

第三章 离散傅里叶变换 (DFT) 及其快速算法 (FFT)

王柯俨

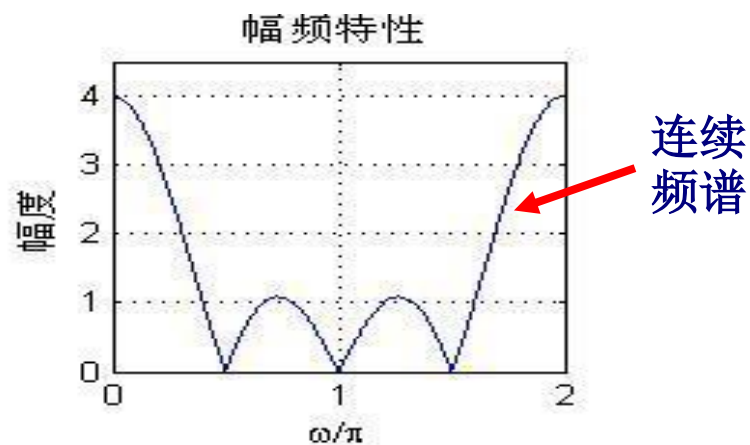
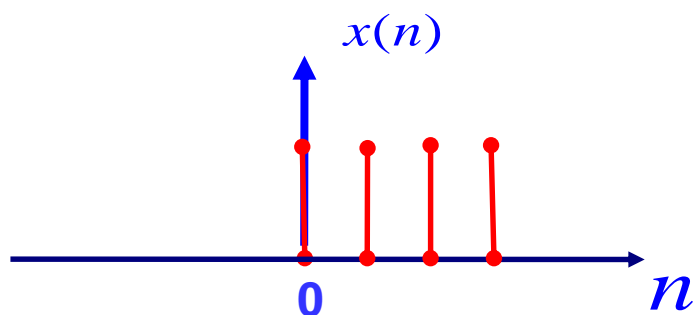
kywang@mail.xidian.edu.cn

<http://web.xidian.edu.cn/kywang/teach.html>

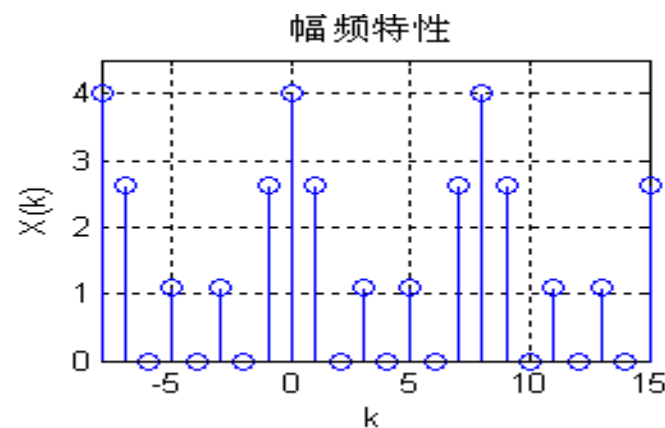
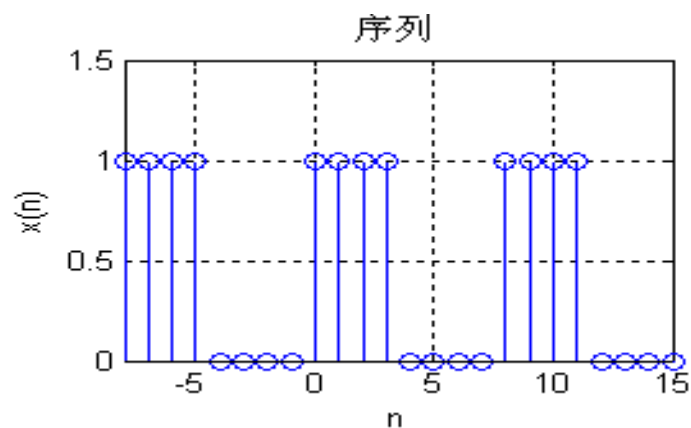
问题:

- 序列的傅里叶变换、 Z 变换是时域离散信号及系统分析与设计的重要数学工具;
- 但变换结果均为连续函数, 无法用计算机进行处理;

■ 绝对可和离散序列的傅立叶变换(DTFT)



■ 周期序列的离散傅里叶级数系数(DFS)



问题:

- 序列的傅里叶变换、 Z 变换是时域离散信号及系统分析与设计的重要数学工具;
- 但变换结果均为连续函数, 无法用计算机进行处理;
- 离散傅里叶变换 (**DFT**) 对有限长时域离散信号的频谱进行等间隔采样, 频域函数被离散化了, 便于信号的计算机处理。
- **DFT**运算量较大, 快速离散傅里叶变换算法 (**FFT**) 是解决方案

3.1 离散傅里叶变换(DFT)的定义

■ 3.1.1 DFT的定义

设 $x(n)$ 是一个长度为 M 的有限长序列， $x(n)$ 的 N 点离散傅立叶变换：

$$X(k) = DFT[x(n)]_N = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn} \quad 0 \leq k \leq N-1$$

□ N 称为DFT变换区间长度， $N \geq M$

- 令 $W_N = e^{-j\frac{2\pi}{N}}$ （简化书写）
- 离散傅立叶变换与逆变换对为：

$$X(k) = DFT[x(n)] = \sum_{n=0}^{N-1} x(n)W_N^{kn} \quad 0 \leq k \leq N-1$$
$$x(n) = IDFT[X(k)] = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-kn} \quad 0 \leq n \leq N-1$$

$$X(k) = DFT[x(n)] = \sum_{n=0}^{N-1} x(n)W_N^{kn} \quad 0 \leq k \leq N-1$$

证明:

$$\begin{aligned} \text{IDFT}[X(k)] &= \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-kn} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \left[\sum_{m=0}^{N-1} x(m)W_N^{mk} \right] W_N^{-kn} \\ &= \sum_{m=0}^{N-1} x(m) \frac{1}{N} \sum_{k=0}^{N-1} W_N^{k(m-n)} \end{aligned}$$

由于

$$\frac{1}{N} \sum_{k=0}^{N-1} W_N^{k(m-n)} = \frac{1}{N} \sum_{k=0}^{N-1} e^{-j\frac{2\pi}{N}k(m-n)} = \begin{cases} 1, & m-n = MN, M \text{ 为整数} \\ 0, & m-n \neq MN, M \text{ 为整数} \end{cases}$$

于是

$$\text{IDFT}[X(k)] = \sum_{m=0}^{N-1} x(m)\delta(m-n) = x(n) \quad 0 \leq n \leq N-1$$

因此离散傅立叶逆变换是唯一的

例3.1.1 $x(n) = R_8(n)$ 分别计算序列的8点、16点DFT

解：8点DFT

$$\begin{aligned} X(k) &= \sum_{n=0}^7 R_8(n) W_8^{kn} = \sum_{n=0}^7 e^{-j\frac{2\pi}{8}kn} \\ &= \begin{cases} 8 & k = 0 \\ 0 & k = 1, 2, 3, \dots, 7 \end{cases} \end{aligned}$$

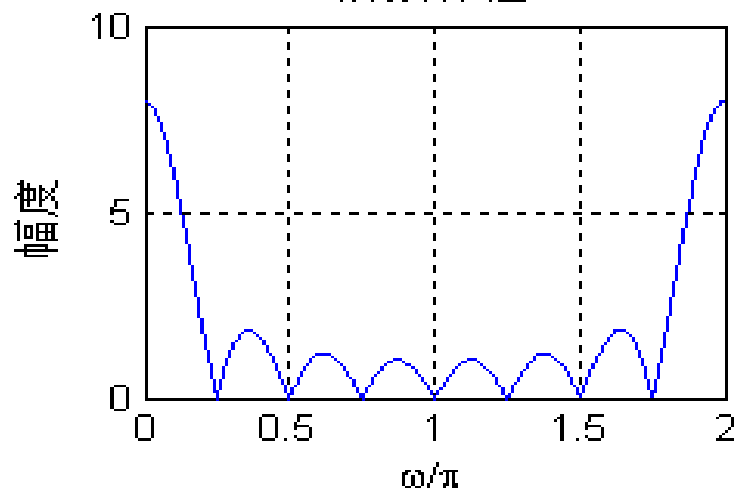
16点DFT

$$\begin{aligned} X(k) &= \sum_{n=0}^{15} R_8(n) W_{16}^{kn} = \sum_{n=0}^7 R_8(n) W_{16}^{kn} = \frac{1 - W_{16}^{k8}}{1 - W_{16}^k} = \frac{1 - e^{-j\frac{2\pi}{16}k8}}{1 - e^{-j\frac{2\pi}{16}k}} \\ &= \frac{1 - e^{-j\pi k}}{1 - e^{-j\frac{\pi}{8}k}} = \frac{e^{-j\frac{\pi k}{2}} (e^{j\frac{\pi k}{2}} - e^{-j\frac{\pi k}{2}})}{e^{-j\frac{\pi}{16}k} (e^{j\frac{\pi}{16}k} - e^{-j\frac{\pi}{16}k})} = e^{-j\frac{7\pi}{16}k} \frac{\sin \frac{\pi}{2} k}{\sin \frac{\pi}{16} k} \end{aligned}$$

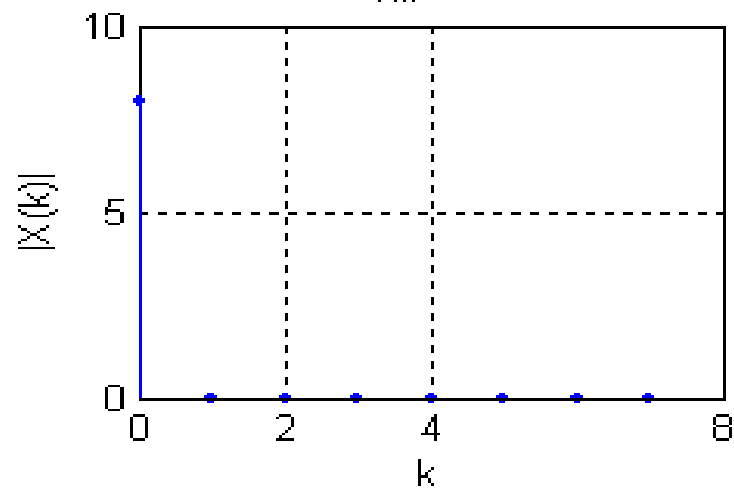
$$k = 0, 1, 2, \dots, 15$$

$$x(n) = R_8(n)$$

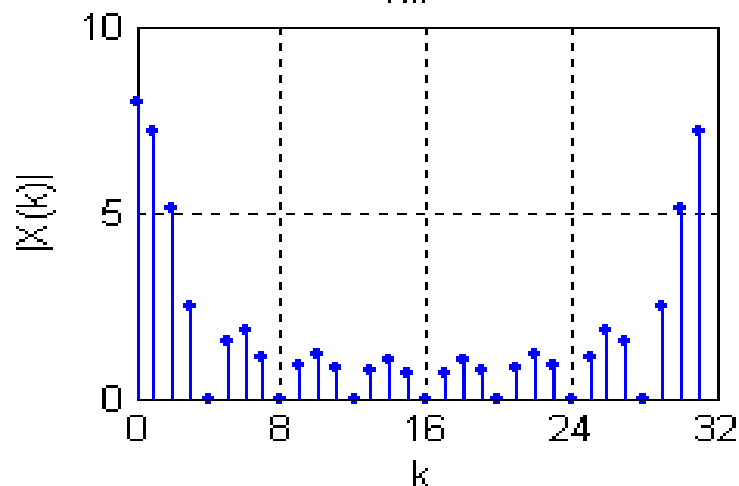
幅频特性



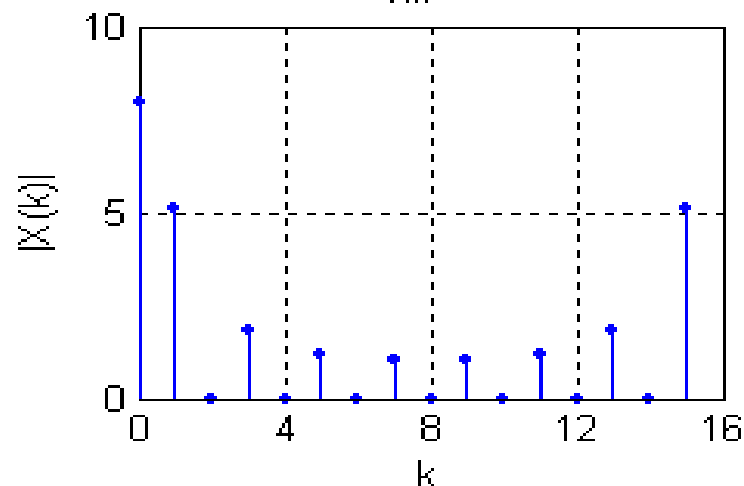
8点 DFT



32点 DFT



16点 DFT



- **N不同，DFT变换结果不同，因此N是DFT的一个参数**
- $|X(k)|$ 是 $|X(e^{j\omega})|$ 在频率区间 $[0, 2\pi]$ 上的**N点等间隔采样**

3.1.2 DFT与FT、ZT、DFS之间的关系

1. DFT与FT、ZT之间的关系

■ 有限长序列

$$x(n) \quad n = 0, 1, 2, \dots, M-1 \quad N \geq M$$

■ DFT与FT、ZT

从公式上看

$$X(k) = DFT[x(n)]_N = \sum_{n=0}^{M-1} x(n) e^{-j\frac{2\pi}{N}kn} \quad 0 \leq k \leq N-1$$

$$X(z) = ZT[x(n)] = \sum_{n=-\infty}^{\infty} x(n) z^{-n} = \sum_{n=0}^{M-1} x(n) z^{-n}$$

$$X(e^{j\omega}) = FT[x(n)] = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n} = \sum_{n=0}^{M-1} x(n) e^{-j\omega n}$$

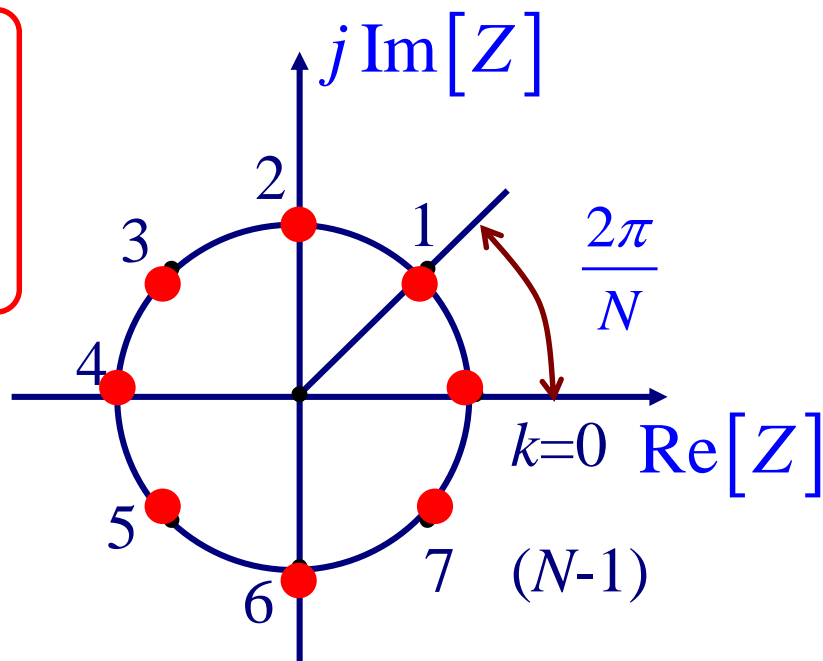
$$X(k) = X(z) \Big|_{z=e^{j\frac{2\pi}{N}k}}, \quad X(k) = X(e^{j\omega}) \Big|_{\omega=\frac{2\pi}{N}k}, \quad 0 \leq k \leq N-1$$

$$X(k) = X(z) \Big|_{z=e^{j\frac{2\pi}{N}k}},$$

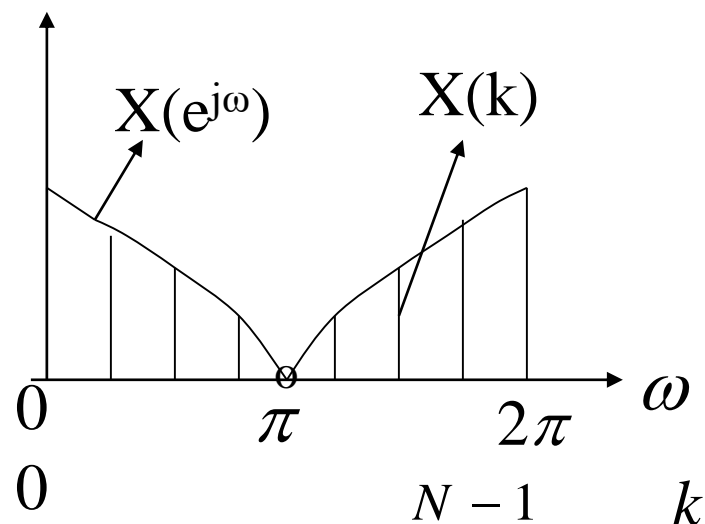
$$X(k) = X(e^{j\omega}) \Big|_{\omega=\frac{2\pi}{N}k}, \quad 0 \leq k \leq N-1$$

- DFT与z变换: 序列 $x(n)$ 的 N 点DFT是 $x(n)$ 的 Z 变换在单位圆上的 N 点等间隔采样, 频率采样间隔为 $2\pi / N$;
- DFT与FT变换: $X(k)$ 为 $x(n)$ 的傅立叶变换 $X(e^{j\omega})$ 在区间 $[0, 2\pi]$ 上的 N 点等间隔采样。

物理含义



DFT与z变换



DFT与FT变换

3.1.2 DFT与DFS、ZT、FT之间的关系

2. DFT与DFS之间的关系

- 长度为M的有限长序列 $x(n) \quad n = 0, 1, 2, \dots, M-1$
- 将 $x(n)$ 以N为周期进行周期延拓, 得到周期序列

$$\tilde{x}_N(n) = \sum_{m=-\infty}^{\infty} x(n + mN) = x((n))_N$$

$$x_N(n) = \tilde{x}_N(n)R_N(n)$$

显然, 当

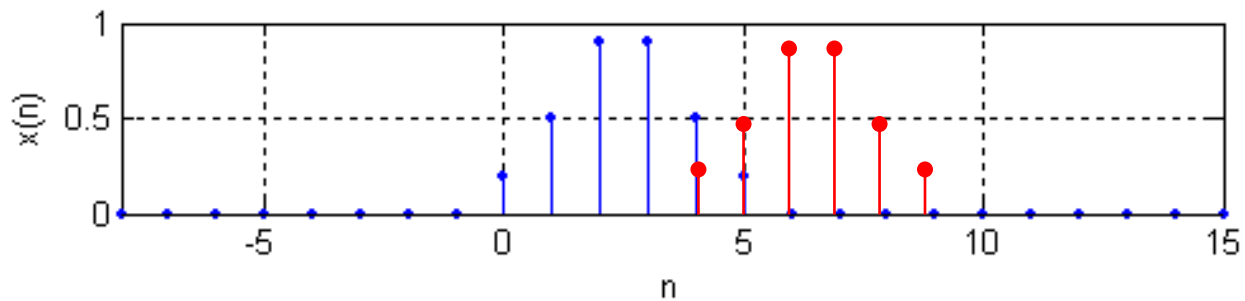
$$N \geq M, \quad x_N(n) = x(n)$$

$\tilde{x}_N(n)$ 是 $x(n)$ 的周期延拓序列

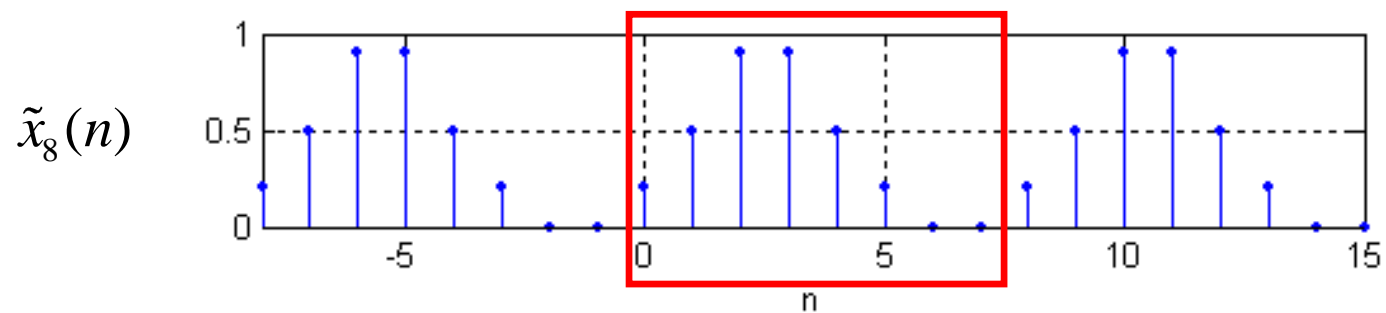
$x(n)$ 是 $\tilde{x}_N(n)$ 的主值区间序列

主值区间: 设有限长序列 $x(n)$, $0 \leq n \leq N-1$, 将其延拓为周期序列 $\tilde{x}(n)$, 周期长度为 N , 则: $n = 0 \sim N-1$ 区间称为主值区间.

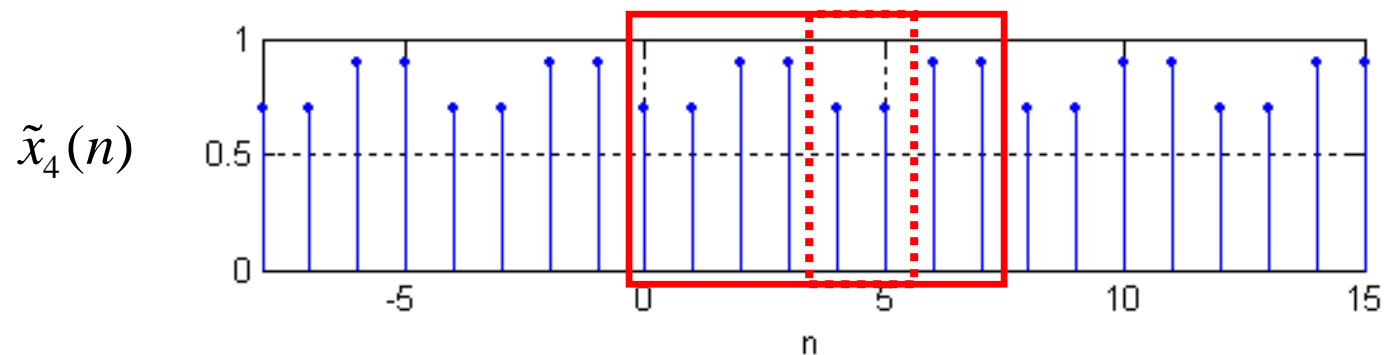
主值区间序列: 在 $n = 0 \sim N-1$ 主值区间内的序列称为主值区间序列.



M=6



N=8, N>M



N=4, N<M

■ 周期序列的DFS

$$\tilde{X}(k) = DFS[\tilde{x}_N(n)] = \sum_{n=0}^{N-1} \tilde{x}_N(n) W_N^{kn}$$

以N为周期

$$= \sum_{n=0}^{M-1} x(n) W_N^{kn} \quad -\infty < k < \infty$$

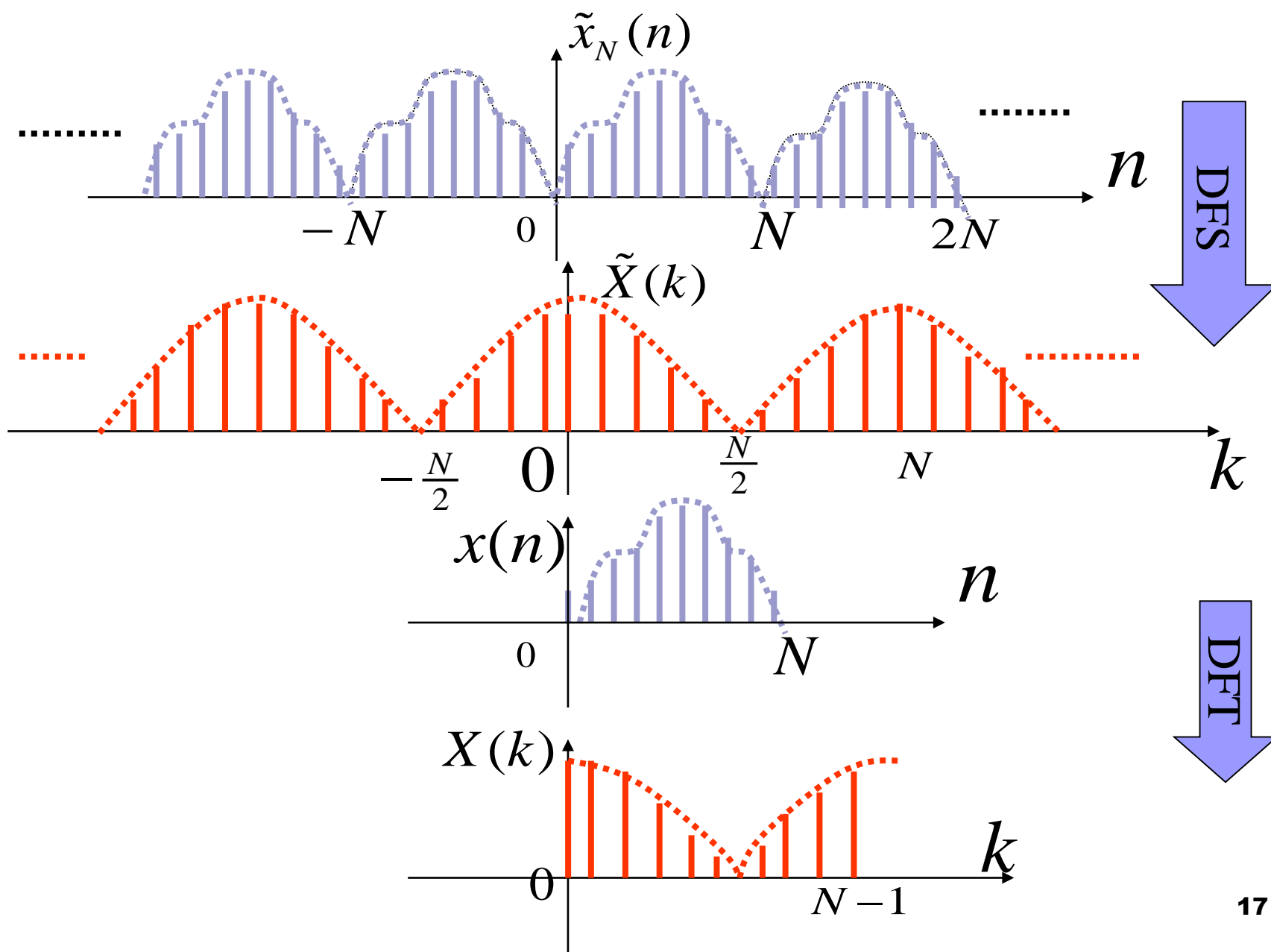
■ 有限长序列的DFT

$$X(k) = DFT[x(n)]_N = \sum_{n=0}^{N-1} x(n) W^{kn}$$

$$= \sum_{n=0}^{M-1} x(n) W^{kn} \quad 0 \leq k \leq N-1$$

- $X(k)$ 是 $\tilde{X}(k)$ 的主值区间序列，成立条件 $N \geq M$

$$\tilde{X}(k) = \sum_{m=-\infty}^{\infty} X(k + mN), \quad X(k) = \tilde{X}(k) R_N(k)$$



DFT与DFS之间的关系:

$$DFT: x(n) \Leftrightarrow X(k)$$

$$DFS: \tilde{x}_N(n) \Leftrightarrow \tilde{X}(k)$$

$$\begin{cases} x(n) = \tilde{x}_N(n)R_N(n) & \tilde{x}_N(n) = x((n))_N \\ X(k) = \tilde{X}(k)R_N(k) & \tilde{X}(k) = X((k))_N \end{cases}$$

$$N \geq M$$

- 有限长序列 $x(n)$ 的DFT变换 $X(k)$, 就是 $x(n)$ 的周期延拓序列 $\tilde{x}_N(n)$ 的DFS系数 $\tilde{X}(k)$ 的主值序列

DFT与DFS之间的关系:

$$DFT : x(n) \Leftrightarrow X(k)$$

$$DFS : \tilde{x}_N(n) \Leftrightarrow \tilde{X}(k)$$

- 当条件不满足, 即 $N < M$ 时, $x(n)$ 的周期延拓序列 $\tilde{x}_N(n)$ 会发生时域混叠, 此时 $\tilde{x}_N(n)$ 的DFS系数 $\tilde{X}(k)$ 的主值序列将不再是 $x(n)$ 的DFT变换 $X(k)$ 。

例： $x(n) = R_8(n)$, 求： (1) $X(z)$ (2) $X(e^{j\omega})$

(3) 令 $x_N(n) = \sum_{m=-\infty}^{\infty} x(n+mN) \cdot R_N(n)$, 求 $X_N(k) = DFT[x_N(n)]$, $N=16$

解： (1)
$$X(z) = \frac{1-z^{-8}}{1-z^{-1}}, \quad |z| > 0$$

(2)
$$X(e^{j\omega}) = \frac{1-e^{-j\omega 8}}{1-e^{-j\omega}} = e^{-j\frac{7}{2}\omega} \frac{\sin \frac{8}{2}\omega}{\sin \frac{1}{2}\omega}$$

(3)
$$X_N(k) = X(e^{j\omega}) \Big|_{\omega=\frac{2\pi}{N}k}$$
$$= e^{-j\frac{7\pi}{16}k} \frac{\sin \frac{\pi}{2}k}{\sin \frac{\pi}{16}k} \quad k = 0, 1, 2, \dots, 15$$

3.1.3 DFT的矩阵方程表示 (1)

$$X(k) = DFT[x(n)]_N = \sum_{n=0}^{N-1} x(n)W_N^{kn} \quad 0 \leq k \leq N-1$$

$$\mathbf{X} = \mathbf{D}_N \mathbf{x}$$

$$\mathbf{X} = \begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(N-1) \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix}$$

**N点
DFT矩阵**

$$\mathbf{D}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^1 & W_N^2 & \cdots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{(N-1)} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix}$$

3.1.3 DFT的矩阵方程表示（2）

$$\mathbf{X} = \mathbf{D}_N \mathbf{x}$$

■ IDFT的矩阵表示

$$x(n) = IDFT[X(k)] = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn} \quad 0 \leq n \leq N-1$$

$$\mathbf{x} = \mathbf{D}_N^{-1} \mathbf{X}$$

**N点
IDFT矩阵**

$$D_N^{-1} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \cdots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \cdots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \cdots & W_N^{-(N-1) \times (N-1)} \end{bmatrix}$$

$$D_N = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N^1 & W_N^2 & \dots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \dots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{(N-1)} & W_N^{2(N-1)} & \dots & W_N^{(N-1) \times (N-1)} \end{bmatrix}$$

$$D_N^{-1} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \dots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \dots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \dots & W_N^{-(N-1) \times (N-1)} \end{bmatrix}$$

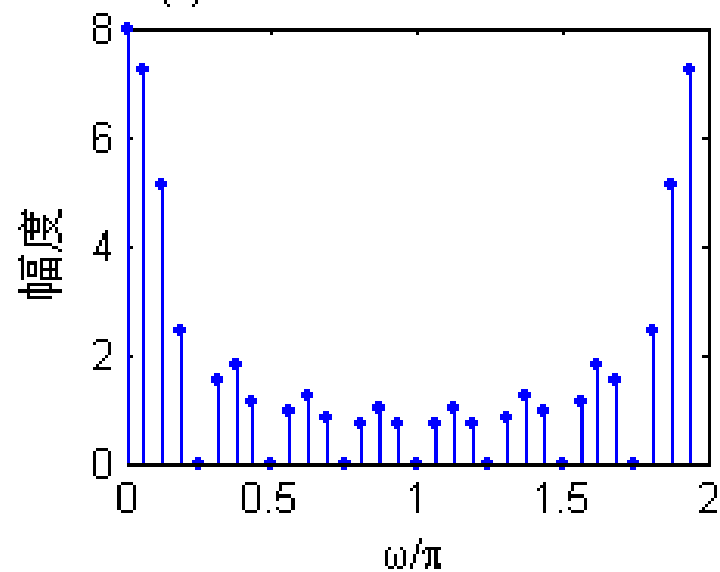
\therefore

$$D_N^{-1} = \frac{1}{N} D_N^*$$

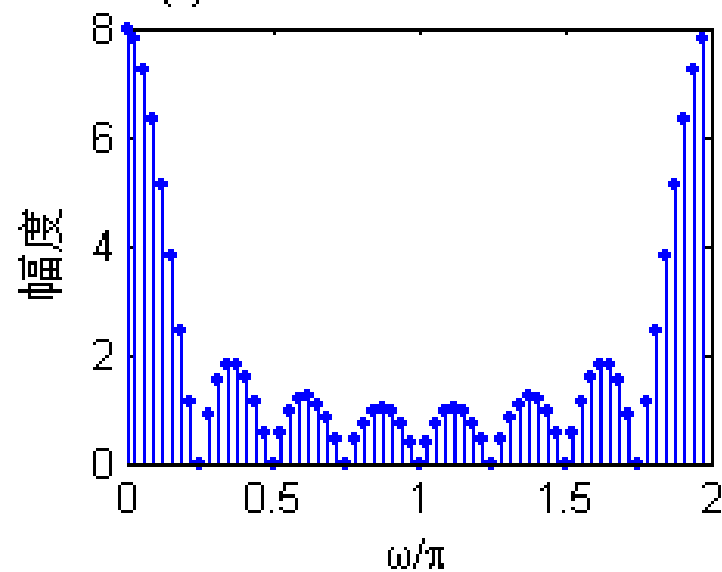
3.1.4用MATLAB计算序列的DFT

- `xn=[1 1 1 1 1 1 1 1];` %输入时域序列向量`xn=R8(n)`
- `Xk32=fft(xn,32);` %计算`xn`的32点DFT
- `Xk64=fft(xn,64);` %计算`xn`的64点DFT
- %以下为绘图部分
- `k=0:31;wk=2*k/32;` %产生32点DFT对应的采样点频率(关于 π 归一化值)
- `subplot(2,2,1);stem(wk,abs(Xk32),'.');` %绘制32点DFT的幅频特性图
- `title('(a)32点DFT的幅频特性图');``xlabel('ω/π');``ylabel('幅度')`
- `subplot(2,2,3);stem(wk,angle(Xk32),'.');` %绘制32点DFT的相频特性图
- `title('(b)32点DFT的相频特性图');`
- `xlabel('ω/π');``ylabel('相位');``axis([0,2,-3.5,3.5])`
- `k=0:63;wk=2*k/64;` %产生64点DFT对应的采样点频率(关于 π 归一化值)
- `subplot(2,2,2);stem(wk,abs(Xk64),'.');` %绘制64点DFT的幅频特性图
- `title('(c)64点DFT的幅频特性图');``xlabel('ω/π');``ylabel('幅度')`
- `subplot(2,2,4);stem(wk,angle(Xk64),'.');` %绘制64点DFT的相频特性图
- `title('(d)64点DFT的相频特性图');`
- `xlabel('ω/π');``ylabel('相位');``axis([0,2,-3.5,3.5])`

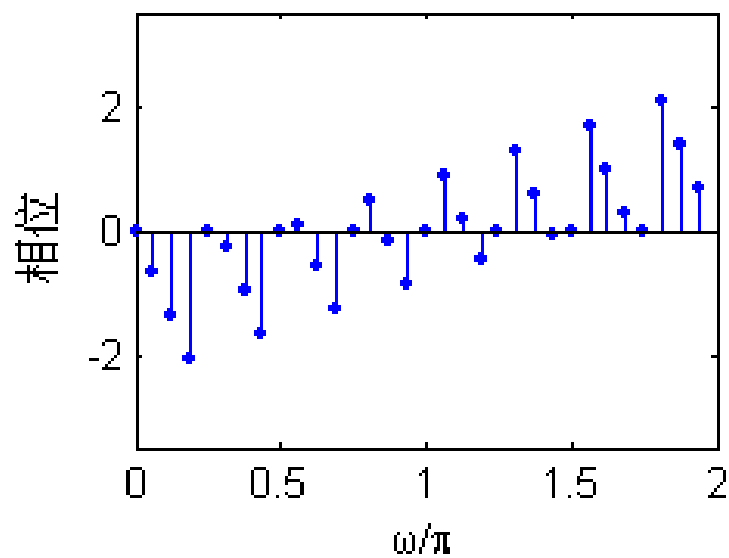
(a) 32点DFT的幅频特性图



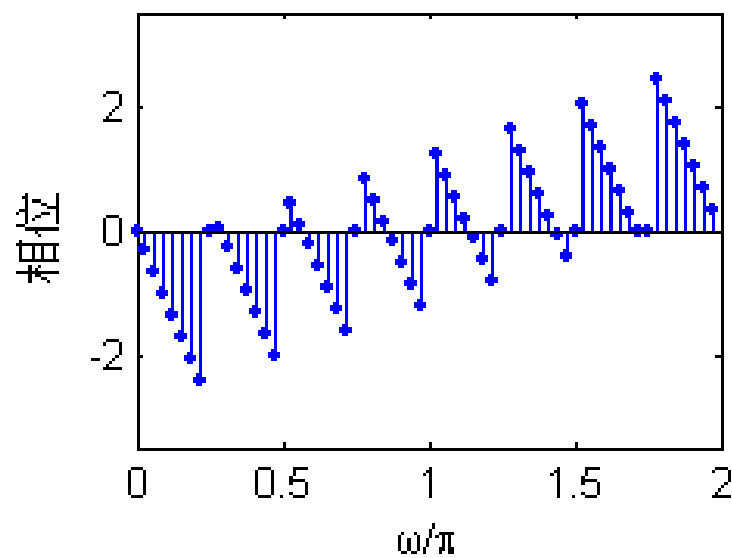
(c) 64点DFT的幅频特性图



(b) 32点DFT的相频特性图



(d) 64点DFT的相频特性图



小结

- DFT 引入的目的
- DFT 的定义
- DFT与DFS、ZT、FT之间的关系
- DFT、IDFT的计算
- DFT的矩阵表示



3.2 离散傅里叶变换 (DFT) 的主要性质

1. 线性性质

- 设 $x_1(n)$, $x_2(n)$ 为有限长序列, 长度分别为 N_1 、 N_2
- 它们的离散付里叶变换分别为

$$X_1(k) = DFT[x_1(n)]_N$$

$$X_2(k) = DFT[x_2(n)]_N \quad N \geq \max[N_1, N_2]$$

- 若

$$x(n) = ax_1(n) + bx_2(n)$$

- 则

$$X(k) = DFT[x(n)]_N = aX_1(k) + bX_2(k),$$

$$k = 0, 1, 2, \dots, N-1$$

2. DFT的隐含周期性

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn}, \quad 0 \leq k \leq N-1$$

■ 由于

$$W_N = e^{-j\frac{2\pi}{N}}, \quad W_N^k = W_N^{(k+mN)}$$

■ 所以 $X(k)$ 满足:

$$\begin{aligned} X(k+mN) &= \sum_{n=0}^{N-1} x(n)W_N^{(k+mN)n} \\ &= \sum_{n=0}^{N-1} x(n)W_N^{kn} = X(k) \end{aligned}$$

- 物理意义: $X(k)$ 为 $X(e^{j\omega})$ 在区间 $[0, 2\pi]$ 上的 N 点等间隔采样。
- $X(e^{j\omega})$ 以 2π 为周期, $X(k)$ 以 N 为周期。

3. 循环移位性质

- 循环移位:

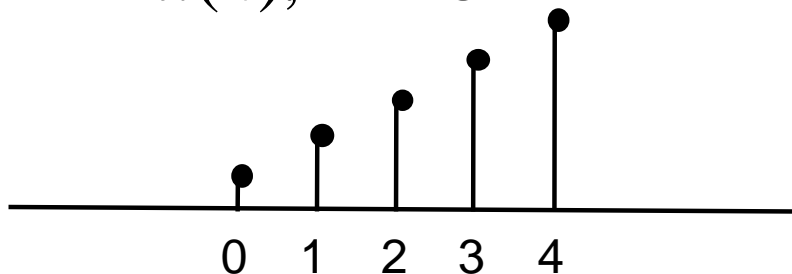
- 有限长序列 $\mathbf{x(n)}$,序列长度为 \mathbf{M} , 对序列进行周期延拓, 周期 $\mathbf{N \geq M}$

$$\tilde{x}_N(n) = \sum_{m=-\infty}^{\infty} x(n + mN) = x((n))_N$$

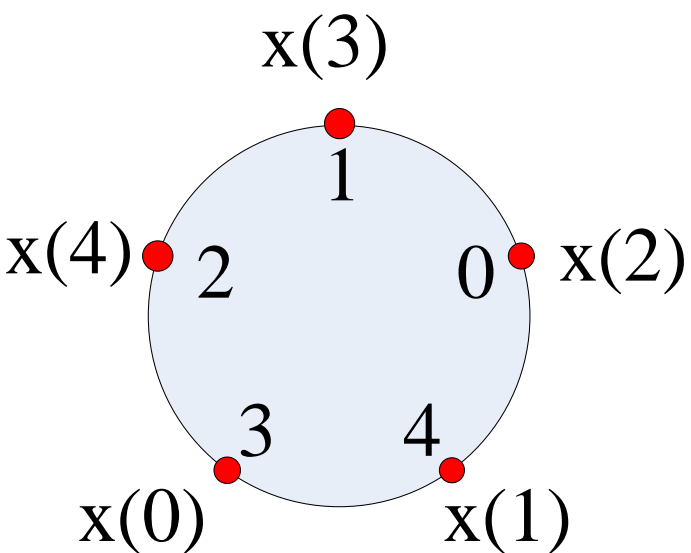
- $\mathbf{x(n)}$ 的循环移位序列: $\tilde{x}_N(n)$ 左移 \mathbf{m} 个单位, 取主值序列

$$y(n) = \tilde{x}_N(n + m)R_N(n) = x((n + m))_N R_N(n)$$

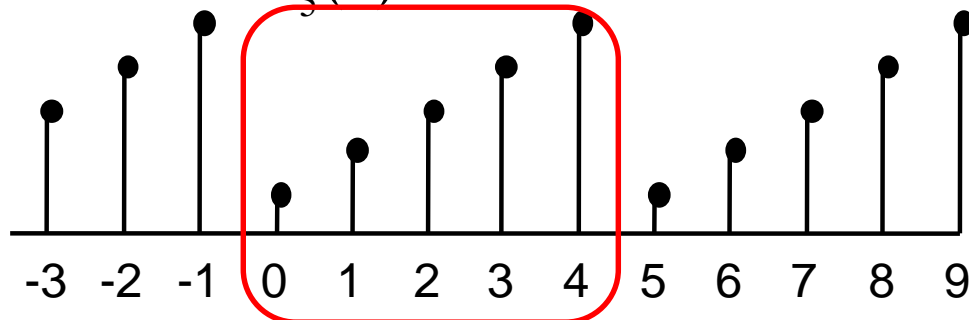
$$x(n), M = 5$$



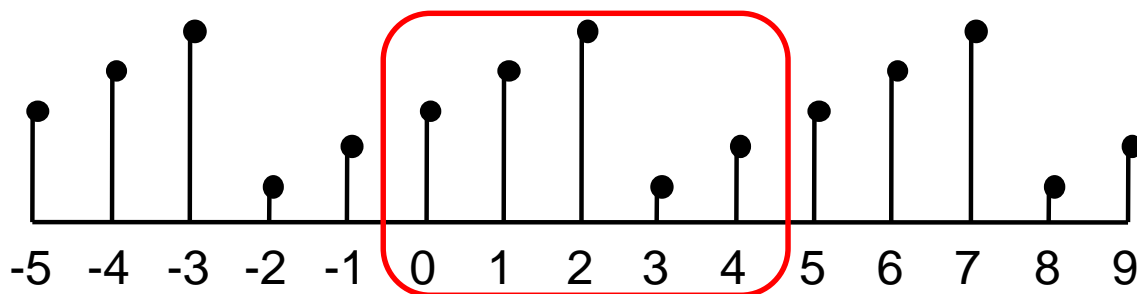
循环移位



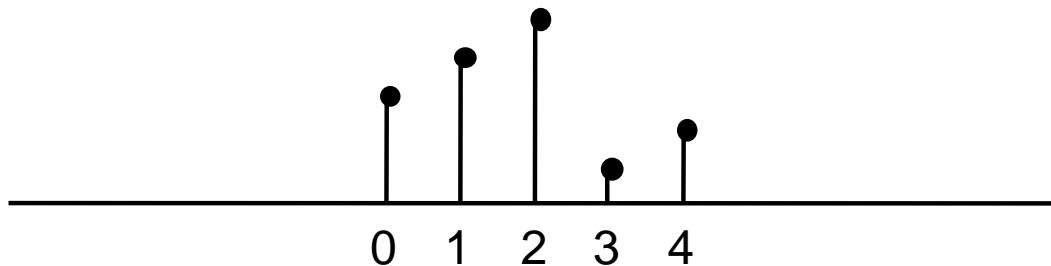
$$\tilde{x}_5(n)$$



$$\tilde{x}_5(n+2)$$



$$\tilde{x}_5(n+2)R_5(n)$$



3. 循环移位性质

- 循环移位:

- 有限长序列 $\mathbf{x(n)}$,序列长度为 \mathbf{M} , 对序列进行周期延拓, 周期 $\mathbf{N \geq M}$

$$\tilde{x}_N(n) = \sum_{m=-\infty}^{\infty} x(n + mN) = x((n))_N$$

- $\mathbf{x(n)}$ 的循环移位序列: $\tilde{x}_N(n)$ 左移 \mathbf{m} 个单位, 取主值序列

$$y(n) = \tilde{x}_N(n + m)R_N(n) = x((n + m))_N R_N(n)$$

- 循环移位序列的DFT与原序列 $\mathbf{x(n)}$ 的DFT有何关系?

序列循环移位后的DFT

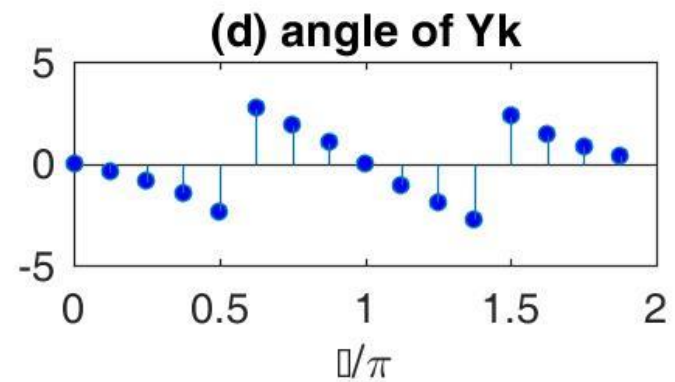
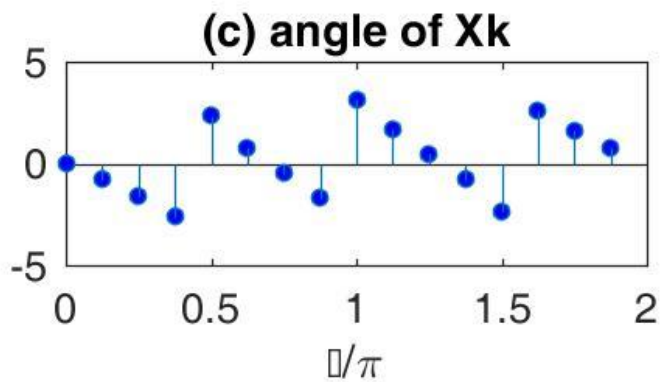
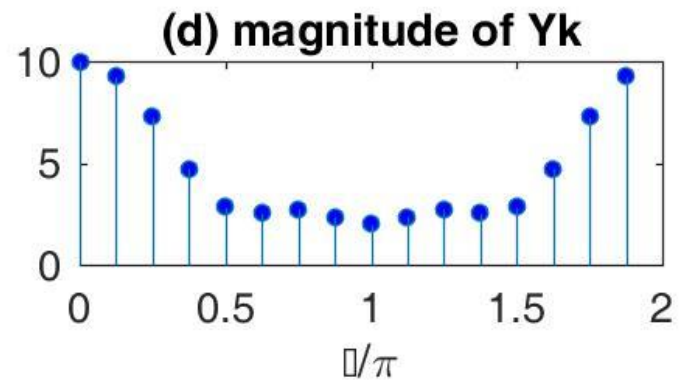
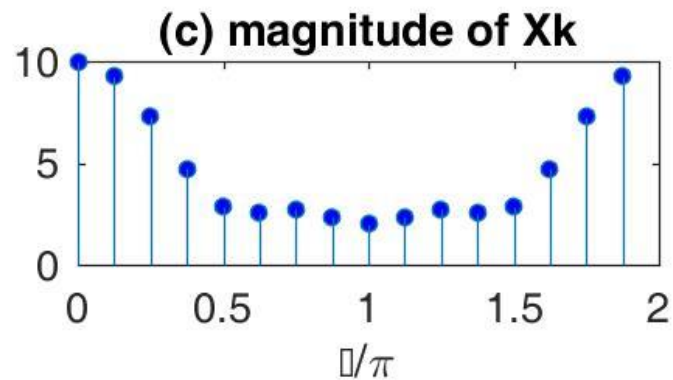
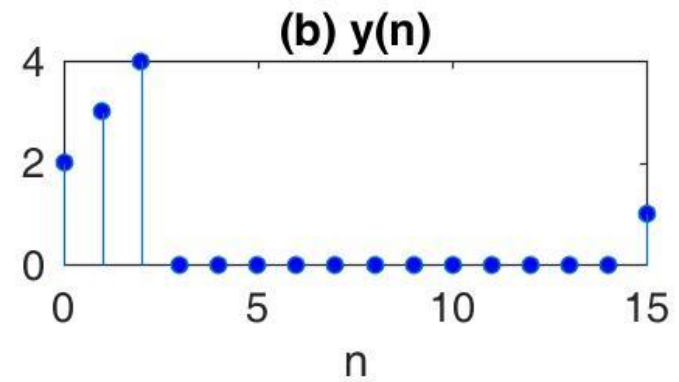
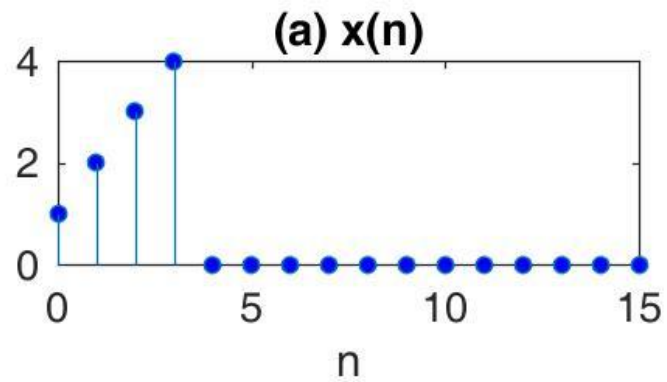
$$y(n) = x((n+m))_N R_N(n), \quad N \geq M \quad X(k) = DFT[x(n)]_N$$

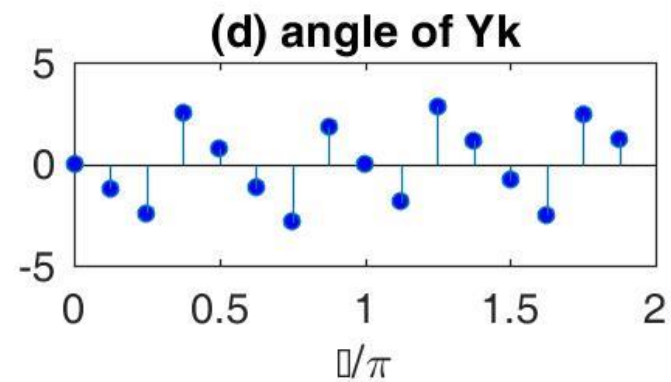
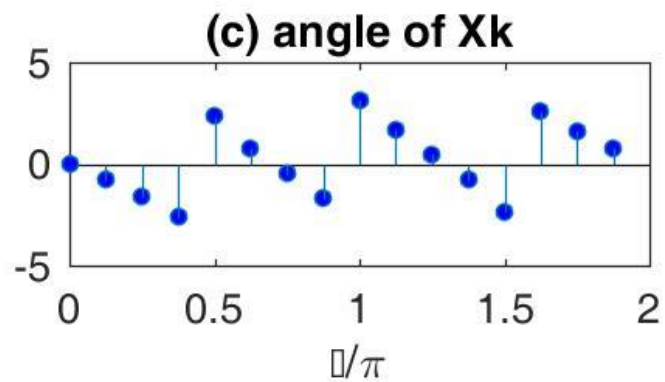
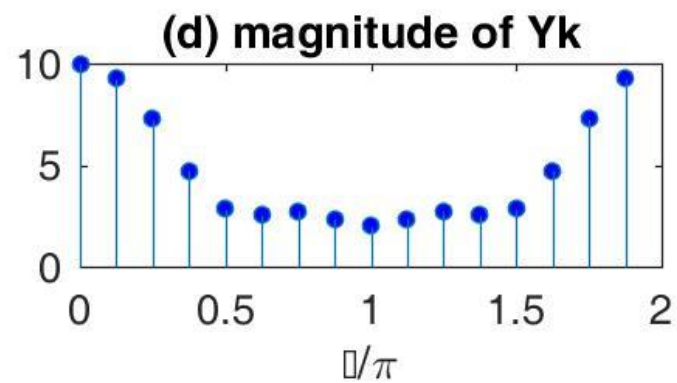
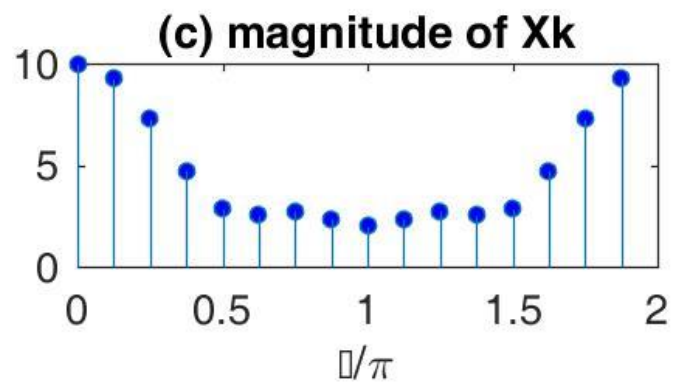
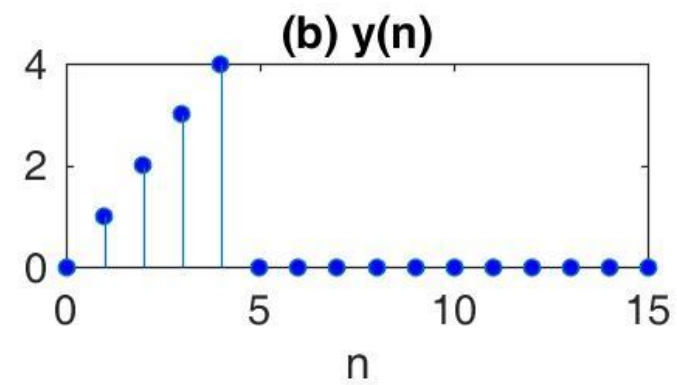
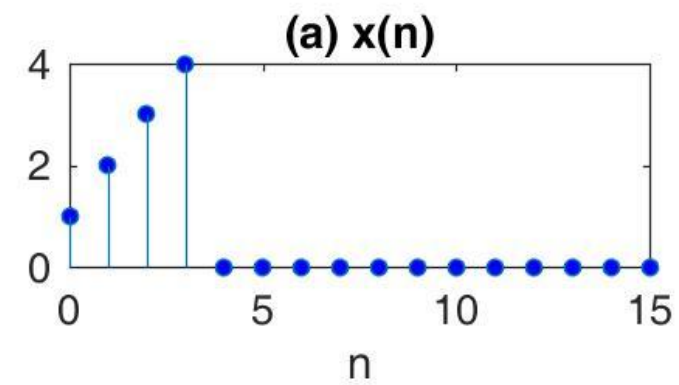
■ 则

$$Y(k) = DFT[y(n)]_N = W_N^{-km} X(k)$$

■ 证明:

$$\begin{aligned} Y(k) &= \sum_{n=0}^{N-1} x((n+m))_N R_N(n) W_N^{kn} \quad k = 0, 1, \dots, N-1 \\ &= \sum_{n=0}^{N-1} x((n+m))_N W_N^{kn} = \sum_{l=m}^{N-1+m} x((l))_N W_N^{k(l-m)} \\ &= W_N^{-km} \sum_{l=m}^{N-1+m} x((l))_N W_N^{kl} = W_N^{-km} \sum_{l=0}^{N-1} x(l)_N W_N^{kl} \\ &= W_N^{-km} X(k) \end{aligned}$$





- 同样，对于频域有限长序列 $X(k)$ 的循环移位，有如下反变换特性：

$$IDFT[X((k+l))_N R_N(k)] = W_N^{nl} x(n)$$

4. 复共轭序列的DFT

- 设 $x^*(n)$ 为 $x(n)$ 的复共轭序列，长度为N

$$X(k) = DFT[x(n)]_N$$

- 则 $DFT[x^*(n)]_N = X^*(N-k), \quad k = 0, 1, \dots, N-1$

- 证明:

$$\begin{aligned} X^*(N-k) &= \left[\sum_{n=0}^{N-1} x(n) W_N^{(N-k)n} \right]^* = \sum_{n=0}^{N-1} x^*(n) W_N^{(k-N)n} \\ &= \sum_{n=0}^{N-1} x^*(n) W_N^{kn} = DFT[x^*(n)]_N \end{aligned}$$

$$X(N) = X(0)$$

- 类似地:

$$DFT[x^*(N-n)]_N = X^*(k) \quad x(0) = x(N)$$

5. DFT的共轭对称性

- FT的共轭对称性：关于坐标原点共轭对称
- DFT的共轭对称性：DFT变换涉及到的 $x(n)$ 和 $X(k)$ 均为有限长序列，定义区间为0到N-1，所以这里的对称性是指关于N/2点的对称性。

- 有限长共轭对称序列 $x_{ep}(n)$

$$x_{ep}(n) = x_{ep}^*(N - n), \quad 0 \leq n \leq N - 1$$

- 当N为偶数时，用N/2-n替代n

$$x_{ep}\left(\frac{N}{2} - n\right) = x_{ep}^*\left(\frac{N}{2} + n\right), \quad 0 \leq n \leq \frac{N}{2} - 1$$

共轭反对称序列 $x_{op}(n)$

$$x_{op}(n) = -x_{op}^*(N-n), \quad 0 \leq n \leq N-1$$

■ 当N为偶数时，用N/2-n替代n

$$x_{op}\left(\frac{N}{2} - n\right) = -x_{op}^*\left(\frac{N}{2} + n\right), \quad 0 \leq n \leq \frac{N}{2} - 1$$

■ 对于任何有限长序列x(n)，均可表示为

$$x(n) = x_{ep}(n) + x_{op}(n), \quad 0 \leq n \leq N-1$$

■ 用N-n替代n，取共轭

$$x^*(N-n) = x_{ep}^*(N-n) + x_{op}^*(N-n) = x_{ep}(n) - x_{op}(n)$$

■ 于是

$$x_{ep}(n) = \frac{1}{2}[x(n) + x^*(N-n)] \quad x_{op}(n) = \frac{1}{2}[x(n) - x^*(N-n)]$$

DFT的共轭对称性

■ 1. 将序列分成实部与虚部之和

$$x(n) = x_r(n) + jx_i(n)$$

其中,

$$x_r(n) = \text{Re}[x(n)] = \frac{1}{2}[x(n) + x^*(n)]$$
$$jx_i(n) = j \text{Im}[x(n)] = \frac{1}{2}[x(n) - x^*(n)]$$

■ 则:

$$\begin{aligned} DFT[x_r(n)] &= \frac{1}{2} DFT[x(n) + x^*(n)] \\ &= \frac{1}{2} [X(k) + X^*(N-k)] = X_{ep}(k) \\ DFT[jx_i(n)] &= \frac{1}{2} DFT[x(n) - x^*(n)] \\ &= \frac{1}{2} [X(k) - X^*(N-k)] = X_{op}(k) \end{aligned}$$

■ 2. 将序列分成共轭对称和共轭反对称之和

$$x(n) = x_{ep}(n) + x_{op}(n), \quad 0 \leq n \leq N-1$$

其中, $x_{ep}(n) = \frac{1}{2}[x(n) + x^*(N-n)]$

$$x_{op}(n) = \frac{1}{2}[x(n) - x^*(N-n)]$$

■ 则:
$$\begin{aligned} DFT[x_{ep}(n)] &= \frac{1}{2} DFT[x(n) + x^*(N-n)] \\ &= \frac{1}{2} [X(k) + X^*(k)] = \text{Re}[X(k)] \end{aligned}$$

$$\begin{aligned} DFT[x_{op}(n)] &= \frac{1}{2} DFT[x(n) - x^*(N-n)] \\ &= \frac{1}{2} [X(k) - X^*(k)] = j \text{Im}[X(k)] \end{aligned}$$

小结：DFT的共轭对称性

实部 \longleftrightarrow 共轭对称分量

J虚部 \longleftrightarrow 共轭反对称分量

$$x(n) = x_r(n) + jx_i(n)$$

$$X(k) = DFT[x(n)] = X_{ep}(k) + X_{op}(k)$$

$$X_{ep}(k) = DFT[x_r(n)], \quad X_{op}(k) = DFT[jx_i(n)]$$

$$x(n) = x_{ep}(n) + x_{op}(n), \quad 0 \leq n \leq N-1$$

$$X(k) = DFT[x(n)] = X_R(k) + jX_I(k)$$

$$X_R(k) = \text{Re}[X(k)] = DFT[x_{ep}(n)],$$

$$jX_I(k) = j \text{Im}[X(k)] = DFT[x_{op}(n)]$$

实序列DFT的特点

- 设 $x(n)$ 为长度为 N 的实序列，且 $X(k)=\text{DFT}[x(n)]$ ，则

$$X(k) = X^*(N-k), \quad 0 \leq k \leq N-1$$

- 写成极坐标形式 $X(k) = |X(k)|e^{j\theta(k)}$

- 则

- $|X(k)|$ 关于 $k=N/2$ 点偶对称
- $\theta(k)$ 关于 $k=N/2$ 点奇对称

$$|X(k)| = |X(N-k)|$$

$$\theta(k) = -\theta(N-k)$$

- 实数序列的**DFT**满足共轭对称性，利用这一特性，只要知道一半数目的 $X(k)$ ，就可得到另一半的 $X(k)$ ，这一特点在 **DFT**运算中可以加以利用，以提高运算效率。

1. 计算一个复序列的**N**点**DFT**，可求得两个不同实序列的**DFT**

例： $x_1(n), x_2(n)$ 是实序列，长度均为**N**，构造一个复序列

$$x(n) = x_1(n) + jx_2(n)$$

- **DFT** $X(k) = DFT[x(n)]_N = X_{ep}(k) + X_{op}(k)$

$$X_1(k) = DFT[x_1(n)] = X_{ep}(k) = \frac{1}{2} [X(k) + X^*(N-k)]$$

$$X_2(k) = DFT[x_2(n)] = -jDFT[jx_2(n)]$$

复数乘法运算量比较
 $2N^2 \rightarrow N^2$

$$= -jX_{op}(k) = \frac{1}{2j} [X(k) - X^*(N-k)]$$

2. 实序列的2N点DFT，可以拆分重组为N点复序列的DFT

例： $v(n)$ 是实序列，长度为2N

- 拆分 $x_1(n) = v(2n) \quad x_2(n) = v(2n+1) \quad 0 \leq n \leq N-1$
- 重组 $x(n) = x_1(n) + jx_2(n) \quad 0 \leq n \leq N-1$
- 计算复序列 $x(n)$ 的N点DFT，可得实序列 $x_1(n)$ 和 $x_2(n)$ 的N点DFT： $X_1(k)$ 和 $X_2(k)$
- 由 $X_1(k)$ 和 $X_2(k)$ 可得实序列 $v(n)$ 的2N点DFT，即 $V(k)$

$$\begin{aligned}
 V(k) &= \sum_{n=0}^{2N-1} v(n)W_{2N}^{kn} = \sum_{n=0}^{N-1} v(2n)W_{2N}^{k2n} + \sum_{n=0}^{N-1} v(2n+1)W_{2N}^{k(2n+1)} \\
 &= \sum_{n=0}^{N-1} x_1(n)W_N^{kn} + W_{2N}^k \sum_{n=0}^{N-1} x_2(n)W_N^{kn} \\
 &= X_1(k) + W_{2N}^k X_2(k) \quad 0 \leq k \leq N-1
 \end{aligned}$$

$$V(k+N) = X_1(k) - W_{2N}^k X_2(k)$$

复数乘法运算量比较
 $(2N)^2 \rightarrow N^2+N$

基2FFT基本思想

6. 循环卷积定理

1. 两个有限长序列的循环卷积

- 序列 $h(n), x(n)$ 的长度分别为 **N** 和 **M**
- $x(n)$ 与 $h(n)$ 的 **L** 点循环卷积定义为

$$y_c(n) = h(n) \textcircled{L} x(n) = \left[\sum_{m=0}^{L-1} h(m) x((n-m))_L \right] R_L(n)$$

$$L \geq \max[N, M]$$

\textcircled{L} L点循环卷积

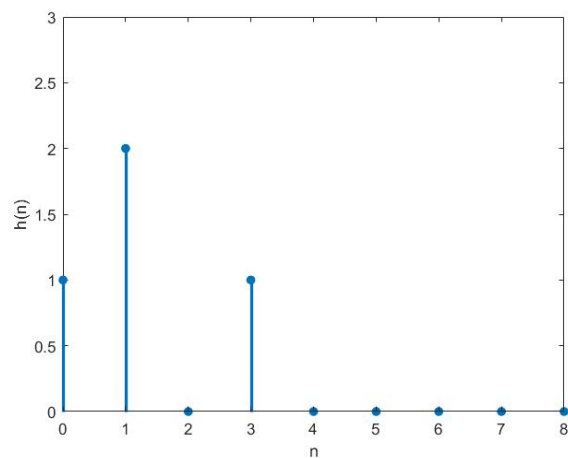
$\textcircled{*}$ 循环卷积

$*$ 线性卷积

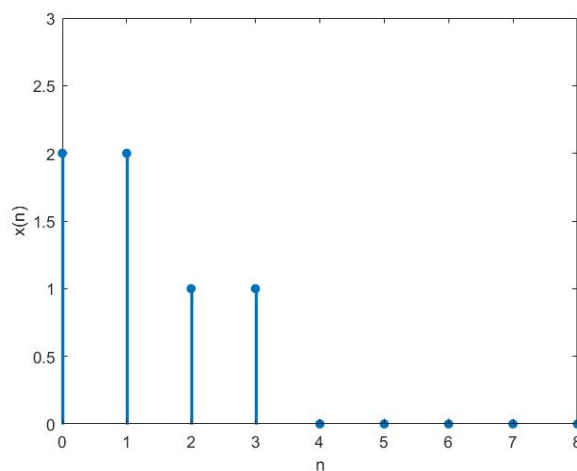
- **m, n** 的范围: **0~L-1**
- 直接计算很麻烦, 采用矩阵相乘计算

6. 循环卷积定理

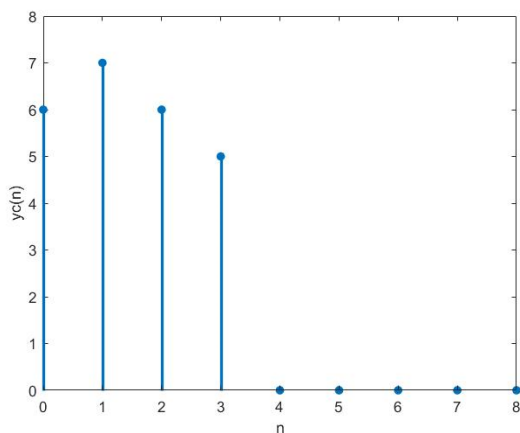
循环卷积图例



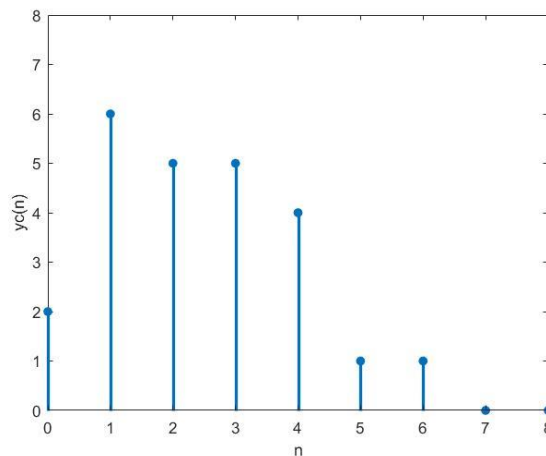
$h(n)$ 波形



$x(n)$ 波形



4点循环卷积波形



8点循环卷积波形

6. 循环卷积定理

■ 循环卷积与线性卷积的区别

- 两个N点序列的N点循环卷积的结果仍为N点序列，而它们的线性卷积的结果的长度则为 $2N-1$ 。
- 循环卷积对序列的位移采取循环位移，而线性卷积对序列采取线性位移。

■ 循环卷积与线性卷积的联系

- 两个序列的N点循环卷积是他们的线性卷积以N为周期延拓。

$$y(n) = [h(n) \otimes x(n)]_N = \left(\sum_{r=-\infty}^{\infty} y'(n - rN) \right) G_N(n)$$

其中， $y'(n) = h(n) * x(n)$ ，

利用矩阵计算循环卷积

$$y_c(n) = \left[\sum_{m=0}^{L-1} h(m)x((n-m))_L \right] R_L(n)$$

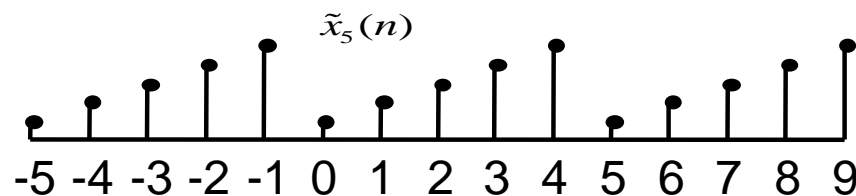
$$n = 0, m = 0, 1, 2, \dots, L-1$$

$x((n-m))_L$ 形成 $x(n)$ 的循环倒相序列

$$\begin{aligned} & \{x((0))_L, x((-1))_L, x((-2))_L, \dots, x((-l+1))_L\} \\ &= \{x(0), x(L-1), x(L-2), \dots, x(1)\} \end{aligned}$$

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

$x((n-m))_L$ 形成的序列为



$$\begin{aligned} & \{x((1))_L, x((0))_L, x((-1))_L, \dots, x((-L+2))_L\} \\ &= \{x(1), x(0), x(L-1), \dots, x(2)\} \end{aligned}$$

当n、m从0变换到L-1时，得到 $x((n-m))_L$ 的矩阵

**X(n)的L点
循环卷积矩阵
(循环矩阵)**

$$\begin{bmatrix} x(0) & x(L-1) & x(L-2) & \cdots & x(1) \\ x(1) & x(0) & x(L-1) & \cdots & x(2) \\ x(2) & x(1) & x(0) & \cdots & x(3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x(L-1) & x(L-2) & x(L-3) & \cdots & x(0) \end{bmatrix}$$

- 第一行是 $\{x(0), x(1), x(2), \cdots, x(L-1)\}$ 的循环倒相序列，若x(n) 的长度M<L，则需在x(n)末尾补L-M个零；
- 前一行向右循环移位形成其它各行；
- 与主对角线平行的线上，各元的值相等。

循环卷积的矩阵计算公式

$$y_c(n) = h(n) \textcircled{L} x(n) = \left[\sum_{m=0}^{L-1} h(m)x((n-m))_L \right] R_L(n)$$

$$L \geq \max[N, M]$$

$$\begin{bmatrix} y_c(0) \\ y_c(1) \\ y_c(2) \\ \vdots \\ y_c(L-1) \end{bmatrix} = \begin{bmatrix} x(0) & x(L-1) & x(L-2) & \cdots & x(1) \\ x(1) & x(0) & x(L-1) & \cdots & x(2) \\ x(2) & x(1) & x(0) & \cdots & x(3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x(L-1) & x(L-2) & x(L-3) & \cdots & x(0) \end{bmatrix} \begin{bmatrix} h(0) \\ h(1) \\ h(2) \\ \vdots \\ h(L-1) \end{bmatrix}$$

若 $h(n)$ 的长度 $N < L$, 则需在 $h(n)$ 末尾补 $L-N$ 个零

例3.2.1: 计算序列 $h(n), x(n)$ 的4点和8点循环卷积

$$h(n) = \{h(0), h(1), h(2), h(3)\} = \{1, 2, 0, 1\},$$

$$x(n) = \{x(0), x(1), x(2), x(3)\} = \{2, 2, 1, 1\},$$

解:

$$y_c(n) = h(n) \bigcirc_L x(n) = \sum_{m=0}^{L-1} h(m)x((n-m))_L R_L(n)$$

■ 4点循环卷积

$$\begin{bmatrix} y_c(0) \\ y_c(1) \\ y_c(2) \\ y_c(3) \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 & 2 \\ 2 & 2 & 1 & 1 \\ 1 & 2 & 2 & 1 \\ 1 & 1 & 2 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 7 \\ 6 \\ 5 \end{bmatrix}$$

8点循环卷积

$$\begin{bmatrix} y_c(0) \\ y_c(1) \\ y_c(2) \\ y_c(3) \\ y_c(5) \\ y_c(6) \\ y_c(7) \\ y_c(8) \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 2 & 2 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 2 & 2 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 2 & 2 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 6 \\ 5 \\ 5 \\ 4 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

- $L \geq N + M - 1$ 循环卷积结果等于线性卷积
- 循环卷积计算复杂

DFT的时域循环卷积定理（1）

- 序列 $h(n), x(n)$ 的长度分别为 **N** 和 **M**
- $y_c(n)$ 为 $x(n)$ 与 $h(n)$ 的 **L** 点循环卷积，即

$$y_c(n) = h(n) \bigcircled{L} x(n) \quad L \geq \max[N, M]$$

- 且 $X(k) = DFT[x(n)]_L \quad H(k) = DFT[h(n)]_L$

- 则

$$Y_c(k) = DFT[y_c(n)]_L = H(k)X(k)$$

DFT的时域循环卷积定理（2）

■ 证明：

$$\begin{aligned} Y_c(k) &= DFT[y_c(n)]_L = \sum_{n=0}^{L-1} y_c(n) W_L^{kn} \\ &= \sum_{n=0}^{L-1} \left\{ \left[\sum_{m=0}^{L-1} h(m) x((n-m))_L \right] R_L(n) \right\} W_L^{kn} \\ &= \sum_{m=0}^{L-1} h(m) \sum_{n=0}^{L-1} x((n-m))_L W_L^{kn} \\ &= \sum_{m=0}^{L-1} h(m) W_L^{km} \sum_{n=0}^{L-1} x((n-m))_L W_L^{k(n-m)} \\ &= \sum_{m=0}^{L-1} h(m) W_L^{km} \sum_{j=-m}^{L-1-m} x((j))_L W_L^{kj} \end{aligned}$$

DFT的时域循环卷积定理（3）

$$Y_c(k) = \sum_{m=0}^{L-1} h(m) W_L^{km} \sum_{j=-m}^{L-1-m} x((j))_L W_L^{kj}$$

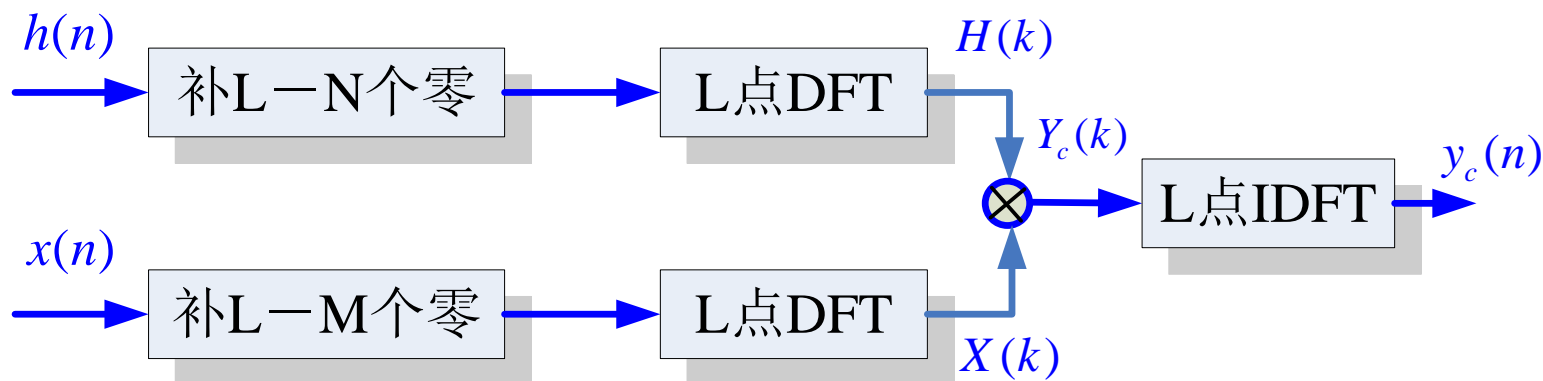
$x((j))_L W_L^{kj}$ 以L为周期，对其在任何一个周期上求和均相等

$$\begin{aligned} \therefore Y_c(k) &= \sum_{m=0}^{L-1} h(m) W_L^{km} \sum_{j=0}^{L-1} x((j))_L W_L^{kj} \\ &= H(k) X(k), \quad 0 \leq k \leq L-1 \end{aligned}$$

DFT: 时域循环卷积  频域乘积

快速卷积法

- 利用DFT计算两个有限长序列L点循环卷积的运算框图



DFT的频域循环卷积定理

- 序列 $h(n), x(n)$ 的长度分别为**N**和**M**

$$y_m(n) = h(n)x(n), \quad H(k) = DFT[h(n)]_L, \quad X(k) = DFT[x(n)]_L,$$

- **则** $Y_m(k) = DFT[y_m(n)]_L = \frac{1}{L} H(k) \textcircled{L} X(k)$

- **证明:**
$$\begin{aligned} Y_m(k) &= DFT[y_m(n)] = \sum_{n=0}^{L-1} h(n)x(n)W_L^{kn} \\ &= \sum_{n=0}^{L-1} \left[\frac{1}{L} \sum_{m=0}^{L-1} H(m)W_L^{-mn} \right] x(n)W_L^{kn} \\ &= \frac{1}{L} \sum_{m=0}^{L-1} H(m) \sum_{n=0}^{L-1} W_L^{-mn} x(n)W_L^{kn} = \frac{1}{L} \sum_{m=0}^{L-1} H(m) \sum_{n=0}^{L-1} x(n)W_L^{(k-m)n} \\ &= \frac{1}{L} \sum_{m=0}^{L-1} H(m) X((k-m))_L R_L(k) = \frac{1}{L} H(k) \textcircled{L} X(k) \end{aligned}$$

7. 离散巴塞伐尔定理 (1)

- 序列 $x(n)$ 的长度为N

- **DFT** $X(k) = DFT[x(n)]_N$

- 则

$$\sum_{n=0}^{N-1} |x(n)|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2$$

7. 离散巴塞伐尔定理 (2)

■ 证明

$$\begin{aligned}\sum_{n=0}^{N-1} |x(n)|^2 &= \sum_{n=0}^{N-1} x(n)x^*(n) = \sum_{n=0}^{N-1} \left[\frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-kn} \right] x^*(n) \\&= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \sum_{n=0}^{N-1} x^*(n)W_N^{-kn} = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \left[\sum_{n=0}^{N-1} x(n)W_N^{kn} \right]^* \\&= \frac{1}{N} \sum_{k=0}^{N-1} X(k)X^*(k) = \frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2\end{aligned}$$

■ 序列在时域计算的能量等于其在频域计算的能量



3.3 频域采样

3.3.1 频域采样定理

- 回顾：时域采样定理

对带限信号，采样频率大于等于奈奎斯特采样频率，可以由离散信号恢复原来的连续信号

- 频域采样定理

频域采样呢？

采样条件？

内插公式？

- 任意序列 $x(n)$ ，其Z变换为

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

- 若Z变换 $X(z)$ 的收敛域包含单位圆，即序列绝对可和，在单位圆上对 $X(z)$ 等间隔采样得到

$$\tilde{X}_N(k) = X(z) \Big|_{z=e^{j\frac{2\pi}{N}k}} = \sum_{n=-\infty}^{\infty} x(n)e^{-j\frac{2\pi}{N}kn}$$

- $\tilde{X}_N(k)$ 是以 N 为周期的周期序列，其对应的时域周期序列为

$$\tilde{x}_N(n) = IDFS \left[\tilde{X}_N(k) \right] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}_N(k) e^{j\frac{2\pi}{N}kn}$$

$$\tilde{x}_N(n) \Rightarrow x(n) \quad ?$$

$$\begin{aligned}\tilde{x}_N(n) &= IDFS[\tilde{X}_N(k)] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}_N(k) e^{j\frac{2\pi}{N}kn} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \left[\sum_{m=-\infty}^{\infty} x(m) e^{-j\frac{2\pi}{N}km} \right] e^{j\frac{2\pi}{N}kn} = \sum_{m=-\infty}^{\infty} x(m) \frac{1}{N} \sum_{k=0}^{N-1} e^{j\frac{2\pi}{N}k(n-m)}\end{aligned}$$

$$\frac{1}{N} \sum_{k=0}^{N-1} e^{j\frac{2\pi}{N}k(n-m)} = \begin{cases} 1, & m = n + iN, i \text{ 为整数} \\ 0, & m \text{ 为其它值} \end{cases}$$

$$\tilde{x}_N(n) = \sum_{i=-\infty}^{\infty} x(n + iN)$$

- $\tilde{x}_N(n)$ 是 $\mathbf{x(n)}$ 的周期延拓序列，延拓周期为 **N**

■ 截取 $\tilde{x}_N(n)$ 和 $\tilde{X}_N(k)$ 的主值序列

$$X_N(k) = \tilde{X}_N(k)R_N(k) = X(z)\Big|_{z=e^{j\frac{\pi}{N}k}}, \quad k = 0, 1, \dots, N-1$$

$$x_N(n) = \tilde{x}_N(n)R_N(n) = x(n)$$

则 $x_N(n)$ 和 $X_N(k)$ 构成一对**DFT**

■ 由 $\tilde{x}_N(n)$ 恢复出**x(n)**的条件:

- **x(n)**序列长度有限
- **N**大于**x(n)**序列长度

总结:

频域采样定理

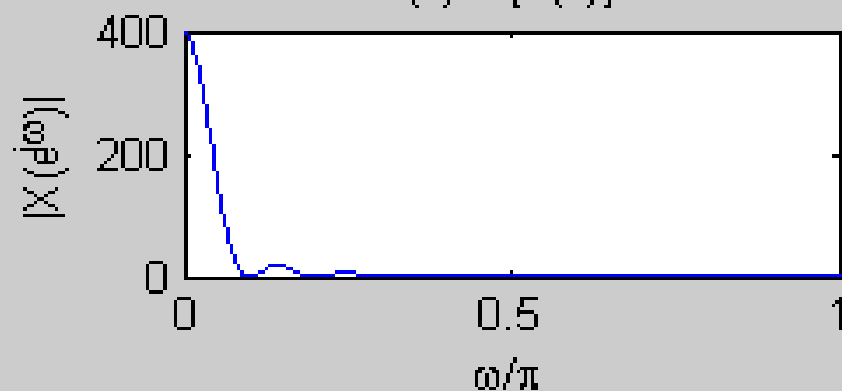
- 原若序列 $x(n)$ 长度为 M ，其傅里叶变换为 $X(e^{j\omega})$
- 对 $X(e^{j\omega})$ 在频率区间 $[0, 2\pi]$ 等间隔采样得到 $X_N(k)$
- 只有当频域采样点数 $N \geq M$
- 有

$$\tilde{x}_N(n)R_N(n) = IDFS[\tilde{X}(k)]R_N(n) = x(n)$$

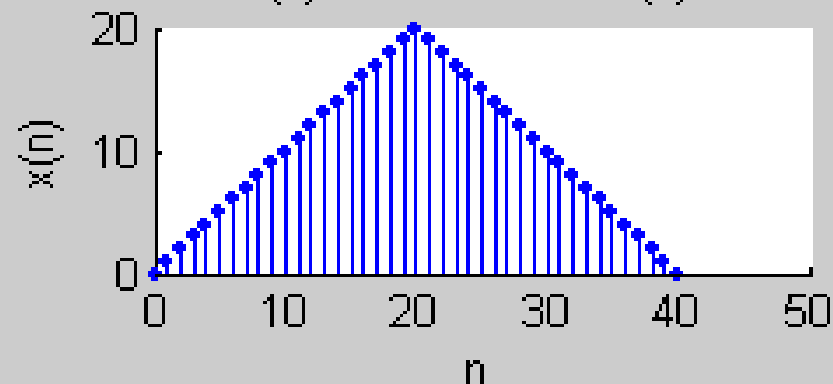
$$\text{或: } x(n) = IDFT[X_N(k)]_N$$

- 即可由频域采样 $X_N(k)$ 不失真地恢复原信号，否则产生时域混叠现象。
- 当时域序列 $x(n)$ 有限长，以大于等于序列长度的采样间隔对其频谱函数进行采样，不会丢失信息。

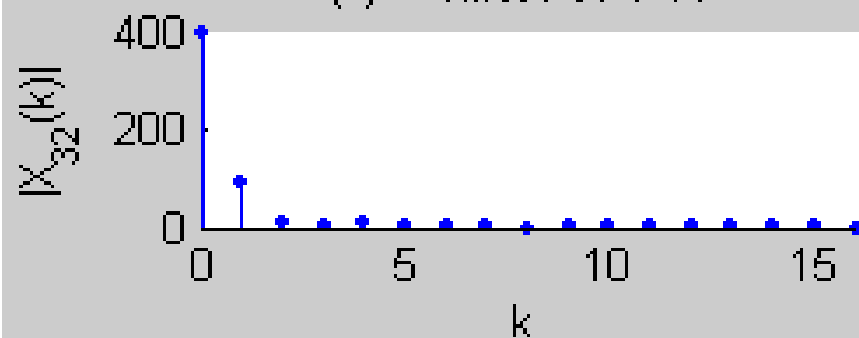
(a) $FT[x(n)]$



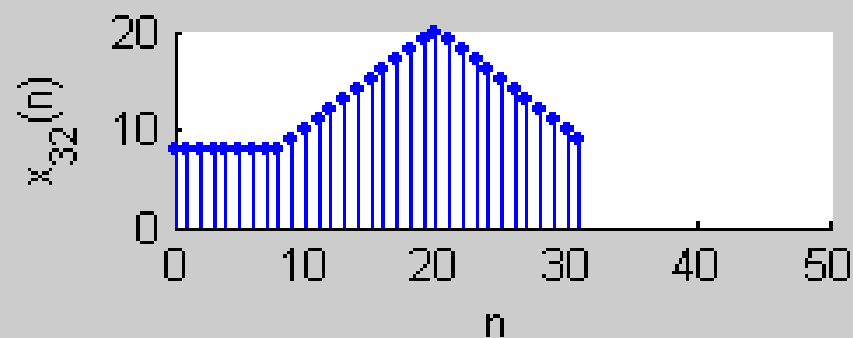
(b) 三角波序列 $x(n)$



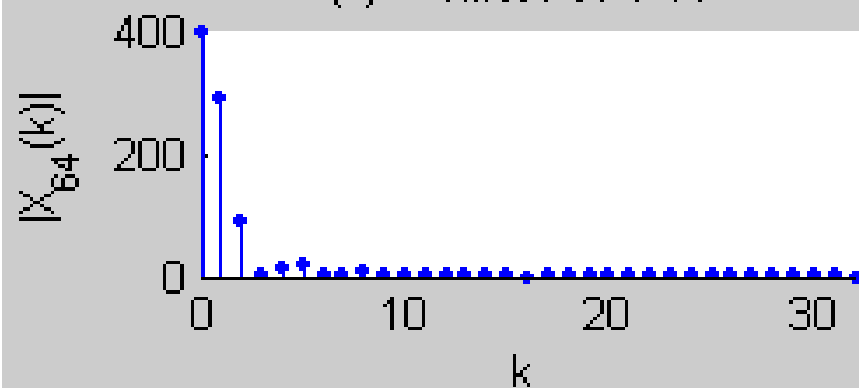
(c) 32点频域采样



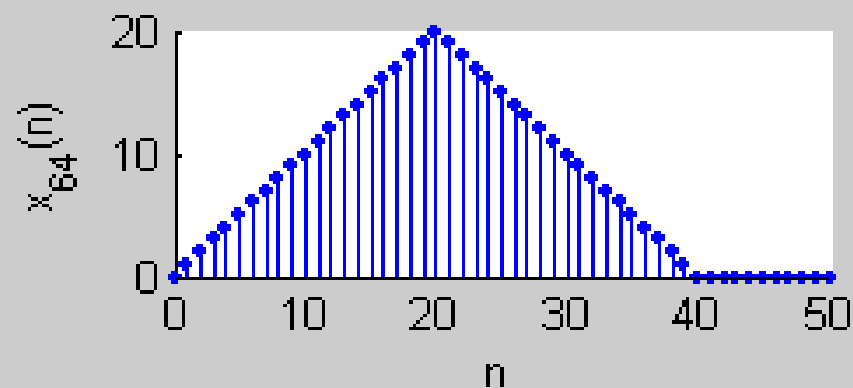
(d) 32点IDFT $[X_{32}(k)]$



(e) 64点频域采样



(f) 64点IDFT $[X_{64}(k)]$



3.3.2 频域内插公式

- 用频域采样 $X(k)$ 表示 $X(z)$ 和 $X(e^{j\omega})$ 的内插公式?
- 序列 $x(n)$ 长度为 M ，其 Z 变换为

$$X(z) = \sum_{n=0}^{N-1} x(n)z^{-n} = \sum_{n=0}^{M-1} x(n)z^{-n}$$

- 其中 $N \geq M$ ，满足频域采样定理
- 在 Z 平面的单位圆上，对 $X(z)$ 进行采样，采样点数为 N

$$X(k) = X(z) \Big|_{z=e^{j\frac{2\pi}{N}k}} \quad k = 0, 1, \dots, N-1$$

问题：如何由 $X(k)$ 恢复 $X(z)$ ？

$$\begin{aligned} X(z) &= \sum_{n=0}^{N-1} x(n) z^{-n} = \sum_{n=0}^{N-1} \left[\frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk} \right] z^{-n} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \left[\sum_{n=0}^{N-1} W_N^{-nk} z^{-n} \right] \\ &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \sum_{n=0}^{N-1} \left(W_N^{-k} z^{-1} \right)^n \\ &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \frac{1 - W_N^{-Nk} z^{-N}}{1 - W_N^{-k} z^{-1}} \end{aligned}$$

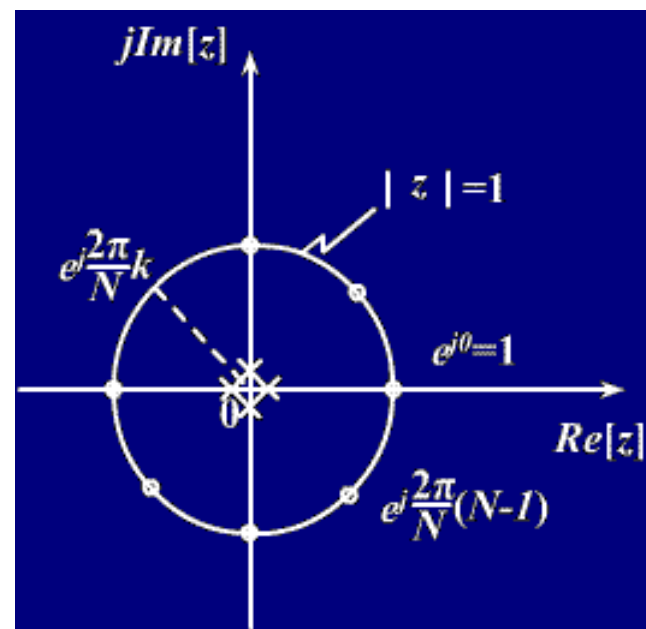
$$\begin{aligned}
 X(z) &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \frac{1 - W_N^{-Nk} z^{-N}}{1 - W_N^{-k} z^{-1}} \quad (W_N^{-Nk} = 1) \\
 &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \frac{1 - z^{-N}}{1 - W_N^{-k} z^{-1}} = \sum_{k=0}^{N-1} X(k) \varphi_k(z)
 \end{aligned}$$

Z域内插函数

$$\varphi_k(z) = \frac{1}{N} \frac{z^N - 1}{z^{N-1} (z - W_N^{-k})}$$

零点: $z = e^{j\frac{2\pi}{N}r}, \quad r = 0, 1, \dots, N-1$

极点: $z = e^{j\frac{2\pi}{N}k}, \quad 0 \quad (N-1 \text{ 阶})$



$$\text{令 } z = e^{j\omega}$$

$$X(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \frac{1 - e^{-j\omega N}}{1 - e^{j\frac{2\pi}{N}k} e^{-j\omega}} = \sum_{k=0}^{N-1} X(k) \varphi_k(\omega)$$

$$\varphi_k(\omega) = \frac{1}{N} \frac{e^{-j(\omega N - 2k\pi)/2} (e^{j(\omega N - 2k\pi)/2} - e^{-j(\omega N - 2k\pi)/2})}{e^{-j(\omega - \frac{2\pi}{N}k)/2} (e^{j(\omega - \frac{2\pi}{N}k)/2} - e^{-j(\omega - \frac{2\pi}{N}k)/2})}$$

$$= \frac{1}{N} \frac{\sin \left[N(\omega - \frac{2\pi}{N}k) / 2 \right]}{\sin \left[(\omega - \frac{2\pi}{N}k) / 2 \right]} e^{-j(\omega - \frac{2\pi}{N}k)(N-1)/2}$$

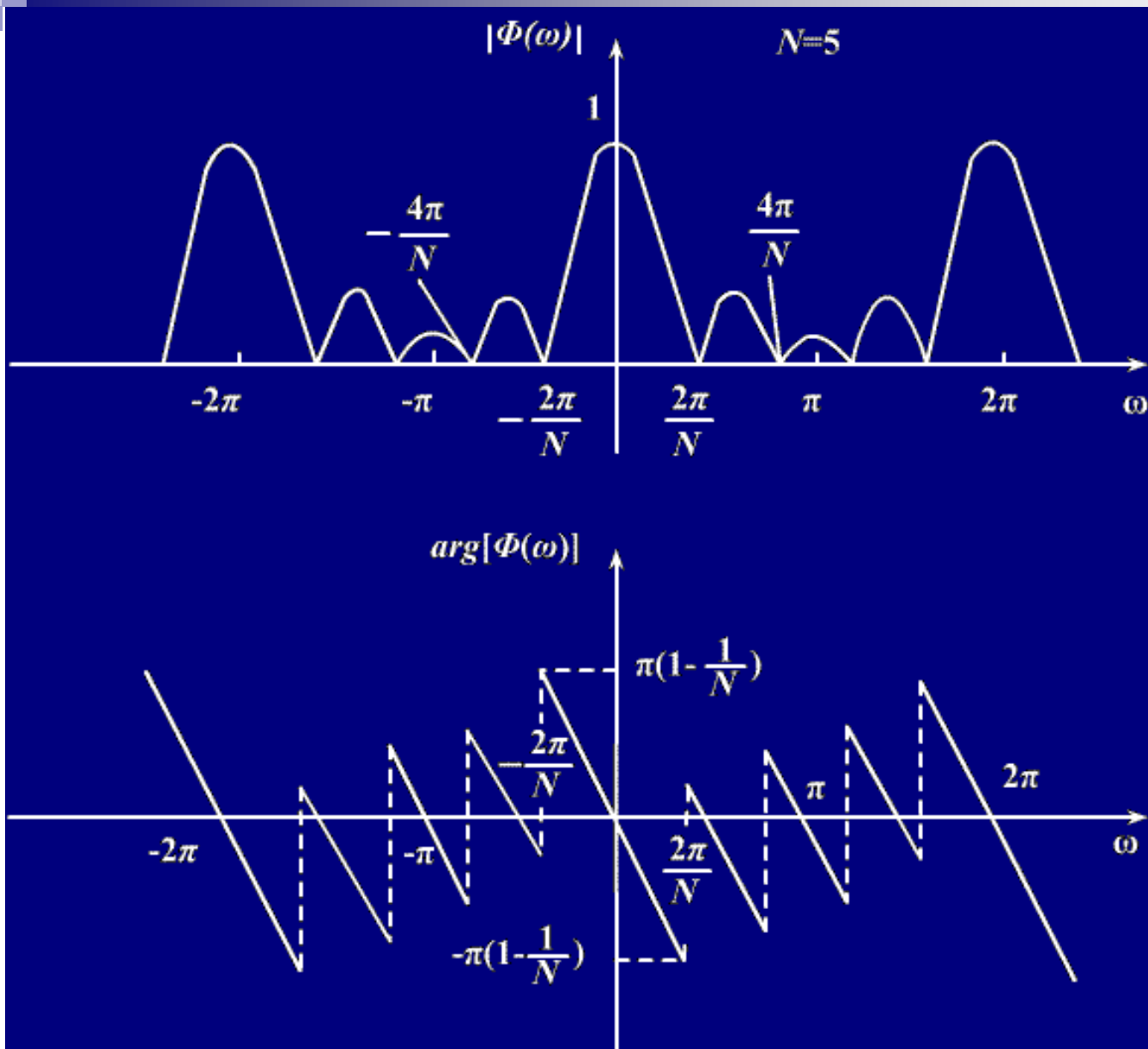
$$\text{令 } \varphi(\omega) = \frac{1}{N} \frac{\sin \frac{\omega N}{2}}{\sin \frac{\omega}{2}} e^{-j\omega(N-1)/2}$$

$$\therefore \varphi_k(\omega) = \varphi(\omega - \frac{2\pi}{N}k)$$

- 用 $X(k)$ 表示 $X(e^{j\omega})$ 的内插公式和内插函数

$$X(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \varphi\left(\omega - \frac{2\pi}{N}k\right)$$

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



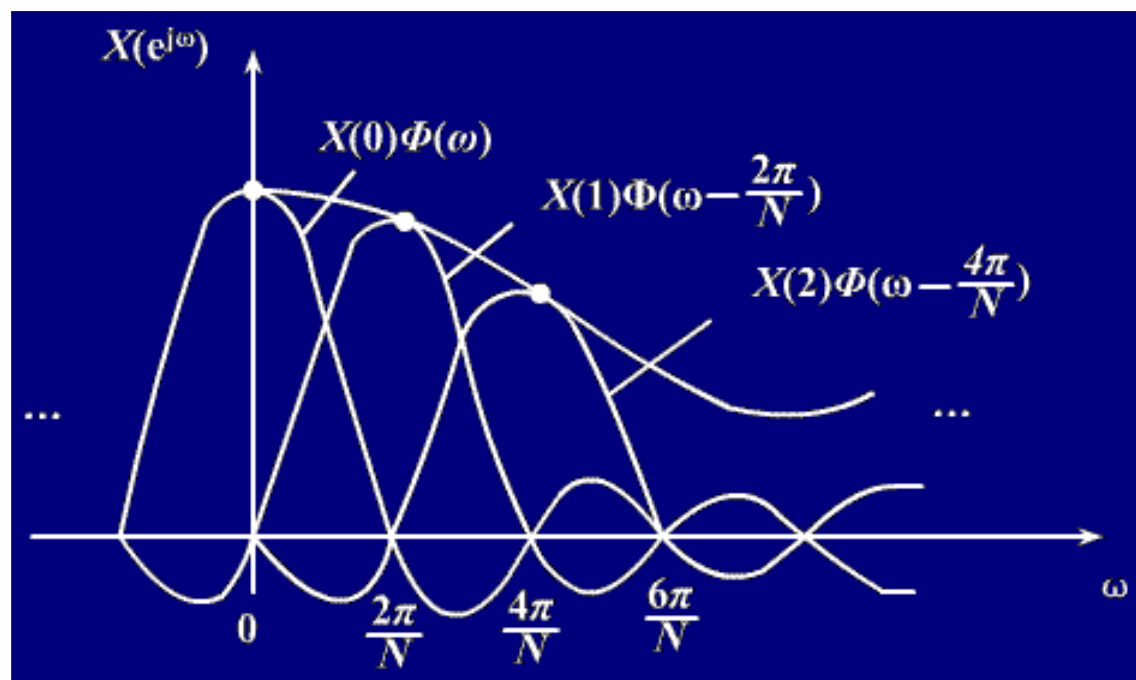
$$\varphi(\omega) = \frac{1}{N} \frac{\sin \frac{\omega N}{2}}{\sin \frac{\omega}{2}} e^{-j\omega(N-1)/2}$$

图3-13 插值函数 $\Phi(\omega)$ 的幅度特性与相位特性 ($N=5$)

内插公式：
$$X(e^{j\omega}) = \sum_{k=0}^{N-1} X(k) \varphi(\omega - \frac{2\pi}{N}k)$$

$$\varphi(\omega - \frac{2\pi}{N}k) = \begin{cases} 1 & \omega = \frac{2\pi}{N}k = \omega_k \\ 0 & \omega = \frac{2\pi}{N}i = \omega_i \quad i \neq k \end{cases}$$

- 保证了各采样点上的值与原序列的频谱相同；
- 采样点之间，为采样值与对应点的内插公式相乘，并叠加而成。



小结

■ 频域采样定理

$$X(e^{j\omega}) \longrightarrow X(k)$$

□ 条件，以保证恢复原序列 $\mathbf{x(n)}$

■ 频域的内插公式

$$X(k) \longrightarrow X(e^{j\omega})$$



3.4 DFT的快速算法—— 快速傅里叶变换 (FFT)

3.4.1问题的提出

4点序列{2, 3, 3, 2} DFT的计算复杂度

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn}, \quad k = 0, 1, \dots, N-1$$

$$X(0) = 2W_N^0 + 3W_N^0 + 3W_N^0 + 2W_N^0 = 10$$

$$X(1) = 2W_N^0 + 3W_N^1 + 3W_N^2 + 2W_N^3 = -1 - j$$

$$X(2) = 2W_N^0 + 3W_N^2 + 3W_N^4 + 2W_N^6 = 0$$

$$X(3) = 2W_N^0 + 3W_N^3 + 3W_N^6 + 2W_N^9 = -1 + j$$

复数加法 $N(N-1)$

复数乘法 N^2

1048576 (N=1024)

如何提高DFT的运算效率?

解决问题的思路

1. 将长序列DFT分解为短序列的DFT
2. 利用旋转因子 W_N^m 的周期性、对称性，
减少重复运算

旋转因子 W_N^m 的性质

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

1) 周期性

$$W_N^{m+lN} = e^{-j\frac{2\pi}{N}(m+lN)} = e^{-j\frac{2\pi}{N}m} = W_N^m$$

2) 对称性

$$(W_N^{N-m})^* = W_N^m$$

$$W_N^{m+\frac{N}{2}} = -W_N^m$$

3) 可约性

$$W_N^m = W_{N/n}^{m/n}, \quad N/n, m/n \text{ 为整数}$$

如何巧妙地利用周期性和对称性

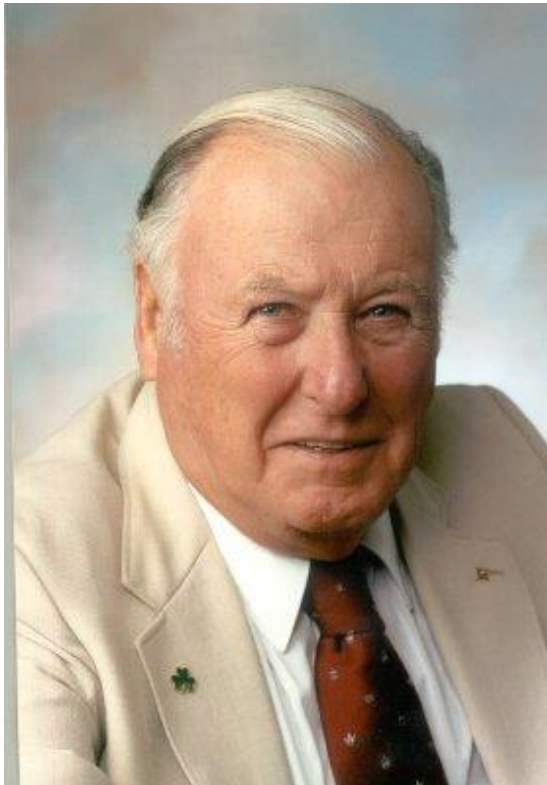
3.4.2 基2FFT算法 (1965)

Cooley–Tukey FFT algorithm

An Algorithm for the Machine Calculation of Complex Fourier Series

James W. Cooley, John W. Tukey

Mathematics of Computation, Volume 19, Issue 90 (Apr., 1965), 297-301.



James William Cooley
(1926 - 2016)



John Wilder Tukey
(1915 - 2000)

An Algorithm for the Machine Calculation of Complex Fourier Series

By James W. Cooley and John W. Tukey

An efficient method for the calculation of the interactions of a 2^m factorial experiment was introduced by Yates and is widely known by his name. The generalization to 3^m was given by Box et al. [1]. Good [2] generalized these methods and gave elegant algorithms for which one class of applications is the calculation of Fourier series. In their full generality, Good's methods are applicable to certain problems in which one must multiply an N -vector by an $N \times N$ matrix which can be factored into m sparse matrices, where m is proportional to $\log N$. This results in a procedure requiring a number of operations proportional to $N \log N$ rather than N^2 . These methods are applied here to the calculation of complex Fourier series. They are useful in situations where the number of data points is, or can be chosen to be, a highly composite number. The algorithm is here derived and presented in a rather different form. Attention is given to the choice of N . It is also shown how special advantage can be obtained in the use of a binary computer with $N = 2^m$ and how the entire calculation can be performed within the array of N data storage locations used for the given Fourier coefficients.

Consider the problem of calculating the complex Fourier series

$$(1) \quad X(j) = \sum_{k=0}^{N-1} A(k) \cdot W^{jk}, \quad j = 0, 1, \dots, N-1,$$

where the given Fourier coefficients $A(k)$ are complex and W is the principal N th root of unity,

$$(2) \quad W = e^{2\pi i/N}.$$

A straightforward calculation using (1) would require N^2 operations where "operation" means, as it will throughout this note, a complex multiplication followed by a complex addition.

The algorithm described here iterates on the array of given complex Fourier amplitudes and yields the result in less than $2N \log_2 N$ operations without requiring more data storage than is required for the given array A . To derive the algorithm, suppose N is a composite, i.e., $N = r_1 \cdot r_2$. Then let the indices in (1) be expressed

$$(3) \quad \begin{aligned} j &= j_1 r_1 + j_0, & j_0 &= 0, 1, \dots, r_1 - 1, & j_1 &= 0, 1, \dots, r_2 - 1, \\ k &= k_1 r_2 + k_0, & k_0 &= 0, 1, \dots, r_2 - 1, & k_1 &= 0, 1, \dots, r_1 - 1. \end{aligned}$$

Then, one can write

$$(4) \quad X(j_1, j_0) = \sum_{k_2} \sum_{k_1} A(k_1, k_0) \cdot W^{j_1 r_1 k_1} W^{j_0 k_0}.$$

Received August 17, 1964. Research in part at Princeton University under the sponsorship of the Army Research Office (Durham). The authors wish to thank Richard Garwin for his essential role in communication and encouragement.

297

$N^2 \rightarrow N/2 \cdot \log_2 N$
 $1048576 \rightarrow 5120$
(eg: 204.8倍, $N=1024$)

3.4.2 基2FFT算法（1965） Cooley–Tukey FFT algorithm

$N=2^M$ ， M 为自然数

将时域序列逐次分解为一组子序列，利用旋转因子的特性，由子序列的DFT来实现整个序列的DFT。

✓ 基2时间抽取(Decimation in time)DIT-FFT算法

$$x(n) \rightarrow \begin{cases} x(2r) \\ x(2r+1) \end{cases} \quad r = 0, 1, \dots, \frac{N}{2} - 1$$

✓ 基2频率抽取(Decimation in frequency)DIF-FFT算法

$$X(k) \rightarrow \begin{cases} X(2m) \\ X(2m+1) \end{cases} \quad m = 0, 1, \dots, \frac{N}{2} - 1$$

DIT-FFT算法

- 序列 $x(n)$ 的 N 点DFT为 $X(k) = DFT[x(n)]_N = \sum_{n=0}^{N-1} x(n)W_N^{kn}$

- 奇偶分解

$$\begin{aligned} X(k) &= \sum_{n \text{ 为偶数}} x(n)W_N^{kn} + \sum_{n \text{ 为奇数}} x(n)W_N^{kn} = \\ &= \sum_{l=0}^{N/2-1} x(2l)W_N^{k2l} + \sum_{l=0}^{N/2-1} x(2l+1)W_N^{k(2l+1)} \end{aligned}$$

$$x_1(l) = x(2l), x_2(l) = x(2l+1), W_N^{2kl} = W_{N/2}^{kl} \quad N \text{ 为 } 2 \text{ 的整数倍}$$

$$X(k) = \sum_{l=0}^{N/2-1} x_1(l)W_{N/2}^{kl} + W_N^k \sum_{l=0}^{N/2-1} x_2(l)W_{N/2}^{kl}$$

$$X(k) = \sum_{l=0}^{N/2-1} x_1(l)W_{N/2}^{kl} + W_N^k \sum_{l=0}^{N/2-1} x_2(l)W_{N/2}^{kl}$$

■ 分解为2个DFT

$$X_1(k) = \sum_{l=0}^{N/2-1} x_1(l)W_{N/2}^{kl}, \quad X_2(k) = \sum_{l=0}^{N/2-1} x_2(l)W_{N/2}^{kl}$$

$$k = 0, 1, 2, \dots, N/2 - 1$$

■ 均以N/2为周期

■ 利用 $W_N^{m+\frac{N}{2}} = -W_N^m$ 及DFT的隐含周期性，得

$$X(k) = X_1(k) + W_N^k X_2(k) \quad k = 0, 1, 2, \dots, \frac{N}{2} - 1$$

$$X(k + \frac{N}{2}) = X_1(k + \frac{N}{2}) + W_N^{k+\frac{N}{2}} X_2(k + \frac{N}{2}) = X_1(k) - W_N^k X_2(k)$$

$$\left. \begin{aligned} X(k) &= X_1(k) + W_N^k X_2(k) \\ X(k + \frac{N}{2}) &= X_1(k) - W_N^k X_2(k) \end{aligned} \right\} \quad k = 0, 1, 2, \dots, \frac{N}{2} - 1$$

$$X_1(k) = \sum_{l=0}^{N/2-1} x_1(l) W_{N/2}^{kl}, \quad X_2(k) = \sum_{l=0}^{N/2-1} x_2(l) W_{N/2}^{kl}$$

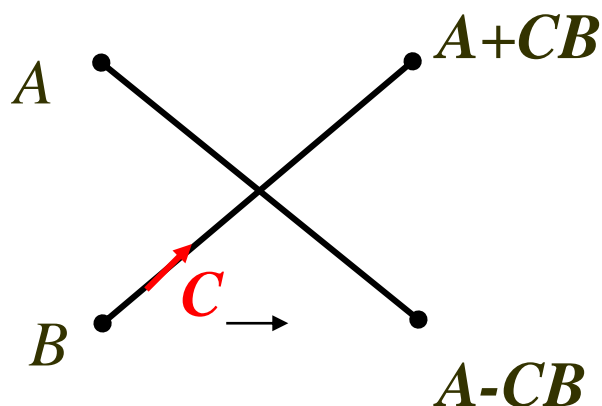
■ 运算量:

复数乘 $\frac{N}{2} \times \frac{N}{2} \times 2 + \frac{N}{2} = \frac{N}{2}(N+1) \approx \frac{N^2}{2} \quad N \gg 1$

复数加 $\frac{N}{2} \times (\frac{N}{2} - 1) \times 2 + \frac{N}{2} \times 2 = \frac{N^2}{2}$

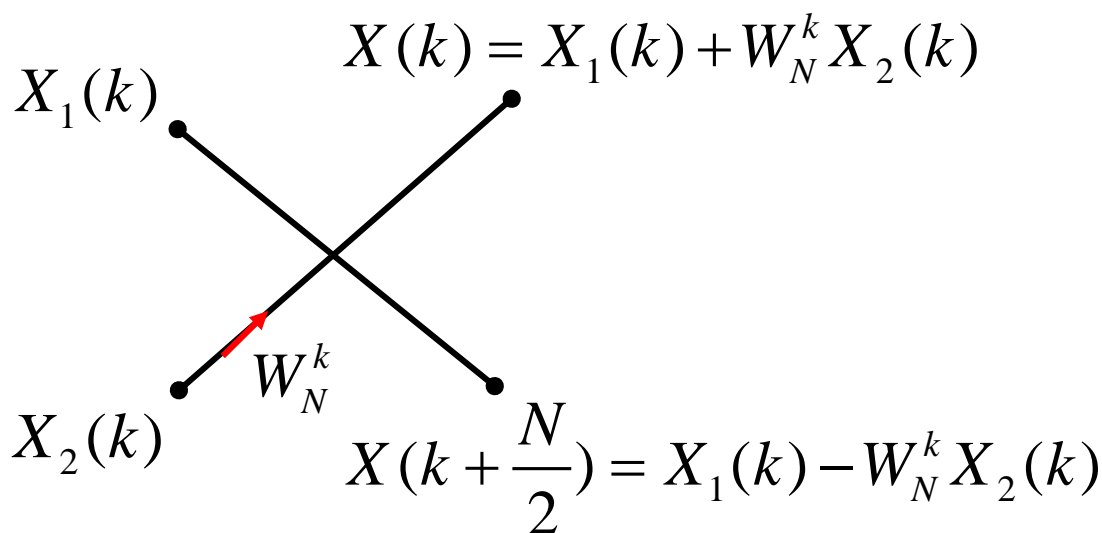
蝶形图及运算功能

$$\left. \begin{aligned} X(k) &= X_1(k) + W_N^k X_2(k) \\ X(k + \frac{N}{2}) &= X_1(k) - W_N^k X_2(k) \end{aligned} \right\} k = 0, 1, 2, \dots, \frac{N}{2} - 1$$



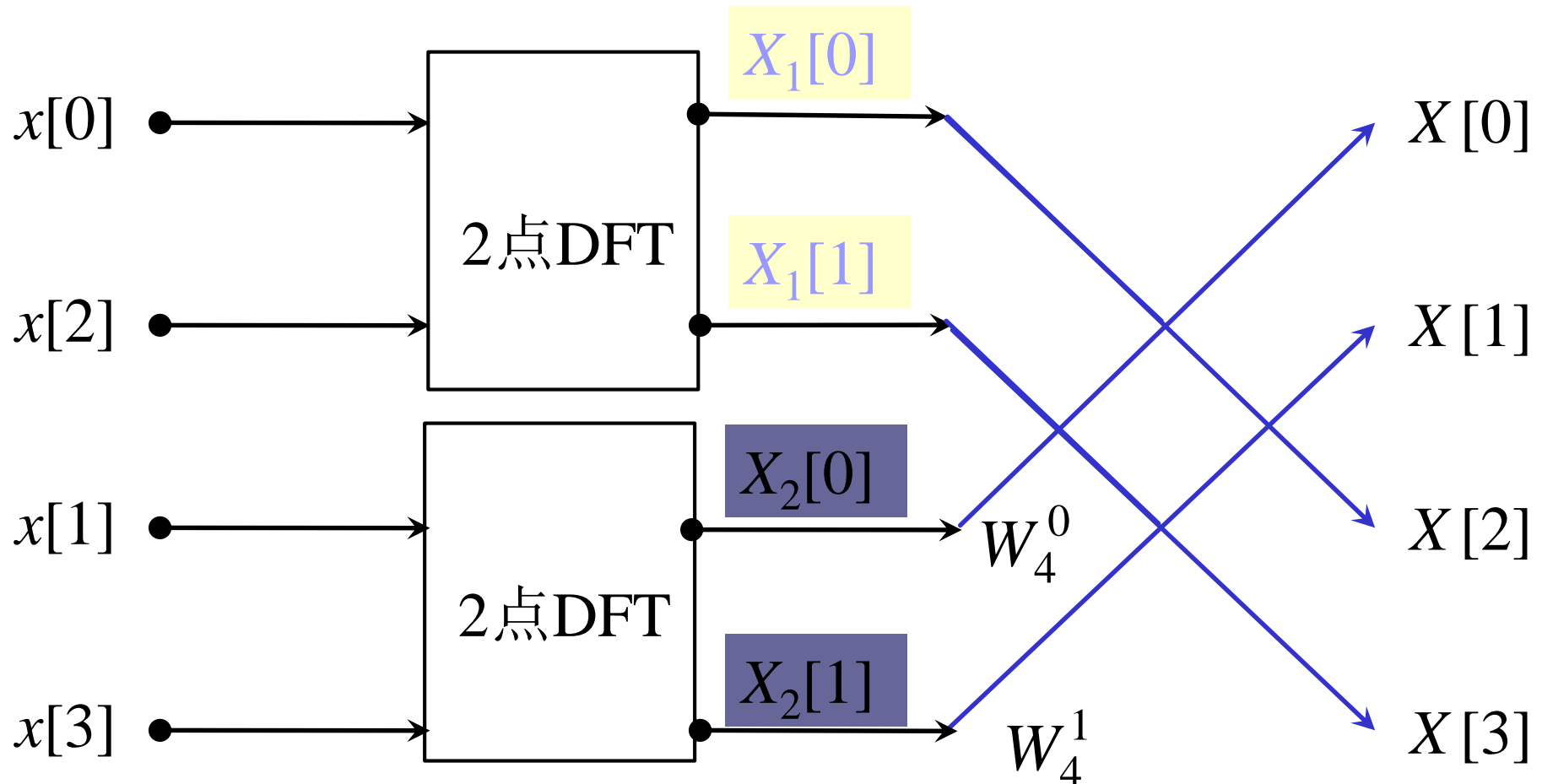
1次乘法

2次加法

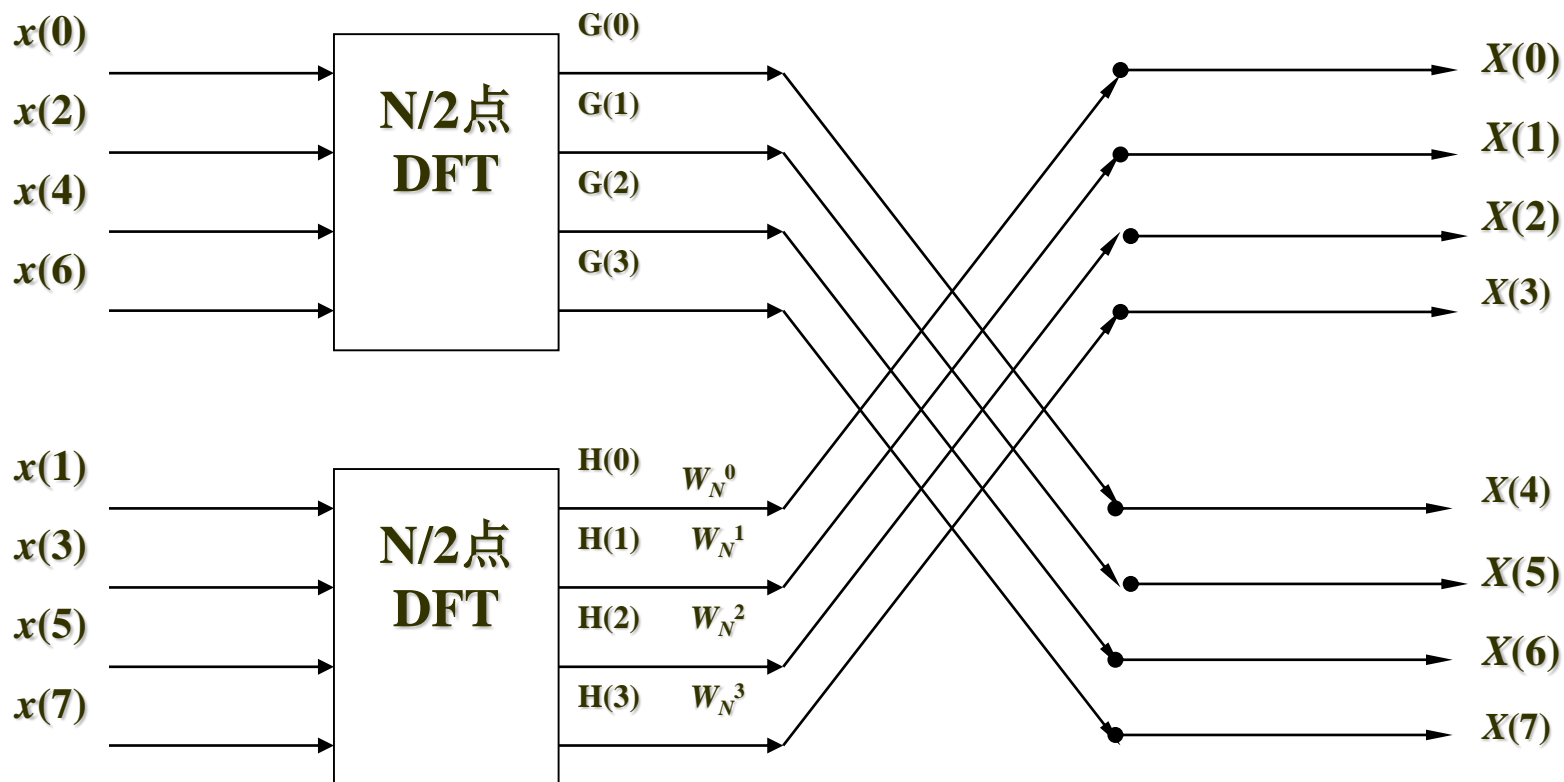


$$X(k) = X_1(k) + W_4^k X_2(k), \quad k = 0, 1$$

$$X(k+2) = X_1(k) - W_4^k X_2(k), \quad k = 0, 1$$

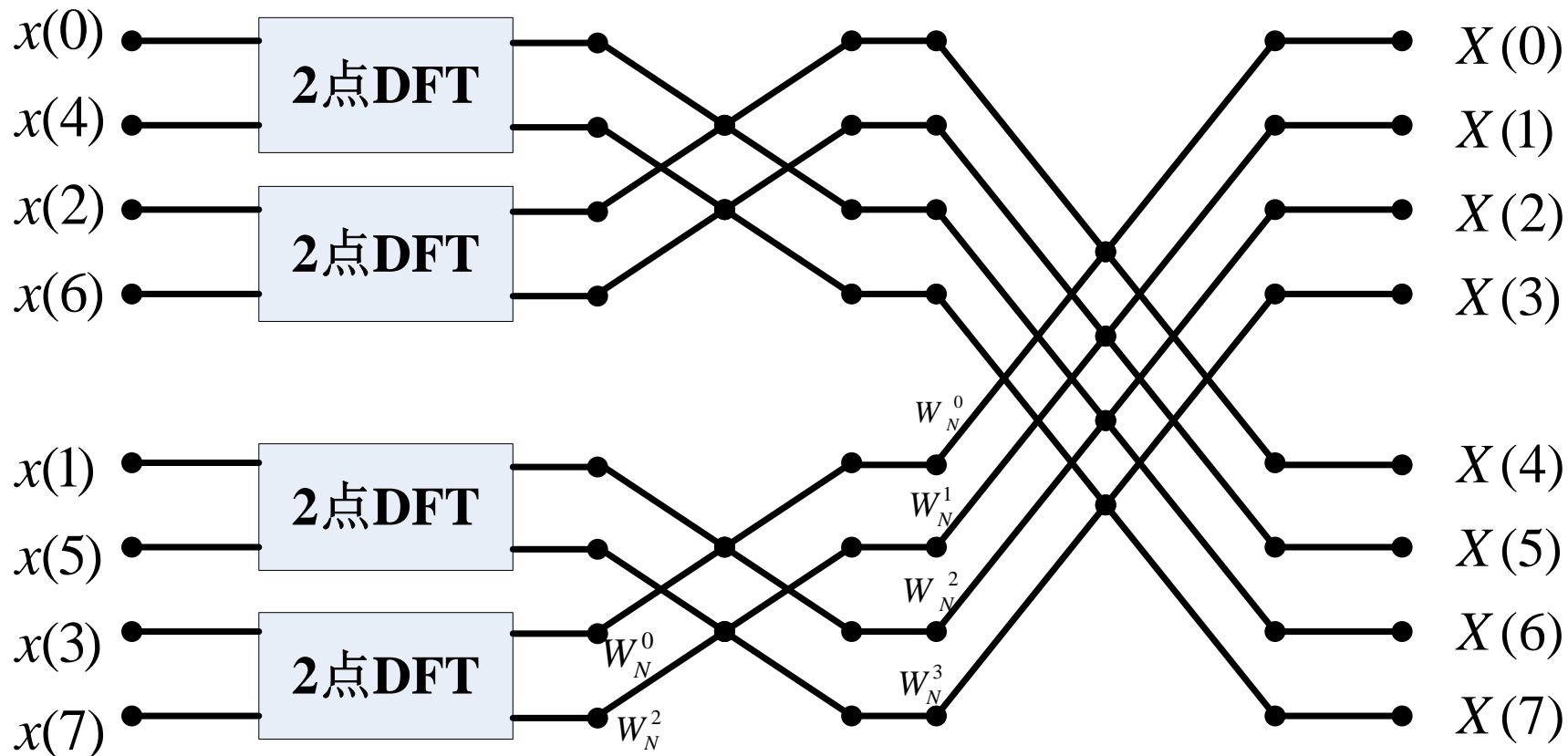


8点DFT一次时域抽取分解运算流图



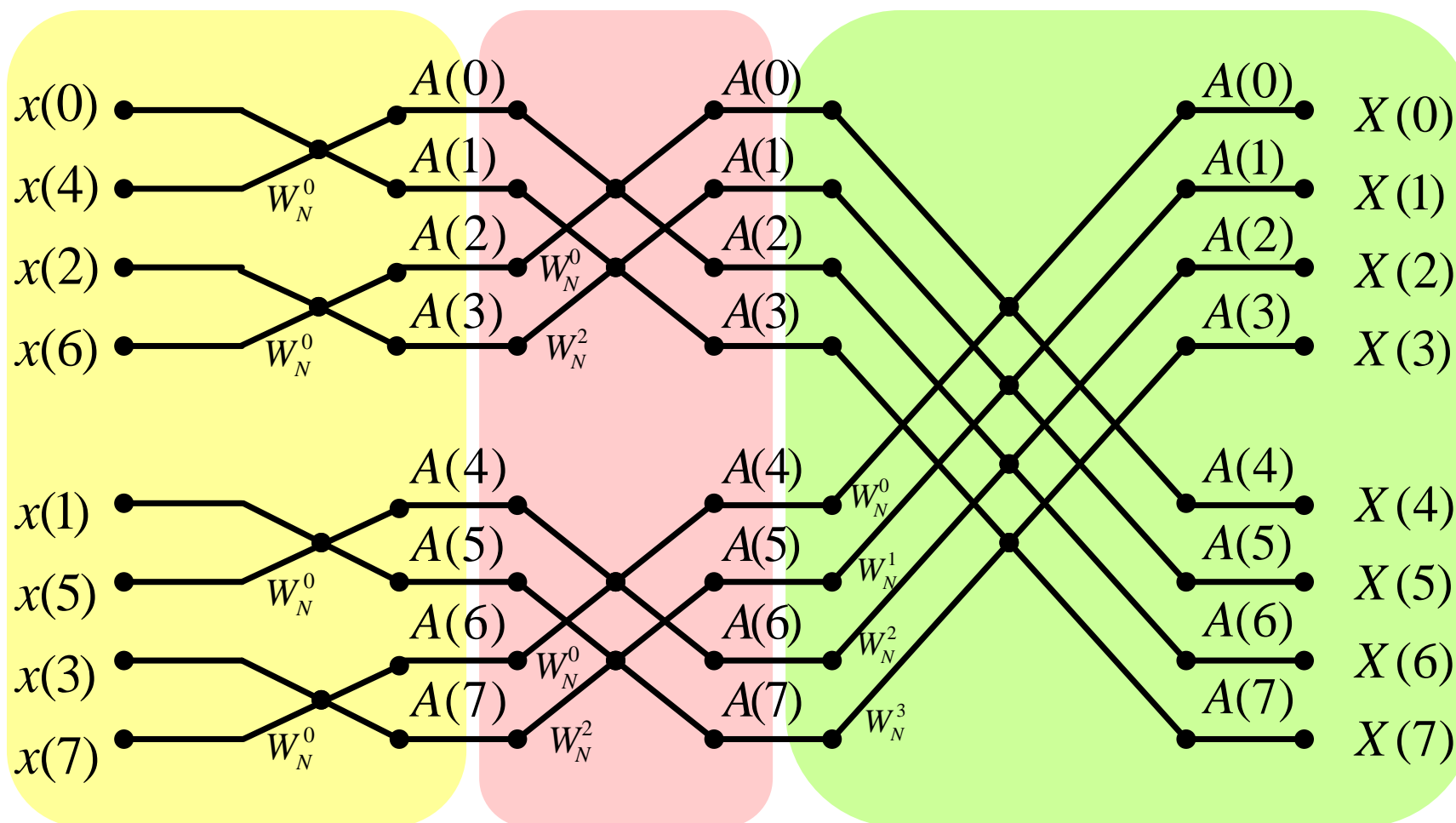
两个4点DFT组成8点DFT

8点DFT两次时域抽取分解运算流图



由四个2点DFT组成8点DFT

8点DIT-FFT运算流图 (N=8, M=3)



经过M级时域奇偶抽取，可分解为N个1点DFT（即时域序列本身）和M级蝶形运算，每一级有N/2个蝶形。

2. DIT-FFT的运算效率

- $N=2^M$ 的序列，通过 M 级分解最后成为1点的DFT运算。构成 M 级运算过程。
- 每一级运算都由 $N/2$ 个蝶形运算构成。每一级运算含 $N/2$ 次复乘和 N 次复加。
- 总运算量：

□ 复乘 $\frac{N}{2} \bullet M = \frac{N}{2} \log_2 N \quad \longleftarrow N^2$

□ 复加 $N \bullet M = N \log_2 N \quad \longleftarrow N(N-1)$

运算效率

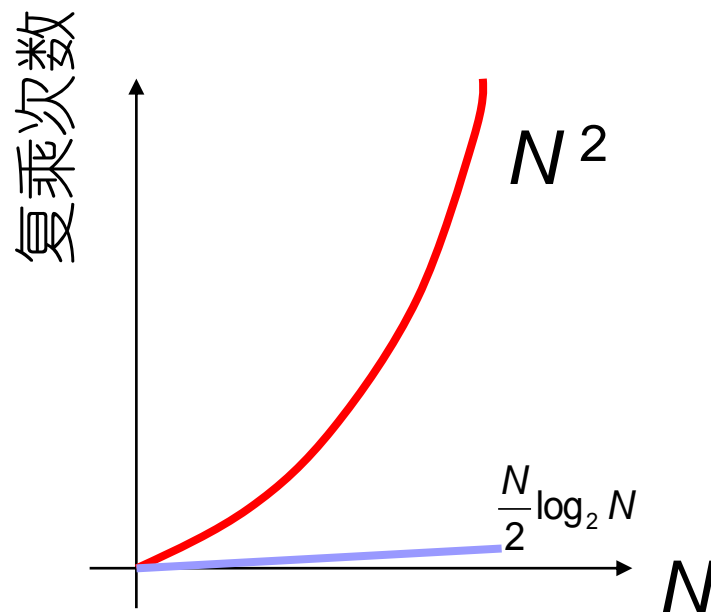
$$\frac{DFT\text{的乘法次数}}{DIT-FFT\text{的乘法次数}} = \frac{N^2}{C_M(2)} = \frac{N^2}{\frac{N}{2} \log_2 N} = \frac{2N}{M}$$

$$N = 2^{10} = 1024$$

■ 运算效率为**204.8**

$$N = 2^{11} = 2048$$

■ 运算效率为**372.37**

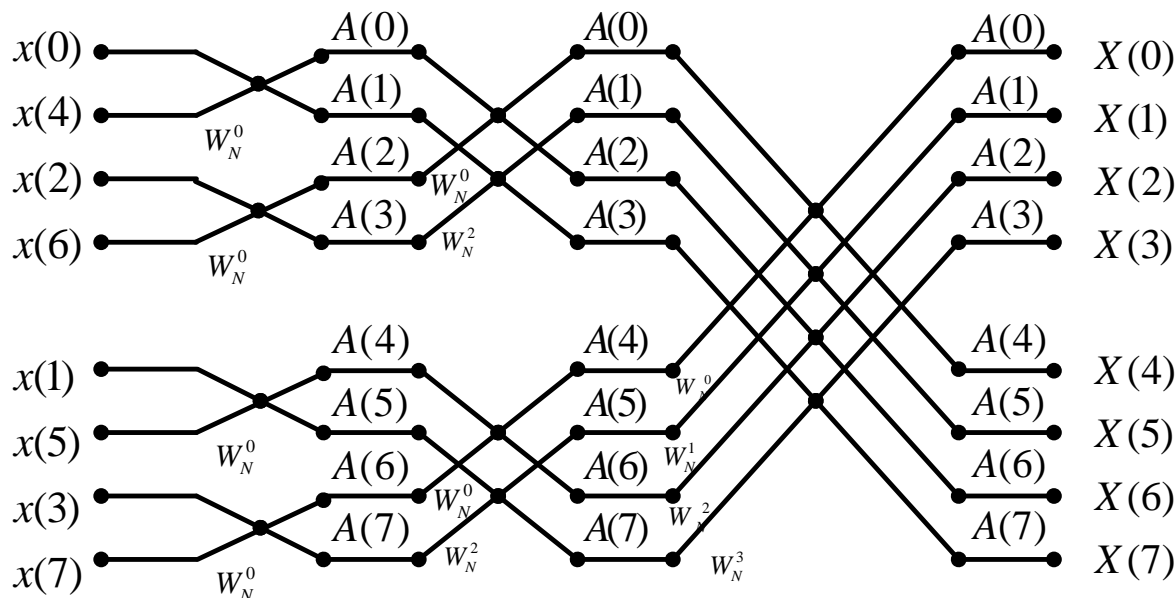


3. DIT-FFT运算规律及编程思想

1) 原位计算

- 观察每个蝶形的两个输入和两个输出
- 蝶形的输出可存入原输入数据所占存储单元
- 利用同一组存储单元存储输入、输出数据的方法，称为原位（址）计算。

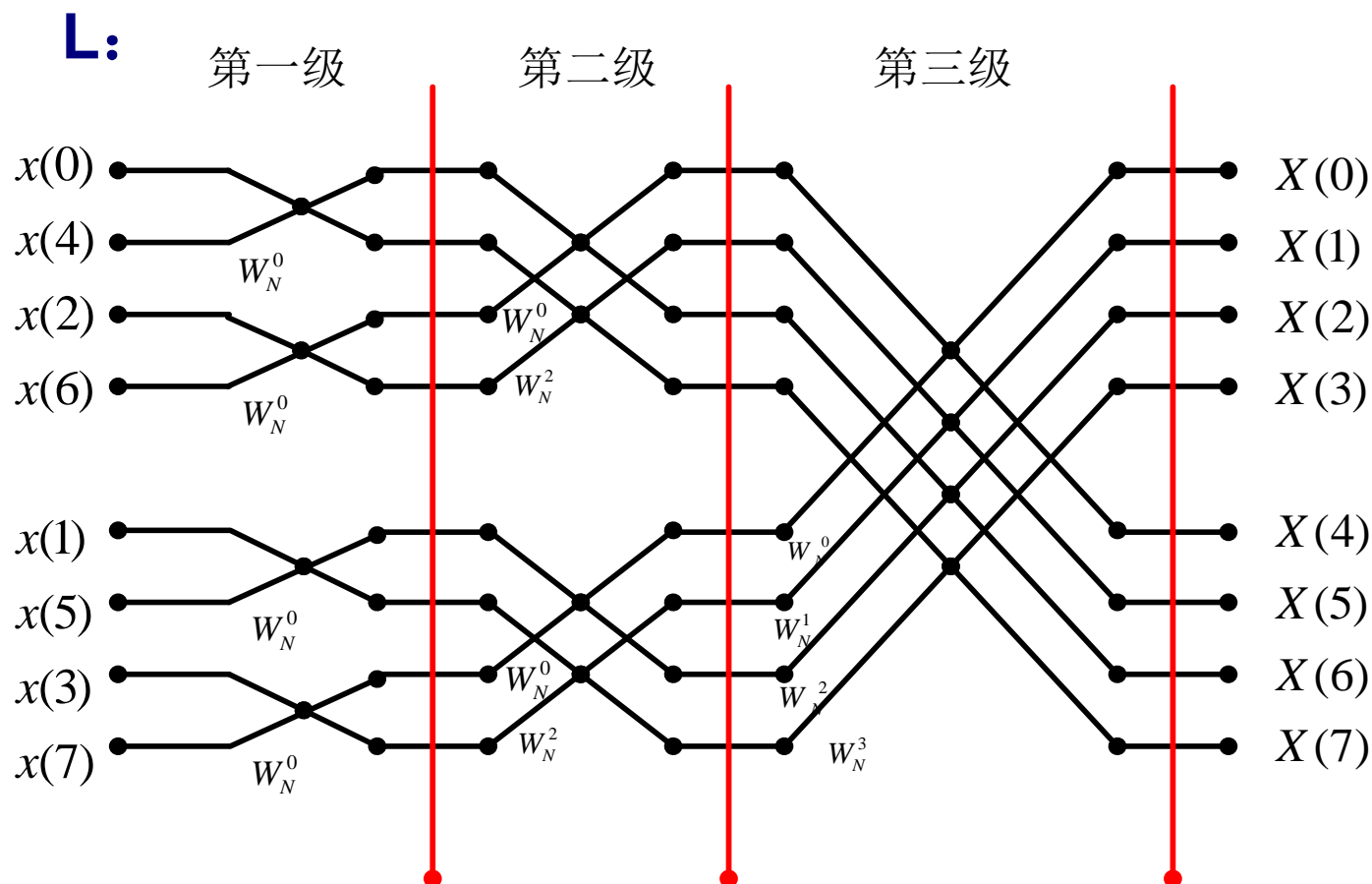
- 节约内存
- 节省寻址的时间
- 降低成本



2) 旋转因子的变化规律

■ 旋转因子 W_N^p 旋转因子指数 p

W_N^p 与运算级数 L 的关系

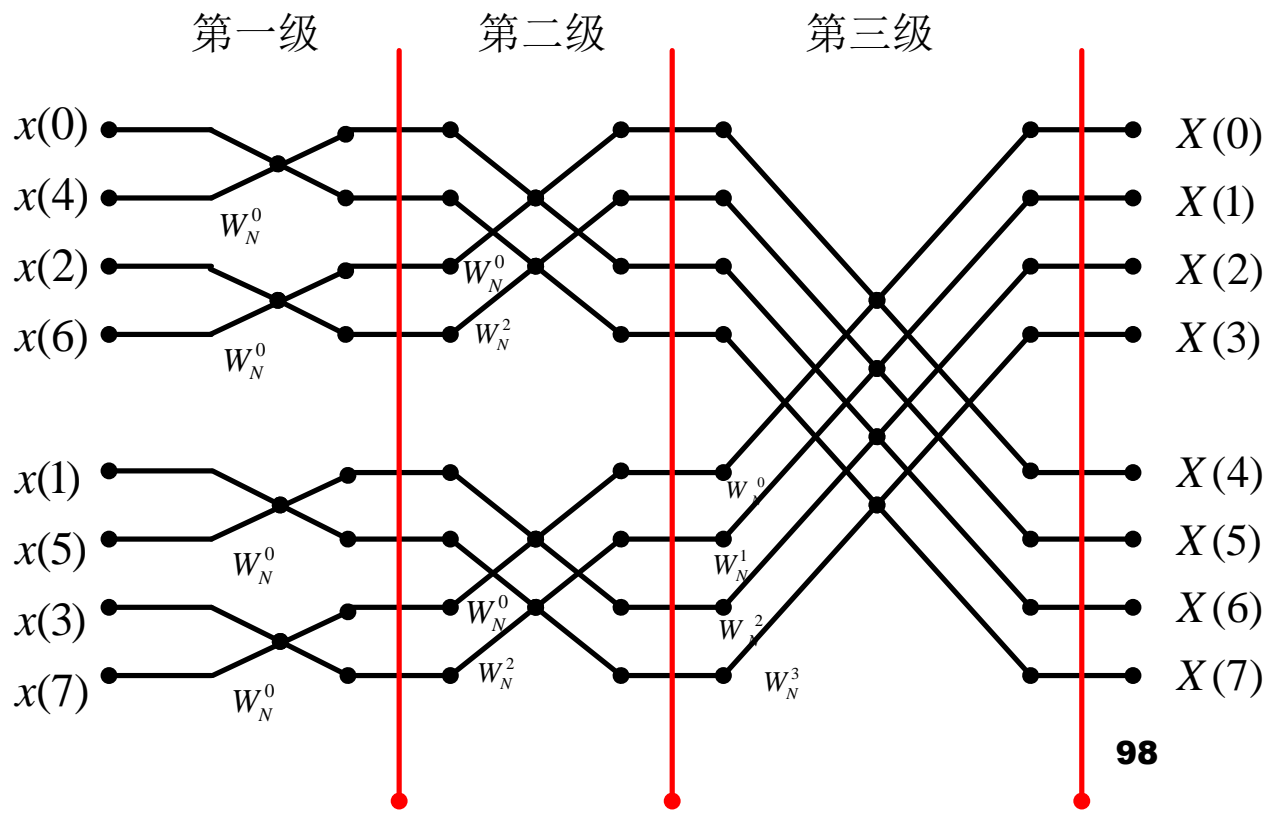


2) 旋转因子的变化规律 (2)

$L = 3$ 时, $W_N^P = W_N^J = W_{2^L}^J, J = 0, 1, 2, 3$

$L = 2$ 时, $W_N^P = W_{N/2}^J = W_{2^L}^J = W_N^{2J}, J = 0, 1$

$L = 1$ 时, $W_N^P = W_{N/4}^J = W_{2^L}^J = W_N^{4J}, J = 0$



一般情况

$N = 2^M$, 第L级的旋转因子为

$$W_N^p = W_{2^L}^J, \quad J = 0, 1, 2, \dots, 2^{L-1} - 1$$

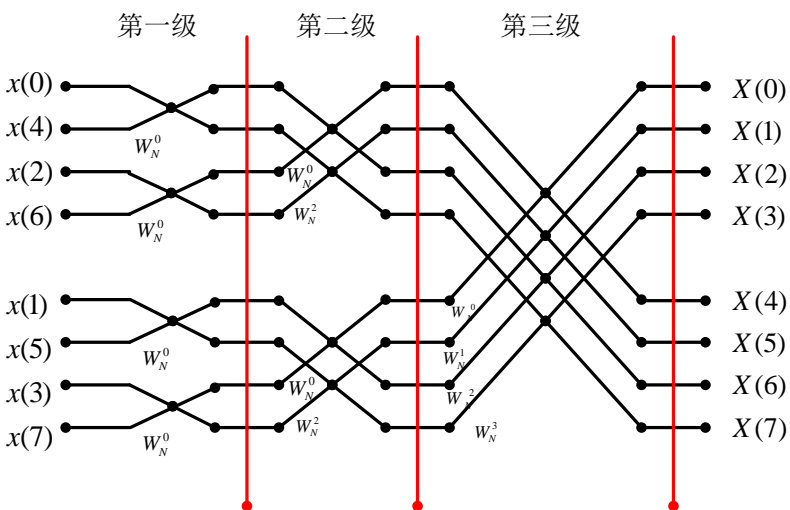
$$\because 2^L = 2^M \times 2^{L-M} = N \cdot 2^{L-M}$$

$$\therefore W_N^p = W_{N \cdot 2^{L-M}}^J = W_N^{J \cdot 2^{M-L}}, \quad J = 0, 1, 2, \dots, 2^{L-1} - 1$$

$$\therefore p = J \cdot 2^{M-L}$$

3)序列的倒序

- $X(0), X(1), X(2), \dots, X(7)$ 按顺序输出;
- $x(0), x(1), x(2), \dots, x(7)$ 则不按顺序排列;



顺		序		倒		序	
十进制数		二进制数		二进制数		十进制数	
I						J	
0		0	0	0	0	0	0
1		0	0	1	1	0	4
2		0	1	0	0	1	2
3		0	1	1	1	0	6
4		1	0	0	0	1	1
5		1	0	1	1	0	5
6		1	1	0	0	1	3
7		1	1	1	1	1	7

- I与J的关系?

4) 蝶形运算规律

- $x(0), x(1), x(2), \dots, x(N-1)$ 倒序;
- 如蝶形运算的两个输入数据相距 **B** 个点, 应用原位计算, 蝶形运算如下:

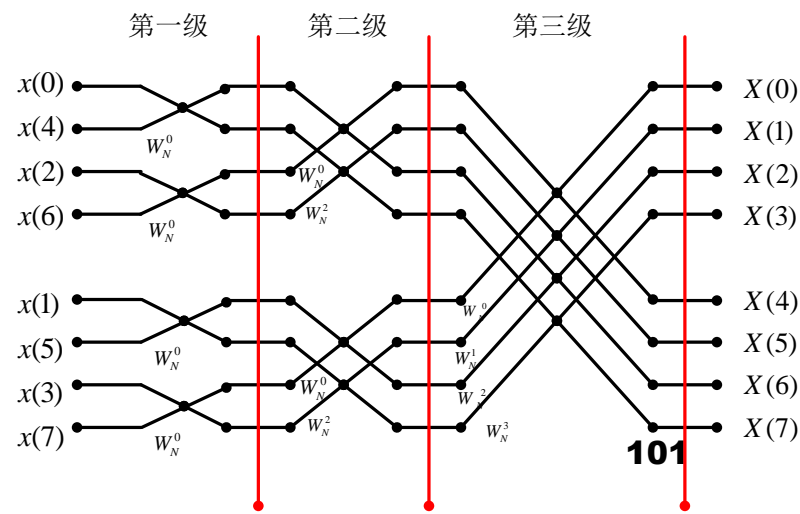
$$A_L(J) \Leftarrow A_{L-1}(J) + A_{L-1}(J+B)W_N^p$$

$$A_L(J+B) \Leftarrow A_{L-1}(J) - A_{L-1}(J+B)W_N^p$$

$$p = J \times 2^{M-L}; J = 0, 1, 2, \dots, 2^{L-1};$$

$$L = 1, 2, \dots, M$$

5) 程序框图



4. IDFT的高效算法 (1)

- **DFT**
$$X(k) = DFT[x(n)] = \sum_{n=0}^{N-1} x(n)W_N^{nk}$$

- **IDFT**
$$x(n) = IDFT[X(k)] = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-nk}$$

- 差别?

- 旋转因子 W_N^{kn}, W_N^{-kn} 和系数

- 把**DFT**中的每一个系数 W_N^{kn} 改为 W_N^{-kn}

- 再乘以常数 **1/N**

4. IDFT的高效算法（2）

■ 用FFT子程序计算IDFT

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk}$$

$$x^*(n) = \frac{1}{N} \sum_{k=0}^{N-1} X^*(k) W_N^{nk}$$

$$x(n) = \left[\frac{1}{N} \sum_{k=0}^{N-1} X^*(k) W_N^{nk} \right]^* = \frac{1}{N} \left\{ DFT[X^*(k)] \right\}^*$$

IFFT计算分三步：

- ① 将**X(k)**取共轭（虚部乘以-1）
- ② 对 $X^*(k)$ 直接作**FFT**
- ③ 对**FFT**的结果取共轭并乘以**1/N**，得**x(n)**。



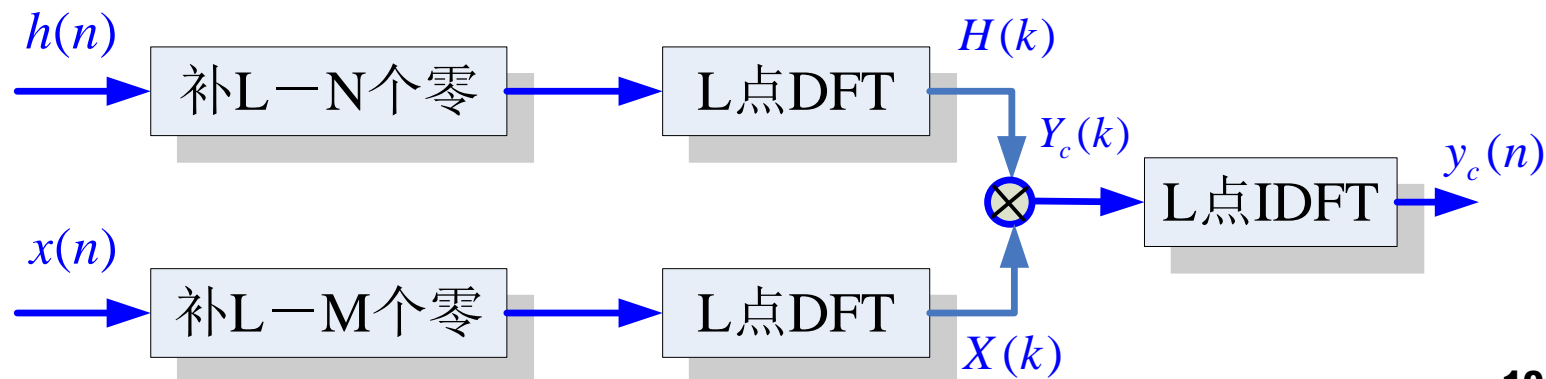
3.5 DFT (FFT) 应用举例

※ 线性卷积

※ 频谱分析

3.5.1 用DFT（FFT）计算两个有限长序列的线性卷积

- 求离散系统响应
- 直接计算时间较长
- 间接计算
 - DFT可计算循环卷积
 - FFT可加快运算速度
 - DFT能否计算线性卷积？
 - 如可以，条件？



循环卷积与线性卷积等价的条件

■ 循环卷积

$$y_c(n) = h(n) \bigcircled{\mathbb{L}} x(n) = \sum_{m=0}^{L-1} h(m)x((n-m))_L R_L(n)$$

$$L \geq \max[N, M]$$

■ 线性卷积

$$y(n) = h(n) * x(n) = \sum_{m=0}^{N-1} h(m)x(n-m)$$

$$y(n) = h(n) * x(n) = \sum_{m=0}^{N-1} h(m)x(n-m)$$

$$x((n-m))_L = \sum_{i=-\infty}^{\infty} x(n-m+iL)$$

$$y_c(n) = h(n) \circledast x(n) = \sum_{m=0}^{L-1} h(m) x((n-m))_L R_L(n)$$

$$= \sum_{m=0}^{L-1} h(m) \sum_{i=-\infty}^{\infty} x(n-m+iL) R_L(n)$$

$$= \sum_{i=-\infty}^{\infty} \left[\sum_{m=0}^{L-1} h(m) x(n-m+iL) \right] R_L(n)$$

$$= \sum_{i=-\infty}^{\infty} y(n+iL) R_L(n) \quad ? \quad y(n)$$

$y_c(n)$ 是 $y(n)$ 以 L 为周期的周期延拓序列的主值序列

$$L \geq N + M - 1$$

循环卷积计算线性卷积的运算量

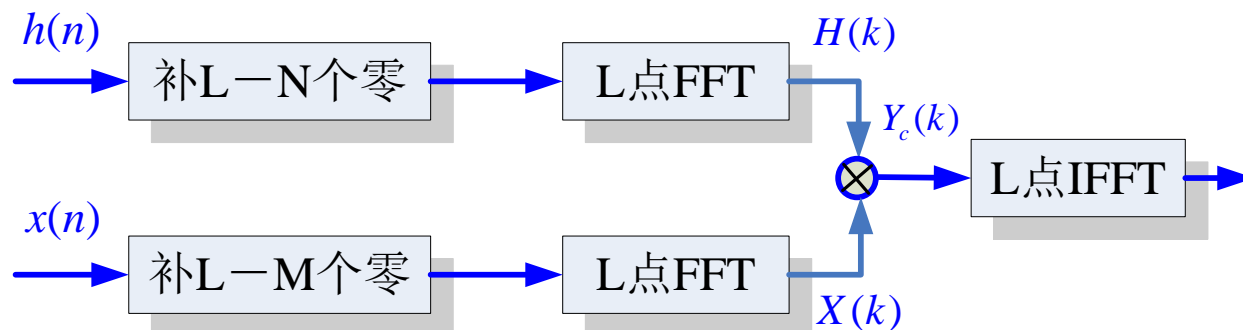
$$N = M \geq 32$$

- 可证明小于直接计算线性卷积的运算量
- $H(k) = DFT[h(n)]$ 可预先计算并存储，乘法的运算次数又可降低

$$0.5L \log_2 L \quad \text{次}$$

■ 快速卷积法

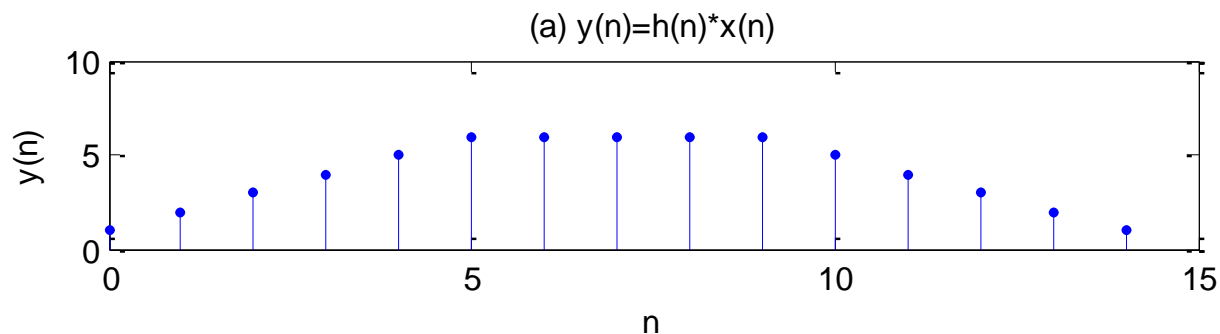
$$L \geq N + M - 1$$



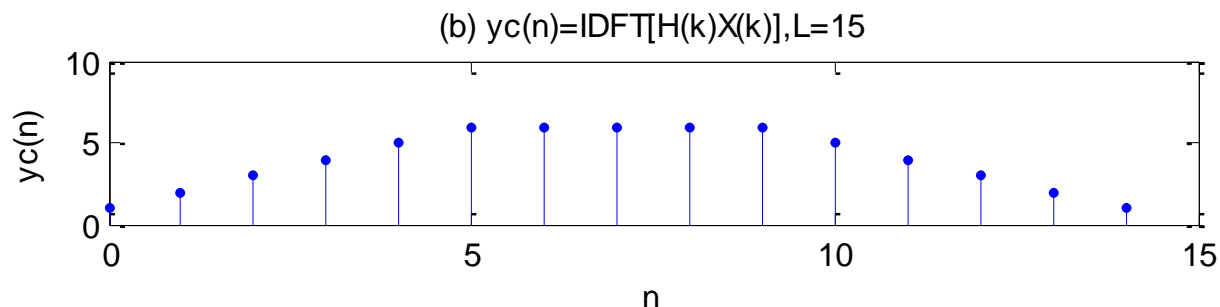
例3.5.1 $x(n) = R_{10}(n), h(n) = R_6(n)$

求: $y(n) = x(n) * h(n)$

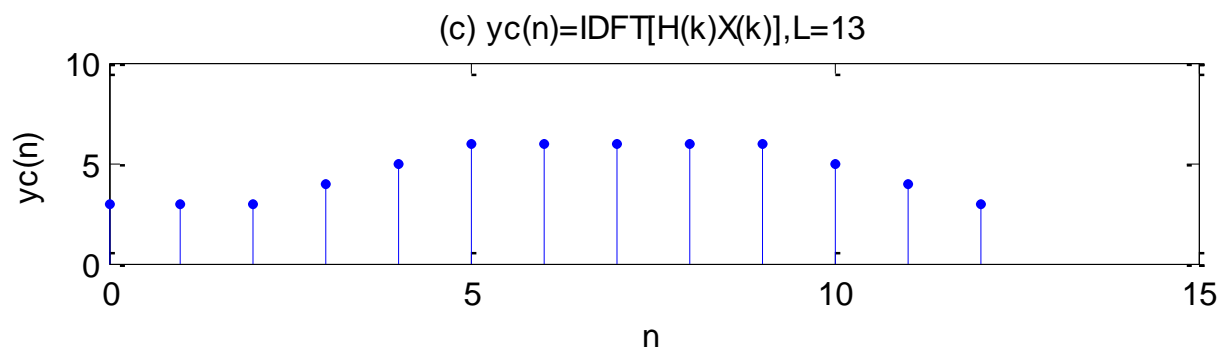
■ 直接计算
conv



■ FFT
L=15



■ FFT
L=13



3.5.2有限长序列和无限长序列的线性卷积

■ 问题:

- $h(n)$ 为某滤波器的单位脉冲响应,长度有限; 但输入信号 $x(n)$ 很长;
- 若存储完 $x(n)$ 再做卷积, 则存储量过大, 并且等待 $x(n)$ 输入的时间过长;
- $h(n)$ 要补许多零再进行计算, 计算量有很大的浪费;

■ 解决方法

- 将长序列分成短序列分别卷积, 再将各自的输出按照一定的规律首尾相加即可
- 重叠相加法
- 重叠保留法

1. 重叠相加法

$$x(n) = \sum_{i=0}^{\infty} x_i(n - iM)$$

- 分段：将 $x(n)$ 分段，每段长度为**M**

$$x_i(n) = x(n + iM)R_M(n)$$

- 各段与 $h(n)$ 卷积

$$y_i(n - iM) = h(n) * x_i(n - iM)$$

- 长度为**N+M-1**

- 求和

$$y(n) = h(n) * x(n) = \sum_{i=0}^{\infty} h(n) * x_i(n - iM) = \sum_{i=0}^{\infty} y_i(n - iM)$$

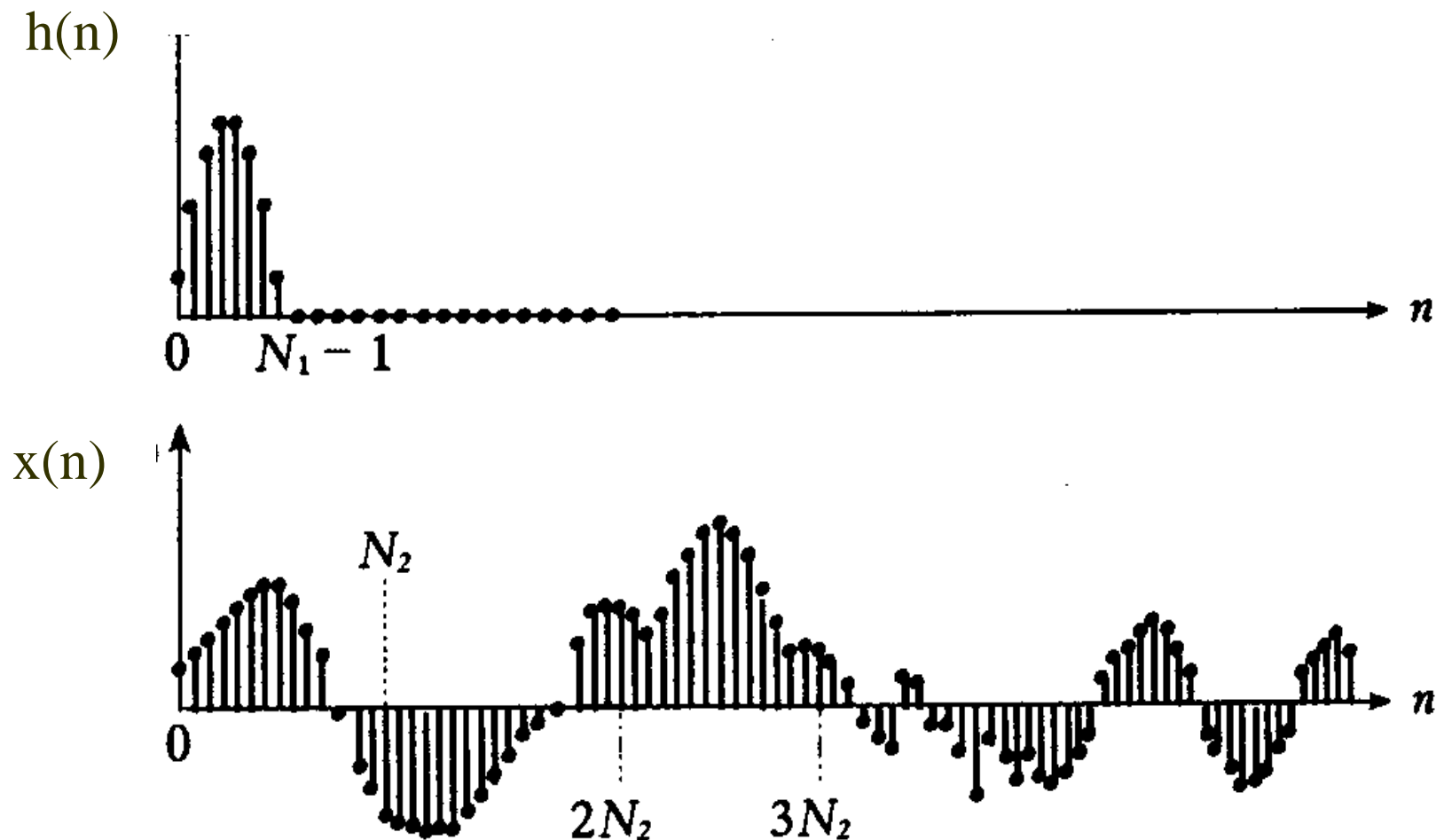
- $y_0(n), y_1(n), y_2(n), \dots$ 的定义区间 $0 \leq n \leq N + M - 2$
- $y_0(n - 0M)$ 的定义区间为 $0 \leq n \leq N + M - 2$
- $y_1(n - M)$ 的定义区间为 $M \leq n \leq 2M + N - 2$
- $y_2(n - 2M)$ 的定义区间为 $2M \leq n \leq N + 3M - 2$
- $y_i(n - iM)$ 的定义区间为 $iM \leq n \leq N + (i + 1)M - 2$
- 重叠区间

$$y_{i-1}(n - (i-1)M) \text{ 与 } y_i(n - iM)$$

$$\text{重叠区间} \quad iM \leq n \leq N + iM - 2$$

对应点相加

(1) 重叠相加法——由分段卷积的各段相加构成总的卷积输出



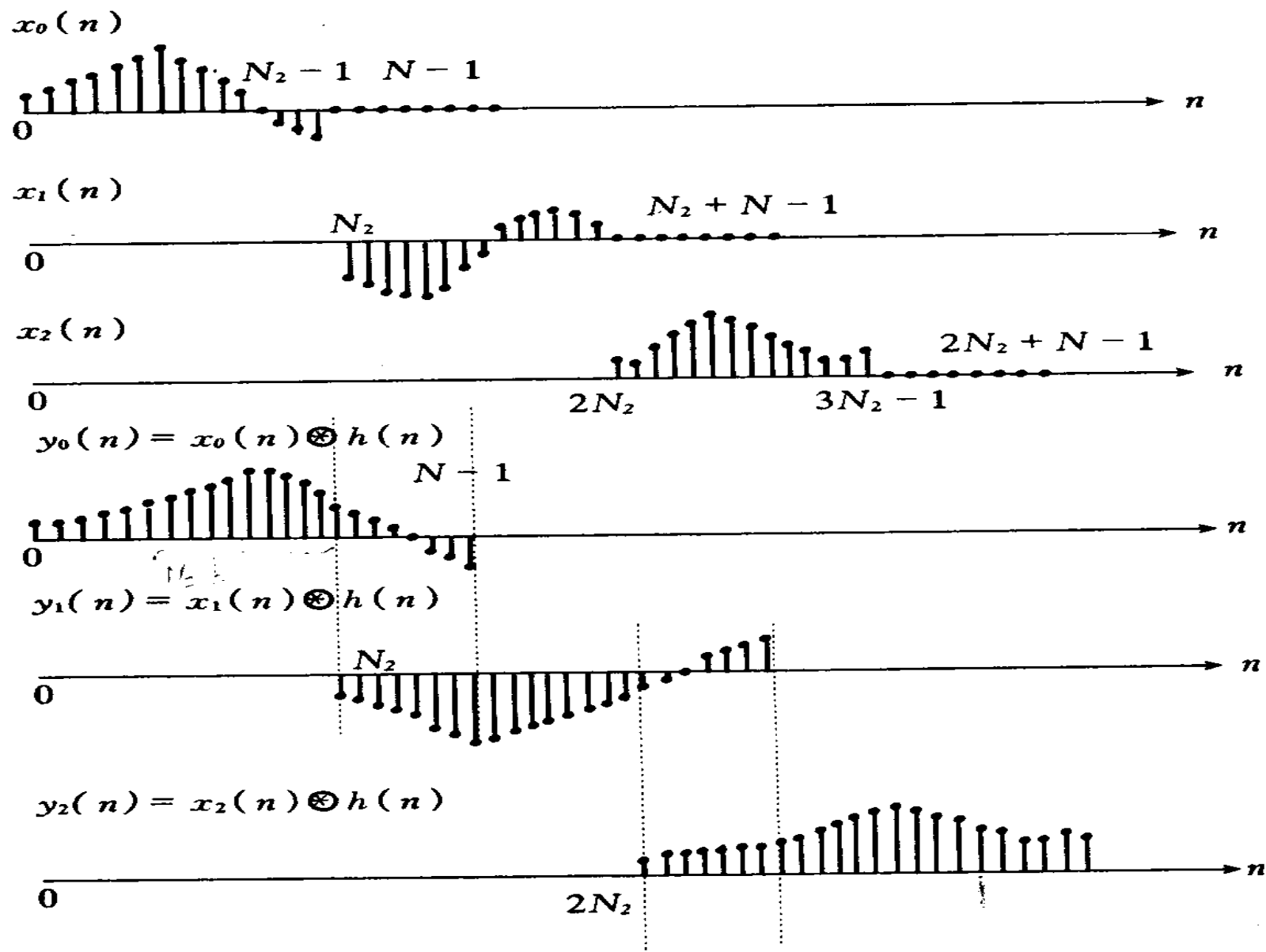


图 2.19 重叠相加法

基于FFT的重叠相加法的计算步骤

1. 计算并保存 $H(k) = DFT[h(n)]_L$, $L = N + M - 1, i = 0$
2. 读入 $x_i(n)$, 计算 L点FFT: $X_i(k) = DFT[x_i(n)]_L$
3. $Y_i(k) = H(k)X_i(k)$.
4. 计算L点IFFT: $y_i(n) = IDFT[Y_i(k)]_L, n = 0, 1, 2, \dots, L-1$
5. 重叠相加

$$y(iM + n) = \begin{cases} y_{i-1}(M + n) + y_i(n), & 0 \leq n \leq N - 2 \\ y_i(n), & N - 1 \leq n \leq N + M - 1 \end{cases}$$

6. $i=i+1$, 返回2

实现实时计算！

例3.5.2 设 $h(n) = R_5(n)$, $x(n) = [\cos(\pi n / 10) + \cos(2\pi n / 5)]u(n)$
用重叠相加法实现 $y(n) = h(n) * x(n)$

```
L=41;N=5;M=10;
```

```
hn=ones(1,N);hn1=[hn zeros(1,L-N)];
```

```
%产生h(n)，补零是为了绘图好看
```

```
n=0:L-1;
```

```
xn=cos(pi*n/10)+cos(2*pi*n/5);
```

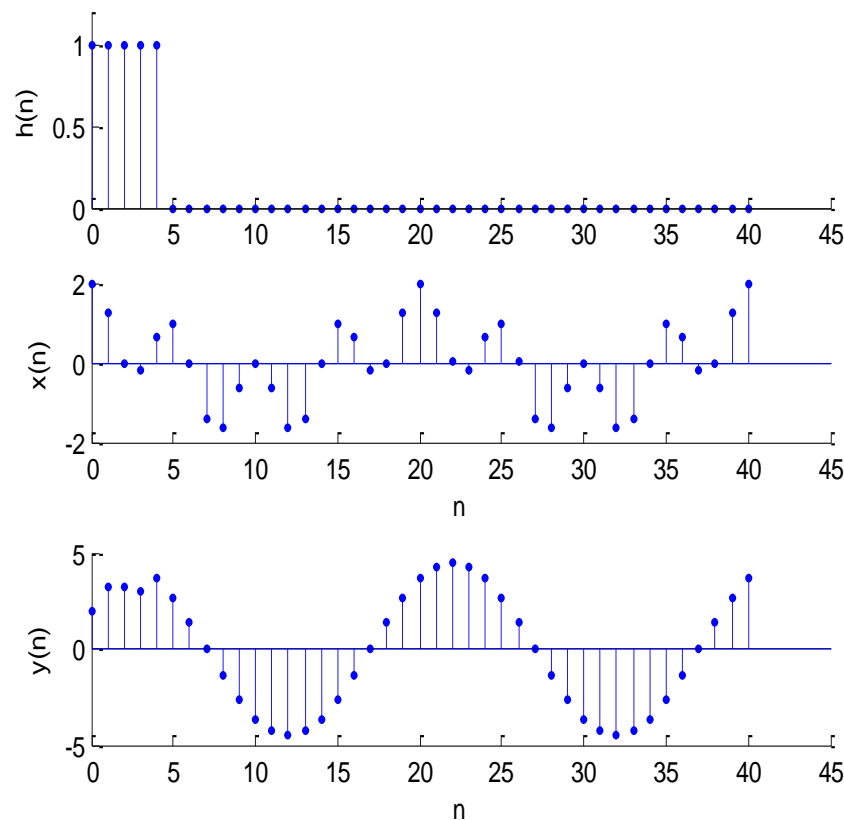
```
%产生x(n)的L个样值
```

```
yn=fftfilt(hn,xn,M);
```

```
%调用fftfilt计算卷积
```

```
%=====
```

```
%以下为绘图部分
```



3.5.3 用DFT对序列进行频谱分析（1）

- 有限长序列的频谱分析：
- 序列 $x(n)$ 的长度为**M**，序列 N ($N \geq M$) 点**DFT**的物理意义为
 - 序列的频谱函数 $X(e^{j\omega})$ 在频率区间 $[0, 2\pi]$ 上的**N**点等间隔采样
- **FFT**是**DFT**的快速算法，可用于有限长序列谱分析
- 问题：如何确定**N**？

3.5.3 用DFT对序列进行频谱分析（2）

1. 根据频率分辨率的要求确定N。

□ **频率分辨率**是指频谱分析中能够分辨的两个相邻频率点谱线的最小间距。

□ 例：在数字频率域，要求频率分辨率为D弧度

$$\frac{2\pi}{N} \leq D \quad \therefore N_{\min} = \left\lceil \frac{2\pi}{D} \right\rceil$$

□ 满足 $N \geq x(n)$ 的长度

□ 满足基2FFT对点数N的要求， $N = 2^M$ ， M为正整数。

3.5.3 用DFT对序列进行频谱分析（3）

2. 计算DFT，注意自变量k所对应的数字频率为：

$$\omega_k = 2\pi k / N$$

绘图时要以数字频率作为横坐标变量。

3. 若无频率分辨率要求，N可依据先验知识和实验进行确定
- 已知谱峰间距B，则分辨率为B/2；
 - 试验确定：随意取N，做DFT，再增大N，再做DFT，比较两次频谱，若差别较大，则再增加N，直到前后两次的频谱差异满足要求。

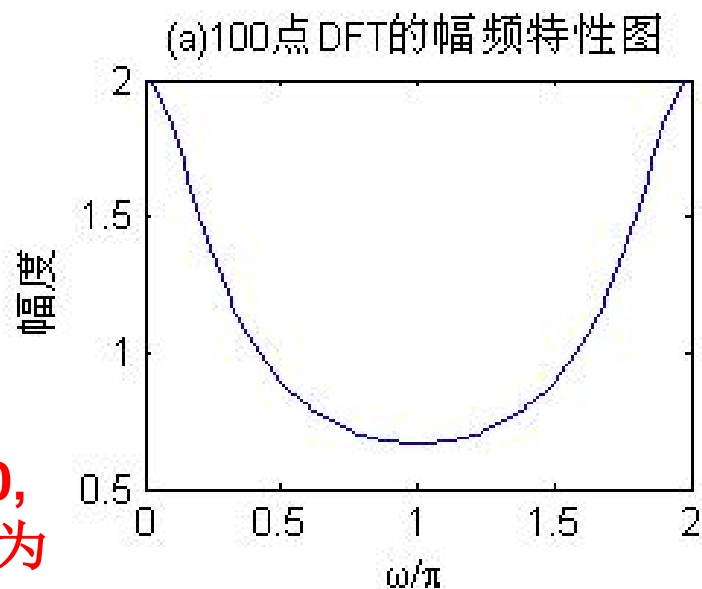
例3.5.3 $x(n) = 0.5^n R_{10}(n)$ 求频谱，分辨率为 **0.02π rad.**

解：

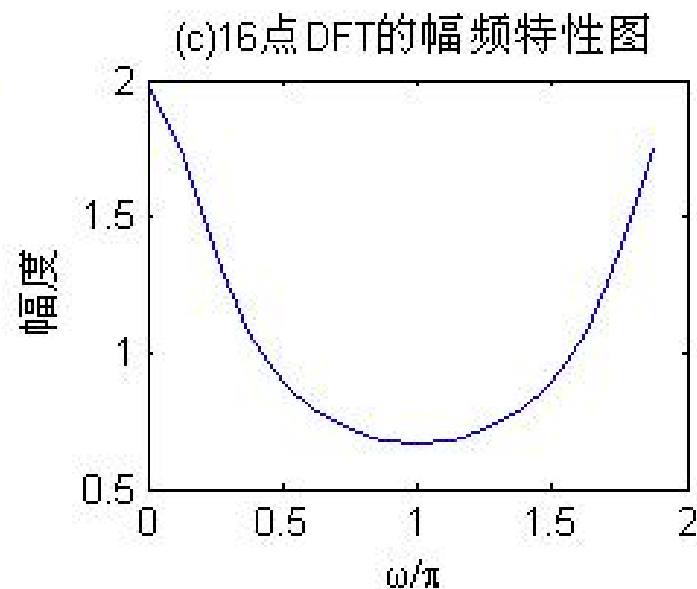
(1) 确定N $\therefore N_{\min} = \left\lceil \frac{2\pi}{D} \right\rceil = \frac{2\pi}{0.02\pi} = 100$

(2) 计算N点DFT

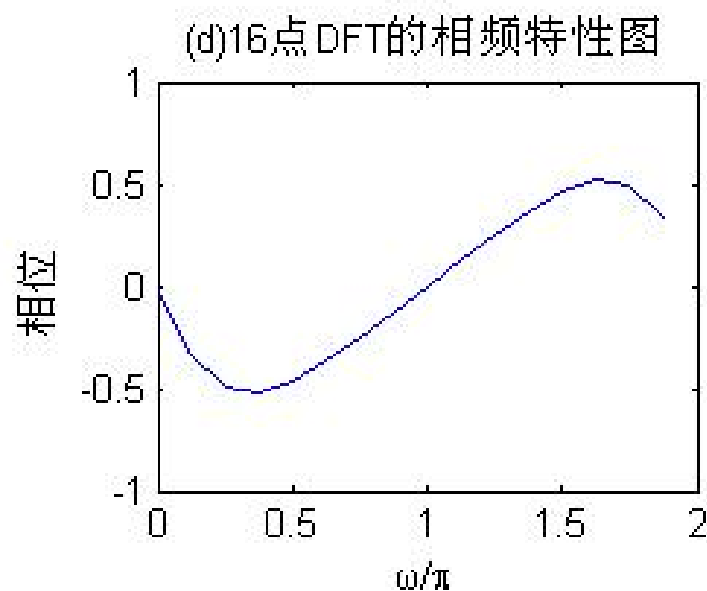
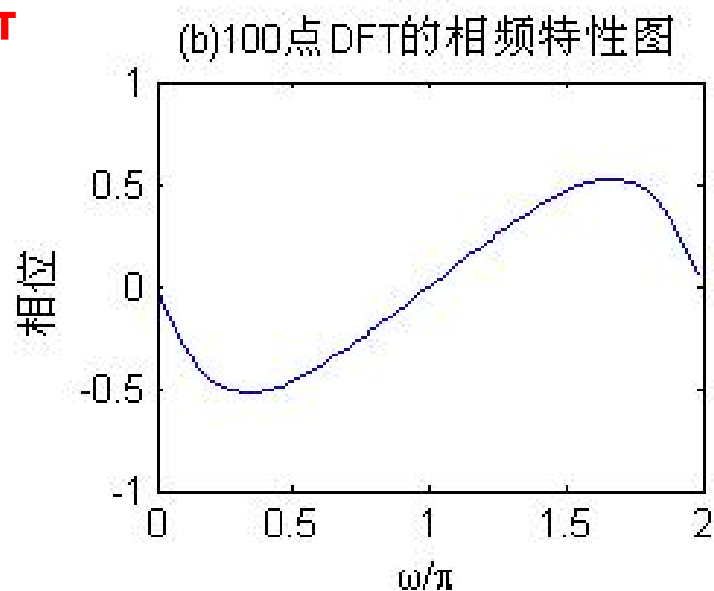
$$\begin{aligned} X(k) &= DFT[x(n)] = \sum_{n=0}^{N-1} x(n) W_N^{kn}, k = 0, 1, \dots, N-1 \\ &= \sum_{n=0}^9 0.5^n W_{100}^{kn} = \frac{1 - 0.5^{10} W_{100}^{10k}}{1 - 0.5 W_{100}^k}, k = 0, 1, \dots, N-1 \end{aligned}$$



N=100,
分辨率为
 0.02π



N=16,
分辨率为
 $\pi/8$



频谱变化缓慢，可以降低谱分析的分辨率

小结（1）

■ DFT提出的目的

- 序列的傅里叶变换、**Z**变换是时域离散信号及系统分析与设计的重要数学工具；
- 但变换结果均为连续函数，无法用计算机进行处理；
- 离散傅里叶变换（**DFT**）对有限长时域离散信号的频谱进行等间隔采样，频域函数被离散化了，便于信号的计算机处理。

■ DFT与ZT、FT、DFS之间的关系

小结：（2）

- 频域采样定理，频域内插公式
- DFT的性质（7个）
- 循环卷积与线性卷积之间的关系
- FFT的引入及基本思想
 - DFT运算量较大，快速离散傅里叶变换算法FFT是解决方案
 - 长序列分解为短序列，利用 W_N^m 的周期性和对称性
- DFT 的应用
 - 线性卷积
 - 频谱分析