

$$m_f \approx 5\,120, \quad m_d = 8\,192$$

循环卷积的运算量比直接卷积略小。

当 $N_2 = 1\,048\,576$, $N_1 = 16$ 则

$$m_f \approx 22\,020\,096, \quad m_d = 16\,777\,216$$

循环卷积的运算量已超过直接卷积的运算量。显然 N_2 太大, 会使循环卷积的优点不能充分发挥出来, 这是因为对 $h(n)$ 必须补很多的零, 以至于 L 点 $h(n)$ 中大部分是零, 很不经济。

另外, 在实际运用中, $h(n)$ 一般为系统函数, $x(n)$ 为外部采集数据, 很长的 $x(n)$ 则意味占用大量的存储单元和处理时间的延误, 对系统的性能等均带来不利影响。

克服这个困难的办法是采用分段卷积或使用称为分段滤波的办法。通常, 将 $x(n)$ 分成与 $h(n)$ 相仿的片段, 采用两种方法处理。

1. 重叠相加法

假定 $x_i(n)$ 表示图 3.30 中第 i 段 $x(n)$ 序列

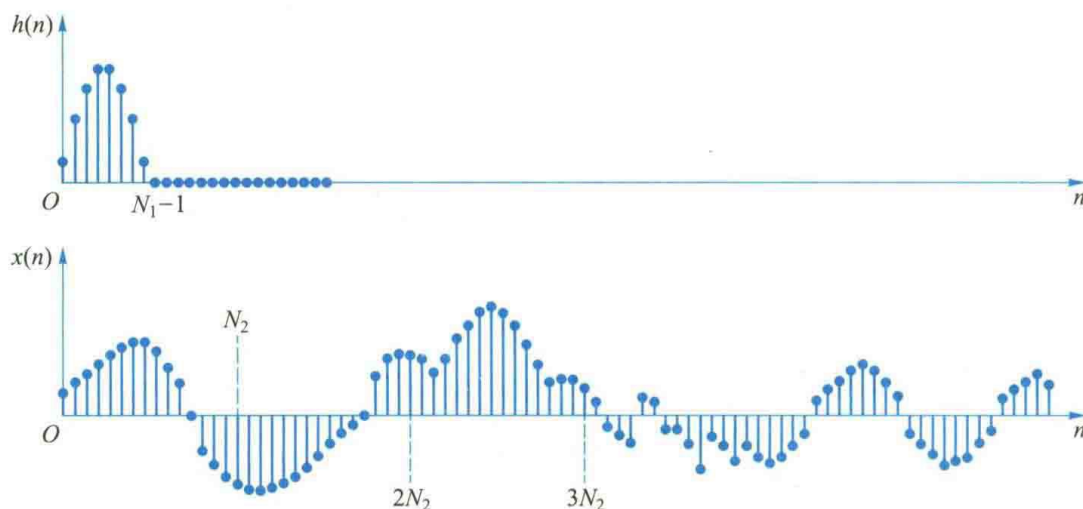


图 3.30 长序列分段滤波

$$x_i(n) = \begin{cases} x(n) & iN_2 \leq n \leq (i+1)N_2 - 1 \\ 0 & \text{其他} \end{cases} \quad (3.107)$$

显然, 输入序列 $x(n)$ 可表示为

$$x(n) = \sum_{i=-\infty}^{\infty} x_i(n) \quad (3.108)$$

这样, $x(n)$ 和 $h(n)$ 的卷积就可表示为

$$y(n) = x(n) * h(n) = \sum_{i=-\infty}^{\infty} x_i(n) * h(n) = \sum_{i=-\infty}^{\infty} y_i(n) \quad (3.109)$$

式中

$$y_i(n) = x_i(n) * h(n) \quad (3.110)$$

在式(3.108)中, 和式的每一项 $x_i(n)$ 只有 N_2 个非零样本, 而 $h(n)$ 的长度为 N_1 , 显而易见, 线性