

离散傅里叶变换(Discrete Fourier Transform, DFT)是数字信号处理中非常有用的一种变换,它对时域和频域都进行了离散化,使得计算机能够对信号进行时频域的计算。DFT 是有限长序列傅里叶变换的离散采样,它有多种快速算法,这些算法不仅在理论上重叠意义,而且在数字信号处理中也起着核心的作用。本章先从周期序列的离散傅里叶级数(Discrete Fourier Series, DFS)入手,再推广到针对有限长序列的 DFT,分析时需要注意 DFT 的隐含周期性。

3.1 周期序列的离散傅里叶级数

在离散时间信号处理中,需要利用计算机技术分析信号。由于信号的时域和频域的需要离散化,因此,在连续时间信号的傅里叶变换和傅里叶级数基础上,发展出了离散傅里叶级数和离散傅里叶变换。本节将首先介绍周期离散序列的傅里叶级数,并在此基础上进一步引出适用于有限长序列的离散傅里叶变换。

3.1.1 离散傅里叶级数的变换式

由于周期序列不满足 DTFT 的绝对可和的条件,因此其 DTFT 可能不收敛于一个有限值或特定函数的和。但是在引入函数 $\delta(n)$ 的情况下,周期序列可以展开成离散傅里叶级数。周期为 N 的周期序列 $\tilde{x}(n)$ 可以表示为

$$\tilde{x}(n) = \sum_{i=-\infty}^{\infty} x(n - iN) = x(n) * \sum_{i=-\infty}^{\infty} \delta(n - iN) \quad (3.1.1)$$

即可以将 $\tilde{x}(n)$ 看作有限长序列与周期为 N 的单位样值序列脉冲串的卷积。如果 $x(n)$ 的傅里叶变换为 $X(e^{j\omega})$, $\tilde{x}(n)$ 的傅里叶变换为 $\tilde{X}(e^{j\omega})$, 则根据卷积定理,可得

$$\begin{aligned} \tilde{X}(e^{j\omega}) &= \text{DTFT}[\tilde{x}(n)] = \text{DTFT}[x(n)] \cdot \text{DTFT}\left[\sum_{i=-\infty}^{\infty} \delta(n - iN)\right] \\ &= X(e^{j\omega}) \left(\frac{2\pi}{N} \sum_{k=-\infty}^{\infty} \delta\left(\omega - \frac{2\pi}{N}k\right)\right) = \frac{2\pi}{N} \sum_{k=-\infty}^{\infty} X(e^{j\frac{2\pi}{N}k}) \delta\left(\omega - \frac{2\pi}{N}k\right) \quad (3.1.2) \\ &= \frac{2\pi}{N} \sum_{k=-\infty}^{\infty} \tilde{X}(k) \delta\left(\omega - \frac{2\pi}{N}k\right) \end{aligned}$$

其中, $\sum_{i=-\infty}^{\infty} \delta(n-iN)$ 不是绝对可和的, 其 DTFT 使用奇异函数表示

$$\text{DTFT} \left[\sum_{i=-\infty}^{\infty} \delta(n-iN) \right] = \sum_{n=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} \delta(n-iN) \cdot e^{-j\omega n} = \sum_{i=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(n-iN) \cdot e^{-j\omega n}$$

交换求和顺序并利用冲激函数的性质有

$$\text{DTFT} \left[\sum_{i=-\infty}^{\infty} \delta(n-iN) \right] = \sum_{i=-\infty}^{\infty} e^{-j\omega iN}$$

根据泊松求和公式

$$\sum_{i=-\infty}^{\infty} e^{-j\omega iN} = 2\pi \sum_{k=-\infty}^{\infty} \delta(N\omega - 2\pi k)$$

利用 $\delta(\omega)$ 函数的缩放性质, 有

$$\text{DTFT} \left[\sum_{i=-\infty}^{\infty} \delta(n-iN) \right] = \frac{2\pi}{N} \sum_{k=-\infty}^{\infty} \delta\left(\omega - \frac{2\pi k}{N}\right)$$

式(3.1.2)说明周期序列 $\tilde{x}(n)$ 的傅里叶变换 $\tilde{X}(e^{j\omega})$ 是频率在 $\frac{2\pi}{N}$ 的整数倍上的一系列冲激函数, 其每个冲激函数的积分面积等于 $\tilde{X}(k)$ 与 $\frac{2\pi}{N}$ 的乘积。 $\tilde{X}(k)$ 为 $X(e^{j\omega})$ 在 $\omega = \frac{2\pi k}{N}$ 上的采样值 ($k=0, 1, \dots, N-1$)。

$$\begin{aligned} \tilde{X}(k) &= X(e^{j\omega}) \Big|_{\omega = \frac{2\pi k}{N}} = \sum_{n=0}^{N-1} \tilde{x}(n) e^{-j\omega n} \Big|_{\omega = \frac{2\pi k}{N}} \\ &= \sum_{n=0}^{N-1} \tilde{x}(n) e^{-j\frac{2\pi}{N}nk} = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}nk} \end{aligned} \quad (3.1.3)$$

说明周期性序列的傅里叶变换仍以 2π 为周期, 且在一个周期中包含 N 个由冲激函数表示的谱线。

对式(3.1.2)求傅里叶逆变换, 得

$$\tilde{x}(n) = \frac{1}{2\pi} \int_0^{2\pi} \left[\frac{2\pi}{N} \sum_{k=-\infty}^{\infty} \tilde{X}(k) \delta\left(\omega - \frac{2\pi k}{N}\right) \right] e^{j\omega n} d\omega$$

积分区间为 $[0, 2\pi]$, 在此区间内共有 N 个采样值, 即 $0 \leq k \leq N-1$, 因而可以写成

$$\begin{aligned} \tilde{x}(n) &= \frac{1}{N} \int_0^{2\pi} \left[\sum_{k=0}^{N-1} \tilde{X}(k) \delta\left(\omega - \frac{2\pi k}{N}\right) \right] e^{j\omega n} d\omega \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}(k) \int_0^{2\pi} \left[\delta\left(\omega - \frac{2\pi k}{N}\right) \right] e^{j\omega n} d\omega \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}(k) e^{j\frac{2\pi}{N}kn} \end{aligned} \quad (3.1.4)$$

这是周期序列 $\tilde{x}(n)$ 的傅里叶展开式, 它表示周期序列可以展开为谐波分量 $e^{j\frac{2\pi}{N}kn}$ ($k=0, 1, \dots, N-1$) 的加权和的形式, 展开系数为 $\tilde{X}(k)$ 。

式(3.1.3)和式(3.1.4)构成了周期序列的离散傅里叶级数变换对。

$$\tilde{X}(k) = \sum_{n=0}^{N-1} \tilde{x}(n) e^{-j\frac{2\pi}{N}nk} \quad (3.1.5)$$

$$\tilde{x}(n) = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}(k) e^{j\frac{2\pi}{N}kn} \quad (3.1.6)$$

【例 3.1.1】 设周期为 10 的周期序列 $\tilde{x}(n)$ 的一个周期 $x(n)$ 为

$$x(n) = R_5(n)$$

求 $\tilde{x}(n)$ 的傅里叶变换 $\tilde{X}(e^{j\omega})$ 。

解 周期序列可以看作由有限长序列 $x(n)$ 与周期采样序列 $\delta_{10}(n)$ 的卷积,如图 3.1.1 所示,即

$$\delta_{10}(n) = \sum_{i=-\infty}^{\infty} \delta(n - 10i)$$

$$\tilde{x}(n) = x(n) * \delta_{10}(n)$$

单个序列 $x(n)$ 的傅里叶变换为

$$X(e^{j\omega}) = \sum_{n=0}^9 x(n) e^{-j\omega n} = \frac{1 - e^{-j5\omega}}{1 - e^{-j\omega}} = e^{-j2\omega} \frac{\sin\left(\frac{5\omega}{2}\right)}{\sin\left(\frac{\omega}{2}\right)}$$

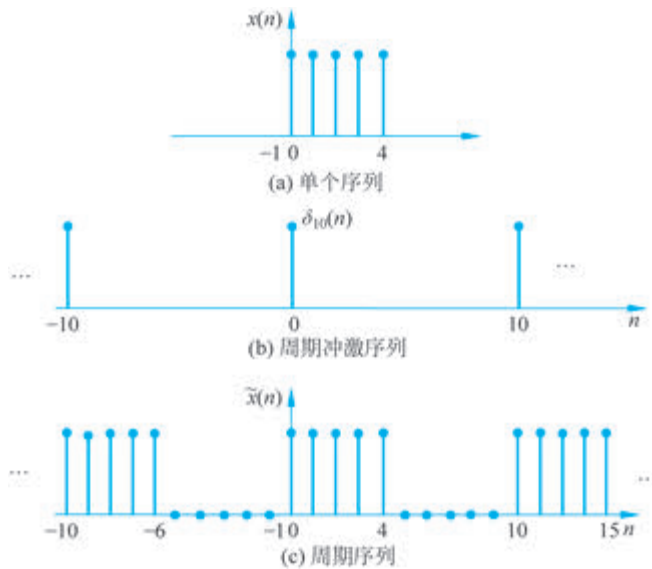


图 3.1.1 单个序列与周期冲激序列卷积得到周期序列

由卷积定理,周期序列 $\tilde{x}(n)$ 的傅里叶变换为单个序列的傅里叶变换与周期冲激序列傅里叶变换的乘积,是一系列在离散频率点的冲激信号,如图 3.1.2(a)所示。

$$\begin{aligned} \tilde{X}(e^{j\omega}) &= \text{DTFT}[\tilde{x}(n)] = \text{DTFT}[x(n)] \cdot \text{DTFT}\left[\sum_{i=-\infty}^{\infty} \delta(n - 10i)\right] \\ &= \frac{\pi}{5} \sum_{k=-\infty}^{\infty} X(e^{j\omega}) \delta\left(\omega - \frac{\pi}{5}k\right) = \frac{\pi}{5} \sum_{k=-\infty}^{\infty} X(e^{j\frac{\pi}{5}k}) \delta\left(\omega - \frac{\pi}{5}k\right) \\ &= \frac{\pi}{5} \sum_{k=-\infty}^{\infty} \tilde{X}(k) \delta\left(\omega - \frac{\pi}{5}k\right) \end{aligned}$$

其中, $\tilde{X}(k)$ 为周期序列的离散傅里叶级数,即

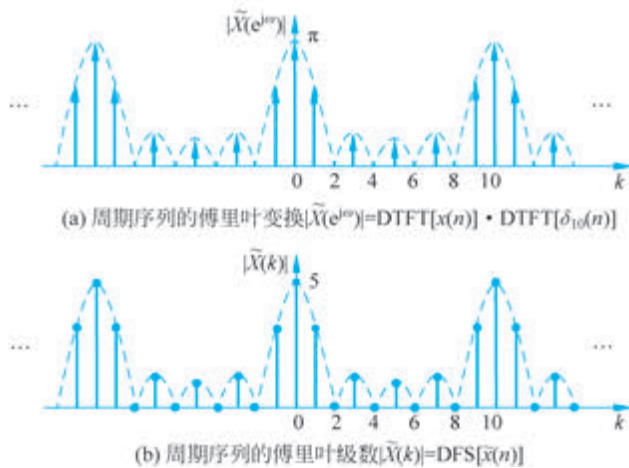


图 3.1.2 周期矩形序列的频谱

$$\begin{aligned} \tilde{X}(k) &= \sum_{n=0}^9 x(n) e^{-j\frac{\pi}{5}nk} = \sum_{n=0}^4 e^{-j\frac{\pi}{5}nk} \\ &= \frac{1 - e^{-j\pi k}}{1 - e^{-j\frac{\pi}{5}k}} = \frac{e^{-j\frac{\pi}{2}k} (e^{j\frac{\pi}{2}k} - e^{-j\frac{\pi}{2}k})}{e^{-j\frac{\pi}{10}k} (e^{j\frac{\pi}{10}k} - e^{-j\frac{\pi}{10}k})} = e^{-j\frac{2\pi}{5}k} \frac{\sin \frac{\pi}{2}k}{\sin \frac{\pi}{10}k} \end{aligned}$$

幅度特性如图 3.1.2(b)所示。

3.1.2 离散傅里叶级数和连续时间周期信号傅里叶级数关系

离散傅里叶级数的变换式(3.1.5)和式(3.1.6)中, $\tilde{x}(n)$ 和 $\tilde{X}(k)$ 都是周期为 N 的周期序列。而连续时间周期信号 $\tilde{x}(t)$ 的傅里叶级数展开形式为

$$\tilde{x}(t) = \sum_{k=-\infty}^{\infty} X(k\Omega_0) e^{jk\Omega_0 t} \quad (3.1.7)$$

对比可得,连续时间周期信号展开的基频信号为连续频率信号 $e^{jk\Omega_0 t}$,而离散时间周期信号展开式中的基频信号是 $e^{j\frac{2\pi}{N}n}$ 。连续时间周期信号有无穷多个谐波分量,而离散时间周期序列只能取有限个谐波分量,即 $k=0 \sim N-1$ 。

表现形式上两者是相似的,但是离散傅里叶级数的谐波成分中只有 N 个独立成分,这与连续傅里叶级数不同,后者有无穷多个谐波成分。连续傅里叶级数和离散傅里叶级数的基频、周期和展开式总结在表 3.1.1 中。离散傅里叶级数在时域和频域都是离散的周期序列。对于周期序列,如果知道了一个周期的序列值,就可以获得周期序列的全部信息。此外,可以将离散傅里叶级数扩展到分析有限长序列(如一个周期长度)的傅里叶变换。

表 3.1.1 连续时间周期信号与周期序列的傅里叶级数对比

信号类型	基频信号(序列)	周期	傅里叶级数展开式
连续时间周期信号	$e^{j\Omega_0 t} = e^{j\frac{2\pi}{T_0}t}$	T_0	$\tilde{x}(t) = \sum_{k=-\infty}^{\infty} X(jk\Omega_0) e^{jk\Omega_0 t}$

续表

信号类型	基频信号(序列)	周期	傅里叶级数展开式
离散时间周期信号	$e^{j\omega_0 n} = e^{j\frac{2\pi}{N}n}$	N	$\tilde{x}(n) = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}(k) e^{j\frac{2\pi}{N}nk}$

3.1.3 离散傅里叶级数和离散时间傅里叶变换的关系

有限长序列 $x(n)$ 的 DTFT (离散时间傅里叶变换) $X(e^{j\omega})$ 是周期为 2π 的周期函数。在一个周期 $0 \sim 2\pi$ 内对 $X(e^{j\omega})$ 采样 N 个点, 采样点之间的频率间距为 $\frac{2\pi}{N}$, 得到的离散频率点可以表示为

$$\omega = \frac{2\pi}{N}k \quad (3.1.8)$$

代入 DTFT 的变换式(2.2.1), 并用 $\tilde{X}(k)$ 表示 $X(e^{j\omega})$ 的第 k 点采样值, $\tilde{X}(k)$ 中的上标“ \sim ”表示其周期性, 以便与后面讲述的有限长序列 $X(k)$ 的 DFT 做区分。 $\tilde{X}(k)$ 可表示为

$$\tilde{X}(k) = X(e^{j\omega}) \Big|_{\omega = \frac{2\pi}{N}k} = \sum_{n=0}^{N-1} \tilde{x}(n) e^{-j\frac{2\pi}{N}nk}, \quad k = 0, 1, 2, \dots, N-1 \quad (3.1.9)$$

这是 DFS (离散傅里叶级数) 的正变换公式。其中, $\tilde{x}(n)$ 为时域周期序列。在 3.5 节频域采样定理部分, 将详细解释时域有限长序列 $x(n)$ 是如何被延拓为周期序列 $\tilde{x}(n)$ 的。此处只需知道, $\tilde{x}(n)$ 在 $n=0, 1, 2, \dots, N-1$ 上与 $x(n)$ 是相同的。

对比 DTFT 的变换式(2.2.1), 可以看到, 周期序列的 DFS 是有限长序列的 DTFT 在频域的采样, DTFT 中的频率变量 ω 是连续量, DFS 中的频率变量变为 $\frac{2\pi}{N}k$, 其中, $k=0, 1, \dots, N-1$ 为离散量。

3.1.4 离散傅里叶级数和 Z 变换的关系

【例 3.1.2】 设 $\tilde{x}(n)$ 为周期为 $N_1=4$ 的周期序列, 其一个周期内的序列为 $x(n) = R_4(n)$, 即 $\tilde{x}(n) = \sum_{i=-\infty}^{\infty} x(n+8i)$ 。 $N=8$, 求 $\tilde{x}(n)$ 的离散傅里叶级数 $\tilde{X}(k)$ 。

解

$$\tilde{X}(k) = \sum_{n=0}^3 \tilde{x}(n) e^{-j\frac{2\pi}{8}nk} = \frac{1 - e^{-j\pi k}}{1 - e^{-j2\pi k/8}} = e^{-\frac{j3\pi k}{8}} \frac{\sin(\pi k/2)}{\sin(\pi k/8)}$$

$\tilde{X}(k)$ 是周期为 8 的周期序列, 如图 3.1.3 所示。

从例 3.1.2 中可以看到, 周期为 N 的序列 $\tilde{x}(n)$ 的离散傅里叶级数 $\tilde{X}(k)$ 是有限长序列 $x(n)$ 的 DTFT 的采样, $\tilde{X}(k)$ 的周期也是 N , 即在 DTFT 的一个周期 $0 \sim 2\pi$ 内等间隔抽取 N 个点, 得到频域上的主值序列, 即一个周期内的离散频率序列下标为 $k=0, 1, 2, \dots, N-1$ 。

由之前的推导已知, 有限长序列 $x(n)$ 的 DTFT 是 $x(n)$ 的 Z 变换在单位圆上的值, 则周期序列 $\tilde{x}(n)$ 的 DFS 就是有限长序列 $x(n)$ 的 Z 变换在单位圆上的等间隔采样, 如图 3.1.4 所示。

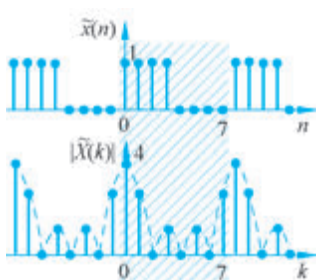


图 3.1.3 周期序列的离散傅里叶级数

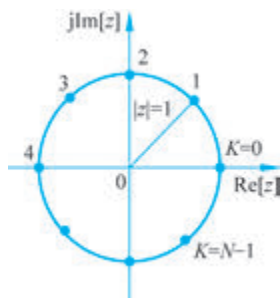


图 3.1.4 DFS 是主值序列的 Z 变换在单位圆上的采样值

3.1.5 离散傅里叶级数的性质

令 $\tilde{x}_1(n)$ 和 $\tilde{x}_2(n)$ 为周期为 N 的周期序列, 它们的 DFS 为 $\tilde{X}_1(k) = \text{DFS}[\tilde{x}_1(n)]$, $\tilde{X}_2(k) = \text{DFS}[\tilde{x}_2(n)]$ 。

1. 线性

$$\text{DFS}[a\tilde{x}_1(n) + b\tilde{x}_2(n)] = a\tilde{X}_1(k) + b\tilde{X}_2(k) \quad (3.1.10)$$

式中, a, b 为任意常数, $\tilde{x}_1(n)$ 和 $\tilde{x}_2(n)$ 线性组合的 DFS 是它们各自 DFS 的线性组合, 所得的频域序列也是周期序列, 周期为 N 。

2. 序列移位性

$$\text{DFS}[\tilde{x}(n+m)] = W_N^{-mk} \tilde{X}(k) = e^{j\frac{2\pi}{N}mk} \tilde{X}(k) \quad (3.1.11)$$

周期序列在时域移位后的 DFS 是原序列的 DFS 与移位复指数序列的乘积。

为了方便表示, 引入变量 $W_N = e^{-j\frac{2\pi}{N}}$, 由 DFS 正变换有:

$$\text{DFS}[\tilde{x}(n+m)] = \sum_{n=0}^{N-1} \tilde{x}(n+m) W_N^{nk} = \sum_{i=m}^{N-1+m} \tilde{x}(i) W_N^{ki} W_N^{-mk}, \quad i = n+m$$

由于 $\tilde{x}(i)$ 及 $e^{-j\frac{2\pi}{N}ki}$ 都是以 N 为周期的周期函数, 所以

$$\text{DFS}[\tilde{x}(n+m)] = W_N^{-mk} \sum_{i=0}^{N-1} \tilde{x}(i) W_N^{ki} = W_N^{-mk} \tilde{X}(k) = e^{j\frac{2\pi}{N}mk} \tilde{X}(k)$$

3. 频域移位性

如果 DFS 的频域周期序列 $\tilde{X}(k)$ 移 l 位, 则时域周期序列 $\tilde{x}(n)$ 需要乘以复指数函数 W_N^{ln} , 即乘以旋转因子 $e^{-j\frac{2\pi}{N}ln}$ 的 ln 次方, 即

$$\text{DFS}[W_N^{ln} \tilde{x}(n)] = \tilde{X}(k+l) \quad (3.1.12)$$

证明: 由 DFS 的定义

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

得证。

4. 对称性

将 DFS 逆变换式

$$\tilde{x}(n) = \tilde{x}(nT) = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}(k) e^{j\frac{2\pi}{N}nk}$$

中的 n 替换为 $-n$, 可以得到

$$N\tilde{x}(-n) = \sum_{k=0}^{N-1} \tilde{X}(k) e^{-j\frac{2\pi}{N}nk} \quad (3.1.13)$$

由于等式右边是与 DFS 正变换表达式相同的形式, 故将式中 n 和 k 互换, 可得

$$N\tilde{x}(-k) = \sum_{n=0}^{N-1} \tilde{X}(n) e^{-j\frac{2\pi}{N}nk} \quad (3.1.14)$$

即周期序列 $\tilde{X}(n)$ 的 DFS 系数是 $N\tilde{x}(-k)$, 因而具有对称性, 其表示为

$$\begin{aligned} \text{DFS}[\tilde{x}(n)] &= \tilde{X}(k) \\ \text{DFS}[\tilde{X}(n)] &= N\tilde{x}(-k) \end{aligned} \quad (3.1.15)$$

即序列 $\tilde{X}(n)$ 在时域和频域的函数形式可以交换并互相对应。

5. 周期序列卷积

(1) 若

$$\tilde{Y}(k) = \tilde{X}_1(k) \cdot \tilde{X}_2(k)$$

则 $\tilde{Y}(k)$ 的 IDFS 为

$$\begin{aligned} \tilde{y}(n) &= \text{IDFS}[\tilde{Y}(k)] = \sum_{m=0}^{N-1} \tilde{x}_1(m) \tilde{x}_2(n-m) \\ &= \sum_{m=0}^{N-1} \tilde{x}_2(m) \tilde{x}_1(n-m) \end{aligned} \quad (3.1.16)$$

即频域周期序列的乘积对应于时域周期序列的周期卷积。

证明:

$$\tilde{y}(n) = \text{IDFS}[\tilde{X}_1(k) \tilde{X}_2(k)] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}_1(k) \tilde{X}_2(k) W_N^{-kn} \quad (3.1.17)$$

$$\tilde{X}_1(k) = \sum_{m=0}^{N-1} \tilde{x}_1(m) W_N^{mk} \quad (3.1.18)$$

将式(3.1.18)代入式(3.1.17)得

$$\begin{aligned} \tilde{y}(n) &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{m=0}^{N-1} \tilde{x}_1(m) \tilde{X}_2(k) W_N^{-(n-m)k} = \sum_{m=0}^{N-1} \tilde{x}_1(m) \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}_2(k) W_N^{-(n-m)k} \\ &= \sum_{m=0}^{N-1} \tilde{x}_1(m) \tilde{x}_2(n-m) \end{aligned}$$

同理

$$\tilde{y}(n) = \sum_{m=0}^{N-1} \tilde{x}_2(m) \tilde{x}_1(n-m)$$

周期卷积的计算与非周期序列的线性卷积计算不同。 $\tilde{x}_1(m)$ 、 $\tilde{x}_1(n-m)$ 、 $\tilde{x}_2(m)$ 与 $\tilde{x}_1(n-m)$ 都是变量 m 的周期序列, 周期为 N , 它们的乘积也是周期为 N 的周期序列; 卷积中求和只在一个周期上进行, 即 $m \in [0, N-1]$, 所以称为周期卷积。得到的卷积结果

也是周期为 N 的周期序列。

在周期卷积的过程中,一个周期的某一序列值移出计算区间时,相邻周期的同一位置的序列值就从另一端移入计算区间。 $m \in [0, N-1]$,先计算出 $n=0, 1, 2, \dots, N-1$,然后将所得结果周期延拓,即可得到所求的整个周期序列 $\tilde{y}(n)$ 。

【例 3.1.3】 设有两个周期为 6 的周期序列 $\tilde{x}_1(n)$ 和 $\tilde{x}_2(n)$, 其中的主值序列 $x_1(n) = \{1, 2, 3, 4\}$, $x_2(n) = \{1, 2, 2, 1\}$, 做周期为 6 的 DFS, $\tilde{X}_1(k) = \text{DFS}(\tilde{x}_1(n))$, $\tilde{X}_2(k) = \text{DFS}(\tilde{x}_2(n))$, $\tilde{Y}(k) = \tilde{X}_1(k) \cdot \tilde{X}_2(k)$, 试求 $\tilde{Y}(k)$ 对应的时域周期序列 $\tilde{y}(n)$ 。

解 采用图解法进行周期卷积,从图 3.1.5 中可以看到,序列的翻褶和移位与线性卷积相同,但是相乘和累加是在一个周期里进行。如图 3.1.5 中阴影部分为一个周期,得到的周期卷积结果 $\tilde{y}(n)$ 仍然是周期为 6 的周期序列。

$$\tilde{y}(n) \cdot R_6(n) = \{5, 4, 9, 15, 16, 11\}$$

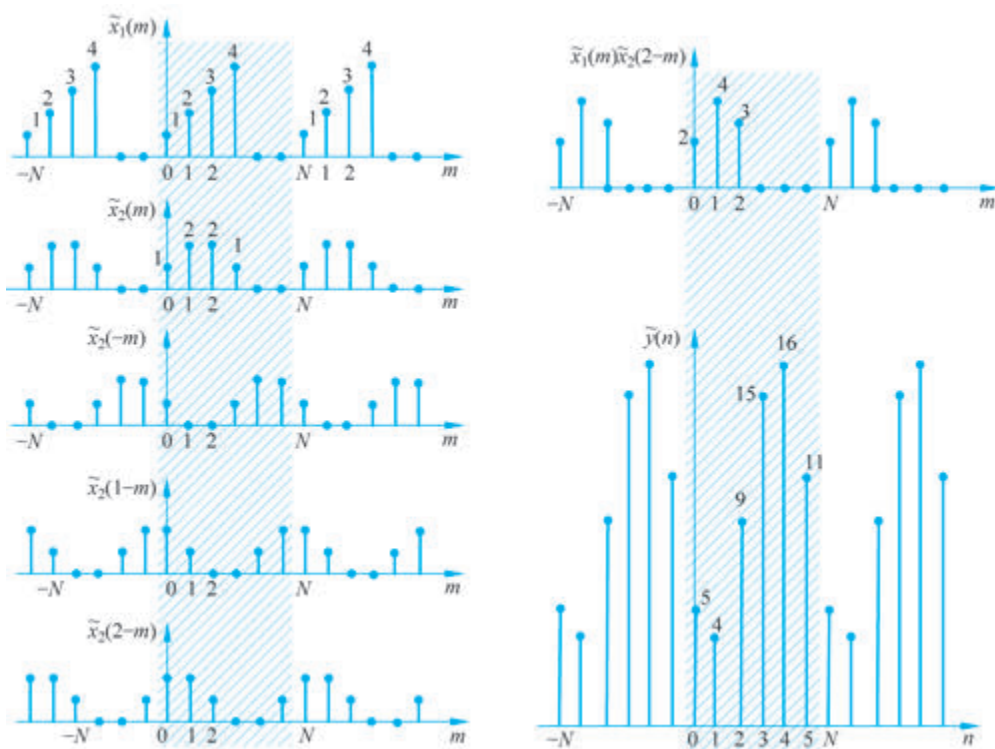


图 3.1.5 两个周期序列 ($N=6$) 的周期卷积过程

(2) 由于 DFS 具有对称性,若

$$\tilde{y}(n) = \tilde{x}_1(n) \tilde{x}_2(n)$$

则

$$\begin{aligned} \tilde{Y}(k) &= \text{DFS}[\tilde{y}(n)] = \sum_{n=0}^{N-1} \tilde{y}(n) W_N^{nk} \\ &= \frac{1}{N} \sum_{l=0}^{N-1} \tilde{X}_1(l) \tilde{X}_2(k-l) = \frac{1}{N} \sum_{l=0}^{N-1} \tilde{X}_2(l) \tilde{X}_1(k-l) \end{aligned} \quad (3.1.19)$$

即时域周期序列的乘积等于频域周期序列的周期卷积结果除以 N 。

3.2 傅里叶变换的四种形式

根据傅里叶变换在时域和频域的离散性和周期性,将傅里叶变换分为四种形式,分别对应时间变量取连续值或离散值、频率变量取连续值或离散值的四种组合情况。

3.2.1 连续时间与连续频率——连续傅里叶变换(CTFT)

连续傅里叶变换,即连续时间傅里叶变换(Continuous Time Fourier Transform, CTFT),可表示为

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

其逆变换可表示为

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\Omega) e^{j\Omega t} d\Omega \quad (3.2.1)$$

其中, $x(t)$ 是连续时间非周期信号如图 3.2.1(a)所示; $\Omega = 2\pi f$ 为角频率,单位为 rad/s; $x(j\Omega)$ 是 Ω 的连续非周期函数,如图 3.2.1(b)所示。

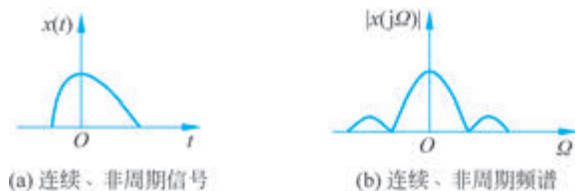


图 3.2.1 连续时间傅里叶变换(CTFT)

3.2.2 连续时间与离散频率——傅里叶级数(FS)

傅里叶级数可表示为

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

其逆变换为

$$x(t) = \sum_{k=-\infty}^{\infty} X(jk\Omega_0) e^{jk\Omega_0 t}$$

其中, $\Omega_0 = 2\pi F_0 = \frac{2\pi}{T_0}$ 为离散频谱相邻两谱线之间的角频率间隔, k 为谐波序号。时域信号是连续周期信号, $X(jk\Omega_0)$ 为离散频率的非周期信号,如图 3.2.2 所示。

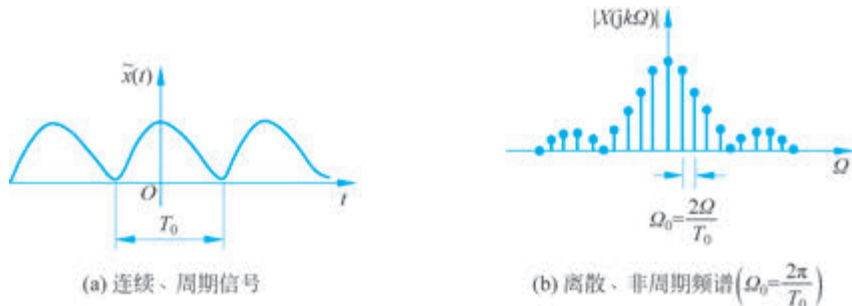


图 3.2.2 傅里叶级数(FS)



3.2.3 离散时间与连续频率——离散时间傅里叶变换(DTFT)

离散时间傅里叶变换表示为

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

其逆变换为

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

其中, ω 为数字频率, 它和模拟角频率 Ω 的关系为 $\omega = \Omega T$ 。

对于 $X(e^{j\omega})$ 有

$$X(e^{j(\omega+2\pi)}) = \sum_{n=-\infty}^{\infty} x(n)e^{-j(\omega+2\pi)n} = e^{-j2\pi n} \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} = X(e^{j\omega})$$

$X(e^{j\omega})$ 是 ω 的周期函数, 周期为 2π 。时域信号是离散的非周期信号, 频域是连续周期信号, 如图 3.2.3 所示。

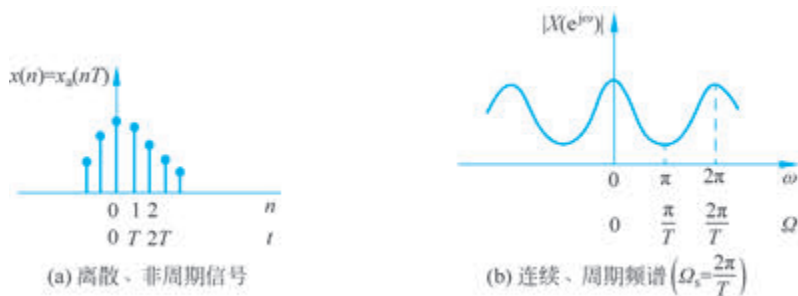


图 3.2.3 离散时间傅里叶变换(DTFT)

上面讨论的三种傅里叶变换至少在一个域(时域或频域)中的函数是连续的, 因而都不适于在计算机上运算。从数字计算角度出发, 需要时域及频域都是离散的情况, 因此第四种周期序列的傅里叶级数, 即离散傅里叶级数是要进一步讨论的变换。

3.2.4 离散时间与离散频率——离散傅里叶级数(DFS)

离散傅里叶级数表示为

$$\tilde{X}(k) = \sum_{n=0}^{N-1} \tilde{x}(n) e^{-j\frac{2\pi}{N}nk}$$

其逆变换为

$$\tilde{x}(n) = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}(k) e^{j\frac{2\pi}{N}nk}$$

周期序列离散傅里叶级数在时域和频域都是离散的周期信号。如图 3.2.4 所示。

将以上四种傅里叶变换类型的时域和频域特性归纳在表 3.2.1 中, 从表中可以发现以下规律: 一个域的连续对应着另一个域一定是非周期的; 一个域的离散对应着另一个域一定是周期的。例如 DTFT 的时域是离散的, 频域一定是周期的。时域是非周期的, 频域一定是连续的, 如表 3.2.1 所示。

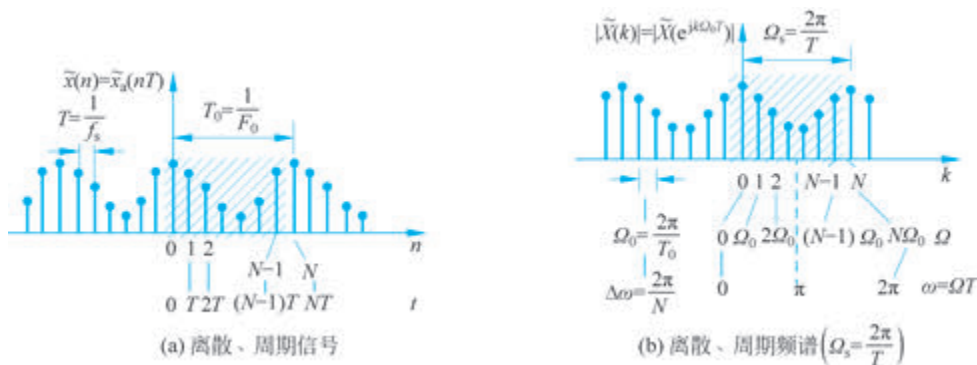


图 3.2.4 离散傅里叶级数(DFS)

表 3.2.1 四种傅里叶变换类型在时域和频域的特性

傅里叶变换类型	时 域	频 域
连续时间傅里叶变换(CTFT)	连续、非周期	连续、非周期
傅里叶级数(FS)	连续、周期	离散、非周期
离散时间傅里叶变换(DTFT)	离散、非周期	连续、周期
离散傅里叶级数(DFS, 主值区间为 DFT)	离散、周期	离散、周期



第 12 集
微课视频

3.3 有限长序列的离散傅里叶变换

3.3.1 离散傅里叶变换的定义

离散傅里叶级数的时域信号 $\tilde{x}(n)$ 和频域信号 $\tilde{X}(k)$ 都是离散且周期的序列。在计算机处理信号时,通常要求时域和频域信号都应是有限长序列。对于周期序列,只需知道其中一个周期的信息,即可推导出整个周期序列的所有信息。因此,从离散傅里叶级数时域和频域中取出主值区间,可以得到一对正逆变换对,称为有限长序列的离散傅里叶变换(DFT),其正变换为

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}nk} = \sum_{n=0}^{N-1} x(n) W_N^{nk} \quad (3.3.1)$$

逆变换为

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi}{N}nk} = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk} \quad (3.3.2)$$

$X(k)$ 和 $x(n)$ 分别是 $\tilde{X}(k)$ 和 $\tilde{x}(n)$ 的主值序列,为了方便表示,引入变量 $W_N = e^{-j\frac{2\pi}{N}}$ 。 $X(k)$ 和 $x(n)$ 也是一一对应的关系,即由其中一个,可以唯一确定另一个序列。

下面证明 IDFT $[X(k)]$ 的唯一性。

把式(3.3.1)代入式(3.3.2),有

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

其中,

$$\frac{1}{N} \sum_{k=0}^{N-1} W_N^{k(m-n)} = \begin{cases} 1, & m = n + iN, i \text{ 为整数} \\ 0, & m \neq n + iN, i \text{ 为整数} \end{cases} \quad (3.3.3)$$

在变换区间上满足

$$\text{IDFT}[X(k)]_N = x(n), \quad 0 \leq n \leq N-1$$

由此可见,式(3.3.2)定义的离散傅里叶逆变换是唯一的。

【例 3.3.1】 序列 $x(n) = \cos \frac{\pi}{6} n R_N(n)$, 计算 $N=12$ 时的 DFT。

解

$$\begin{aligned} X(k) &= \sum_{n=0}^{N-1} x(n) W_N^{nk} = \sum_{n=0}^{11} \frac{1}{2} [e^{j\frac{2\pi}{12}n} + e^{-j\frac{2\pi}{12}n}] e^{-j\frac{2\pi}{12}nk} \\ &= \frac{1}{2} \sum_{n=0}^{11} (e^{-j\frac{2\pi}{12}n(k-1)} + e^{-j\frac{2\pi}{12}n(k+1)}) \\ &= \frac{1}{2} \times \frac{1 - e^{-j2\pi(k-1)}}{1 - e^{-j\frac{2\pi}{12}(k-1)}} + \frac{1}{2} \times \frac{1 - e^{-j2\pi(k+1)}}{1 - e^{-j\frac{2\pi}{12}(k+1)}} \\ &= \begin{cases} 6, & k=1, 11 \\ 0, & k=0, 2, 3, 4, 5, 6, 7, 8, 9, 10 \end{cases} \end{aligned}$$

余弦序列的 DFT 如图 3.3.1 所示。

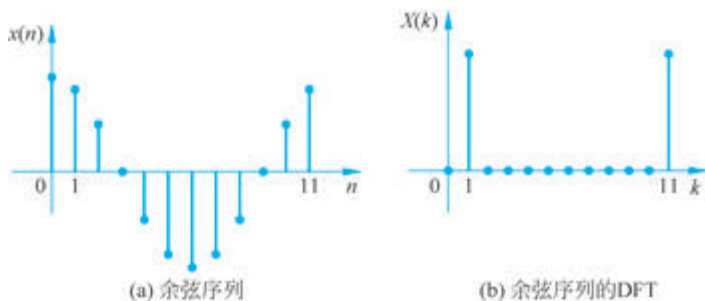


图 3.3.1 余弦序列的 DFT

可以看到,余弦序列 $x(n)$ 的 DFT 是在 $k=1$ 和 $k=11$ 处的两条谱线。

【例 3.3.2】 $x(n) = R_4(n)$, 求 $x(n)$ 的 DTFT 以及 $N=4, 8, 16$ 时的 DFT。

解 设 DFT 长度为 $N=4$, 则

$$X(k) = \sum_{n=0}^3 x(n) W_4^{kn} = \sum_{n=0}^3 e^{-j\frac{2\pi}{4}kn} = \frac{1 - e^{-j2\pi k}}{1 - e^{-j\frac{\pi}{2}k}} = \begin{cases} 4, & k=0 \\ 0, & k=1, 2, 3 \end{cases}$$

设 DFT 长度为 $N=8$, 则

$$X(k) = \sum_{n=0}^7 x(n) W_8^{kn} = \sum_{n=0}^3 e^{-j\frac{2\pi}{8}kn} = e^{-j\frac{3}{8}\pi k} \frac{\sin\left(\frac{\pi}{2}k\right)}{\sin\left(\frac{\pi}{8}k\right)}, \quad k=0, 1, \dots, 7$$

设 DFT 长度为 $N=16$, 则

$$X(k) = \sum_{n=0}^{15} x(n)W_{16}^{kn} = \sum_{n=0}^3 e^{-j\frac{2\pi}{16}kn} = e^{-j\frac{3}{16}\pi k} \frac{\sin\left(\frac{\pi}{4}k\right)}{\sin\left(\frac{\pi}{16}k\right)}, \quad k = 0, 1, \dots, 15$$

离散傅里叶变换的结果与指定的变换长度有关,运行结果如图 3.3.2 所示。

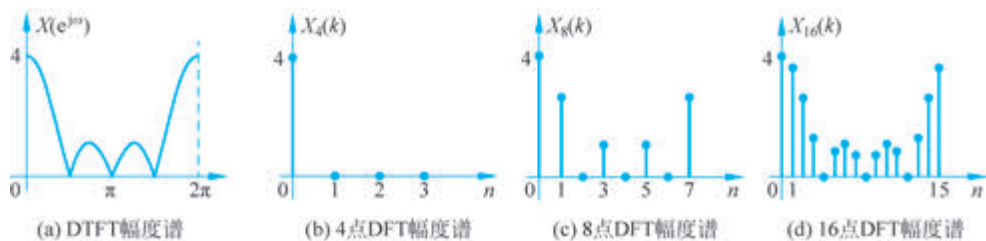


图 3.3.2 矩形序列 $R_4(n)$ 的 DTFT 和 DFT 的幅度谱

$x(n)$ 的 N 点 DFT 是 $x(n)$ 的 DTFT 在 $[0, 2\pi)$ 的 N 点等间隔采样,也是 DFS 的主值区间。图 3.3.2(b)~图 3.3.2(d)分别是图 3.3.2(a)DTFT 在 4 点、8 点、16 点的采样。 N 越大, $[0, 2\pi)$ 取得的采样值越多。

3.3.2 DFT 与 DTFT 及 Z 变换的关系

长度为 M 的序列 $x(n)$, 其 Z 变换为

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

$x(n)$ 的 DFT 为

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

比较 $X(z)$ 和 $X(k)$ 的变换式, 可得

$$X(k) = X(z) \Big|_{z=W_N^k = e^{j\frac{2\pi}{N}k}}, \quad k = 0, 1, \dots, N-1 \quad (3.3.4)$$

或者

$$X(k) = X(e^{j\omega}) \Big|_{\omega = \frac{2\pi}{N}k}, \quad k = 0, 1, \dots, N-1 \quad (3.3.5)$$

前面已介绍过, $x(n)$ 的 DTFT 是 $x(n)$ 的 Z 变换在单位圆上的值, $x(n)$ 的 N 点 DFT 是 $x(n)$ 的 Z 变换在单位圆上的 $[0, 2\pi)$ 内的 N 点等间隔采样值, 如式(3.3.4)所示。这也表明, $x(n)$ 的 N 点 DFT 是 $x(n)$ 的 DTFT 在 $[0, 2\pi)$ 上的 N 点等间隔采样值。此外, DFT 是 DFS 的主值区间, 是 Z 变换在单位圆上的等间隔采样值, 如图 3.1.4 所示。

3.3.3 DFT 的矩阵表示

式(3.3.1)和式(3.3.2)的 DFT 正逆变换也可以用矩阵运算表示。

\mathbf{x} 是 N 点时域序列的列向量, 即

$$\mathbf{x} = [x(0), x(1), x(2), \dots, x(N-2), x(N-1)]^T$$

\mathbf{X} 是 N 点 DFT 频域的列向量, 即

$$\mathbf{X} = [X(0), X(1), X(2), \dots, X(N-2), X(N-1)]^T$$

\mathbf{W}_N 是 N 点的 DFT 矩阵, 定义为

$$\mathbf{W}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^1 & W_N^2 & \cdots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix} \quad (3.3.6)$$

则式(3.3.1)可以表示为

$$\mathbf{X} = \mathbf{W}_N \mathbf{x} \quad (3.3.7)$$

即

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ \vdots \\ X(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^1 & W_N^2 & \cdots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(N-1) \end{bmatrix} = \mathbf{W}_N \mathbf{x} \quad (3.3.8)$$

\mathbf{W}_N^{-1} 是 N 点的 IDFT 矩阵, 定义为

$$\mathbf{W}_N^{-1} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \cdots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \cdots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \cdots & W_N^{-(N-1)(N-1)} \end{bmatrix} \quad (3.3.9)$$

式(3.3.2)也可以用矩阵表示

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

其中, $\mathbf{W}_N^{-1} = \frac{1}{N} \mathbf{W}_N^*$, 即

$$\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(N-1) \end{bmatrix} = \frac{1}{N} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N^{-1} & W_N^{-2} & \cdots & W_N^{-(N-1)} \\ 1 & W_N^{-2} & W_N^{-4} & \cdots & W_N^{-2(N-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & W_N^{-(N-1)} & W_N^{-2(N-1)} & \cdots & W_N^{-(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ \vdots \\ X(N-1) \end{bmatrix} \\ = \mathbf{W}_N^* \mathbf{X} / N \quad (3.3.10)$$

【例 3.3.3】 编程在 z 平面上画出 DFT 的旋转因子 $W_N^{nk} = e^{-j\frac{2\pi}{N}nk}$, 其中 $N=8$ 。

解 编写程序如下。

```
import numpy as np
import matplotlib.pyplot as plt
```

```

n = 8
plt.rc("font", family = 'Microsoft YaHei')
plt.figure(figsize = (3,3))
for k in range(0,n):
    z = np.exp(2 * np.pi * 1j * k/n)
    plt.plot(np.real(z), np.imag(z), 'o',ms = 10),
    plt.xlabel('实部'); plt.ylabel('虚部');plt.axis('square')
theta = np.linspace(0, np.pi * 2, 500)
circle = np.exp(1j * theta)
plt.plot(circle.real, circle.imag, 'k-- ')
plt.tight_layout();plt.show()

```

运行结果如图 3.3.3 所示。

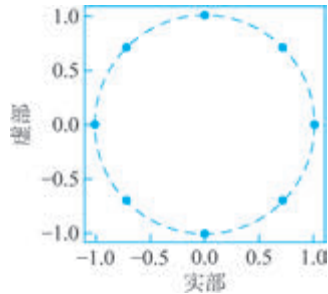


图 3.3.3 $N=8$ 的旋转因子 W_8^{nk}

$N=8$ 的旋转因子将单位圆分成了等间隔的 8 份。

【例 3.3.4】 求 $N=4$ 点的 DFT 矩阵。

解 (1) 编写 DFT 矩阵程序如下。

```

import numpy as np
N = 4; n = np.arange(0,N,1)
nk = np.outer(n, n)
WN = np.exp(-2j * np.pi * nk/N)
np.set_printoptions(precision = 4, suppress = True)
print('{}点的 DFT 矩阵:\n'.format(N), WN)

```

程序运行结果如下。

4 点的 DFT 矩阵:

```

[[ 1.+0.j 1.+0.j 1.+0.j 1.+0.j]
 [ 1.+0.j 0.-1.j -1.-0.j -0.+1.j]
 [ 1.+0.j -1.-0.j 1.+0.j -1.-0.j]
 [ 1.+0.j -0.+1.j -1.-0.j 0.-1.j]]

```

$$\mathbf{W}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

(2) 调用 SciPy. linalg 线性代数工具库中的函数 `dft()` 生成 DFT 矩阵, 并计算序列的 DFT。

编写程序如下。

```

from scipy.linalg import dft
import numpy as np

```

```

N = 4
x = np.ones(N)
WN = dft(N)
X = WN @ x
np.set_printoptions(precision = 4, suppress = True)
print(X)

```

程序运行结果如下。

```
[ 4. +0.j -0. -0.j  0. -0.j  0. -0.j]
```

(3) 求 IDFT 矩阵。

编写程序如下。

```

from scipy.linalg import dft
WN = dft(N)
IWN = WN.conj()/N

```

在已有 DFT 矩阵的情况下,利用 NumPy 库中的 `conj()` 方法取共轭可以获得 IDFT 矩阵。

3.3.4 DFT 隐含的周期性

由 DFS 引申出来的离散傅里叶变换,在时域、频域都是离散的。DFT 是针对有限长序列的,实际上,它是取周期序列的 DFS 的一个周期的对应关系来加以定义和分析的,下面将讨论 DFT 隐含的周期性。

离散傅里叶变换的频域为离散的,其频谱对应时域信号为周期的,时域序列信号以变换长度 N 为周期做周期延拓。在离散傅里叶变换中,有限长序列是作为周期序列的一个周期来表示的,隐含周期性的含义。

从 W_N^{kn} 的周期性 $W_N^{(k+mN)n} = W_N^{kn}$ 也可证明 $X(k)$ 隐含周期性,其周期为

$$X(k + mN) = \sum_{n=0}^{N-1} x(n) W_N^{(k+mN)n} = \sum_{n=0}^{N-1} x(n) W_N^{kn} = X(k) \quad (3.3.11)$$

因而,任何周期为 N 的周期序列 $\tilde{x}(n)$ 都可以看作长度为 N 的有限长序列 $x(n)$ 的周期延拓,而 $x(n)$ 是 $\tilde{x}(n)$ 的一个周期,即

$$\tilde{x}(n) = \sum_{m=-\infty}^{\infty} x(n + mN) \quad (3.3.12)$$

$$x(n) = \tilde{x}(n) R_N(n) \quad (3.3.13)$$

一般有限长序列 $x(n)$ 称为周期序列 $\tilde{x}(n)$ 在 $n=0 \sim N-1$ 的主值序列, $\tilde{x}(n)$ 是 $x(n)$ 的周期延拓序列。同样, $X(k)$ 是 $\tilde{X}(k)$ 的主值序列, $\tilde{X}(k)$ 是 $X(k)$ 的周期延拓序列。可以将 $\tilde{x}(n)$ 表示为

$$\tilde{x}(n) = x((n))_N \quad (3.3.14)$$

其中, $((n))_N$ 表示 n 对 N 取模,即若

$$n = mN + n_1, \quad 0 \leq n_1 \leq N-1, m \text{ 为整数}$$

则

$$((n))_N = n_1 \quad (3.3.15)$$

例如, $N=8$, $\tilde{x}(n) = x((n))_8$,则有, $\tilde{x}(8) = x((8))_8 = x(0)$, $\tilde{x}(9) = x((9))_8 = x(1)$ 。

对频域序列 $\tilde{X}(k)$ 也可以表示为

$$\tilde{X}(k) = \sum_{m=-\infty}^{\infty} X(k+rN) \quad (3.3.16)$$

$$X(k) = \tilde{X}(k)R_N(n) \quad (3.3.17)$$

周期延拓序列频谱 $\tilde{x}(n)$ 完全由其离散傅里叶级数系数 $\tilde{X}(k)$ 确定, 离散傅里叶变换对可以看作离散傅里叶级数变换对的主值区间。考虑 DFT 时, 要考虑隐含周期性的影响。

3.4 离散傅里叶变换的性质

3.4.1 线性

若两个有限长序列 $x_1(n)$ 和 $x_2(n)$ 的 DFT 分别为 $X_1(k)$ 和 $X_2(k)$, 长度分别为 N_1 和 N_2 , 则有

$$\text{DFT}[ax_1(n) + bx_2(n)] = aX_1(k) + bX_2(k) \quad (3.4.1)$$

其中, a 、 b 为常数。序列线性组合后的 DFT 长度 $N \geq \max[N_1, N_2]$, 即如果两序列不等长, 可以给短序列补零, 至少补到同长序列相同, 再进行 DFT 运算。

【例 3.4.1】 求两个序列 $x_1 = \{1, 1, 1, 1\}$, $x_2 = \{1, -2, 3, -4\}$ 相加后的 DFT, 验证结果等于它们各自 DFT 的相加。

解 编写如下程序验证 DFT 的线性。

```
import numpy as np
import matplotlib.pyplot as plt
x1 = [1, 1, 1, 1]
x2 = [1, -2, 3, -4]
a, b = 2, 3
x3 = np.add(np.multiply(a, x1), np.multiply(b, x2))
print(x3)
N = len(x2)
n = np.arange(0, N, 1)
k = np.arange(0, N, 1)
k1 = np.outer(n, k)
D = np.exp(-1j * 2 * np.pi * k1/N) # DFT matrix
X1 = np.dot(D, x1)
X2 = np.dot(D, x2)
X3 = np.dot(D, x3)
X4 = a * X1 + b * X2
plt.subplot(3, 2, 1),
plt.stem(n, x3, basefmt = '-- '), plt.xlabel('n'),
plt.title('x$_3$(n) = a * x$_1$(n) + b * x$_2$(n)')
plt.subplot(3, 2, 3)
plt.stem(k, abs(X3), basefmt = '-- '),
plt.xlabel('k'), plt.title('|X$_3$(k)|')
plt.subplot(3, 2, 4)
plt.stem(k, np.angle(X3), basefmt = '-- '),
plt.xlabel('k'), plt.title('\Phi$(X$_3$(k))')
plt.subplot(3, 2, 5)
plt.stem(k, abs(X4), basefmt = '-- '), plt.xlabel('k')
```

```

plt.title('a * X$_1$(k) + b * X$_2$(k)')
plt.subplot(3,2,6)
plt.stem(k, np.angle(X3), basefmt = '--')
plt.xlabel('k'), plt.title('$\Phi$(a * X$_1$(k) + b * X$_2$(k))')
plt.tight_layout()
plt.show()

```

程序结果如图 3.4.1 所示。

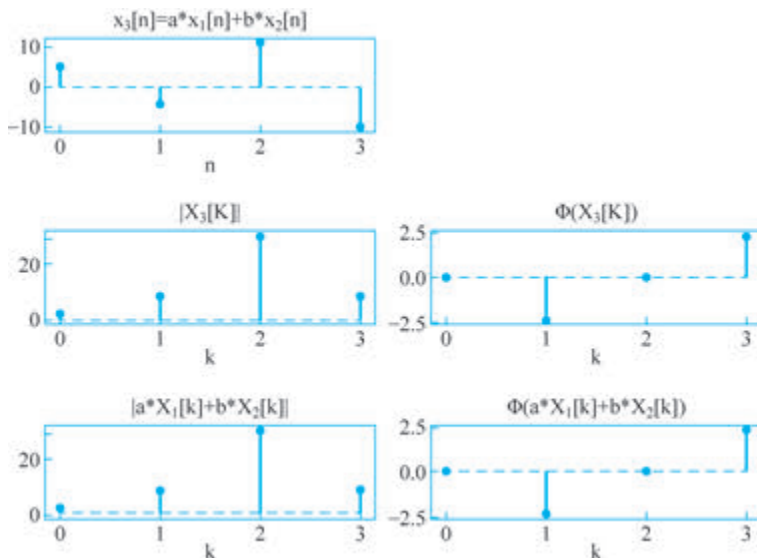


图 3.4.1 例 3.4.1 DFT 的线性性质

从图 3.4.1 中可以看出幅度响应和相位响应都满足线性叠加原理,即 $\text{DFT}[x_3(n)] = \text{DFT}[ax_1(n) + bx_2(n)] = a\text{DFT}[x_1(n)] + b\text{DFT}[x_2(n)]$ 。

3.4.2 圆周移位

因为 $x(n)$ 和 $h(n)$ 隐含周期性,所以在圆周卷积中,序列的翻褶和移位都是圆周翻褶和圆周移位。将序列 $x(n)$ 和 $h(n)$ 延拓成周期为 N 的周期序列 $\tilde{x}(n)$ 和 $\tilde{h}(n)$, 具体而言,即对其进行翻褶和移位后,取主值区间 $[0, N-1]$ 内的部分,即可得到圆周翻褶和圆周移位序列。这一过程可理解为将序列 $x(n)$ 或 $h(n)$ 均匀分布在一个 N 等分的圆上,移位操作就如同在这个圆上进行,如图 3.4.2 所示。

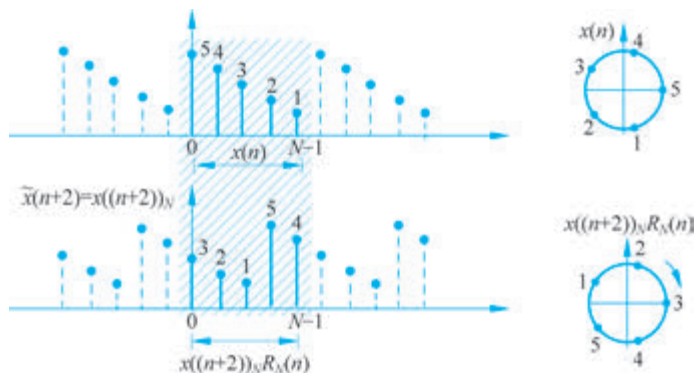
m 点的圆周移位可以表示为

$$x_m(n) = x((n+m))_N R_N(n) \quad (3.4.2)$$

$x((n+m))_N$ 表示 $x(n)$ 的周期延拓序列 $\tilde{x}(n)$, 也是一个周期为 N 的周期序列。 $x((n+m))_N R_N(n)$ 表示取主值区间,得到长度为 N 的有限长序列 $x_m(n)$ 。注意, $x_m(n)$ 是 $x(n)$ 的以 N 为周期的圆周移位,和线性移位 $x(n+m)$ 不相同。

从图 3.4.2 中可以观察到,由于隐含周期性,因此在圆周移位中,主值序列的采样值从一边移出后又从另一边补了进来,圆周移位的结果还是一个长度为 N 的有限长序列,相当于周期信号移位后取主值区间的结果。

m 点圆周移位后的序列用 $x_m(n)$ 表示,其 N 点的 DFT $X_m(k)$ 为

图 3.4.2 $x(n)$ 的圆周移位

$$\begin{aligned} X_m(k) &= \text{DFT}[x_m(n)] = \text{DFT}[x((n+m))_N R_N(n)] = \text{DFT}[\tilde{x}((n+m))_N R_N(n)] \\ &= W_N^{-mk} \tilde{X}(k) R_N(k) = W_N^{-mk} X(k) \end{aligned} \quad (3.4.3)$$

可以看到,有限长序列时域圆周移位后的 DFT,等于有限长序列的 DFT $X(k)$ 乘以复指数 ($W_N^{-mk} = e^{j\frac{2\pi}{N}km}$),即在频域引起了线性相移,相位移量为 $\frac{2\pi}{N}km$ 与 k, m 成正比, $|m|$ 越大,移动量越大。当然, $\frac{2\pi}{N}km$ 为实际相位增量,需要再对 2π 取模,映射到 $[-\pi, \pi]$ 内,而幅度频谱没有变化。

N 点有限长序列的 DFT 的长度同样是 N 点,在频域也可以做圆周移位

$$X_l(k) = X((k+l))_N R_N(k) \quad (3.4.4)$$

圆周移位后对应的时域序列为

$$x_l(n) = \text{IDFT}[X((k+l))_N R_N(k)] = W_N^{nl} x(n) = e^{-j\frac{2\pi}{N}ln} x(n) \quad (3.4.5)$$

频域的圆周移位对应时域乘以复指数信号 $e^{-j\frac{2\pi}{N}ln}$,这就是通信系统的调制特性。

【例 3.4.2】 序列 $x(n) = \{1, 2, 3, 4, 5\}$, $x(n)$ 的圆周移位序列为 $x_m(n) = x((n+m))_N R_N(n)$,当 $N=5, m=3$ 和 $m=-3$ 时,试验证 DFT 的圆周移位性质。

解 $m=3$ 时, $N=5$ 的圆周左移位序列为 $x_1(n) = \{4, 5, 1, 2, 3\}$;

$m=-3$ 时, $N=5$ 的圆周右移位序列为 $x_2(n) = \{3, 4, 5, 1, 2\}$ 。

编写程序如下。

```
import numpy as np
import matplotlib.pyplot as plt
def cirshift(x: np.array, m: int, N: int) -> np.array:
    if x.size > N:
        print("N must be greater than the size of x")
        return np.array([])
    x = np.concatenate((x, np.zeros(N - x.size)), axis=0)
    n = list(range(N))
    # n = (np.array(n) - m) % N
    n = np.mod(np.array(n) - m, N)
    return x[n]
if __name__ == "__main__":
    N = 5
    m = 3
    x = np.linspace(1, N, N)
```

```

x_mr = cirshift(x, m, N)
x_ml = cirshift(x, -m, N)
print("original sequence: ", x)
print("5 rounded circle shift to right by 3 steps: ", x_mr )
print("6 rounded circle shift to left by 3 steps: ", x_ml )
n = np.arange(0,N,1)
k = np.arange(0,N,1)
nk = np.outer(n,k)
WN = np.exp(-1j * 2 * np.pi * nk / N)
X1 = np.dot(WN, x) # DFT computation of x[n] X2 = np.dot(D, z) # D
X2 = np.dot(WN, x_mr)
X3 = np.dot(WN, x_ml)
plt.rc("font", family='Microsoft YaHei')
plt.figure(figsize=[12,4])
plt.subplot(3, 3, 1), plt.stem(n, x,basefmt='--')
plt.title('原信号')
plt.subplot(3, 3, 4), plt.stem(n, x_mr,basefmt='--')
plt.title('圆周右移序列')
plt.subplot(3, 3, 7), plt.stem(n, x_ml,basefmt='--'), plt.xlabel('n')
plt.title('圆周左移序列')
plt.subplot(3, 3, 2), plt.stem(n, np.abs(X1),basefmt='--')
plt.title('原信号的幅度谱')
plt.subplot(3, 3, 3), plt.stem(k, np.angle(X1),basefmt='--')
plt.title('原信号的相位谱')
plt.subplot(3, 3, 5), plt.stem(n, np.abs(X2),basefmt='--'),
plt.title('圆周右移序列的幅度谱')
plt.subplot(3, 3, 6), plt.stem(k, np.angle(X2),basefmt='--')
plt.title('圆周右移序列的相位谱')
plt.subplot(3, 3, 8), plt.stem(n, np.abs(X3),basefmt='--'),plt.xlabel('k')
plt.title('圆周左移序列的幅度谱')
plt.subplot(3, 3, 9), plt.stem(k, np.angle(X3),basefmt='--'), plt.xlabel('k')
plt.title('圆周左移序列的相位谱')
plt.tight_layout()
plt.show()

```

画出圆周移位 $m=3$ 和 -3 时的移位序列的幅度谱和相位谱,如图 3.4.3 所示。

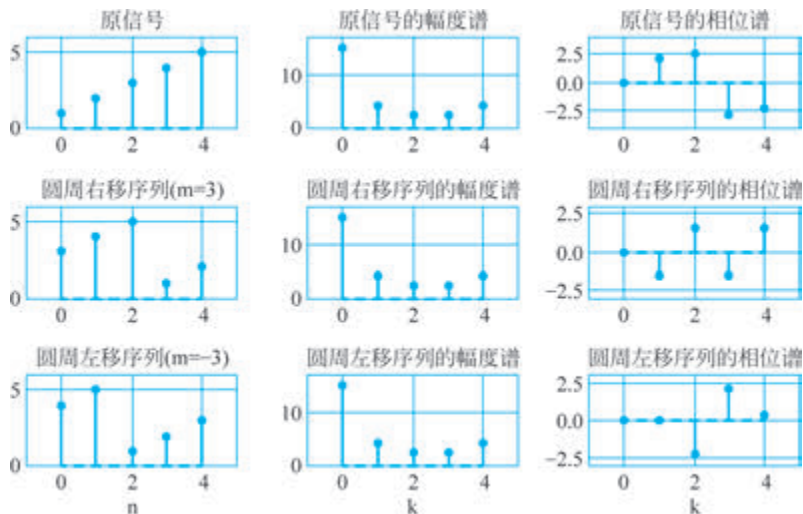


图 3.4.3 原序列和圆周左移 ($m=3$) 和右移 ($m=-3$) 序列的幅度谱与相位谱

从图 3.4.3 中可以看到,圆周移位左移 3 位和右移 3 位,从一边移出的序列从另一边补了进来,幅度频谱没有变化,相位频谱发生了变化,变化量为 $\frac{2\pi}{N} \times 3k$ ($k=0,1,\dots,4$) 映射到 $[-\pi,\pi]$ 内的值。圆周移位性质得以验证。

3.4.3 圆周翻褶

圆周翻褶序列是周期性序列翻褶后取主值序列的一种形式,即对于序列 $x(n)$ ($n=0,1,\dots,N-1$),其圆周翻褶序列定义为

$$x((-n))_N R_N(n) = x((N-n))_N R_N(n) = x(N-n) \quad (3.4.6)$$

相当于保持序列 $x(n)$ 的 $x(0)$ 值不动,同时将序列第 n 个值和第 $N-n$ 个值互换 ($n=1,\dots,\lfloor \frac{N}{2} \rfloor$),从而得到圆周翻褶序列 $x((-n))_N R_N(n) = x(N-n)$ 。

翻褶序列的 DFT 为

$$\text{DFT}[x((-n))_N R_N(n)] = X((-k))_N R_N(k) \quad (3.4.7)$$

即

$$\text{DFT}[x(N-n)] = X(N-k) \quad (3.4.8)$$

【例 3.4.3】 编程计算 $x(n) = \{2, 3, 4, 5, 6\}$ 的周期为 8 的圆周翻褶序列。程序如下。

```
import numpy as np
import matplotlib.pyplot as plt
def cirflip(x: np.array, N:int) -> np.array:
    x = np.concatenate((x, np.zeros(N - x.size)))
    nx = np.linspace(0, N - 1, N, dtype = np.int32)
    return x[(-nx % N)]
x = np.array([2, 3, 4, 5, 6])
N = 8
y = cirflip(x, N)
ny = np.linspace(0, y.size - 1, y.size)
x = np.concatenate((x, np.zeros(N - x.size)))
nx = np.linspace(0, x.size - 1, x.size)
plt.rc("font", family = 'Microsoft YaHei', size = 14)
plt.figure()
plt.subplot(211)
plt.stem(nx, np.real(x), basefmt = '--');plt.title("原序列")
plt.subplot(212)
plt.stem(ny, np.real(y), basefmt = '--');plt.title("圆周翻褶序列")
plt.tight_layout();plt.show()
```

解 程序运行结果如图 3.4.4 所示。圆周翻褶序列 $x(n) = \{2, 0, 0, 0, 6, 5, 4, 3\}$ 。

【例 3.4.4】 编程计算序列 $x(n) = e^{(0.4+0.6j)n}$ ($n=0,1,\dots,5$) 在 $N=9$ 时的圆周翻褶序列。

解 编程计算画出原序列和圆周翻褶的实部和虚部,程序如下。

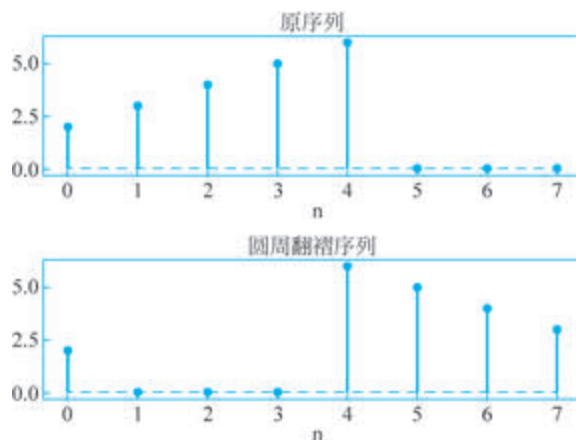


图 3.4.4 原序列和圆周翻褶序列

```

import numpy as np
import matplotlib.pyplot as plt
from scipy.linalg import dft
def cirflip(x: np.array, N: int) -> np.array:
    x = np.concatenate((x, np.zeros(N - x.size)))
    nx = np.linspace(0, N - 1, N, dtype=np.int32)
    return x[(-nx % N)]
n = np.arange(0, 6); x = np.e ** ((.4 + 0.6j) * n)
N = 9;
y = cirflip(x, N); WNy = dft(N); Y = WNy@y
print(Y)
ny = np.linspace(0, y.size - 1, y.size)
x = np.concatenate((x, np.zeros(N - x.size)))
WN = dft(len(x)); X = WN@x
print(X)
X = np.concatenate((X, np.zeros(N - X.size)))
nx = np.linspace(0, x.size - 1, x.size)
plt.rc("font", family='Microsoft YaHei', size=14)
plt.figure()
plt.subplot(211); plt.stem(nx, np.real(x), basefmt='--')
plt.title("原序列的实部")
plt.subplot(212); plt.stem(ny, np.real(y), basefmt='--')
plt.title("圆周翻褶序列的实部")
plt.tight_layout(); plt.rc("font", family='Microsoft YaHei')
plt.figure()
plt.subplot(211); plt.stem(nx, np.imag(x), basefmt='--')
plt.title("原序列的虚部")
plt.subplot(212); plt.stem(ny, np.imag(y), basefmt='--')
plt.title("圆周翻褶序列的虚部"); plt.tight_layout()
plt.figure()
plt.subplot(2,1,1); plt.stem(abs(X), basefmt='--')
plt.title("原序列 DFT 的幅度")
plt.subplot(2,1,2); plt.stem(abs(Y), basefmt='--')
plt.title("圆周翻褶序列 DFT 的幅度"); plt.tight_layout()
plt.figure()
plt.subplot(2,1,1); plt.stem(np.angle(X), basefmt='--')
plt.title("原序列 DFT 的相位")
plt.subplot(2,1,2); plt.stem(np.angle(Y), basefmt='--')
plt.title("圆周翻褶序列 DFT 的相位")
plt.figure()

```

对比原序列和圆周翻褶后 DFT 的幅度和相位,如图 3.4.5 所示。

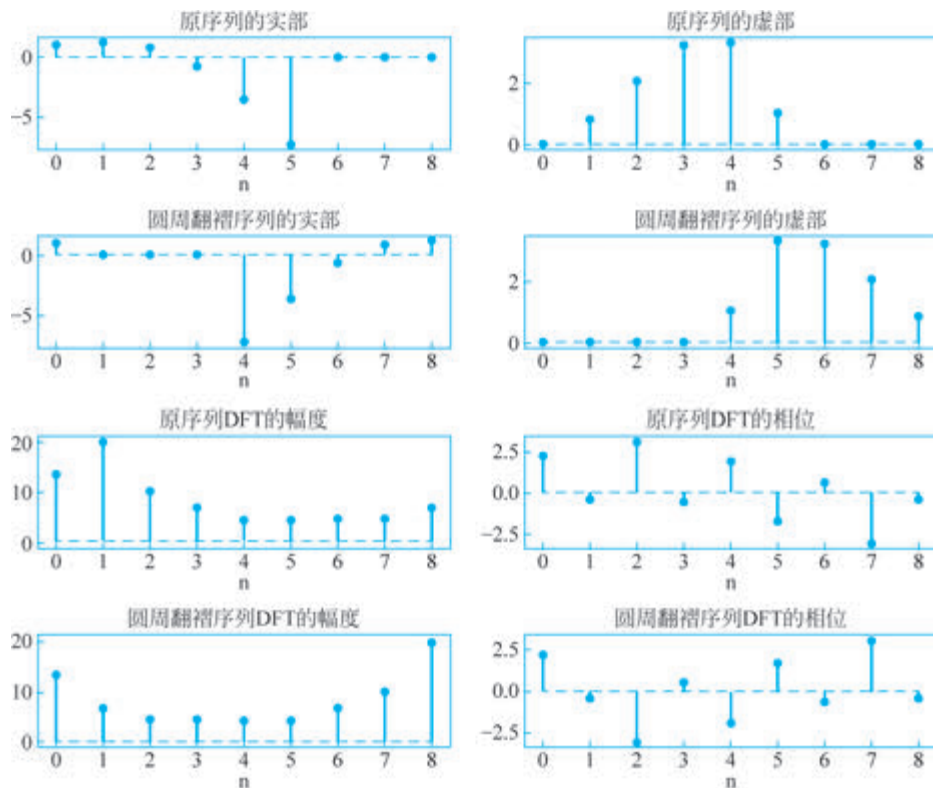


图 3.4.5 原序列和圆周翻褶序列及它们 DFT 的对比图

从图 3.4.5 中可以看到,序列 $x(0)$ 值保持不变,圆周翻褶序列是原序列以 $n = \frac{N}{2}$ 为对称中心翻褶后得到的。其幅度谱和相位谱均为原序列 DFT 在频域上(以 N 为周期)进行圆周翻褶的结果,即 $X(0)$ 值也保持不变,且这些谱也以 $n = \frac{N}{2}$ 为对称中心翻褶,从而说明圆周翻褶序列的 DFT 是原序列 DFT 的圆周翻褶。

3.4.4 圆周共轭对称性

1. 圆周共轭对称序列定义

离散傅里叶变换中的时域和频域序列都是长度为 N 的有限序列,且有周期性的含义。在考虑有限长序列的共轭对称性和共轭反对称性时,需特别注意这种隐含周期性的影响。因此,可定义有限长序列的圆周共轭对称序列和圆周共轭反对称序列。

任一周期性序列 $\tilde{x}(n)$ 都可表示成周期性共轭对称分量 $\tilde{x}_e(n)$ 及周期性共轭反对称分量 $\tilde{x}_o(n)$ 之和,即

$$\tilde{x}(n) = \tilde{x}_e(n) + \tilde{x}_o(n) \quad (3.4.9)$$

其中,共轭对称分量为

$$\tilde{x}_e(n) = \frac{1}{2} [\tilde{x}(n) + \tilde{x}^*(-n)] \quad (3.4.10)$$

共轭反对称分量为

$$\tilde{x}_o(n) = \frac{1}{2} [\tilde{x}(n) - \tilde{x}^*(-n)] \quad (3.4.11)$$

分别取它们的主值序列可以得到圆周共轭对称序列为

$$x_{ep}(n) = \tilde{x}_e(n)R_N(n) \quad (3.4.12)$$

圆周共轭反对称序列为

$$x_{op}(n) = \tilde{x}_o(n)R_N(n) \quad (3.4.13)$$

它们分别满足圆周共轭对称和圆周共轭反对称关系,即

$$x_{ep}(n) = x_{ep}^*((-n))_N R_N(n) = x_{ep}^*(N-n), \quad n=1,2,\dots,N-1 \quad (3.4.14)$$

$$x_{op}(n) = -x_{op}^*((-n))_N R_N(n) = -x_{op}^*(N-n), \quad n=1,2,\dots,N-1 \quad (3.4.15)$$

圆周共轭对称和反对称序列之和就可以组成有限长序列 $x(n)$,即

$$x(n) = x_{ep}(n) + x_{op}(n) \quad (3.4.16)$$

因此,任意有限长序列 $x(n)$ 可以表示为圆周共轭对称分量 $x_{ep}(n)$ 和圆周共轭反对称分量 $x_{op}(n)$ 之和。其中,

$$\begin{aligned} x_{ep}(n) &= \tilde{x}_e(n)R_N(n) = \frac{1}{2} [\tilde{x}(n) + \tilde{x}^*(-n)] R_N(n) \\ &= \frac{1}{2} [x((n))_N + x^*((-n))_N] R_N(n) \\ &= \frac{1}{2} [x(n) + x^*(N-n)], \quad n=0,1,2,\dots,N-1 \end{aligned} \quad (3.4.17)$$

$$\begin{aligned} x_{op}(n) &= \tilde{x}_o(n)R_N(n) = \frac{1}{2} [\tilde{x}(n) - \tilde{x}^*(-n)] R_N(n) \\ &= \frac{1}{2} [x((n))_N - x^*((-n))_N] R_N(n) \\ &= \frac{1}{2} [x(n) - x^*(N-n)], \quad n=0,1,2,\dots,N-1 \end{aligned} \quad (3.4.18)$$

这里的 $x_{ep}(n)$ 与 $x_{op}(n)$ 都是与 $x(n)$ 长度相同(为 N)的有限长序列,且同样隐含周期性。与第2章中介绍的共轭(反)对称 $x_e(n)$ 与 $x_o(n)$ 不同。若 $x(n)$ 长度为 N ,则由于线性翻褶, $x_e(n)$ 与 $x_o(n)$ 的长度为 $2N-1$ 点,与隐含周期性的 $x_{ep}(n)$ 与 $x_{op}(n)$ 长度不同。

2. DFT 的圆周对称性

对于有限长序列 $x(n)$,其 DFT 为 $X(k)$, $x_{ep}(n)$ 、 $x_{op}(n)$ 分别表示 $x(n)$ 的共轭对称和共轭反对称序列, $X_{ep}(k)$ 、 $X_{op}(k)$ 分别表示 $X(k)$ 的共轭对称和共轭反对称序列,则它们满足如表 3.4.1 所示的共轭对称性。

表 3.4.1 一般序列 DFT 的共轭对称性

时域序列	时域序列的 DFT
$x(n)$	$X(k)$
$x(n)$ 的实部	$X_{ep}(k)$
$x(n)$ 的虚部乘以 j	$X_{op}(k)$
$x_{ep}(n)$	$X(k)$ 的实部
$x_{op}(n)$	$X(k)$ 的虚部乘以 j

证明如下。

将 $x(n)$ 表示为实部和虚部的形式, 即

$$x(n) = \text{Re}[x(n)] + j\text{Im}[x(n)]$$

其中, 实部为

$$\text{Re}[x(n)] = \frac{1}{2} [x(n) + x^*(n)]$$

则有

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

同理, 虚部乘以 j 为

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

则有

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

由于

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

则有

$$\text{DFT}[x_{\text{ep}}(n)] = \frac{1}{2}\text{DFT}[x(n) + x^*(N-n)] = \frac{1}{2} [X(k) + X^*(k)] = \text{Re}[X(k)] \quad (3.4.19)$$

同样, 由于

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

则有

$$\text{DFT}[x_{\text{op}}(n)] = \frac{1}{2}\text{DFT}[x(n) - x^*(N-n)] = \frac{1}{2} [X(k) - X^*(k)] = j\text{Im}[X(k)] \quad (3.4.20)$$

一般序列 DFT 的共轭对称性得证。

如果 $x(n)$ 为实数序列, 那么表 3.4.1 中的对称性可进一步简化, 如表 3.4.2 所示。

表 3.4.2 实数序列 DFT 的共轭对称性

时域序列	时域序列的 DFT
$x(n)$ 为实数序列	$X(k) = X_{\text{ep}}(k)$
$x(n)$ 的虚部为零	$X_{\text{op}}(k) = 0$
$x_{\text{ep}}(n)$	$X(k)$ 的实部
$x_{\text{op}}(n)$	$X(k)$ 的虚部乘以 j

如果 $x(n)$ 为纯虚序列, 那么其共轭对称性如表 3.4.3 所示。

表 3.4.3 纯虚序列 DFT 的共轭对称性

时域序列	时域序列的 DFT
$x(n)$ 为纯虚序列	$X(k) = X_{\text{op}}(k)$
$x(n)$ 的实部为零	$X_{\text{ep}}(k) = 0$
$x_{\text{ep}}(n)$	$X(k)$ 的实部
$x_{\text{op}}(n)$	$X(k)$ 的虚部乘以 j

利用共轭对称性,可以减少 DFT 的计算量,方便求解序列的 DFT。

3.4.5 序列的圆周卷积定理

1. 圆周卷积的定义

设序列 $x(n)$ 和 $h(n)$ 的长度分别为 L 和 M 。 $x(n)$ 和 $h(n)$ 在 N 点的圆周卷积定义为

$$y_c(n) = x(n) \otimes h(n) = \left[\sum_{m=0}^{N-1} h(m)x((n-m))_N \right] R_N(n), N \geq \max[L, M] \quad (3.4.21)$$

其中, N 为圆周卷积的区间长度,符号 \otimes 表示 N 点的圆周卷积。 $x(n)$ 和 $h(n)$ 交换前后顺序,圆周卷积结果不变。

圆周卷积可以用矩阵来计算,式(3.4.21)中 $x((n-m))_N R_N(n)$ 的矩阵表示为

$$\begin{bmatrix} x(0) & x(N-1) & x(N-2) & \cdots & x(1) \\ x(1) & x(0) & x(N-1) & \cdots & x(2) \\ x(2) & x(1) & x(0) & \cdots & x(3) \\ \vdots & \vdots & \vdots & & \vdots \\ x(N-1) & x(N-2) & x(N-3) & \cdots & x(0) \end{bmatrix} \quad (3.4.22)$$

该矩阵称为 $x(n)$ 的 N 点圆周卷积矩阵。其第一行是 $x(n)$ 的 N 点圆周翻褶序列,若 $x(n)$ 的长度小于 N ,则需要在 $x(n)$ 的末尾补 $N-L$ 个零以作为第一行的圆周翻褶序列。其他各行是第一行的圆周右移序列,每向下一行,圆周右移 1 位。根据式(3.4.22),式(3.4.21)可以写成矩阵形式,即

$$\begin{bmatrix} y_c(0) \\ y_c(1) \\ y_c(2) \\ \vdots \\ y_c(N-1) \end{bmatrix} = \begin{bmatrix} x(0) & x(N-1) & x(N-2) & \cdots & x(1) \\ x(1) & x(0) & x(N-1) & \cdots & x(2) \\ x(2) & x(1) & x(0) & \cdots & x(3) \\ \vdots & \vdots & \vdots & & \vdots \\ x(N-1) & x(N-2) & x(N-3) & \cdots & x(0) \end{bmatrix} \begin{bmatrix} h(0) \\ h(1) \\ h(2) \\ \vdots \\ h(N-1) \end{bmatrix} \quad (3.4.23)$$

同样,如果 $h(n)$ 的长度小于 N ,则需要在尾部先补零,补到 N 点后,再写成圆周卷积矩阵。可以在计算机上利用矩阵运算计算两序列的圆周卷积。 N 取值不同,周期延拓不同,圆周卷积的结果也不同。

参与圆周卷积的两个序列长度应为圆周卷积的点数 N ,若长度不同,则可以用补零值



的方法,补到相同的长度 N 。圆周卷积的结果长度也为 N 。由于有限长序列圆周卷积隐含周期性的含义,因此圆周卷积实际上是周期卷积的主值区间。

【例 3.4.5】 采用矩阵法计算下面两个序列的 4 点和 8 点的圆周卷积。 $x(n) = \{1, 2, 3, 4\}$, $h(n) = \{1, 1, 1\}$ 。

解 按照式(3.4.23)写出 $x(n)$ 和 $h(n)$ 的 4 点圆周卷积为

$$\begin{bmatrix} y_c(0) \\ y_c(1) \\ y_c(2) \\ y_c(3) \end{bmatrix} = \begin{bmatrix} 1 & 4 & 3 & 2 \\ 2 & 1 & 4 & 3 \\ 3 & 2 & 1 & 4 \\ 4 & 3 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 8 \\ 7 \\ 6 \\ 9 \end{bmatrix}$$

$$y(n) = x(n) \textcircled{4} h(n) = \{8, 7, 6, 9\}$$

$x(n)$ 和 $h(n)$ 的 8 点圆周卷积为

$$\begin{bmatrix} y_c(0) \\ y_c(1) \\ y_c(2) \\ y_c(3) \\ y_c(4) \\ y_c(5) \\ y_c(6) \\ y_c(7) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 4 & 3 & 2 \\ 2 & 1 & 0 & 0 & 0 & 0 & 4 & 3 \\ 3 & 2 & 1 & 0 & 0 & 0 & 0 & 4 \\ 4 & 3 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 4 & 3 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 4 & 3 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 4 & 3 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ 6 \\ 9 \\ 7 \\ 4 \\ 0 \\ 0 \end{bmatrix}$$

$$y(n) = x(n) \textcircled{8} h(n) = \{1, 3, 6, 9, 7, 4, 0, 0\}$$

8 点的圆周卷积中, $x(n)$ 和 $h(n)$ 需要补零到 8 个长度 ($n=0, 1, \dots, 7$), 才能进行 8 点的圆周卷积。圆周卷积过程如图 3.4.6 所示, 其中虚线表示隐含周期性的序列值。圆周卷积由于隐含周期性, 它实际上是周期卷积在一个周期内的主值区间。

2. 圆周卷积定理

设有限长序列 $x_1(n)$ 的长度为 N_1 , $x_2(n)$ 的长度为 N_2 , 取 $N \geq \max[N_1, N_2]$, 将两个序列补到 N 个长度, 做 N 点的圆周卷积, 即

$$x(n) = x_1(n) \textcircled{N} x_2(n) = \left[\sum_{m=0}^{N-1} x_1(m) x_2((n-m))_N \right] R_N(n) \quad (3.4.24)$$

则 $x(n)$ 的离散傅里叶变换 $X(k)$ 为

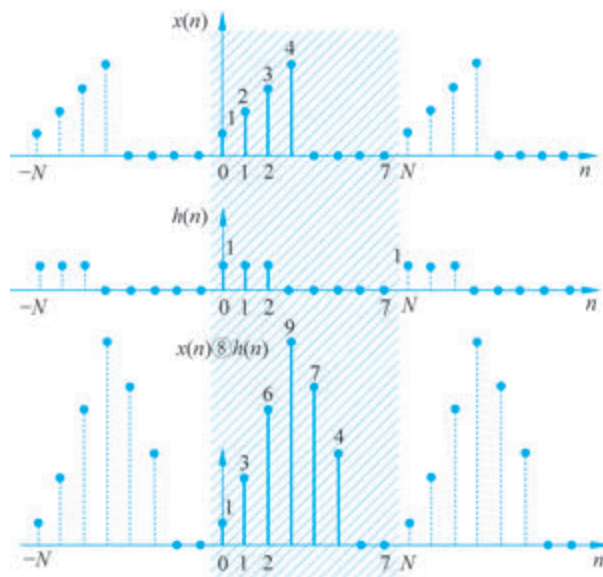
$$X(k) = \text{DFT}[x(n)] = X_1(k) X_2(k) \quad (3.4.25)$$

其中, $X_1(k) = \text{DFT}[x_1(n)]$, $X_2(k) = \text{DFT}[x_2(n)]$, $k=0, 1, \dots, N-1$ 。

证明如下。

对式(3.4.24)两边进行 DFT 运算, 则有

$$\begin{aligned} X(k) &= \text{DFT}[x(n)] = \sum_{n=0}^{N-1} \left[\sum_{m=0}^{N-1} x_1(m) x_2((n-m))_N \right] R_N(n) W_N^{kn} \\ &= \sum_{m=0}^{N-1} x_1(m) \left[\sum_{n=0}^{N-1} x_2((n-m))_N \right] W_N^{kn} \end{aligned}$$


 图 3.4.6 $x(n)$ 和 $h(n)$ 的圆周卷积(隐含周期性)

令 $n-m=n'$, 则有

$$\begin{aligned} X(k) &= \sum_{m=0}^{N-1} x_1(m) \left[\sum_{n=-m}^{N-1-m} x_2((n'))_N \right] W_N^{k(n'+m)} \\ &= \sum_{m=0}^{N-1} x_1(m) W_N^{km} \sum_{n=-m}^{N-1-m} x_2((n'))_N W_N^{kn'} \\ &= \sum_{m=0}^{N-1} x_1(m) W_N^{km} \sum_{n=-m}^{N-1-m} x_2((n'))_N W_N^{kn'} \end{aligned}$$

因为 $x_2((n'))_N W_N^{kn'}$ 是以 N 为周期的, 所以对其在任一个周期上求和的结果不变。因此

$$\begin{aligned} X(k) &= \sum_{m=0}^{N-1} x_1(m) W_N^{km} \sum_{n=0}^{N-1} x_2(n') W_N^{kn'} \\ &= X_1(k) X_2(k), \quad 0 \leq k \leq N-1 \end{aligned} \quad (3.4.26)$$

此定理说明, 对于两个序列若其在时域作圆周卷积, 则在离散频域中 DFT 作相乘运算。

圆周卷积也满足交换律, 即

$$X(k) = \text{DFT}[x(n)] = X_1(k) X_2(k) = X_2(k) X_1(k)$$

因此

$$x(n) = \text{IDFT}[X(k)] = x_1(n) \otimes x_2(n) = x_2(n) \otimes x_1(n), \quad 0 \leq k \leq N-1 \quad (3.4.27)$$

下面给出频域卷积定理, 若 $x(n) = x_1(n)x_2(n)$, 则

$$\begin{aligned} X(k) &= \text{DFT}[x(n)]_N = \frac{1}{N} X_1(k) \otimes X_2(k) \\ &= \frac{1}{N} \left[\sum_{l=0}^{N-1} X_1(l) X_2((k-l))_N \right] R_N(n) \end{aligned} \quad (3.4.28)$$

其中, $X_1(k) = \text{DFT}[x_1(n)]$, $X_2(k) = \text{DFT}[x_2(n)]$, $k=0, 1, \dots, N-1$ 。

频域卷积定理可总结为两个时域序列乘积的 DFT 等于各序列 DFT 在频域的圆周卷积的 $\frac{1}{N}$ 。

3. 圆周卷积与线性卷积的关系

(1) 圆周卷积要求参与的两个序列长度相同,若不相同,则采用补零的方法,使其长度相同;而线性卷积对两序列长度没有要求。

(2) 圆周卷积要先对其中一个参与卷积的序列做翻褶,再做圆周移位;而线性卷积是做线性移位。

(3) 圆周卷积和的长度为补零后的序列长度 N ,而线性卷积的结果长度为 $N_1 + N_2 - 1$ 。两者长度不同。

(4) 比较线性卷积计算公式和圆周卷积计算公式,可以发现,当圆周卷积长度 N 大于或等于线性卷积长度 $N_1 + N_2 - 1$ 时,圆周卷积结果与线性卷积结果相同。

将序列 $x_2(n)$ 做周期延拓,即

$$\tilde{x}_2(n) = x_2((n))_N = \sum_{r=-\infty}^{\infty} x_2(n + rN)$$

$x_1(n)$ 和 $x_2(n)$ 的圆周卷积为

$$\begin{aligned} y_c(n) &= x_1(n) \circledast x_2(n) = \left[\sum_{m=0}^{N-1} x_1(m) x_2((n-m))_N \right] R_N(n) \\ &= \left[\sum_{m=0}^{N-1} x_1(m) \sum_{r=-\infty}^{\infty} x_2(n + rN - m) \right] R_N(n) \\ &= \left[\sum_{r=-\infty}^{\infty} \sum_{m=0}^{N-1} x_1(m) x_2(n + rN - m) \right] R_N(n) \\ &= \left[\sum_{r=-\infty}^{\infty} y_1(n + rN) \right] R_N(n) \end{aligned} \quad (3.4.29)$$

由此看出, N 点的圆周卷积 $y_c(n)$ 是周期卷积的主值序列,是线性卷积 $y_1(n)$ 以 N 为周期的周期延拓序列的主值序列。线性卷积 $y_1(n)$ 的长度为 $N_1 + N_2 - 1$ 。因此,当 $N \geq N_1 + N_2 - 1$ 时,线性卷积 $y_1(n)$ 以 N 为周期进行周期延拓无混叠现象,即

$$\begin{aligned} y_c(n) &= x_1(n) \circledast x_2(n) = x_1(n) * x_2(n) = y_1(n), \\ N &\geq N_1 + N_2 - 1 \text{ 且 } 0 \leq n \leq N_1 + N_2 - 2 \end{aligned} \quad (3.4.30)$$

因此,当 $N \geq N_1 + N_2 - 1$ 时,两序列在 N 点的圆周卷积可以代替线性卷积。

当 $N < N_1 + N_2 - 1$,即圆周卷积长度 N 小于线性卷积长度 $N_1 + N_2 - 1$ 时,若线性卷积以 N 为周期进行周期延拓,序列首尾则会出现混叠现象,如图 3.4.7 所示,设 $M = N_1 + N_2 - 1$ 为线性卷积的长度,则在圆周卷积序列中,只有 $M - N \leq n \leq N - 1$ 时的序列值和线性卷积序列值相同,在这一范围内,圆周卷积可以代替线性卷积。

【例 3.4.6】 已知有限长序列 $x(n) = \{1, 2, 3, 4\}$, $h(n) = \{1, 1, 1\}$ 。讨论两个序列的线性卷积与其圆周卷积的关系。

解 $x(n)$ 是长度为 $N_1 = 4$ 的序列, $h(n)$ 是长度为 $N_2 = 3$ 的序列,则它们的线性卷积计算公式为

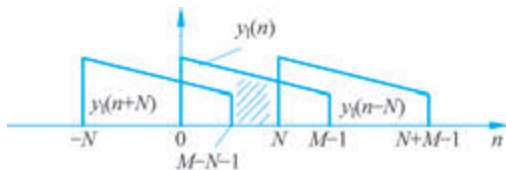


图 3.4.7 当 $N < N_1 + N_2 - 1$ 时, 线性卷积与 N 点圆周卷积的关系

$$y_1(n) = \sum_{m=0}^{N_1+N_2-1} x(m)h(n-m)$$

对有限长序列的线性卷积, 可以采用对位相乘法或列表法比较方便。对位相乘法结果如下。

$x(n)$	<u>1</u>	2	3	4
$h(n)$		<u>1</u>	1	1
		1	2	3
		1	2	3
	<u>1</u>	2	3	4
$y_1(n)$	<u>1</u>	3	6	9
				7
				4

则 $y_1(n) = [1, 3, 6, 9, 7, 4]$ 。

由于 N 点的圆周卷积 $y_c(n)$ 是线性卷积 $y_1(n)$ 以 N 为周期进行周期延拓后, 再取主值序列得到的, 因此可以根据这一过程计算出圆周卷积结果。

例如, 4 点的圆周卷积是将线性卷积以 4 为周期做周期延拓后, 再叠加的结果。过程如下。

n		0	1	2	3	4	5
$y_1(n+4)$...	6	9	7	4		
$y_1(n)$		<u>1</u>	3	6	9	7	4
$y_1(n-4)$						1	3
$y_4(n) = x(n) \textcircled{4} h(n)$			<u>8</u>	7	6	9	

因此 $y_4(n) = x(n) \textcircled{4} h(n) = [8, 7, 6, 9]$, 同理, 可以得到其他长度的圆周卷积与线性卷积的关系。从图 3.4.8 所示的圆周卷积过程可以看到, 当 $N \geq 6$ 以后, 圆周卷积就可以代替线性卷积了。

3.4.6 序列的圆周相关定理

设有限长序列 $x(n)$ 的长度为 N_1 , $y(n)$ 的长度为 N_2 , 取 $N \geq \max[N_1, N_2]$, 将两个序列补到长度为 N , 做 N 点的 DFT 得到 $X(k)$ 和 $Y(k)$, 若

$$R_{xy}(k) = X(k)Y(N-k) = X(k)Y^*(k)$$

则圆周相关序列 $\bar{r}_{xy}(m)$ 为

$$\bar{r}_{xy}(m) = \text{IDFT}[R_{xy}(k)] = \sum_{n=0}^{N-1} x(n)y((n-m))_N R_N(m) \quad (3.4.31)$$

证明如下。

先将 $R_{xy}(k)$ 延拓为周期序列, 即

$$\tilde{R}_{xy}(k) = \tilde{X}(k)\tilde{Y}^*(k)$$

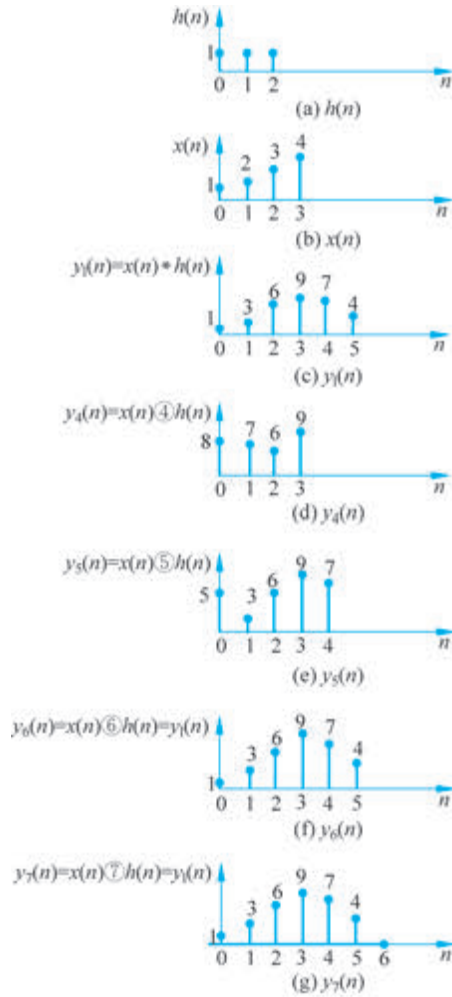


图 3.4.8 有限长序列的圆周卷积和线性卷积

则

$$\begin{aligned}
 \tilde{r}_{xy}(m) &= \text{IDFS} [\tilde{R}_{xy}(k)] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{Y}^*(k) \tilde{X}(k) W_N^{-m} \\
 &= \frac{1}{N} \sum_{k=0}^{N-1} \tilde{Y}^*(k) \sum_{n=0}^{N-1} \tilde{x}(n) W_N^{nk} W_N^{-mk} = \sum_{n=0}^{N-1} \tilde{x}(n) \frac{1}{N} \sum_{k=0}^{N-1} \tilde{Y}^*(k) W_N^{(n-m)k} \\
 &= \sum_{n=0}^{N-1} \tilde{x}(n) \tilde{y}^*(n-m) = \sum_{n=0}^{N-1} \tilde{y}^*(n) \tilde{x}(n+m)
 \end{aligned}$$

取主值序列,即

$$\begin{aligned}
 \bar{r}_{xy}(m) &= \sum_{n=0}^{N-1} y^*(n) x((n+m))_N R_N(n) \\
 &= \sum_{n=0}^{N-1} x(n) y^*((n-m))_N R_N(n)
 \end{aligned}$$

当 $x(n)$ 与 $y(n)$ 为实序列时,有

$$\bar{r}_{xy}(m) = \text{IDFT} [R_{xy}(k)] = \sum_{n=0}^{N-1} x(n)y((n-m))_N R_N(m)$$

当 $y(n)$ 也为 $x(n)$ 时, 序列为自相关序列 $r_{xx}(m)$, 满足

$$\bar{r}_{xx}(m) = \sum_{n=0}^{L-1} x(n)x((n-m))_L R_L(m)$$

类似于圆周卷积和线性卷积之间的关系。当 $N \geq N_1 + N_2 - 1$ 时, 圆周相关可以代表线性相关。

3.4.7 对称性

序列 $x(n)$ 的傅里叶变换为 $X(k)$, 即

$$\text{DFT}[x(n)] = X(k)$$

若另一个时域序列的形式与 $X(k)$ 相同, 记为 $X(n)$, 则 $X(n)$ 的离散傅里叶变换形式与 $x(n)$ 函数形式相同, 即

$$\text{DFT}[X(n)] = Nx((-k))_N R_N(k) = Nx((N-k))_N R_N(k) = Nx(N-k) \quad (3.4.32)$$

证明如下。

根据傅里叶级数的对称性, DFT 是 DFS 的主值区间, 因此由式(3.1.15)得

$$\text{DFT}[\tilde{X}(n)R_N(n)] = N\tilde{x}(-k)R_N(k) = Nx((-k))_N R_N(k)$$

取主值区间, 可得

$$\text{DFT}[X(n)] = Nx((-k))_N R_N(k) = Nx((N-k))_N R_N(k) = Nx(N-k)$$

3.4.8 帕塞瓦尔定理

若长度为 N 的序列 $x(n)$ 在 N 点处的 DFT 为 $X(k)$, 则有

$$\sum_{n=0}^{N-1} x(n)y^*(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)Y^*(k) \quad (3.4.33)$$

当 $x(n)=y(n)$ 时, 有

$$\sum_{n=0}^{N-1} |x(n)|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2 \quad (3.4.34)$$

若 $x(n)=y(n)$ 都是实序列, 则有

$$\sum_{n=0}^{N-1} x^2(n) = \frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2 \quad (3.4.35)$$

虽然信号形式不同, 但是信号在一个域的能量与另一个域相等, 即能量守恒原理。

DFT 的主要性质总结如表 3.4.4 所示。

表 3.4.4 DFT 的主要性质总结

序 列	离散傅里叶变换
$x(n)$	$X(k)$
$ax_1(n)+bx_2(n)$	$aX_1(k)+bX_2(k)$
$x((n+m))_N R_N(n)$	$W_N^{-mk} X(k)$
$X(n)$	$Nx(N-k)$

续表

序 列	离散傅里叶变换
$W_N^{nl}x(n)$	$X((k+l))_N R_N(k)$
$x_1(n) \otimes x_2(n) = \sum_{m=0}^{N-1} x_1(m)x_2((n-m))_N R_N(n)$	$X_1(k)X_2(k)$
$r_{x_1x_2}(m) = \sum_{n=0}^{N-1} x_1(n)x_2((n-m))_N R_N(m)$	$X_1(k)X_2^*(k)$
$x_1(n)x_2(n)$	$\frac{1}{N} \sum_{l=0}^{N-1} X_1(l)X_2((k-l))_N R_N(k)$
$x^*(n)$	$X^*(N-k)$
$x(N-n)$	$X(N-k)$
$x^*(N-n)$	$X^*(k)$
$\text{Re}[x(n)]$	$X_{\text{ep}}(k) = \frac{1}{2} [X(k) + X^*(N-k)]$
$\text{jIm}[x(n)]$	$X_{\text{op}}(k) = \frac{1}{2} [X(k) - X^*(N-k)]$
$x_{\text{ep}}(n) = \frac{1}{2} [x(n) + x^*(N-n)]$	$\text{Re}[X(k)]$
$x_{\text{op}}(n) = \frac{1}{2} [x(n) - x^*(N-n)]$	$\text{jIm}[X(k)]$
$x(n)$ 为任意实序列	$\begin{cases} X(k) = X^*(N-k) \\ \text{Re}[X(k)] = \text{Re}[X(N-k)] \\ \text{Im}[X(k)] = -\text{Im}[X(N-k)] \\ X(k) = X(N-k) \\ \arg[X(k)] = -\arg[X(N-k)] \end{cases}$
$x_{\text{ep}}(n) = \frac{1}{2} [x(n) + x(N-n)]$, $x(n)$ 为实序列	$\text{Re}[X(k)]$
$x_{\text{op}}(n) = \frac{1}{2} [x(n) - x(N-n)]$, $x(n)$ 为实序列	$\text{jIm}[X(k)]$

帕塞瓦尔定理

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

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

第 14 集
微课视频

3.5 频域采样定理

在时域采样理论中曾提到,信号的时域采样使得信号频域的频谱以采样频率为周期进行周期延拓。在用计算机处理信号时,对信号的连续频谱也需要离散化,那么频域的采样又会引起时域的哪些变化?频域采样后恢复原信号的条件是什么?恢复信号的内插公式是什么?下面来讨论这些问题。

3.5.1 频域采样引起时域周期延拓

对任意绝对可和的非周期序列 $x(n)$, 其 Z 变换为

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

且 $X(z)$ 的收敛域包含单位圆(因为 $x(n)$ 存在傅里叶变换)。在单位圆上对 $X(z)$ 等间隔采样 N 点, 得到频域采样序列为

$$X(k) = X(z) \Big|_{z=e^{j\frac{2\pi}{N}k}} = \sum_{n=-\infty}^{\infty} x(n)e^{-j\frac{2\pi}{N}kn} = \sum_{n=-\infty}^{\infty} x(n)W_N^{kn}, \quad 0 \leq k \leq N-1 \quad (3.5.1)$$

即对 $x(n)$ 的离散时间傅里叶变换 $X(e^{j\omega})$ 在 $[0, 2\pi]$ 进行 N 点等间隔采样得到 $X(k)$ 。设由频域采样序列 $X(k)$ 恢复的有限长序列记为 $x_N(n)$, 即

$$x_N(n) = \text{IDFT}[X(k)], \quad 0 \leq n \leq N-1$$

下面推导 $x_N(n)$ 和原序列 $x(n)$ 之间的关系及频域采样定理。

由 DFT 是 DFS 的主值区间, 主值序列 $X(k)$ 对应的周期序列为 $\tilde{X}(k)$, 即

$$\tilde{X}(k) = X((k))_N = \text{DFS}[\tilde{x}(n)]$$

$$X(k) = \tilde{X}(k)R_N(k)$$

可得 $\tilde{X}(k)$ 对应的时域序列为

$$\tilde{x}_N(n) = \text{IDFS}[\tilde{X}(k)] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}(k)W_N^{-kn} = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-kn} \quad (3.5.2)$$

将式(3.5.1)代入式(3.5.2)得

$$\tilde{x}_N(n) = \frac{1}{N} \sum_{k=0}^{N-1} \left[\sum_{m=-\infty}^{\infty} x(m)W_N^{km} \right] W_N^{-kn} = \sum_{m=-\infty}^{\infty} x(m) \frac{1}{N} \sum_{k=0}^{N-1} W_N^{(m-n)k}$$

其中,

$$\frac{1}{N} \sum_{k=0}^{N-1} W_N^{(m-n)k} = \begin{cases} 1, & m = n + iN \text{ 且 } i \text{ 为任意整数} \\ 0, & \text{其他} \end{cases}$$

因此

$$\tilde{x}_N(n) = \sum_{m=-\infty}^{\infty} x(n + iN) \quad (3.5.3)$$

$$x_N(n) = \tilde{x}_N(n)R_N(n) = \sum_{i=-\infty}^{\infty} x(n + iN)R_N(n) \quad (3.5.4)$$

这说明频域等间隔采样得到的 $X(k)$ 所对应的时域序列 $x_N(n)$ 是由有限长序列 $x(n)$ 以 N 为周期进行周期延拓, 然后取主值序列 $0 \sim (N-1)$ 而得到的。

若 $x(n)$ 的长度为 M , 则当频域采样点数 N 大于或等于 M 时, 有

$$x_N(n) = \text{IDFT}[X(k)] = x(n)$$

则由频域采样 $X(k)$ 可以恢复时间域序列 $x(n)$ 。这就是频域采样定理。

当 N 小于 M , $x(n)$ 以 N 为周期进行周期延拓取主值序列将引起序列混叠。

因此可以看到, 时域采样造成了频域周期延拓。同样, 频域采样也造成了时域周期

延拓。

【例 3.5.1】 已知序列 $x(n) = \{1, 2, 3, 4, 5, 6, 7, 8\}$, 现对 $x(n)$ 的离散时间傅里叶变换 $X(e^{j\omega})$ 在 $N=6$ 时做一个周期内的采样, 得到离散频谱 $X(k) = X(e^{j\omega})|_{\omega=2\pi k/6}$, 求 IDFT $[X(k)]$ 对应的时域序列 $x_6(n), n=0, 1, \dots, 5$ 。

解 利用频域采样定理, 频域采样点为 $N=6$, 则时域序列 $x(n)$ 以 6 为周期做周期延拓, 延拓后取主值区间为 $x_6(n)$ 。因此, 在主值区间内有

$$x_6(n) = [x(n) + x(n+6)]R_6(n)$$

n		0	1	2	3	4	5	6	7		
$x(n+6)$...	5	6	7	8						
$x(n)$		<u>1</u>	2	3	4	5	6	7	8		
$x(n-6)$								1	2	3	...
$x_6(n)$		<u>8</u>	10	3	4	5	6				

即

$$x_6(n) = \{\underline{8}, 10, 3, 4, 5, 6\}$$

可以看到, 由于频域采样点数 $N=6$ 小于序列的长度 $M=8$, 因此由频域采样恢复的序列 $x_6(n)$ 只有 6 个长度, 前两个序列值存在混叠失真现象, 而后面 4 个序列值和原序列值对应。

3.5.2 频域采样后的恢复

下面推导由频域采样 $X(k)$ 恢复连续频谱 $X(e^{j\omega})$ 或者 $X(z)$ 。当满足频域采样定理时, 频域采样 $X(k)$ 就是序列 $x(n)$ 的 N 点 DFT, 对 $X(k)$ 进行 IDFT 可得到序列 $x(n)$, 即

$$x(n) = \text{IDFT}[X(k)] = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn} \quad (3.5.5)$$

将式(3.5.5)代入序列的 Z 变换公式, 有

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

其中, $W_N^{-kN} = e^{-j\frac{2\pi}{N}(-kN)} = e^{j2\pi k} = 1$, 因此

$$X(z) = \frac{1 - z^{-N}}{N} \sum_{k=0}^{N-1} \frac{X(k)}{1 - W_N^{-k} z^{-1}} \quad (3.5.6)$$

令复频域插值函数为

$$\Phi_k(z) = \frac{1}{N} \frac{1 - z^{-N}}{1 - W_N^{-k} z^{-1}} \quad (3.5.7)$$

则复频域内插公式可简化为

$$X(z) = \sum_{k=0}^{N-1} X(k) \Phi_k(z) \quad (3.5.8)$$

式(3.5.8)即为由频率域采样重构 $X(z)$ 的公式。

当 Z 变换在单位圆上取值时, $z = e^{j\omega}$ 。将其代入式(3.5.7)和式(3.5.8), 得到由 $X(k)$ 插值重构 $X(e^{j\omega})$ 的内插公式为

$$X(e^{j\omega}) = \sum_{k=0}^{N-1} X(k) \Phi_k(e^{j\omega}) \quad (3.5.9)$$

其中,

$$\begin{aligned} \Phi_k(e^{j\omega}) &= \frac{1 - e^{-j\omega N}}{N(1 - W_N^{-k} e^{-j\omega})} = \frac{1 - e^{-j\omega N}}{N(1 - e^{-j(\omega - 2\pi k/N)})} \\ &= \frac{1}{N} \frac{\sin(N\omega/2)}{\sin[(\omega - 2\pi k/N)/2]} e^{-j[\omega(N-1)/2 + k\pi/N]} \\ &= \frac{1}{N} \frac{\sin\left[N\left(\frac{\omega}{2} - \frac{k\pi}{N}\right)\right]}{\sin\left(\frac{\omega}{2} - \frac{k\pi}{N}\right)} e^{-\frac{jk\pi(N-1)}{N}} e^{-\frac{j(N-1)\omega}{2}} \end{aligned} \quad (3.5.10)$$

令内插函数为

$$\Phi(\omega) = \frac{1}{N} \frac{\sin\left(\frac{\omega N}{2}\right)}{\sin\left(\frac{\omega}{2}\right)} e^{-j\omega\left(\frac{N-1}{2}\right)} \quad (3.5.11)$$

将 $\Phi_k(e^{j\omega})$ 写成更方便的形式, 即

$$\Phi_k(e^{j\omega}) = \Phi\left(\omega - \frac{2\pi k}{N}\right) \quad (3.5.12)$$

式(3.5.9)可以写为

$$X(e^{j\omega}) = \sum_{k=0}^{N-1} X(k) \Phi\left(\omega - \frac{2\pi k}{N}\right) \quad (3.5.13)$$

这就是由 $X(k)$ 插值重构 $X(e^{j\omega})$ 的公式。

内插函数 $\Phi_k(e^{j\omega}) = \Phi\left(\omega - \frac{2\pi k}{N}\right)$ 在 $\omega = \frac{2\pi k}{N}$ 处即第 k 个采样点的值为 1, 保证了根据式(3.5.13)插值后得到的 $X(e^{j\omega})$ 在各采样点 k 的值保持为 $X(k)$ 不变, 采样点之间的值是式(3.5.13)中 N 项函数 $X(k)\Phi\left(\omega - \frac{2\pi k}{N}\right)$ 的叠加, 如图 3.5.1 所示。

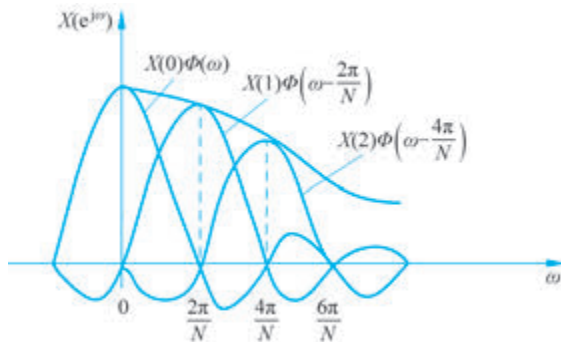


图 3.5.1 由插值函数重构 $X(e^{j\omega})$ 的示意图

频域采样理论及有关重构公式提供了一种有用的滤波器结构和滤波器设计途径,在后续章节中会详细讨论滤波器的设计方法。

3.6 DFT 的应用

图 3.6.1 用于说明 DFT 的导出过程,以便进一步理解 DFT 的周期延拓特性。该图不仅包含了之前讨论过的四种变换,还通过图形直观地解释了时域和频域采样引起的变化。

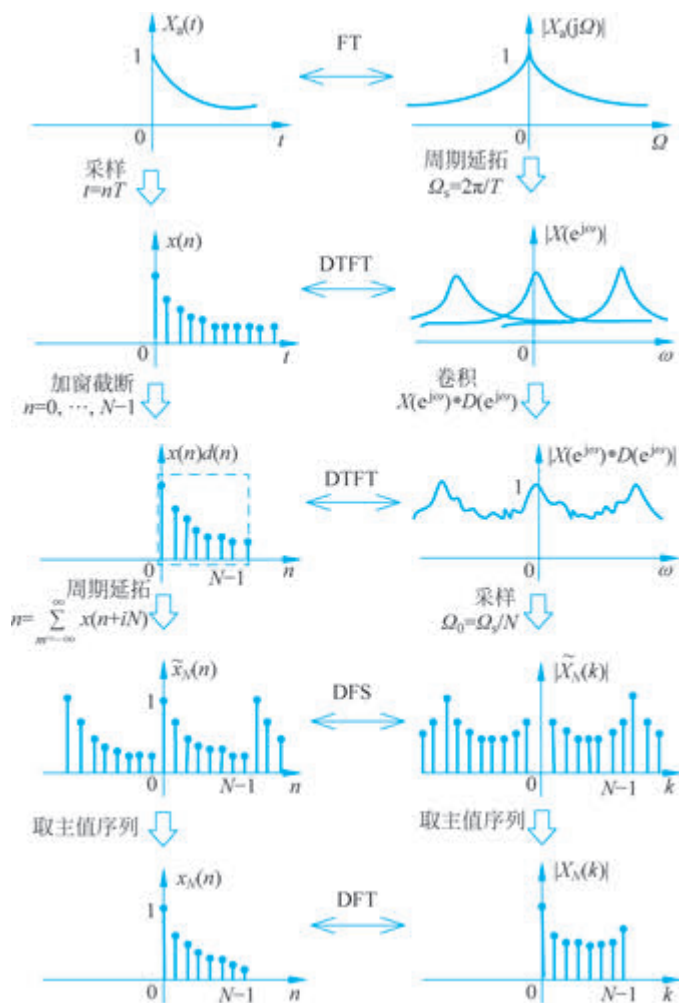


图 3.6.1 DFT 的图形解释

从图 3.6.1 中可以看到,模拟信号的数字化处理涉及时域采样、加窗截断以及频域采样等过程,在这些过程中会出现由周期延拓、频域卷积等因素导致的失真。在满足时域和频域采样定理的前提下,合理选择参数(包括采样频率,窗函数形状等),可以减小由数字化处理引起的失真,确保失真控制在允许的误差范围内,从而实现利用数字方法处理实际模拟信号并加以应用的目的。

3.6.1 DFT 对连续时间非周期信号的 DFT 逼近

结合前面的讨论,利用 DFT 来对连续信号进行谱分析时,需要考虑前述因素的影响。对于连续非周期信号 $x_a(t)$,其频谱函数 $X_a(j\Omega)$ 也是连续的。为了利用 DFT 对 $x_a(t)$ 进行频谱分析,首先需要对 $x_a(t)$ 进行时域采样,得到离散信号 $x(n)=x_a(nT)$,其中, T 为采样间隔。接着,对 $x(n)$ 进行 DFT,得到的 $X(k)$ 则是 $x(n)$ 的傅里叶变换 $X(e^{j\omega})$ 在 $[0,2\pi]$ 上的 N 点等间隔采样结果。这里 $x(n)$ 和 $X(k)$ 均为有限长序列。需要注意的是,使用 DFT 对连续信号进行频谱分析是一种近似方法,其近似程度与信号的带宽、采样频率和采样序列的截取长度有关。

(1) 由 DFT 估计连续时间信号频谱 $X_a(j\Omega)$ 。

$x(n)$ 的傅里叶变换 $X(e^{j\omega})$ 与 $x_a(t)$ 的傅里叶变换 $X_a(j\Omega)$ 满足

$$X(e^{j\omega}) = \frac{1}{T} \sum_{m=-\infty}^{\infty} X_a \left[j \left(\frac{\omega}{T} - \frac{2\pi}{T} m \right) \right] \quad (3.6.1)$$

将 $\omega = \Omega T$ 代入式(3.6.1)得

$$X(e^{j\Omega T}) = \frac{1}{T} \sum_{m=-\infty}^{\infty} X_a \left[j \left(\Omega - \frac{2\pi}{T} m \right) \right] \quad (3.6.2)$$

由 $x(n)$ 的 N 点 DFT 的定义有

$$X(k) = \text{DFT}[x(n)] = X(e^{j\omega}) \Big|_{\omega = \frac{2\pi}{N}k}, \quad 0 \leq k \leq N-1 \quad (3.6.3)$$

即

$$X(k) = X(e^{j\frac{2\pi}{N}k}) \quad 0 \leq k \leq N-1 \quad (3.6.4)$$

当满足采样定理时, $X(e^{j\omega})$ 的主值区间等于模拟信号的频谱 $X_a(j\Omega)$ 除以采样间隔 T 。同时 $X(k)$ 又是 $X(e^{j\omega})$ 的采样,说明可以用 $X(k)$ 来近似估计模拟信号的频谱。

$$X_a(j\Omega) \Big|_{\Omega = 2\pi k/N T} = T X(k) = T \cdot \text{DFT}[x(n)], \quad k = 0, 1, 2, \dots, N-1 \quad (3.6.5)$$

这就是使用 DFT 估计连续信号频谱的公式。

(2) 利用 IDFT 估计连续时域信号 $x(t)$ 。

为了方便表示,令 $\Omega_0 = \frac{2\pi}{NT}$,则

$$X_a(j\Omega) \Big|_{\Omega = 2\pi k/N T} = X_a(jk\Omega_0) \quad (3.6.6)$$

对式(3.6.6)两边做傅里叶逆变换,有

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\Omega) e^{j\Omega t} d\Omega \quad (3.6.7)$$

则

$$x(nT) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\Omega) e^{j\Omega nT} d\Omega \quad (3.6.8)$$

将离散频谱 $X_a(jk\Omega_0)$ 代入式(3.6.8),并且将积分变为求和,即

$$\int_{-\infty}^{\infty} d\Omega = \sum_{k=0}^{N-1} \Omega_0$$

则

$$\begin{aligned}
 x(nT) &\approx \frac{1}{2\pi} \sum_{k=0}^{N-1} X(jk\Omega_0) e^{jk\Omega_0 nT} \cdot \Omega_0 \\
 &\approx \frac{1}{NT} \sum_{k=0}^{N-1} X(jk\Omega_0) e^{j\frac{2\pi}{N}nk} \approx \frac{1}{T} \text{IDFT}[X(jk\Omega_0)]
 \end{aligned} \quad (3.6.9)$$

因此,使用 DFT 来近似分析连续信号频谱的正逆变换公式为

$$X(jk\Omega_0) = X(j\Omega) \Big|_{\Omega=k\Omega_0} = T \cdot \text{DFT}[x(n)], \quad 0 \leq k \leq N-1 \quad (3.6.10)$$

$$x(nT) = x(t) \Big|_{t=nT} \approx \frac{1}{T} \text{IDFT}[X(jk\Omega_0)], \quad 0 \leq k \leq N-1 \quad (3.6.11)$$

需要注意的是,在近似时,需要考虑信号时域采样、加窗截断、频域采样、频谱泄漏、栅栏效应等问题。

3.6.2 DFT 分析信号频谱时参数的选择

对模拟信号进行频谱分析时,需要确定时域采样频率、频率分辨率、频域采样点数等参数。需要考虑频谱泄漏的影响,以及模拟频率和数字频率之间的对应关系。

1. 采样频率

时域采样定理要求采样频率 f_s 要大于信号最高频率 f_h 的两倍,即

$$f_s > 2f_h \quad (3.6.12)$$

采样导致了频谱的周期延拓,这里 f_s 对应数字角频率 2π , $f_s/2$ 称为折叠频率,对应数字角频率 π 。

在实际中对 $x(t)$ 进行采样时,如果事先不知道信号的最高频率,一般可以从观测记录的数据(波形)中确定一个近似的频率。具体方法是,取数据(波形)中变化速度最快的两相邻峰谷点之间的时间间隔 t_0 作为半个周期(如图 3.6.2 所示),则有

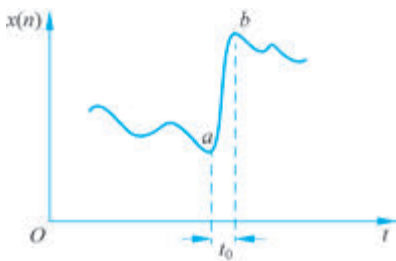


图 3.6.2 估算信号最高频率 f_h

$$t_0 = \frac{T_h}{2}, \quad f_h = \frac{1}{T_h} = \frac{1}{2t_0} \quad (3.6.13)$$

确定这一近似最高频率分量后,就可按前面的方法选取 f_s 。

若采样频率不满足 $f_s > 2f_h$,则频谱分量在周期延拓后会产生频谱混叠,引起频谱失真。实际采样前,为了避免频谱混叠,会在采样前让信号先通过截止频率为 $f_s/2$ 的低通滤波器,这样的滤波器被称为抗混叠滤波器。这样可以滤除信号的高频部分,避免采样时产生混叠。

2. 频率分辨率

频率分辨率指信号频谱中能够分辨的最接近的两个谱峰的间隔。两个谱峰越接近,频率分辨率越低,说明分辨能力越高。可以用 F_0 表示频率分辨率。

如果一个周期 $[0, 2\pi]$ 中的频率采样点数为 N ,则频率分辨率为

$$F_0 = \frac{f_s}{N} \quad (3.6.14)$$

而

$$N = \frac{T_0}{T} \quad (3.6.15)$$

其中, T_0 为信号的时间长度, T 为采样间隔, 因而频率分辨率可化为

$$F_0 = \frac{1}{T_0} \quad (3.6.16)$$

可以看出, 频率分辨率的大小和信号的时间长度成反比。因此, 如果频率分辨率不够小, 可以通过增加 N 来减小 F_0 。由于 $X(k)$ 是 $X(e^{j\omega})$ 的采样, 因此, 可以通过给时间域序列补零来增加采样点, 即增加 N 让频域采样点变密, 让原先看不见的谱线变得可见, 而并没有增加 $X(e^{j\omega})$ 包含的实际物理信号的频谱分辨率。

通过上面的讨论可以看到, 频谱分辨率的大小与信号的实际长度成正反。这被称为 DFT 的**栅栏效应**。即, 当使用 DFT 计算频谱时, 就如同通过一个栅栏观察景色, 只能在离散点处看到真实的景象, 而离散点之间的景象无法观测到, 如果实际谱线在离散点之间有很大变化, 那么可能会丢失信息。因此, 为了提高实际信号的分辨频谱的能力, 需要增加信号实际记录长度以降低频谱分辨率。补零只能使 $X(e^{j\omega})$ 采样点变密, 而不能真正提高物理上的分辨能力。由例 3.3.2 的图 3.3.2 可以看到, 无论是 8 点还是 16 点的 DFT, 在对序列进行补零后, 确实让看不见的谱线变得可见, 但这不能真正补充 $X(e^{j\omega})$ 的物理信息。

3. 信号最高频率与频率分辨率之间的矛盾

时域和频域的采样点 N 可以用信号的时间长度 T_0 与采样间隔 T 的比值表示, 也可以用采样频率 f_s 与频率间隔 F_0 的比值表示, 即

$$N = \frac{T_0}{T} = \frac{f_s}{F_0} \quad (3.6.17)$$

如果信号的最高频率 f_h 增加, 则对应的采样频率 f_s 就需要增大。当采样点数 N 固定不变时, 频率间隔 F_0 也会增大, 即频率分辨率下降。

要提高频率分辨率, 即减小 F_0 , 则需要增加记录的时间长度 T_0 。然而, 在采样点 N 固定不变时, 增加 T_0 意味着采样间隔 T 会增大, 从而导致采样频率 f_s 会减小。因此, 如果不希望产生混叠现象, 能够处理的信号的最高频率 f_h 也需要相应减小。

如果要同时提高能处理信号的最高频率和分辨频率的能力, 就需要增加采样点数 N 。在使用 FFT 算法做 DFT 时, N 一般取 2 的整数次幂。

【例 3.6.1】 在使用 FFT(DFT) 做信号频谱分析时, 其采样点数为 2 的整数次幂, 信号的最高频率为 5kHz, 要求频率分辨率小于或等于 10Hz。试确定: (1) 采样点间的最大时间间隔 T ; (2) 最小记录长度 T_0 ; (3) 在一个记录中最少点数 N 。

解

(1) 求解最大时间间隔。

由 $f_s \geq 2f_h, f_s = \frac{1}{T}$, 则

$$T < \frac{1}{2f_h} = \frac{1}{2 \times 5 \times 10^3} = 0.1 \text{ms}$$

因此, 最大采样间隔为 0.1ms。

(2) 求解最小记录长度。

$$T_0 \geq \frac{1}{F_0} = 0.1 \text{s}$$

因此,最小记录长度为 0.1s 。

(3) 求解最小记录点数。

$$N > \frac{2f_h}{F_0} = \frac{2 \times 5 \times 10^3}{10} = 10^3$$

取最接近 10^3 的 2 的整数次幂,因此 $N = 2^m = 2^{10} = 1024$ 。所以最小记录点数为 1024。

4. 频谱泄漏

由于实际序列信号可能较长,计算机在处理时会序列做截断,即在时域序列上乘以一个窗口函数,例如矩形窗 $R_N(n)$ 。除了可以加矩形窗截断之外,也可以用其他形状的窗,后面章节会详细讨论各种窗函数。

现在来分析加矩形窗后频谱的变化,加窗的过程可以用图 3.6.3 说明。时域序列经过矩形窗截断变为

$$x_N(n) = x(n)R_N(n) \quad (3.6.18)$$

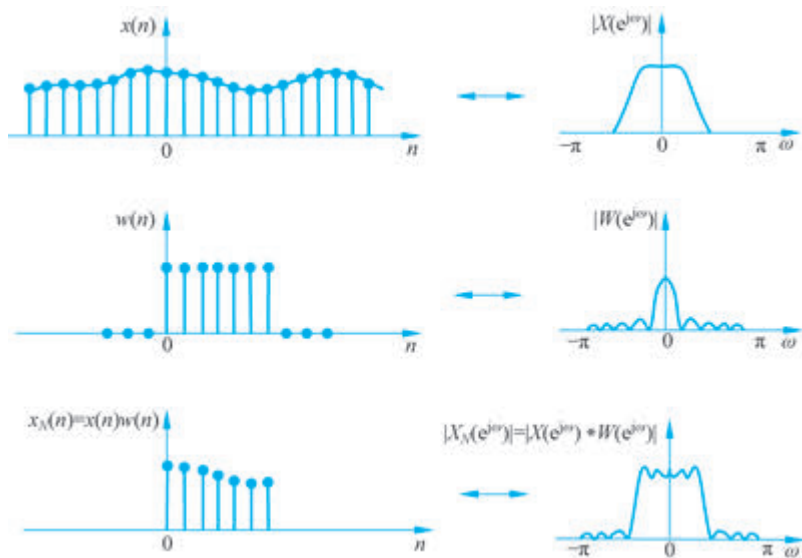


图 3.6.3 加窗截断后频谱泄漏示意图

根据卷积定理,时域相乘对应着频谱的复卷积运算,即

$$X_N(e^{j\omega}) = \frac{1}{2\pi} X(e^{j\omega}) * W_N(e^{j\omega}) \quad (3.6.19)$$

其中, $X(e^{j\omega}) = \text{DTFT}[x(n)]$ 表示原序列的频谱, $X_N(e^{j\omega}) = \text{DTFT}[x_N(n)]$ 表示截断后序列的频谱,式中 $W_N(e^{j\omega})$ 为矩形窗谱,可表示为

$$\begin{aligned} W_N(e^{j\omega}) &= \text{DTFT}[R_N(n)] = \sum_{n=-\infty}^{\infty} R_N(n) e^{-j\omega n} = \sum_{n=0}^{N-1} e^{-j\omega n} = \frac{1 - e^{-j\omega N}}{1 - e^{-j\omega}} \\ &= \frac{e^{-j\frac{\omega N}{2}} (e^{j\frac{\omega N}{2}} - e^{-j\frac{\omega N}{2}})}{e^{-j\frac{\omega}{2}} (e^{j\frac{\omega}{2}} - e^{-j\frac{\omega}{2}})} = e^{-j\frac{N-1}{2}\omega} \frac{\sin\left(\frac{\omega N}{2}\right)}{\sin\left(\frac{\omega}{2}\right)} \end{aligned} \quad (3.6.20)$$

其幅度响应为

$$|W_N(e^{j\omega})| = \left| \frac{\sin\left(\frac{\omega N}{2}\right)}{\sin\left(\frac{\omega}{2}\right)} \right| \quad (3.6.21)$$

相位响应为

$$\arg[W_N(e^{j\omega})] = -\frac{N-1}{2}\omega + \arg\left[\frac{\sin\left(\frac{\omega N}{2}\right)}{\sin\left(\frac{\omega}{2}\right)}\right] \quad (3.6.22)$$

从图 3.6.3 可以看到,与窗谱的卷积运算使得信号的谱线发生了展宽。截断的时域序列长度越长,即 N 越大,则 $4\pi/N$ 越小,频谱的展宽越窄,泄漏也就越小。这种由于卷积运算导致的频谱展宽现象,称为**频谱泄漏**。频谱泄露会使得分辨频谱的能力降低,当两个信号的频率离得很近时,频谱的泄漏会使得两个信号无法分辨。

为了减轻截断效应,可以采用两种方式。

(1) **增加截断窗的长度**,使得窗谱的主瓣宽度 $4\pi/N$ 减小,从而降低频谱泄漏。

(2) **采用缓慢变化的窗函数来截断信号**,这些窗函数及其性质在第 7 章详细介绍。这样的窗谱旁瓣幅度更小,使得频谱泄漏减弱。

当对信号进行非整周期截断时,会造成严重的频谱泄漏现象和频率值的偏差,通过增加周期数可以降低非完整周期部分所占的比例,从而减轻频谱泄漏的程度,减小频率偏差。当周期数足够多时能够得到较准确的频率值,但是始终不能消除该频率处幅值的偏差。

【例 3.6.2】 设有单频正弦周期信号 $x(t) = \cos(2\pi f_0 t)$, $f_0 = 20\text{Hz}$, 采样频率 $f_s = 100\text{Hz}$ 。试分析它的频谱。

解 信号的周期为 $T_0 = \frac{1}{f_0} = 0.05\text{s}$ 。采样频率 $f_s = 100\text{Hz}$, 则采样间隔为

$$T = \frac{1}{f_s} = 0.01\text{s}$$

则一个周期内采样点数为

$$N = \frac{2\pi}{\omega_0} = \frac{f_s}{2\pi f_0} = 5$$

所以,采样后的序列是以 5 为周期的周期序列。

取两个周期的数据长度 $T_0 = NT = 10 \times 0.01 = 0.1\text{s}$, 进行离散傅里叶变换可得

$$X(k) = \sum_{n=0}^9 \cos\left(\frac{2\pi}{5}n\right) W_{10}^{nk} = \begin{cases} 5, & k=2,8 \\ 0, & k=0,1,3,4,5,6,7,9 \end{cases}$$

取 $T_0 = NT = 16 \times 0.01 = 0.16\text{s}$, 此时,截断长度不是正弦序列的整周期的整数倍,则时域周期延拓后,一定不是周期序列,频谱会产生泄漏,如图 3.6.4(b)所示。

取 $T_0 = NT = 39 \times 0.01 = 0.396\text{s}$, 此时,截断长度也不是正弦序列的整周期的整数倍,频谱会产生泄漏,如图 3.6.4(c)所示。

如果不知道周期信号的周期,可以采用试探法。选择较长的观察时间,以减小截断效应的影响;如果效果不佳,再选择较大跨度。此外,也可以通过多次迭代,比较峰值处的误差,

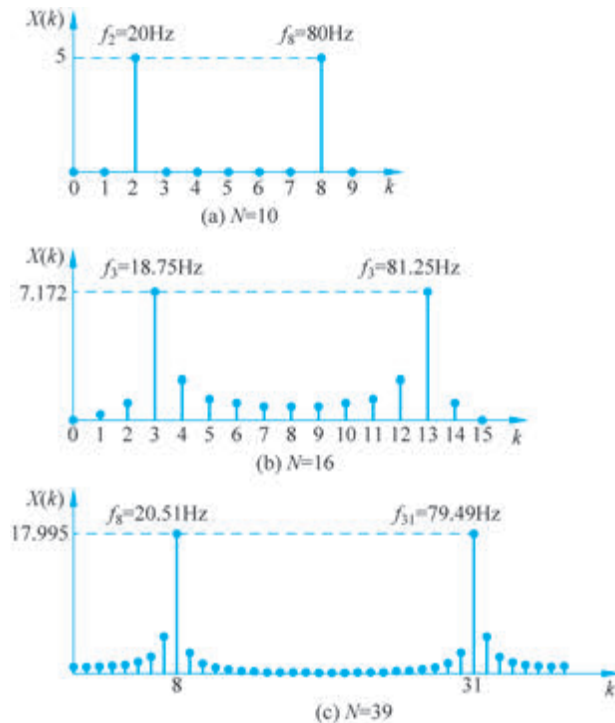


图 3.6.4 周期信号不同长度截断对频谱的影响

当误差满足预定条件时,停止迭代计算。

5. 离散频率点的对应关系

如表 3.6.1 所示为离散频率与连续频率及其角频率的对应关系。从表中可以看到数字频率与模拟频率的对应关系,从而可以利用数字方法来处理模拟信号的频谱。在第 4 章讲解了 FFT 快速算法后,再详细列举一些频谱分析的实例。

表 3.6.1 DFT 离散频率与连续频率及其角频率的对应关系(采样频率为 f_s)

离散频率 k	离散角频率 ω_k	连续频率 f	连续角频率 Ω
0	0	0	0
1	$\frac{2\pi}{N}$	$\frac{f_s}{N}$	$\frac{2\pi f_s}{N}$
2	$2 \times \frac{2\pi}{N}$	$2 \times \frac{f_s}{N}$	$2 \times \frac{2\pi f_s}{N}$
k	$k \times \frac{2\pi}{N}$	$k \times \frac{f_s}{N}$	$k \times \frac{2\pi f_s}{N}$
$N/2$	π	$\frac{f_s}{2}$	πf_s
...
$N-1$	$(N-1) \times \frac{2\pi}{N}$	$(N-1) \times \frac{f_s}{N}$	$(N-1) \times \frac{2\pi f_s}{N}$
N	2π	f_s	$2\pi f_s$

第 3 章中的内置函数和功能如表 3.6.2 所示。

表 3.6.2 第 3 章中的内置函数和功能

内置函数	功能
np. outer()	向量的外积
np. dot()	向量的内积(点积)
x@y	矩阵乘法
np. mod()	模运算, 计算除法后的余数
scipy. linalg. dft()	生成离散傅里叶变换(DFT)矩阵

习题

3.1 设序列 $x(n) = R_3(n)$, $\tilde{x}(n) = \sum_{r=-\infty}^{\infty} x(n+7r)$, 求 $\tilde{X}(k)$, 并作图表示 $\tilde{x}(n)$ 、 $\tilde{X}(k)$ 。

3.2 设 $x(n) = R_3(n)$, $h(n) = R_4(n-2)$, 令 $\tilde{x}(n) = x((n))_6$, $\tilde{h}(n) = h((n))_6$ 试求 $\tilde{x}(n)$ 与 $\tilde{h}(n)$ 的周期卷积, 并作图。

3.3 计算下列序列的 DFT(闭合表达式形式)。

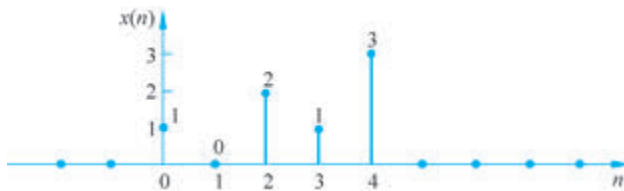
- (1) $x(n) = e^{j\omega_0 n} R_N(n)$
- (2) $x(n) = a(\sin\omega_0 n) R_N(n)$
- (3) $x(n) = a^n R_N(n)$
- (4) $x(n) = \delta(n-n_0), 0 < n_0 < N$
- (5) $x(n) = n R_N(n)$
- (6) $x(n) = n^2 R_N(n)$

3.4 已知 $x(n) = \{1, 2, 3, 5, 4\}$, (1) 计算 $X(e^{j\omega}) = \text{DTFT}[x(n)]$ 及 $X(k) = \text{DFT}[x(n)]$, 并说明二者的关系; (2) 将 $x(n)$ 的尾部补零, 得到 $x_0(n) = \{1, 2, 3, 5, 4, 0, 0, 0\}$, 计算 $X_0(e^{j\omega}) = \text{DTFT}[x_0(n)]$, $X_0(k) = \text{DFT}[x_0(n)]$; (3) 将(1)、(2)的结果加以比较, 得出相应的结论。

3.5 设 $\tilde{x}(n)$ 是周期为 N 的周期序列, 若 $\tilde{X}(k) = \sum_{n=0}^{N-1} \tilde{x}(n) W_N^{nk}$, $\tilde{X}_1(k) = \sum_{n=0}^{2N-1} \tilde{x}(n) W_{2N}^{nk}$, 试用 $\tilde{X}(k)$ 来表示 $\tilde{X}_1(k)$ 。

3.6 题图 3.1 表示一个 5 点序列 $x(n)$, 试画出:

- (1) $x(n) * x(n)$; (2) $x(n) \textcircled{5} x(n)$; (3) $x(n) \textcircled{10} x(n)$



题图 3.1

3.7 设采样频率为 720Hz 的时域采样序列为 $x(n) = \cos\left(\frac{\pi}{6}n\right) + 5\cos\left(\frac{\pi}{3}n\right) + 4\sin\left(\frac{\pi}{7}n\right)$, 对 $x(n)$ 作 72 点 DFT 运算。试回答:

- (1) 所选 72 点截断是否能保证得到周期序列? 说明理由。

(2) 是否会产生频谱泄漏? 请粗略画出信号谱线, 并进行说明。

3.8 设有两个序列

$$x(n) = \begin{cases} x(n), & 0 \leq n \leq 6 \\ 0, & \text{其他} \end{cases}$$

$$y(n) = \begin{cases} y(n), & 0 \leq n \leq 15 \\ 0, & \text{其他} \end{cases}$$

对 $x(n)$ 和 $y(n)$ 各做 18 点处的 DFT, 然后将两个 DFT 相乘, 再求乘积的 IDFT, 设所得结果为 $f(n)$, 问 $f(n)$ 的哪些点(用序号 n 表示)的值对应于 $x(n) * y(n)$ 得到的值。

3.9 已知 $X(k)$ 为 8 点实序列的 DFT, 且已知

$$X(0) = 6, \quad X(1) = 4 + j3, \quad X(2) = -3 - j2, \quad X(3) = 2 - j, \quad X(4) = 4$$

试利用 DFT 的性质(不必求 IDFT)来确定以下各表达式的值。

$$(1) x(0) \quad (2) x(4) \quad (3) \sum_{n=0}^7 x(n) \quad (4) \sum_{n=0}^7 |x(n)|^2$$

3.10 已知 $x(n)$ 是 N 点的有限长序列, $X(k) = \text{DFT}[x(n)]$ 。现将 $x(n)$ 的每两点之间补进 $r-1$ 个零值点, 得到一个 rN 点的有限长序列

$$y(n) = \begin{cases} x(n/r), & n = ir \text{ 且 } i = 0, 1, \dots, N-1 \\ 0, & \text{其他} \end{cases}$$

试求 rN 点处的 DFT $[y(n)]$ 与 $X(k)$ 的关系。

3.11 设有一谱分析用的信号处理器, 采样点数必须为 2 的整数幂, 假定没有采用任何特殊数据处理措施, 要求频率分辨率不大于 10Hz, 如果采用的采样时间间隔为 0.1ms, 试确定:

- (1) 最小记录长度;
- (2) 所允许处理信号的最高频率;
- (3) 在一个记录中的最少点数。

3.12 令 $X(k)$ 表示 N 点序列 $x(n)$ 的 N 点 DFT, 试证明:

- (1) 若 $x(n)$ 满足 $x(n) = -x(N-1-n)$, 则 $X(0) = 0$;
- (2) 当 N 为偶数时, 若 $x(n) = x(N-1-n)$, 则 $X(N/2) = 0$ 。

3.13 设 $x(n)$ 是 n 点的有限长序列, $0 \leq n \leq N-1$, $X(k) = \text{DFT}[x(n)]$ 。

$$y(n) = \begin{cases} x(n), & 0 \leq n \leq N-1 \\ 0, & N \leq n \leq rN-1 \end{cases}$$

$$Y(k) = \text{DFT}[y(n)], \quad 0 \leq k \leq rN-1$$

试用 $X(k)$ 表示 $Y(k)$ 。

3.14 已知一个模拟信号为 $x(t) = \sin(180\pi t) + 1.3\sin(260\pi t) + 1.6\sin(640\pi t)$ 。用 $f_s = 600\text{Hz}$ 对 $x(t)$ 采样, 取其长度为 $N = 64$, 得到序列 $x(n)$ 。

(1) 用 Python 编程, 作出 $x(n)$ 在 64 点处的 DFT, 并画出频谱幅度 $|X(k)|$, $k = 0, 1, 2, \dots, 63$ 。

(2) 讨论 $|X(k)|$ 中与 $x(t)$ 有关的各个频谱值, 并说明这些值代表的意义。