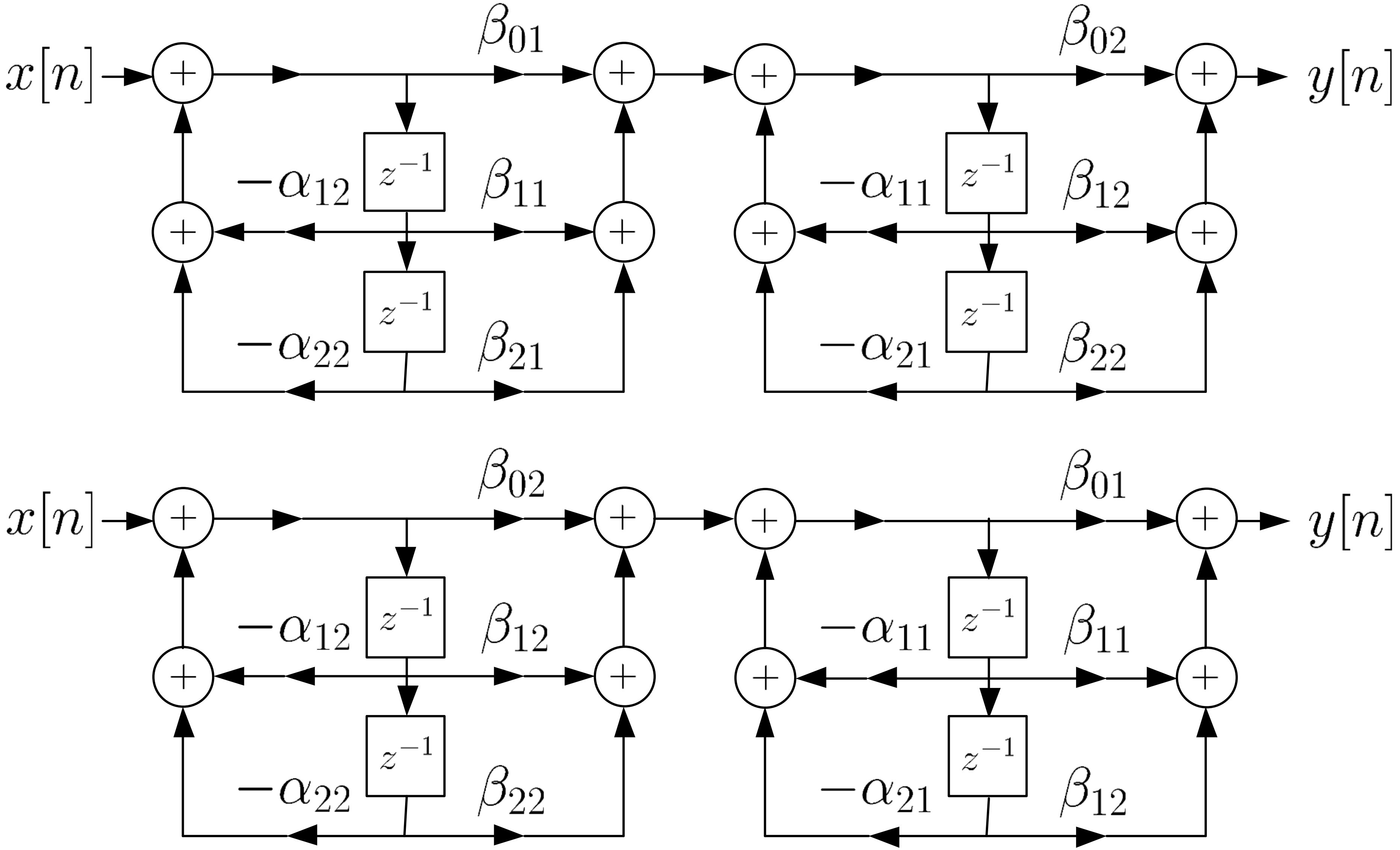


Fig.9.14: 4 possible cascade realizations for 4th-order IIR filter

To save the computational complexity, we express (9.29) as

$$H(z) = G \prod_{k=1}^{N_c} \frac{1+\beta'_{1k}z^{-1}+\beta'_{2k}z^{-2}}{1+\alpha_{1k}z^{-1}+\alpha_{2k}z^{-2}} \qquad (9.31)$$

where $G = \beta_{01}\beta_{02}\cdots\beta_{0N_c}$, $\beta'_{1k} = \beta_{1k}/\beta_{0k}$ and $\beta'_{2k} = \beta_{2k}/\beta_{0k}$, $k = 1, 2, \cdots, N_c$.

Assuming that N is even with $N = M$, the cascade implementation of (9.31) needs N or $2N$ delay elements, $(2N+1)$ multiplications and $2N$ additions, for computing each $y[n]$. That is, its memory and computational requirements are equal to those of the direct form.

Example 9.4

Draw the signal flow graph using the cascade form with first-order sections for the LTI system whose transfer function is:

$$H(z) = \frac{1 - 3z^{-1} + 2z^{-2}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$

For each first-order section, canonic form is assumed.

Solving the quadratic equations of the numerator and denominator polynomials, we can factorize $H(z)$ as:

$$H(z) = \frac{\left(1 - 2z^{-1}\right)\left(1 - z^{-1}\right)}{\left(1 + 0.5z^{-1}\right)\left(1 - 0.2z^{-1}\right)}$$

There are four possible cascade forms for $H(z)$:

$$\begin{aligned}
H(z) &= \frac{1-2z^{-1}}{1+0.5z^{-1}} \cdot \frac{1-z^{-1}}{1-0.2z^{-1}} \\
&= \frac{1-z^{-1}}{1+0.5z^{-1}} \cdot \frac{1-2z^{-1}}{1-0.2z^{-1}} \\
&= \frac{1-2z^{-1}}{1-0.2z^{-1}} \cdot \frac{1-z^{-1}}{1+0.5z^{-1}} \\
&= \frac{1-z^{-1}}{1-0.2z^{-1}} \cdot \frac{1-2z^{-1}}{1+0.5z^{-1}}
\end{aligned}$$

Note that although all four realizations are equivalent for infinite precision, they may differ in actual implementation when finite-precision numbers are employed.

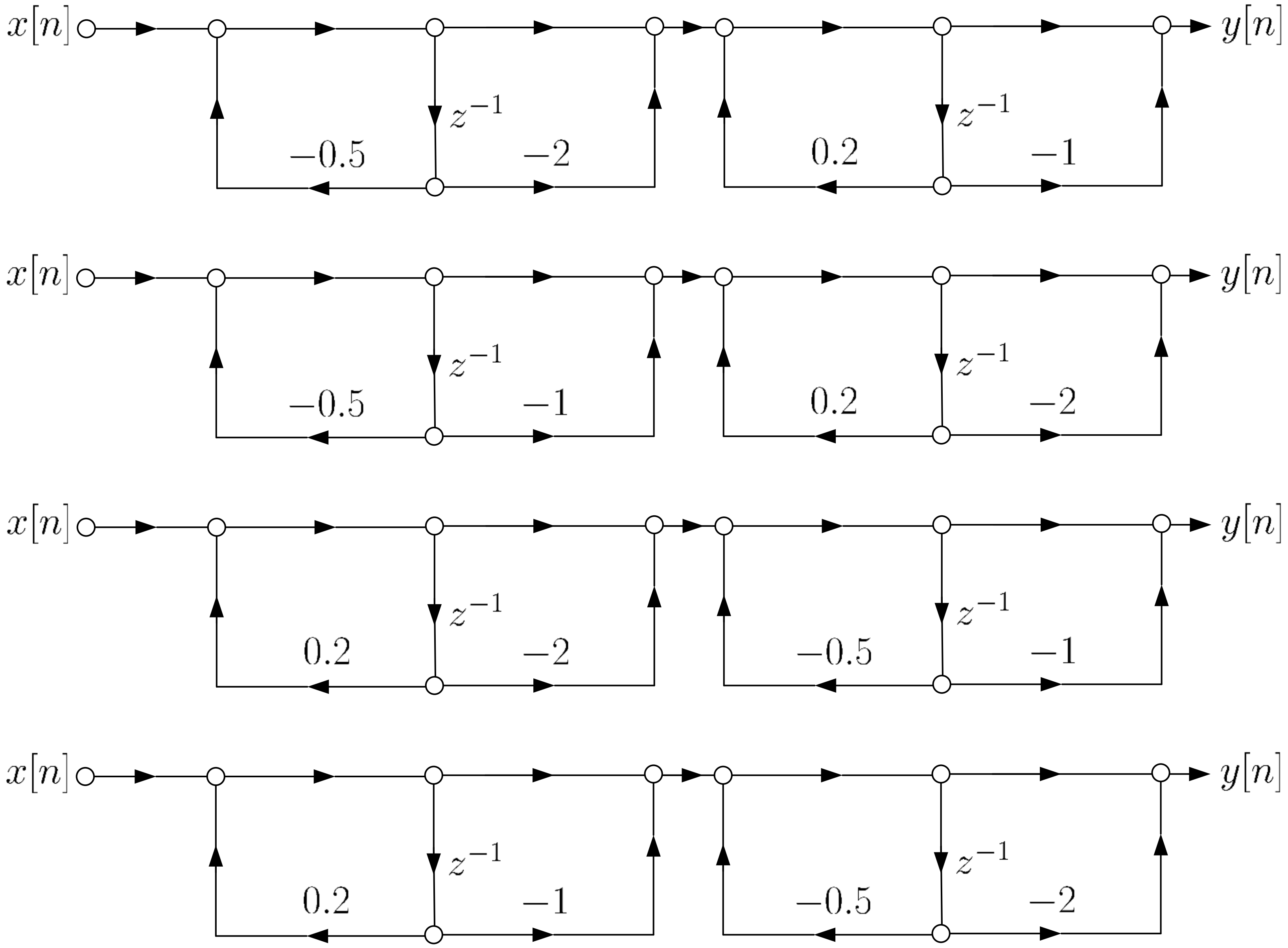


Fig.9.15: 4 possible cascade realizations with 1st-order sections

Example 9.5

Consider a LTI system whose transfer function $H(z)$ is:

$$H(z)=\frac{\left(1-\frac{13}{10}z^{-1}+\frac{2}{5}z^{-2}\right)\left(1-\frac{17}{12}z^{-1}+\frac{1}{2}z^{-2}\right)}{\left(1-3z^{-1}+2z^{-2}\right)\left(1-7z^{-1}+12z^{-2}\right)}$$

Find the number of possible combinations of cascade form with second-order sections. Determine if the system is stable or not.

Further factorizing $H(z)$ yields:

$$H(z)=\frac{\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{2}{3}z^{-1}\right)\left(1-\frac{3}{4}z^{-1}\right)\left(1-\frac{4}{5}z^{-1}\right)}{\left(1-z^{-1}\right)\left(1-2z^{-1}\right)\left(1-3z^{-1}\right)\left(1-4z^{-1}\right)}$$

For the numerator polynomials, there are 3 grouping possibilities in terms of second-order sections as follows:

$$\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{2}{3}z^{-1}\right), \qquad \left(1-\frac{3}{4}z^{-1}\right)\left(1-\frac{4}{5}z^{-1}\right)$$
$$\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{3}{4}z^{-1}\right), \qquad \left(1-\frac{2}{3}z^{-1}\right)\left(1-\frac{4}{5}z^{-1}\right)$$
$$\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{4}{5}z^{-1}\right), \qquad \left(1-\frac{3}{4}z^{-1}\right)\left(1-\frac{2}{3}z^{-1}\right)$$

There are also 3 grouping possibilities for the denominator:

$$\left(1-z^{-1}\right)\left(1-2z^{-1}\right), \qquad \left(1-3z^{-1}\right)\left(1-4z^{-1}\right)$$
$$\left(1-z^{-1}\right)\left(1-3z^{-1}\right), \qquad \left(1-2z^{-1}\right)\left(1-4z^{-1}\right)$$
$$\left(1-z^{-1}\right)\left(1-4z^{-1}\right), \qquad \left(1-3z^{-1}\right)\left(1-2z^{-1}\right)$$

Each fourth-order IIR filter can be realized in four possible cascade forms.

As a result, the number of possible cascade form combinations with second-order sections for $H(z)$ is $3 \cdot 3 \cdot 4 = 36$.

When implementing a filter, causality is always assumed because it is impossible to realize a noncausal system where the output depends on future input. It is clear that the region of convergence (ROC) for causal $H(z)$ is $|z| > 4$. Since the ROC does not include the unit circle, the system is not stable.

In summary, a causal system is stable when all poles are inside the unit circle.

4. Parallel Form

The idea of the parallel form is similar to the partial fraction expansion of z transform:

$$H(z)=\sum_{l=0}^{M-N} B_l z^{-l}+\sum_{k=1}^{N_c}\frac{\gamma_{0k}+\gamma_{1k}z^{-1}}{1+\alpha_{1k}z^{-1}+\alpha_{2k}z^{-2}} \qquad (9.32)$$

where $N_c=\lfloor (N+1)/2\rfloor$. But now we use second-order sections in order to ensure all $\{\gamma_{0k}\}$, $\{\gamma_{1k}\}$, $\{\alpha_{1k}\}$ and $\{\alpha_{2k}\}$ are real.

Note that when $M<N$, the first summation term in (9.32) will not be included.

Example 9.6

Draw the block diagram using parallel form for a LTI system whose transfer function is:

$$H(z) = \frac{1 - 3z^{-1} + 2z^{-2}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$

Following the long division as in Example 5.9, we obtain:

$$H(z) = -20 + \frac{21 + 3z^{-1}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$

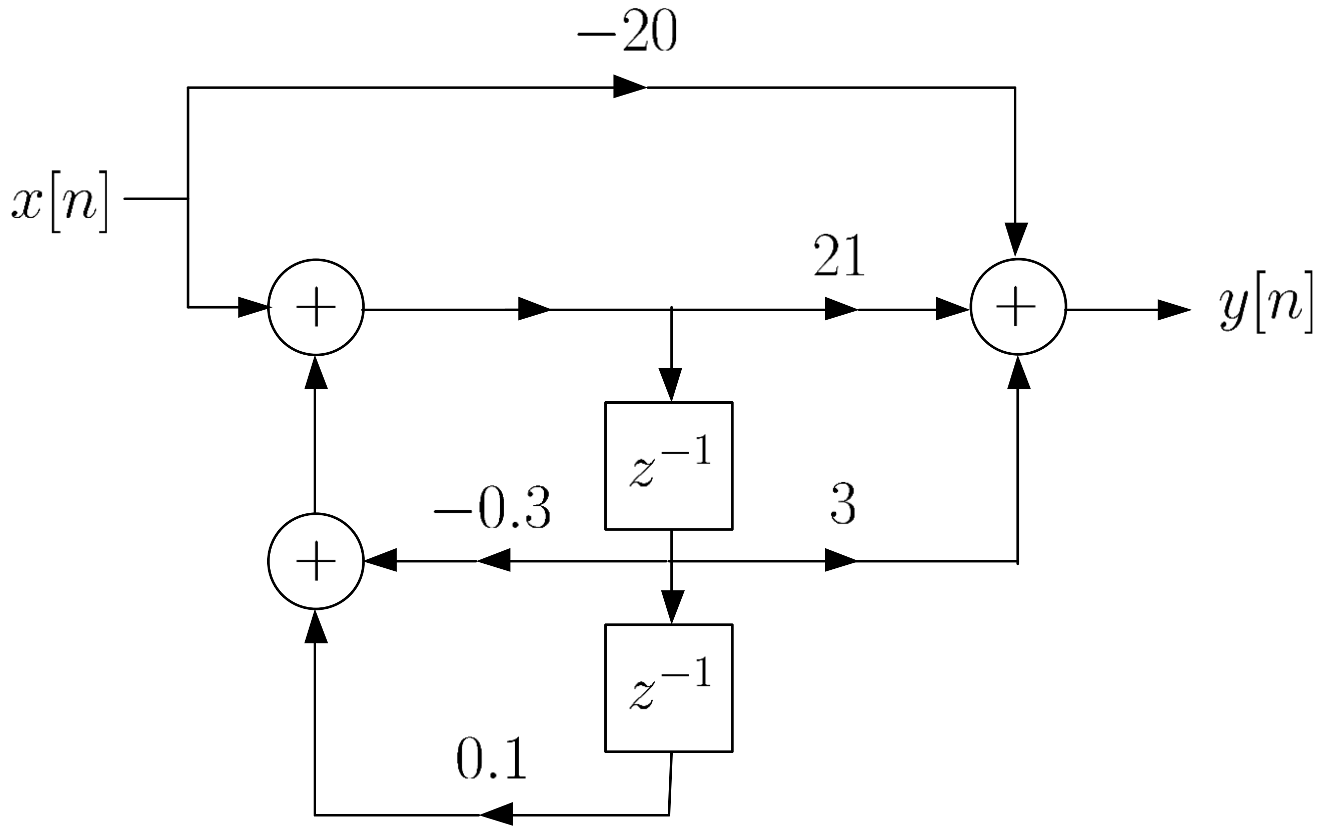


Fig.9.16: Parallel form with second-order section

As the poles of $H(z)$ are real, we can also express $H(z)$ in terms of first-order sections as:

$$H(z) = -20 + \frac{\frac{75}{7}}{1+0.5z^{-1}} + \frac{\frac{72}{7}}{1-0.2z^{-1}}$$

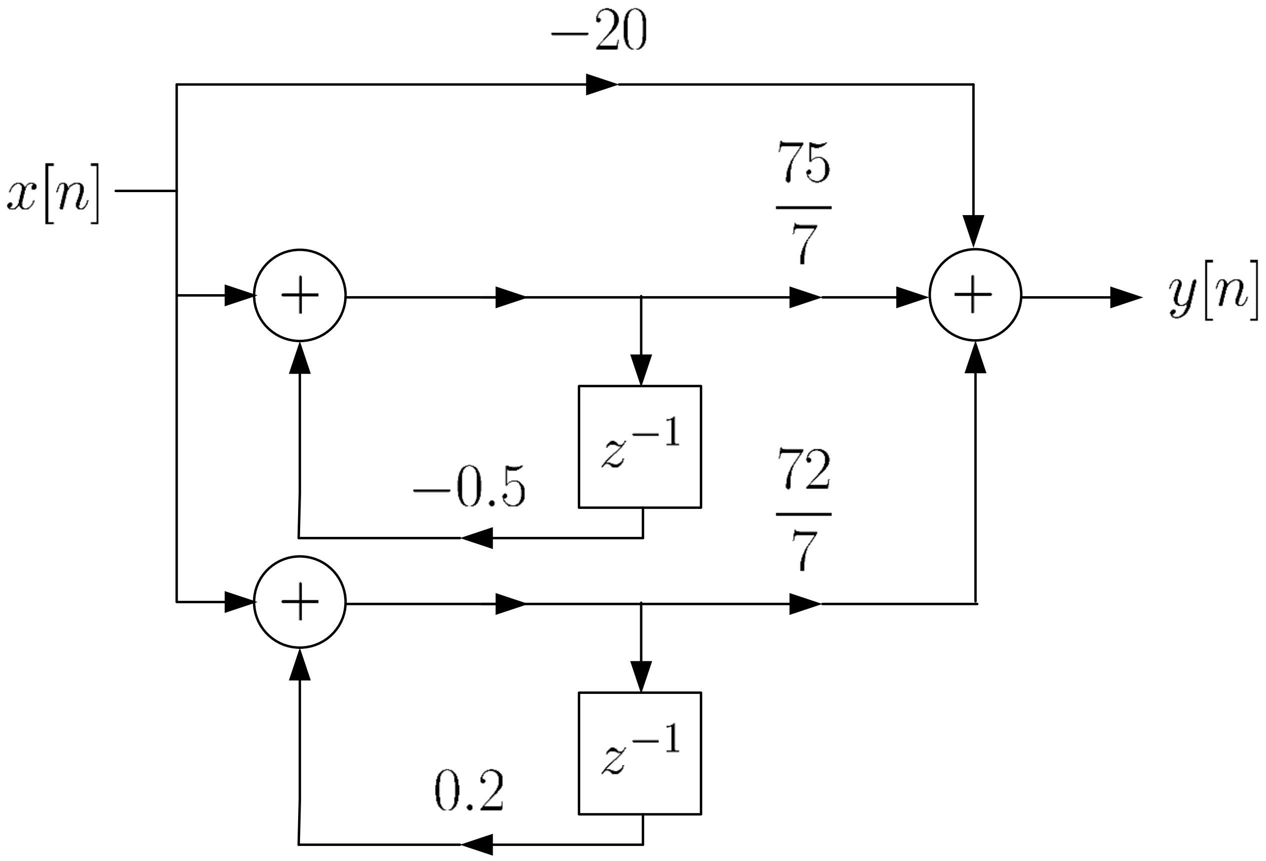


Fig.9.17: Parallel form with first-order sections

Example 9.7

Determine the transfer function $H(z)$ and the difference equation which relates $x[n]$ and $y[n]$ for

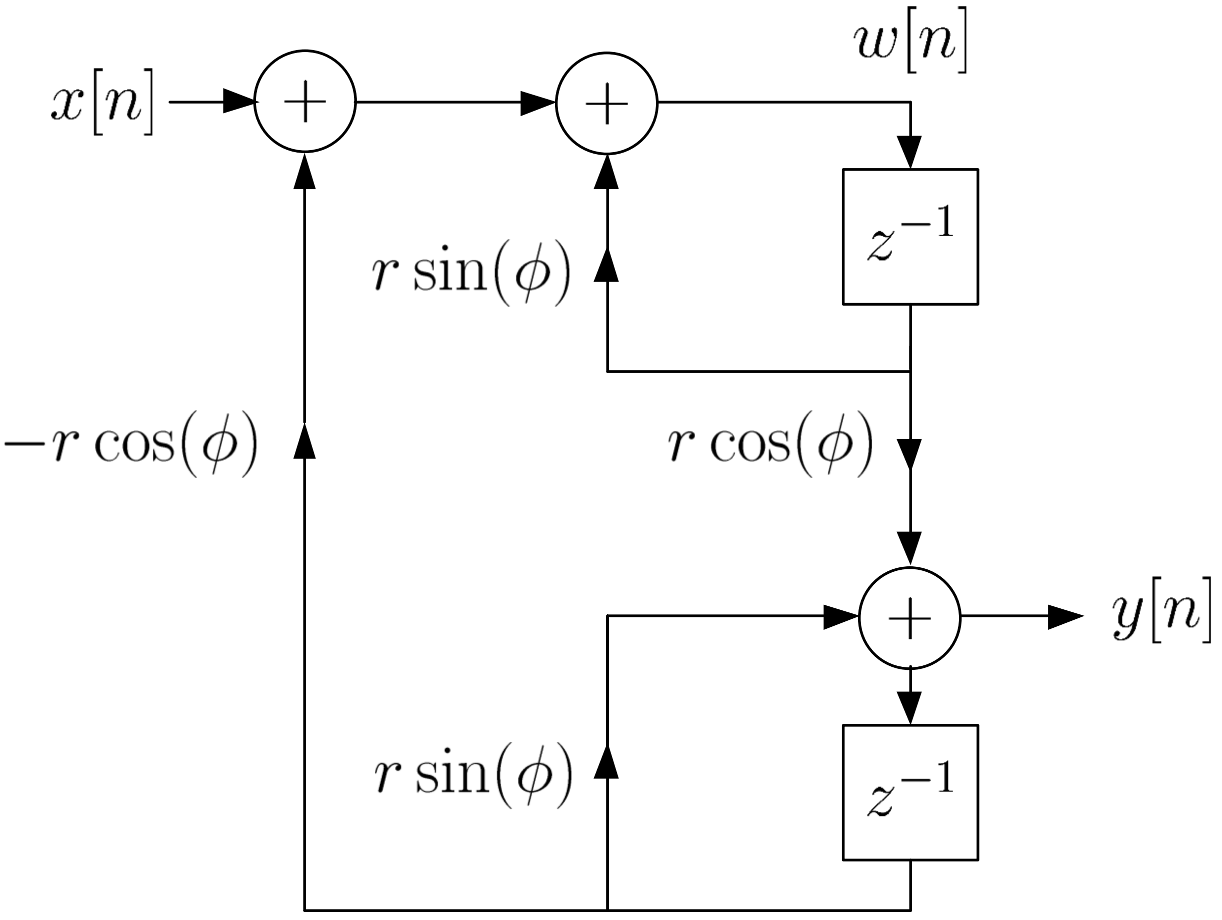


Fig.9.18: LTI system parameterized by r and ϕ

We first introduce an intermediate sequence $w[n]$ to relate $x[n]$ and $y[n]$. Then we can establish:

$$w[n] = w[n-1]\cdot r\sin(\phi) - y[n-1]\cdot r\cos(\phi) + x[n]$$

and

$$w[n-1]\cdot r\cos(\phi) + y[n-1]\cdot r\sin(\phi) = y[n]$$

Applying z transform yields:

$$W(z) = W(z)z^{-1}r\sin(\phi) - Y(z)z^{-1}r\cos(\phi) + X(z)$$

and

$$W(z)z^{-1}r\cos(\phi) + Y(z)z^{-1}r\sin(\phi) = Y(z)$$

From the second equation, we have:

$$W(z) = \frac{1 - z^{-1}r\sin(\phi)}{z^{-1}r\cos(\phi)}Y(z)$$

Substituting it into the first equation, we finally have:

$$\frac{1 - z^{-1} r \sin(\phi)}{z^{-1} r \cos(\phi)} Y(z) \left(1 - z^{-1} r \sin(\phi)\right) = -Y(z) z^{-1} r \cos(\phi) + X(z)$$
$$\Rightarrow Y(z) \left(1 - 2z^{-1} r \sin(\phi) + z^{-2} r^2\right) = X(z) z^{-1} r \cos(\phi)$$
$$\Rightarrow H(z) = \frac{Y(z)}{X(z)} = \frac{z^{-1} r \cos(\phi)}{1 - 2z^{-1} r \sin(\phi) + z^{-2} r^2}$$

Taking the inverse z transform gives:

$$y[n] - 2r \sin(\phi) y[n-1] + r^2 y[n-2] = r \cos(\phi) x[n-1]$$

Comparison of Different Structures

The major factors that affect our choice of a specific realization are computational complexity, memory requirement, and finite word-length effects. Assuming that M is even with $M = N$:

Structure	Multiplication	Addition	Register
Direct form	$M+1$	M	M
Cascade form	$M+1$	M	M

Table 9.1: FIR filter structure comparison

Structure	Multiplication	Addition	Register
Direct form	$2M+1$	$2M$	$2M$
Canonic form	$2M+1$	$2M$	M
Cascade form	$2M+1$	$2M$	M
Parallel form	$2M+1$	$3M/2+1$	M

Table 9.2: IIR filter structure comparison

Computations of the direct form can be reduced if the FIR filter coefficients are symmetric or anti-symmetric.

When the filter coefficients are expressed using infinite precision numbers, all realizations are same. However, in practice, they are processed in registers which have finite word-lengths. In the presence of quantization errors, the cascade and parallel realizations are more robust than the direct and canonic forms, that is, they have frequency responses closer to the desired responses.

FIR filters are less sensitive than IIR filters to finite word-length effects.

For a feasible system, it should be causal and stable.

In cascade and parallel realizations of IIR filters, system stability can be easily monitored by checking pole locations.

Chapter 10: FIR Filter Design

Chapter Intended Learning Outcomes:

(i) Understanding of the characteristics of linear-phase finite impulse response (FIR) filters

(ii) Ability to design linear-phase FIR filters according to predefined specifications using the window and frequency sampling methods

Steps in Digital Filter Design

1. Specification Determination

The first step is to obtain the filter specifications or requirements which are determined by the applications.

Suppose the requirement is a lowpass filter with cutoff frequency of ω_c to achieve a task of noise reduction

The filter specification can be described by discrete-time Fourier transform (DTFT) $H(e^{j\omega})$:

$$H(e^{j\omega}) = \begin{cases} 1, & |\omega| < \omega_c \\ 0, & -\pi < \omega < -\omega_c, \quad \omega_c < \omega < \pi \end{cases} \tag{10.1}$$

which specifies both the magnitude and phase

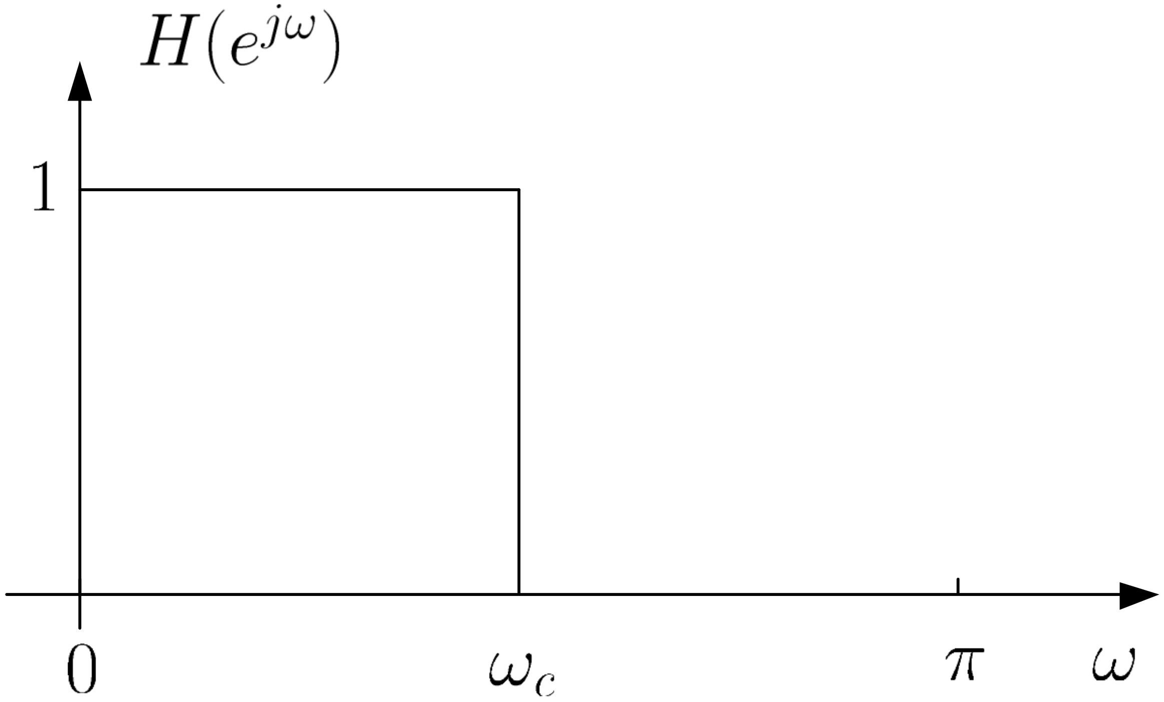


Fig.10.1: Ideal lowpass filter

- unity gain for the whole range of $\omega \in (0, \omega_c)$
- complete suppression for $\omega \in (\omega_c, \pi)$
- step change in frequency response at $\omega = \omega_c$

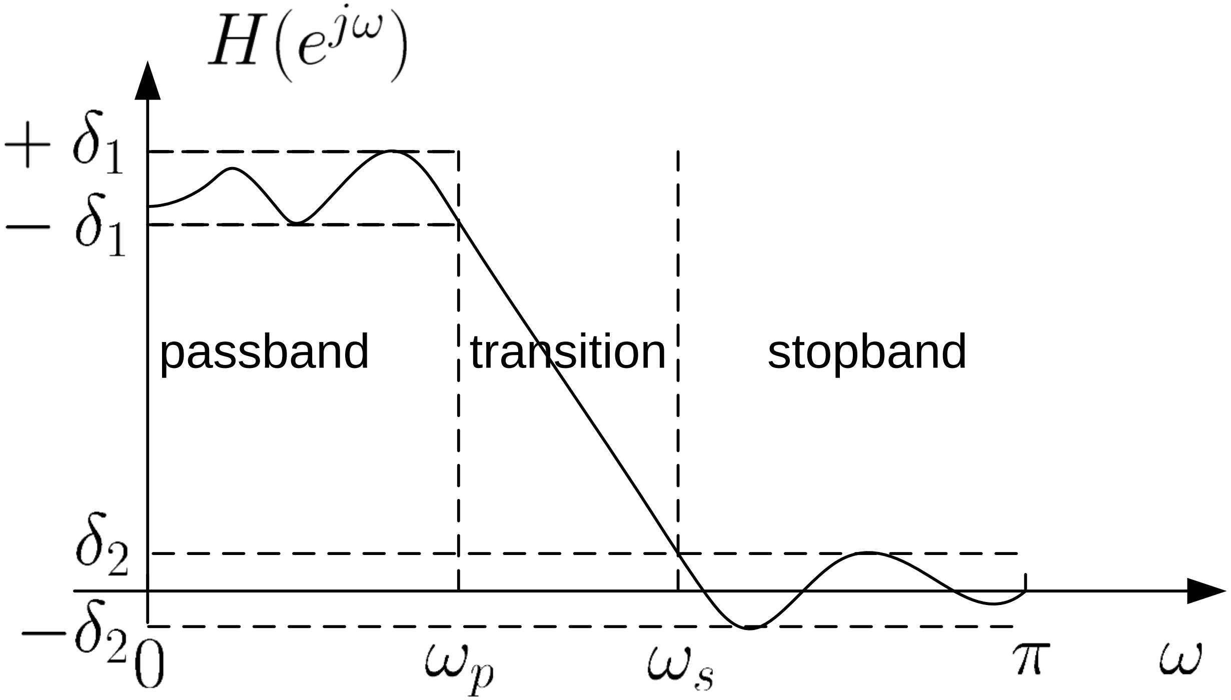


Fig.10.2: Practical $H(e^{j\omega})$

- passband corresponds to $\omega \in (0, \omega_p)$ where ω_p is the passband frequency and δ_1 is the passband ripple or tolerance which is the maximum allowable deviation from unity in this band

- stopband corresponds to $\omega \in (\omega_s, \pi)$ where ω_s is the stopband frequency and δ_2 is the stopband ripple or tolerance which is the maximum allowable deviation from zero in this band

- transition band corresponds to $\omega_c \in (\omega_p, \omega_s)$ where there are no restrictions on $H(e^{j\omega})$ in this band

2. Filter Response Calculation

We then use digital signal processing techniques to obtain a filter description in terms of transfer function $H(z)$ or impulse response $h[n]$ that fulfills the given specifications

3. Implementation

When $H(z)$ or $h[n]$ are known, the filter can then be realized in hardware or software according to a given structure

Advantages of FIR Filter

Transfer function of a causal FIR filter $H(z)$ with length N is:

$$H(z) = \sum_{n=0}^{N-1} h[n] z^{-n} \qquad (10.2)$$

where $h[n]$ is the finite-duration impulse response

- Phase response can be exactly linear which results in computation reduction and zero phase distortion. Note that when there is phase distortion, different frequency components of a signal will undergo different delays in the filtering process

- It is always stable because $\sum_{n=0}^{N-1} |h[n]| < \infty$ for finite N

- Less sensitive to finite word-length effects

- Efficient implementation via digital signal processors with multiply-and-add (MAC) instruction
- Existence of optimality theorem for FIR filter design

Linear-Phase FIR Filter

The phase response of a linear-phase filter is a linear function of the frequency ω:

$$\angle H(e^{j\omega}) = \phi - \tau\omega, \quad -\pi < \omega < \pi \tag{10.3}$$

where ϕ equals 0 or $\pi/2$ and τ is a constant which is function of the filter length.

Note that (10.3) is slightly different from that in previous chapters where the phase response complements with the magnitude response $|H(e^{j\omega})|$.

Here, the phase response is related to $H(e^{j\omega})$ by

$$H(e^{j\omega}) = H_r(e^{j\omega})e^{j\angle H(e^{j\omega})} \tag{10.4}$$

where $H_r(e^{j\omega})$ is real and is called the amplitude response.

Comparing

$$H(e^{j\omega}) = |H(e^{j\omega})|e^{j\angle H(e^{j\omega})} \tag{10.5}$$

where $|H(e^{j\omega})|$ is nonnegative, we see that when $H_r(e^{j\omega})$ is positive, the phase responses in (10.4) and (10.5) are identical but there is a phase difference of π for a negative $H_r(e^{j\omega})$.

Example 10.1

Determine the amplitude response and the corresponding phase response for a FIR filter with impulse response $h[n]$:

$$h[n] = u[n] - u[n-N]$$

Plot the spectra for $N = 10$. Compare the results with the magnitude response and the corresponding phase response.

Using Example 6.2:

$$H(e^{j\omega}) = \frac{\sin(\omega N/2)}{\sin(\omega/2)} \cdot e^{-j\omega(N-1)/2}$$

Hence the amplitude response is

$$H_r(e^{j\omega}) = \frac{\sin(\omega N/2)}{\sin(\omega/2)}$$

and the corresponding phase response is

$$\angle H(e^{j\omega}) = -\frac{N-1}{2}\omega$$

On the other hand, the magnitude response and the corresponding phase response are:

$$|H(e^{j\omega})| = \left|\frac{\sin(\omega N/2)}{\sin(\omega/2)}\right|$$

and

$$\angle(H(e^{j\omega})) = -\frac{\omega(N-1)}{2} + \angle\left(\frac{\sin(\omega N/2)}{\sin(\omega/2)}\right)$$

The MATLAB program is provided as `ex10_1.m`.

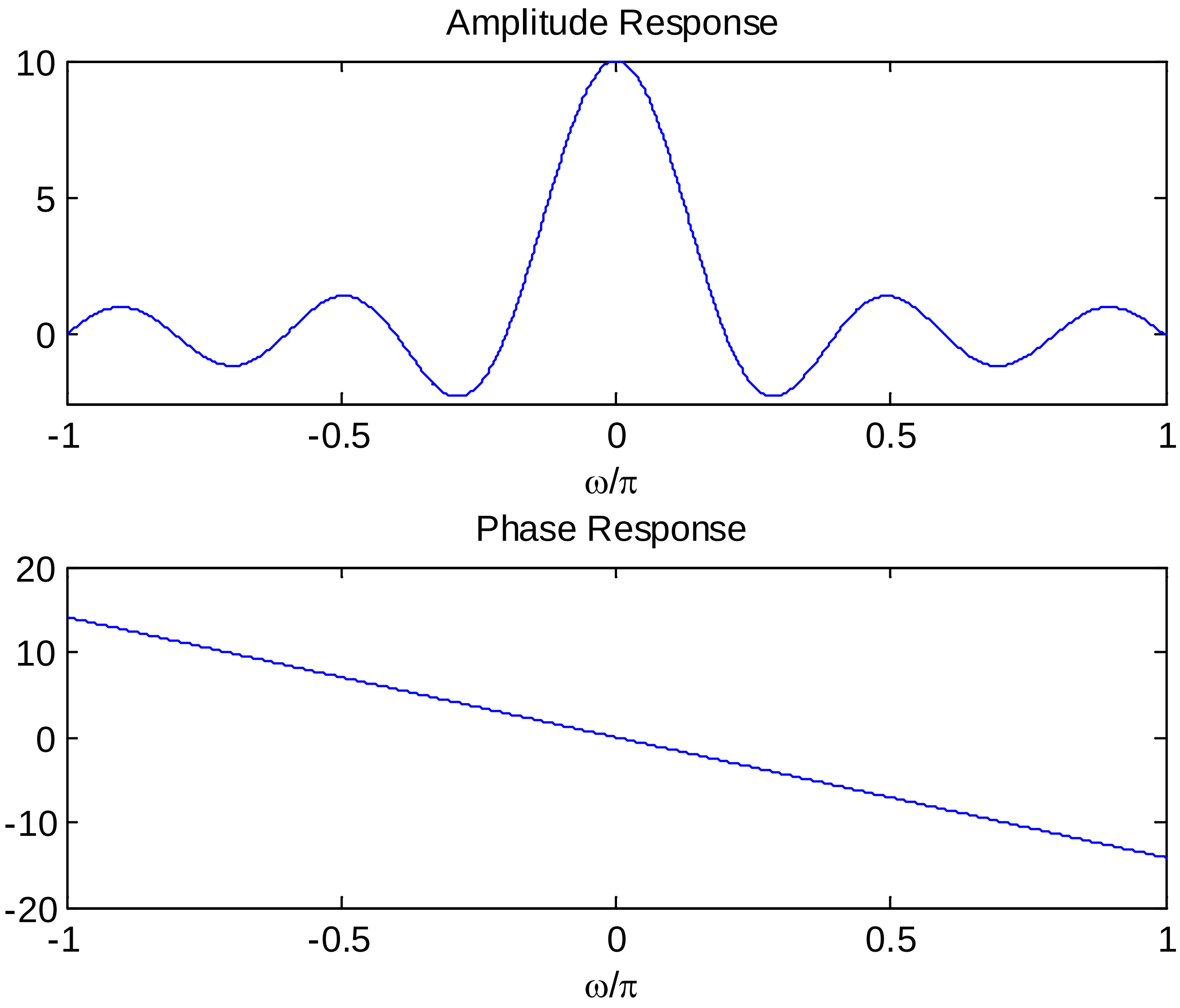


Fig.10.3: Amplitude and phase spectra

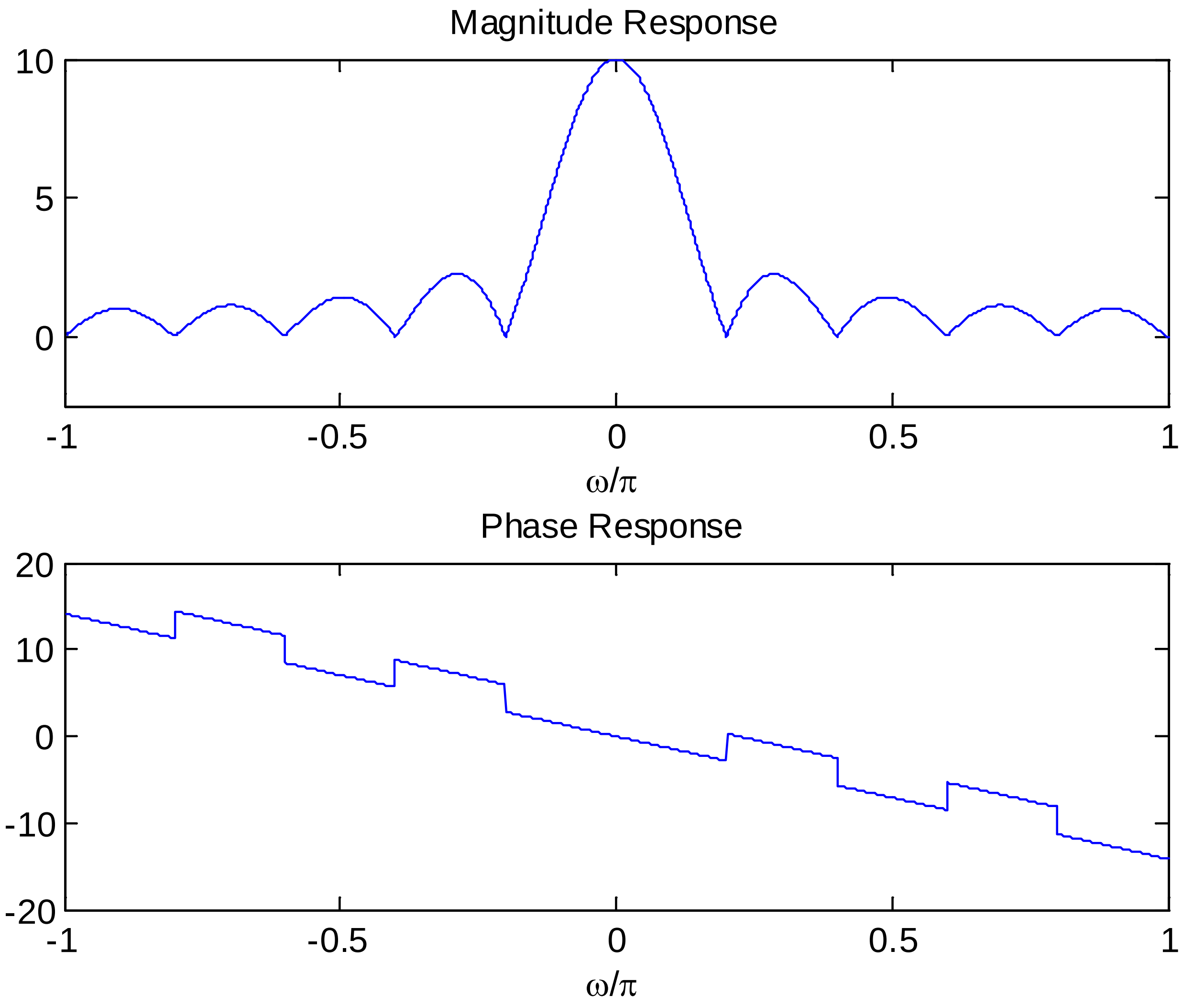


Fig.10.4: Magnitude and phase spectra

For a causal linear-phase FIR filter, the impulse response $h[n]$ is either symmetric or anti-symmetric:

$$h[n] = h[N-1-n], \quad 0 \leq n \leq N-1 \tag{10.6}$$

or

$$h[n] = -h[N-1-n], \quad 0 \leq n \leq N-1 \tag{10.7}$$

For example, when N is even in (10.7):

$$h[0] = -h[N-1],\ h[1] = -h[N-2],\ \cdots, h[N/2-1] = -h[N/2]$$

$$\begin{aligned} y[n] &= \sum_{k=0}^{N-1} h[k]x[n-k] \\ &= h[0]x[n] + h[1]x[n-1] + \cdots + h[N/2-1]x[n-N/2+1] \\ &\quad +h[N/2]x[n-N/2] + \cdots + h[N-1]x[n-N+1] \\ &= h[0]\left(x[n] - x[n-N+1]\right) + h[1]\left(x[n-1] - x[n-N+2]\right) + \cdots \\ &\quad +h[N/2-1]\left(x[n-N/2+1] - x[n-N/2]\right) \end{aligned} \tag{10.8}$$

How many multipliers and additions are needed?

Example 10.2

Draw the block diagram using the direct form with minimum number of multiplications for the FIR filter whose impulse response is:

$$h[n] = h[4-n], \quad 0 \le n \le 4$$

Following (10.8), we obtain:

$$\begin{aligned} y[n] &= \sum_{k=0}^{4} h[k]x[n-k] \\ &= h[0]\left(x[n] + x[n-4]\right) + h[1]\left(x[n-1] + x[n-3]\right) + h[2]x[n-2] \end{aligned}$$

The number of multiplications is reduced from 5 to 3 while the number of additions remains unchanged, which is 4.

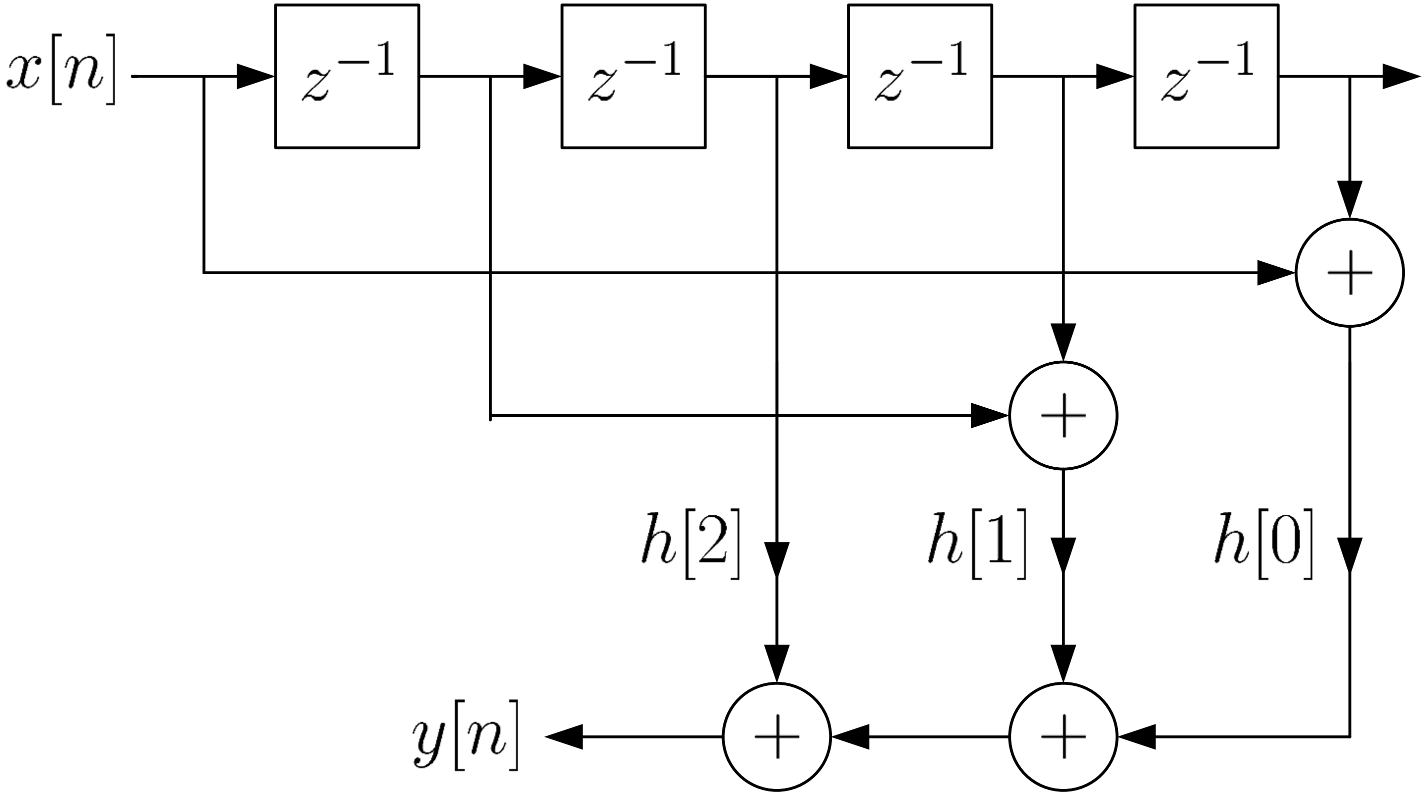


Fig.10.5: Block diagram for symmetric impulse response

There are four types of linear-phase FIR filters:

1. Symmetric Impulse Response with Odd N

Taking the DTFT of $h[n]$ yields

$$H(e^{j\omega}) = \left(2 \sum_{n=0}^{\frac{N-1}{2}-1} h[n] \cos\left(\left(\frac{N-1}{2} - n \right) \omega \right) + h[(N-1)/2] \right) e^{-j\omega(N-1)/2} \quad (10.9)$$

where $\phi = 0$ and $\tau = (N-1)/2$ is an integer.

Example 10.3

The impulse response $h[n]$ of a causal FIR filter satisfies

$$h[n] = h[N-1-n],\ 0 \leq n \leq N-1$$

with N is odd. Show that

$$\angle H(e^{j\omega}) = -\frac{N-1}{2}\omega$$

According to (10.6) with odd N, we have

$$\begin{gathered} h[0] = h[N-1] \\ h[1] = h[N-2] \\ \cdots\cdots\cdots\cdots\cdot \\ h[(N-1)/2-1] = h[(N-1)/2+1] \end{gathered}$$

Hence

$$\begin{aligned}
H(e^{j\omega}) &= \sum_{n=0}^{N-1} h[n]e^{-j\omega n} \\
&= h[0] + \cdots + h[(N-1)/2-1]e^{-j\omega((N-1)/2-1)} + h[(N-1)/2]e^{-j\omega(N-1)/2} \\
&\quad +h[(N-1)/2+1]e^{-j\omega((N-1)/2+1)} + \cdots + h[N-1]e^{-j\omega(N-1)} \\
&= e^{-j\omega(N-1)/2}\Big(h[0]e^{j\omega(N-1)/2} + h[N-1]e^{-j\omega(N-1)/2} \\
&\quad + \cdots + h[(N-1)/2-1]e^{j\omega} + h[(N-1)/2+1]e^{-j\omega} + h[(N-1)/2]\Big) \\
&= e^{-j\omega(N-1)/2}\Big(2h[0]\cos(\omega(N-1)/2) + \cdots + 2h[(N-1)/2-1]\cos(\omega) \\
&\quad +h[(N-1)/2]\Big) \\
&= e^{-j\omega(N-1)/2}\left(2\sum_{n=0}^{\frac{N-1}{2}-1} h[n]\cos\left(\left(\frac{N-1}{2}-n\right)\omega\right) + h[(N-1)/2]\right)
\end{aligned}$$

which validates (10.9) with $\angle H(e^{j\omega}) = -\omega(N-1)/2$.

2. Symmetric Impulse Response with Even N

The DTFT of $h[n]$ is:

$$H(e^{j\omega}) = 2\left(\sum_{n=0}^{\frac{N}{2}-1} h[n]\cos\left(\left(\frac{N-1}{2}-n\right)\omega\right)\right)e^{-j\omega(N-1)/2} \quad (10.10)$$

where $\phi = 0$ and $\tau = (N-1)/2$ is not an integer.

3. Anti-symmetric Impulse Response with Odd N

The DTFT of $h[n]$ is:

$$H(e^{j\omega}) = 2\left(\sum_{n=0}^{\frac{N-1}{2}-1} h[n]\sin\left(\left(\frac{N-1}{2}-n\right)\omega\right)\right)e^{j(\pi/2-\omega(N-1)/2)} \quad (10.11)$$

where $\phi = \pi/2$ and $\tau = (N-1)/2$ is an integer

Note that $h[(N-1)/2] = 0$ due to anti-symmetric property

4. Anti-symmetric Impulse Response with Even N

Taking the DTFT of $h[n]$ yields:

$$H(e^{j\omega}) = 2\left(\sum_{n=0}^{\frac{N}{2}-1} h[n]\sin\left(\left(\frac{N-1}{2} - n\right)\omega\right)\right) e^{j(\pi/2 - \omega(N-1)/2)} \quad (10.12)$$

where $\phi = \pi/2$ and $\tau = (N-1)/2$ is not an integer.

Example 10.4

Consider an input sequence $x[n]$ of length 400 such that $x[n]=0$ for $0 \le n \le 99$ and $200 \le n \le 299$, and it is a sinusoid with frequencies 0.03π and 0.08π for $100 \le n \le 199$ and $300 \le n \le 399$, respectively.

Examine the filter output $y[n]$ with the following two FIR systems:

(a) $h[n]=\{2,-0.9,-0.72,-0.58,-0.46,-0.37\}$ with $h[0]=2$

(b) $h[n]=\{0.6,0.9,-1.2,0.9,0.6\}$ with $h[0]=0.6$

Are they linear phase filters?

The MATLAB program for this example is provided as `ex10_4.m`.

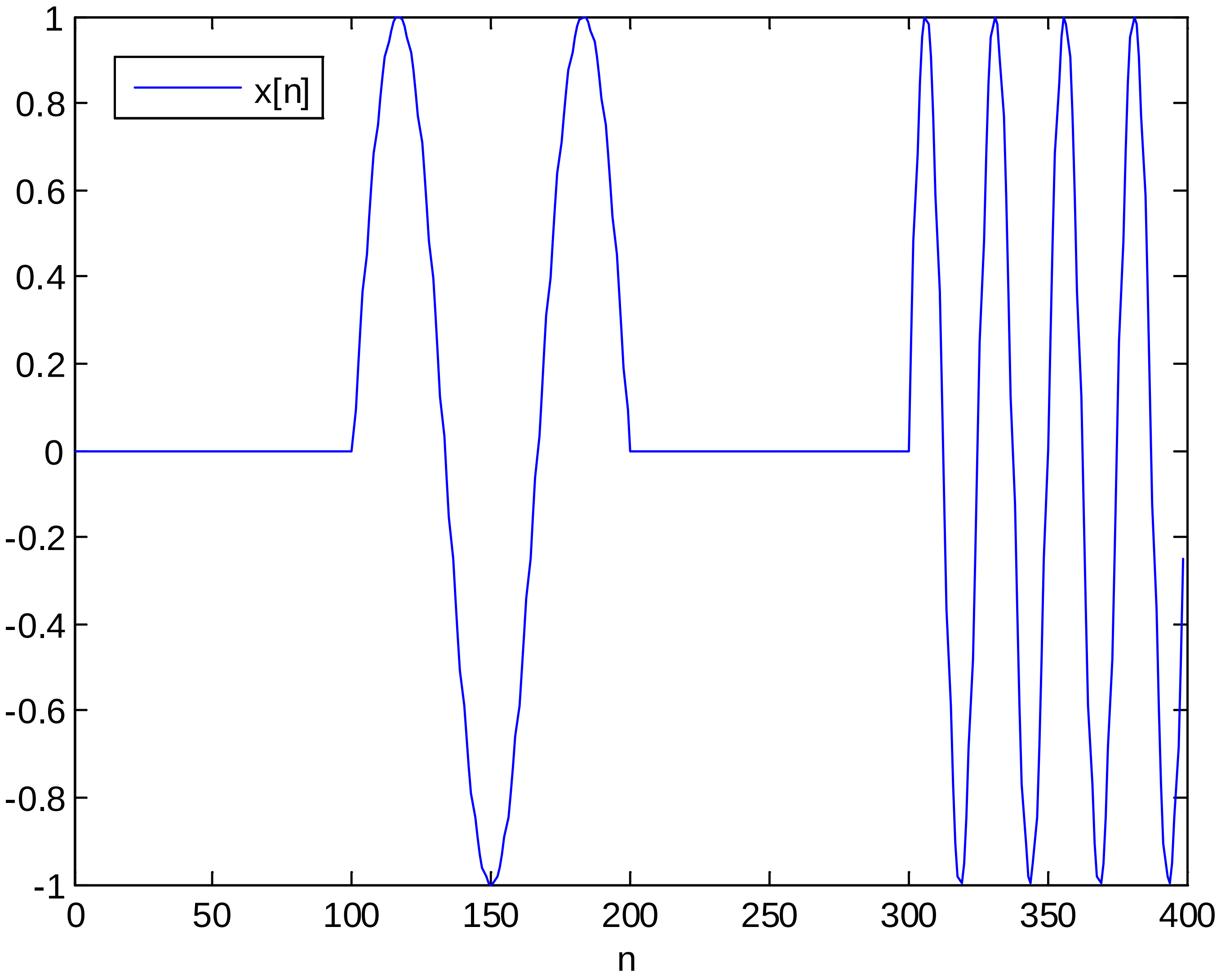


Fig.10.6: Pulsed sinusoid with two frequencies

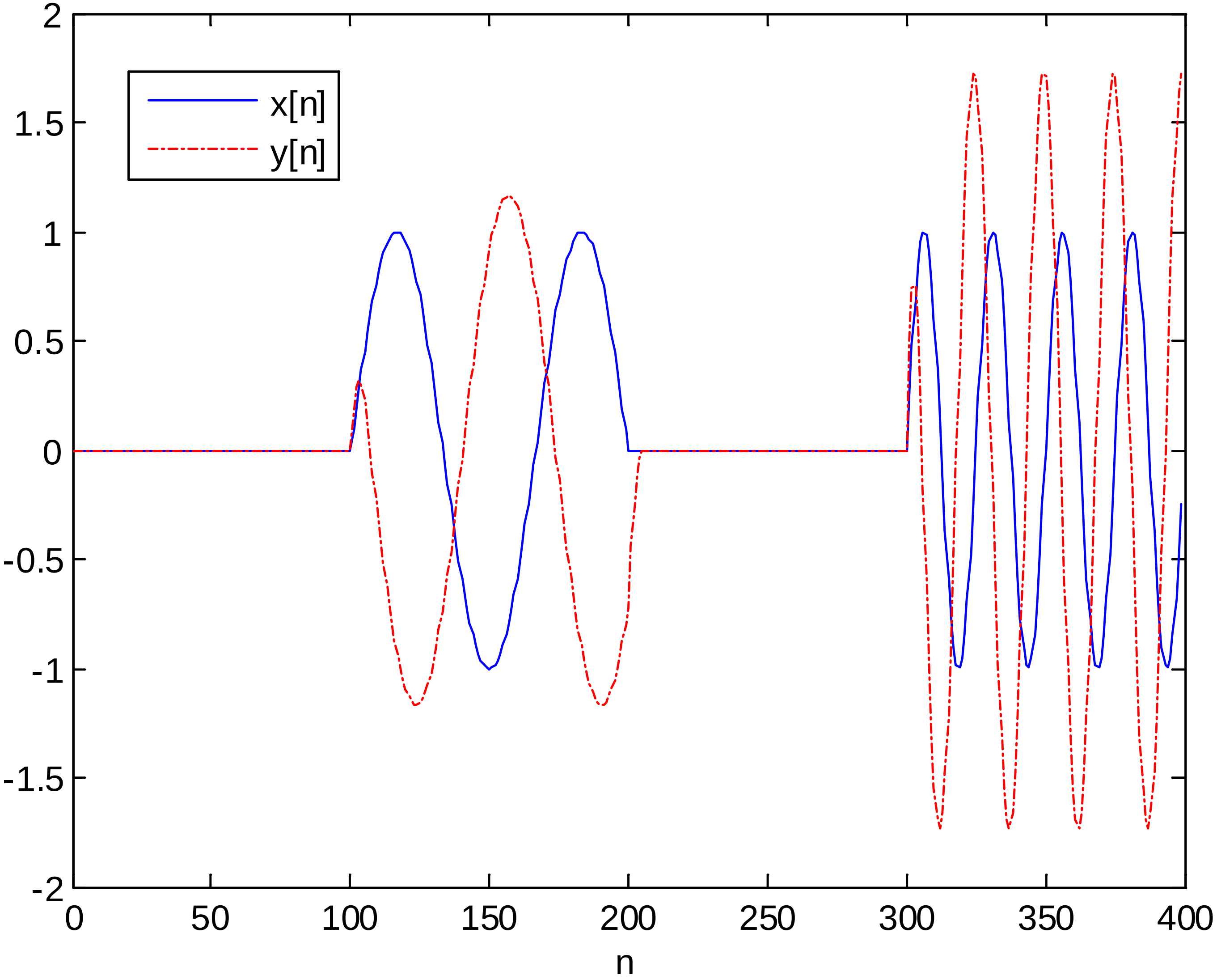


Fig.10.7: Filter output with nonlinear-phase filter

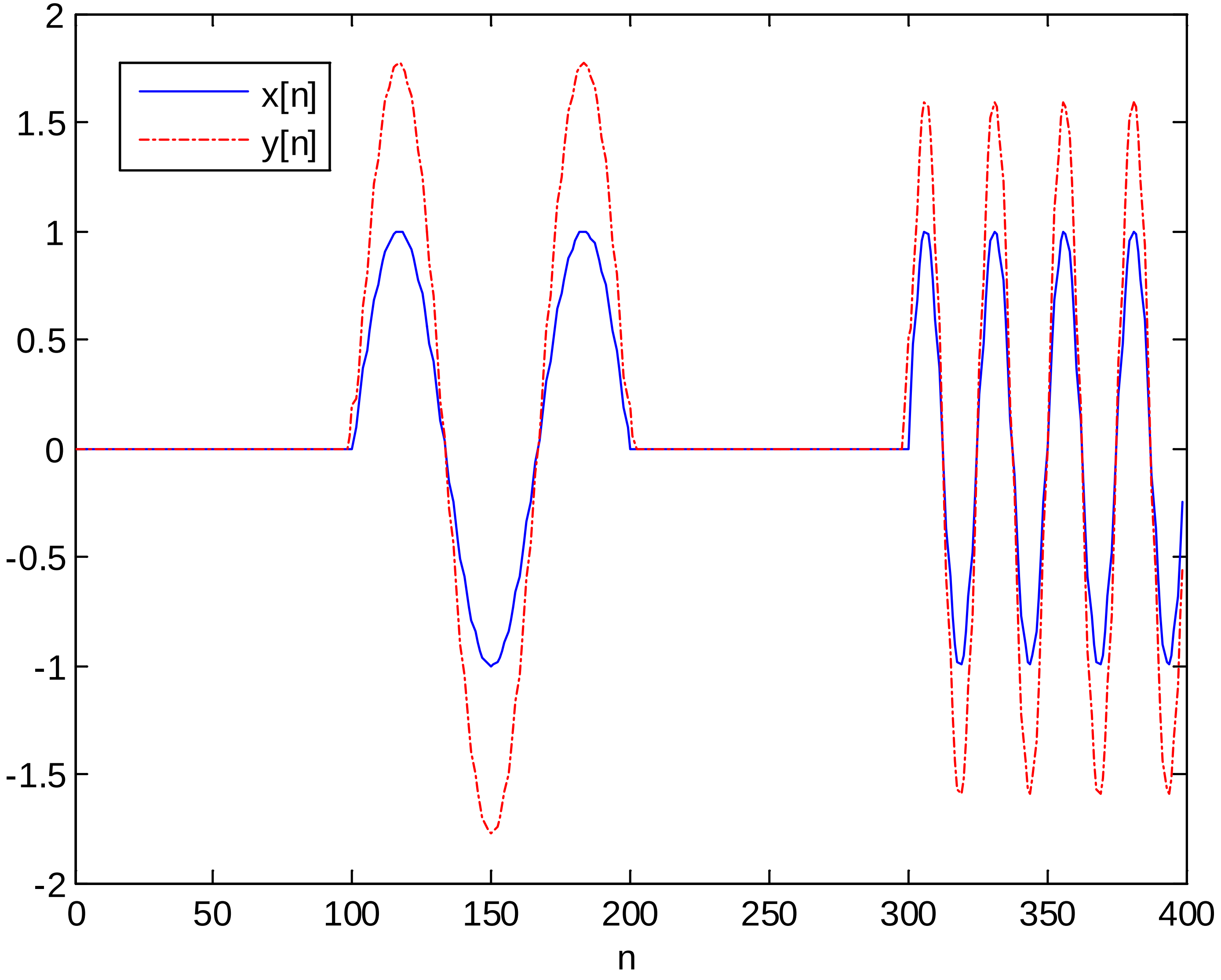


Fig.10.8: Filter output with linear-phase filter

Example 10.5

The impulse response of a causal linear-phase FIR $h[n]$ is:

$$h[n] = \{1/5, -1/4, 1/3, -1/2, 1, 0, -1, 1/2, -1/3, 1/4, -1/5\}$$

where $h[0] = 1/5$. Determine $H(e^{j\omega})$. Plot the amplitude and phase responses and then deduce the function of the filter. Find the expected filter output $y[n]$ when passing an input sequence of the form $x[n] = \sin(0.6n)$ through the filter.

As $N = 11$ is odd and $h[n]$ is anti-symmetric, we apply (10.11):

$$H(e^{j\omega}) = \left(2\sin(\omega) - \sin(2\omega) + \frac{2}{3}\sin(3\omega) - \frac{1}{2}\sin(4\omega) + \frac{2}{5}\sin(5\omega)\right) e^{j(\pi/2-5\omega)}$$

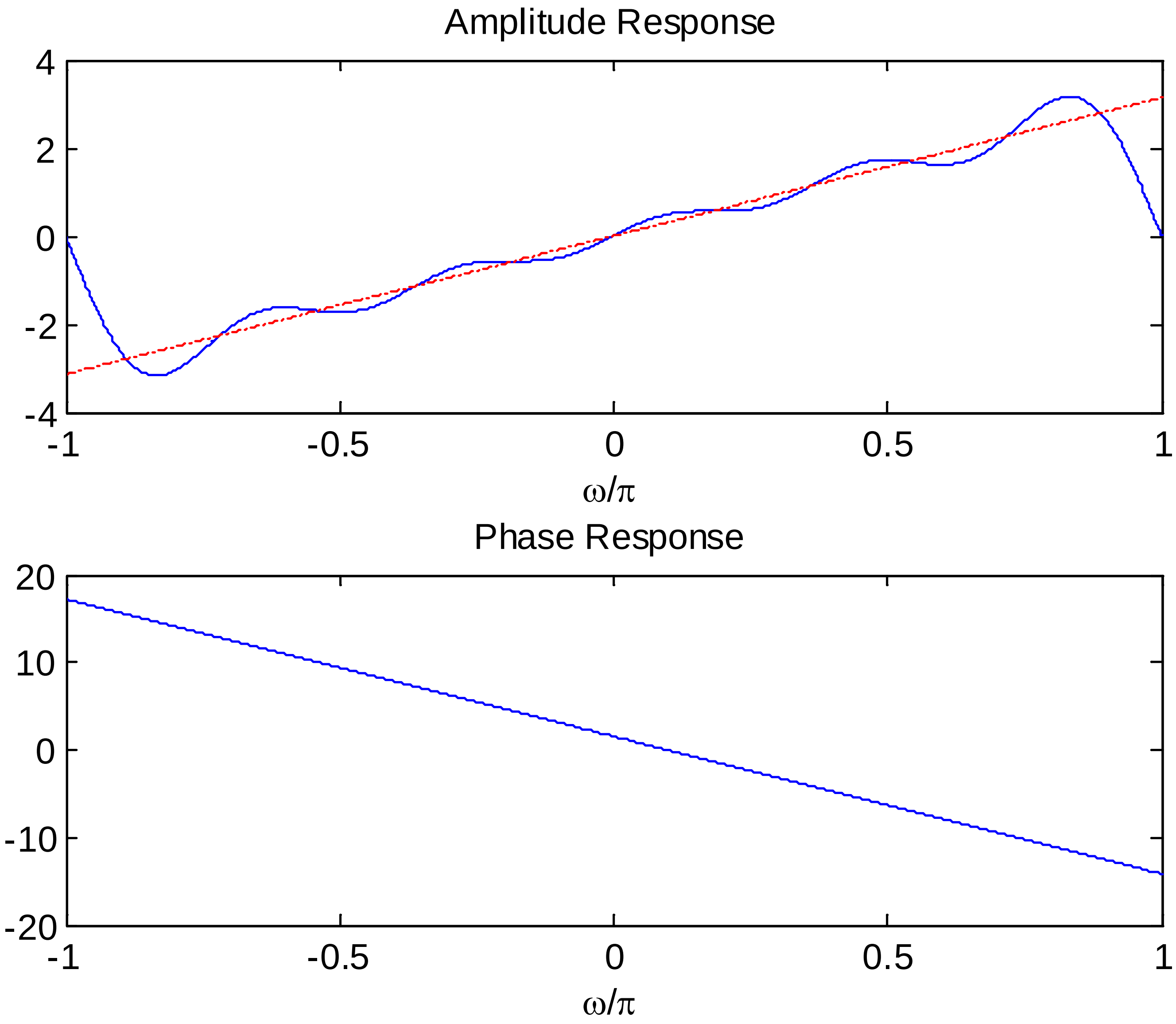


Fig.10.9: Amplitude and phase responses of differentiator

The amplitude response can be approximated as:

$$H_r(e^{j\omega}) = \left(2\sin(\omega) - \sin(2\omega) + \frac{2}{3}\sin(3\omega) - \frac{1}{2}\sin(4\omega) + \frac{2}{5}\sin(5\omega)\right) \approx \omega$$

According to the time-shifting property, the DTFT of $h[n]$ with a time advance of 5 samples is then:

$$\left(2\sin(\omega) - \sin(2\omega) + \frac{2}{3}\sin(3\omega) - \frac{1}{2}\sin(4\omega) + \frac{2}{5}\sin(5\omega)\right) e^{j\pi/2}$$
$$\approx \omega e^{j\pi/2} = \omega\left[\cos\left(\frac{\pi}{2}\right) + j\sin\left(\frac{\pi}{2}\right)\right] = j\omega$$

This means that the causal FIR filter has a frequency response approximately equals $j\omega$ with a delay of 5 samples.

The system with frequency response $j\omega$ is known as the differentiator. Recall:

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

Differentiating both sides with respect to n yields:

$$\begin{aligned} \frac{dx[n]}{dn} &= \frac{d}{dn}\left(\frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega \right) \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) \frac{de^{j\omega n}}{dn} d\omega = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left[X(e^{j\omega}) \cdot j\omega \right] e^{j\omega n} d\omega \end{aligned}$$

which confirms:

$$\frac{dx[n]}{dn} \leftrightarrow X(e^{j\omega}) \cdot j\omega$$

As a result, we expect the output $y[n]$ is close to:

$$y[n] = \frac{dx[n]}{dn} = 0.6\cos(0.6n)$$

with a delay of 5 samples

The MATLAB program is provided as `ex10_5.m`.

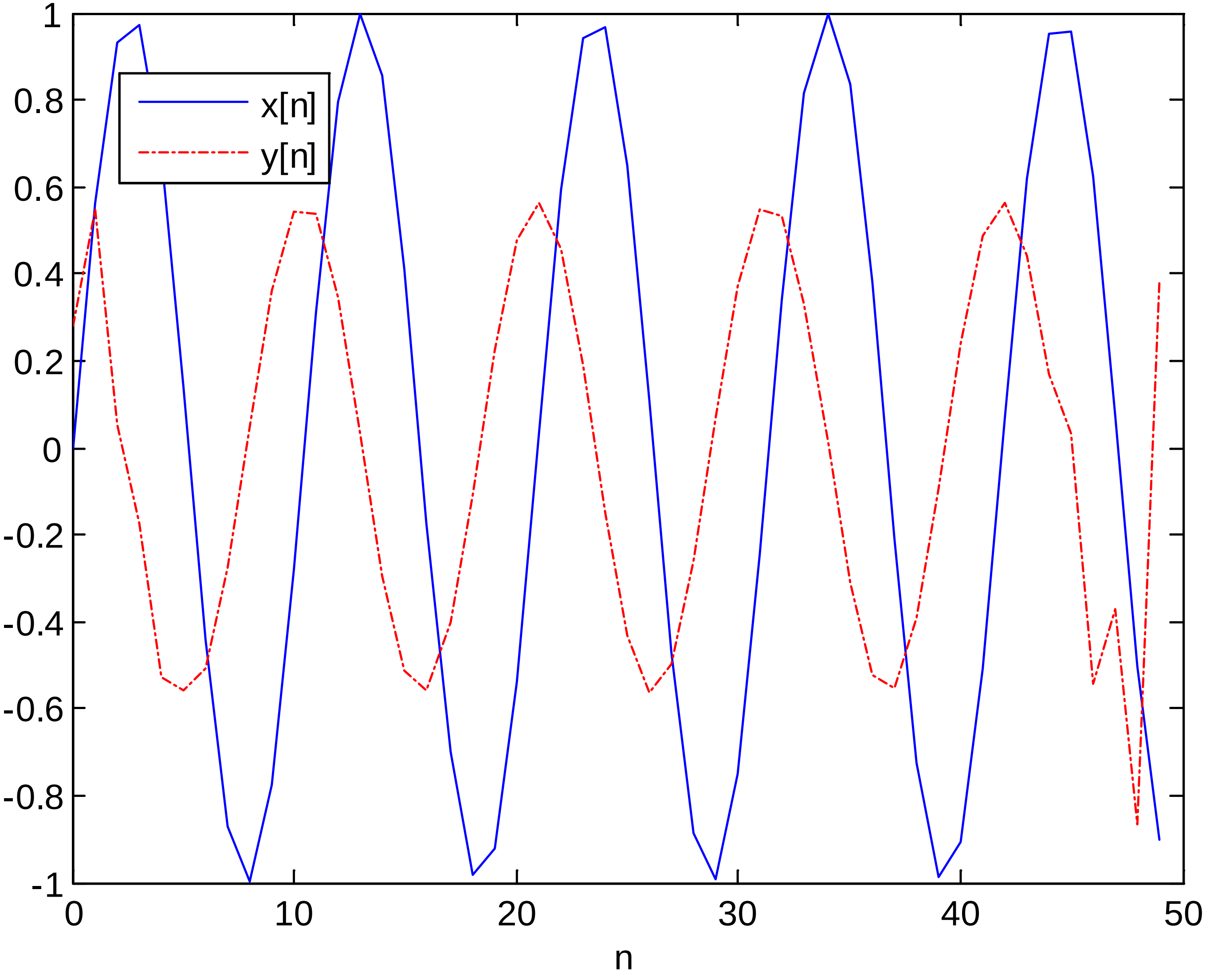


Fig.10.10: Differentiator output with sinusoidal input

Window Method

Based on directly approximating the desired frequency response $H_d(e^{j\omega})$

Ideal impulse response $h_d[n]$ is calculated from inverse DTFT:

$$h_d[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}d\omega, \quad n = \cdots, -1, 0, 1, 2, \cdots \qquad (10.13)$$

What are the problems in using (10.13)?

Example 10.6

Determine the impulse response $h_d[n]$ of an ideal lowpass filter with a cutoff frequency of ω_c. The DTFT of the filter is:

$$H_d(e^{j\omega}) = \begin{cases} 1, & |\omega| < \omega_c \\ 0, & -\pi < \omega < -\omega_c, \quad \omega_c < \omega < \pi \end{cases}$$

Using Example 6.3, $h_d[n]$ is:

$$h_d[n] = \frac{\sin(\omega_c n)}{\pi n} = \frac{\omega_c}{\pi}\mathrm{sinc}\left(\frac{\omega_c n}{\pi}\right)$$

where

$$\mathrm{sinc}(u) = \frac{\sin(\pi u)}{\pi u}$$

Ideal filter is noncausal and its impulse response is of infinite length

The basic idea of window method is to truncate $h_d[n]$ to obtain a linear-phase and causal FIR filter via 2 steps:

- Windowing
  - Extract a finite set of $h_d[n]$ with positive and negative time indexes because the impulse response is generally symmetric or anti-symmetric around $n=0$, say, $\{h_d[-M], h_d[-M+1], \cdots, h_d[M]\}$ corresponding to a length of $(2M+1)$
  - Filter response is linear phase but is only an approximation of $H_d(e^{j\omega})$ due to the coefficient truncation
- Time Shifting
  - Delay $\{h_d[-M], h_d[-M+1], \cdots, h_d[M]\}$ by M samples to obtain $h[n]$ with $h[0]=h_d[-M]$ to achieve causality
  - Filter output is received after a delay of M samples

Example 10.7

Use the window method to design a linear-phase and causal FIR system with 7 coefficients to approximate an ideal lowpass filter whose cutoff frequency is $\omega_c = 0.1\pi$.

From Example 10.6:

$$h_d[n] = \frac{\sin(0.1\pi n)}{\pi n} = 0.1\text{sinc}\left(0.1n\right)$$

Via windowing, we extract the set of $\{h_d[-3], h_d[-2], \cdots, h_d[3]\}$. Notice that $h_d[n]$ is symmetric around at $n=0$ as $h_d[n] = h_d[-n]$.

Their values are calculated as:

$$h_d[3] = h_d[-3] = \frac{\sin(0.1\pi \cdot 3)}{\pi \cdot 3} = 0.08584$$

$$h_d[2] = h_d[-2] = \frac{\sin(0.1\pi \cdot 2)}{\pi \cdot 2} = 0.09355$$

$$h_d[1] = h_d[-1] = \frac{\sin(0.1\pi)}{\pi} = 0.09836$$

$$h_d[0] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\omega}) e^{j\omega \cdot 0} d\omega = \frac{1}{2\pi} \int_{-0.1\pi}^{0.1\pi} d\omega = 0.1$$

Via time-shifting of 3 samples, a causal filter is obtained as:

$$h[n] = h_d[n-3], \quad n = 0, 1, \cdots, 6:$$

or

$$h[n] = \{0.08584, 0.09355, 0.9836, 0.1, 0.9836, 0.09355, 0.08584\}$$

with $h[0] = 0.08584$.

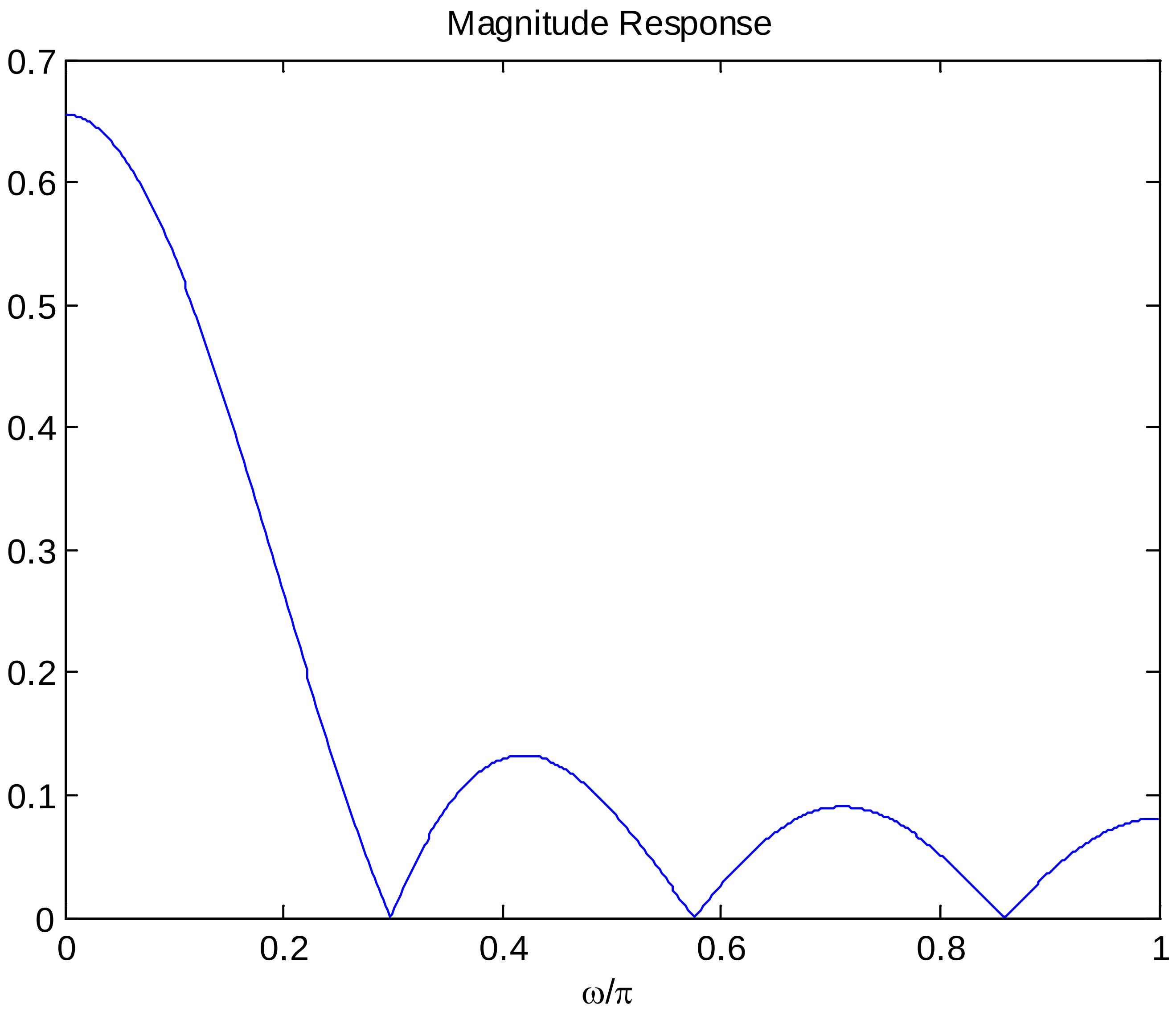


Fig.10.11: Magnitude response of lowpass filter with $N = 7$

Note that we can use the MATLAB command `fir1(6,0.1,boxcar(7),'noscale')` to produce $h[n]$

Alternatively, we can first perform the time shifting prior to windowing

As there should be a phase of $-\tau\omega$ where $\tau = (N-1)/2$ in the practical filter, we modify the desired frequency response as:

$$H_d(e^{j\omega}) = \begin{cases} 1 \cdot e^{-j\omega\tau}, & |\omega| < \omega_c \\ 0, & -\pi < \omega < -\omega_c, \quad \omega_c < \omega < \pi \end{cases}$$

Note that multiplying $e^{-j\omega\tau}$ in the frequency domain corresponds to a time shift of τ in the time domain.

The corresponding impulse response is

$$h_d[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}d\omega = \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} e^{-j\omega\tau}e^{j\omega n}d\omega = \frac{\sin(\omega_c(n-\tau))}{\pi(n-\tau)}$$

As the filter length is N, we then perform windowing on $h_d[n]$:

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

Substituting $N=7$ and $\omega_c = 0.1\pi$ yields the same FIR impulse response.

Note that we can base on this approach to determine $h[n]$ even when N is an even integer

The truncation operation can be considered as multiplying $h_d[n]$ by a rectangular window function:

$$w[n] = \begin{cases} 1, \ 0 \leq n \leq N-1 \\ 0, \ \text{otherwise} \end{cases}$$

That is,

$$h[n] = h_d[n] \cdot w[n]$$

Generally speaking, $w[n]$ is not restricted to be rectangular and it can be any symmetric function so that the resultant filter is linear phase

Example 10.8

Use the window method to design a linear-phase and causal FIR filter of length 101 such that the sampled version of a continuous-time sinusoid with frequency of 80 Hz can pass through it with negligible attenuation while the sampled signal corresponds to a tone of frequency 120 Hz will be suppressed. The sampling frequency is 1000 Hz.

Let the continuous-time sinusoids be

$$s_i(t) = \cos(\Omega_i t), \quad \Omega_1 = 80 \cdot 2\pi,\ \Omega_2 = 120 \cdot 2\pi$$

From (4.1), the corresponding discrete-time versions with sampling interval of $T = 1/1000$ s are

$$s_i[n] = s_i(nT) = \cos(\Omega_i n / 1000)$$

which gives:

$$s_1[n] = \cos(0.16\pi n)$$

and

$$s_2[n] = \cos(0.24\pi n)$$

The frequencies are 0.16π and 0.24π in discrete-time domain.

In order to suppress $s_2[n]$ while keeping $s_1[n]$, we can use a lowpass filter with cutoff frequency of $\omega_c = 0.2\pi$, which is simply the average of the two discrete frequencies.

Using Example 10.7 with $N = 101$ and $\omega_c = 0.2\pi$, the required filter impulse response $h[n]$ is:

$$h[n] = \frac{\sin(0.2\pi(n-50))}{\pi(n-50)} = 0.2\mathrm{sinc}(0.2(n-50)), \quad 0 \leq n \leq 100$$

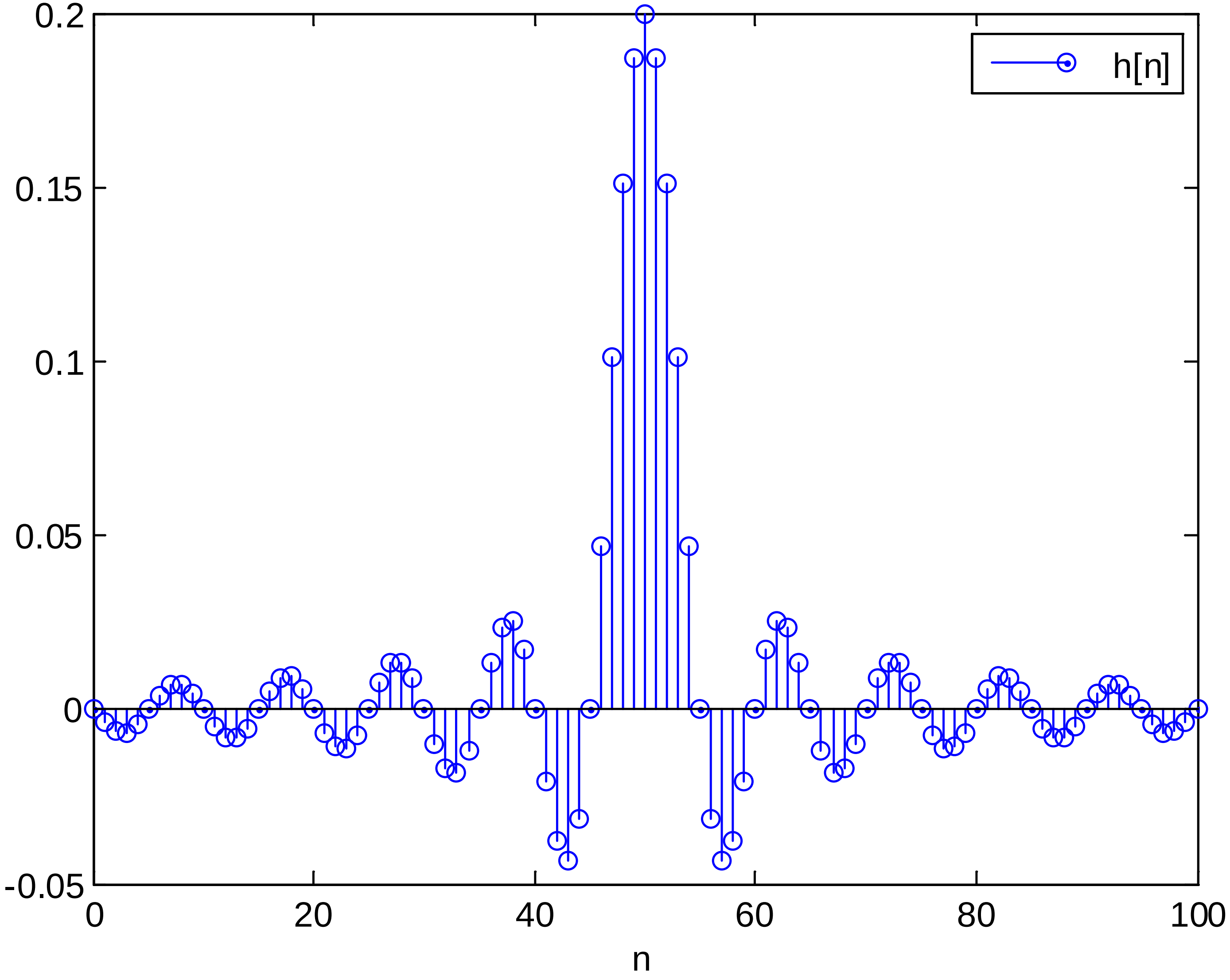


Fig.10.12: Impulse response of lowpass filter with $N = 101$

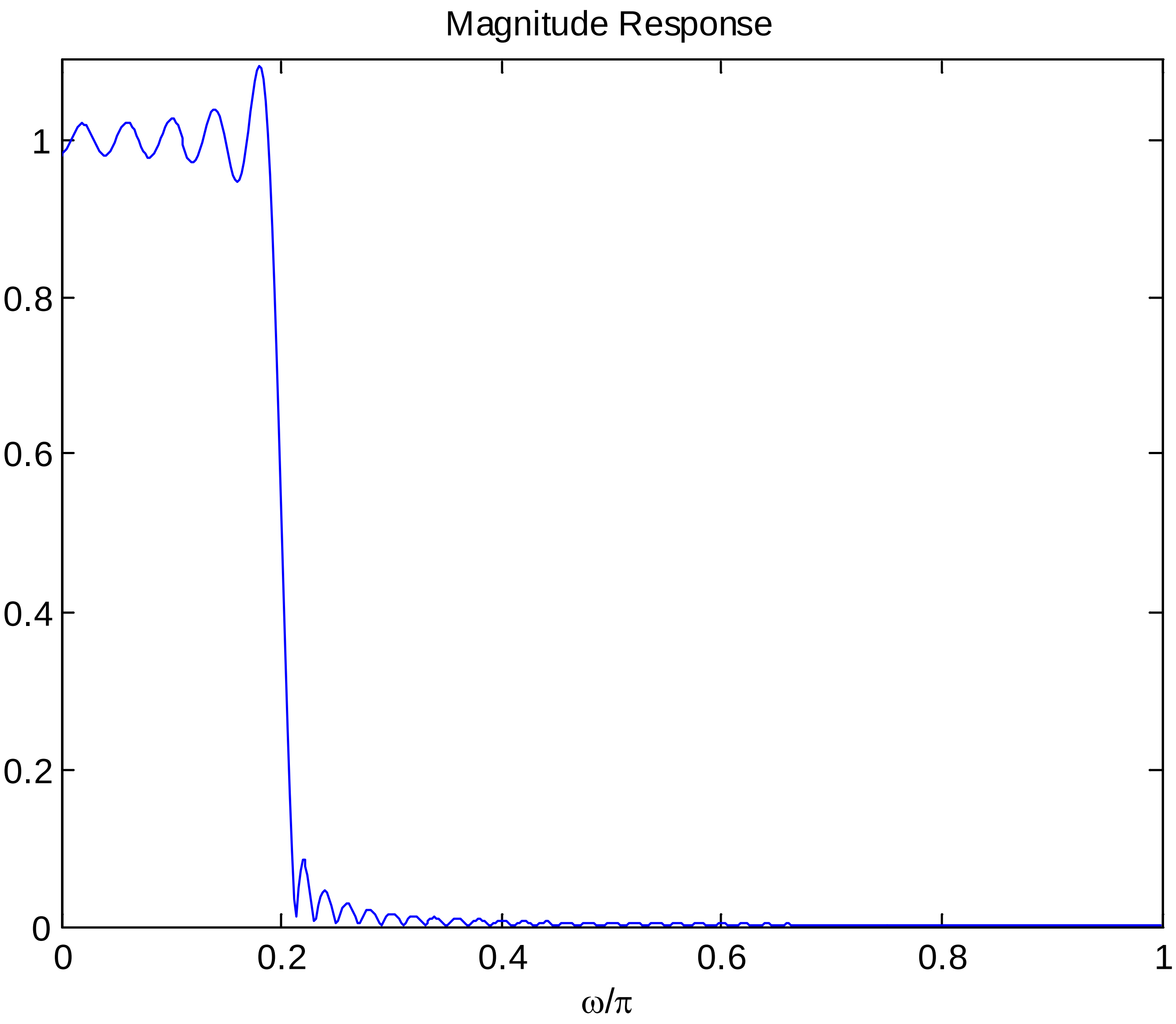


Fig.10.13: Magnitude response of lowpass filter with $N = 101$

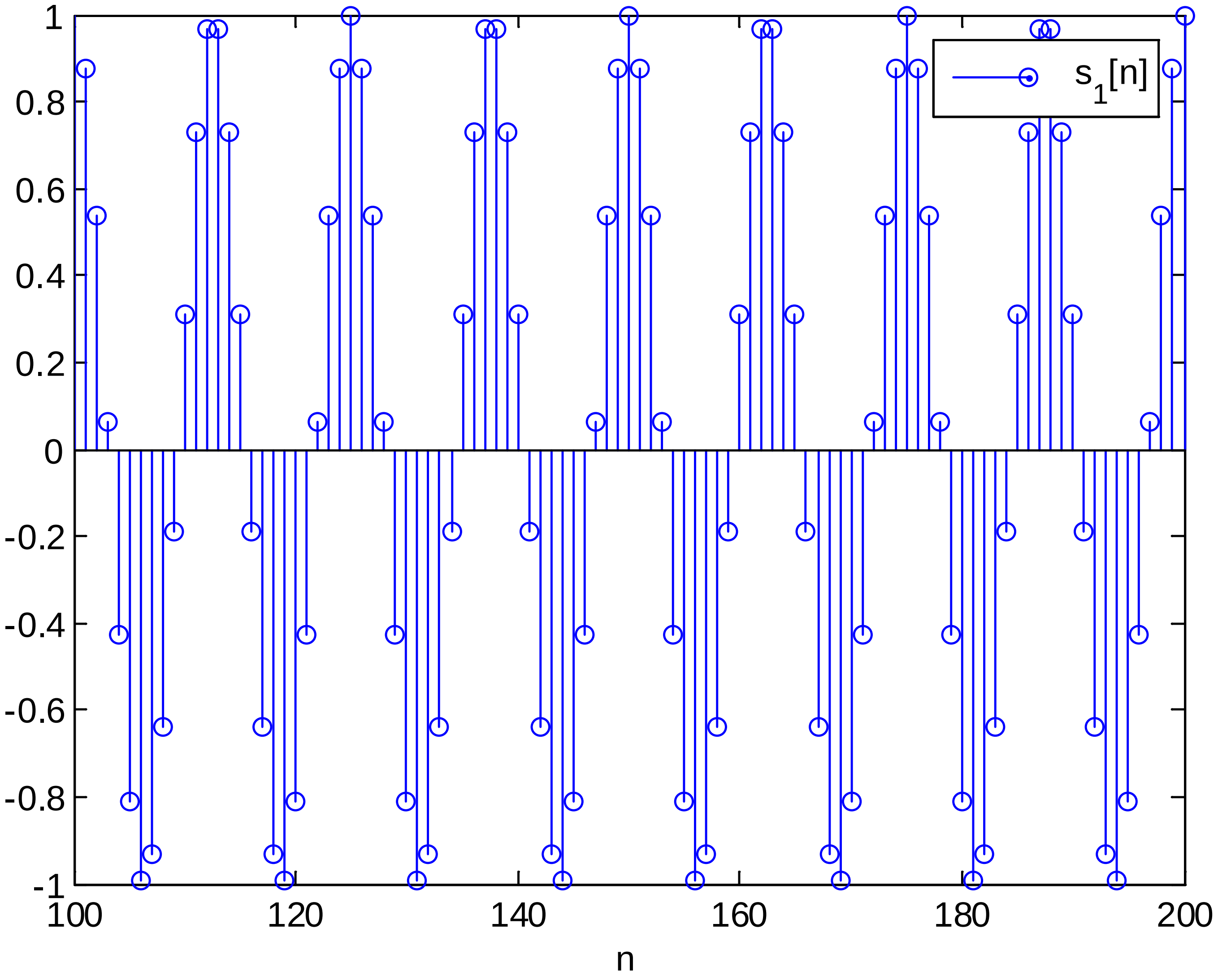


Fig.10.14: Discrete-time tone with frequency 0.16π

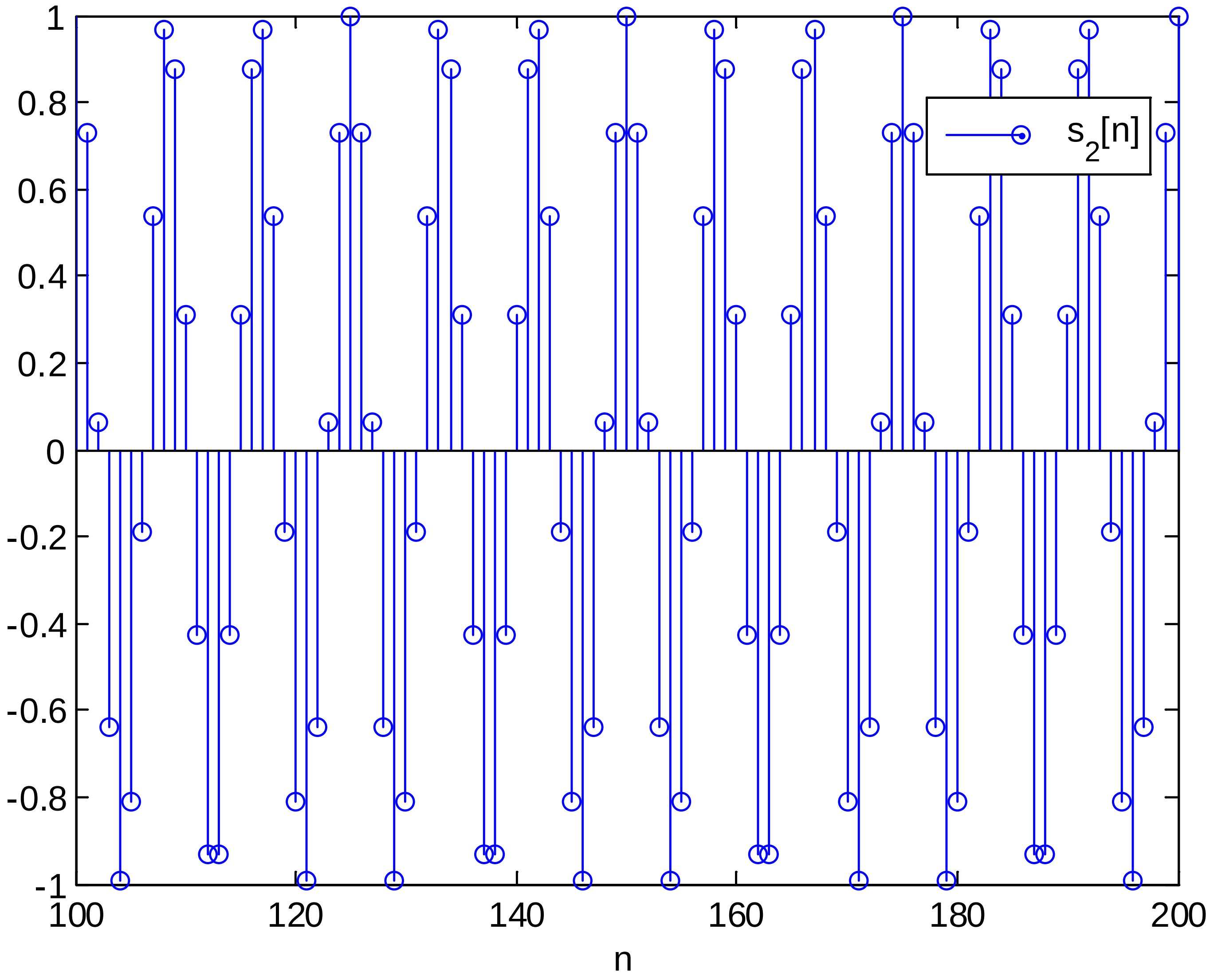


Fig.10.15: Discrete-time tone with frequency 0.24π

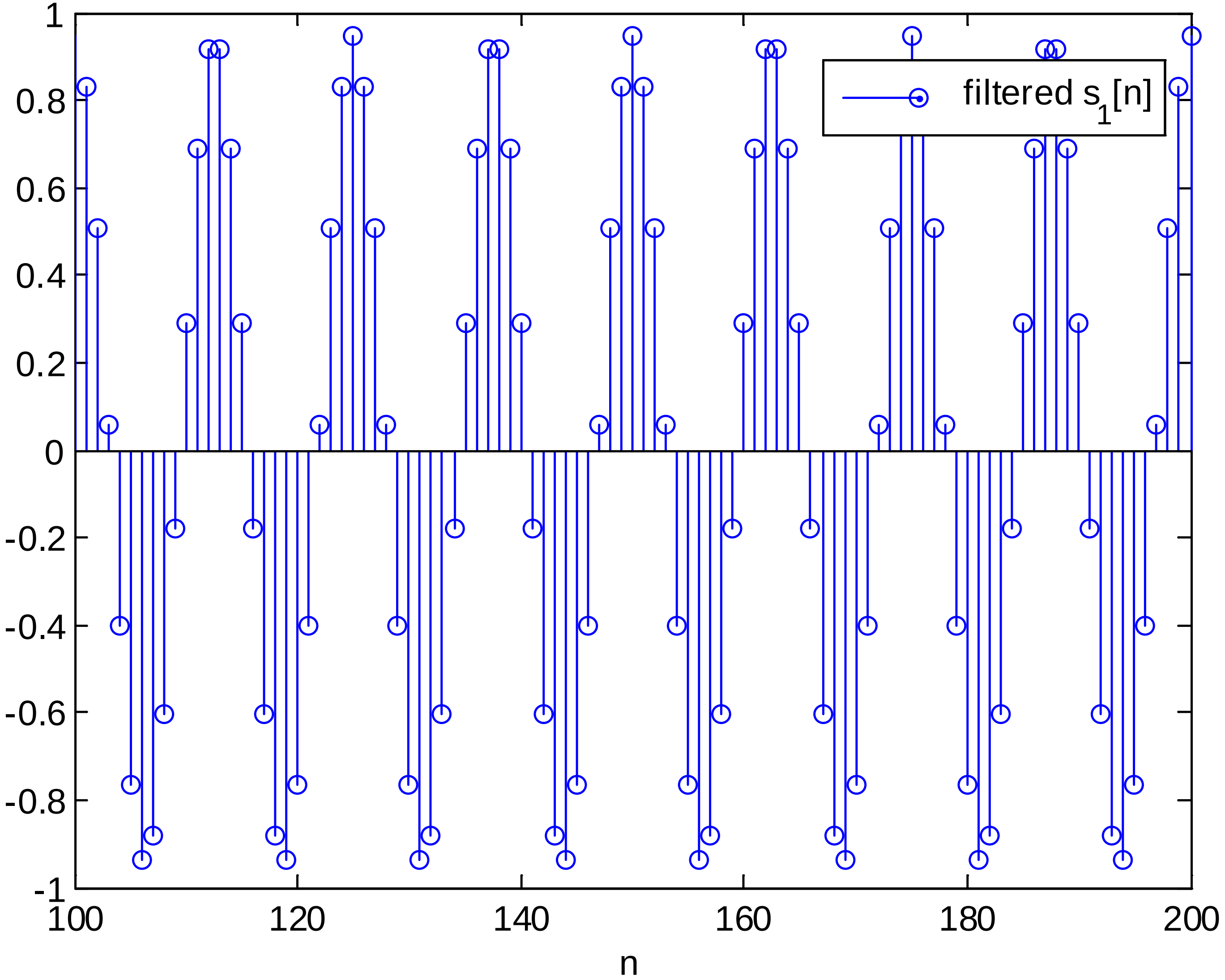


Fig.10.16: Filter output for frequency 0.16π

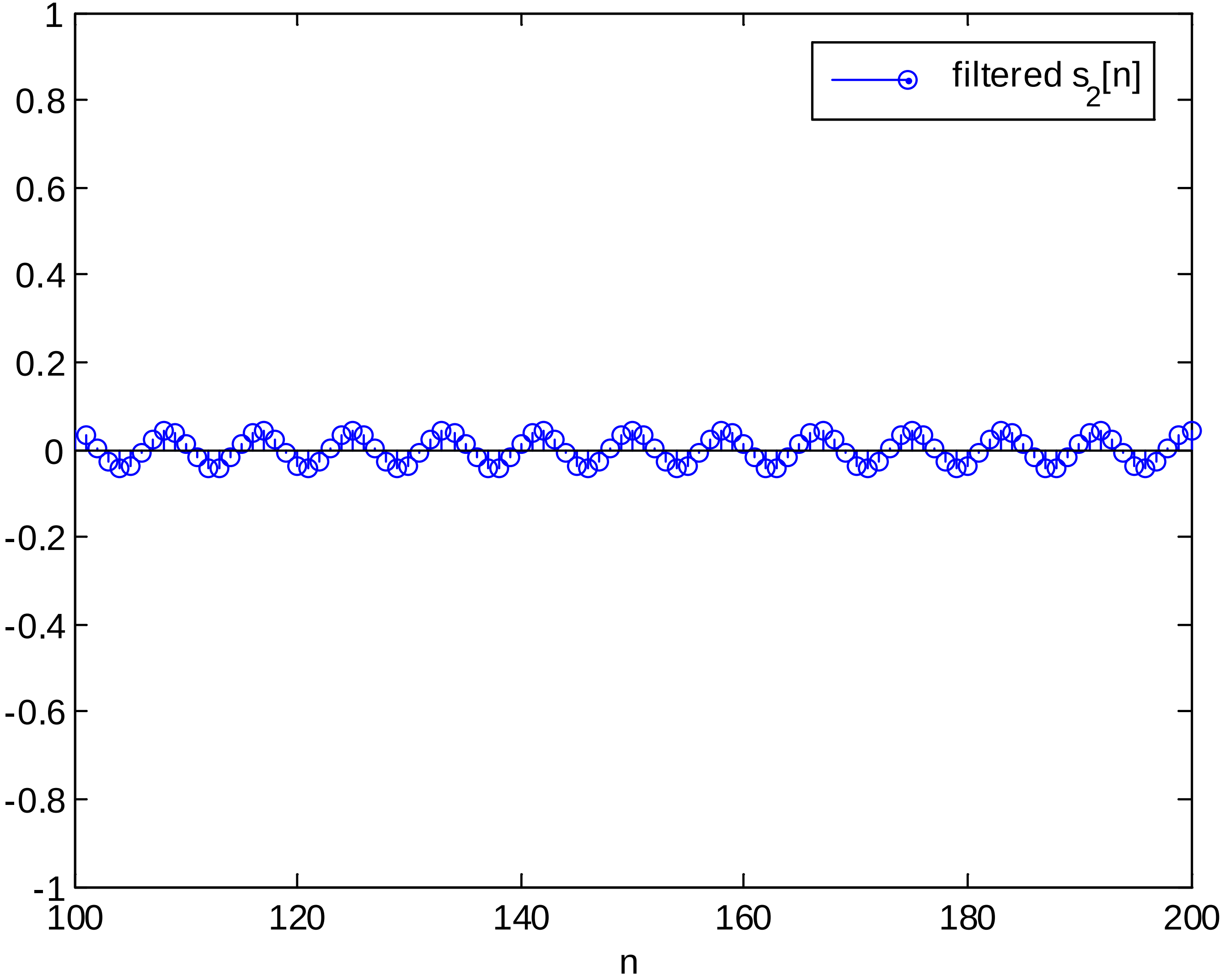


Fig.10.17: Filter output for frequency 0.24π

Analysis of Windowing

To incorporate passband and stopband frequencies and ripples in the FIR filter design, we need to study the windowing effect

Recall

$$h[n] = h_d[n] \cdot w[n] \qquad (10.14)$$

According to the multiplication property of (6.18):

$$H(e^{j\omega}) = H_d(e^{j\omega}) \tilde{\otimes} W(e^{j\omega}) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\tau}) W(e^{j(\omega-\tau)}) d\tau \qquad (10.15)$$

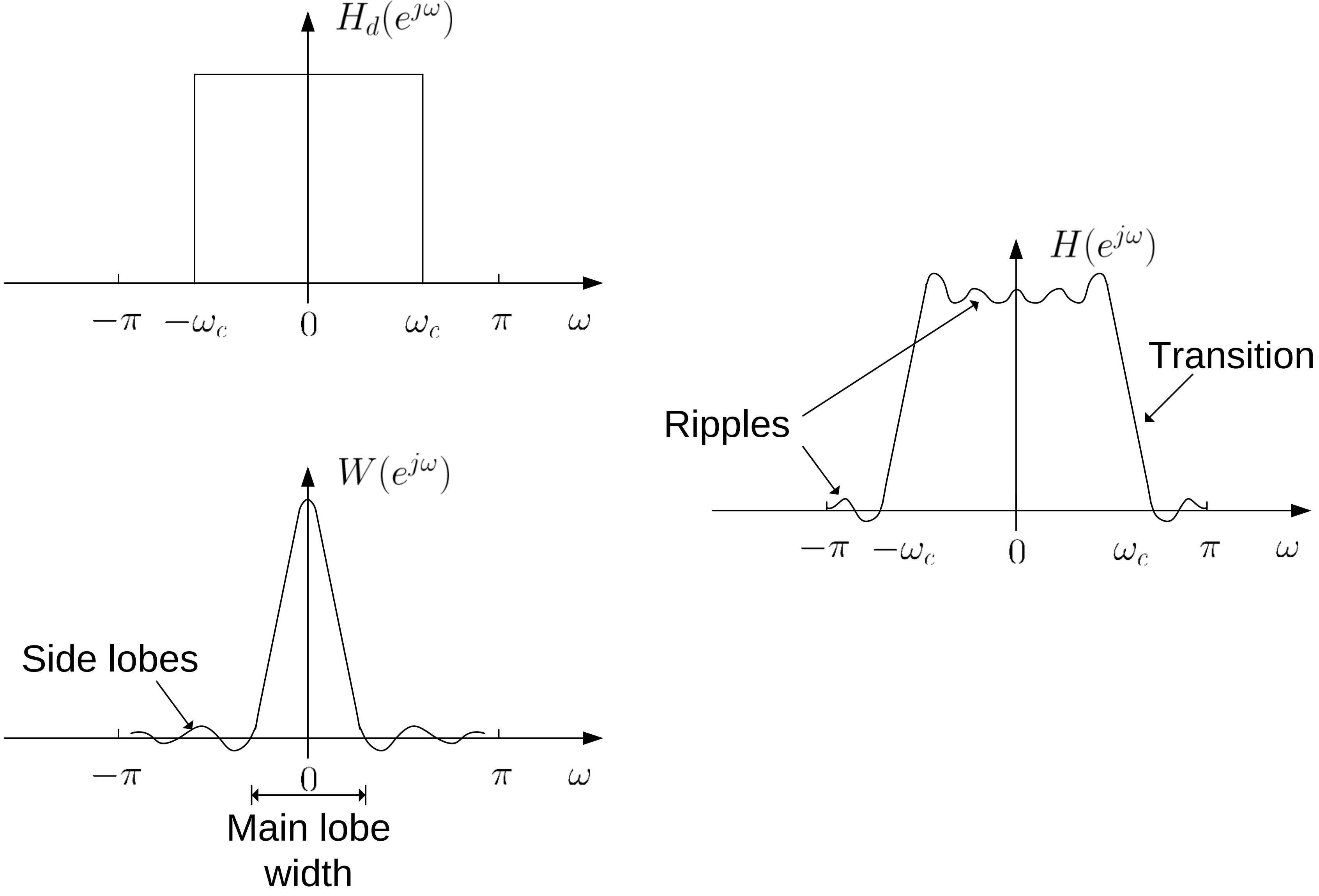


Fig.10.18: Illustration of $H(e^{j\omega}) = H_d(e^{j\omega})\tilde{\otimes}W(e^{j\omega})$

$H(e^{j\omega})$ is a smeared version of ideal $H_d(e^{j\omega})$

$W(e^{j\omega})$ has a peaky main lobe and several side lobes of smaller magnitudes

Main lobe produces the transition band in $H(e^{j\omega})$. That is, the transition width is proportional to the main lobe width, which is inversely proportional to the filter length N

Side lobes are responsible to produce the ripples in the passband and stopband with $\delta_1 = \delta_2$

$H(e^{j\omega})$ approaches $H_d(e^{j\omega})$ when $W(e^{j\omega})$ has the smallest main lobe width and side lobe magnitude. That is, the ideal but not practical form of $W(e^{j\omega}) = \delta(\omega)$ gives $H(e^{j\omega}) = H_d(e^{j\omega})$

- Rectangular Window

$$w[n] = u[n] - u[n-N] \qquad (10.16)$$

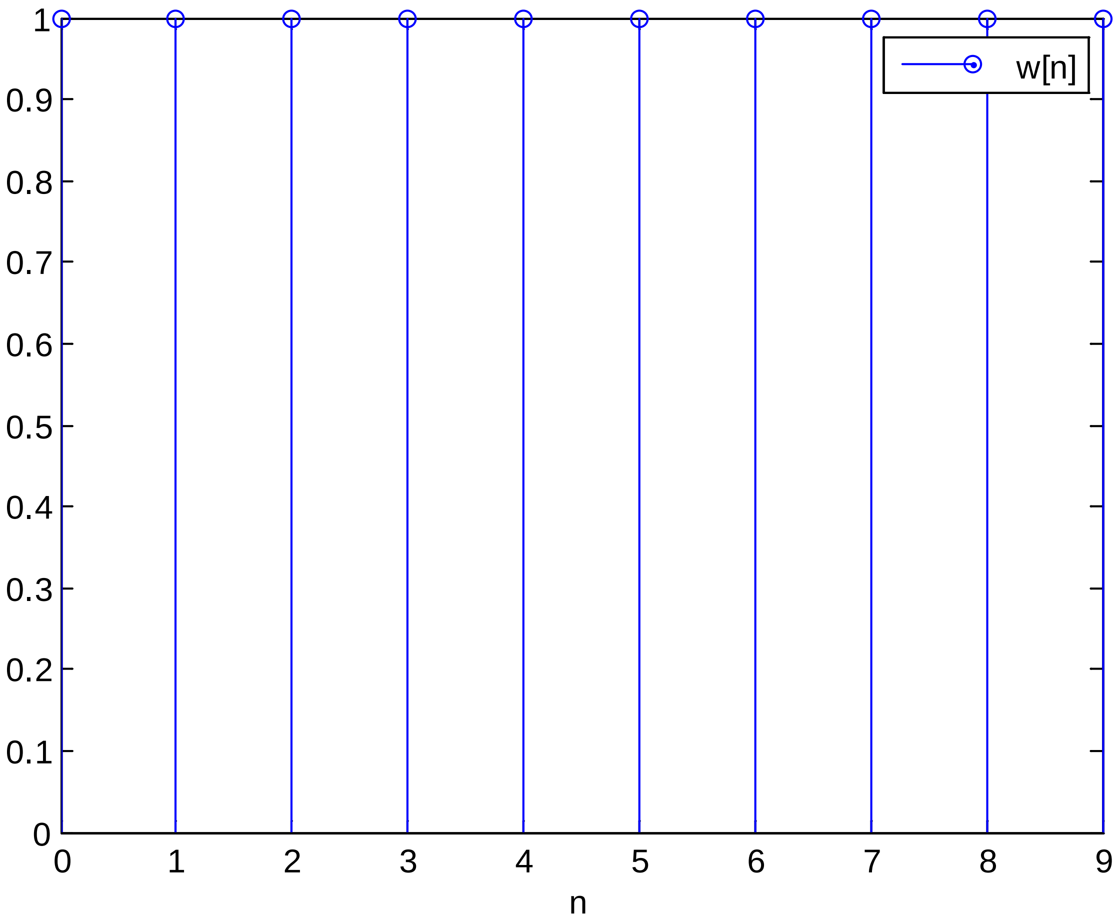


Fig.10.19: Rectangular window function for $N = 10$

From Example 10.1:

$$W(e^{j\omega}) = \frac{\sin(\omega N/2)}{\sin(\omega/2)} \cdot e^{-j\omega(N-1)/2} \qquad (10.17)$$

The main lobe width is

$$\Delta\omega_m = \frac{4\pi}{N} \qquad (10.18)$$

which is proportional to the filter transition width

That is, transition width decreases as the filter length N increases.

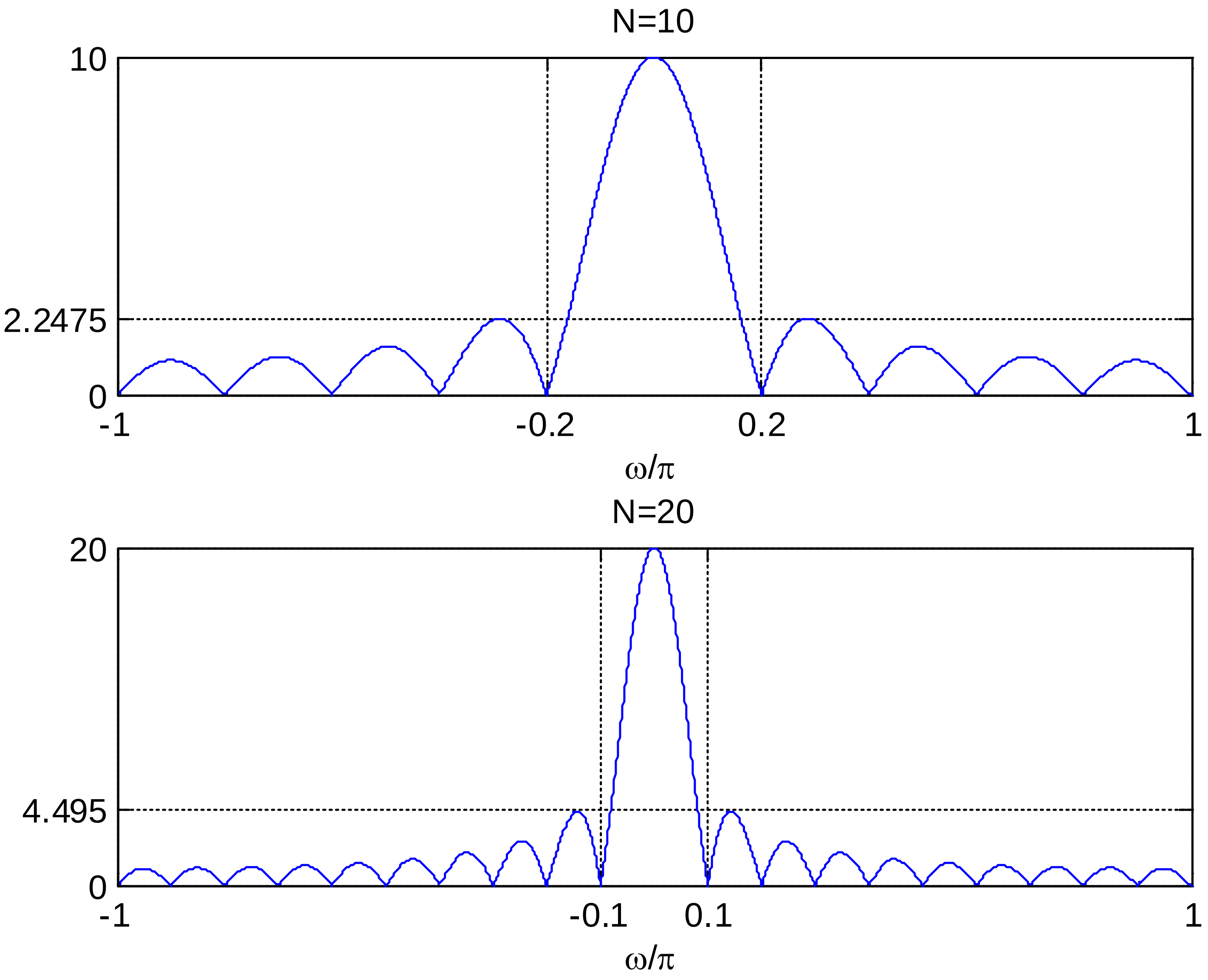


Fig.10.20: Magnitude responses of rectangular window for $N = 10$ & $N = 20$

Relative peak side lobe, r_p, is the ratio of the peak side lobe height to main lobe height in dB:

$$r_p = 20\log_{10}\left(\frac{2.2475}{10}\right) = 20\log_{10}\left(\frac{4.495}{20}\right) = -13 \text{ dB} \qquad (10.19)$$

A larger r_p implies ripples with bigger magnitudes

As N increases, widths of the main lobe and all side lobes decrease but their areas remain unchanged, meaning that the ripples cannot be reduced.

This is known as Gibbs phenomenon due to sudden change in transition from 0 to 1 and 1 to 0. This can be removed by tapering the window smoothly to zero at each end. However, the heights of the side lobes can be diminished but at the expense of wider main lobe and hence a wider transition.

- Bartlett Window

$$w[n] = \begin{cases} 1 - \dfrac{2\left|n - \dfrac{N-1}{2}\right|}{N-1}, & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases} \qquad (10.20)$$

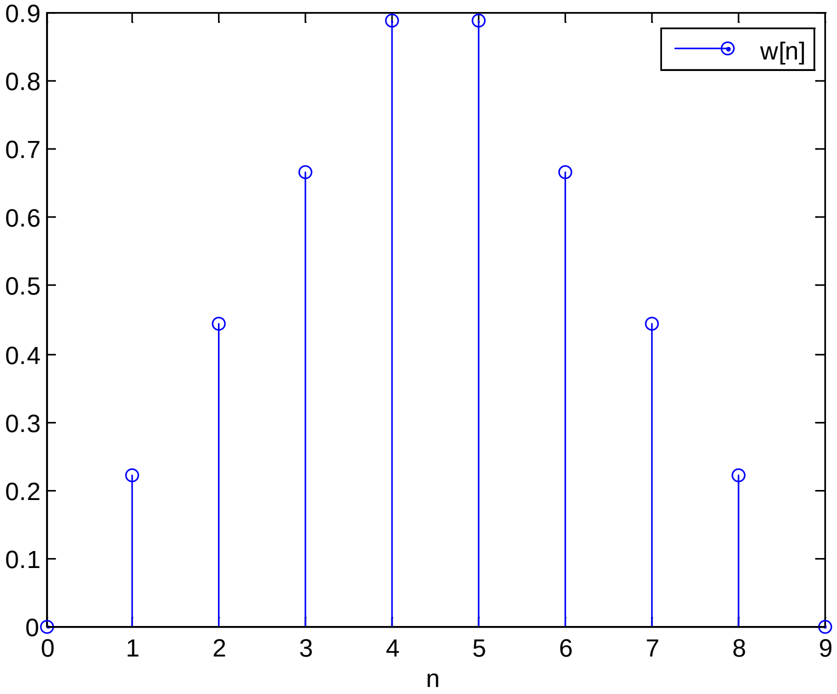


Fig.10.21: Bartlett window function for $N = 10$

- Hanning Window

$$w[n] = \begin{cases} \frac{1}{2}\left(1-\cos\left(\frac{2\pi n}{N-1}\right)\right), & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases} \qquad (10.21)$$

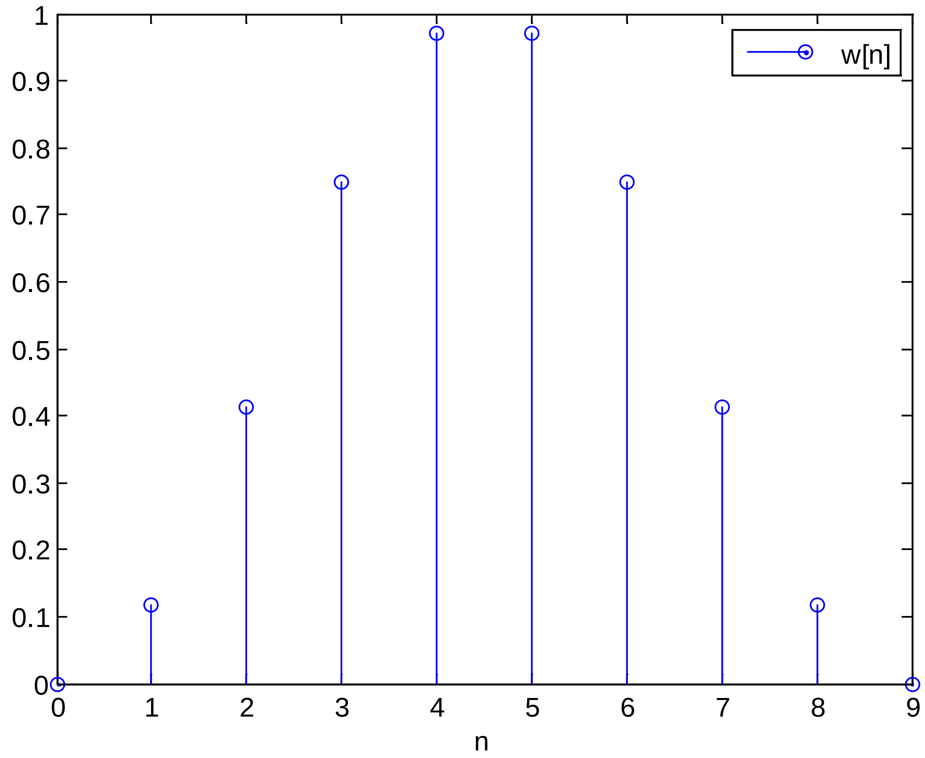


Fig.10.22: Hanning window function for $N=10$

- Hamming Window

$$w[n] = \begin{cases} 0.54 - 0.46\cos\left(\dfrac{2\pi n}{N-1}\right), & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases} \qquad (10.22)$$

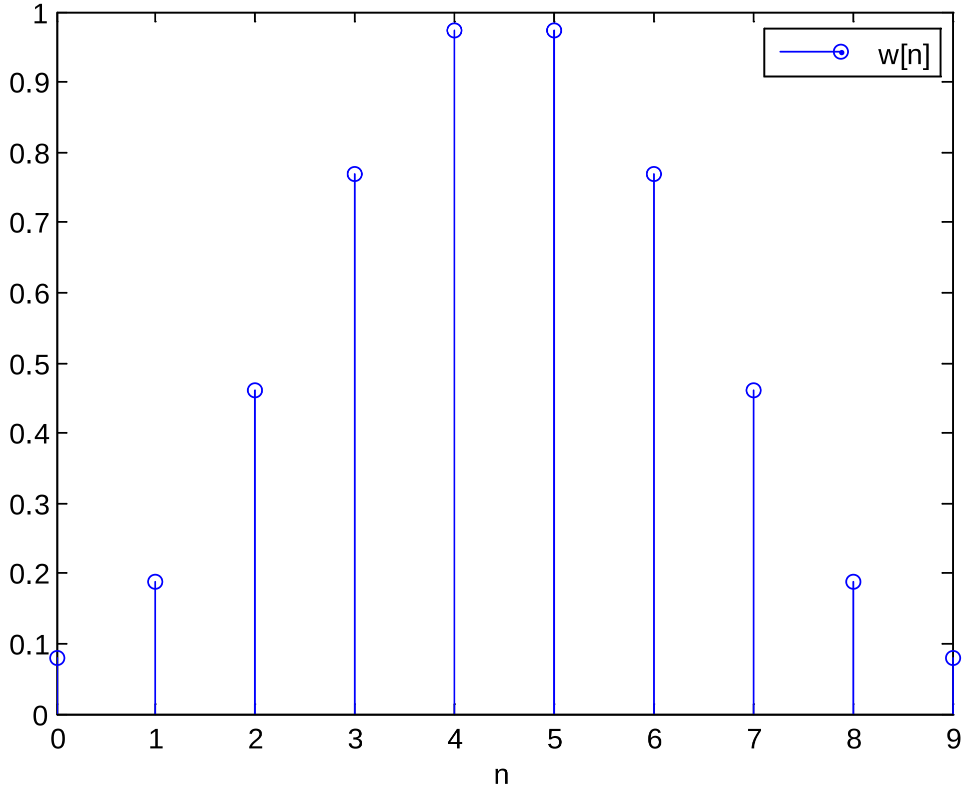


Fig.10.23: Hamming window function for $N = 10$

- Blackman Window

$$w[n] = \begin{cases} 0.42 - 0.5\cos\left(\frac{2\pi n}{N-1}\right) + 0.08\cos\left(\frac{4\pi n}{N-1}\right), & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases} \quad (10.23)$$

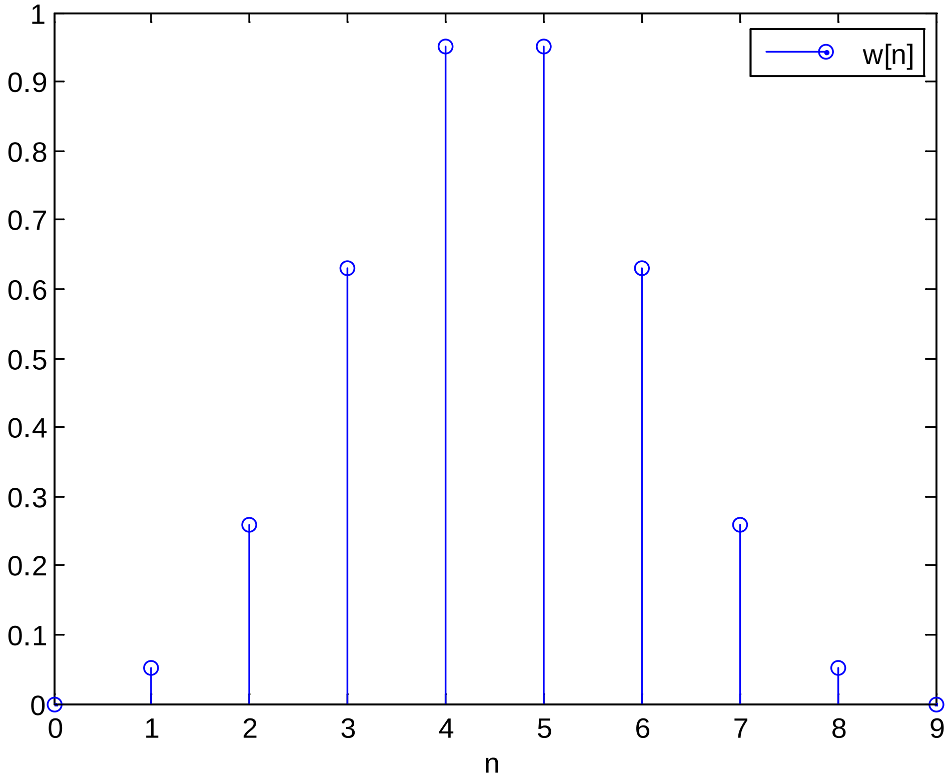


Fig.10.24: Blackman window function for $N=10$

Note that by using $(N+2)$ terms for a filter length of N and then discarding the end points, we can avoid zero coefficients at both ends for Bartlett, Hanning and Blackman windows.

Window	$\Delta\omega_m$	r_p (dB)
Rectangular	$4\pi/N$	-13
Bartlett	$8\pi/N$	-27
Hanning	$8\pi/N$	-32
Hamming	$8\pi/N$	-43
Blackman	$12\pi/N$	-58

Table 10.1: Characteristics of window functions

As a result, there is a tradeoff between transition and ripple

Example 10.9

Use the window method to design a linear-phase and causal FIR system to approximate an ideal lowpass filter whose cutoff frequency is $\omega_c = 0.5\pi$. Try the rectangular and Bartlett window functions with filter lengths of 7, 21, 61 and 101, to investigate the frequency magnitude responses.

Using Example 10.7, $h[n]$ with the rectangular window is:

$$h[n] = \frac{\sin(\omega_c(n-\tau))}{\pi(n-\tau)}, \quad 0 \le n \le N-1$$

with N equals 7, 21, 61 and 101 and $\tau = (N-1)/2$.

Multiplying $h[n]$ by (10.20) yields the corresponding filter coefficients for the Bartlett window.

The MATLAB program is provided as `ex10_9.m`.

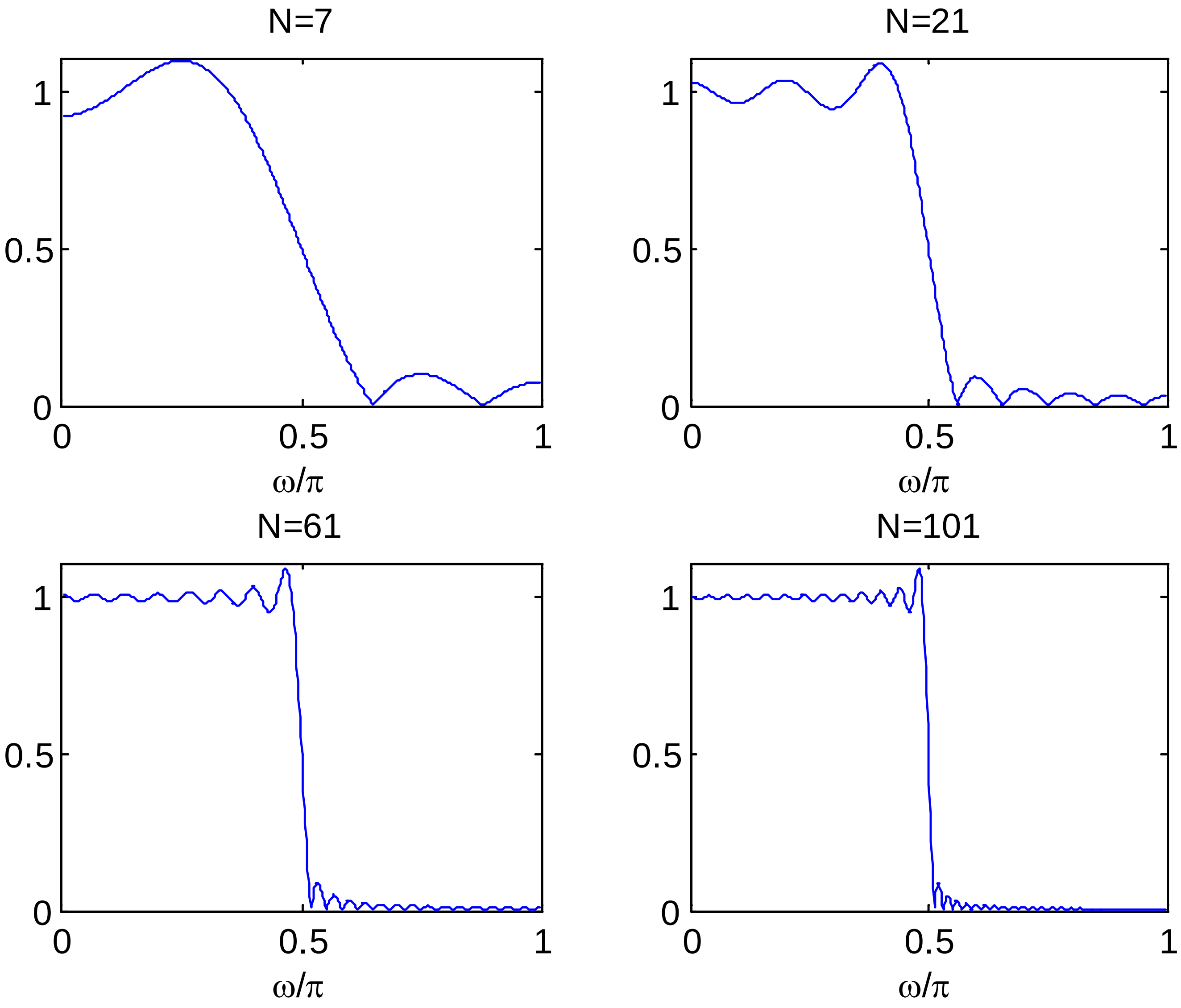


Fig.10.25: Magnitude responses with rectangular window at different N

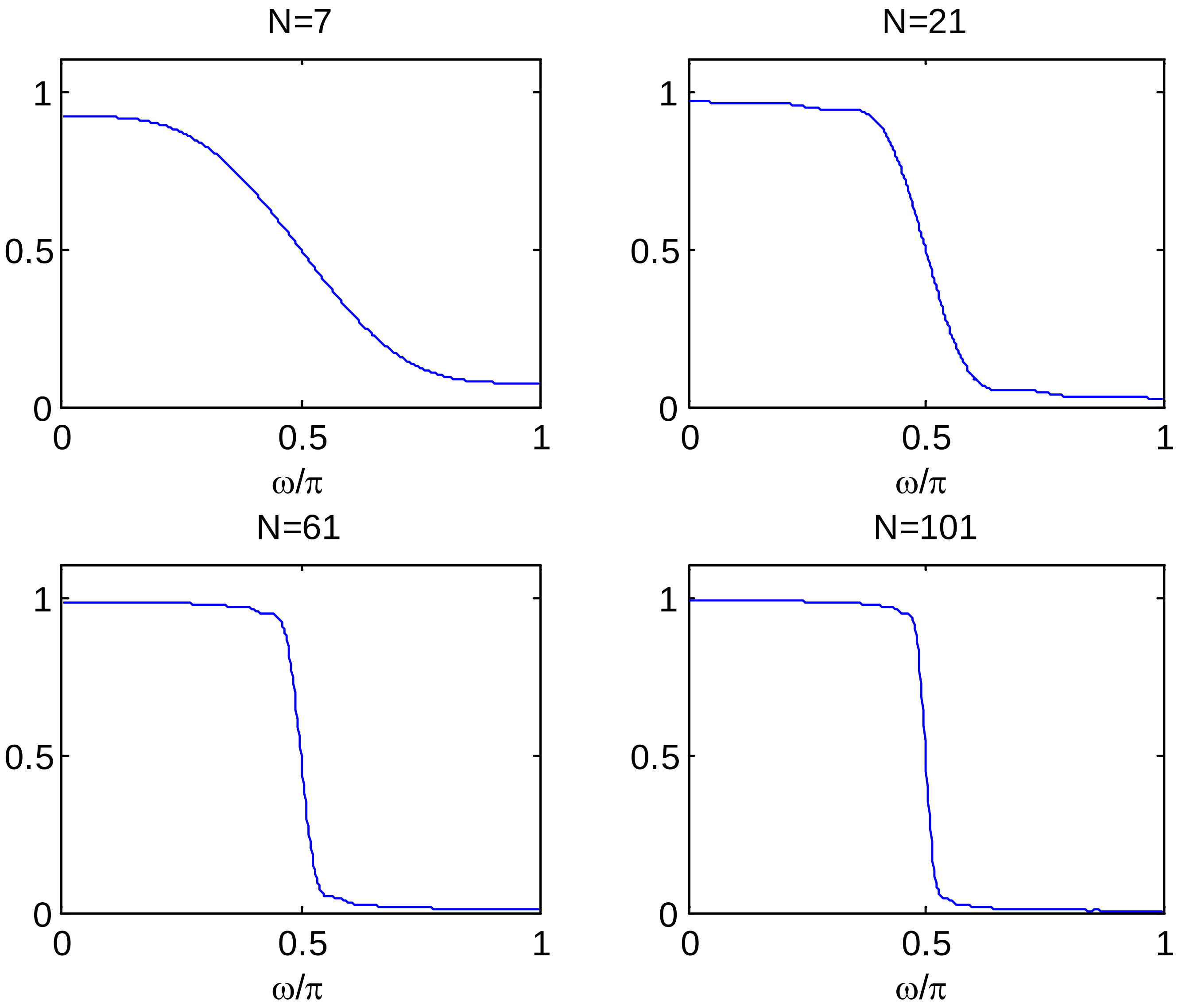


Fig.10.26: Magnitude responses with Bartlett window at different N

Example 10.10

Use the window method to find the impulse response of a linear-phase and causal FIR filter which approximates an ideal lowpass filter whose cutoff frequency is $\omega_c = 0.25\pi$. It is required that the filter is of fourth-order and Bartlett window is employed.

A fourth-order filter implies a filter length of $N = 5$. Using Example 10.7 with $N = 5$ and $\omega_c = 0.25\pi$:

$$h[n] = \frac{\sin(\omega_c(n-2))}{\pi(n-2)} \cdot w[n], \quad 0 \le n \le 4$$

To avoid zero coefficients, we set $N = 7$ in (10.20) and extract the middle 5 values to yield $w[n] = \{1/3, 2/3, 1, 2/3, 1/3\}$ with $w[0] = 1/3$.

As a result

$$h[n] = \{0.0531, 0.1501, 0.25, 0.1501, 0.0531\}$$

with $h[0] = 0.0531$

$H(z)$ is:

$$H(z) = 0.0531 + 0.1501z^{-1} + 0.25z^{-2} + 0.1501z^{-3} + 0.0531z^{-4}$$

The MATLAB command `fir1(4,0.25,triang(5),'noscale')` can also produce $h[n]$.

Note that if we use $N = 5$ in (10.20), $w[n] = \{0, 1/2, 1, 1/2, 0\}$ with $w[0] = 0$. The resultant impulse response will be $h[n] = \{0, 0.1125, 0.25, 0.1125, 0\}$ with $h[0] = 0$, indicating that the effective length is only 3

In window method, the ripple is:

$$\delta = \delta_1 = \delta_2$$

And the transition width is:

$$\Delta\omega = \omega_s - \omega_p$$

with

$$\omega_c = (\omega_p + \omega_s)/2$$

They are determined from $H(e^{j\omega}) = H_d(e^{j\omega})\tilde{\otimes}W(e^{j\omega})$ in (10.15), which is not straightforward for computation

Nevertheless, by noting that (10.15) corresponds to one of the forms in (10.9)-(10.12), $H_r(e^{j\omega})$ is easily obtained by extracting the amplitude

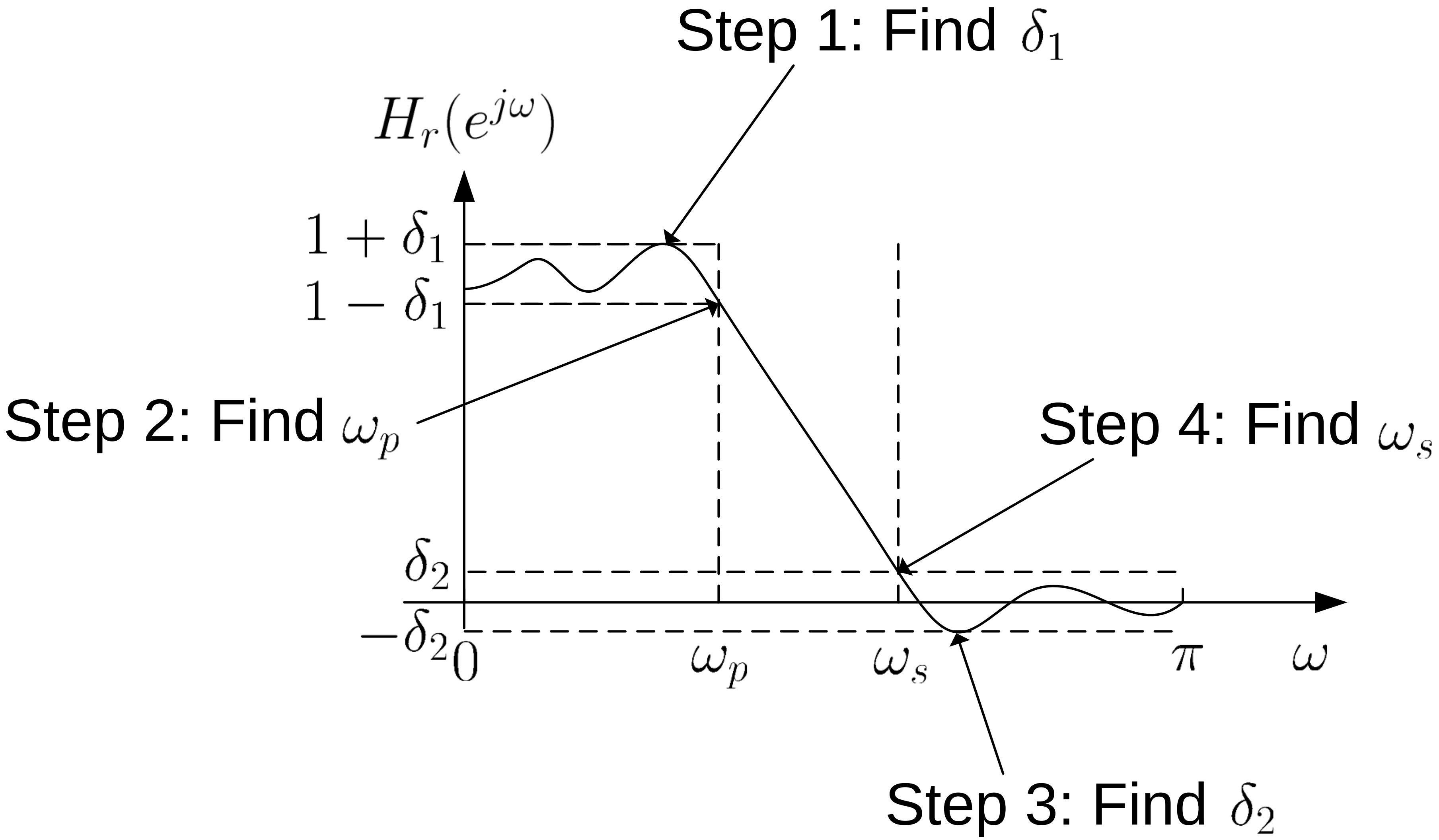


Fig.10.27: Steps to find passband and stopband ripples and frequencies

Example 10.11

Use the window method with the rectangular window to find the impulse response of a linear-phase and causal FIR filter which approximates an ideal lowpass filter whose cutoff frequency is $\omega_c = 0.5\pi$. It is required that the filter has a length of $N = 21$. Plot its amplitude response and then determine δ_1, δ_2, ω_p, ω_s and $\Delta\omega$.

Using Example 10.7 with $N = 21$ and $\omega_c = 0.5\pi$, $h[n]$ is:

$$h[n] = \frac{\sin(0.5\pi(n-10))}{\pi(n-10)}, \quad 0 \le n \le 20$$

As $h[n]$ is symmetric and N is odd, from (10.9):

$$H_r(e^{j\omega}) = 2\sum_{n=0}^{9} h[n]\cos\left((10-n)\,\omega\right) + h[10]$$

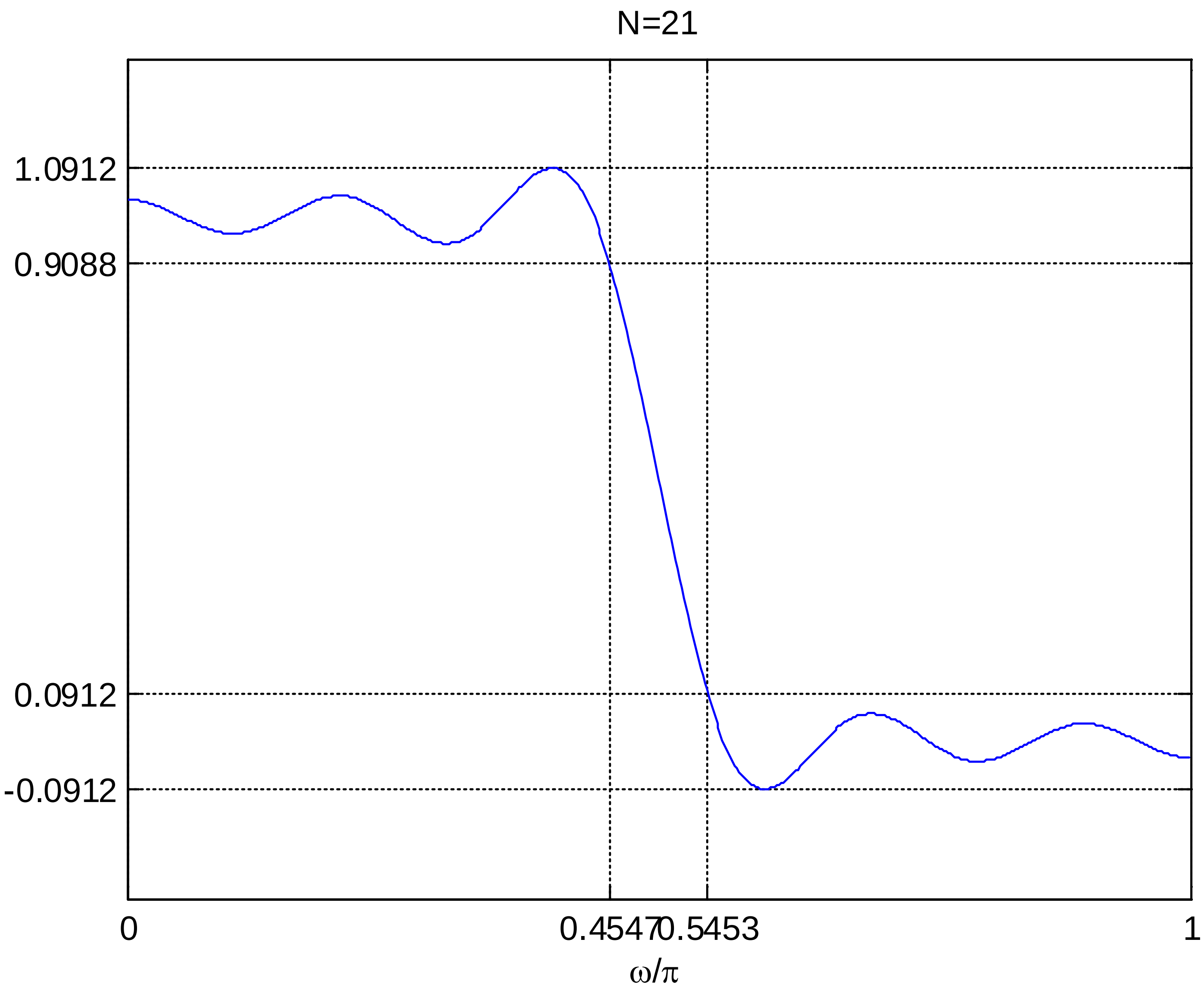


Fig.10.28: Measuring parameters from $H_r(e^{j\omega})$

We see that

$$\delta = \delta_1 = \delta_2 = 0.0912$$

Peak approximation error, e_p, which is the ripple in dB, is:

$$e_p = 20\log_{10}\delta = 20\log_{10}0.0912 = -21 \text{ dB}$$

Furthermore

$$\omega_p = 0.4547\pi$$

and

$$\omega_s = 0.5453\pi$$

which gives

$$\Delta\omega = \omega_s - \omega_p = 0.2846$$

with

$$\omega_c = (\omega_p + \omega_s)/2 = 0.5\pi$$

The MATLAB program is provided as `ex10_11.m`.

Window	$\Delta\omega$	e_p (dB)
Rectangular	$1.8\pi/N$	-21
Bartlett	$6.1\pi/N$	-25
Hanning	$6.2\pi/N$	-44
Hamming	$6.6\pi/N$	-53
Blackman	$11\pi/N$	-74

Table 10.2: Transition width and ripple due to different window functions

Example 10.12

Use the window method to design a linear-phase and causal FIR filter which approximates an ideal lowpass filter whose cutoff frequency is $\omega_c = 0.5\pi$. The maximum allowable transition width is $\Delta\omega = 0.05\pi$ and the maximum allowable tolerance is $\delta = 0.005$.

The ripple of $\delta = 0.005$ corresponds to:

$$e_p = 20\log_{10}(0.005) = -46 \text{ dB}$$

From Table 10.2, Hamming and Blackman windows are the two candidates which can meet the ripple requirement

We choose the former because it involves a shorter filter length. The required length for the Hamming window is:

$$\Delta\omega = 0.05\pi = 6.6\pi/N \Rightarrow N = 132$$

Using Example 10.7 with $N = 132$ and $\omega_c = 0.5\pi$, the filter impulse response is:

$$h[n] = \frac{\sin(0.5\pi(n-65.5))}{\pi(n-65.5)} \cdot w[n], \quad 0 \le n \le 131$$

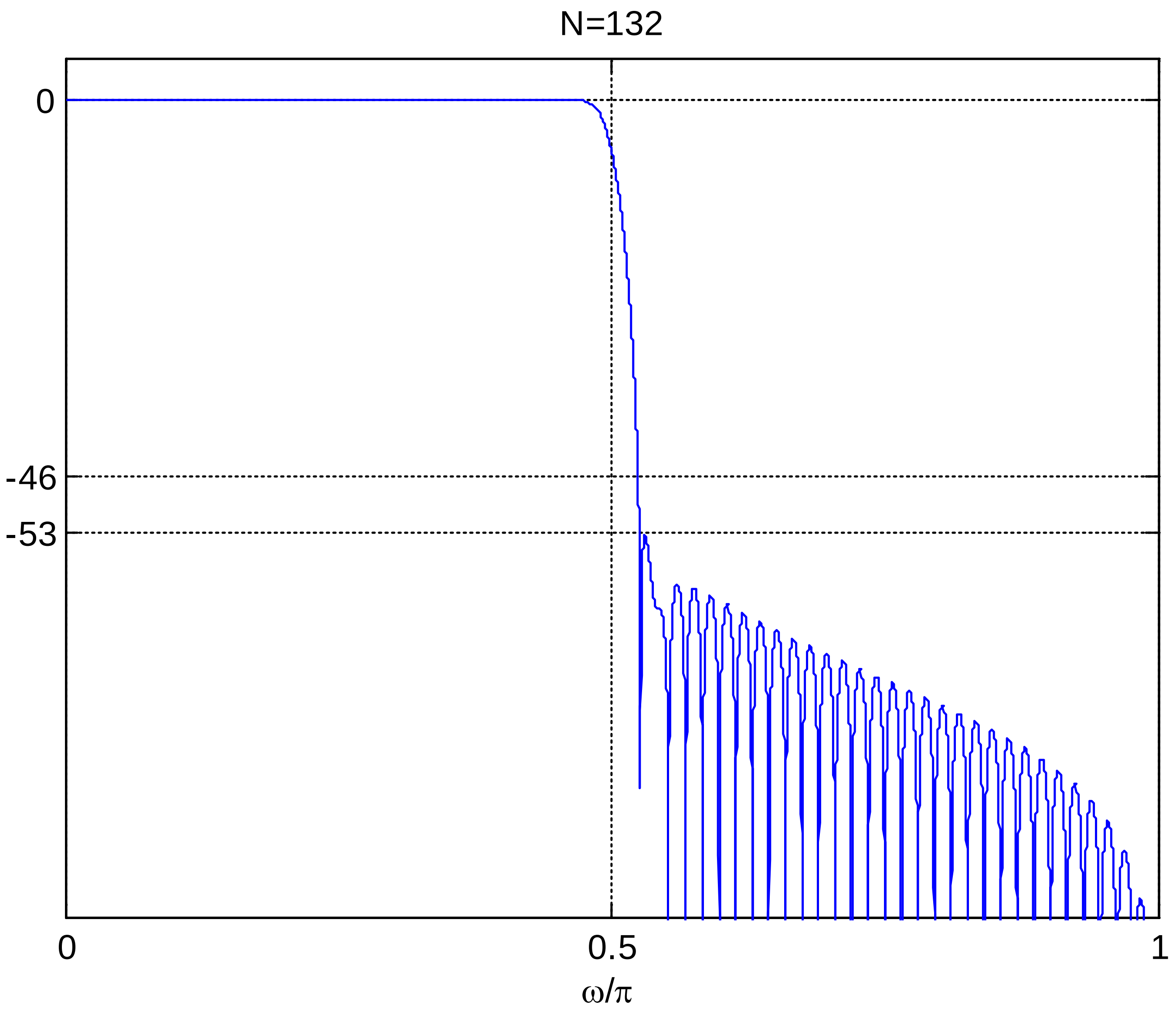


Fig.10.29: Magnitude response with Hamming window

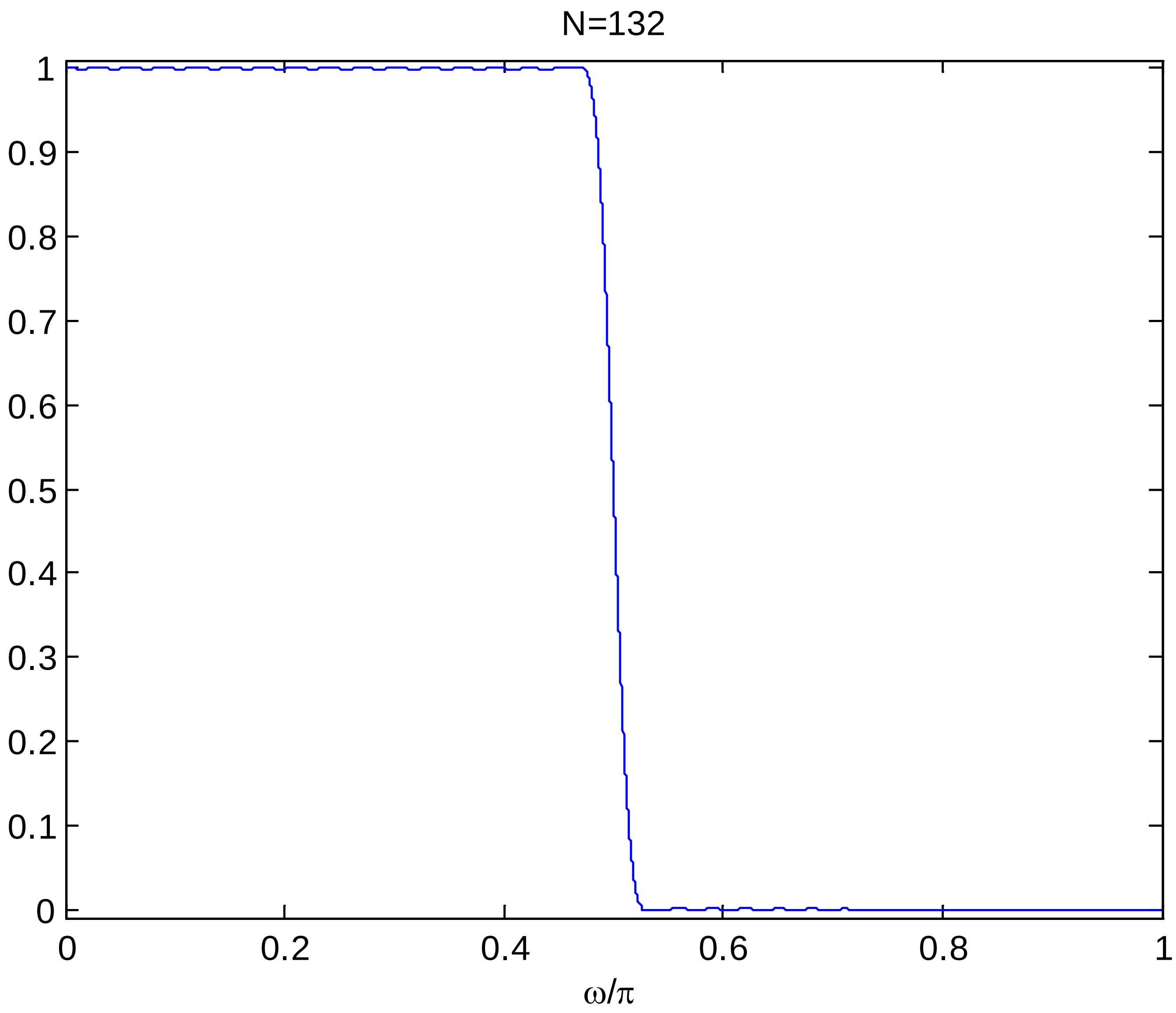


Fig.10.30: Amplitude response with Hamming window

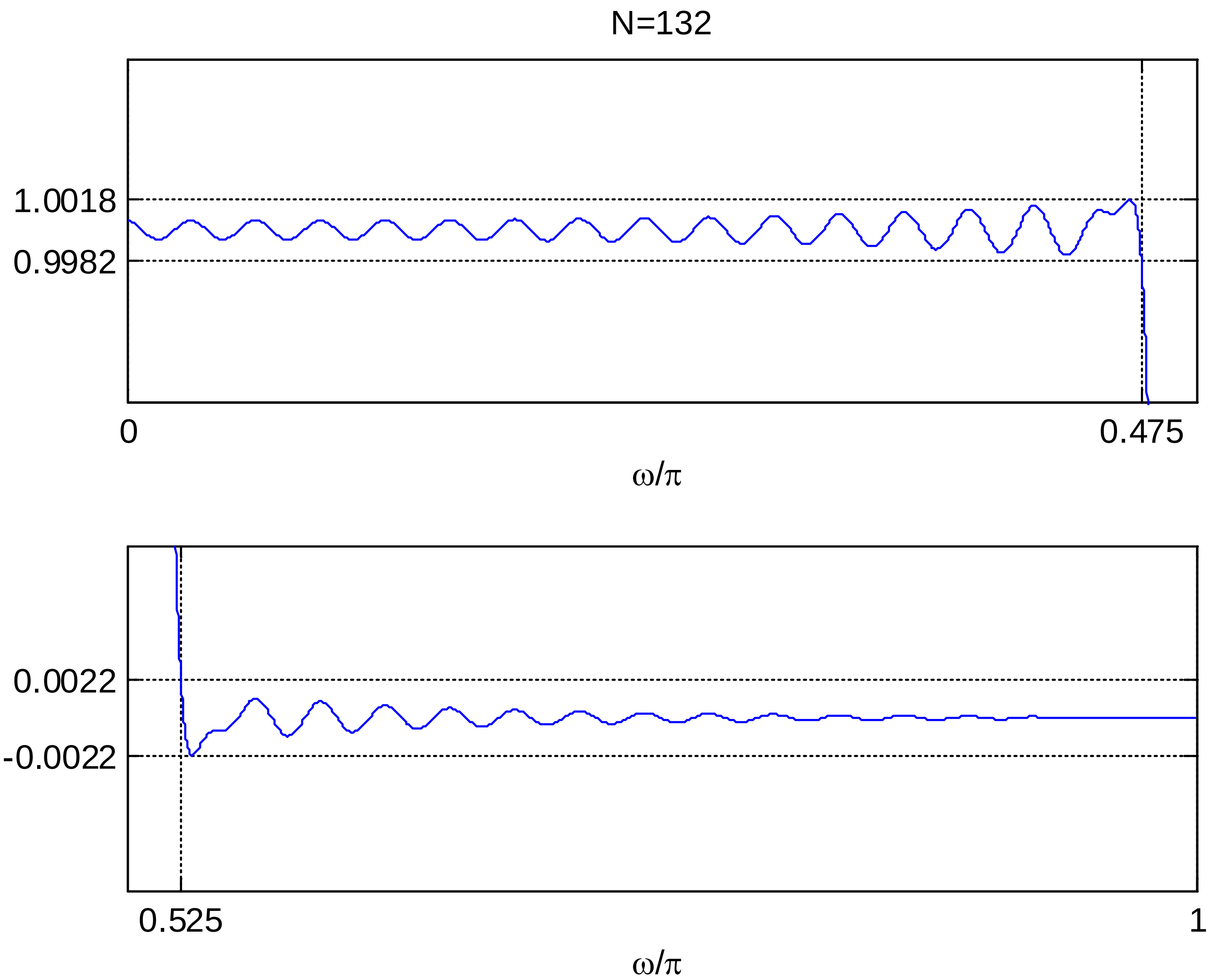


Fig.10.31: Zoomed amplitude responses with Hamming window

In summary:

$$e_p = -53 \text{ dB} \ \text{ or } \ \delta \approx 0.002$$

and

$$\Delta\omega = 0.05\pi$$

with

$$\omega_c = (\omega_p + \omega_s)/2 = 0.5\pi$$

The designed filter meets the given specifications although it has a much smaller ripple than the desired value.

The MATLAB program is provided as `ex10_12.m`.

Extension to Typical Frequency Selective Filter Design

For typical frequency selective filters, namely, highpass, bandpass and bandstop filters, inverse DTFT can be used

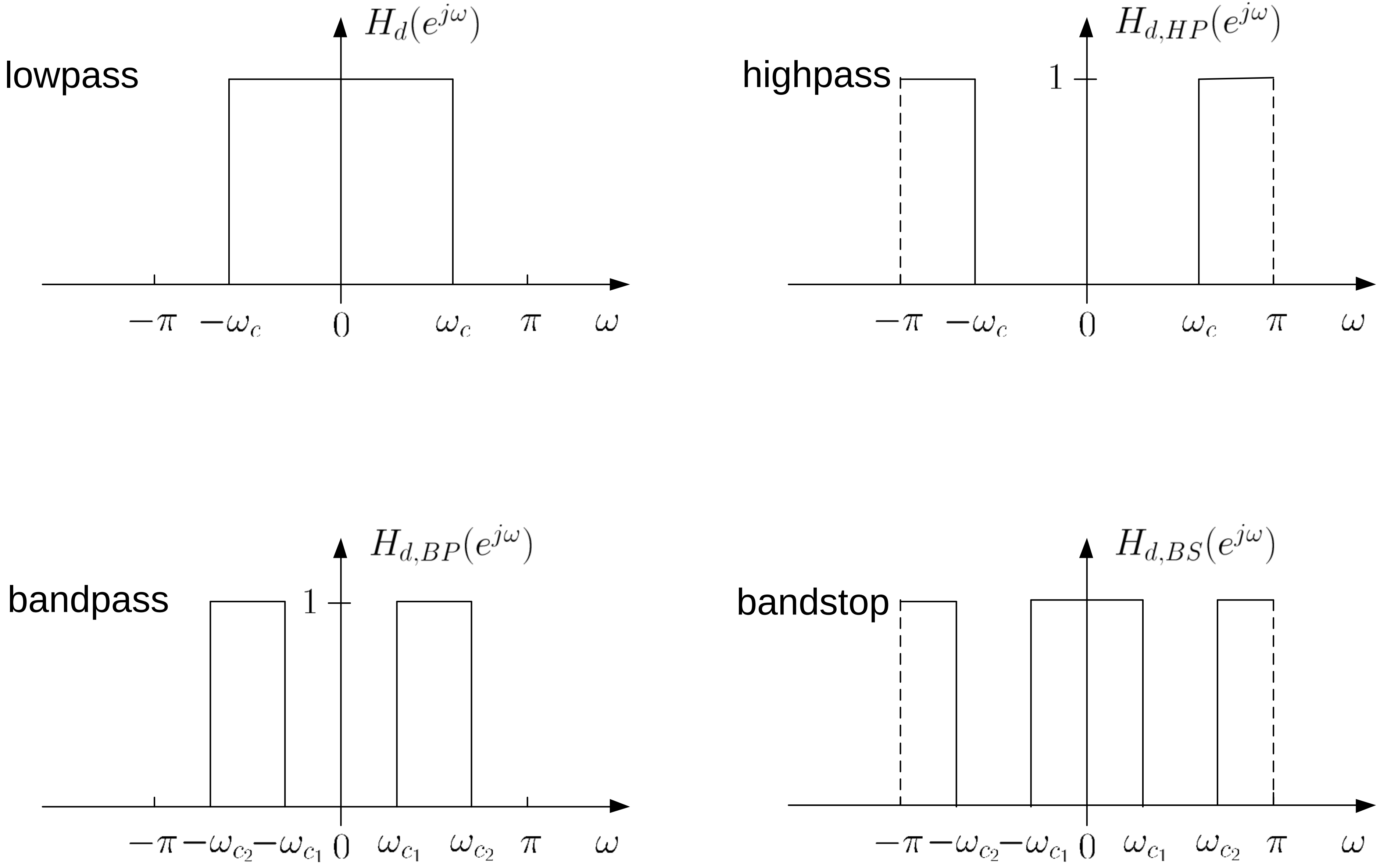


Fig.10.32: Typical frequency-selective filters

Nevertheless, we can utilize the lowpass filter to obtain impulse responses for $H_{d,HP}(e^{j\omega})$, $H_{d,BP}(e^{j\omega})$ and $H_{d,BS}(e^{j\omega})$

A highpass filter can be considered as subtraction of a lowpass filter from an allpass filter:

$$H_{d,HP}(e^{j\omega}) = 1 - H_d(e^{j\omega})$$

$$\Rightarrow h_{d,HP}[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{j\omega n} d\omega - \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} e^{j\omega n} d\omega = \delta[n] - \frac{\omega_c}{\pi}\text{sinc}\left(\frac{\omega_c n}{\pi}\right) \quad (10.24)$$

Assuming rectangular window:

$$h_{HP}[n] = \text{sinc}[n-\tau] - \frac{\omega_c}{\pi}\text{sinc}\left(\frac{\omega_c(n-\tau)}{\pi}\right) \qquad (10.25)$$

where $\tau = (N-1)/2$

Example 10.13

Use the window method with the rectangular window to find the impulse response of a linear-phase and causal FIR filter which approximates an ideal highpass filter whose cutoff frequency is $\omega_c = 0.5\pi$. It is required that the filter has a length of $N = 21$.

According to (10.25) with $N = 21$ and $\omega_c = 0.5\pi$:

$$h_{HP}[n] = \delta[n-10] - 0.5\text{sinc}\left(0.5(n-10)\right), \quad 0 \leq N \leq 20$$

Note that we can also use the MATLAB command `fir1(20,0.5,'high',boxcar(21),'noscale')` to get the same result

The MATLAB program is provided as `ex10_13.m`.

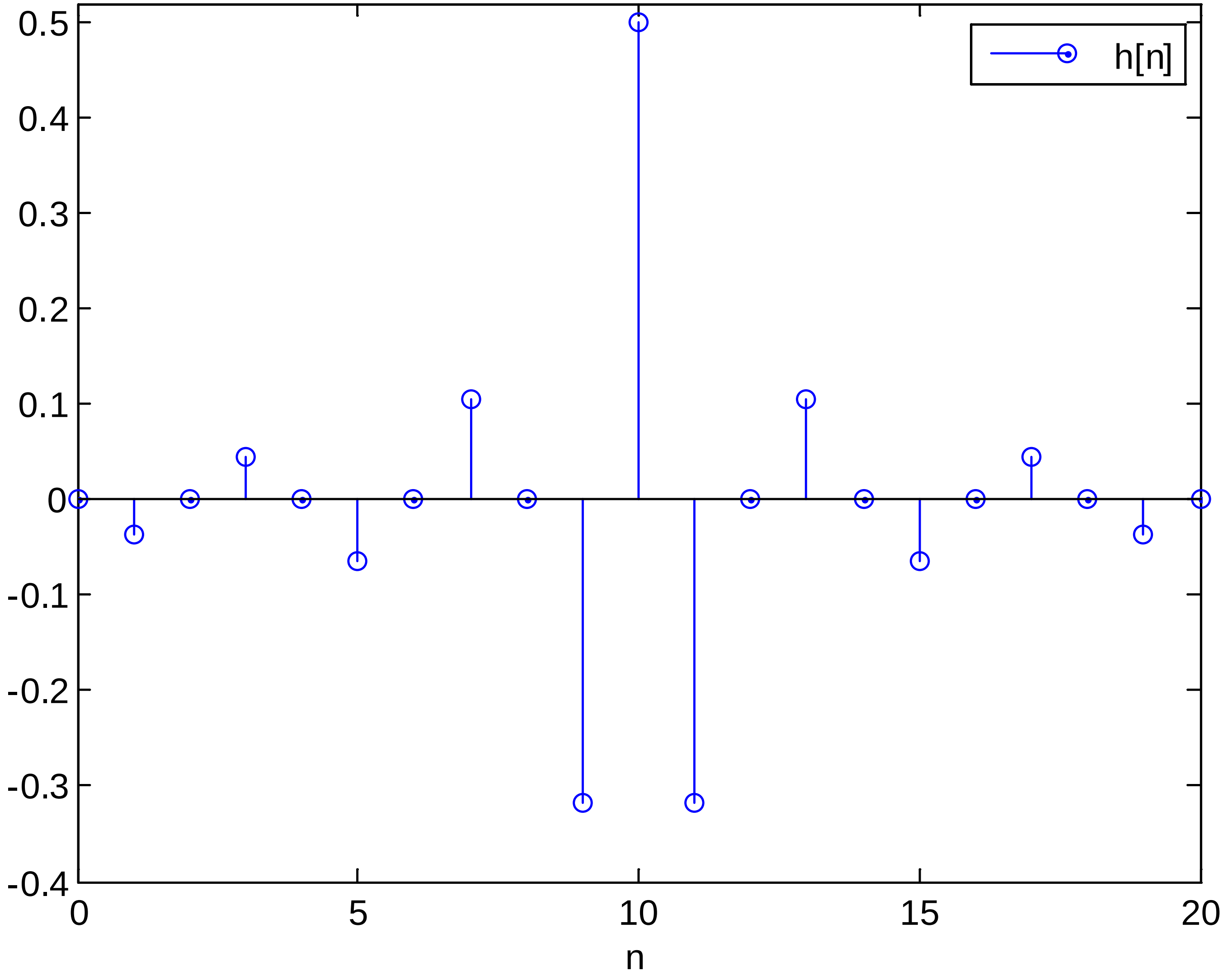


Fig.10.33: Impulse response of highpass filter with $N = 21$

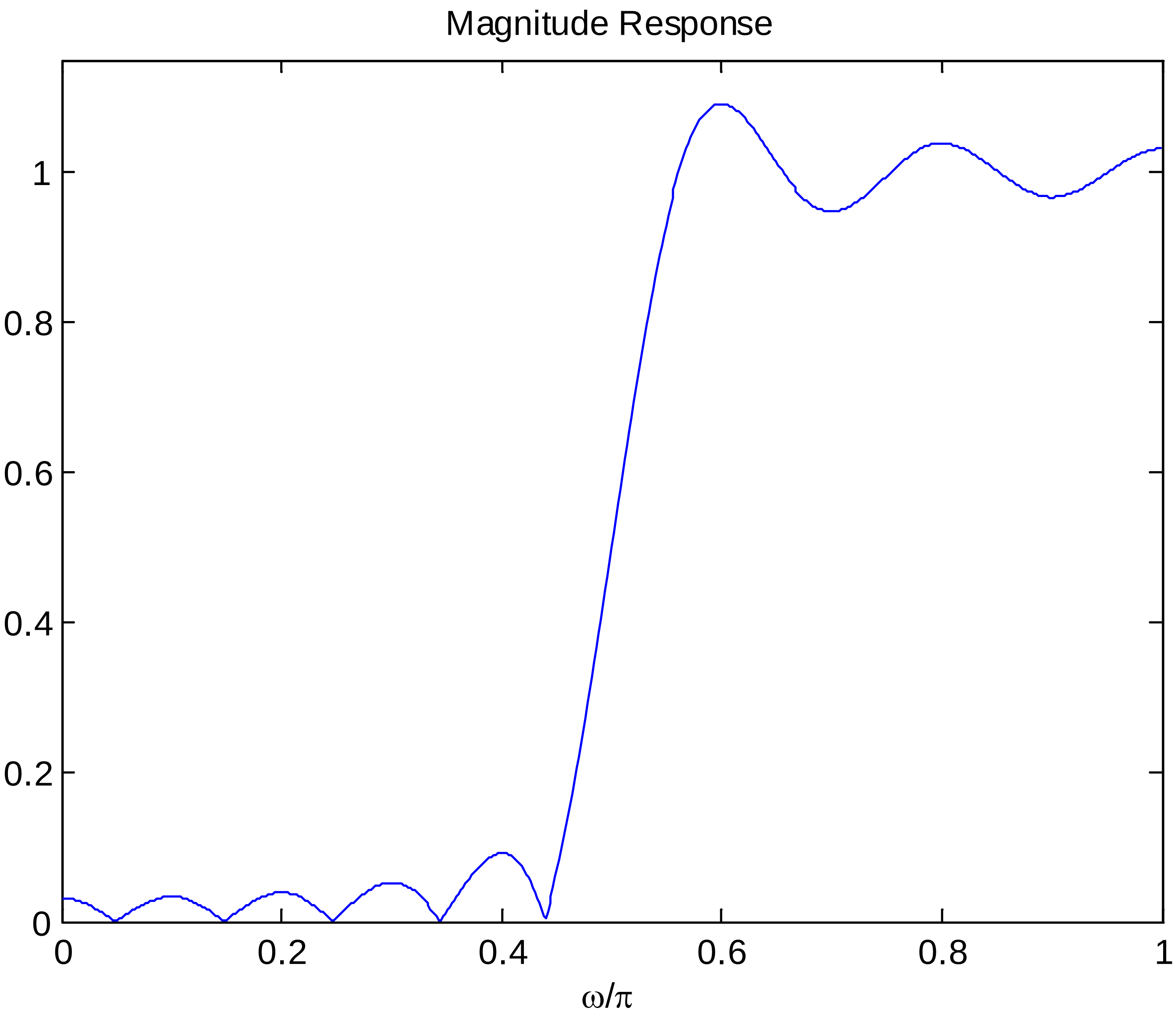


Fig.10.34: Magnitude response of highpass filter with $N = 21$

Similarly, an ideal bandpass filter can be designed from subtraction of two lowpass filters with cutoff frequencies ω_{c_1} and ω_{c_2}:

$$\begin{aligned} h_{d,BP}[n] &= \frac{1}{2\pi}\int_{-\omega_{c_2}}^{\omega_{c_2}} e^{j\omega n}d\omega - \frac{1}{2\pi}\int_{-\omega_{c_1}}^{\omega_{c_1}} e^{j\omega n}d\omega \\ &= \frac{\omega_{c_2}}{\pi}\text{sinc}\left(\frac{\omega_{c_2}n}{\pi}\right) - \frac{\omega_{c_1}}{\pi}\text{sinc}\left(\frac{\omega_{c_1}n}{\pi}\right) \end{aligned} \qquad (10.26)$$

An ideal bandstop filter impulse response can be obtained by subtracting a bandpass filter from an allpass filter:

$$h_{d,BS}[n] = \delta[n] - \frac{\omega_{c_2}}{\pi}\text{sinc}\left(\frac{\omega_{c_2}n}{\pi}\right) + \frac{\omega_{c_1}}{\pi}\text{sinc}\left(\frac{\omega_{c_1}n}{\pi}\right) \qquad (10.27)$$

Kaiser Window

A problem in typical window functions is that we cannot control the ripple δ

Kaiser window can control both δ and $\Delta\omega$ in a nearly optimal manner and it has the form of:

$$w[n]=\frac{I_0\left(\beta\sqrt{1-\left(1-\frac{2n}{N-1}\right)^2}\right)}{I_0(\beta)},\quad 0\leq n\leq N-1 \qquad (10.28)$$

where I_0 is the modified zero-order Bessel function. Apart from N, there is β which is responsible for the window shape

That is, by properly choosing N and β, the design specifications in terms of δ and $\Delta\omega$ can be precisely met

N is computed as:

$$N = \left\lceil \frac{-e_p - 8}{2.285\Delta\omega} \right\rceil + 1 \qquad (10.29)$$

where $e_p = 20\log_{10}\delta$ is the peak approximation error and $\lceil u \rceil$ rounds up u to the nearest integer

β is determined from:

$$\beta = \begin{cases} 0.1102(-e_p - 8.7), & -e_p > 50 \\ 0.5842(-e_p - 21)^{0.4} + 0.07886(-e_p - 21), & 21 \le -e_p \le 50 \\ 0, & -e_p < 21 \end{cases} \qquad (10.30)$$

Example 10.14

Use the window method with the Kaiser window to design a linear-phase and causal FIR filter which approximates an ideal lowpass filter whose cutoff frequency is $\omega_c = 0.5\pi$. The maximum allowable transition width is $\Delta\omega = 0.05\pi$ and the maximum allowable tolerance is $\delta = 0.005$.

Since

$$-e_p = -20\log_{10}(0.005) = 46$$

and $\Delta\omega = 0.05\pi$, we then have:

$$N = \left\lceil \frac{46-8}{2.285 \cdot 0.05\pi} \right\rceil + 1 = 107$$

and

$$\beta = 0.5842(46-21)^{0.4} + 0.07886(46-21) = 4.09$$

As a result, the filter impulse response is:

$$h[n] = \frac{\sin(0.5\pi(n-53))}{\pi(n-53)} \cdot w[n], \quad 0 \le n \le 106$$

where $w[n]$ is determined from (10.28).

In summary:

$$e_p = -46 \text{ dB} \;\; \text{or} \;\; \delta \approx 0.005$$

$$\Delta\omega = 0.05\pi$$

with

$$\omega_c = (\omega_p + \omega_s)/2 = 0.5\pi$$

The designed filter exactly meets the given specifications

The MATLAB program is provided as `ex10_14.m`.

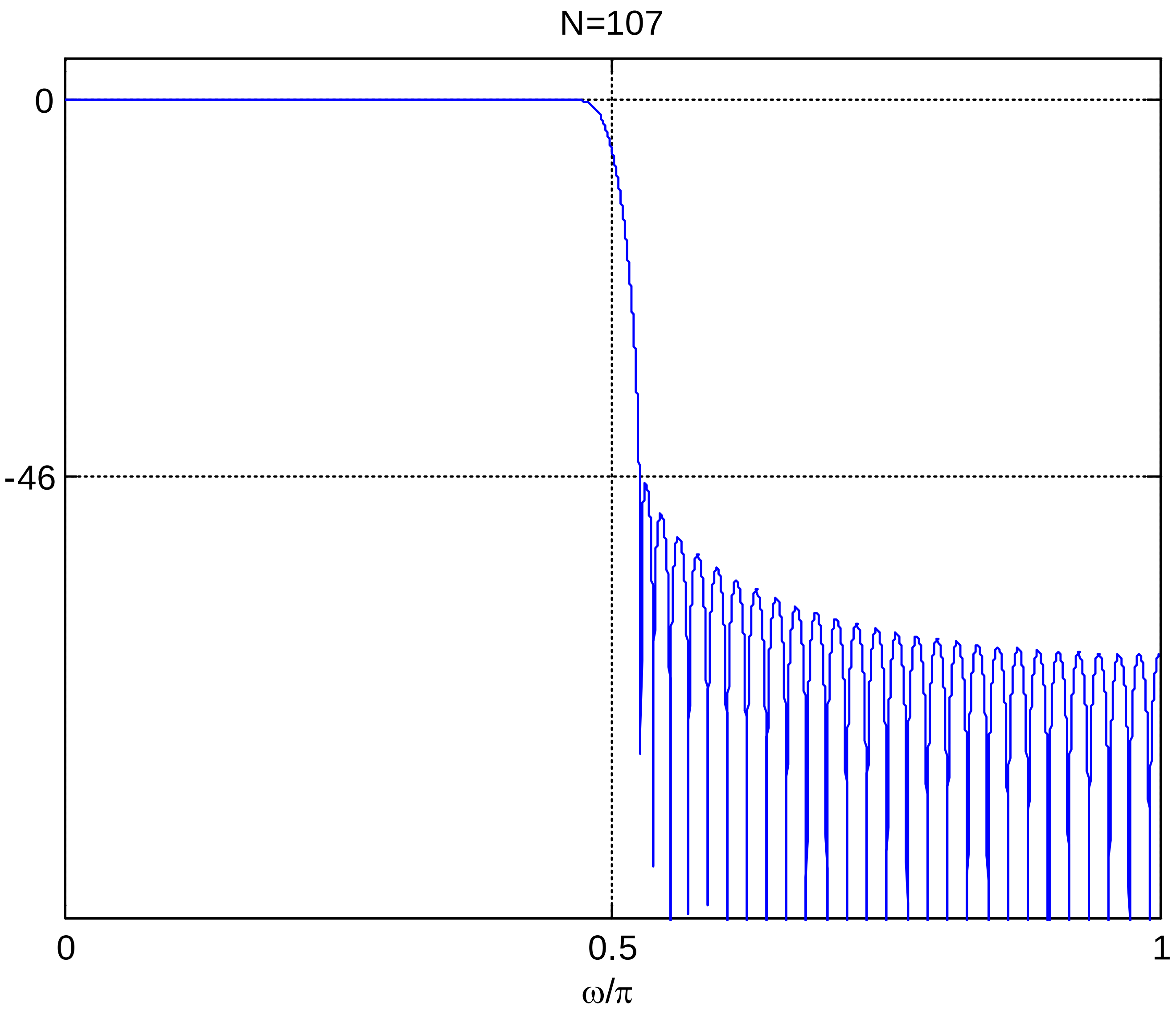


Fig.10.35: Magnitude response with Kaiser window

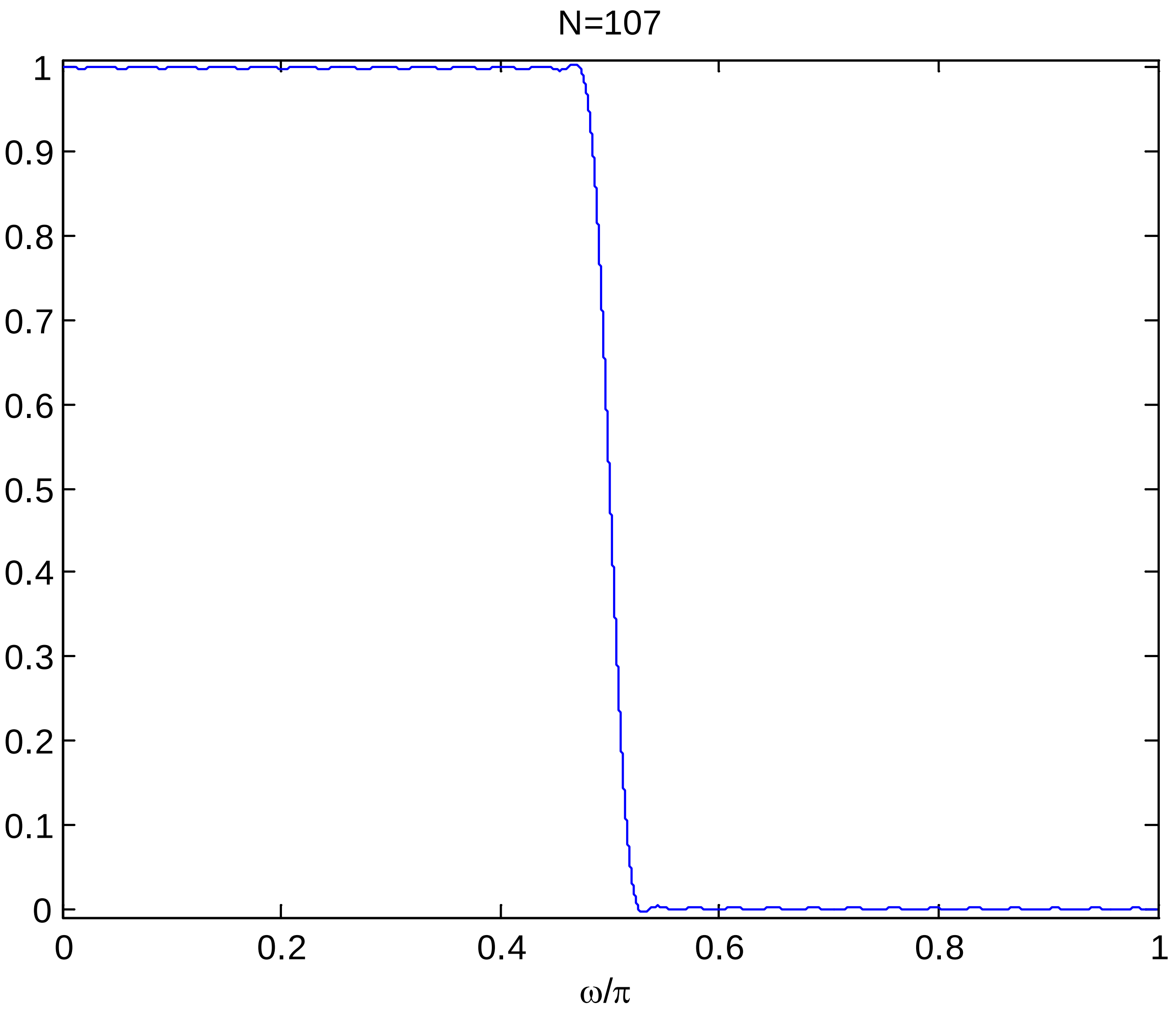


Fig.10.36: Amplitude response with Kaiser window

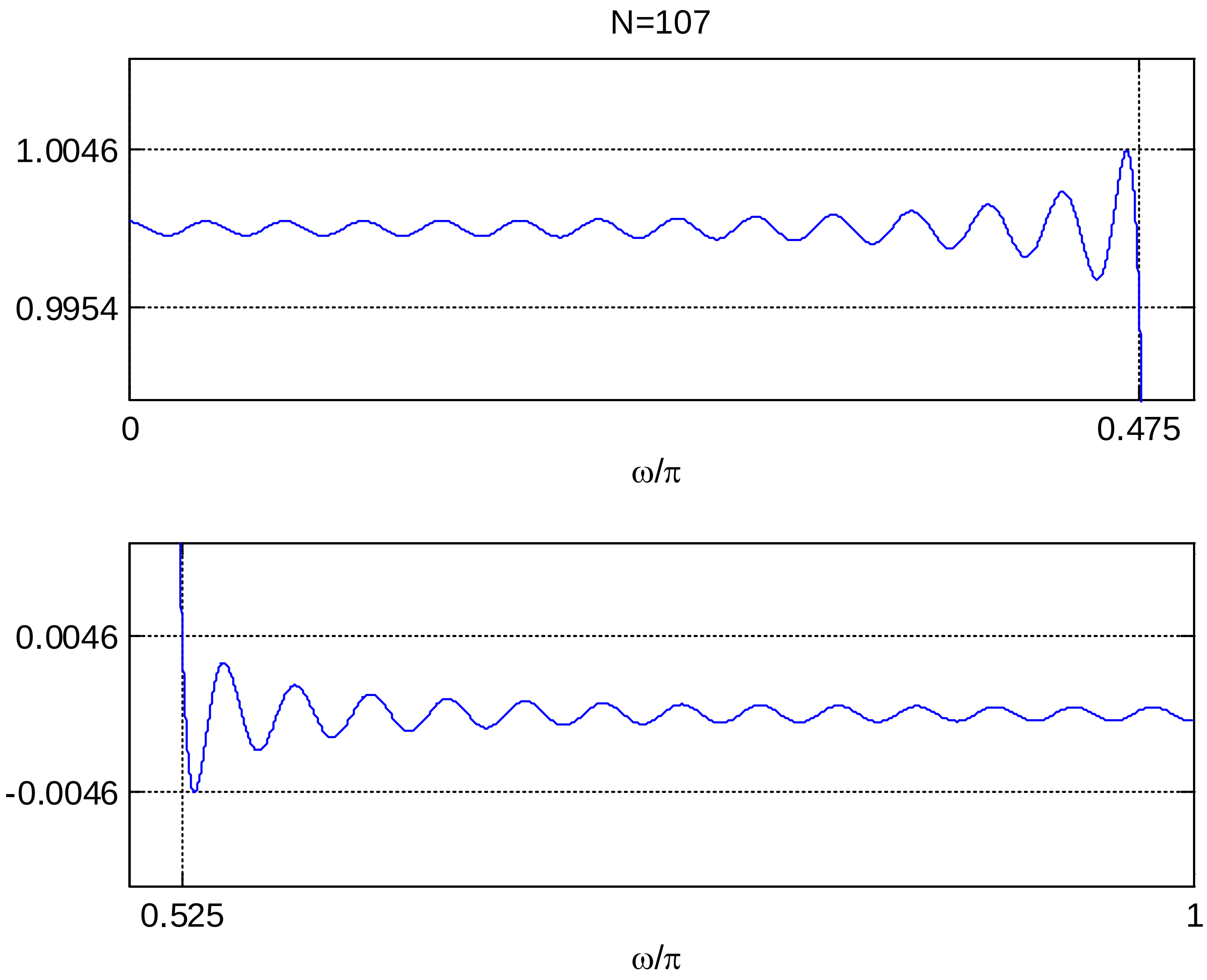


Fig.10.37: Zoomed amplitude responses with Kaiser window

Frequency Sampling Method

The basic idea is to utilize the discrete Fourier transform (DFT), which corresponds to samples of the desired frequency response $H_d(e^{j\omega})$, to produce $h[n]$

Recall:

$$H[k] = H_d(e^{j\omega})|_{\omega=\frac{2\pi k}{N}}, \quad k = 0, 1, \cdots, N-1 \tag{10.31}$$

is equal to $H_d(e^{j\omega})$ sampled at N distinct frequencies between $\omega \in [0, 2\pi]$ with a uniform frequency spacing of $2\pi/N$

A causal and linear-phase filter is designed with 2 steps:

- Extract N uniformly-spaced samples from $H_d(e^{j\omega})$ in the frequency range of $\omega \in [0, 2\pi]$
- Compute $h[n]$ by taking the inverse DFT of $H[k]$:

$$h[n] = \frac{1}{N}\sum_{k=0}^{N-1} H[k] W_N^{-kn}, \quad W_N = e^{-j2\pi/N},\ 0 \le n \le N-1 \qquad (10.32)$$

Taking z transform of $h[n]$:

$$\begin{aligned} H(z) &= \sum_{n=0}^{N-1} h[n] z^{-n} \\ &= \sum_{n=0}^{N-1} \left(\frac{1}{N} \sum_{k=0}^{N-1} H[k] e^{j2\pi kn/N} \right) z^{-n} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} H[k] \sum_{n=0}^{N-1} \left(e^{j(2\pi/N)k} z^{-1} \right)^n \\ &= \frac{1-z^{-N}}{N} \sum_{k=0}^{N-1} \frac{H[k]}{1 - e^{j(2\pi/N)k} z^{-1}} \qquad (10.33) \end{aligned}$$

The filter frequency response is then:

$$H(e^{j\omega}) = \frac{1-e^{-j\omega N}}{N}\sum_{k=0}^{N-1}\frac{H[k]}{1-e^{j(2\pi/N)k}e^{-j\omega}} \qquad (10.34)$$

From (10.33) and (7.7) together with (10.31):

$$H(e^{j2\pi r/N}) = \frac{1}{N}\sum_{k=0}^{N-1}H[k]\sum_{n=0}^{N-1}e^{j(2\pi/N)(k-r)n} = H[r]$$

$$\Rightarrow H(e^{j\omega})\big|_{\omega=\frac{2\pi k}{N}} = H[k] = H_d(e^{j\omega})\big|_{\omega=\frac{2\pi k}{N}}, \quad k=0,1,\cdots,N-1 \qquad (10.35)$$

It is simple as only uniformly-spaced samples of the desired frequency response or the DFT coefficients are needed

At $\omega=(2\pi/N)k$, $k=0,1,\cdots,N-1$, $H(e^{j\omega})$ equals $H_d(e^{j\omega})$ but we cannot control the values of the remaining frequency points

It lacks flexibility in specifying the passband and stopband cutoff frequencies since placement of “1” & “0” & transition samples is constrained to integer multiples of $2\pi/N$

Example 10.15

Use the frequency sampling method to design a linear-phase and causal FIR filter with a length of $N=7$ to approximate an ideal lowpass filter whose cutoff frequency is $\omega_c = 0.1\pi$.

From Example 10.7, the ideal frequency response with linear phase is

$$H_d(e^{j\omega}) = \begin{cases} 1 \cdot e^{-j\omega\tau}, & |\omega| < \omega_c \\ 0, & -\pi < \omega < -\omega_c, \quad \omega_c < \omega < \pi \end{cases}$$

where $\tau = (N-1)/2 = 3$ and $\omega_c = 0.1\pi$

We consider the frequency interval of $\omega \in [0, 2\pi]$:

$$H_d(e^{j\omega}) = \begin{cases} 1 \cdot e^{-j\omega\tau}, & 0 < \omega < \omega_c \\ 0, & \omega_c < \omega < 2\pi - \omega_c \\ 1 \cdot e^{-j(\omega-2\pi)\tau}, & 2\pi - \omega_c < \omega < 2\pi \end{cases}$$

Extracting the values of $H_d(e^{j\omega})$ at $\omega = (2\pi/7)k$, $k = 0, 1, \cdots, 6$:

$$H[k] = \begin{cases} 1, \ k = 0 \\ 0, \ k = 1, 2, \cdots, 6 \end{cases}$$

Taking the inverse DFT:

$$h[n] = \frac{1}{7}, \quad n = 0, 1, \cdots, 6$$

The MATLAB program is provided as `ex10_15.m`.

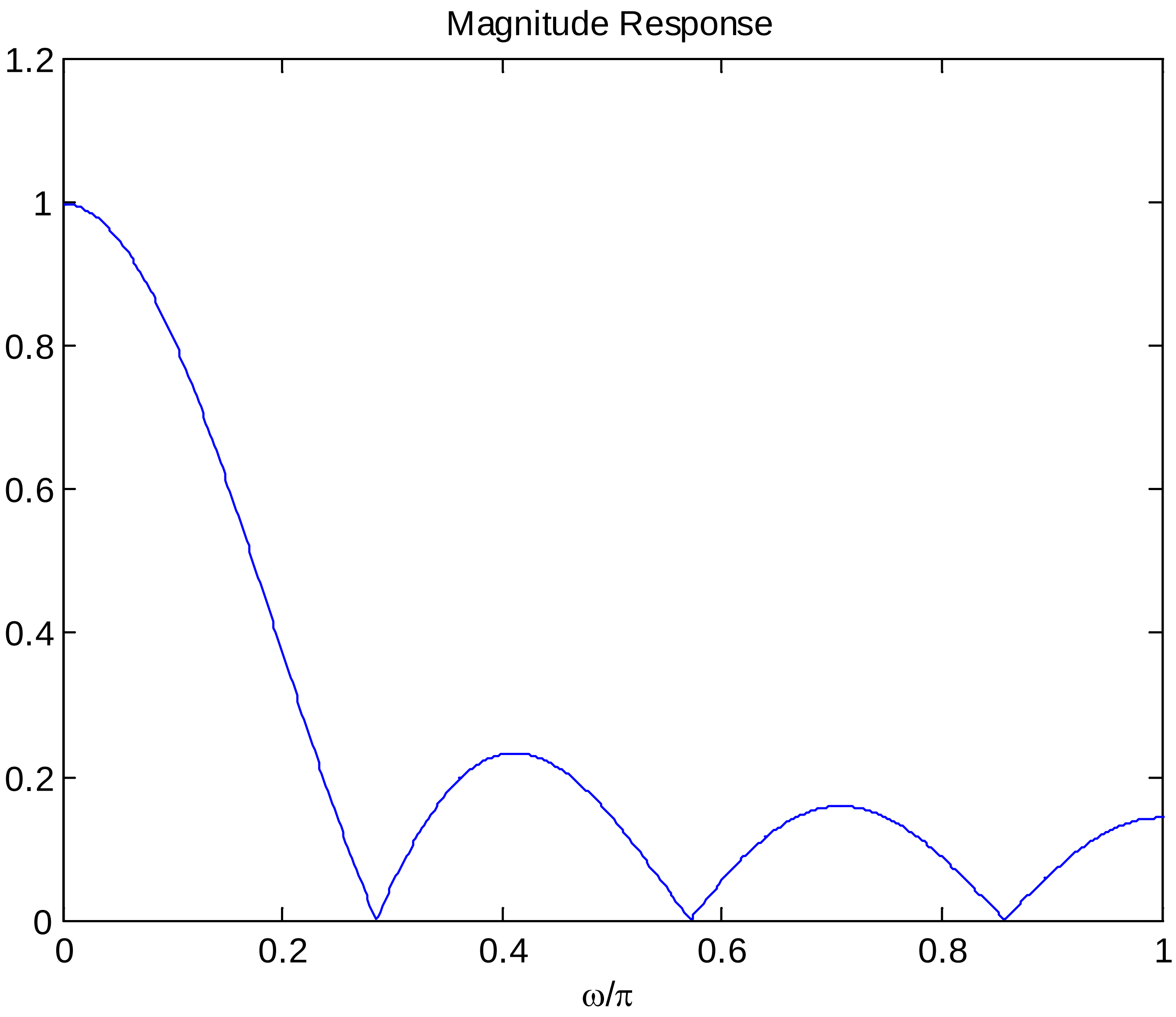


Fig.10.38: Magnitude response based on frequency sampling

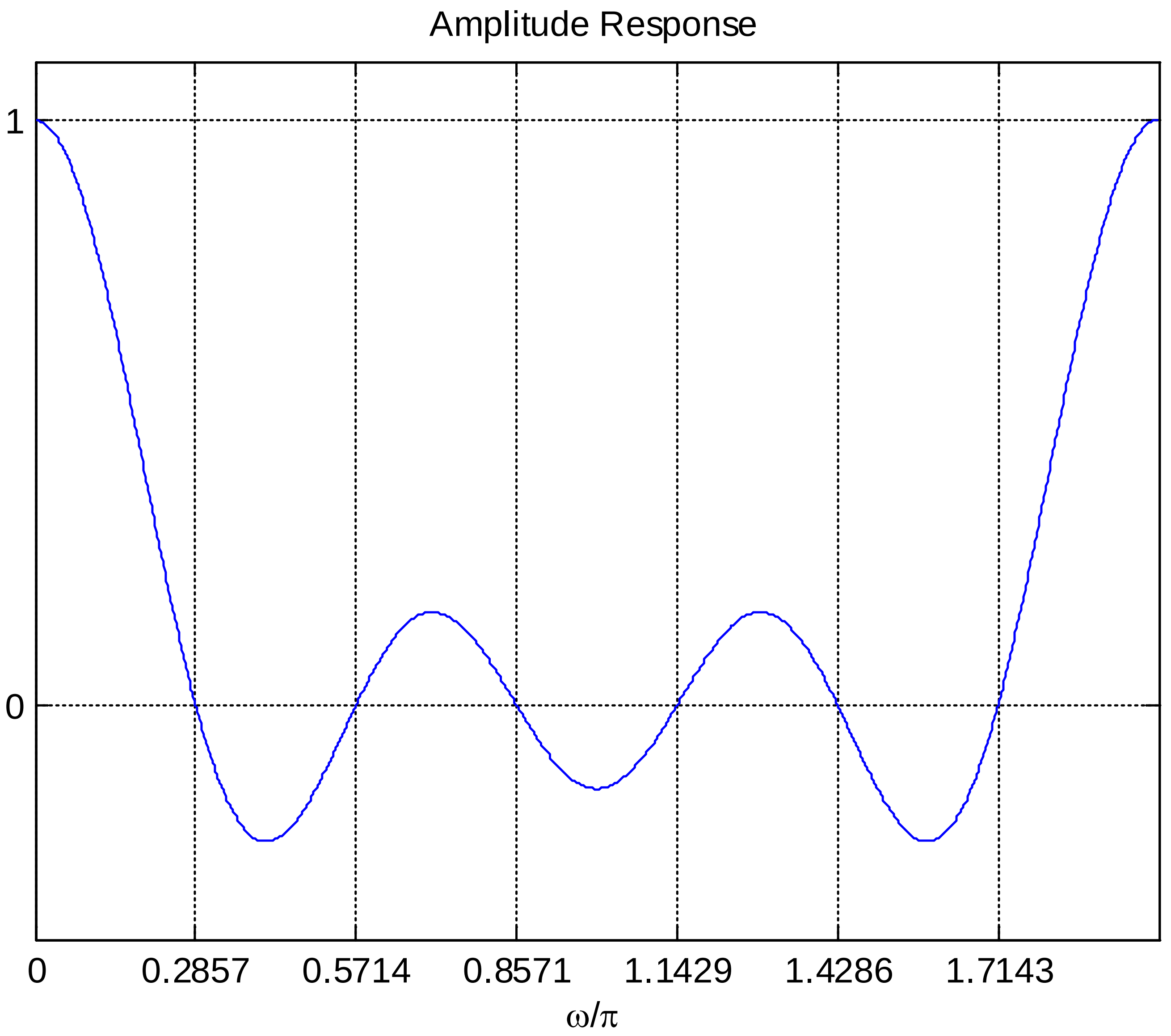


Fig.10.39: Amplitude response based on frequency sampling

Optimal Equiripple Method

The basic idea is to evenly distribute the ripples in both passband and stopband

- Required filter length will be shorter than that of the window method where its $H(e^{j\omega})$ exactly meets the passband or stopband ripple specification at one frequency and is superior to it at other frequencies in the band

- Allow $\delta_1 \neq \delta_2$

- Passband and stopband frequencies can be precisely specified although $\omega_p = \omega_c - \Delta\omega/2$ and $\omega_s = \omega_c + \Delta\omega/2$ are implicitly implied in the window method

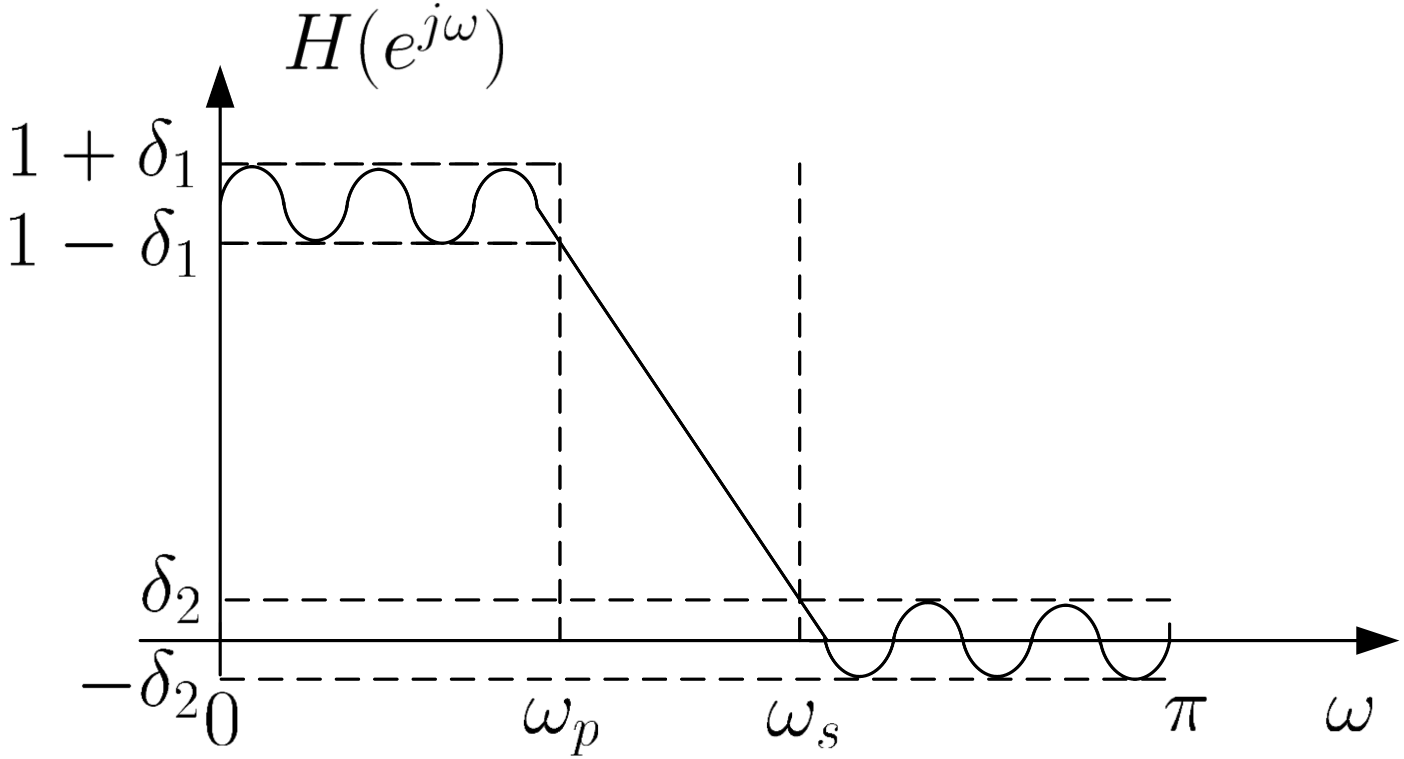


Fig.10.40: Illustration of optimal equiripple lowpass filter

Ripples are uniformly distributed such that $H(e^{j\omega})$ reaches its maximum deviations of δ_1 and δ_2 more than once

The impulse response $h[n]$ of optimal equiripple design is determined from:

$$h[n] = \min_{\{\tilde{h}[n]\}} \left(\max_{0 \le \omega \le \omega_p, \omega_s \le \omega \le \pi} |E(e^{j\omega})| \right), \quad n = 0, 1, \cdots, N-1 \quad (10.36)$$

where

$$E(e^{j\omega}) = W(e^{j\omega})(H_d(e^{j\omega}) - \tilde{H}(e^{j\omega})) \quad (10.37)$$

which corresponds to a minmax optimization problem

$E(e^{j\omega})$ is the frequency-domain error between the desired and actual responses weighted by $W(e^{j\omega})$

$W(e^{j\omega})$ is the weighting function incorporates all specification parameters, namely, δ_1, δ_2, ω_p and ω_s, into the design process

For example, in lowpass filter design, $W(e^{j\omega})$ has the form of:

$$W(e^{j\omega}) = \begin{cases} \dfrac{\delta_2}{\delta_1}, & 0 \le \omega \le \omega_p \\ 1, & \omega_s \le \omega \le \pi \end{cases} \tag{10.38}$$

When $\delta_1 > \delta_2$, there is a larger weighting at the stopband. On the other hand, $\delta_1 < \delta_2$ implies a larger weighting at the passband

To solve for the minmax problem of (10.36), we can make use of the Parks-McClellan algorithm which requires iterations. The corresponding MATLAB command is `firpm` where the filter length parameter N is also required.

We can employ

$$N = \left\lceil \frac{-10\log_{10}(\delta_1\delta_2) - 13}{2.324(\omega_s - \omega_p)} \right\rceil \quad (10.39)$$

to get its initial estimate and then compute $h[n]$. If the tolerance specifications are not met, we increment N until the maximum deviations are bounded by δ_1 and δ_2.

Example 10.16

Use the optimal equiripple method to design a linear-phase and causal FIR filter which approximates an ideal lowpass filter whose passband frequency is $\omega_p = 0.475\pi$ and stopband frequency is $\omega_p = 0.525\pi$. The maximum allowable tolerance is $\delta = \delta_1 = \delta_2 = 0.005$ in both passband and stopband.

According to (10.39), an initial value of N is computed as:

$$N = \left\lceil \frac{-10\log_{10}(0.005 \cdot 0.005) - 13}{2.324(0.525\pi - 0.425\pi)} \right\rceil = 91$$

Starting with $N = 91$ in the Parks-McClellan algorithm, we increment its value until $N = 96$ so that the tolerance specifications are met.

The MATLAB program is provided as `ex10_16.m`.

In summary:

$$e_p = -46 \text{ dB} \quad \text{or} \quad \delta_1 = \delta_2 = 0.005$$

$$\omega_p = 0.475\pi, \ \omega_s = 0.525\pi$$

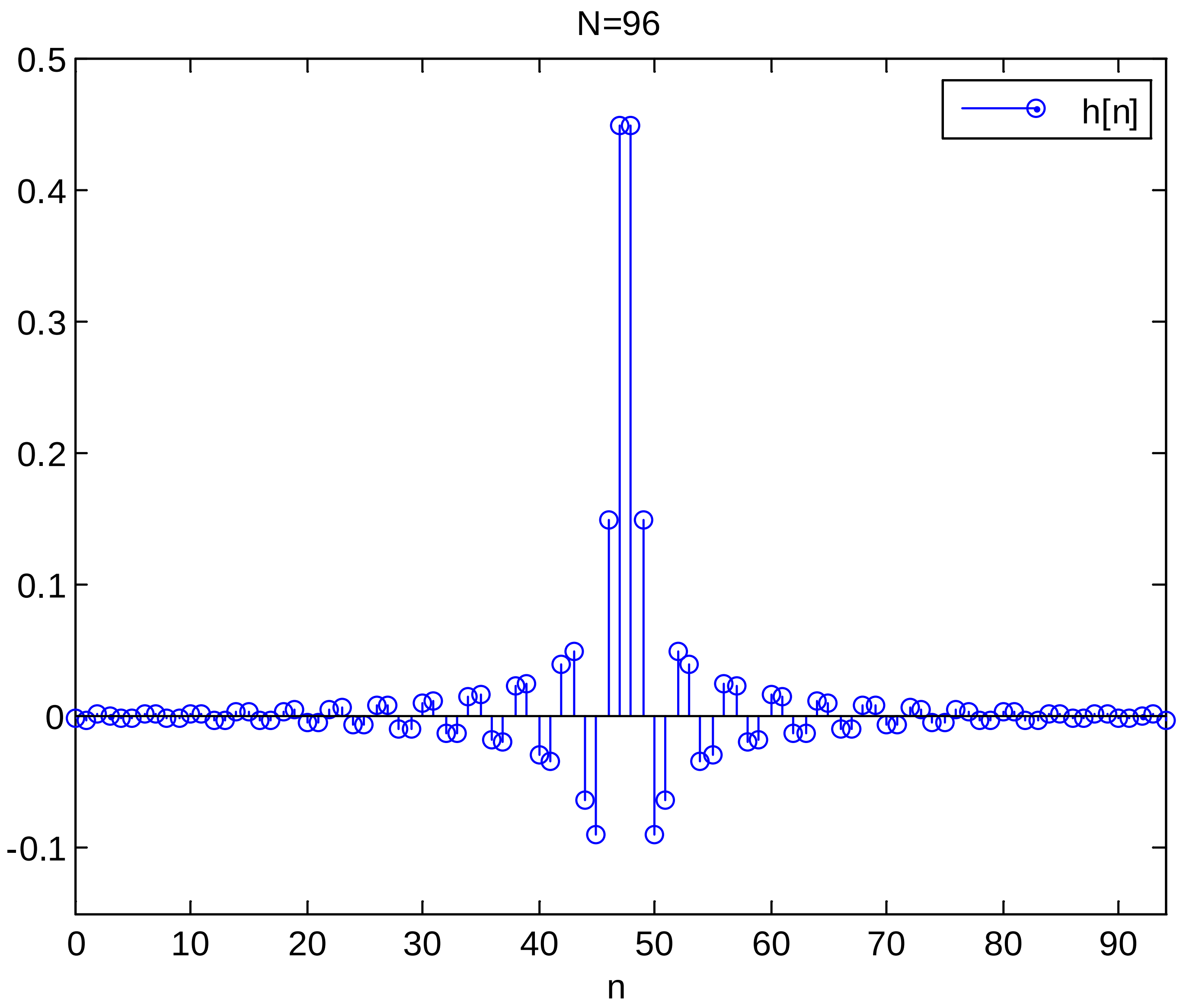


Fig.10.41: Impulse response of optimal equiripple lowpass filter

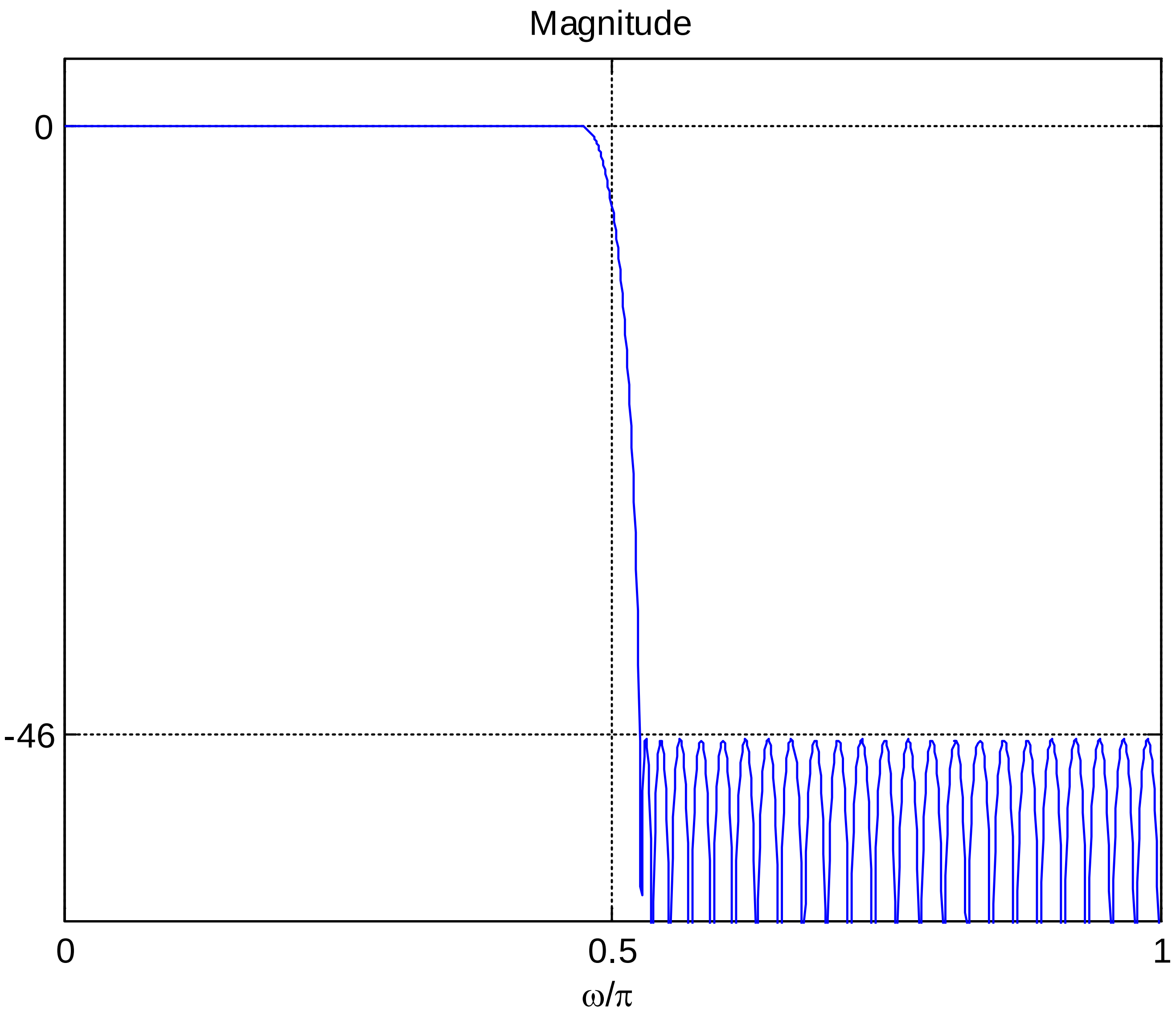


Fig.10.42: Magnitude response of optimal equiripple lowpass filter

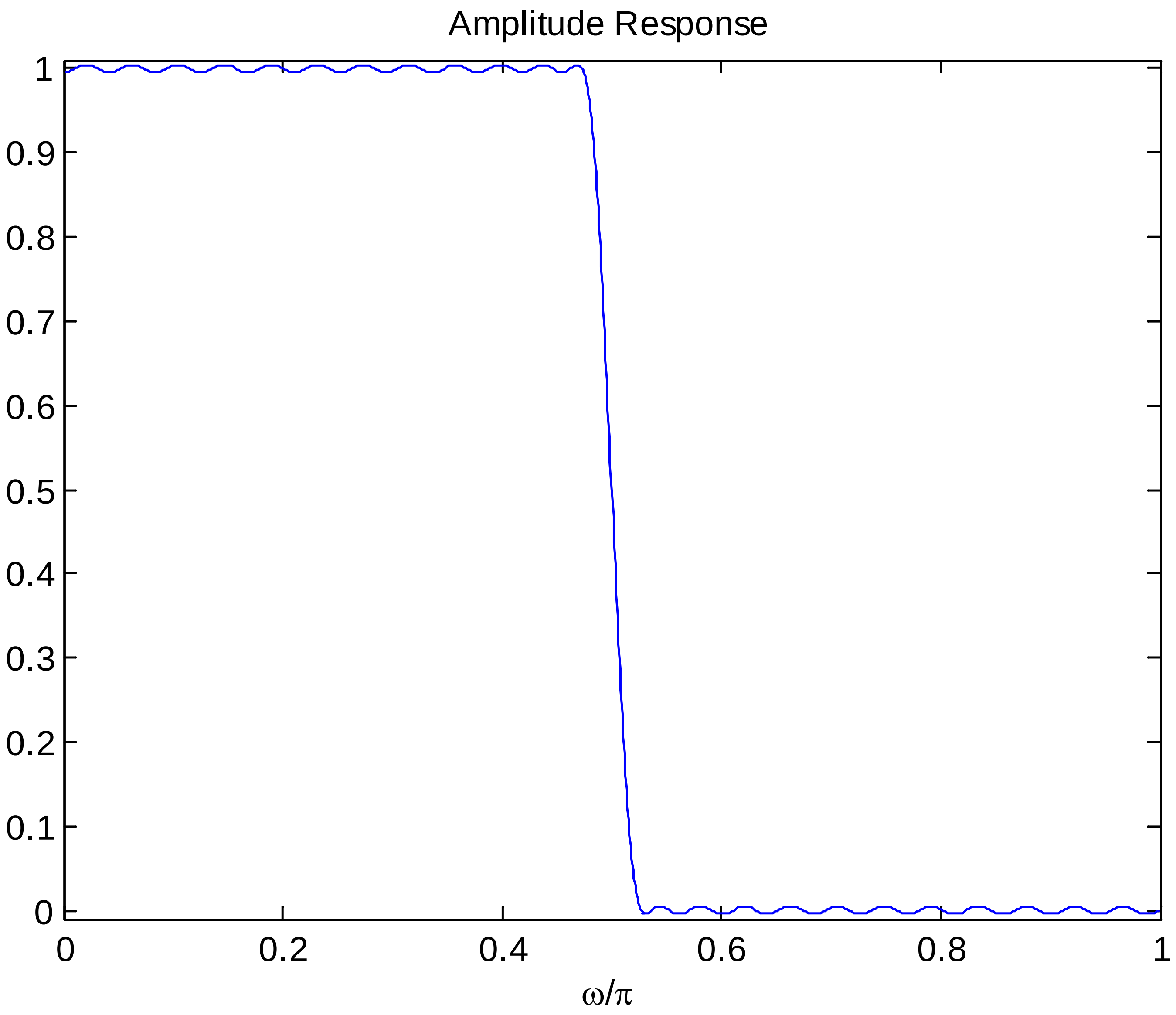


Fig.10.43: Amplitude response of optimal equiripple lowpass filter

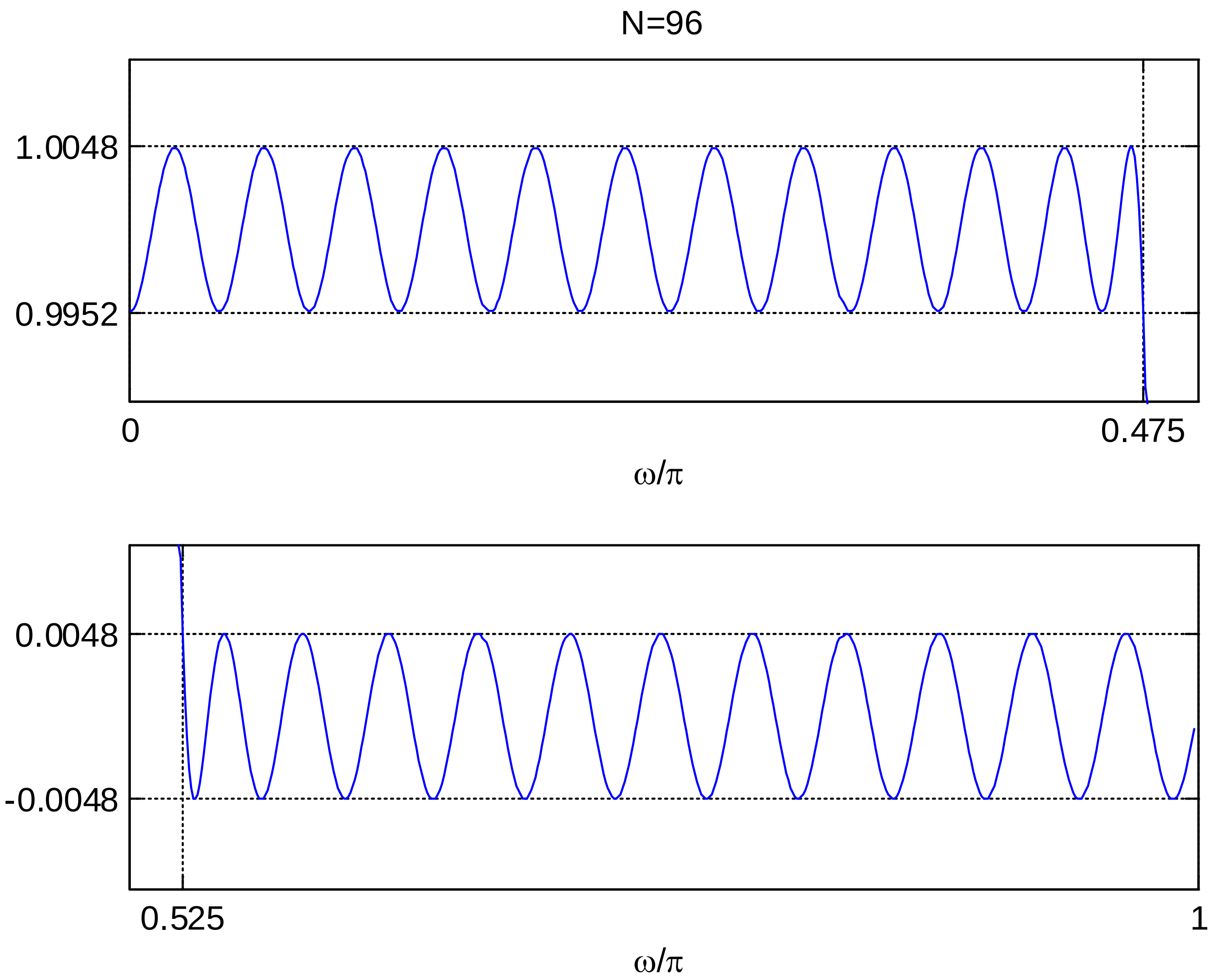


Fig.10.44: Zoomed amplitude responses of optimal equiripple lowpass filter

Chapter 11: IIR Filter Design

Chapter Intended Learning Outcomes:

(i) Ability to design analog Butterworth filters

(ii) Ability to design lowpass IIR filters according to predefined specifications based on analog filter theory and analog-to-digital filter transformation

(iii) Ability to construct frequency-selective IIR filters based on a lowpass IIR filter

Steps in Infinite Impulse Response Filter Design

The system transfer function of an IIR filter is:

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} \tag{11.1}$$

The task in IIR filter design is to find $\{a_k\}$ and $\{b_k\}$ such that $H(z)$ satisfies the given specifications.

Once $H(z)$ is computed, the filter can then be realized in hardware or software according to a direct, canonic, cascade or parallel form

We make use of the analog filter design to produce the required $H(z)$

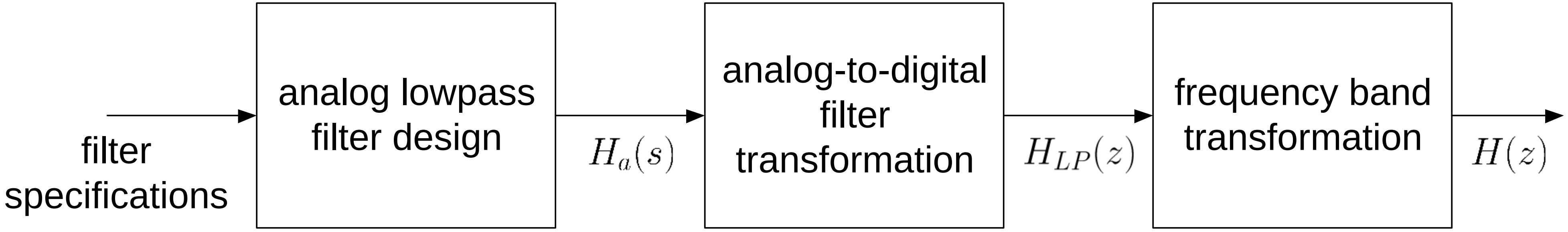

filter specifications
analog lowpass filter design
$H_a(s)$
analog-to-digital filter transformation
$H_{LP}(z)$
frequency band transformation
$H(z)$

Fig.11.1: Steps in determining transfer function of IIR filter

Note that s is the Laplace transform parameter and substituting $s = j\Omega$ in $H_a(s)$ yields the Fourier transform of the filter, that is, $H_a(j\Omega)$

Main drawback is that there is no control over the phase response of $H(e^{j\omega})$, implying that the filter requirements can only be specified in terms of magnitude response

Butterworth Lowpass Filter Design

In analog lowpass filter design, we can only specify the magnitude of $H_a(j\Omega)$. Typically, we employ the magnitude square response, that is, $|H_a(j\Omega)|^2$:

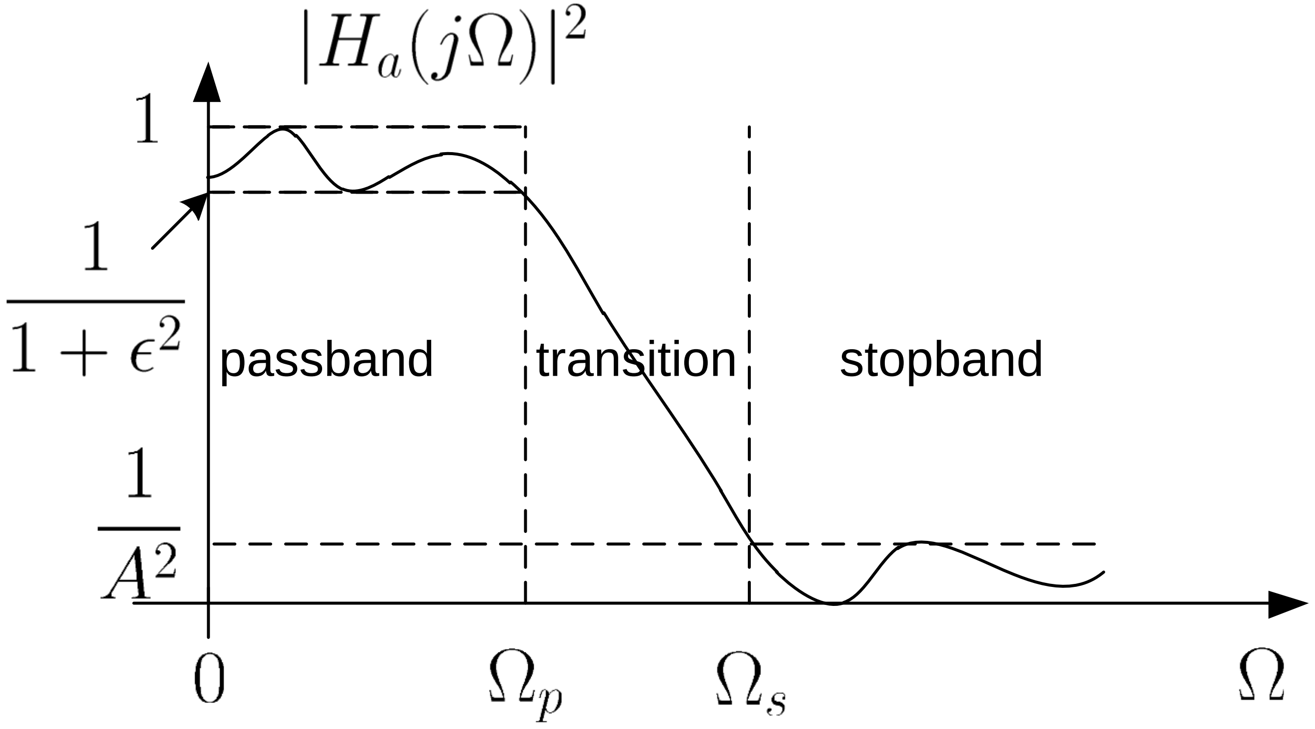


Fig.11.2: Specifications of analog lowpass filter

Passband corresponds to $\Omega \in [0, \Omega_p]$ where Ω_p is the passband frequency and ϵ is called the passband ripple

Stopband corresponds to $\Omega \in [\Omega_s, \infty)$ where Ω_s is the stopband frequency and A is called the stopband attenuation

Transition band corresponds to $\Omega \in [\Omega_p, \Omega_s]$

The specifications are represented as the two inequalities:

$$\frac{1}{1+\epsilon^2} \leq |H_a(j\Omega)|^2 \leq 1, \quad 0 \leq \Omega \leq \Omega_p \tag{11.2}$$

and

$$0 \leq |H_a(j\Omega)|^2 \leq \frac{1}{A^2}, \quad \Omega \geq \Omega_s \tag{11.3}$$

In particular, at $\Omega = \Omega_p$ and $\Omega = \Omega_s$, we have:

$$|H_a(j\Omega_p)|^2 = \frac{1}{1+\epsilon^2} \tag{11.4}$$

and

$$|H_a(j\Omega_s)|^2 = \frac{1}{A^2} \tag{11.5}$$

Apart from ϵ and A, it is also common to use their respective dB versions, denoted by R_p and A_s:

$$R_p = -10\log_{10}\left(\frac{1}{1+\epsilon^2}\right) \Rightarrow \epsilon = \sqrt{10^{R_p/10}-1} \tag{11.6}$$

and

$$A_s = -10\log_{10}\left(\frac{1}{A^2}\right) \Rightarrow A = 10^{A_s/20} \tag{11.7}$$

The magnitude square response of a Nth-order Butterworth lowpass filter is:

$$|H_a(j\Omega)|^2 = \frac{1}{1 + \left(\frac{\Omega}{\Omega_c}\right)^{2N}} \qquad (11.8)$$

The filter is characterized by Ω_c and N, which represent the cutoff frequency and filter order

- $|H_a(j\Omega)|^2 = 1$ at $\Omega = 0$ and $|H_a(j\Omega)|^2 = 0.5$ at $\Omega = \Omega_c$ for all N
- $|H_a(j\Omega)|^2$ is a monotonically decreasing function of frequency which indicates that there is no ripple
- filter shape is closer to the ideal response as N increases, although the filter with order of $N \rightarrow \infty$ is not realizable.

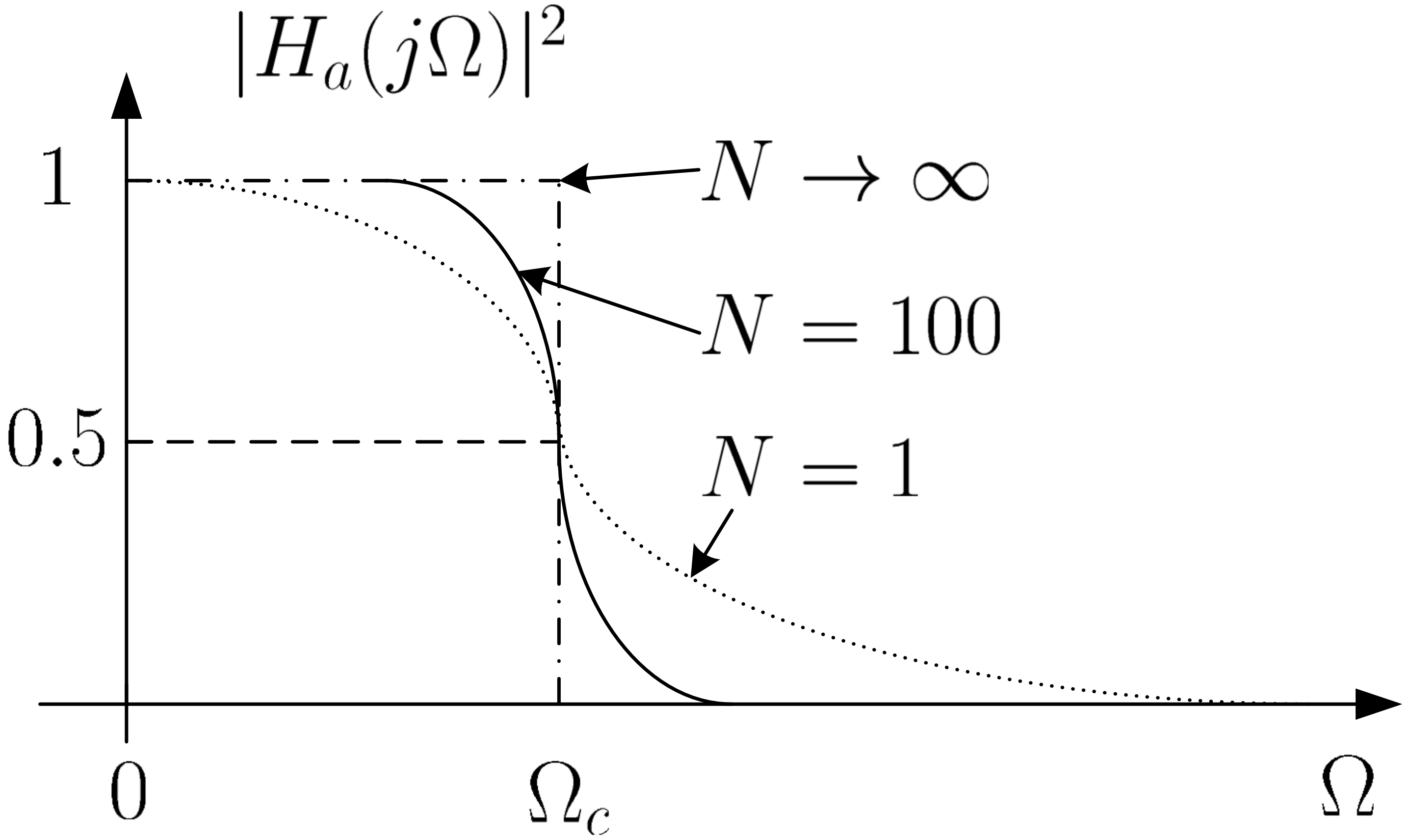


Fig.11.3: Magnitude square responses of Butterworth lowpass filter

To determine $H_a(s)$, we first make use of its relationship with $H_a(j\Omega)$:

$$H_a(s)\big|_{s=j\Omega} = H_a(j\Omega) \qquad (11.9)$$

From (11.8)-(11.9), we obtain:

$$H_a(s)H_a(-s) = \left|H_a(j\Omega)\right|^2\Big|_{\Omega=s/j} = \frac{(j\Omega_c)^{2N}}{s^{2N}+(j\Omega_c)^{2N}} \qquad (11.10)$$

The $2N$ poles of $H_a(s)H_a(-s)$, denoted by c_k, $k=0,1,\cdots,2N-1$, are given as:

$$c_k = \begin{cases} \Omega_c e^{jk\pi/N}, & \text{odd } N \\ \Omega_c e^{jk\pi/N}\cdot e^{j\pi/(2N)}, & \text{even } N \end{cases} \qquad (11.11)$$

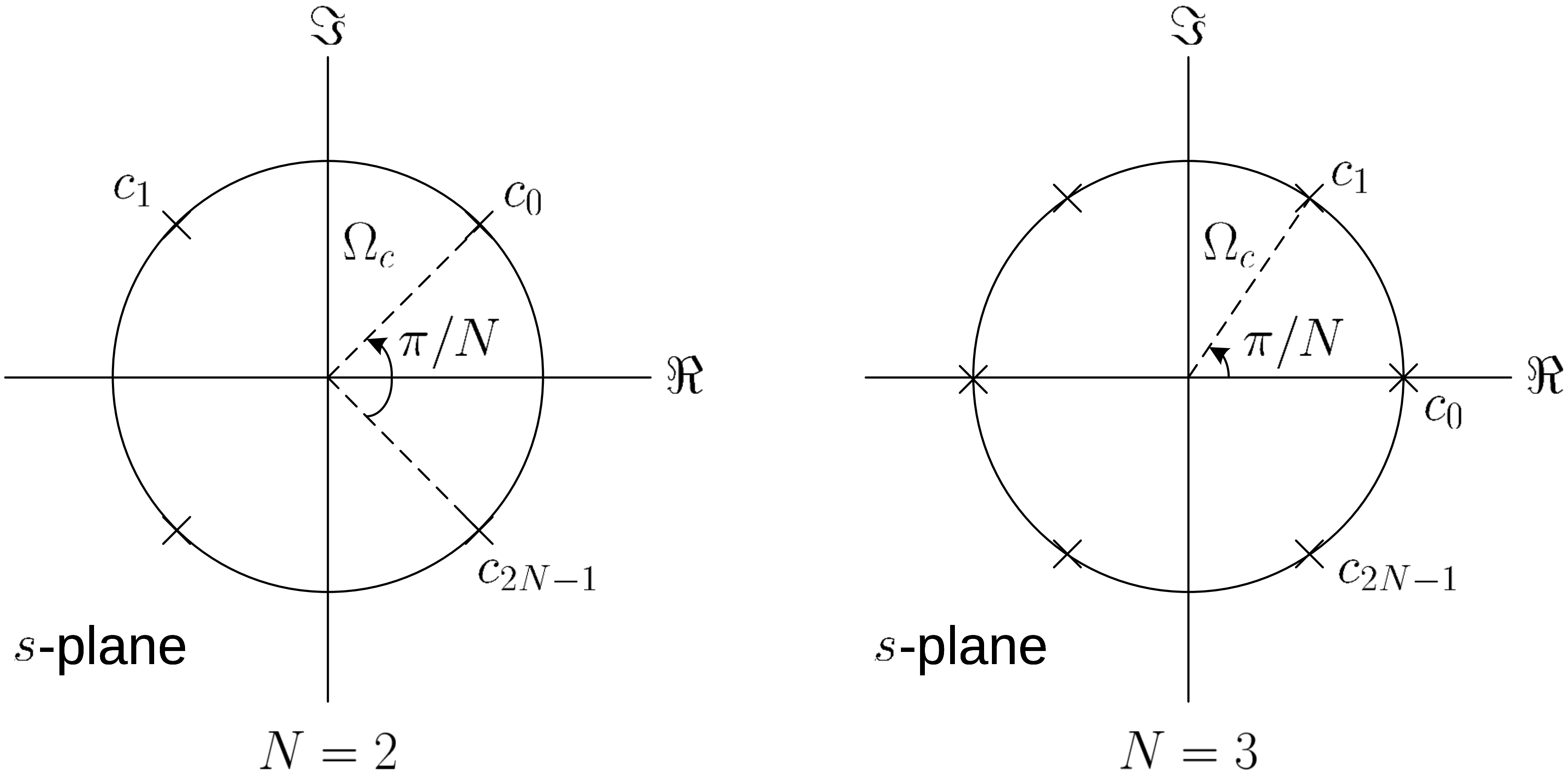


Fig.11.4: Poles of Butterworth lowpass filter

- $\{c_k\}$ are uniformly distributed on a circle of radius Ω_c with angular spacing of π/N in the s-plane

- poles are symmetrically located with respect to the imaginary axis

- there are two real-valued poles when N is odd

To extract $H_a(s)$ from (11.10), we utilize the knowledge that all poles of a stable and causal analog filter should be on the left half of the s-plane. As a result, $H_a(s)$ is:

$$H_a(s) = \frac{\Omega_c^N}{\displaystyle\prod_{\Re\{c_k\}<0} (s - c_k)} \tag{11.12}$$

Example 11.1

The magnitude square response of a Butterworth lowpass filter has the form of:

$$|H_a(j\Omega)|^2 = \frac{1}{1 + 0.000001\Omega^6}$$

Determine the filter transfer function $H_a(s)$.

Expressing $|H_a(j\Omega)|^2$ as:

$$|H_a(j\Omega)|^2 = \frac{1}{1 + \left(\dfrac{\Omega}{10}\right)^{2\cdot 3}}$$

From (11.8), $\Omega_c = 10$ and $N = 3$

From (11.11):

$$c_k = 10e^{jk\pi/3}, \quad k = 2,3,4$$

Finally, we apply (11.12) to obtain:

$$\begin{aligned} H_a(s) &= \frac{\Omega_c^3}{(s-c_2)(s-c_3)(s-c_4)} \\ &= \frac{1000}{(s-10e^{j2\pi/3})(s+10)(s-10e^{j4\pi/3})} \\ &= \frac{1000}{(s+10)(s^2+10s+100)} \end{aligned}$$

To find Ω_c and N given the passband and stopband requirements in terms of Ω_p, Ω_s, R_p and A_s, we exploit (11.4)-(11.5) together with (11.6)-(11.7) to obtain

$$-10\log_{10}\left(\frac{1}{1+\left(\frac{\Omega_p}{\Omega_c}\right)^{2N}}\right) = R_p \qquad (11.13)$$

and

$$-10\log_{10}\left(\frac{1}{1+\left(\frac{\Omega_s}{\Omega_c}\right)^{2N}}\right) = A_s \qquad (11.14)$$

Solving (11.13)-(11.14) and noting that N should be an integer, we get

$$N = \left\lceil \frac{\log_{10}\left[\left(10^{R_p/10}-1\right)/\left(10^{A_s/10}-1\right)\right]}{2\log_{10}(\Omega_p/\Omega_s)} \right\rceil \tag{11.15}$$

where $\lceil u \rceil$ rounds up u to the nearest integer.

The Ω_c is then obtained from (11.13) or (11.14) so that the specification can be exactly met at Ω_p or Ω_s, respectively

From (11.13), Ω_c is computed as:

$$\Omega_c = \frac{\Omega_p}{\left(10^{R_p/10}-1\right)^{1/(2N)}} \tag{11.16}$$

From (11.14), Ω_c is computed as:

$$\Omega_c = \frac{\Omega_s}{\left(10^{A_s/10} - 1\right)^{1/(2N)}} \qquad (11.17)$$

As a result, the admissible range of Ω_c is:

$$\Omega_c \in \left[\frac{\Omega_p}{\left(10^{R_p/10} - 1\right)^{1/(2N)}}, \frac{\Omega_s}{\left(10^{A_s/10} - 1\right)^{1/(2N)}}\right] \qquad (11.18)$$

Example 11.2

Determine the transfer function of a Butterworth lowpass filter whose magnitude requirements are $\Omega_p = 4\pi \quad \mathrm{rads}^{-1}$, $\Omega_s = 6\pi \ \mathrm{rads}^{-1}$, $R_p = 8$ dB and $A_s = 16$ dB.

Employing (11.15) yields:

$$N = \left\lceil \frac{\log_{10}\left[\left(10^{8/10}-1\right)/\left(10^{16/10}-1\right)\right]}{2\log_{10}(4\pi/(6\pi))} \right\rceil = \lceil 2.45 \rceil = 3$$

Putting $N = 3$ in (11.18), the cutoff frequency is:

$$\Omega_c \in \left[\frac{4\pi}{\left(10^{8/10}-1\right)^{1/(2\cdot3)}}, \frac{6\pi}{\left(10^{16/10}-1\right)^{1/(2\cdot3)}} \right] = [9.5141, 10.2441]$$

For simplicity, we select $\Omega_c = 10$. Using Example 11.1, the filter transfer function $H_a(s)$ is:

$$H_a(s) = \frac{1000}{(s+10)(s^2+10s+100)} = \frac{1000}{s^3+20s^2+200s+1000}$$

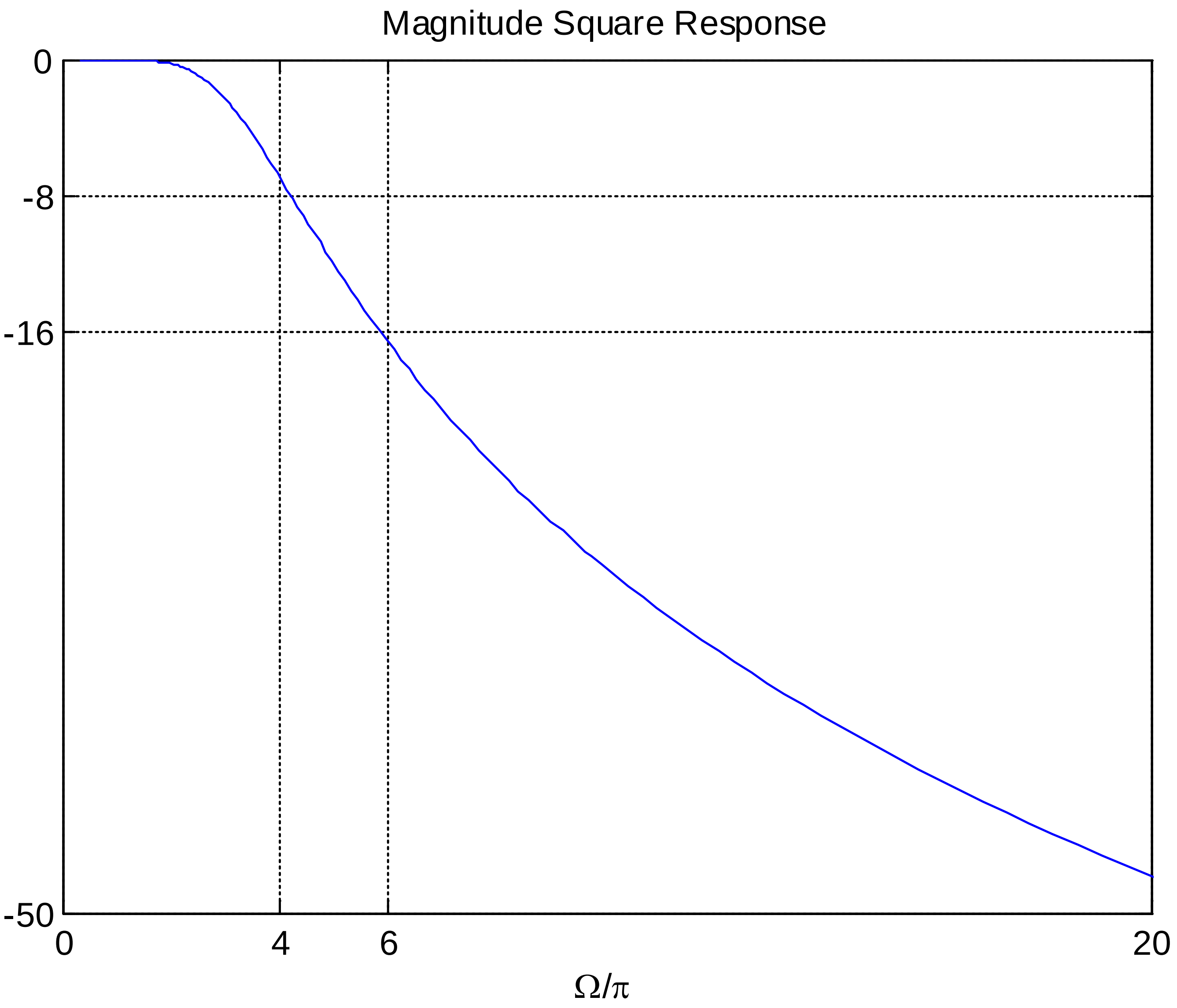


Fig.11.5: Magnitude square response of Butterworth lowpass filter

The MATLAB program is provided as `ex11_2.m` where the command `freqs`, which is analogous to `freqz`, is used to plot $|H_a(j\Omega)|^2$

Analog-to-Digital Filter Transformation

Typical methods include impulse invariance, bilinear transformation, backward difference approximation and matched-z transformation

Their common feature is that a stable analog filter will transform to a stable system with transfer function $H_{LP}(z)$.

Left half of s-plane maps into inside of unit circle in z-plane

Each has its pros and cons and thus optimal transformation does not exist

- Impulse Invariance

The idea is simply to sample impulse response of the analog filter $h_a(t)$ to obtain the digital lowpass filter impulse response $h_{LP}[n]$

The relationship between $h_{LP}[n]$ and $h_a(t)$ is

$$h_{LP}[n] = T \cdot h_a(nT), \quad n = \cdots - 1, 0, 1, 2, \cdots \qquad (11.19)$$

where T is the sampling interval

Why there is a scaling of *T* ?

With the use of (4.5) and (5.3)-(5.4), $H_{LP}(e^{j\omega})$ is:

$$H_{LP}(e^{j\omega}) = \sum_{k=-\infty}^{\infty} H_a\left(j\left(\frac{\omega}{T} - \frac{2\pi k}{T}\right)\right) \qquad (11.20)$$

where the analog and digital frequencies are related as:

$$\omega = \Omega T \qquad (11.21)$$

The impulse response of the resultant IIR filter is similar to that of the analog filter

Aliasing due to the overlapping of $\{H_a\left(j\left(\omega - 2\pi k\right)/T\right)\}$ which are not bandlimited. However, $H_a(j\Omega)$ corresponds to a lowpass filter and thus aliasing effect is negligibly small.

To derive the IIR filter transfer function $H_{LP}(z)$ from $H_a(s)$, we first obtain the partial fraction expansion:

$$H_a(s) = \sum_{k=1}^{N} \frac{A_k}{s - s_k} \qquad (11.22)$$

where $\{s_k\}$ are the poles on the left half of the s-plane

The inverse Laplace transform of (11.22) is given as:

$$h_a(t) = \begin{cases} \displaystyle\sum_{k=1}^{N} A_k e^{s_k t}, & t \geq 0 \\ 0, & t < 0 \end{cases} \qquad (11.23)$$

Substituting (11.23) into (11.19), we have:

$$h_{LP}[n] = \sum_{k=1}^{N} T A_k e^{s_k n T} u[n] \qquad (11.24)$$

The z transform of $h_{LP}[n]$ is:

$$H_{LP}(z) = \sum_{k=1}^{N} \frac{T A_k}{1 - e^{s_k T} z^{-1}} \qquad (11.25)$$

Comparing (11.22) and (11.25), it is seen that a pole of $s = s_k$ in the s-plane transforms to a pole at $z = e^{s_k T}$ in the z-plane:

$$z = e^{sT} \qquad (11.26)$$

Expressing $s = \sigma + j\Omega$:

$$z = e^{\sigma T} \cdot e^{j\Omega T} = e^{\sigma T} \cdot e^{j(\Omega + 2\pi k/T)T} \qquad (11.27)$$

where k is any integer, indicating a many-to-one mapping

Each infinite horizontal strip of width $2\pi/T$ maps into the entire z-plane

$\sigma = 0$ maps to $|z| = 1$, that is, $j\Omega$ axis in the s -plane transforms to the unit circle in the z-plane

$\sigma < 0$ maps to $|z| < 1$, stable $H_a(s)$ produces stable $H_{LP}(z)$

$\sigma > 0$ maps to $|z| > 1$, right half of the s-plane maps into the outside of the unit circle in the z-plane

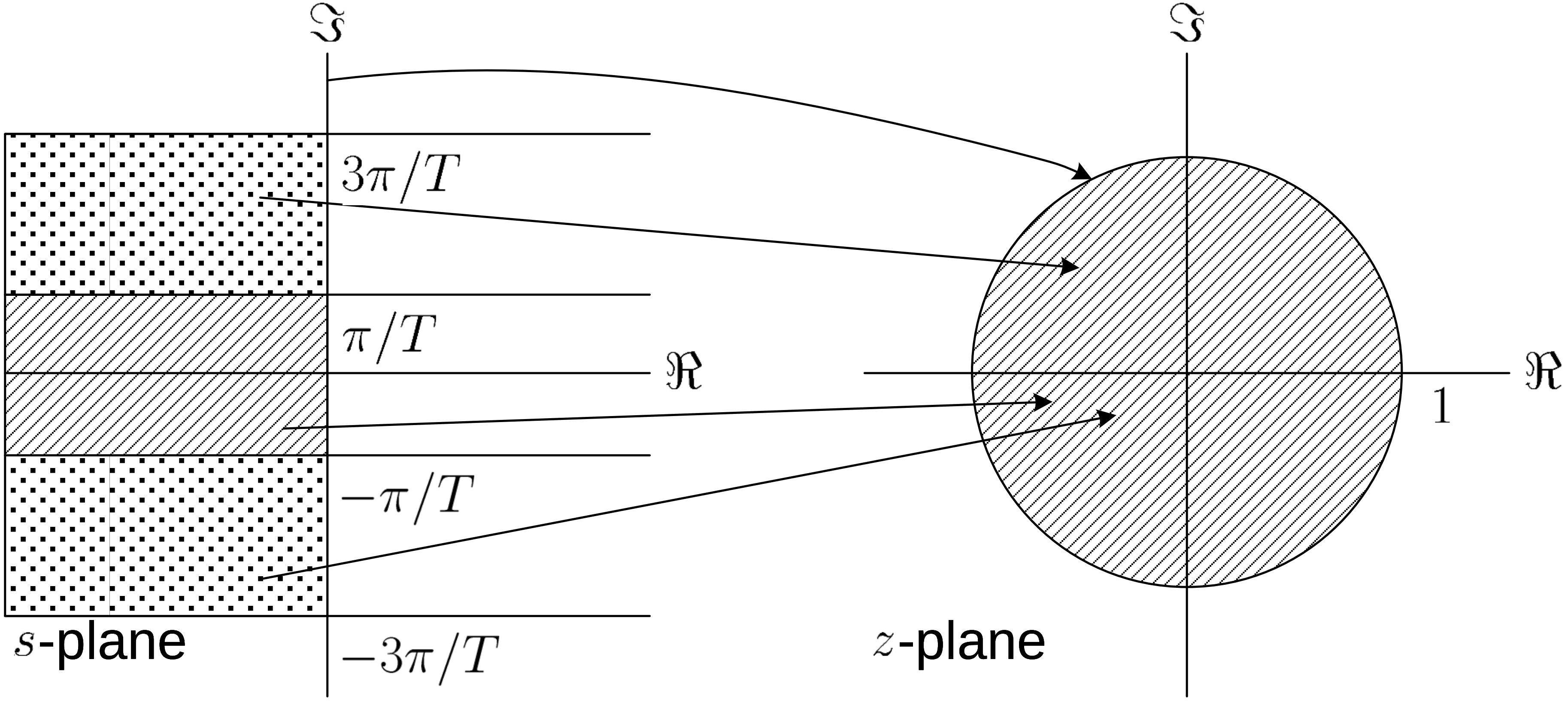


Fig.11.6: Mapping between s and z in impulse invariance method

Given the magnitude square response specifications of $H_{LP}(e^{j\omega})$ in terms of ω_p, ω_s, R_p and A_s, the design procedure for $H_{LP}(z)$ based on the impulse invariance method is summarized as the following steps:

(i) Select a value for the sampling interval T and then compute the passband and stopband frequencies for the analog lowpass filter according to $\Omega_p = \omega_p/T$ and $\Omega_s = \omega_s/T$

(ii) Design the analog Butterworth filter with transfer function $H_a(s)$ according to Ω_p, Ω_s, R_p and A_s

(iii) Perform partial fraction expansion on $H_a(s)$ as in (11.22)

(iv) Obtain $H_{LP}(z)$ using (11.25)

Example 11.3

The transfer function of an analog filter has the form of

$$H_a(s) = \frac{2s}{s^2 + 6s + 8}$$

Use impulse invariance method with sampling interval $T = 1$ to transform $H_a(s)$ to a digital filter transfer function $H(z)$.

Performing partial fraction expansion on $H_a(s)$:

$$H_a(s) = \frac{-2}{s+2} + \frac{4}{s+4}$$

Applying (11.25) with $T = 1$ yields

$$H(z) = \frac{-2}{1 - e^{-2}z^{-1}} + \frac{4}{1 - e^{-4}z^{-1}} = \frac{2 - 0.5047z^{-1}}{1 - 0.1537z^{-1} + 0.0025z^{-2}}$$

Example 11.4

Determine the transfer function $H_{LP}(z)$ of a digital lowpass filter whose magnitude requirements are $\omega_p = 0.4\pi$, $\omega_s = 0.6\pi$, $R_p = 8$ dB and $A_s = 16$ dB. Use the Butterworth lowpass filter and impulse invariance method in the design.

Selecting the sampling interval as $T = 0.1$, the analog frequency parameters are computed as:

$$\Omega_p = \frac{\omega_p}{T} = 4\pi$$

and

$$\Omega_s = \frac{\omega_s}{T} = 6\pi$$

Using Example 11.2, a Butterworth filter which meets the magnitude requirements are:

$$H_a(s) = \frac{1000}{(s+10)(s^2+10s+100)} = \frac{1000}{s^3+20s^2+200s+1000}$$

Performing partial fraction expansion on $H_a(s)$ with the use of the MATLAB command `residue`, we get

$$H_a(s) = \frac{10}{s+10} + \frac{-5-j2.8868}{s+5-j8.6603} + \frac{-5+j2.8868}{s+5+j8.6603}$$

Applying (11.25) with $T=0.1$ yields

$$
\begin{aligned}
H_{LP}(z) &= \frac{0.1 \cdot 10}{1 - e^{-10 \cdot 0.1} z^{-1}} + \frac{0.1 \cdot (-5 - j2.8868)}{1 - e^{(-5+j8.6603) \cdot 0.1} z^{-1}} + \frac{0.1 \cdot (-5 + j2.8868)}{1 - e^{(-5-j8.6603) \cdot 0.1} z^{-1}} \\
&= \frac{1}{1 - 0.3679 z^{-1}} + \frac{-0.5 - j0.2887}{1 - (0.3929 + j0.4620) z^{-1}} \\
&\quad + \frac{-0.5 + j0.2887}{1 - (0.3929 - j0.4620) z^{-1}} \\
&= \frac{1}{1 - 0.3679 z^{-1}} + \frac{-1 + 0.6597 z^{-1}}{1 - 0.7859 z^{-1} + 0.3679 z^{-2}} \\
&= \frac{0.2417 z^{-1} + 0.1262 z^{-2}}{1 - 1.1538 z^{-1} + 0.6570 z^{-2} - 0.1354 z^{-3}}
\end{aligned}
$$

The MATLAB program is provided as `ex11_4.m`.

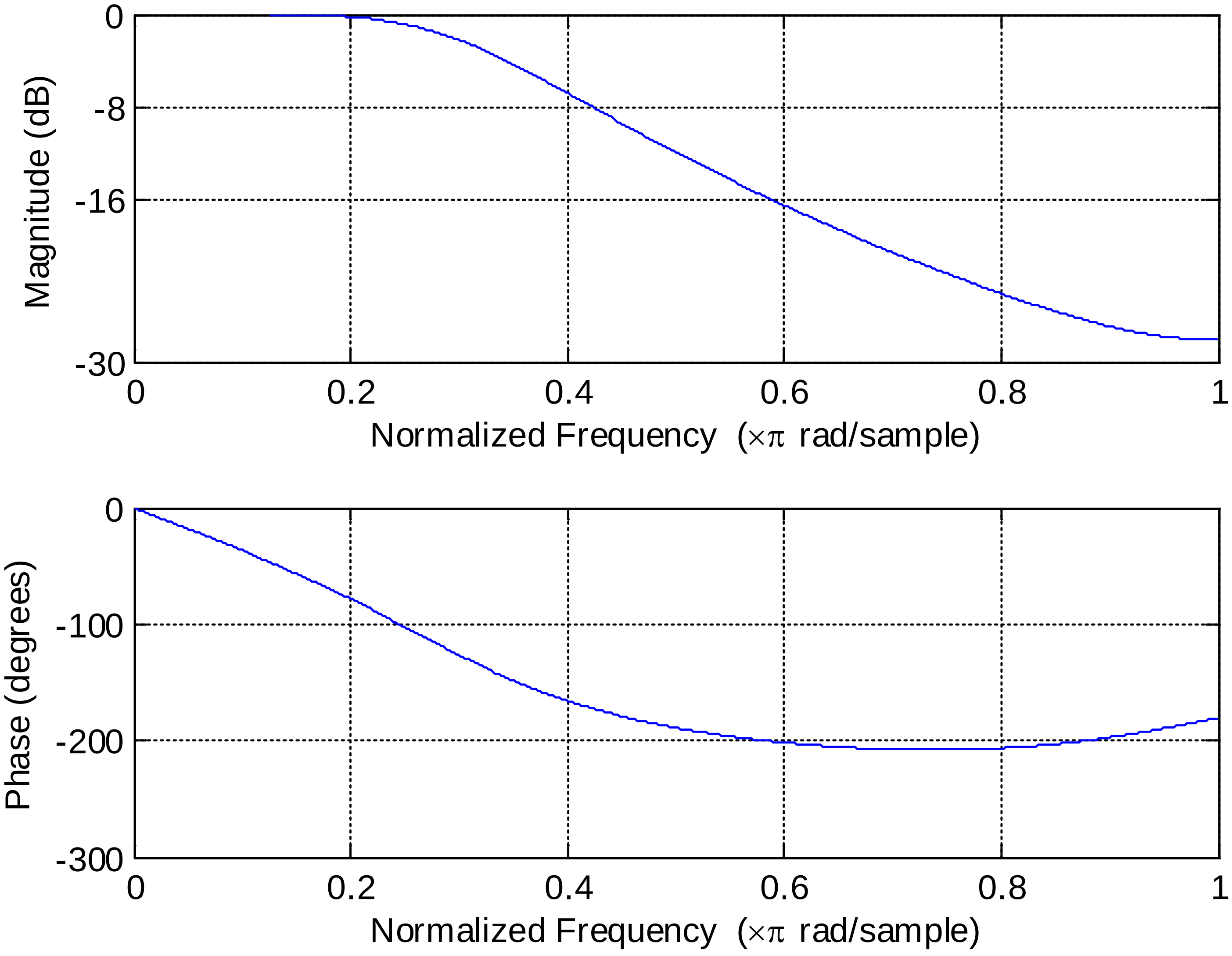


Fig.11.7: Magnitude and phase responses based on impulse invariance

- Bilinear Transformation

It is a conformal mapping that maps the $j\Omega$ axis of the s-plane into the unit circle of the z-plane only once, implying there is no aliasing problem as in the impulse invariance method

It is a one-to-one mapping

The relationship between s and z is:

$$s = \frac{2}{T} \cdot \frac{1 - z^{-1}}{1 + z^{-1}} \Leftrightarrow z = \frac{1 + sT/2}{1 - sT/2} \qquad (11.28)$$

Employing $s = \sigma + j\Omega$, z can be expressed as:

$$z = \frac{(1 + \sigma T/2) + j\Omega T/2}{(1 - \sigma T/2) - j\Omega T/2} \qquad (11.29)$$

$\sigma = 0$ maps to $|z| = 1$, that is, $j\Omega$ axis in the s-plane transforms to the unit circle in the z-plane

$\sigma < 0$ maps to $|z| < 1$, stable $H_a(s)$ produces a stable $H_{LP}(z)$

$\sigma > 0$ maps to $|z| > 1$, right half of the s-plane maps into the outside of the unit circle in the z-plane

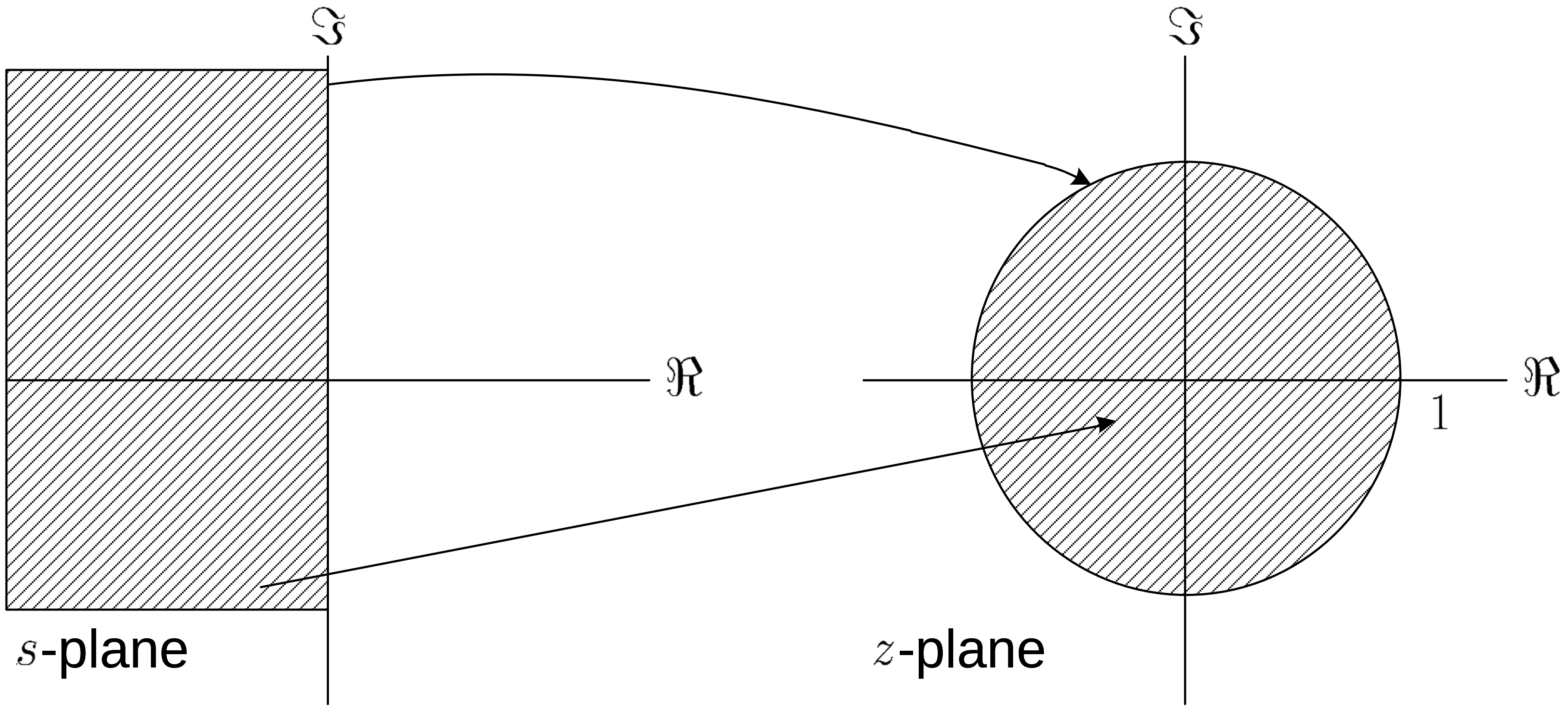


Fig.11.8: Mapping between s and z in bilinear transformation

Although aliasing is avoided, the drawback of the bilinear transformation is that there is no linear relationship between ω and Ω

Putting $z = e^{j\omega}$ and $s = j\Omega$ in (11.28), ω and Ω are related as:

$$j\Omega = \frac{2}{T} \cdot \frac{1 - e^{-j\omega}}{1 + e^{-j\omega}} = \frac{2}{T} \cdot \frac{e^{j\omega/2} - e^{-j\omega/2}}{e^{j\omega/2} + e^{-j\omega/2}} = j\frac{2}{T}\tan\left(\frac{\omega}{2}\right)$$

$$\Rightarrow \Omega = \frac{2}{T}\tan\left(\frac{\omega}{2}\right) \Leftrightarrow \omega = 2\tan^{-1}\left(\frac{\Omega T}{2}\right) \qquad (11.30)$$

Given the magnitude square response specifications of $H_{LP}(e^{j\omega})$ in terms of ω_p, ω_s, R_p and A_s, the design procedure for $H_{LP}(z)$ based on the bilinear transformation is summarized as the following steps:

(i) Select a value for T and then compute the passband and stopband frequencies for the analog lowpass filter according $\Omega_p = (2/T)\tan(\omega_p/2)$ and $\Omega_s = (2/T)\tan(\omega_s/2)$

(ii) Design the analog Butterworth filter with transfer function $H_a(s)$ according to Ω_p, Ω_s, R_p and A_s.

(iii) Obtain $H_{LP}(z)$ from $H_a(s)$ using the substitution of (11.28).

Example 11.5

The transfer function of an analog filter has the form of

$$H_a(s) = \frac{2s}{s^2 + 6s + 8}$$

Use the bilinear transformation with $T = 1$ to transform $H_a(s)$ to a digital filter with transfer function $H(z)$.

Applying (11.28) with $T=1$ yields

$$H(z)=\frac{2\cdot 2\dfrac{1-z^{-1}}{1+z^{-1}}}{\left(2\dfrac{1-z^{-1}}{1+z^{-1}}\right)^2+6\cdot 2\dfrac{1-z^{-1}}{1+z^{-1}}+8}=\frac{4-4z^{-2}}{15+14z^{-1}+9z^{-2}}$$

Example 11.6

Determine the transfer function $H_{LP}(z)$ of a digital lowpass filter whose magnitude requirements are $\omega_p=0.4\pi$, $\omega_s=0.6\pi$, $R_p=8$ dB and $A_s=16$ dB. Use the Butterworth lowpass filter and bilinear transformation in the design.

Selecting $T=0.1$, the analog frequency parameters are computed according to (11.30) as:

$$\Omega_p = \frac{2}{T}\tan\left(\frac{\omega_p}{2}\right) = 14.5309$$

and

$$\Omega_s = \frac{2}{T}\tan\left(\frac{\omega_s}{2}\right) = 27.5276$$

Employing (11.15) yields:

$$N = \left\lceil \frac{\log_{10}\left[\left(10^{8/10}-1\right)/\left(10^{16/10}-1\right)\right]}{2\log_{10}(14.5309/27.5276)} \right\rceil = \lceil 1.56 \rceil = 2$$

Putting $N=2$ in (11.18), the cutoff frequency is:

$$\Omega_c \in \left[\frac{14.5309}{\left(10^{8/10}-1\right)^{1/(2\cdot 2)}}, \frac{27.5276}{\left(10^{16/10}-1\right)^{1/(2\cdot 2)}}\right] = [9.5725, 11.0289]$$

For simplicity, $\Omega_c = 10$ is employed.

Following (11.11)-(11.12):

$$H_a(s) = \frac{100}{s^2 + 14.1421s + 100}$$

Finally, we use (11.28) with $T = 0.1$ to yield

$$\begin{aligned} H_{LP}(z) &= \frac{100}{\left(\frac{2}{0.1} \cdot \frac{1-z^{-1}}{1+z^{-1}}\right)^2 + 14.1421 \cdot \frac{2}{0.1} \cdot \frac{1-z^{-1}}{1+z^{-1}} + 100} \\ &= \frac{1 + 2z^{-1} + z^{-2}}{7.8284 - 6z^{-1} + 2.1716z^{-2}} \end{aligned}$$

The MATLAB program is provided as `ex11_6.m`.

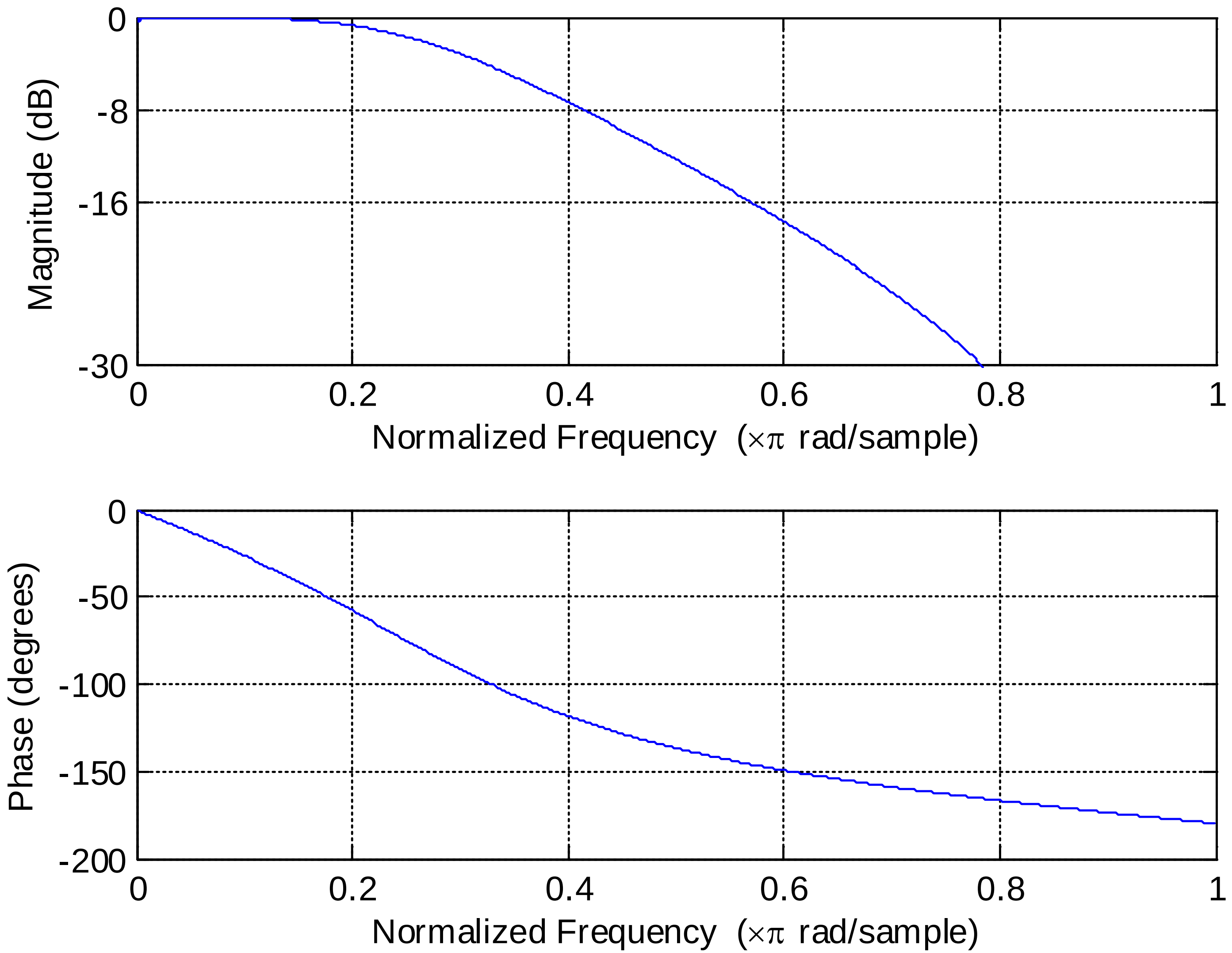


Fig.11.9: Magnitude and phase responses based on bilinear transformation

Frequency Band Transformation

The operations are similar to that of the bilinear transformation but now the mapping is performed only in the z-plane:

$$z_o^{-1} = T(z^{-1}) \tag{11.31}$$

where z_o and z correspond to the lowpass and resultant filters, respectively, and T denotes the transformation operator.

To ensure the transformed filter to be stable and causal, the unit circle and inside of the z_o-plane should map into those of the z-plane, respectively.

Filter Type	Transformation Operator	Design Parameter
Lowpass	$z_o^{-1} = \dfrac{z^{-1} - \alpha}{1 - \alpha z^{-1}}$	$\alpha = \dfrac{\sin\left(\dfrac{\omega_{c_o} - \omega_c}{2}\right)}{\sin\left(\dfrac{\omega_{c_o} + \omega_c}{2}\right)}$
Highpass	$z_o^{-1} = -\dfrac{z^{-1} + \alpha}{1 + \alpha z^{-1}}$	$\alpha = -\dfrac{\cos\left(\dfrac{\omega_{c_o} + \omega_c}{2}\right)}{\cos\left(\dfrac{\omega_{c_o} - \omega_c}{2}\right)}$
Bandpass	$z_o^{-1} = \dfrac{z^{-2} - \dfrac{2\alpha\beta}{\beta+1} z^{-1} + \dfrac{\beta-1}{\beta+1}}{\dfrac{\beta-1}{\beta+1} z^{-2} - \dfrac{2\alpha\beta}{\beta+1} z^{-1} + 1}$	$\alpha = \dfrac{\cos\left(\dfrac{\omega_{c_2} + \omega_{c_1}}{2}\right)}{\cos\left(\dfrac{\omega_{c_2} - \omega_{c_1}}{2}\right)}$ $\beta = \cot\left(\dfrac{\omega_{c_2} - \omega_{c_1}}{2}\right) \tan\left(\dfrac{\omega_{c_o}}{2}\right)$

Bandstop	$z_o^{-1}=\dfrac{z^{-2}-\dfrac{2\alpha}{1+\beta}z^{-1}+\dfrac{1-\beta}{1+\beta}}{\dfrac{1-\beta}{1+\beta}z^{-2}-\dfrac{2\alpha}{1+\beta}z^{-1}+1}$	$\alpha=\dfrac{\cos\left(\dfrac{\omega_{c_2}+\omega_{c_1}}{2}\right)}{\cos\left(\dfrac{\omega_{c_2}-\omega_{c_1}}{2}\right)}$ $\beta=\cot\left(\dfrac{\omega_{c_2}-\omega_{c_1}}{2}\right)\tan\left(\dfrac{\omega_{c_o}}{2}\right)$

Table 11.1: Frequency band transformation operators

Example 11.7

Determine the transfer function $H(z)$ of a digital highpass filter whose magnitude requirements are $\omega_p = 0.6\pi$, $\omega_s = 0.4\pi$, $R_p = 8$ dB and $A_s = 16$ dB. Use the Butterworth lowpass filter and bilinear transformation in the design.

Using Example 11.6, the corresponding lowpass filter transfer function $H_{LP}(z_o)$ is:

$$H_{LP}(z_o) = \frac{1 + 2z_o^{-1} + z_o^{-2}}{7.8284 - 6z_o^{-1} + 2.1716z_o^{-2}}$$

Assigning the cutoff frequencies as the midpoints between the passband and stopband frequencies, we have

$$\omega_{c_o} = \omega_c = \frac{0.4\pi + 0.6\pi}{2} = 0.5\pi$$

With the use of Table 11.1, the corresponding value of α is:

$$\alpha = -\frac{\cos\left(\dfrac{\omega_{c_o} + \omega_c}{2}\right)}{\cos\left(\dfrac{\omega_{c_o} - \omega_c}{2}\right)} = -\frac{\cos(0.5\pi)}{\cos(0)} = 0$$

which gives the transformation operator:

$$z_o^{-1} = -\frac{z^{-1}+0}{1+0\cdot z^{-1}} = -z^{-1}$$

As a result, the digital highpass filter transfer function is:

$$H(z) = H_{LP}(z_o)\big|_{z_o^{-1}=-z^{-1}} = \frac{1-2z^{-1}+z^{-2}}{7.8284+6z^{-1}+2.1716z^{-2}}$$

The MATLAB program is provided as `ex11_7.m`.

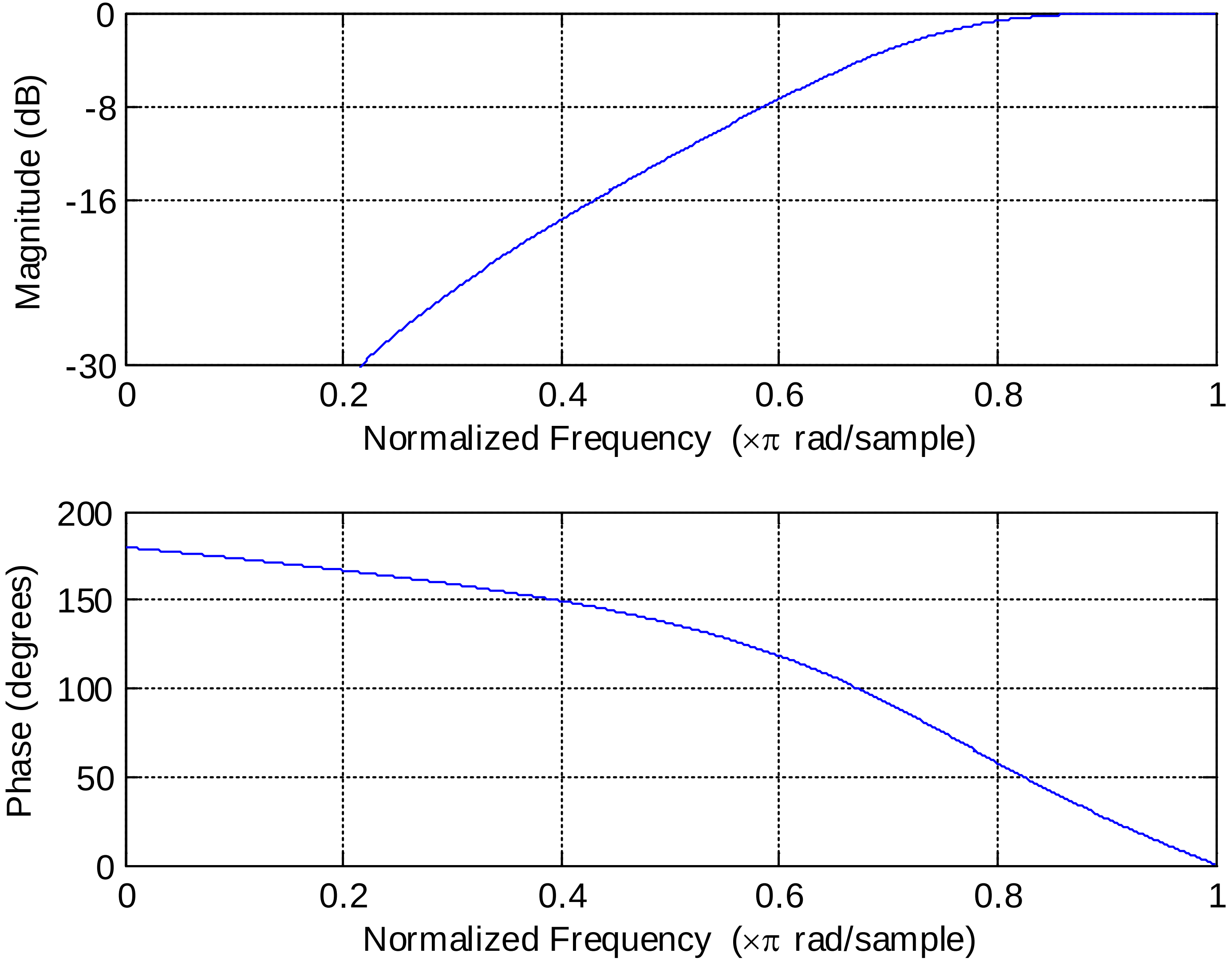


Fig.11.10: Magnitude and phase responses based on frequency band transformation

Chapter 12: Tools and Applications

Chapter Intended Learning Outcomes:

(i) Understanding the basic concepts of signal modeling, correlation, maximum likelihood estimation, least squares and iterative numerical methods

(ii) Ability to solve simple signal processing problems with the use of investigated tools

Tools

- Signal Modeling

1. White Noise

A random sequence $q[n]$ is white if

$$E\{q[n]\} = 0 \tag{12.1}$$

and

$$E\{q[n]q[m]\} = \begin{cases} \sigma_q^2, & m = n \\ 0, & \text{otherwise} \end{cases} \tag{12.2}$$

where E denotes the expectation operator and σ_q^2 is the power or variance of $q[n]$

The mean value of $q[n]$ is zero for all n

There is no correlation between $q[m]$ and $q[n]$ if $m \neq n$

The probability density function (PDF) is not specified, which means that $q[n]$ can be a uniform or Gaussian sequence

The MATLAB command to generate a white uniform number with $\sigma_q^2 = 1/12$ is `rand-0.5` and that of a white Gaussian random variable with $\sigma_q^2 = 1$ is `randn`

$q[n]$ is a wideband signal and has a flat spectrum in the mean sense

2. Sinusoidal Model

Many real-world signals in the areas such as radar, sonar and communications, can be well described as:

$$x[n] = \sum_{k=1}^{L} \alpha_k \cos(\omega_k n + \phi_k) + q[n] \tag{12.3}$$

where α_k , $\omega_k \in (0, \pi)$ and $\phi_k \in [0, 2\pi)$ denote the amplitude, frequency and phase of the k-th sinusoid, respectively

3. Autoregressive Model

Autoregressive (AR) process has been used to represent many real-world signals such as speech and electroencephalography (EEG)

A M-th order AR process is:

$$x[n] = \sum_{k=1}^{M} a_k x[n-k] + q[n] \qquad (12.4)$$

where $\{a_k\}$ are the AR parameters and $q[n]$ is an additive white noise

Taking the z transform on (12.4), we see that the transfer function $H(z)$ of the system with input $q[n]$ and output $x[n]$ is:

$$H(z) = \frac{X(z)}{Q(z)} = \frac{1}{1 - \sum_{k=1}^{M} a_k z^{-k}} \qquad (12.5)$$

4. Moving Average Model

A moving average (MA) process of order N is:

$$x[n] = \sum_{k=0}^{N} b_k q[n-k] \qquad (12.6)$$

where $q[n]$ is an additive white noise and $\{b_k\}$ are called the MA parameters

Taking the z transform on (12.6), we see that the transfer function $H(z)$ of the system with input $q[n]$ and output $x[n]$ is:

$$H(z) = \frac{X(z)}{Q(z)} = \sum_{k=0}^{N} b_k z^{-k} \qquad (12.7)$$

and its finite-duration impulse response $h[n]$ is

$$h[n] = \begin{cases} b_n, & 0 \leq n \leq N \\ 0, & \text{otherwise} \end{cases} \qquad (12.8)$$

5. Autoregressive Moving Average Model

The autoregressive moving average (ARMA) process generalizes the AR and MA models:

$$x[n] = \sum_{k=1}^{M} a_k x[n-k] + \sum_{k=0}^{N} b_k q[n-k] \qquad (12.9)$$

where $\{a_k\}$ and $\{b_k\}$ are the ARMA parameters and $q[n]$ is an additive white noise

The system transfer function is:

$$H(z) = \frac{\displaystyle\sum_{k=0}^{N} b_k z^{-k}}{1 - \displaystyle\sum_{k=1}^{M} a_k z^{-k}} \tag{12.10}$$

- Correlation

It provides a simple and useful measure for determining the similarity between two sequences

Autocorrelation measures the similarity of the same signal $x[n]$ at different time indices and it is defined as:

$$r_{xx}[l] = E\{x[n]x[n-l]\} \tag{12.11}$$

where l is called the lag

If $|r_{xx}[l]|$ is large, $x[n]$ is similar to $x[n-l]$ while they are not similar for a small $|r_{xx}[l]|$

Two important properties for $r_{xx}[l]$ are:

$$r_{xx}[0] \geq |r_{xx}[l]|, \quad l \neq 0 \tag{12.12}$$

and

$$r_{xx}[l] = r_{xx}[-l] \tag{12.13}$$

When ergodicity holds, that is, the mean value is equal to the time average, $r_{xx}[l]$ is also expressed as:

$$r_{xx}[l] = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x[n]x[n-l] \tag{12.14}$$

An estimate of $r_{xx}[l]$ using a finite-length $x[n]$ for $0 \leq n \leq N-1$, denoted by $\hat{r}_{xx}[l]$, is:

$$\hat{r}_{xx}[l] = \hat{r}_{xx}[-l] = \frac{1}{N-l} \sum_{n=l}^{N-1} x[n]x[n-l], \quad 0 \leq l \leq N-1 \quad (12.15)$$

Cross-correlation is used for two different sequences $x[n]$ and $y[n]$:

$$r_{xy}[l] = E\{x[n]y[n-l]\} = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x[n]y[n-l] \quad (12.16)$$

and

$$\hat{r}_{xy}[l] = \frac{1}{N-l} \sum_{n=l}^{N-1} x[n]y[n-l], \quad 0 \leq l \leq N-1 \quad (12.17)$$

with $\hat{r}_{xy}[-l] = \hat{r}_{yx}[l]$, $l = 1, 2, \cdots, N-1$

Example 12.1

Use the MATLAB command `randn` to generate a zero-mean Gaussian sequence with length 10000 and power $\sigma_q^2 = 1$. Verify its whiteness.

We first use `q=randn(1,10000)` to generate the sequence and then use the MATLAB command `xcorr(q)` to compute $\hat{r}_{xx}[l]$

The MATLAB program is provided as `ex12_1.m`.

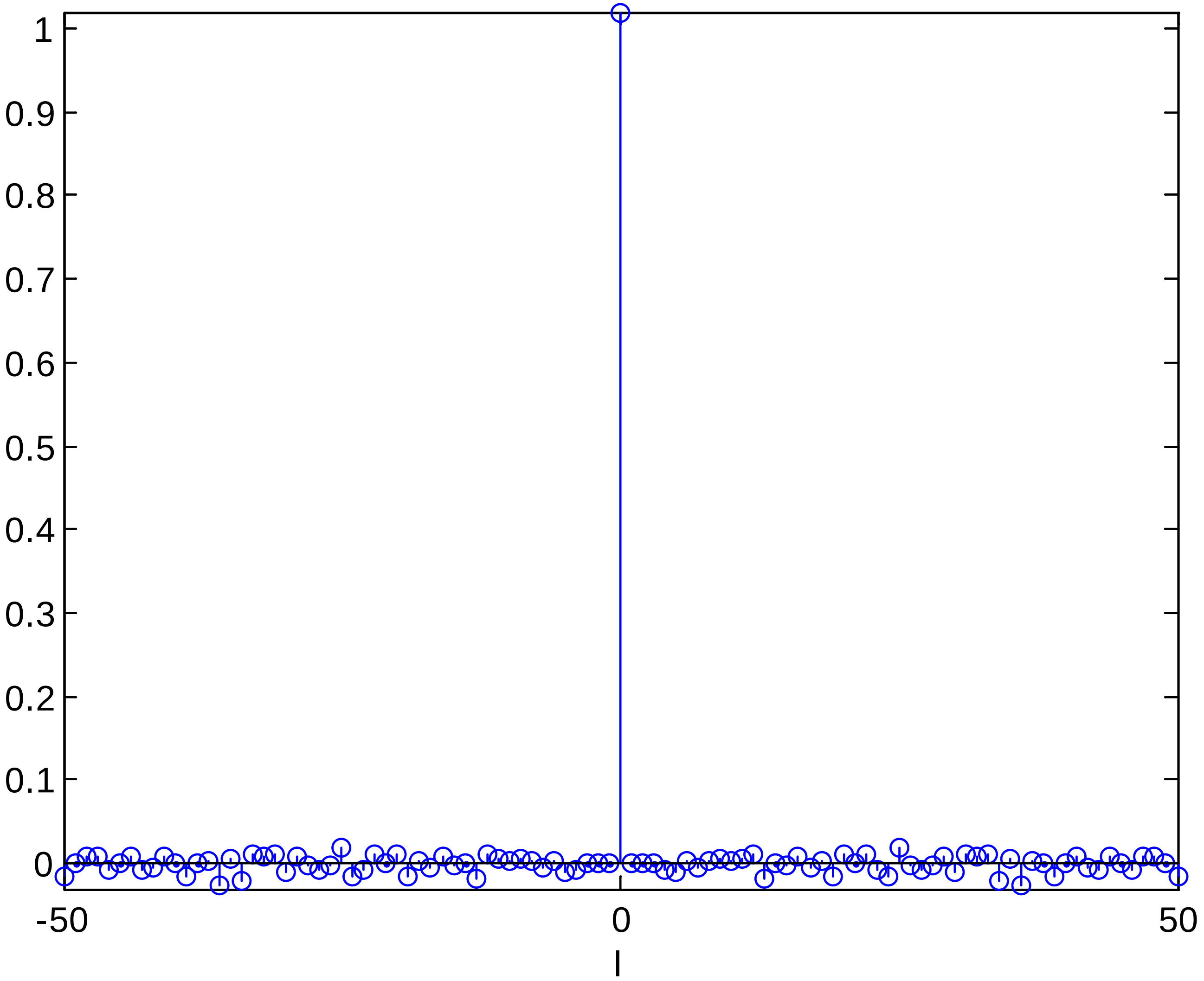


Fig. 12.1: Autocorrelation of white Gaussian sequence

- Maximum Likelihood Estimation

Suppose there is an observed sequence $x[n]$ which contains a deterministic signal in the presence of noise:

$$x[n] = f(\boldsymbol{\psi}, n) + q[n], \quad n = 0, 1, \cdots, N-1 \qquad (12.18)$$

where $f(\cdot)$ is the known function of the parameter vector $\boldsymbol{\psi}$ and $q[n]$ is the noise

The task is to find $\boldsymbol{\psi}$ from $x[n]$

The maximum likelihood approach aims to provide optimum estimation for $\boldsymbol{\psi}$ when the PDF of $q[n]$ is available

Assume $q[n]$ is zero-mean Gaussian variable with variance σ_q^2

Zero-mean implies that the mean value of $x[n]$ is:

$$E\{x[n]\} = f(\boldsymbol{\psi}, n) \qquad (12.19)$$

The PDF of each scalar $x[n]$ parameterized by $\boldsymbol{\psi}$ is:

$$p(x[n]; \boldsymbol{\psi}) = \frac{1}{\sqrt{2\pi\sigma_q^2}} e^{-\frac{1}{2\sigma_q^2}\left(x[n]-f(\boldsymbol{\psi},n)\right)^2} \qquad (12.20)$$

Defining $\mathbf{x} = [x[0]\ x[1]\ \cdots\ x[N-1]]^T$, $\mathbf{q} = [q[0]\ q[1]\ \cdots\ q[N-1]]^T$ and $\mathbf{f}(\boldsymbol{\psi}) = [f(\boldsymbol{\psi},0)\ f(\boldsymbol{\psi},1)\ \cdots\ f(\boldsymbol{\psi},N-1)]^T$, we extend (12.20) to $\mathbf{x}$:

$$p(\mathbf{x}; \boldsymbol{\psi}) = \frac{1}{(2\pi)^{N/2}|\mathbf{C}|^{1/2}} e^{-\frac{1}{2}\left(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi})\right)^T \mathbf{C}^{-1}\left(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi})\right)} \qquad (12.21)$$

where $\mathbf{C}$ is called the covariance matrix:

$$\mathbf{C} = E\{(\mathbf{x} - \mathbf{f}(\boldsymbol{\psi}))(\mathbf{x} - \mathbf{f}(\boldsymbol{\psi}))^T\} = E\{\mathbf{q}\mathbf{q}^T\} \qquad (12.22)$$

The maximum likelihood estimate for $\boldsymbol{\psi}$ is

$$\hat{\boldsymbol{\psi}} = \arg\max_{\boldsymbol{\psi}} p(\mathbf{x}; \boldsymbol{\psi}) \qquad (12.23)$$

For white $q[n]$, according to (12.1)-(12.2), $\mathbf{C}$ is simply determined as:

$$\mathbf{C} = \sigma_q^2 \mathbf{I} \qquad (12.24)$$

where $\mathbf{I}$ is the identity matrix, resulting in $|\mathbf{C}| = \sigma_q^{2N}$ and $\mathbf{C}^{-1} = \sigma_q^{-2}\mathbf{I}$

As a result, the PDF of $\mathbf{x}$ is simplified as:

$$p(\mathbf{x};\boldsymbol{\psi}) = \frac{1}{(2\pi\sigma_q^2)^{N/2}} e^{-\frac{1}{2\sigma_q^2}\left(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi})\right)^T\left(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi})\right)} \quad (12.25)$$

Maximizing $p(\mathbf{x};\boldsymbol{\psi})$ is equal to minimizing $(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi}))^T(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi}))$, the maximum likelihood estimate is also equal to:

$$\hat{\boldsymbol{\psi}} = \arg\min_{\boldsymbol{\psi}} \left[(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi}))^T(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi}))\right] \quad (12.26)$$

- Least Squares

Compared with the maximum likelihood approach, it does not require the PDF information but its estimation performance may be inferior

The least squares estimate for $\boldsymbol{\psi}$ is:

$$\begin{aligned}\hat{\boldsymbol{\psi}} &= \arg\min_{\boldsymbol{\psi}} \left[(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi}))^T(\mathbf{x}-\mathbf{f}(\boldsymbol{\psi}))\right] \\ &= \arg\min_{\boldsymbol{\psi}} \sum_{n=0}^{N-1} [x[n]-f(\boldsymbol{\psi},n)]^2 \end{aligned} \qquad (12.27)$$

That is, the maximum likelihood and least squares estimates are equivalent for white noise

Note that when $f(\boldsymbol{\psi},n)$ is a linear function of $\boldsymbol{\psi}$, then $\hat{\boldsymbol{\psi}}$ can be simplified to a global closed-form solution

Example 12.2

Given N samples of $x[n]$ which has the form of:

$$x[n] = \alpha n + \beta + q[n], \quad n = 0, 1, \cdots, N-1$$

where α and β are the parameters of interest and $q[n]$ is an unknown zero-mean noise sequence. Determine the least squares estimates for α and β. This problem is also known as linear regression in statistics.

According to (12.27), the least squares cost function is:

$$J(\alpha, \beta) = \sum_{n=0}^{N-1} \left(x[n] - \alpha n - \beta\right)^2$$

which is quadratic with parameters α and β.

To find the least squares estimates, we differentiate $J(\alpha,\beta)$ with respect to α and β, and then set the resultant expressions to zero:

$$\left.\frac{\partial J(\alpha,\beta)}{\partial \alpha}\right|_{\alpha=\hat{\alpha},\beta=\hat{\beta}} = 2\sum_{n=0}^{N-1}\left(x[n]-\hat{\alpha}n-\hat{\beta}\right)(-n)=0$$

$$\Rightarrow \sum_{n=0}^{N-1} n x[n] = \hat{\alpha}\sum_{n=0}^{N-1} n^2 + \hat{\beta}\sum_{n=0}^{N-1} n$$

$$\left.\frac{\partial J(\alpha,\beta)}{\partial \beta}\right|_{\alpha=\hat{\alpha},\beta=\hat{\beta}} = 2\sum_{n=0}^{N-1}\left(x[n]-\hat{\alpha}n-\hat{\beta}\right)(-1)=0$$

$$\Rightarrow \sum_{n=0}^{N-1} x[n] = \hat{\alpha}\sum_{n=0}^{N-1} n + N\hat{\beta}$$

Hence:

$$\begin{bmatrix} \sum_{n=0}^{N-1} n^2 & \sum_{n=0}^{N-1} n \\ \sum_{n=0}^{N-1} n & N \end{bmatrix} \begin{bmatrix} \hat{\alpha} \\ \hat{\beta} \end{bmatrix} = \begin{bmatrix} \sum_{n=0}^{N-1} n x[n] \\ \sum_{n=0}^{N-1} x[n] \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} \hat{\alpha} \\ \hat{\beta} \end{bmatrix} = \begin{bmatrix} \sum_{n=0}^{N-1} n^2 & \sum_{n=0}^{N-1} n \\ \sum_{n=0}^{N-1} n & N \end{bmatrix}^{-1} \begin{bmatrix} \sum_{n=0}^{N-1} n x[n] \\ \sum_{n=0}^{N-1} x[n] \end{bmatrix}$$

which corresponds to a closed-form global solution.

The MATLAB program is provided as `ex12_2.m`

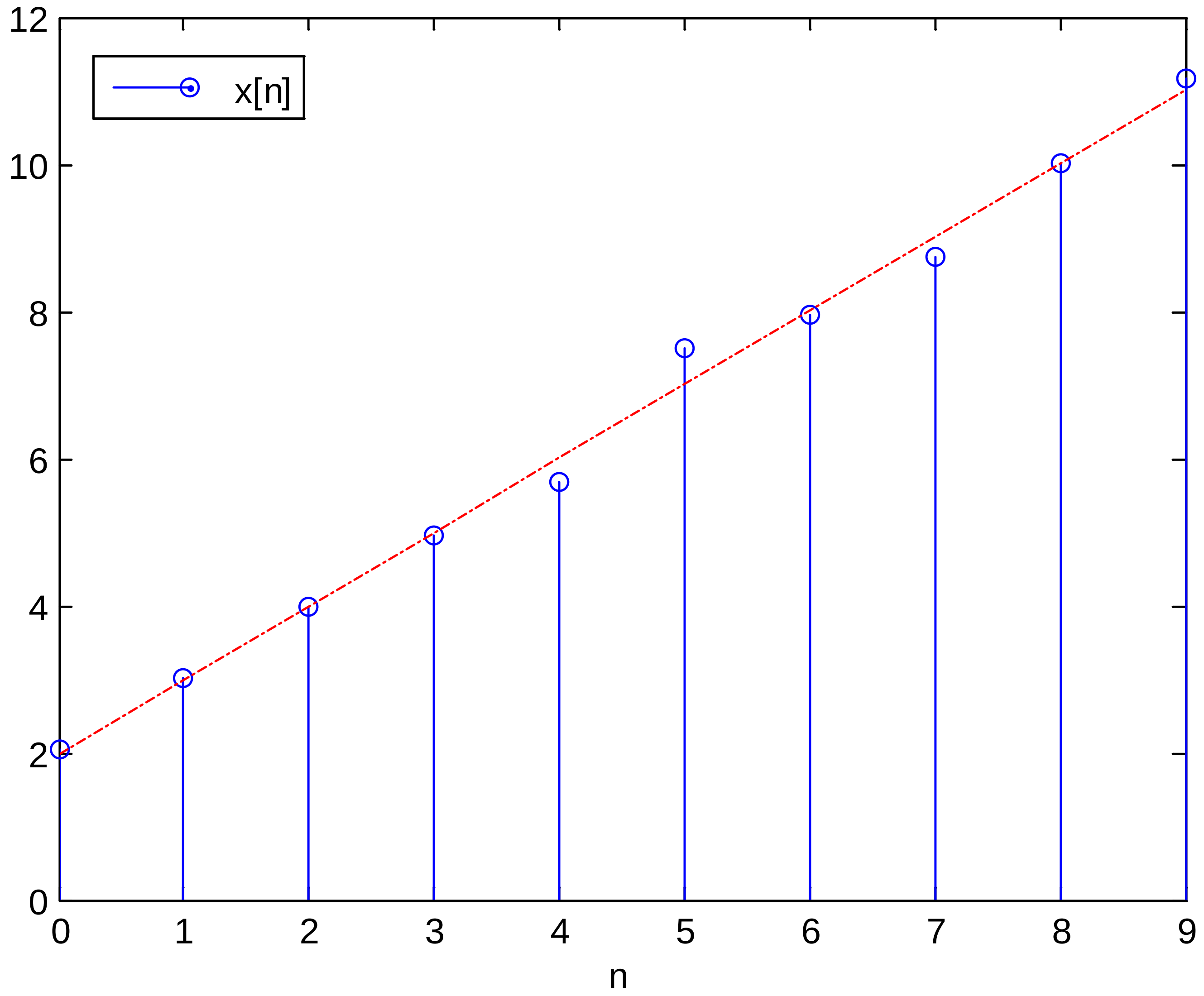


Fig.12.2: Linear regression with $\alpha = 1$, $\beta = 2$ and $N = 10$

- Iterative Numerical Methods

When the parameters of interest are not linear in the observed data, the corresponding cost functions based on the maximum likelihood and least squares methods contain local minima and maxima

If we have an initial estimate $\hat{\boldsymbol{\psi}}_0$ sufficiently close to the global optimum, $\hat{\boldsymbol{\psi}}$ can be obtained using iterative schemes:

Assuming minimization of $J(\boldsymbol{\psi})$, Netwon-Raphson method is

$$\hat{\boldsymbol{\psi}}_{k+1} = \hat{\boldsymbol{\psi}}_k - \mathbf{H}^{-1}(J(\hat{\boldsymbol{\psi}}_k))\boldsymbol{\nabla}(J(\hat{\boldsymbol{\psi}}_k)) \qquad (12.28)$$

where $\mathbf{H}(J(\hat{\boldsymbol{\psi}}_k))$ is the Hessian matrix and $\boldsymbol{\nabla}(J(\hat{\boldsymbol{\psi}}_k))$ is the gradient vector, computed at the k-th iteration estimate $\hat{\boldsymbol{\psi}}_k$.

Steepest descent method is

$$\hat{\boldsymbol{\psi}}_{k+1} = \hat{\boldsymbol{\psi}}_k - \mu \boldsymbol{\nabla}(J(\hat{\boldsymbol{\psi}}_k)) \tag{12.29}$$

where μ is a positive constant which controls the convergence rate and stability of the algorithm

Typical choices of stopping criteria include number of iterations and $|\hat{\boldsymbol{\psi}}_{k+1} - \hat{\boldsymbol{\psi}}_k| < \epsilon$ where ϵ is a sufficiently small positive constant

The Newton-Raphson algorithm provides fast convergence but matrix inverse is required

The steepest descent method is stable but its convergence rate is slow

Applications

- Signal Generation with Non-Integer Sample Delay

Given a discrete-time sequence $x[n]$, we can generate $x[n-\Delta]$ where Δ is not an integer, according to the time shifting property of discrete-time Fourier transform (DTFT) as follows. The DTFT transform pair for $x[n-\Delta]$ is:

$$x[n-\Delta] \leftrightarrow e^{-j\omega\Delta} X(e^{j\omega}) \qquad (12.30)$$

The ideal frequency response for a non-integer sample delay is:

$$H_d(e^{j\omega}) = e^{-j\omega\Delta} \qquad (12.31)$$

Applying the inverse DTFT, the ideal impulse response is:

$$h_d[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\omega})e^{j\omega n}d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{j\omega(n-\Delta)}d\omega = \mathrm{sinc}(n-\Delta) \quad (12.32)$$

In theory, $x[n-\Delta]$ is computed as:

$$x[n-\Delta] = \sum_{k=-\infty}^{\infty} x[n-k]\mathrm{sinc}\left(k-\Delta\right) \quad (12.33)$$

which aligns with Example 4.3.

We utilize time-shifting and truncation to obtain a practical solution:

$$y[n] = \sum_{k=0}^{2M} x[n-k]\mathrm{sinc}\left(k - M - \Delta\right) \qquad (12.34)$$

which corresponds to a causal system because only $x[n-2M], x[n-2M+1], \cdots, x[n]$, are required at time n

The value of M should be chosen sufficiently large to reduce the truncation error

The relationship between $y[n]$ and $x[n-\Delta]$ is:

$$y[n] \approx x[n - \Delta - M] \qquad (12.35)$$

That is, the desired time-shifted signal is obtained after a delay of M samples

Example 12.3

Design a causal finite impulse response (FIR) system with 10 coefficients to approximate a time-shifter whose input is $x[n]$ and output is $y[n] = x[n-\Delta]$ where $\Delta = 0.8$.

From (12.32), the ideal impulse response is:

$$h_d[n] = \mathrm{sinc}(n - 0.8)$$

Investigating $h_d[n]$ for $-5 \le n \le 6$, we have $h_d[n] = \{-0.0323, 0.0390, -0.0492, 0.0668, -0.1039, 0.2339, 0.9355, -0.1559, 0.0850, -0.0585, 0.0445, -0.0360\}$ with $h_d[-5] = -0.0323$. The coefficients $h_d[-4], h_d[-3], \cdots, h_d[5]$, are chosen because they have the largest energy. Similar to (12.34), $y[n]$ is:

$$y[n] = \sum_{k=0}^{9} x[n-k]\mathrm{sinc}\left(k - 4 - 0.8\right)$$

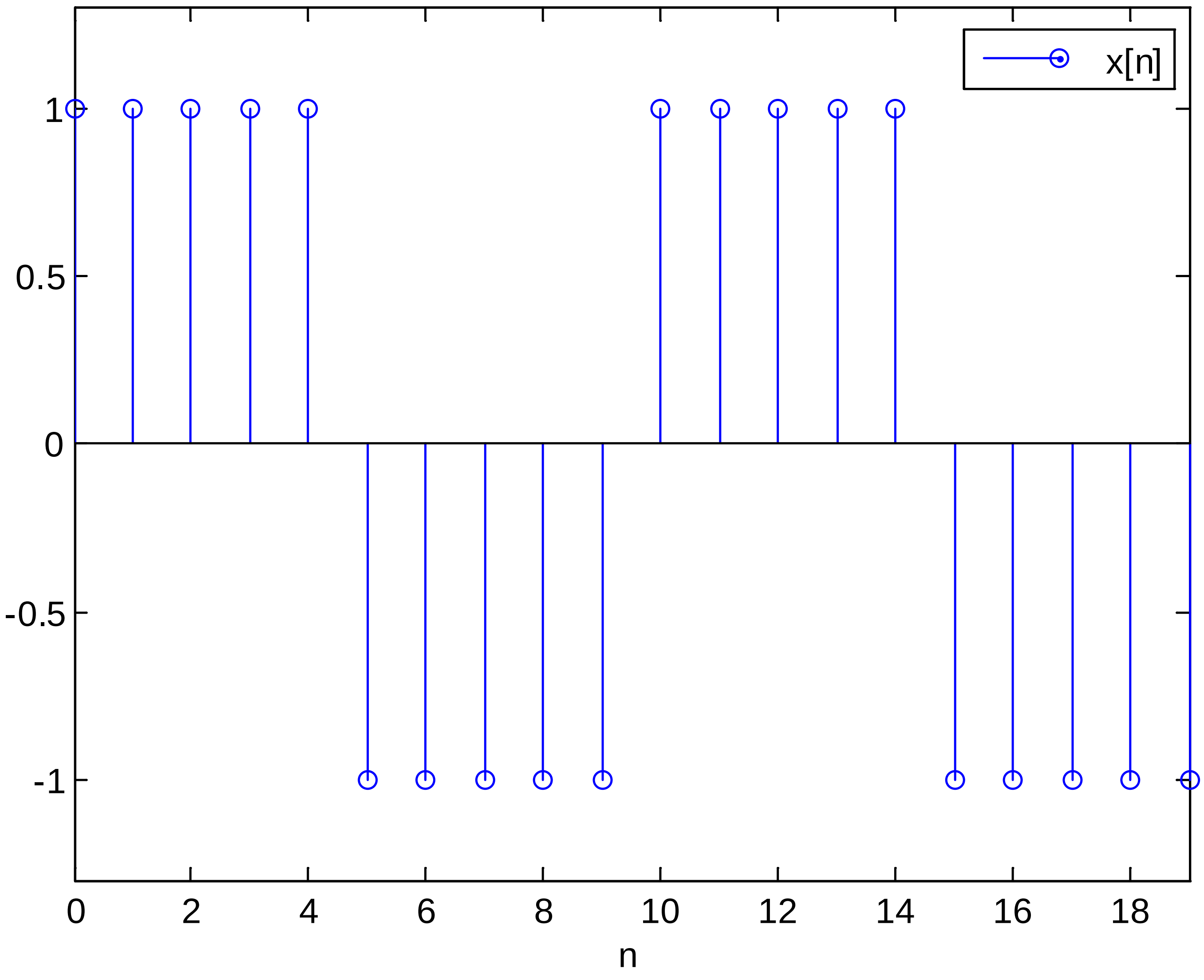


Fig.12.3: Square wave sequence

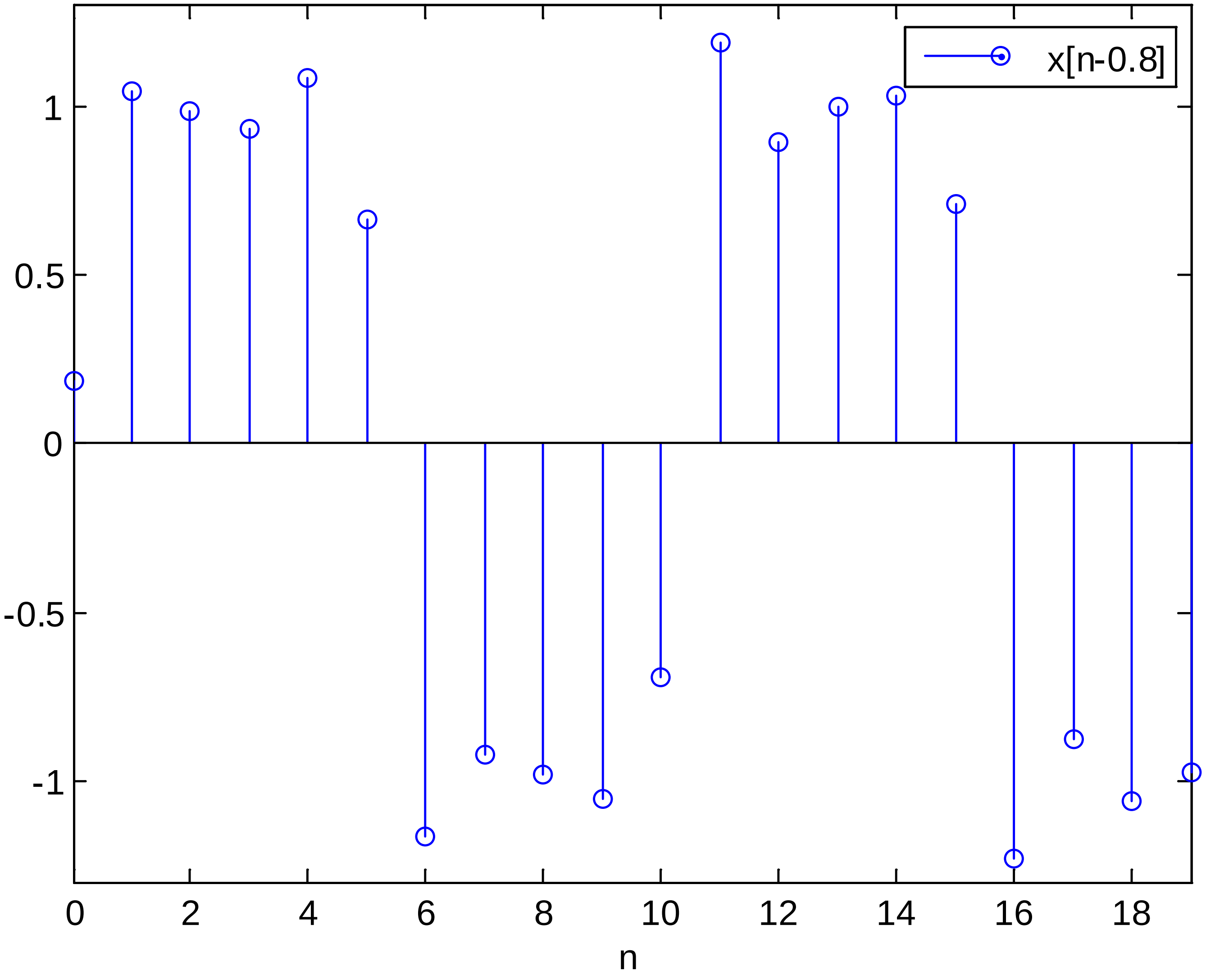
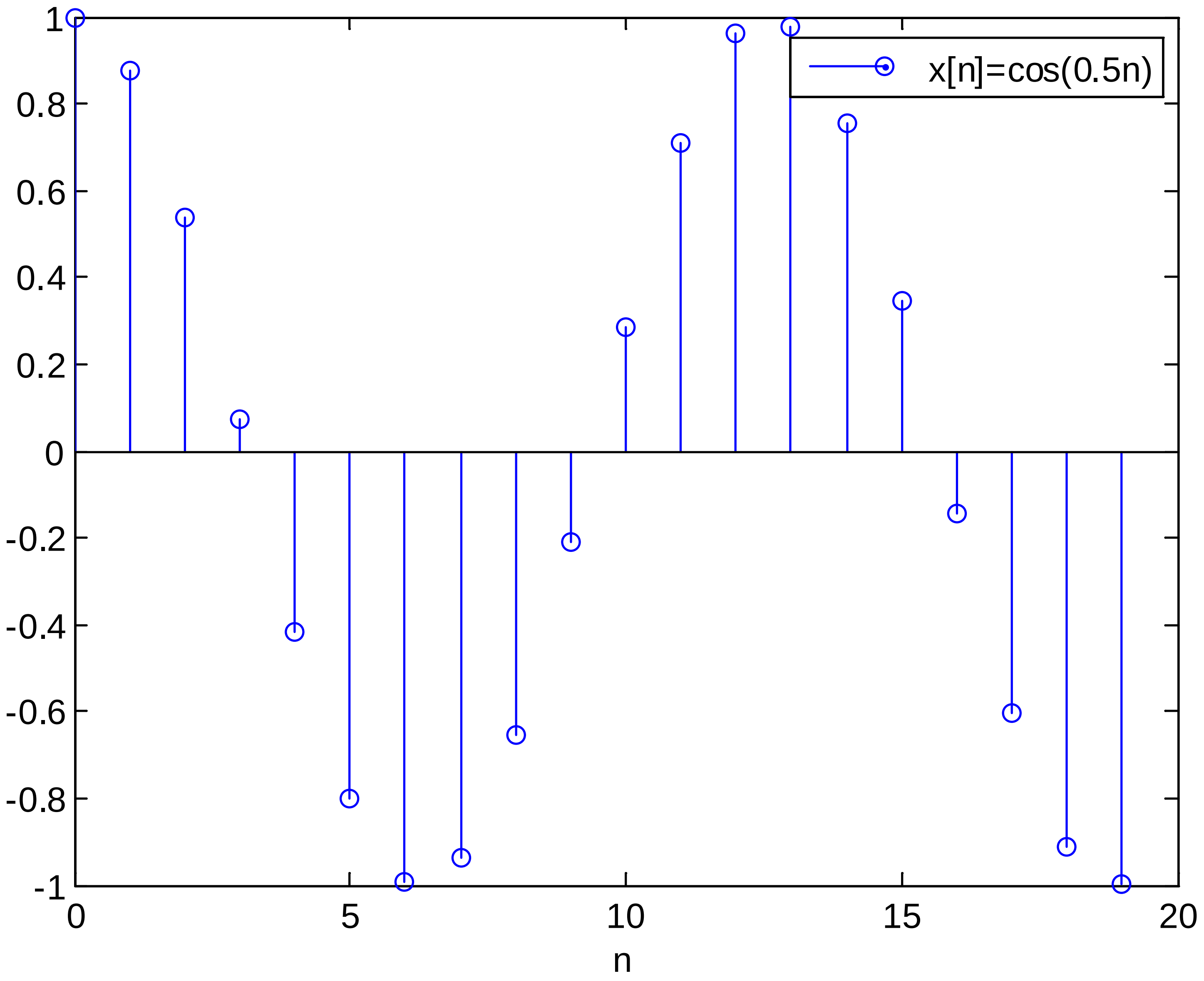
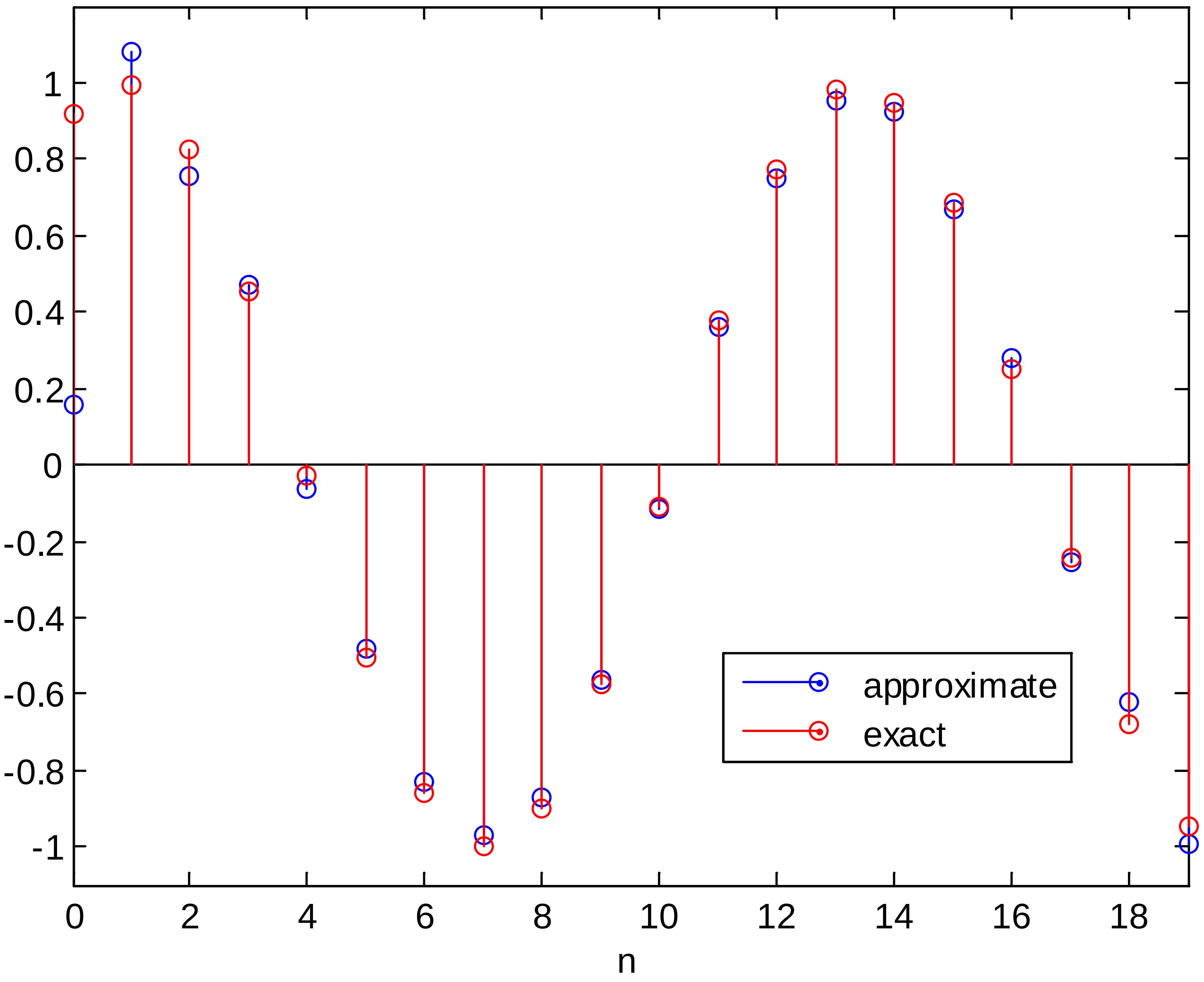


Fig.12.4: Time-shifted square wave sequence


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
5
10
15
20
n
x[n]=cos(0.5n)


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10
12
14
16
18
n
approximate
exact

The MATLAB program is provided as `ex12_3.m`

- Moving Average for Security Analysis

MA is one of the standard technical indicators for security analysis, which aims to identify the trend direction or define potential support and resistance levels. Basically, it is the average price of a security over a specified time period. The MA signal $y[n]$ for $x[n]$ in security analysis is:

$$y[n] = \frac{1}{N}\sum_{k=0}^{N-1} x[n-k] \qquad (12.36)$$

(12.36) can be viewed as $x[n] \otimes h[n]$ where $h[n]$ is:

$$h[n] = \begin{cases} \frac{1}{N}, & 0 \leq n \leq N-1 \\ 0, & \text{otherwise} \end{cases} \qquad (12.37)$$

Example 12.4

Plot the 20-day MA for the Dow Jones Industrial Average (DJIA), which is a well-known stock market index, at the year of 2009.

According to (12.36), $N = 20$ and $x[n]$ corresponds to the close prices of the DJIA. Note that apart from the 252 trading days of 2009, we also need the last 19 close prices of 2008 for computation.

The MA curve is a smoothed and delayed version

The smoothing is resulted from the lowpass filtering process where the high frequency components in $x[n]$ are removed.

There is a delay because 19 previous samples are employed

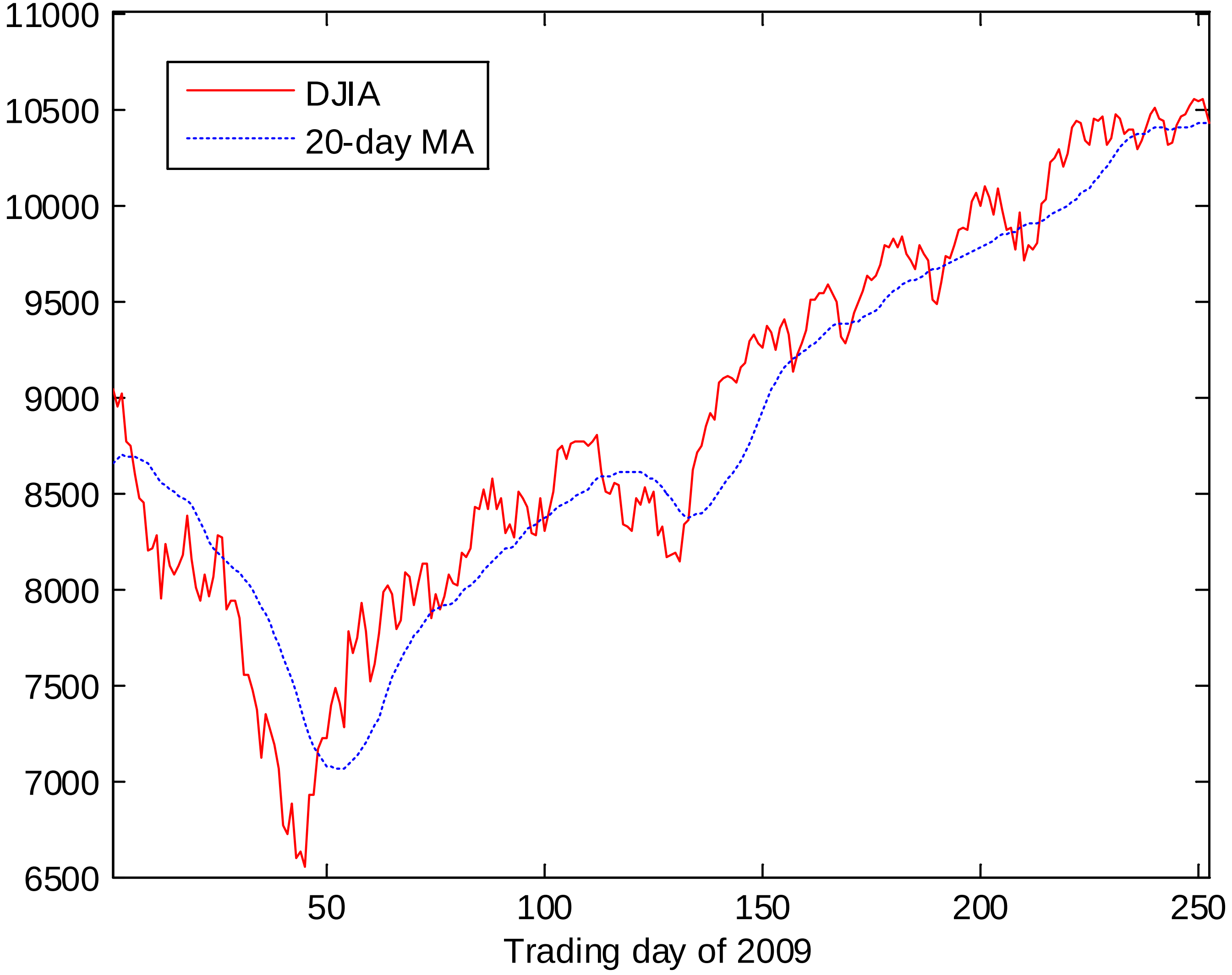


Fig.12.5: 20-day MA for Dow Jones Industrial Average at 2009

The MATLAB program is provided as `ex12_4.m`

- Frequency Shift-Keying Signal Detection

Frequency-shift keying (FSK) is a simple frequency modulation scheme in digital communications

The basic idea is to use $L=2^v$ distinct frequencies, say, ω_i, $i=0,1,\cdots,L-1$, to represent a v-bit symbol for data transmission

The transmitted FSK signal is modeled as:

$$s_i[n]=\cos(\omega_i n), \quad i=0,1,\cdots,L-1, \quad n=0,1,\cdots,N-1 \qquad (12.38)$$

where $\omega_l \neq \omega_k$ for $l \neq k$

Based on (12.16), it can be shown that $r_{s_i s_j}[l] = 0$ for all l when $i \neq j$, while $r_{s_i s_i}[0] \geq |r_{s_i s_i}[l]|$, $l \neq 0$, according to (12.12)

Suppose a noisy FSK signal $r[n]$ with unknown frequency is received

With the use of (12.15) and (12.17), the frequency index can be estimated as:

$$\hat{i} = \arg\max_i \sum_{n=0}^{N-1} r[n] s_i[n] \tag{12.39}$$

Consider $L = 2$ such that ω_0 and ω_1 represent bits “0” and “1”, respectively, bit “0” is detected if $\sum_{n=0}^{N-1} r[n] s_0[n] > \sum_{n=0}^{N-1} r[n] s_1[n]$. Otherwise, bit “1” is detected

- Spectral Analysis using Discrete Time Fourier Transform

As DTFT $X(e^{j\omega})$ shows the frequency components of its time-domain representation $x[n]$, we can find the signal frequency information by investigating the magnitude response $|X(e^{j\omega})|$.

Example 12.5

Determine the fundamental frequency of the speech segment in Fig.1.1

The sampling frequency is 22000 Hz. We use the MATLAB command `freqz` to plot the frequency magnitude response

There are three peaks at 245 Hz, 493 Hz and 740 Hz. As voiced speech is periodic, and $245 \times 2 \approx 493$ and $245 \times 3 \approx 740$, we deduce that the speech fundamental frequency is 245 Hz

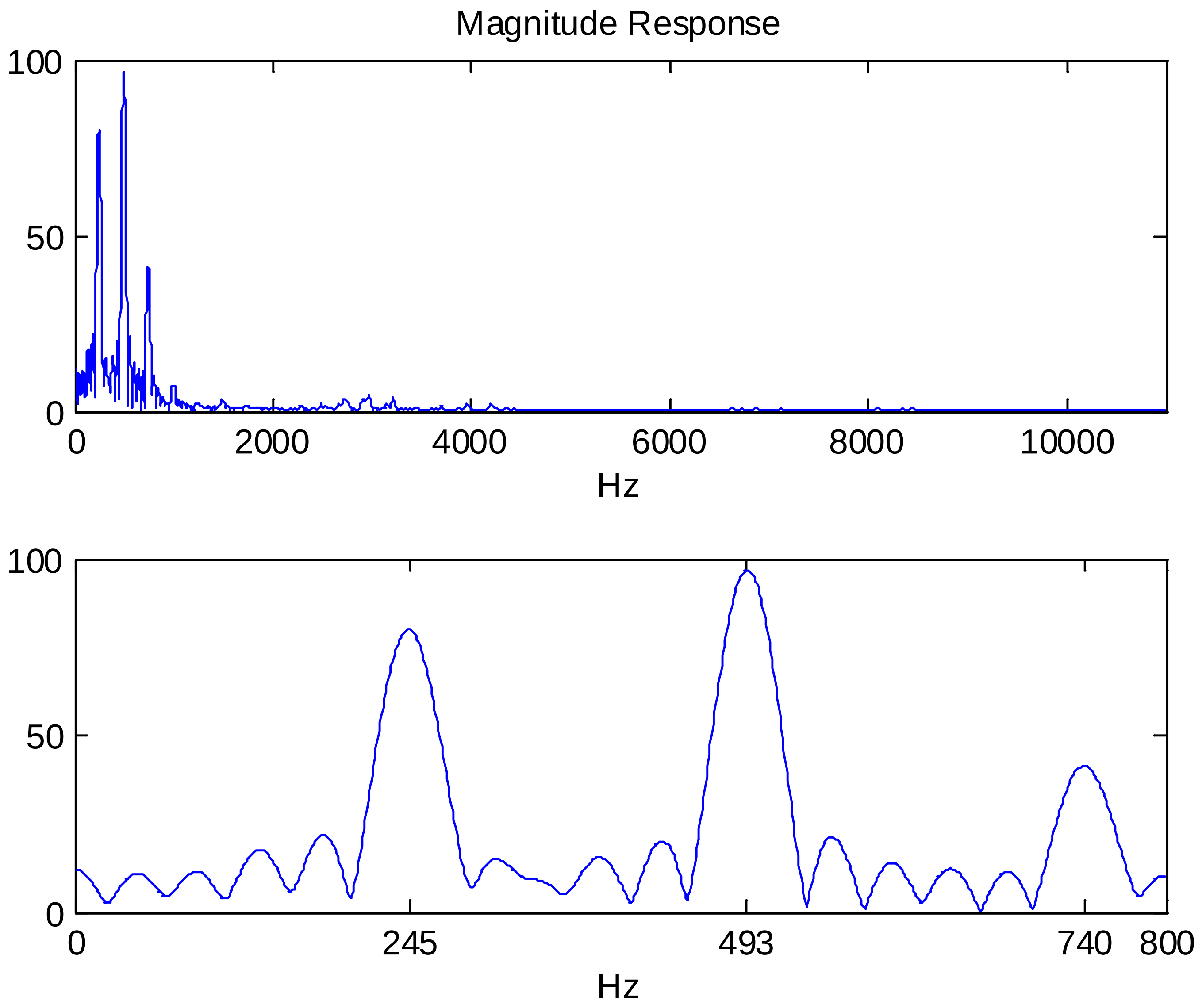


Fig.12.6: Magnitude plots for speech segment

The MATLAB program is provided as `ex12_5.m`

- Noise Reduction using Discrete Fourier Transform

Suppose we are given a noisy signal $r[n]$:

$$r[n] = s[n] + q[n], \quad n = 0, 1, \cdots, N-1 \tag{12.40}$$

where $s[n]$ is the signal of interest and $q[n]$ is the additive noise

The task is to extract $s[n]$ from $r[n]$

Suppose $s[n]$ is a narrowband signal while $q[n]$ is a wideband sequence, discrete Fourier transform (DFT) may be utilized to retrieve $s[n]$

First, we compute $R[k]$ from $r[n]$. An estimate of $S[k]$, denoted by $\hat{S}[k]$, is then obtained as:

$$\hat{S}[k] = \begin{cases} R[k], & \{\omega_k\} \text{ correspond to } s[n] \\ 0, & \text{otherwise} \end{cases} \tag{12.41}$$

where $\omega_k = 2\pi k/N$, $k = 0, 1, \cdots, N-1$. Finally, the estimate of $s[n]$ is given by the inverse DFT of $\hat{S}[k]$

Example 12.6

Given N samples of a noisy signal $r[n]$ of the form:

$$r[n] = s[n] + q[n], \quad n = 0, 1, \cdots, N-1$$

where $s[n]$ is a real sinusoid and $q[n]$ is a white noise with variance σ_q^2. Estimate $s[n]$ from $r[n]$.

A MATLAB simulation study with $s[n] = \sin(0.2n + 1)$, $\sigma_q^2 = 0.3$ and $N = 100$, is performed

The two peaks and the remaining coefficients roughly correspond to the narrowband $s[n]$ and wideband $q[n]$, respectively

As a result, we only keep the two peaks as $\hat{S}[k]$ and the estimate of $s[n]$

The MATLAB program is provided as `ex12_6.m`

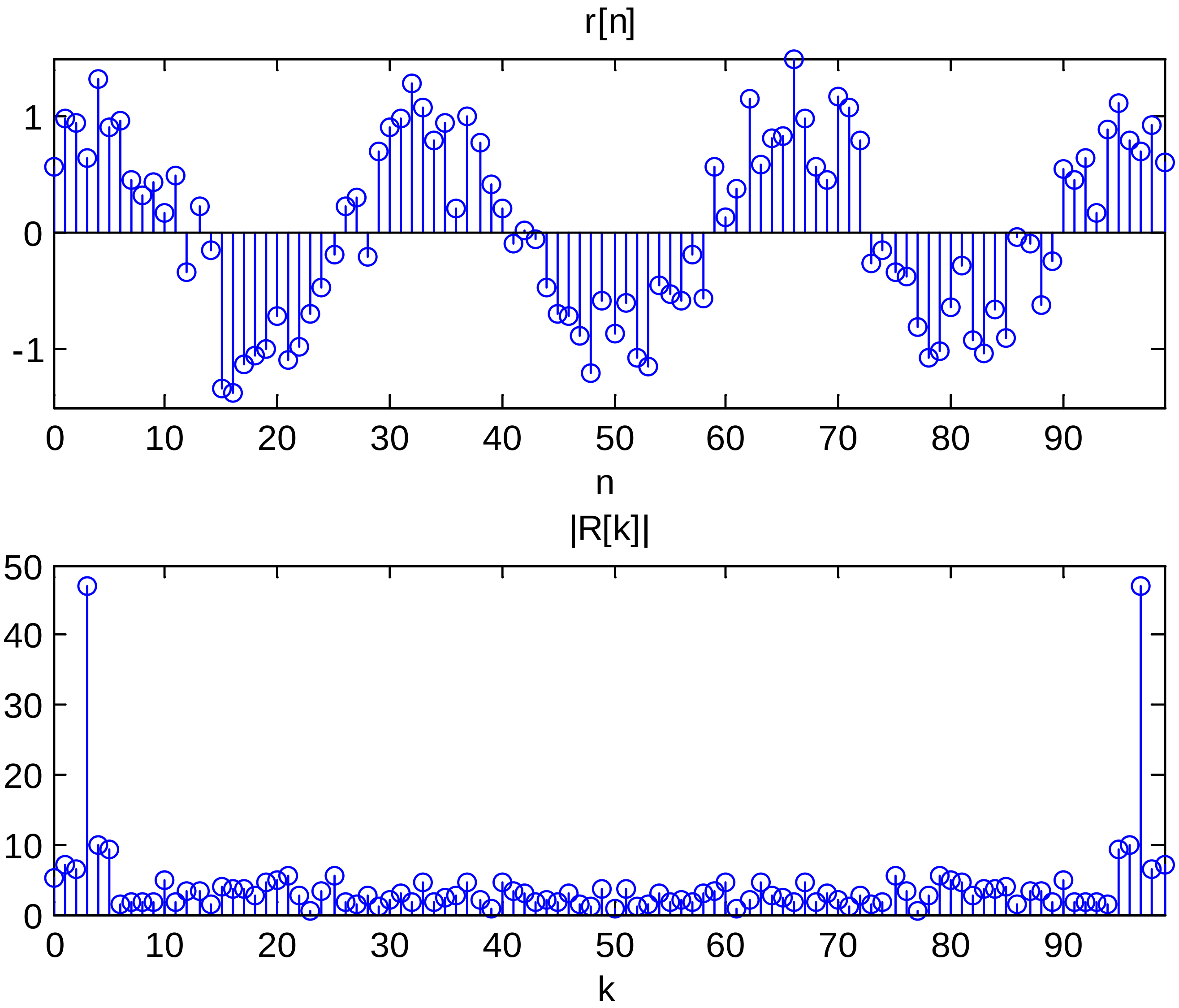


Fig.12.7: Plots for noisy sequence

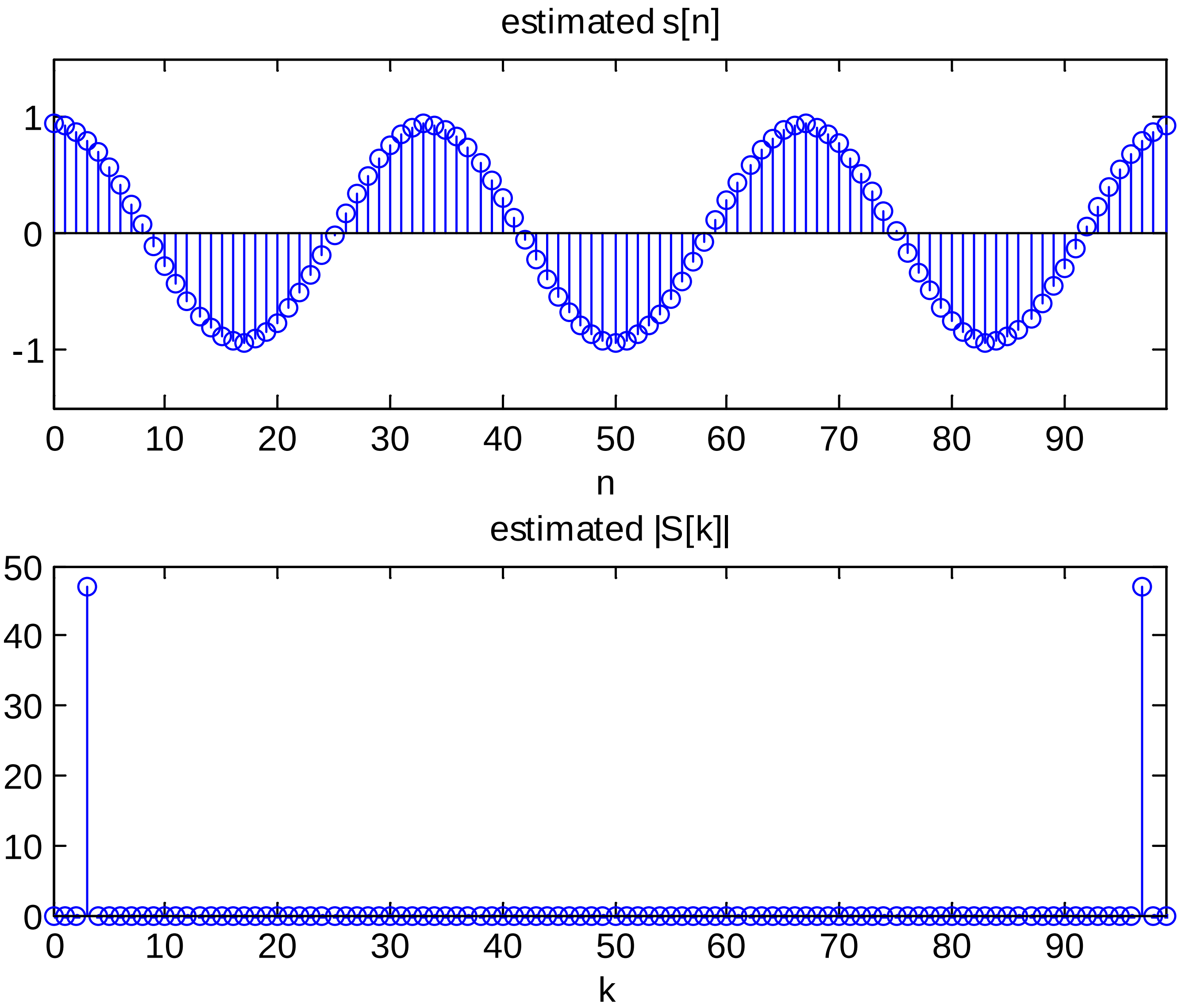


Fig.12.8: Plots for recovered signal

- Estimation of Constant in Noise

When $f(\boldsymbol{\psi}, n) = \alpha$, (12.18) becomes

$$x[n] = \alpha + q[n], \quad n = 0, 1, \cdots, N-1 \tag{12.42}$$

Assume the noise $q[n]$ is a white Gaussian process with variance $\sigma^2_{q_n}$, $n = 0, 1, \cdots, N-1$

The noise covariance matrix is now equal to $\mathbf{C} = \mathrm{diag}(\sigma^2_{q_0}, \sigma^2_{q_1}, \cdots, \sigma^2_{q_{N-1}})$ which is a diagonal matrix of $\{\sigma^2_{q_n}\}$

The maximum likelihood estimate of α is

$$\hat{\alpha} = \arg\min_{\alpha} J(\alpha) \tag{12.43}$$

where

$$J(\alpha) = \sum_{n=0}^{N-1} \frac{(x[n]-\alpha)^2}{\sigma_{q_n}^2} \tag{12.44}$$

Differentiating $J(\alpha)$ with respect to α and then setting the resultant expression to zero, we get:

$$\left.\frac{dJ(\alpha)}{\partial\alpha}\right|_{\alpha=\hat{\alpha}} = 2\sum_{n=0}^{N-1} \frac{(x[n]-\hat{\alpha})(-1)}{\sigma_{q_n}^2} = 0 \Rightarrow \hat{\alpha} = \frac{\sum_{n=0}^{N-1} \sigma_{q_n}^{-2} x[n]}{\sum_{n=0}^{N-1} \sigma_{q_n}^{-2}} \tag{12.45}$$

where each $x[n]$ is weighted by $1/\sigma_{q_n}^2$. That is, a smaller weight is used for $x[n]$ with a larger noise and vice versa

The least squares solution is obtained by substituting $\sigma_{q_n}^2 = 1$:

$$\hat{\alpha} = \frac{1}{N}\sum_{n=0}^{N-1} x[n] \tag{12.46}$$

which is simply the average value of $x[n]$

In terms of estimation performance, (12.45) is more accurate than (12.46) but the latter has the advantage that prior knowledge of the noise powers is not required

- Time Delay Estimation

The problem is to find the time-shift between two versions of the same signal

Radar ranging is a representative application

From Fig.1.3 and let the transmitted and received signals be $s[n]$ and $r[n]$, respectively, the time delay estimation problem is to find τ with the use of $s[n]$ and $r[n]$

The received signal $r[n]$ is:

$$r[n] = \alpha s[n-\tau] + q[n] \qquad (12.47)$$

where α is the attenuation parameter, τ is the round-trip propagation time, and $q[n]$ is a zero-mean noise.

For simplicity, here we assume that τ is an integer

We can use correlation to estimate τ:

$$\hat{\tau} = \arg\max_{\tau} \sum_n r[n]s[n-\tau] \qquad (12.48)$$

Or using least squares approach where the cost function is:

$$J(\alpha, \tau) = \sum_n \left(r[n] - \alpha s[n-\tau]\right)^2 \qquad (12.49)$$

which contains two variables

Differentiate $J(\alpha, \tau)$ with respect to α and set the resultant expression to zero to find the least squares estimate of α in terms of τ, and then substitute it back to (12.49)

The least squares solution of τ can also be written as

$$\hat{\tau} = \arg\min_{\tau} \sum_n \left(r[n] - \frac{\sum_k r[k]s[k-\tau]}{\sum_k s^2[k-\tau]} s[n-\tau] \right)^2 \qquad (12.50)$$

- Sinusoidal Parameter Estimation

The problem is to find the amplitudes, frequencies and phases of tones embedded in noise

The simplest model corresponds to $L=1$ in (12.3):

$$x[n] = \alpha\cos(\omega n + \phi) + q[n], \quad n = 0, 1, \cdots, N-1 \qquad (12.51)$$

Assume $q[n]$ is white Gaussian with variance σ_q^2

The maximum likelihood estimate is equal to the least squares solution:

$$\hat{\boldsymbol{\psi}} = \arg\min_{\boldsymbol{\psi}} J(\boldsymbol{\psi}), \quad \boldsymbol{\psi} = [\alpha \ \omega \ \phi]^T \tag{12.52}$$

where

$$J(\boldsymbol{\psi}) = \sum_{n=0}^{N-1} \left[x[n] - \alpha\cos(\omega n + \phi)\right]^2 \tag{12.53}$$

which corresponds to a multimodal function

With an initial estimate $\hat{\boldsymbol{\psi}}_0$ sufficiently close to the global minimum of (12.53), the steepest descent algorithm for finding $\boldsymbol{\psi}$ is:

$$\begin{bmatrix} \hat{\alpha}_{k+1} \\ \hat{\omega}_{k+1} \\ \hat{\phi}_{k+1} \end{bmatrix} = \begin{bmatrix} \hat{\alpha}_{k} \\ \hat{\omega}_{k} \\ \hat{\phi}_{k} \end{bmatrix} - \mu \begin{bmatrix} \dfrac{\partial J(\hat{\boldsymbol{\psi}}_k)}{\partial \hat{\alpha}_k} \\ \dfrac{\partial J(\hat{\boldsymbol{\psi}}_k)}{\partial \hat{\omega}_k} \\ \dfrac{\partial J(\hat{\boldsymbol{\psi}}_k)}{\partial \hat{\phi}_k} \end{bmatrix}$$

$$= \begin{bmatrix} \hat{\alpha}_{k} \\ \hat{\omega}_{k} \\ \hat{\phi}_{k} \end{bmatrix} - 2\mu \begin{bmatrix} -\displaystyle\sum_{n=0}^{N-1} \left[x[n] - \hat{\alpha}_k \cos(\hat{\omega}_k n + \hat{\phi}_k)\right] \cos(\hat{\omega}_k n + \hat{\phi}_k) \\ \displaystyle\sum_{n=0}^{N-1} \left[x[n] - \hat{\alpha}_k \cos(\hat{\omega}_k n + \hat{\phi}_k)\right] \sin(\hat{\omega}_k n + \hat{\phi}_k) n \\ \displaystyle\sum_{n=0}^{N-1} \left[x[n] - \hat{\alpha}_k \cos(\hat{\omega}_k n + \hat{\phi}_k)\right] \sin(\hat{\omega}_k n + \hat{\phi}_k) \end{bmatrix} \quad (12.54)$$