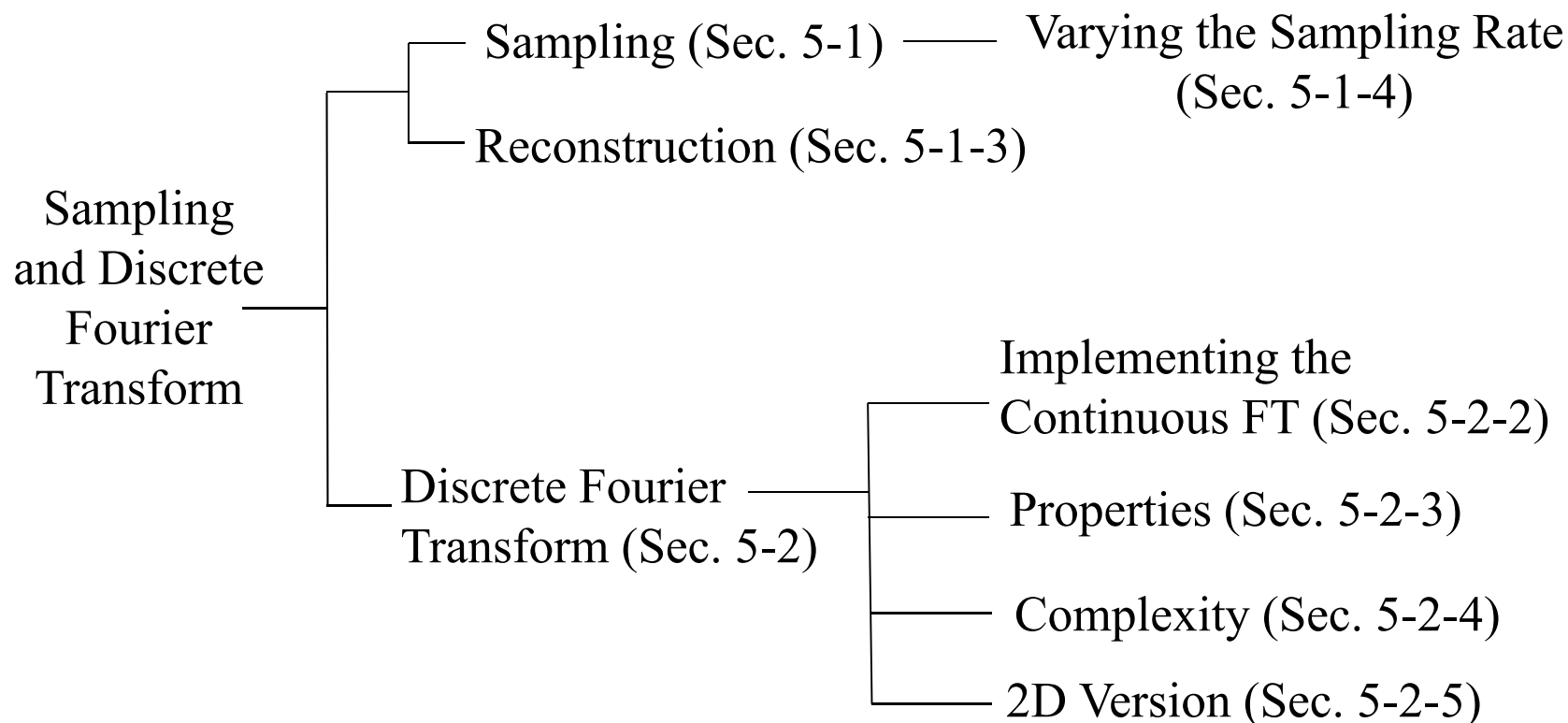


## 5. Sampling and Discrete Fourier Transform



[1] R. N. Bracewell, *The Fourier Transform and Its Applications*, 3<sup>rd</sup> ed., McGraw Hill, Boston, 2000.

[2] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*, London: Prentice-Hall, 3<sup>rd</sup> ed., 2010.

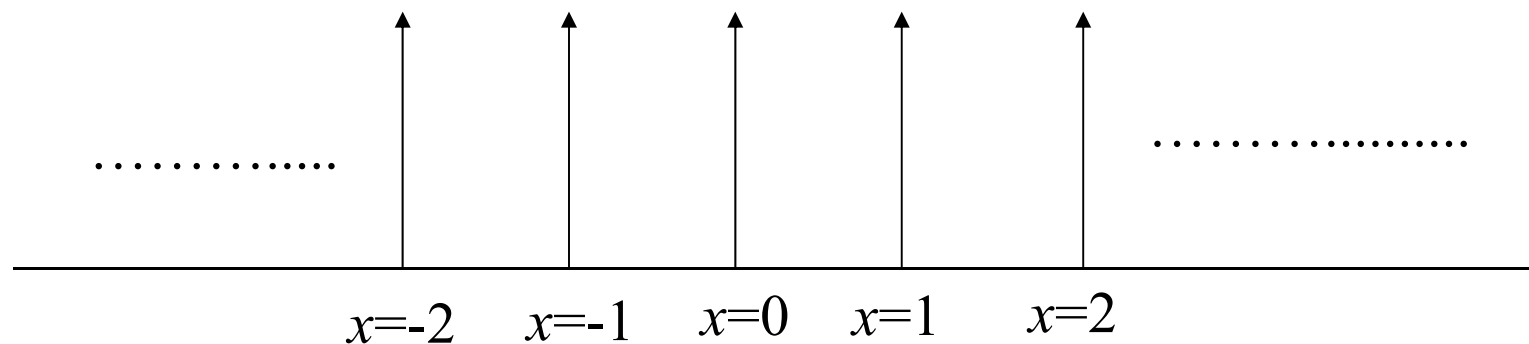
## 5.1 Sampling

### 5.1.1 Impulse Train

Impulse Train

$$p(x) = \sum_n \delta(x-n) = \cdots + \delta(x+1) + \delta(x) + \delta(x-1) + \delta(x-2) + \cdots$$

It is also called the **comb function**.



Signal sampling can be express in terms of the impulse train

$$\begin{aligned}
 g(x) \xrightarrow{\text{sampling}} g_s(x) &= g(x) \sum_n \delta(x - n\Delta_x) \\
 &= \sum_n g_n \delta(x - n\Delta_x) \quad \text{where } g_n = g(n\Delta_x)
 \end{aligned}$$

Since

$$\sum_n \delta(x - n\Delta_x) = \frac{1}{\Delta_x} \sum_n \delta\left(\frac{x}{\Delta_x} - n\right) = \frac{1}{\Delta_x} p\left(\frac{x}{\Delta_x}\right)$$

the sampled signal  $g_s(x)$  can be expressed in terms of

$$g_s(x) = \frac{1}{\Delta_x} g(x) p\left(\frac{x}{\Delta_x}\right)$$

**[Theorem 5.1.1]** The impulse train is also an eigenfunction of the Fourier transform, i.e.,

$$P(f) = \mathfrak{F}\{p(x)\} = \sum_n \delta(f - n)$$

(Proof): Note that the impulse train is a periodic function

$$p(x) = p(x + 1)$$

Therefore, it can be expanded by the Fourier series (page 323) of the complex form with  $T = 1$

$$p(x) = \sum_n c_n \exp(j2\pi nx)$$

where

$$c_n = \frac{1}{1} \int_{-1/2}^{1/2} p(x) \exp(-j2\pi nx) dx = \int_{-1/2}^{1/2} \delta(x) \exp(-j2\pi nx) dx = 1$$

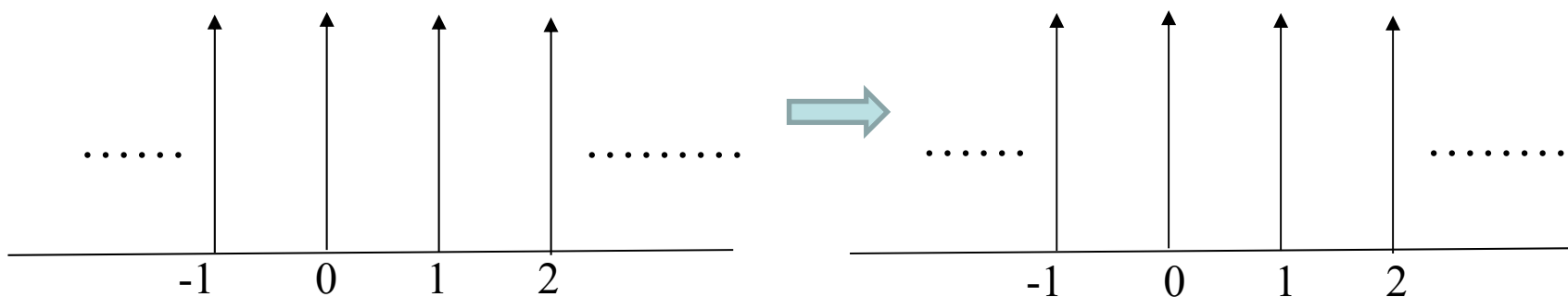
(page 344(2))

Therefore,

$$p(x) = \sum_n \exp(j2\pi nx)$$

$$\begin{aligned} \mathcal{F}[p(x)] &= \sum_n \mathcal{F}[\exp(j2\pi nx)] \\ &= \sum_n \delta(f - n) = p(f) \end{aligned}$$

$$\mathfrak{T}\left\{\sum_n \delta(x - n)\right\} = \sum_n \delta(f - n)$$

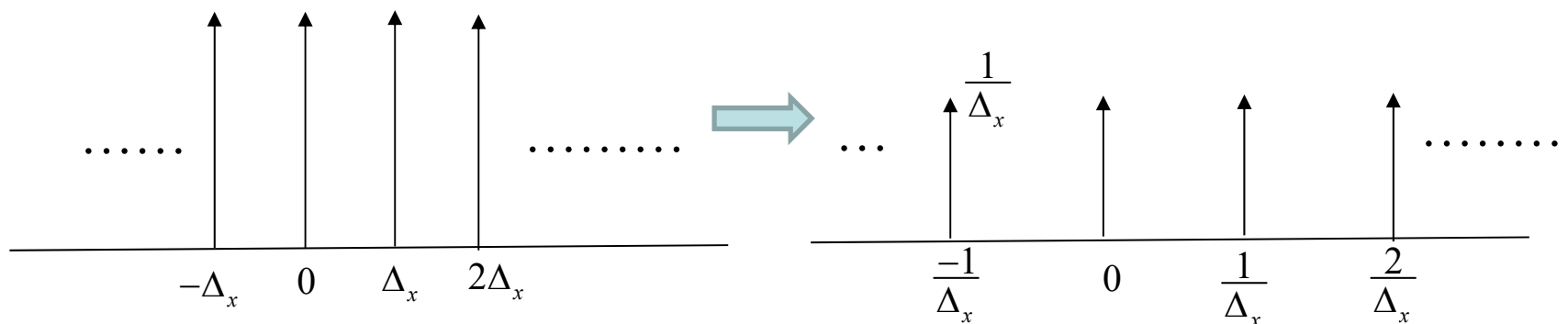


## Varying the interval of the impulse train

$$\sum_n \delta(x - n\Delta_x) = \frac{1}{\Delta_x} \sum_n \delta\left(\frac{x}{\Delta_x} - n\right) = \frac{1}{\Delta_x} p\left(\frac{x}{\Delta_x}\right) \quad (\text{page 345(4)})$$

$$\mathcal{F}\left[\sum_n \delta(x - n\Delta_x)\right] = \mathcal{F}\left[\frac{1}{\Delta_x} p\left(\frac{x}{\Delta_x}\right)\right] = p(\Delta_x f) \quad (\text{page 351(5)})$$

$$= \sum_n \delta(\Delta_x f - n) = \frac{1}{\Delta_x} \sum_n \delta\left(f - \frac{n}{\Delta_x}\right)$$



## 5.1.2 Sampling Theory

**[Theorem 5.1.2]** Suppose that we perform sampling for a continuous signal with sampling interval  $\Delta_x$

$$g(x) \xrightarrow{\text{sampling}} g_s(x) = g(x) \sum_n \delta(x - n\Delta_x)$$

$$= \sum_n g_n \delta(x - n\Delta_x) \quad \text{where}$$

then

$$G_s(f) = \frac{1}{\Delta_x} \sum_n G\left(f - \frac{n}{\Delta_x}\right)$$

$$g_n = g(n\Delta_x)$$

(Proof): Since  $g_s(x) = g(x) \frac{1}{\Delta_x} p\left(\frac{x}{\Delta_x}\right)$

$$\mathcal{F}[g_s(x)] = \mathcal{F}[g(x)] * \mathcal{F}\left[\frac{1}{\Delta_x} p\left(\frac{x}{\Delta_x}\right)\right]$$

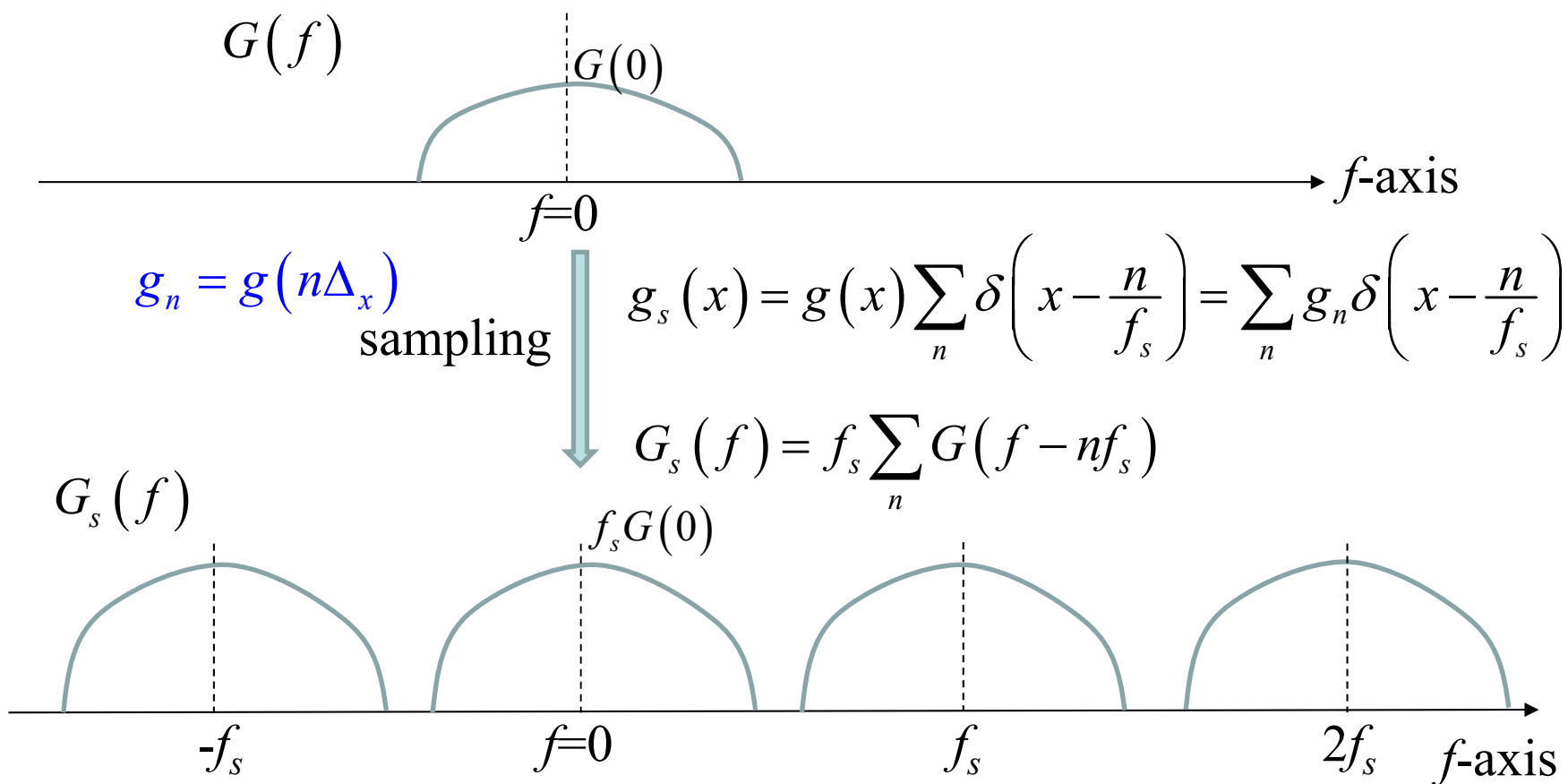
$$G_s(f) = G(f) * p(\Delta_x f) = \frac{1}{\Delta_x} \sum_n G(f) * \delta\left(f - \frac{n}{\Delta_x}\right) = \frac{1}{\Delta_x} \sum_n G\left(f - \frac{n}{\Delta_x}\right)$$

(page 389)

If we set  $f_s = \frac{1}{\Delta_x}$  ( $f_s$  is call the **sampling frequency**)

then

$$G_s(f) = f_s \sum_n G(f - nf_s)$$



## [Sampling Theory]

The sampling frequency  $f_s = \frac{1}{\Delta_x}$  should be larger than twice of the bandwidth of the original continuous function:

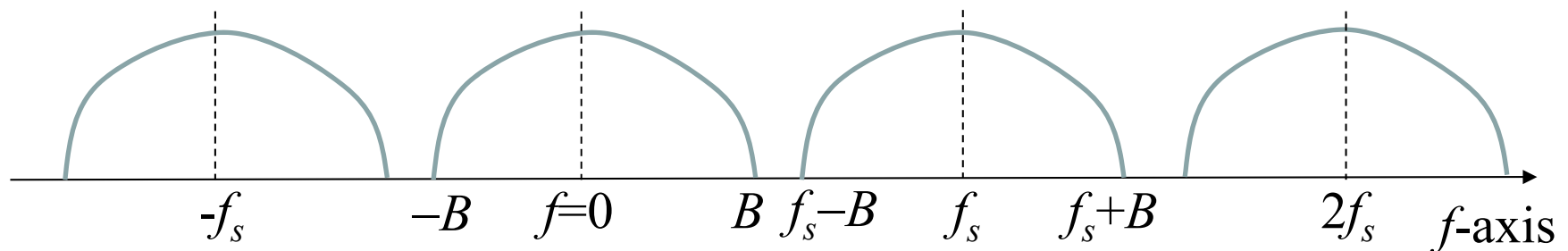
$$f_s - B > B$$

$$f_s > 2B \quad (\text{Nyquist criterion})$$

where

$$G(f) = 0 \quad \text{when } f > B.$$

Otherwise, the original function cannot be reconstructed and the aliasing effect is led.



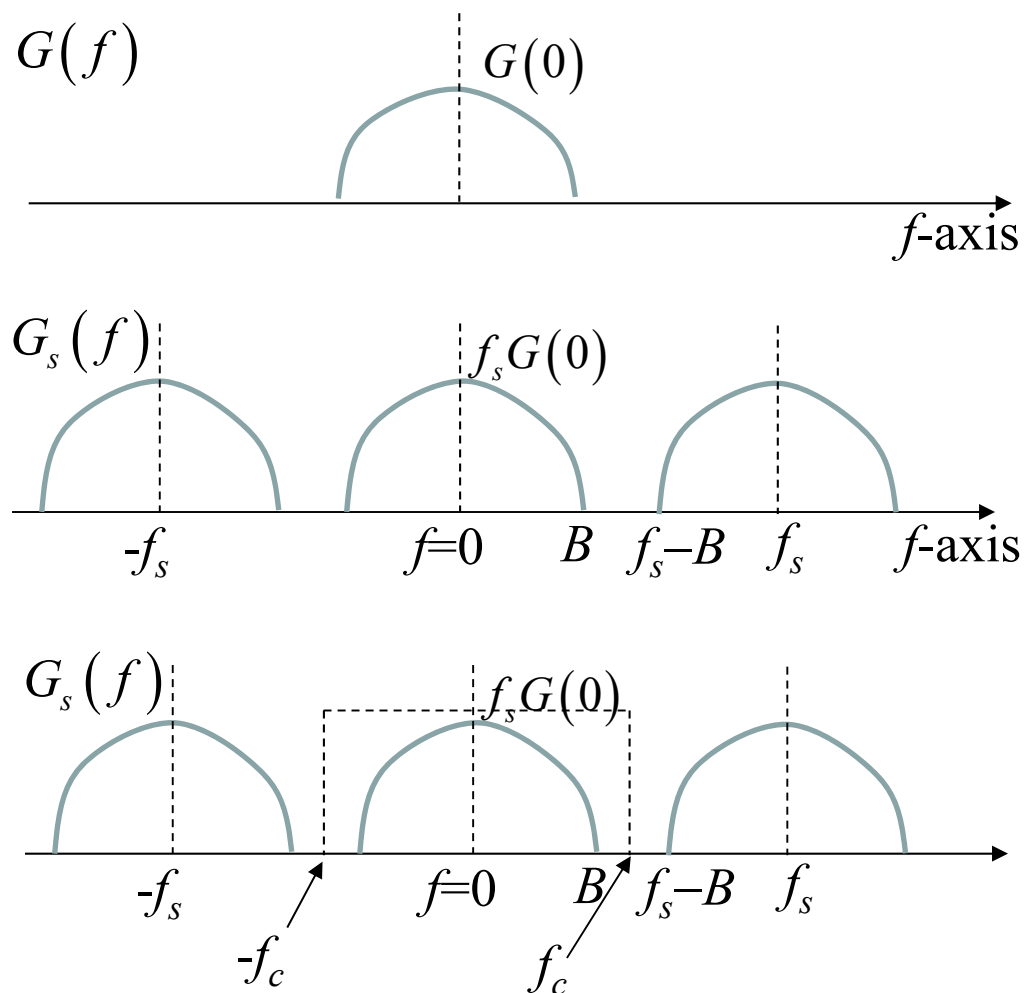
Q: What will happen if  $f_s = 2B$ ?

Even component with frequency  $f = \pm B \rightarrow$  preserved

Odd component with frequency  $f = \pm B \rightarrow$  destroyed

### 5.1.3 Reconstruction (Digital to Analogous)

When the Nyquist criterion is satisfied, one can apply the lowpass filter to reconstruct the original signal.



Frequency Domain

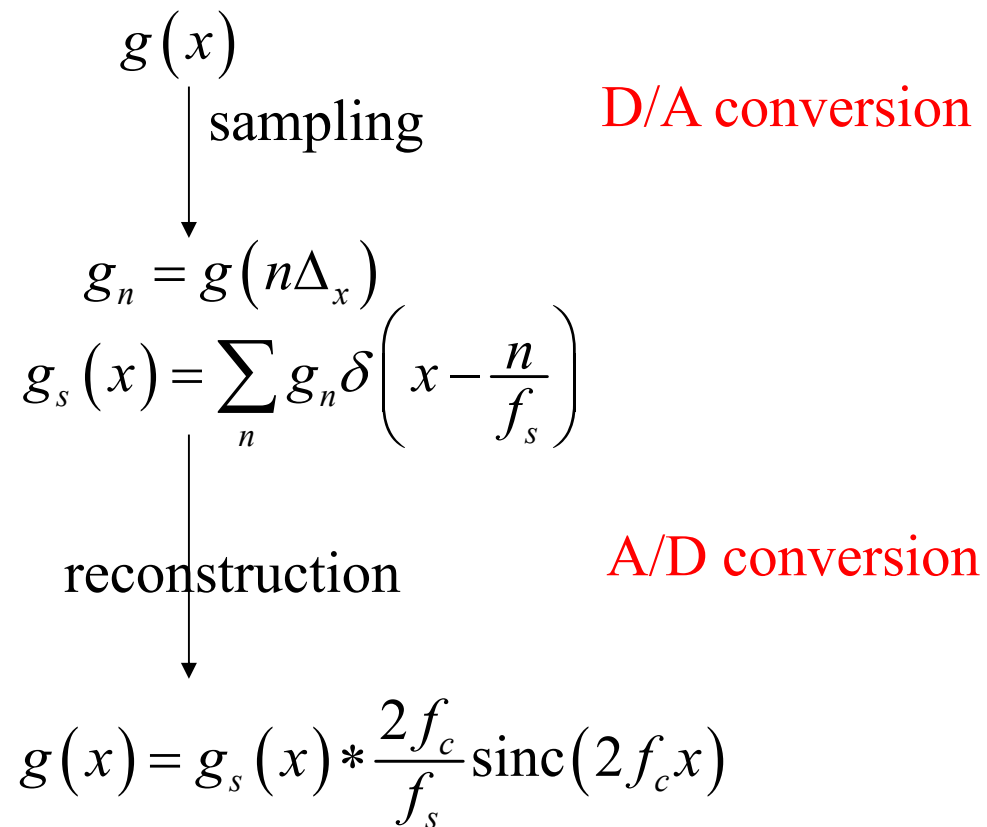
$$G(f)$$

$$G_s(f) = f_s \sum_n G(f - nf_s)$$

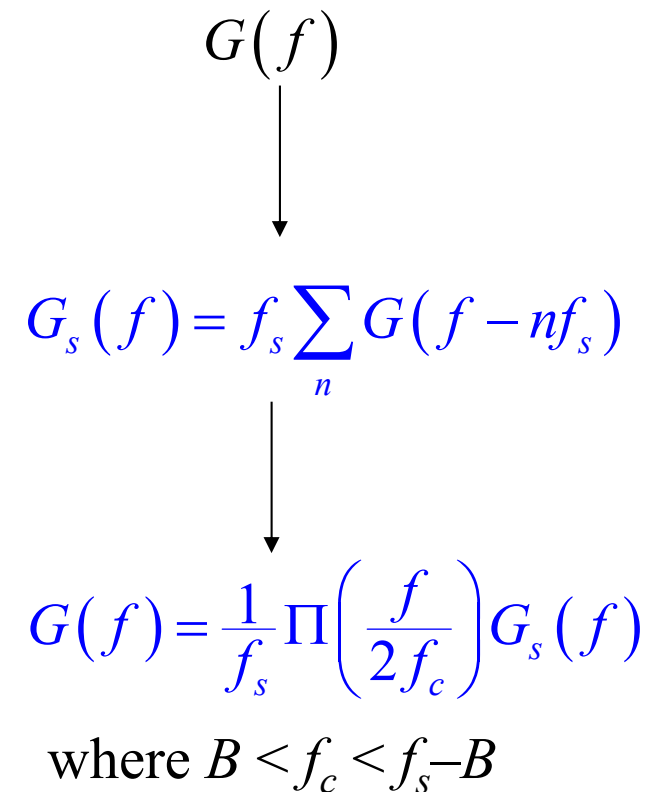
$$G(f) = \frac{1}{f_s} \Pi\left(\frac{f}{2f_c}\right) G_s(f)$$

where  $B < f_c < f_s - B$

## Time Domain



## Frequency Domain



$$\begin{aligned}
 g(x) &= \frac{2f_c}{f_s} \int g_s(\tau) \operatorname{sinc}(2f_c(x - \tau)) d\tau \\
 &= \frac{2f_c}{f_s} \int \sum_n g_n \delta\left(\tau - \frac{n}{f_s}\right) \operatorname{sinc}(2f_c(x - \tau)) d\tau
 \end{aligned}$$

$$g(x) = \frac{2f_c}{f_s} \sum_n g_n \operatorname{sinc}\left(2f_c\left(x - \frac{n}{f_s}\right)\right)$$

Specially, when  $f_c = f_s / 2$

$$g(x) = \sum_n g_n \operatorname{sinc}(f_s x - n)$$

Signal Reconstruction Formula:

$$g(x) = \sum_n g_n \operatorname{sinc}\left(\frac{x}{\Delta_x} - n\right)$$

where  $g_n = g(n\Delta_x)$

constraint:  $\frac{1}{\Delta_x} > 2B$

**[Example 1]** Suppose that

$$g_n = g\left(\frac{n}{2}\right)$$

$$g_{-1} = g_1 = 1, \quad g_0 = 2, \quad g_n = 0 \quad \text{otherwise}$$

$$G(f) = 0 \quad \text{for } f \geq 1$$

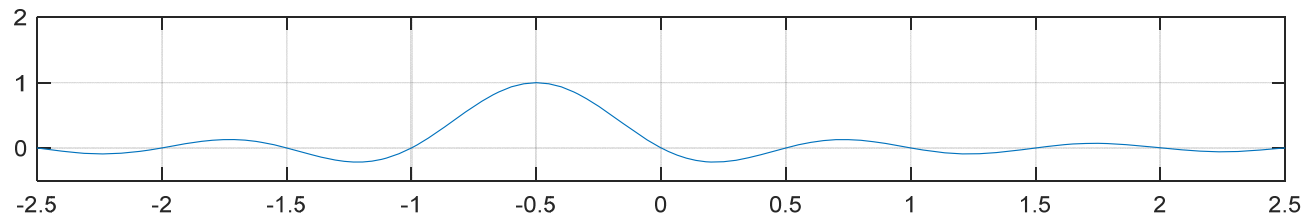
Try to reconstruct  $g(x)$ .

(Solution):  $\Delta_x = 1/2$

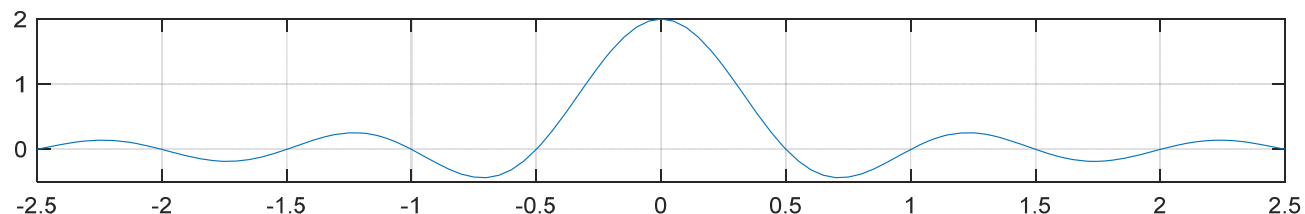
$$g(x) = \text{sinc}(2x+1) + 2\text{sinc } 2x + \text{sinc}(2x-1)$$

$$g(x) = \text{sinc}(2x+1) + 2\text{sinc}2x + \text{sinc}(2x-1)$$

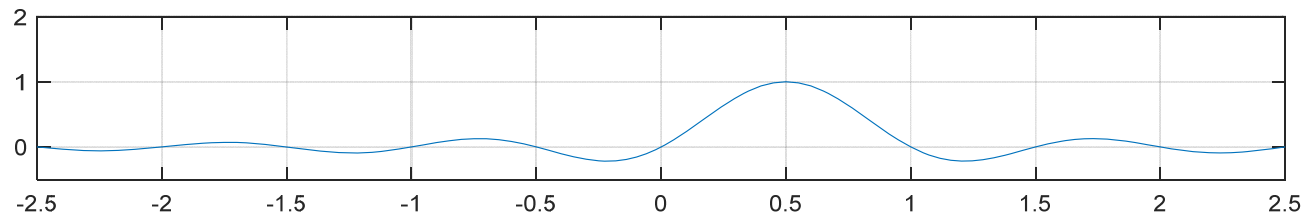
$\text{sinc}(2x+1)$



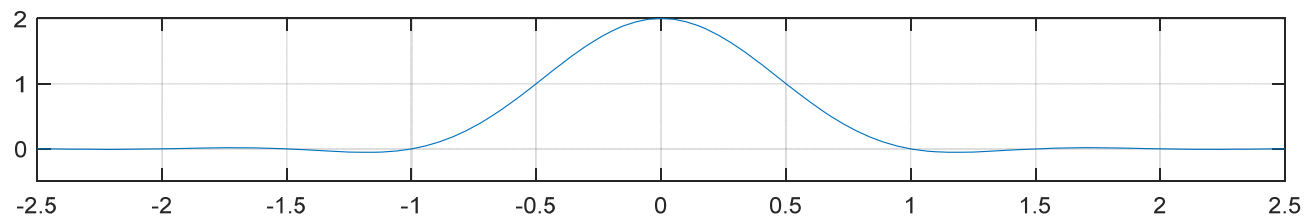
$2\text{sinc}(2x)$



$\text{sinc}(2x-1)$



$g(x)$



(Note):  $\text{sinc}(2x+1)$ ,  $2\text{sinc}(2x)$ ,  $\text{sinc}(2x-1)$   
do not interfere with one another at  $x = n/2$ .

## 5.1.4 Varying the Sampling Rate

(i) D/A conversion

$$g(x) = \sum_n g_n \operatorname{sinc}\left(\frac{x}{\Delta_x} - n\right)$$

(ii) Re-sampling

$$\hat{g}_n = g(n\Delta_{new}) = \sum_m g_m \operatorname{sinc}\left(\frac{n\Delta_{new}}{\Delta_x} - m\right)$$

Note: When  $\Delta_{new} = k\Delta_x$  and  $k$  is an integer

$$\hat{g}_n = g_{kn}$$

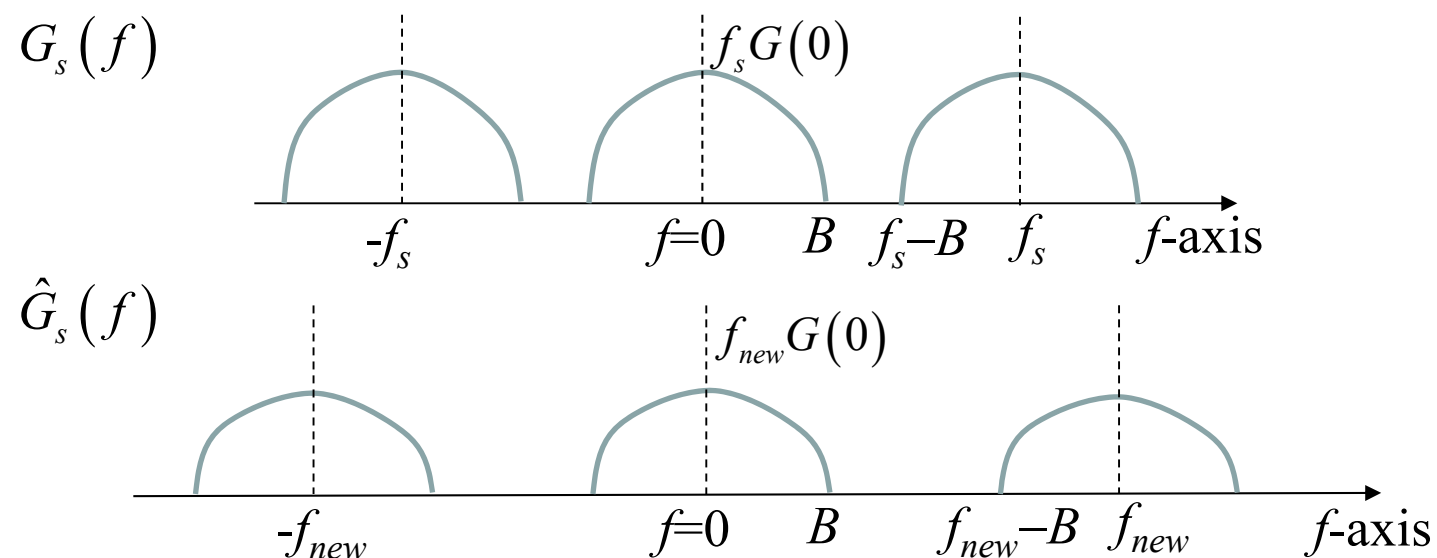
From the view point of the spectrum, if

$$g_s(x) = \sum_n g_n \delta(x - n\Delta_x) \quad \hat{g}_s(x) = \sum_n \hat{g}_n \delta(x - n\Delta_{new})$$

then

$$G_s(f) = f_s \sum_n G(f - nf_s) \quad \hat{G}_s(f) = f_{new} \sum_n G(f - nf_{new})$$

where  $f_s = 1/\Delta_x$ ,  $f_{new} = 1/\Delta_{new}$



## 5.2 Discrete Fourier Transform

### 5.2.1 Derivation and Definitions of the Discrete Fourier Transform

To process discrete functions, the continuous Fourier transform should be converted into the discrete version.

Continuous Fourier transform:

$$G(f) = \int_{-\infty}^{\infty} g(x) e^{-j2\pi fx} dx$$

If we set

$$f = m\Delta_f, \quad x = n\Delta_x$$

then

$$G(m\Delta_f) = \sum_n g(n\Delta_x) e^{-j2\pi mn\Delta_f\Delta_x} \Delta_x$$

$$G(m\Delta_f) = \sum_n g(n\Delta_x) e^{-j2\pi mn\Delta_f\Delta_x} \Delta_x$$

Specially, if

$$\Delta_f\Delta_x = 1/N$$

then

$$G(m\Delta_f) = \sum_n g(n\Delta_x) e^{-j\frac{2\pi mn}{N}} \Delta_x$$

Similarly, for the continuous inverse Fourier transform:

$$g(x) = \int_{-\infty}^{\infty} G(f) e^{j2\pi fx} df$$

$$g(n\Delta_x) = \sum_m G(m\Delta_f) e^{j2\pi mn\Delta_f\Delta_x} \Delta_f$$

$$g(n\Delta_x) = \sum_m G(m\Delta_f) e^{j\frac{2\pi mn}{N}} \Delta_f$$

## Discrete Fourier Transform (DFT)

$$G[m] = DFT \{g[n]\} = \sum_{n=0}^{N-1} g[n] e^{-j\frac{2\pi mn}{N}}$$

$m, n = 0, 1, 2, \dots, N-1$

## Inverse Discrete Fourier Transform (IDFT)

$$g[n] = IDFT \{G[m]\} = \frac{1}{N} \sum_{m=0}^{N-1} G[m] e^{j\frac{2\pi mn}{N}}$$

Note that the output of  $g[n]$  is periodic

$$G[m] = G[m + N]$$

Also note that, on page 440,

$$G(m\Delta_f) = DFT(g(n\Delta_x))\Delta_x$$

The DFT and the IDFT form a transform pair since

$$\frac{1}{N} \sum_{m=0}^{N-1} G[m] e^{j\frac{2\pi mn}{N}} = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{k=0}^{N-1} g[k] e^{-j\frac{2\pi mk}{N}} e^{j\frac{2\pi mn}{N}} = \frac{1}{N} \sum_{k=0}^{N-1} g[k] \left( \sum_{m=0}^{N-1} e^{-j\frac{2\pi mk}{N}} e^{j\frac{2\pi mn}{N}} \right)$$

Because

$$\sum_{m=0}^{N-1} e^{j\frac{2\pi a}{N}m} = \frac{1 - e^{j\frac{2\pi a}{N}N}}{1 - e^{j\frac{2\pi a}{N}}} = \frac{1 - e^{j2\pi a}}{1 - e^{j\frac{2\pi a}{N}}} = 0 \quad \text{if } a \neq 0,$$

$$\sum_{m=0}^{N-1} e^{j\frac{2\pi a}{N}m} = N \quad \text{if } a = 0,$$

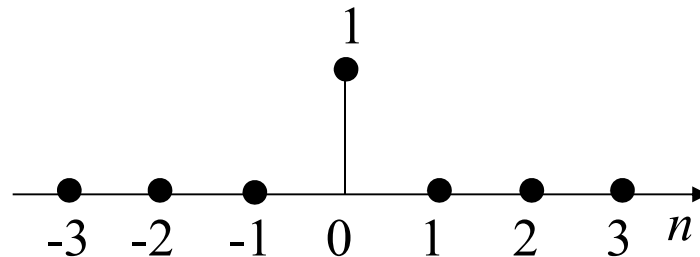
$$\sum_{m=0}^{N-1} e^{j\frac{2\pi m}{N}(n-k)} = N\delta_d[n-k]$$

unit impulse function  
(discrete Dirac delta function)

$$\frac{1}{N} \sum_{m=0}^{N-1} G[m] e^{j\frac{2\pi mn}{N}} = \frac{1}{N} \sum_{k=0}^{N-1} g[k] N\delta_d[n-k] = g[n]$$

## Unit Impulse Function (discrete Dirac delta function)

$$\delta_d[n] = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{if } n \neq 0 \end{cases}$$



The unit impulse function has an explicit form. It does not a limitation of a distribution.

Compared to page 343,

$$\sum_n \delta_d[n] = 1$$
$$\delta_d[n] = 0 \quad \text{if } n \neq 0$$
$$\delta_d[n] = \delta_d[-n]$$

## Other possible definitions of the DFT

$$\text{DFT} \quad G[m] = \sum_{n=n_0}^{n_0+N-1} g[n] e^{-j\frac{2\pi mn}{N}}$$

$$\text{IDFT} \quad g[n] = \frac{1}{N} \sum_{m=m_0}^{m_0+N-1} G[m] e^{j\frac{2\pi mn}{N}}$$

$$\text{DFT} \quad G[m] = \sqrt{\frac{1}{N}} \sum_{n=0}^{N-1} g[n] e^{-j\frac{2\pi mn}{N}}$$

$$\text{IDFT} \quad g[n] = \sqrt{\frac{1}{N}} \sum_{m=0}^{N-1} G[m] e^{j\frac{2\pi mn}{N}}$$

## 5.2.2 Implementing Continuous FT by the DFT

Suppose that we want to calculate the continuous FT of  $g(x)$  digitally and

$$g(x) = 0 \quad \text{for } x \notin [x_1, x_1 + T]$$

(i) Shifting

$$g_1(x) = g(x + x_1)$$

Note:

$$g_1(x) = 0 \quad \text{for } x \notin [0, T]$$

(ii) Sampling

$$g_d[n] = g_1(n\Delta_x)$$

(iii) DFT

$$G_d[m] = \sum_{n=0}^{N-1} g_d[n] e^{-j\frac{2\pi mn}{N}}$$

(iv) Mapping to the true frequency

$$G_1(m\Delta_f) = G_d[m]\Delta_x \quad (\text{from pages 440 and 441})$$

Since

$$\Delta_f\Delta_x = 1/N \quad \Delta_f = \frac{1}{N\Delta_x} = \frac{f_s}{N}$$

Therefore,

$$G_1\left(m\frac{f_s}{N}\right) = G_d[m]\Delta_x \quad \text{if } 0 \leq m \leq N/2,$$

$$G_1\left(m\frac{f_s}{N} - f_s\right) = G_d[m]\Delta_x \quad \text{if } N/2 \leq m \leq N-1$$

(v) Using the modulation property

$$G(f) = e^{-j2\pi x_1 f} G_1(f)$$

**[Example 1]** : Suppose that

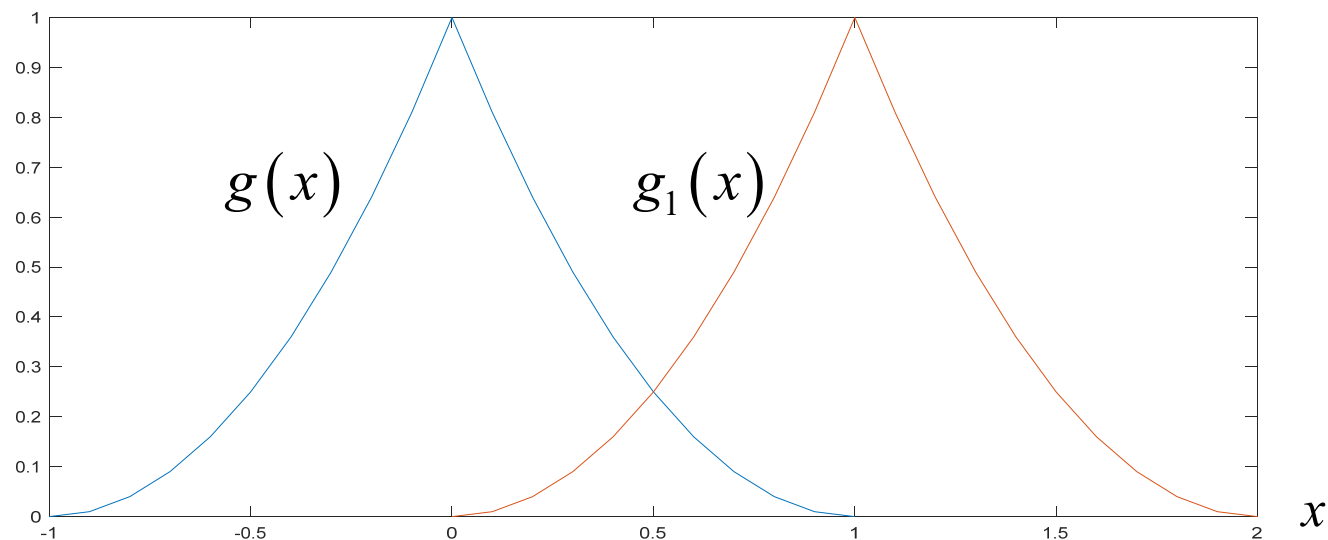
$$g(x) = (1-|x|)^2 \quad \text{for } -1 \leq x \leq 1 \quad g(x) = 0 \quad \text{otherwise}$$

Sampling interval :  $\Delta_x = 0.1$

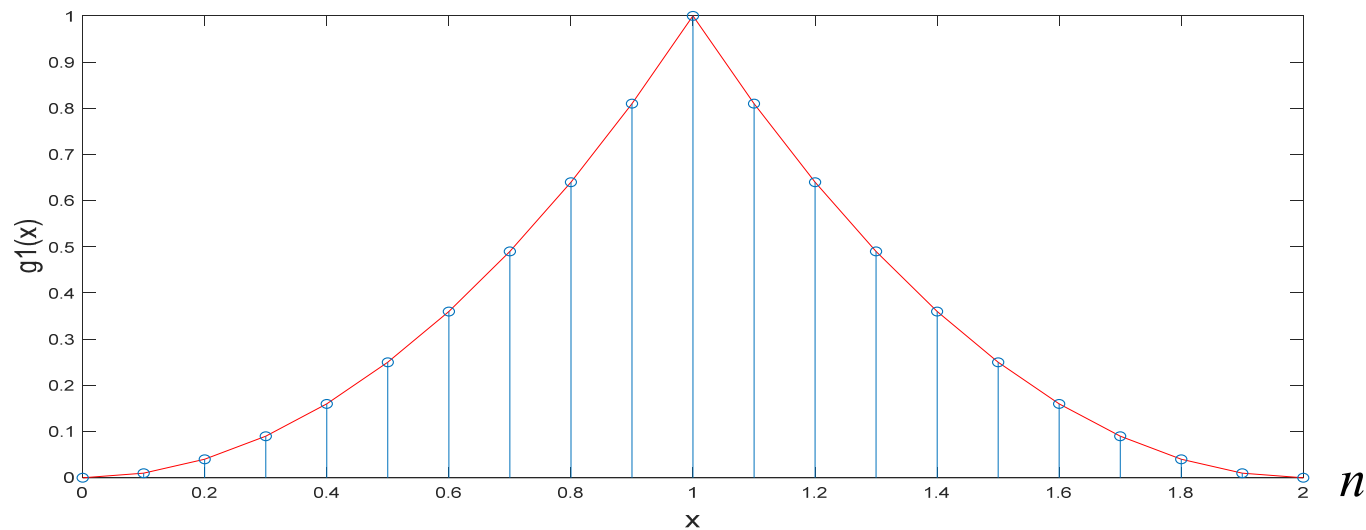
How do we obtain the FT of  $g(x)$  by the DFT?

(Solution):

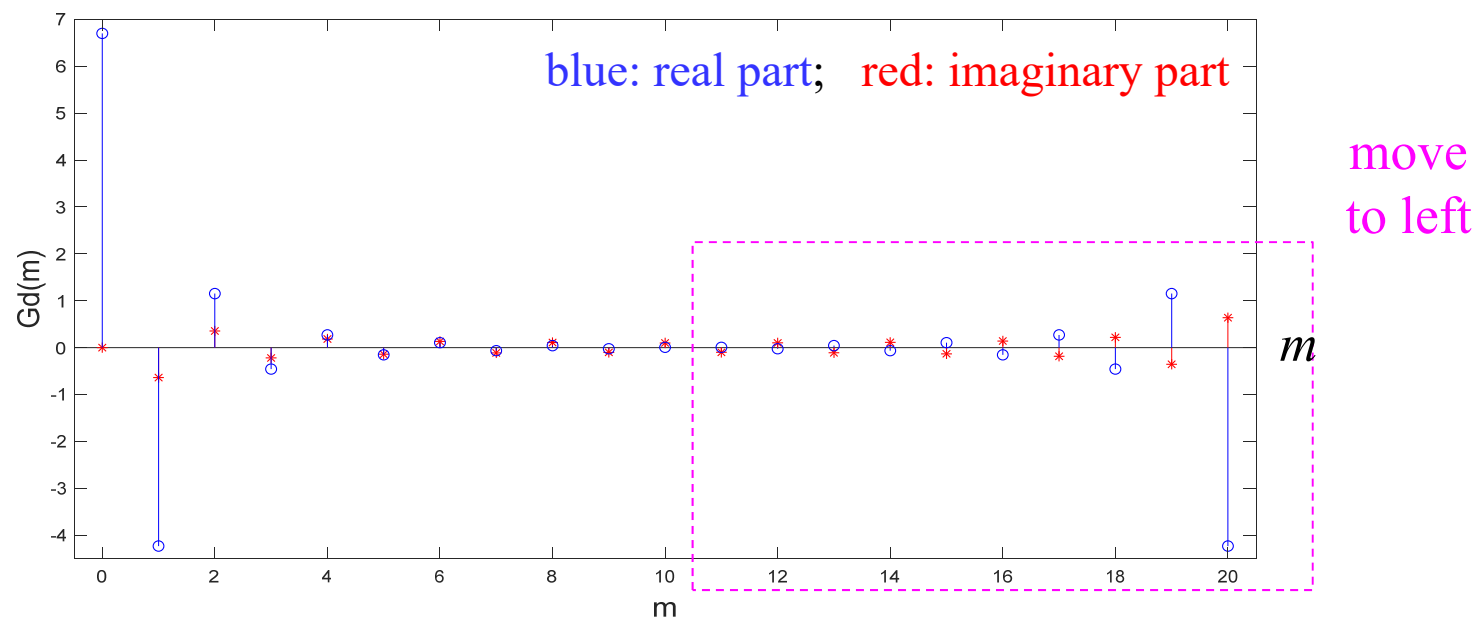
(i)  $g_1(x) = g(x-1)$



$$(ii) \quad g_d[n] = g_1(n\Delta_x)$$

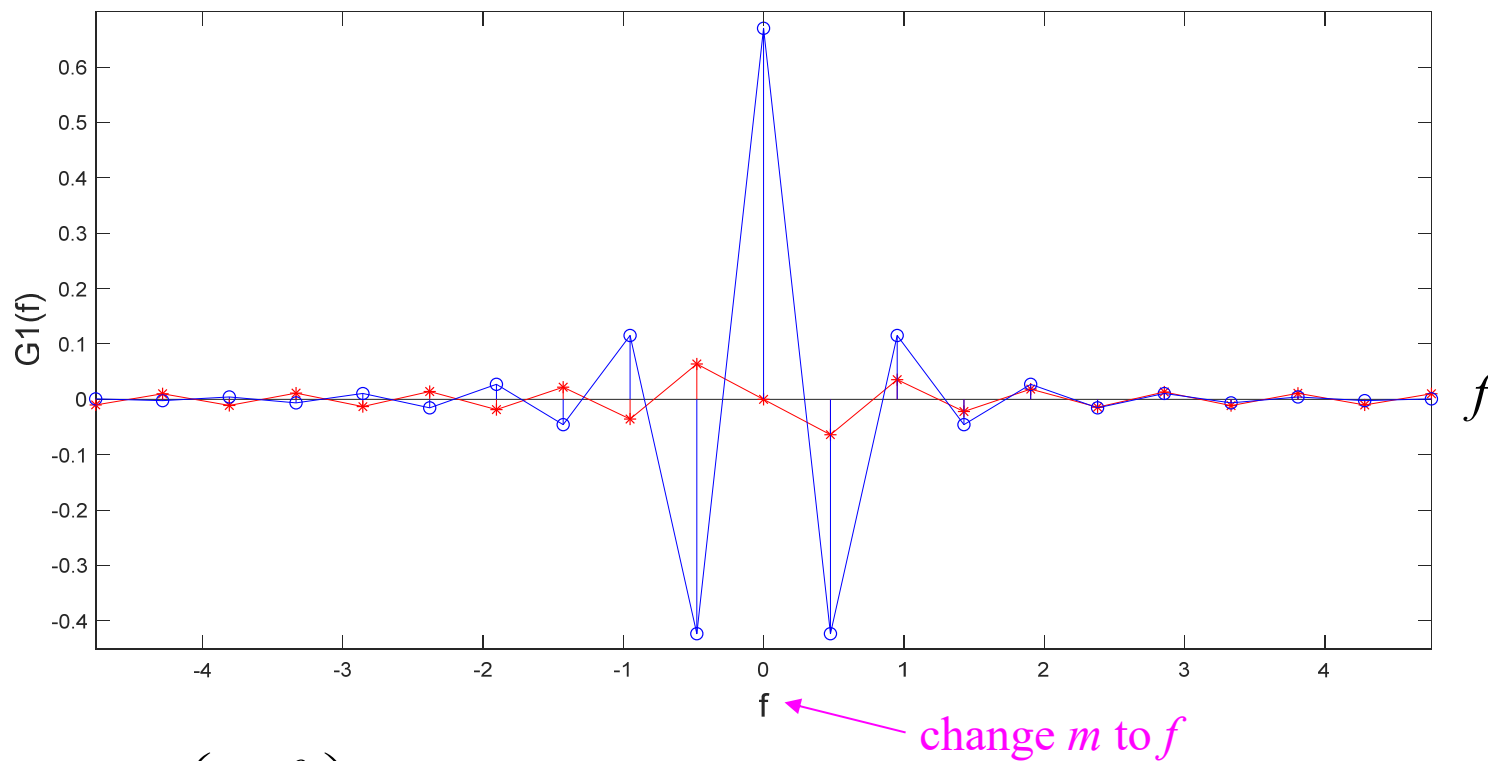


$$(iii) \quad G_d[m] = DFT(g_d[n])$$



## (iv) Mapping

blue: real part; red: imaginary part

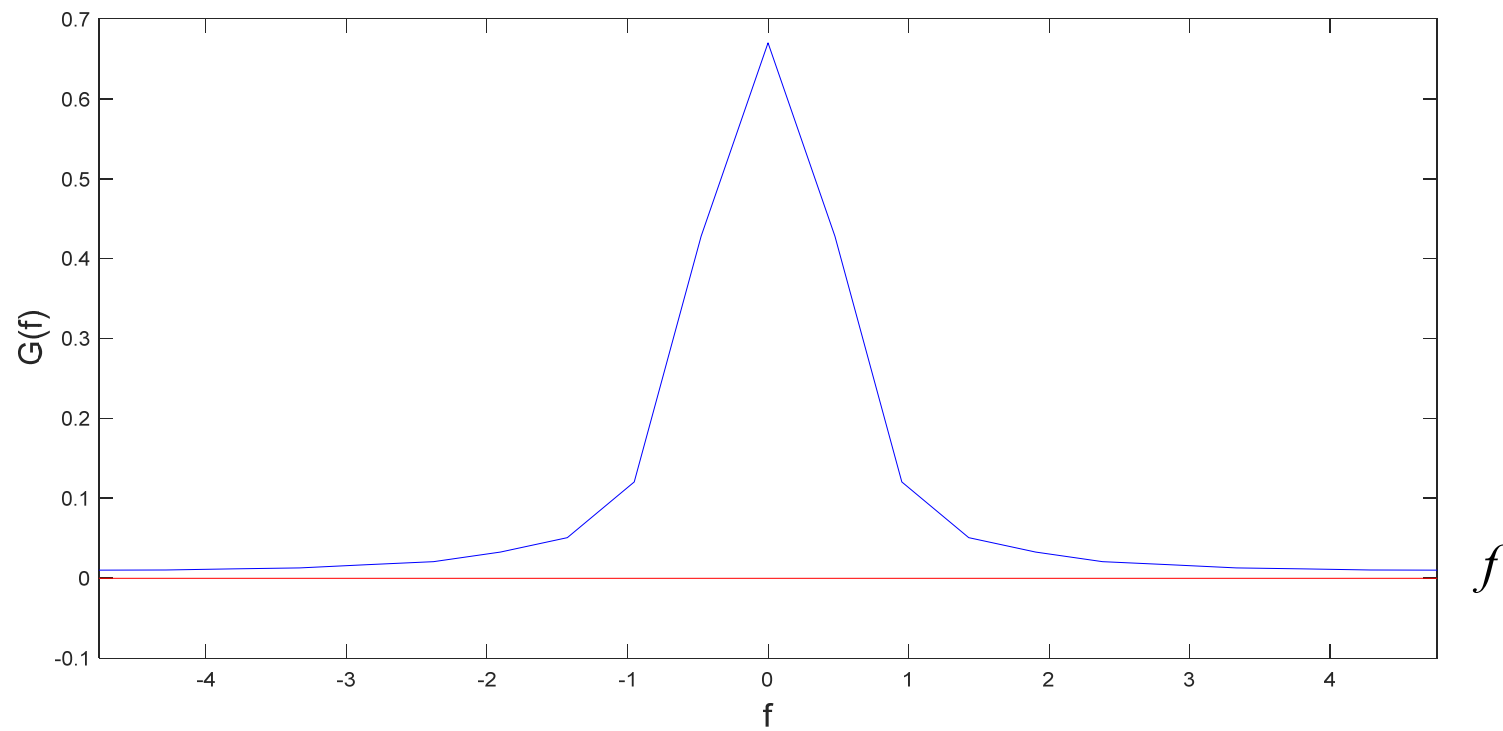


$$G_1\left(m \frac{f_s}{N}\right) = G_d[m] \Delta_x \quad \text{if } 0 \leq m \leq N/2,$$

$$G_1\left(m \frac{f_s}{N} - f_s\right) = G_d[m] \Delta_x \quad \text{if } N/2 \leq m \leq N-1$$

$$(v) \quad G(f) = e^{j2\pi f} G_1(f)$$

blue: real part; red: imaginary part



## 5.2.3 Transform Pairs and Properties

### [Duality Property]

If  $G[m] = DFT\{g[n]\}$

then  $g[(-m)_N] = \frac{1}{N} DFT\{G[n]\}$      $DFT\{G[n]\} = Ng[(-m)_N]$

$((a))_N$ :  $a$  modulo  $N$

the remainder of  $a$  after divided by  $N$

$$g[(-m)_N] = \begin{cases} g[N-m] & \text{if } m = 1, 2, \dots, N-1 \\ g[0] & \text{if } m = 0 \end{cases}$$

If  $G[m] = DFT\{g[n]\}$  then  $g[((-m))_N] = \frac{1}{N} DFT\{G[n]\}$

$$\begin{aligned}
 \text{(Proof): } \frac{1}{N} DFT\{G[n]\} &= \frac{1}{N} \sum_{n=0}^{N-1} e^{-j\frac{2\pi mn}{N}} G[n] \\
 &= \frac{1}{N} \sum_{n=0}^{N-1} e^{-j\frac{2\pi mn}{N}} \sum_{k=0}^{N-1} e^{-j\frac{2\pi nk}{N}} g[k] = \frac{1}{N} \sum_{k=0}^{N-1} \left( \sum_{n=0}^{N-1} e^{-j\frac{2\pi mn}{N}} e^{-j\frac{2\pi nk}{N}} \right) g[k]
 \end{aligned}$$

Since  $\sum_{n=0}^{N-1} e^{j\frac{2\pi n}{N}a} = \delta_d [((a))_N]$

(proved on the next page)

$$\begin{aligned}
 \frac{1}{N} DFT\{G[n]\} &= \frac{1}{N} \sum_{k=0}^{N-1} N \delta_d [((-m-k))_N] g[k] \\
 &= \sum_{k=0}^{N-1} \delta_d [((m+k))_N] g[k] = g[((-m))_N]
 \end{aligned}$$

$$\sum_{n=0}^{N-1} e^{j\frac{2\pi a}{N}n} = N\delta_d [((a))_N] \quad ((a))_N: \text{the remainder of } a \text{ after divided by } N$$

When  $a \neq bN$  where  $b$  is some integer

$$\sum_{n=0}^{N-1} e^{j\frac{2\pi a}{N}n} = \frac{1 - e^{j\frac{2\pi a}{N}N}}{1 - e^{j\frac{2\pi a}{N}}} = \frac{1 - 1}{1 - e^{j\frac{2\pi a}{N}}} = 0$$

When  $a = bN$  where  $b$  is some integer (i.e.,  $((a))_N = 0$ )

$$\sum_{n=0}^{N-1} e^{j\frac{2\pi a}{N}n} = \sum_{n=0}^{N-1} e^{j2\pi bn} = \sum_{n=0}^{N-1} 1 = N$$

**[Determine the IDFT by the DFT]**

$$g_1[m] = DFT\{G[n]\}$$

$$g[n] = \frac{1}{N} g_1[-n]$$

Note:

- (i) Computation loading of the IDFT = Computation loading of the DFT
- (ii) In industry, only the chip of the DFT is required.

**[Transform Pairs]**

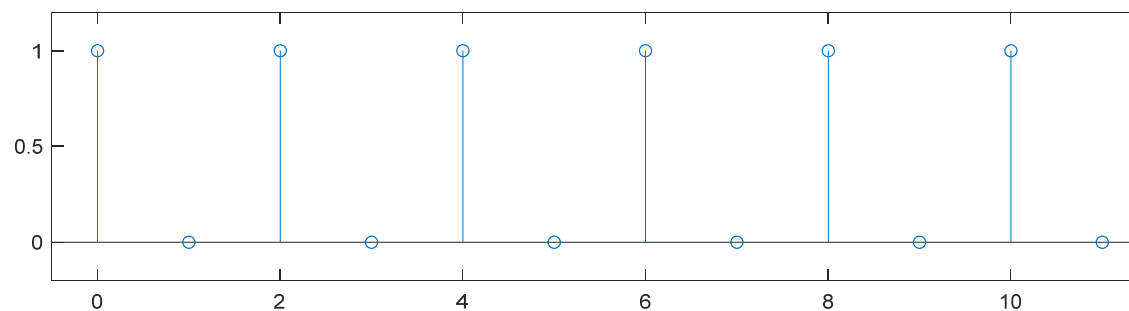
$g[n]$	$G[m]$
(1) $\delta_d[n]$ (see page 443)	1
(2) 1	$N\delta_d[m]$
(3) $\delta_d[n-k]$	$\exp[-j2\pi km/N]$
(4) $\exp[j2\pi kn/N]$	$N\delta_d[m-k]$
(5) $\cos[2\pi kn/N]$	$\frac{N}{2}\delta_d[m-k] + \frac{N}{2}\delta_d[m-(N-k)]$
(6) $\sin[2\pi kn/N]$	$-j\frac{N}{2}\delta_d[m-k] + j\frac{N}{2}\delta_d[m-(N-k)]$
(7) $g[n] = 1$ for $0 \leq n \leq W$ $g[n] = 0$ otherwise	$e^{-j\frac{\pi W}{N}m} \frac{\sin(\pi m(W+1)/N)}{\sin(\pi m/N)} \quad \text{for } m \neq 0,$ $W+1 \quad \text{for } m = 0$
(8) $\exp[-kn]$ , $k \neq 0$	$\frac{1 - e^{-Nk}}{1 - e^{-k-j2\pi m/N}}$

**[Discrete Impulse Train]**

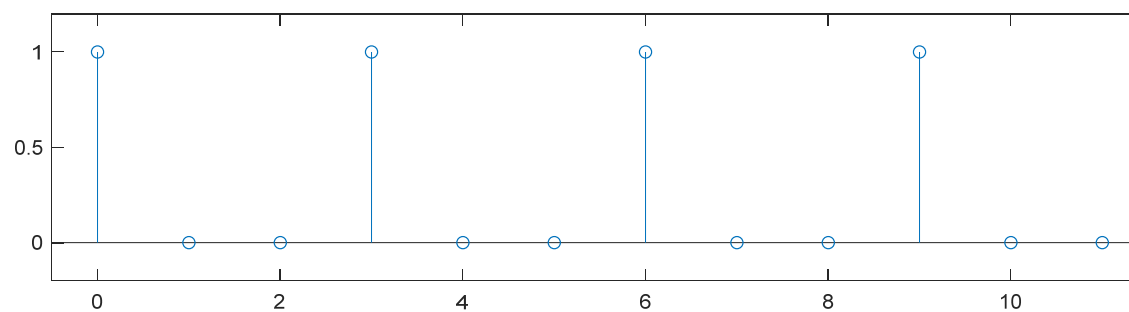
$$p_c[n] = \begin{cases} 1 & \text{if } n \text{ is a multiple of } c \\ 0 & \text{otherwise} \end{cases} \quad c \text{ is a factor of } N$$

$$N = 12$$

$$p_2[n]$$



$$p_3[n]$$



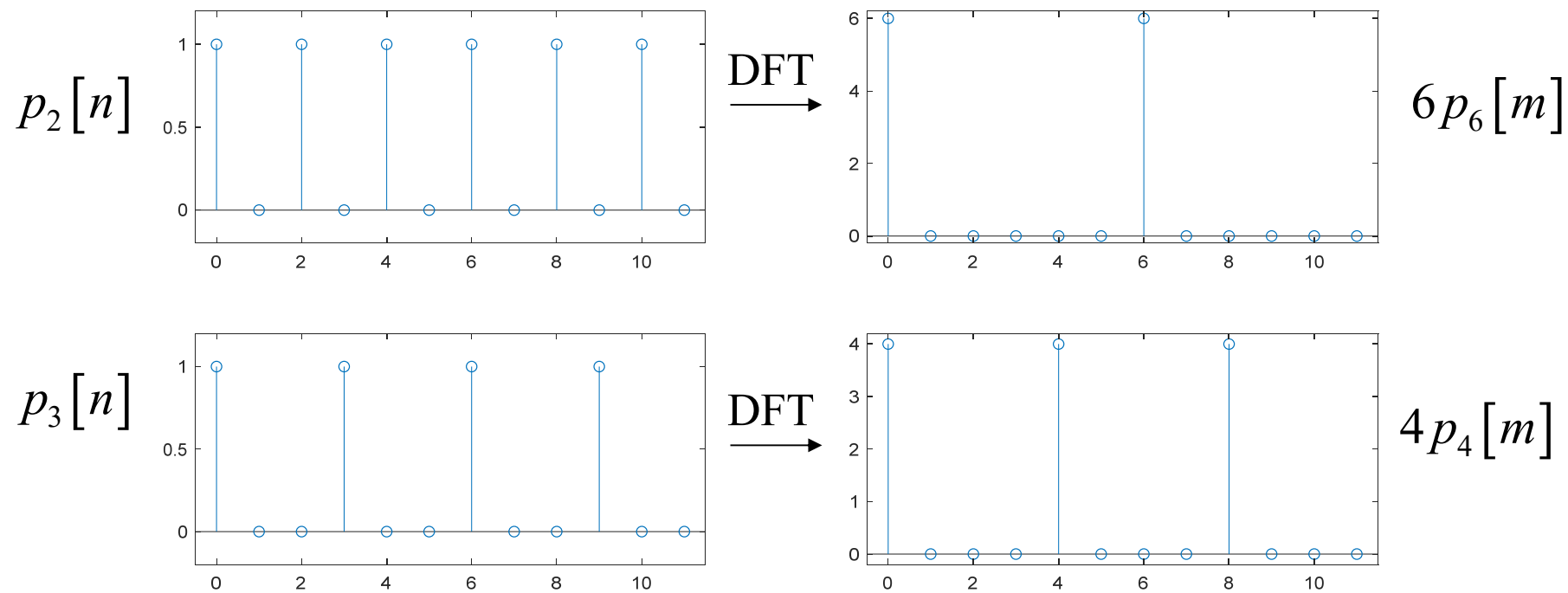
**[Example 2]** Determine the DFT of  $p_c[n]$

$$\sum_{n=0}^{N-1} e^{-j\frac{2\pi m}{N}n} p_c[n] = \sum_{k=0}^{N/c-1} e^{-j\frac{2\pi m}{N}ck} = \sum_{k=0}^{N/c-1} e^{-j\frac{2\pi m}{N/c}k} \quad \begin{array}{l} c \text{ is a factor of } N \\ n = ck \end{array}$$

$$\sum_{k=0}^{N/c-1} e^{-j\frac{2\pi m}{N/c}k} = \frac{1 - e^{-j\frac{2\pi m}{N/c}c}}{1 - e^{-j\frac{2\pi m}{N/c}}} = 0 \quad \text{if } m \text{ is not a multiple of } N/c$$

$$\sum_{k=0}^{N/c-1} e^{-j\frac{2\pi m}{N/c}k} = \frac{N}{c} \quad \text{if } m \text{ is a multiple of } N/c$$

Therefore,  $DFT \{p_c[n]\} = \frac{N}{c} p_{N/c}[m]$



## Properties

(1) Linear	$DFT\{ax[n] + by[n]\} = aX[m] + bY[m]$
(2) DC Values	$G[0] = \sum_{n=0}^{N-1} g[n], \quad g[0] = \frac{1}{N} \sum_{n=0}^{N-1} G[m]$
(3) Shifting	$DFT\{g[((n-k))_N]\} = W^{km} G[m]$ <p>where <math>((n))_N = n</math> if <math>0 \leq n \leq N-1</math>  <math>((n))_N = n+N</math> if <math>-N \leq n \leq -1</math>  <math>((n))_N = n-N</math> if <math>N \leq n \leq 2N-1</math></p> $W = \exp(-j2\pi / N)$
(4) Modulation	$DFT\{W^{kn} g[n]\} = G[((m+k))_N]$
(5) Time Reverse	$DFT\{g[(-n))_N]\} = G[(-m))_N]$
(6) Even /Odd Input	<p>If <math>g[n] = g[(-n))_N]</math>, then <math>G[m] = G[(-m))_N]</math>;          If <math>g[n] = -g[(-n))_N]</math>, then <math>G[m] = -G[(-m))_N]</math>;</p>

(7) Conjugate	$DFT\{g^*[n]\} = G^*[((-m))_N]$
(8) Real/Imaginary Input	If $g[n]$ is real, then $G[m] = G^*[((-m))_N]$ ; If $g[n]$ is pure imaginary, then $G[m] = -G^*[((-m))_N]$ ;
(9) Circular Convolution	If $y[n] = g[n] *_c h[n] = \sum_{k=0}^{N-1} g[k] h[((n-k))_N]$ then $Y[m] = G[m] H[m]$
(10) Circular Correlation	If $y[n] = g[n] *_c h^* [(((-n))_N)]$ $= \sum_{k=0}^{N-1} g[((k+n))_N] h^* [k]$ then $Y[m] = G[m] H^* [m]$
(11) Parseval's Theorem (Energy Preservation)	$N \sum_{n=0}^{N-1}  g[n] ^2 = \sum_{m=0}^{N-1}  G[m] ^2$
(12) Generalized Parseval's Theorem	$N \sum_{n=0}^{N-1} g[n] h^* [n] = \sum_{m=0}^{N-1} G[m] H^* [m]$

## 5.2.4 Discrete Circular Convolution

461

**[Discrete Circular Convolution]**  $g[n] *_c h[n] = \sum_{k=0}^{N-1} g[k] h[((n-k))_N]$

(Proof of the convolution property)

$$\begin{aligned} \frac{1}{N} \sum_{m=0}^{N-1} G[m] H[m] e^{j \frac{2\pi m}{N} n} &= \frac{1}{N} \sum_{m=0}^{N-1} \sum_{k=0}^{N-1} g[k] e^{-j \frac{2\pi m}{N} k} \sum_{s=0}^{N-1} h[s] e^{-j \frac{2\pi m}{N} s} e^{j \frac{2\pi m}{N} n} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{s=0}^{N-1} g[k] h[s] \sum_{m=0}^{N-1} e^{j \frac{2\pi (n-s-k)}{N} m} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{s=0}^{N-1} g[k] h[s] N \delta_d [((n-s-k))_N] \\ &= \sum_{k=0}^{N-1} g[k] h[((n-k))_N] \end{aligned}$$

Here we apply  $\sum_{n=0}^{N-1} e^{j \frac{2\pi a}{N} n} = N \delta_d [((a))_N]$   $((a))_N$ : the remainder of  $a$  after divided by  $N$

## [Discrete Circular Convolution and Discrete Linear Convolution]

A discrete linear time-invariant (LTI) system can always be expressed a **discrete linear convolution**:

$$y_1[n] = g[n] * h[n] = \sum_{k=0}^{N-1} g[k] h[n-k]$$

However, the convolution implemented by the DFT is the **discrete circular convolution**:

If

$$y[n] = IDFT(DFT\{g[n]\} DFT\{h[n]\}) = IDFT(G[m]H[m])$$

then

$$y[n] = g[n] *_c h[n] = \sum_{k=0}^{N-1} g[k] h[((n-k))_N]$$

$((a))_N$ : the remainder of  $a$   
after divided by  $N$

linear convolution:  $y_1[n] = g[n] * h[n] = \sum_{k=0}^{N-1} g[k]h[n-k]$

circular convolution:  $y[n] = g[n] *_c h[n] = \sum_{k=0}^{N-1} g[k]h[((n-k))_N]$

For example,

$$y_1[2] = g[0]h[2] + g[1]h[1] + g[2]h[0] + g[3]h[-1] + g[4]h[-2] + \dots$$

$$y[2] = g[0]h[2] + g[1]h[1] + g[2]h[0] + g[3]h[N-1] + g[4]h[N-2] \\ + \dots + g[N-1]h[3]$$

The condition where the circular convolution is equal to the linear convolution:

- (i)  $g[n] = 0$  for  $n < 0$  or  $n \geq M$
- (ii)  $h[n] = 0$  for  $n < 0$  or  $n \geq L$
- (iii)  $N \geq M + L - 1$

The condition where the circular convolution is equal to the linear convolution:

$$(i) \quad g[n] = 0 \quad \text{for } n < 0 \text{ or } n \geq M$$

$$(ii) \quad h[n] = 0 \quad \text{for } n < 0 \text{ or } n \geq L$$

$$(iii) \quad N \geq M + L - 1$$

$$\text{(Proof): } y[n] = \sum_{k=0}^{N-1} g[k]h[((n-k))_N]$$

$$\begin{aligned} y[n] &= g[0]h[n] + g[1]h[n-1] + \cdots + g[n]h[0] + g[n+1]h[N-1] + \\ &\quad g[n+2]h[N-2] + \cdots + g[N+n+1-L]h[L-1] + \cdots + g[N-1]h[n+1] \\ &= g[0]h[n] + g[1]h[n-1] + \cdots + g[n]h[0] \\ &\quad + g[N+n+1-L]h[L-1] + \cdots + g[N-1]h[n+1] \\ &= g[0]h[n] + g[1]h[n-1] + \cdots + g[n]h[0] = y_1[n] \end{aligned}$$

(Since  $N+n+1-L \geq N+1-L \geq M$ )

## 5.2.5 Complexity

$$G[m] = \sum_{n=0}^{N-1} g[n] e^{-j\frac{2\pi mn}{N}}$$

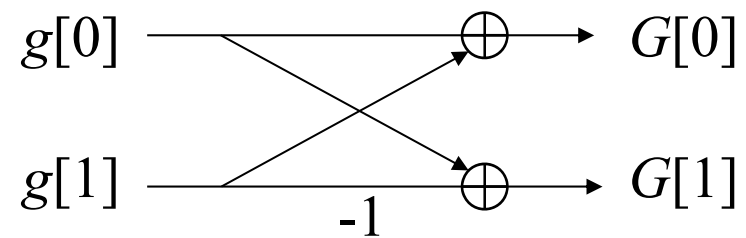
Direct implementation: Complexity =  $O(N^2)$

With the fast algorithm: Complexity =  $O(N \log_2 N)$

$$G[m] = \sum_{n=0}^{N-1} g[n] e^{-j\frac{2\pi mn}{N}}$$

2-point DFT

$$\begin{bmatrix} G[0] \\ G[1] \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} g[0] \\ g[1] \end{bmatrix}$$



When  $N = 2^k$

$$\begin{aligned}
 G[m] &= \sum_{n=0}^{N-1} g[n] e^{-j \frac{2\pi mn}{N}} \\
 &= \sum_{n=0}^{N/2-1} g[2n] e^{-j \frac{2\pi m(2n)}{N}} + \sum_{n=0}^{N/2-1} g[2n+1] e^{-j \frac{2\pi m(2n+1)}{N}} \\
 &= \sum_{n=0}^{N/2-1} g_1[n] e^{-j \frac{2\pi mn}{N/2}} + e^{-j \frac{2\pi m}{N}} \sum_{n=0}^{N/2-1} g_2[n] e^{-j \frac{2\pi mn}{N/2}}
 \end{aligned}$$

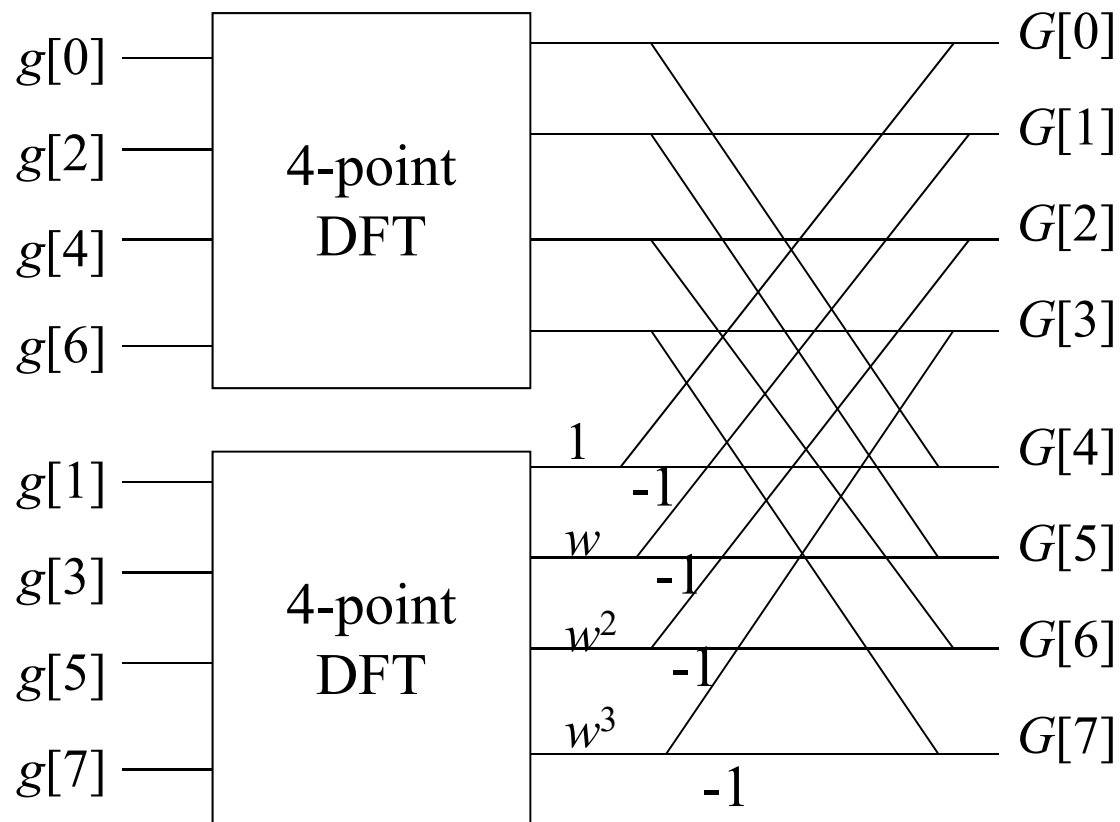
twiddle factors

$$g_1[n] = g[2n], \quad g_2[n] = g[2n+1]$$

Therefore,

one  $N$ -point DFT = two  $(N/2)$ -point DFT + twiddle factors

## 8-point DFT



$$w = e^{-j\frac{2\pi}{N}}$$

$$w^{\left(m+\frac{N}{2}\right)} = e^{-j\frac{2\pi}{N}\left(m+\frac{N}{2}\right)} = e^{-j\frac{2\pi}{N}m} e^{-j\frac{2\pi}{N}\frac{N}{2}} = -e^{-j\frac{2\pi}{N}m} = -w^m$$

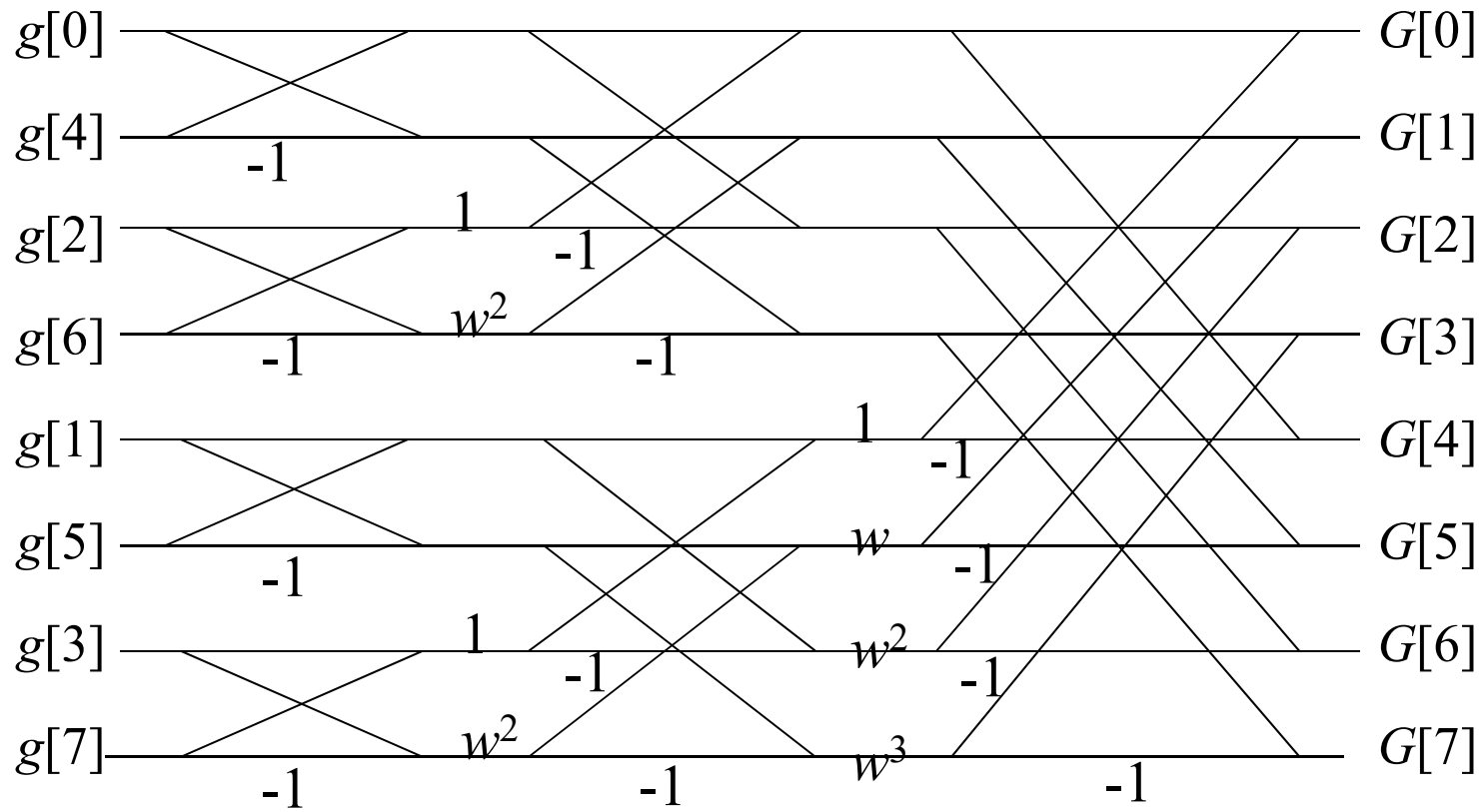
When  $N = 8$

$$w = e^{-j\frac{2\pi}{8}}$$

$$w^4 = -1$$

$$w^8 = 1$$

## 8-point DFT



$$w = e^{-j\frac{2\pi}{8}}$$

complexity:

$$N \log_2 N$$

$N/2$  twiddle factors are required between two stages

Number of stages

- J. W. Cooley and J. W. Tukey, "An algorithm for the machine computation of complex Fourier series," *Mathematics of Computation*, vol. 19, pp. 297-301, Apr. 1965. (Cooley-Tukey)
- C. S. Burrus, "Index Mappings for multidimensional formulation of the DFT and convolution," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 25, pp. 1239-242, June 1977. (Prime factor)

## 5.2.6 2D DFTs

2-D Discrete Fourier Transform (2-D DFT)

$$G[p, q] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} g[m, n] e^{-j \frac{2\pi m p}{M}} e^{-j \frac{2\pi n q}{N}}$$

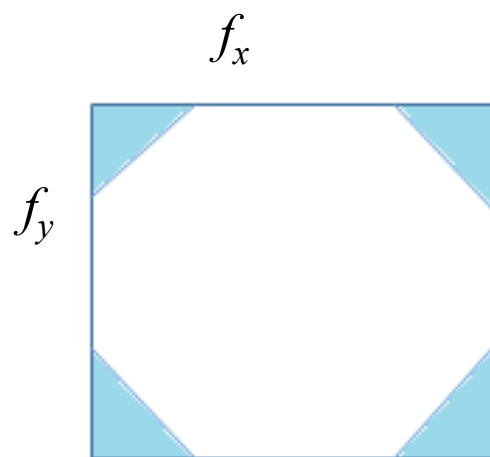
2-D Inverse Discrete Fourier Transform (2-D IDFT)

$$g[m, n] = \frac{1}{MN} \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} G[p, q] e^{j \frac{2\pi m p}{M}} e^{j \frac{2\pi n q}{N}}$$

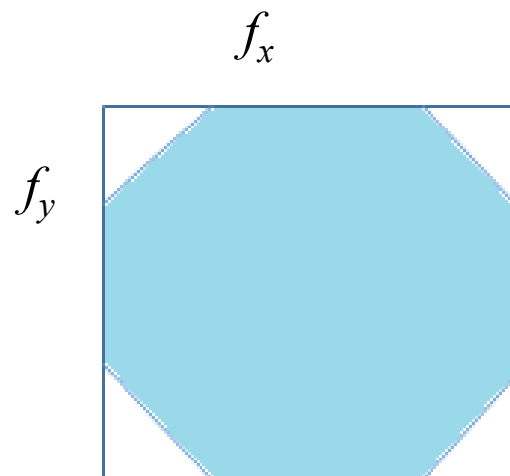
Low Frequency Part  $\rightarrow$  Mild Variation  $\rightarrow$  Plane

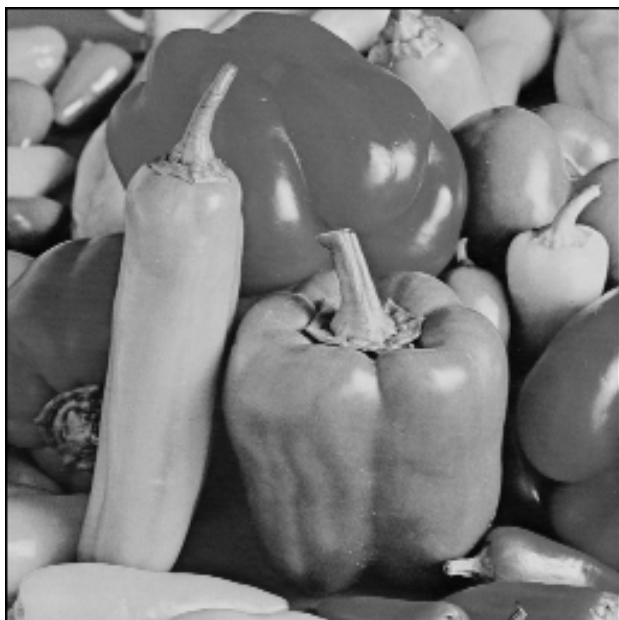
High Frequency Part  $\rightarrow$  Large Variation  $\rightarrow$  Edge and Noise

$G[m, n]$  low frequency part

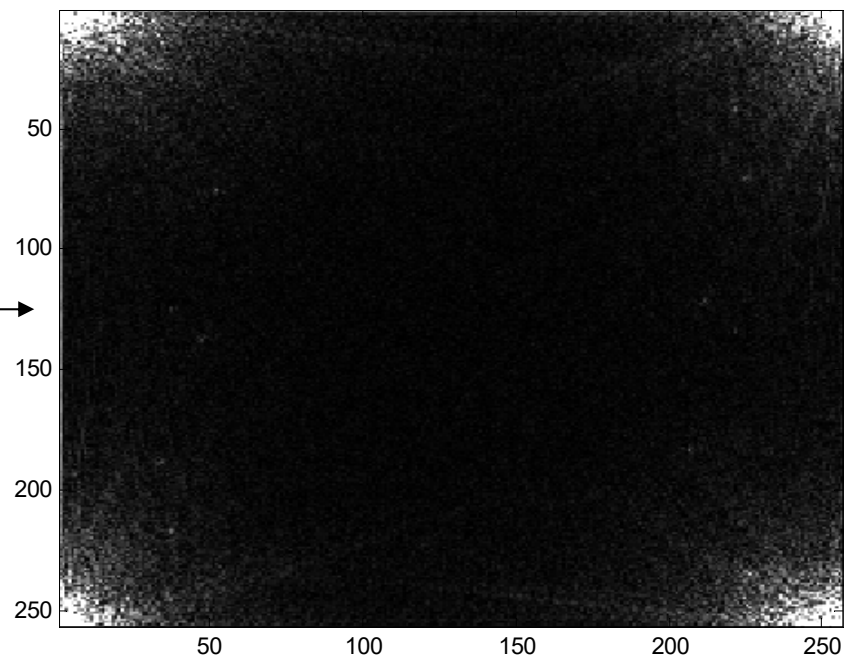


high frequency part





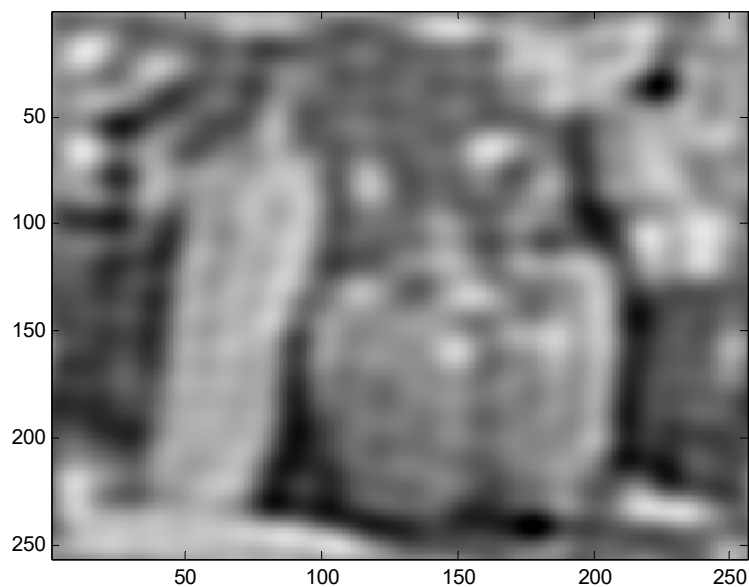
2D  
DFT →



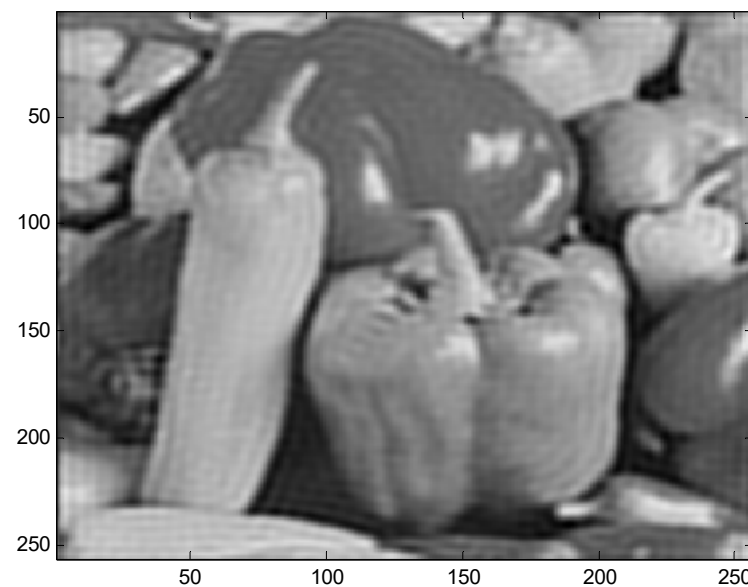
Using the gray level to show the intensity

```
image(.....)  
colormap(gray(256))
```

Low Frequency Part (similar to the blurred version of the input image)

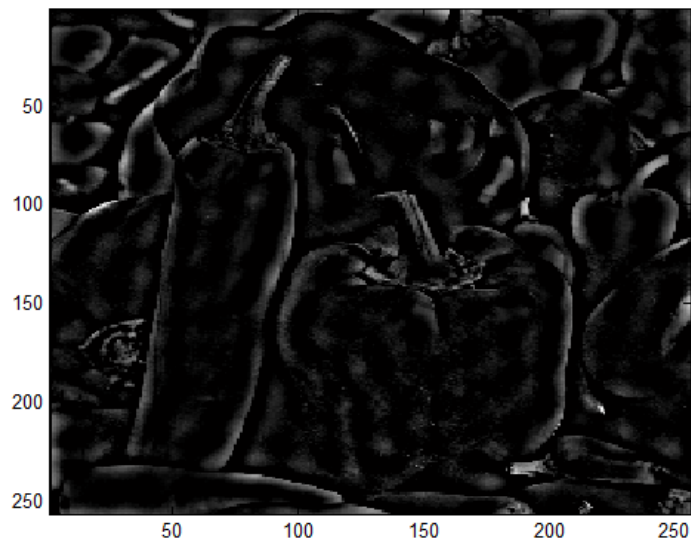


Passband:  $|f_x| + |f_y| \leq N/30$

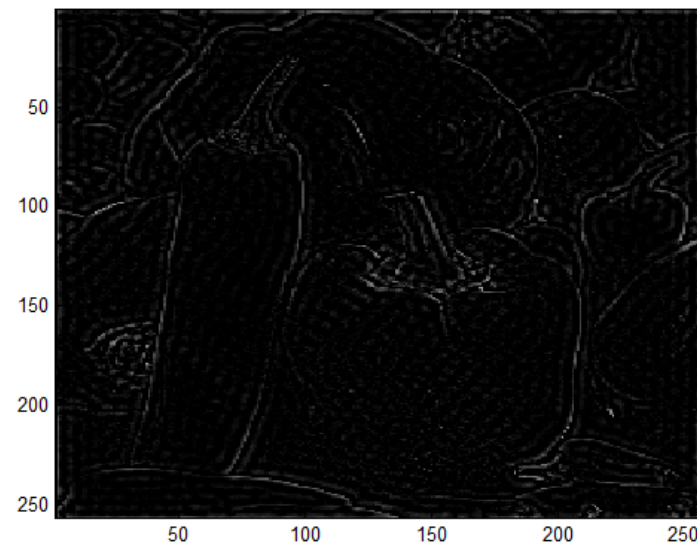


Passband:  $|f_x| + |f_y| \leq N/10$

High Frequency Part (similar to the edges)



Passband:  $|f_x| + |f_y| > N/30$



Passband:  $|f_x| + |f_y| > N/10$