绪　　论

信息技术的蓬勃发展正在改变着人类的生产和生活。计算机技术及电子技术的发展推动了数字信号处理技术的发展，数字信号处理技术也同样促进了计算机技术的发展。信号是信息的载体，信号处理是研究对含有信息的信号进行处理（变换），以获得人们所希望的信号，从中提取信息、利用信息的一门学科。由于模拟信号难以做到高精度，并且可靠性差、不灵活，故随着大规模集成电路和数字信号处理技术的成熟和发展，现在用数字方法处理信号（数字信号处理）已基本取代了模拟信号处理。

数字信号处理就是利用数字计算机或专用数字硬件对数字信号进行变换、加工处理运算，用数字的数值计算方法处理，达到提取有用信息的目的。国际上一般把 1965 年由 Cooley-Turkey 提出的快速傅里叶变换（FFT）作为数字信号处理这一学科的产生时间。

数字信号处理把数字或符号表示成序列，通过计算机或专用处理设备，用数字的方式去处理这些序列，以达到更符合人们要求的信号形式。例如，对信号的滤波，提取和增强信号的有用分量，削弱无用的分量，或估计信号的某些特征参数。总之，凡是用数字方式对信号进行滤波、变换、增强、压缩、估计、识别等，都是数字信号处理的研究对象。

1．数字信号处理系统的组成

数字信号处理的目的是利用计算机或专用数字信号处理设备来处理信号以得到有用信息，但是计算机或专用数字信号处理设备只能处理时域上是离散的、频域上也是离散的信号。由于现实世界中的信号多为连续信号，要实现对现实世界连续信号的处理，必须先进行模数转换。通常，数字信号处理系统由前置预滤波器、A/D 转换器、数字信号处理器、D/A 转换器、模拟滤波器组成。数字信号处理系统的组成框图如图 0-1 所示。

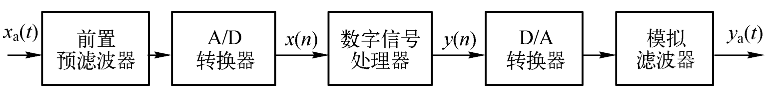


图 0-1　数字信号处理系统的组成框图

模拟信号先经过前置预滤波器，将输入信号中高于某一频率（称为折叠频率，等于抽样频率的一半）的分量加以滤除，使进入 A/D 转换器中的信号在某一频率范围内。A/D 转换器通过抽样保持、量化、编码的过程，对信号进行数字化处理，将模拟信号用数字信号表示，时间离散化、幅度离散化是这种数字信号的特点，可以用二进制序列 $x(n)$ 来表示。数字信号处理器（DSP）对输入的数字信号 $x(n)$ 进行变换，加工处理得到输出信号 $y(n)$ 。D/A 转换器将数字信号 $y(n)$ 变成模拟信号，这些模拟信号在时间上的幅度等于序列 $y(n)$ 中相应数码所代表的数值。模拟滤波器滤除不需要的高频分量，形成平滑的模拟输出信号 $y_a(t)$ 。

当然实际的系统并不一定包括所有框图，例如，有些系统只需数字输出，可直接以数字形式显示或打印，那么就不需要 D/A 转换器了。另外一些系统，其输入就是数字量，因而就不需要 A/D 转换器了。对于纯数字系统，则只需要数字信号处理器这一核心部分即可。

2．数字信号处理的学科概貌

数字信号处理的基本工具包括微积分、概率统计、随机过程、高等代数、数值分析、近世代数和

复变函数。语音信号处理、数字图像处理和模式识别都是在数字信号处理基础上发展起来的学科。数字信号处理的理论基础是线性移不变（LSI）系统理论和离散傅里叶变换（DFT）。

数字信号处理的学科概貌如图 0-2 所示，其中，时域离散线性移不变（LSI）系统理论和离散傅里叶变换（DFT）是数字信号处理领域的理论基础，而数字滤波和数字频谱分析是数字信号处理的两个基本学科分支。

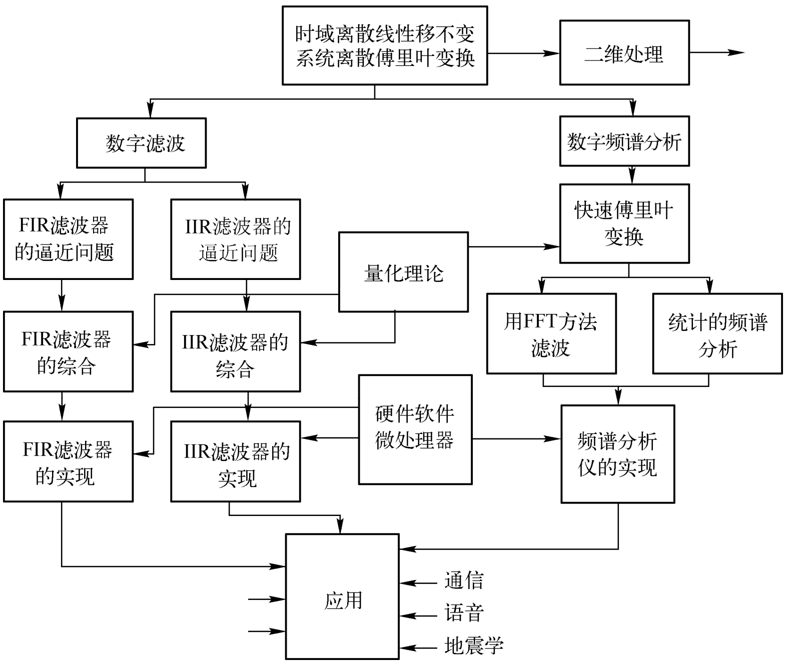


图 0-2 数字信号处理的学科概貌

数字滤波领域则分为无限长单位脉冲响应（IIR）数字滤波器和有限长单位脉冲响应（FIR）数字滤波器两部分内容。包括它们的数学逼近问题、综合问题（包括选择滤波器结构和选择运算字长）及具体的硬件或计算机软件实现问题。

数字频谱分析领域包括两部分内容：（1）确定信号的频谱分析，这可采用离散傅里叶变换（DFT）法来进行分析，对于较复杂的情况，可采用线性调频 Z 变换（CZT）；（2）随机信号的频谱分析，这就是统计谱分析法。实际谱分析技术中都要用到快速傅里叶变换（FFT）和一些快速卷积算法。FFT 还可用来实现 FIR 数字滤波运算，而统计频谱分析法又可用来研究数字信号处理系统的量化噪声效应（二维和多维信号处理则是最新发展的领域）。

3．数字信号处理的特点

数字信号处理采用数字系统完成信号处理的任务，它具有数字系统的一些共同优点，如抗干扰、可靠性强、便于大规模集成等。除此之外，与传统的模拟信号处理方法相比较，它还具有以下一些明显优点。

（1）精度高。模拟网络的精度由元器件决定，在模拟系统的电路中，元器件精度要达到 10^{-3} 以上已经不容易了，而数字系统 17 位字长可以达到 10^{-5} 的精度，这是很平常的。例如，基于离散傅里叶变换的数字式频谱分析仪，其幅值精度和频率分辨率均远远高于模拟频谱分析仪。

（2）灵活性高。数字信号处理采用了专用或通用的数字系统，数字系统的性能主要取决于运算程序和乘法器的系数，而系数是存放在系数存储器中的，只需改变运算程序或存储的系数，就可得到不

同的系统，比改变模拟系统方便得多。

（3）可靠性高。因为数字系统只有两个信号电平“0”“1”，因而受周围环境温度及噪声的影响较小，而模拟系统中，各元器件都有一定的温度系数，且电平是连续变化的，易受温度、噪声、电磁感应等的影响。如采用大规模集成电路，可靠性就更高。

（4）易于大规模集成。由于数字部件有高度规范性，便于大规模集成和大规模生产。

（5）时分复用。也就是利用数字信号处理器同时处理几个通道的信号。处理器运算速度越高，能处理的信道数目也就越多。

（6）可获得高性能指标。例如，对信号进行频谱分析，模拟频谱仪在频率低端只能分析 10Hz 以上频率，且难于做到高分辨率（足够窄的带宽），但在数字的谱分析中，已能做到 10^{-3}Hz 的谱分析。又如有限长冲激响应数字滤波器，可实现准确的线性相位特性，这在模拟系统中是很难达到的。并且，数字系统可以实现模拟系统很难达到的指标或特性。例如，有限长单位脉冲响应数字滤波器可以实现严格的线性相位；在数字信号处理中可以将信号存储起来，用延迟的方法实现非因果系统，从而提高了系统的性能指标。

（7）二维与多维处理。利用庞大的存储单元，可以存储二维的图像信号或多维的阵列信号，实现二维或多维的谱分析，以及实现二维甚至多维信号的处理，包括二维或多维滤波及谱分析等。

数字信号处理系统也有局限性，例如，增加了系统的复杂性，需要模拟接口及比较复杂的数字系统；应用的频率范围受到限制，主要是 A/D 转换器的抽样频率的限制；系统的功率消耗比较大，数字信号处理系统中集成了几十万甚至更多的晶体管，而模拟信号处理系统中大量使用的是电阻、电容、电感等无源器件，随着系统复杂性的增加，这一矛盾会更加突出。另外，在实时性方面，还不如模拟系统（但随着计算机处理速度和 DSP 处理实时性的不断提高，该问题并不突出）。

4．数字信号处理的发展及应用

数字信号处理由于其独特的学科特点形成了其发展规律。

（1）由简单的运算向复杂的运算发展，目前几十位乘几十位的全并行乘法器可以在几纳秒的时间内完成一次浮点乘法运算，这在运算速度上和运算精度上均为复杂的数字信号处理算法提供了先决条件。

（2）由低频向高频发展，模数转换器的抽样频率已高达数百兆赫兹，可以将视频甚至更高频率的信号数字化后送入计算机处理。

（3）由一维向多维发展，像高分辨率彩色电视、雷达、地质和石油勘探等多维信号处理的应用领域已与数字信号处理结下了不解之缘。

（4）各种数字信号处理系统几经更新换代，在图像处理方面，图像数据压缩是多媒体通信、影碟机（VCD 或 DVD）和高清晰度电视（HDTV）的关键技术。国际上先后制定的标准 H.261、JPEG、MPEG-1 和 MPEG-2 中均使用了离散余弦变换（DCT）算法。近年来发展起来的小波（Wavelet）变换也是一种具有高压缩比和快速运算特点的崭新压缩技术，应用前景十分广阔，有望成为新一代压缩技术的标准。

数字信号处理具有突出优点，因而在通信、语音、雷达、地震测报、声呐、遥感、生物医学、电视、仪器中得到愈来愈多的应用，如程控交换机、移动通信系统、数字照相机、液晶电视、家庭影院、全球定位系统（GPS）、医院用的 B 超、CT、核磁共振、卫星遥感遥测、天气预报、地震预报、地震探矿、数字化士兵和数字化战争等。

数字信号处理还在不断开辟新的应用领域，在机械制造中，基于 FFT 算法的频谱分析仪用于振动分析和机械故障诊断；医学中使用数字信号处理技术对心电（ECG）和脑电（EEG）等生物电信号进

行分析和处理；数字音频广播（DAB）广泛地使用了数字信号处理技术。除此之外，数字信号处理在机器人控制、虚拟仪器系统和汽车电子方面也有着长足的应用。可以说，数字信号处理技术已在信息处理领域引起了广泛的关注和高度的重视，数字化已经进入各个领域并影响着人们的生活。

5．数字信号处理系统的实现

数字信号处理系统的实现主要有以下几种方法。

（1）利用通用计算机用软件实现。软件采用高级语言编写，也可利用商品化的各种 DSP 软件。MATLAB 是美国 MathWorks 公司出品的商业数学软件，用于算法开发、数据可视化、数据分析及数值计算的高级技术计算语言和交互式环境，主要包括 MATLAB、Simulink 和符号运算三部分。该方法实现简单、灵活，但实时性较差，很少用于实时系统，主要用于教学和科研的前期研制阶段。另外，Lyrtech 推出了 Lyrtech Signal Processing（LSP）快速原型开发平台。该平台继承了 MATLAB/Simulink 算法仿真环境和 DSP+FPGA 的快速原型开发板，无缝地实现了自顶向下的开发流程。LSP 信号处理快速原型开发系统由一系列硬件板卡和相对应的软件模块组成，该系统为用户提供了两种开发流程——系统级开发流程和底层开发流程，用户可以根据自己的开发习惯进行自由选择。LSP 快速开发平台已广泛应用于无线通信、音视频信号处理、航天航空国防控制和汽车电子等各个领域。

（2）利用单片机。单片机技术发展现已相当成熟，不仅价格便宜，而且功能很强。可根据不同环境选用不同型号的单片机，用来进行实时控制，但数据运算量不能太大。

（3）利用通用 DSP 芯片。DSP 芯片与单片机相比有更为突出的优点，如 DSP 内部带有乘法器、累加器，采用流水线工作方式及并行结构，多总线速度快，配有适于信号处理的指令（如 FFT 指令）等。

目前市场上的 DSP 芯片有美国德州仪器公司（TI）的 TMS320CX 系列，AT&T 公司的 DSP16、DSP32 系列，Motorola 公司的 DSP56x、DSP96x 系列和 AD 公司的 ADSP21×、ADSP210×系列等。

（4）利用特殊用途的 DSP 芯片。是指专门用于 FFT、FIR 滤波器，实现卷积、相关运算等的专用数字芯片。目前此类 DSP 芯片有 BB 公司的 DF17××系列，MAXIM 公司的 MAXIM27×、MAXIM28×系列，National 公司的 National-SEMI 系列和 MF 系列等。其软件算法已在芯片内部用硬件电路实现，使用者只需给出输入数据，便可在输出端直接得到输出数据。

（5）一些典型的数字信号处理算法，如 FFT、FIR 和 IIR 滤波器，可以采用 FPGA 实现。FPGA 是现场可编程门阵列（Field Programmable Gate Array）的简称，能够以低系统开销和低成本实现高速乘-累加操作，并且可以高效地实行并行运算，尤其是对于滤波这样的重复任务，并行处理远超串行架构。FPGA 在硬件设计方面，使用硬件描述语言 HDL（Hardware Description Language）。现在有两种 HDL 语言，一种是 VHDL，另一种是 Verilog HDL，两种语言对 FPGA 实现数字信号处理都适用。

第 1 章　时域离散信号和系统

学习重点

★ 掌握信号的分类、时域离散信号的表示。

★ 掌握序列的运算、序列的周期性及常用典型序列，学会判断序列的周期性。

★ 掌握时域离散系统的线性、因果性、稳定性。

★ 掌握信号的抽样频率选取条件，以及抽样信号恢复原始信号的原理。

★ 了解数字信号处理的分析方法及 MATLAB 仿真。

1.1　引言

时域离散信号和系统的时域、频域分析是数字信号处理的基础理论。在信号分析与处理理论中，通常将信号的自变量作为时间变量。信号可以按变量的取值是否连续进行分类：时间连续的信号称为时域连续信号；时间连续且幅值连续的信号称为模拟信号；时间离散的信号常称为时域离散信号。

信号是传递信息的函数。按照信号的分类，信号处理分为模拟信号处理和数字信号处理两类。处理信号的设施称为系统，如果系统的输入、输出都是模拟信号，则称为时域连续系统（简称模拟系统）；如果系统的输入、输出是时域离散信号，则称为时域离散系统（简称离散系统）。

按照信号特点的不同，信号可以表示成一个或几个独立变量的函数。例如，某个地区气温随时间变化的函数是一元函数；全国各地气温随时间变化的函数是二元函数。一维变量可以是时间，也可以是其他参量，习惯上将其看成时间。信号可以分为以下几种。

（1）时域连续信号，其特点是时间连续，信号幅度函数取值可以是连续的，也可以是离散的。

时域连续信号是在连续时间范围内定义的信号，但信号的幅值可以是连续数值，也可以是离散数值。当幅值为连续变量时，又常称为模拟信号。

（2）时域离散信号，其特点是时间离散，函数取值连续。

时域离散信号是时间为离散变量的信号，即时间被量化了，而幅度仍是连续变化的信号。

（3）数字信号，其特点是时间离散，幅度函数取值量化。

数字信号是时间离散且幅度量化的信号，即时间变量被量化了，幅度函数取值为离散量化值的信号。

在本书中，主要把一维、确定的时域离散信号和时域离散系统作为研究的对象。

1.2　时域离散信号与序列运算

1.2.1　时域离散信号及其表示

时域离散信号只在离散时间上给出函数值，是时间上不连续的序列。

离散时间信号可以通过对连续时间信号抽样获得，用序列 $x(n)$ 来表示。但 $x(n)$ 具有更加广泛的意义，它不仅可以表示时间信号，也可以表示非时间信号。例如，某一时刻世界各地的气温就不是按时间顺序排列的序列。在研究这类信号时，通常可以将自变量视为时间信号。

若连续时间信号为 $x(t)$，抽样周期为 T，则抽样得到离散时间信号（序列）

$$x(n) = x_{\mathrm{a}}(t)\big|_{t=nT} = x_{\mathrm{a}}(nT) = x(nT) \tag{1-1}$$

式中，序号 n 为整数（$n=0,\pm1,\pm2,\cdots$），表示第 n 个抽样时间点。当 n 为非整数时，序列 $x(n)$ 无定义。$x(n)$ 既是序列的第 n 个序列值，又代表整个序列。

序列的表示方式有列举法、表达式表示法和序列波形表示法。例如，一个有限长序列在区间 $-2\leqslant n\leqslant 4$ 具有非零值，则相应序列值依次为：$x(-2)=-1$，$x(-1)=0$，$x(0)=1$，$x(1)=2$，$x(2)=3$，$x(3)=4$，$x(4)=5$。

用列举法表示为

$$x(n)=\{-1,\underset{\uparrow}{0},1,2,3,4,5\}$$

其中箭头所指项为 $x(0)$，若无箭头，则序列从 $x(0)$ 项开始。

表达式为

$$x(n)=\begin{cases} n+1 & -2\leqslant n\leqslant 4 \\ 0 & 其他 \end{cases}$$

序列波形如图 1-1 所示。

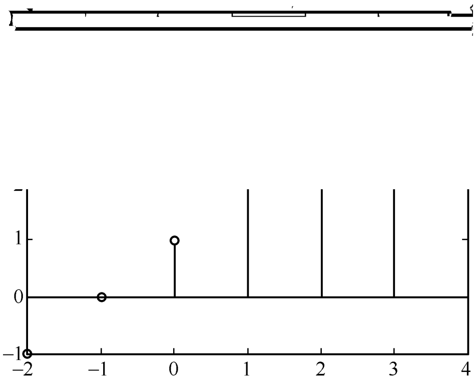


图 1-1 序列波形表示

要点一：时域离散信号的表示方法有列举法、表达式表示法和序列波形表示法。

1.2.2 序列的运算

序列的基本运算主要包括相加、相乘、累加和、差分、移位、反褶、尺度变换（抽取和插值）、线性卷积和。其中，差分、累加和分别对应于连续时间信号的微分和积分运算。

假设有两个序列，其波形如图 1-2 所示。

$$x(n)=\begin{cases} 1 & n=-1 \\ 1 & n=0 \\ 2 & n=1 \\ -1 & n=2 \\ 0 & n=3 \\ 1 & n=4 \\ 0 & 其他 \end{cases} \qquad y(n)=\begin{cases} 1 & n=-2 \\ -1 & n=-1 \\ 0 & n=0 \\ 2 & n=1 \\ 1 & n=2 \\ 0 & n=3 \\ -1 & n=4 \\ 0 & 其他 \end{cases}$$

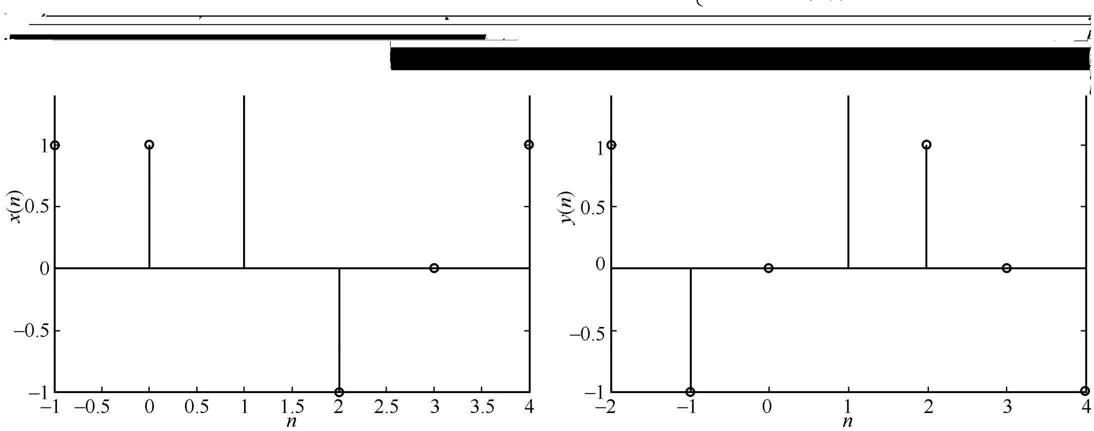


图 1-2 序列波形图

下面以这两个序列为例，介绍序列的基本运算。

1．相加

两个序列相加是指同序列号的两个序列值逐项对应相加。表达式为

$$z(n)=x(n)+y(n) \tag{1-2}$$

【例 1-1】 计算如图 1-2 所示的序列 $x(n)$ 与序列 $y(n)$ 的和序列 $z(n)$ 。

解：$z(-2)=x(-2)+y(-2)=0+1=1$

$z(-1)=x(-1)+y(-1)=1+(-1)=0$

$z(0)=x(0)+y(0)=1+0=1$

$z(1)=x(1)+y(1)=2+2=4$

$z(2)=x(2)+y(2)=(-1)+1=0$

$z(3)=x(3)+y(3)=0+0=0$

$z(4)=x(4)+y(4)=1+(-1)=0$

其余的序列值为 0，序列 $z(n)$ 为序列 $x(n)$ 与序列 $y(n)$ 的和序列。

要点二：两个序列相加是指同序列号的两个序列值逐项对应相加，得到一个新的序列。

2．相乘

两个序列相乘是指同序列号的两个序列值逐项对应相乘。表达式为

$$z(n)=x(n)\cdot y(n) \tag{1-3}$$

【例 1-2】 计算如图 1-2 所示的序列 $x(n)$ 与序列 $y(n)$ 的乘积序列 $z(n)$ 。

解：同序列号的两个序列值对应相乘得：

$$z(n)=\begin{cases}0 & n=-2\\ -1 & n=-1\\ 0 & n=0\\ 4 & n=1\\ -1 & n=2\\ 0 & n=3\\ -1 & n=4\\ 0 & \text{其他}\end{cases}$$

要点三：两个序列相乘是指同序列号的两个序列值逐项对应相乘，得到一个新的序列。

MATLAB 编程求例 1-1 和例 1-2 两个序列的和、乘积。

1．建立一个主函数，命名为 main.m，程序如下：

```
n1=-1:4;
n2=-2:4;
x1=[1,1,2,-1,0,1,0];
x2=[1,-1,0,2,1,0,-1,0];
[y3,n]=sigadd(x1,n1,x2,n2);
[y,n]=sigcheng(x1,n1,x2,n2);
subplot(2,2,1);
stem(n1,x1,'.');
grid;
xlabel('n1');
ylabel('x1');
subplot(2,2,2);
```

```
        stem(n2,x2,'.');
        grid;
        xlabel('n2');
        ylabel('x2');
        subplot(2,2,3);
        stem(n,y3,'.');
        grid;
        title('序列加');
        xlabel('n');
        ylabel('y3');
        subplot(2,2,4);
        stem(n,y,'.');
        grid;
        title('序列积');
        xlabel('n');
        ylabel('y');
```

2．建立一个加法函数，命名为 sigadd.m，程序如下：

```
        function[y3,n]=sigadd(x1,n1,x2,n2)n=min(min(n1),min(n2)):max(max(n1),max(
n2));
        y1=zeros(1,length(n));
        y2=zeros(1,length(n));
        y1(find((n>=min(n1))&(n<=max(n1))==1))=x1;
        y2(find((n>=min(n2))&(n<=max(n2))==1))=x2;
        y3=y1+y2;
```

3．建立一个乘法函数，命名为 sigcheng.m，程序如下：

```
        function[y,n]=sigcheng(x1,n1,x2,n2)
        n=min(min(n1),min(n2)):max(max(n1),max(n2));
        y1=zeros(1,length(n));
        y2=zeros(1,length(n));
        y1(find((n>=min(n1))&(n<=max(n1))==1))=x1;
        y2(find((n>=min(n2))&(n<=max(n2))==1))=x2;
        y=y1.*y2;
```

注：三个程序分别编译。

编程结果如图 1-3、图 1-4 所示。

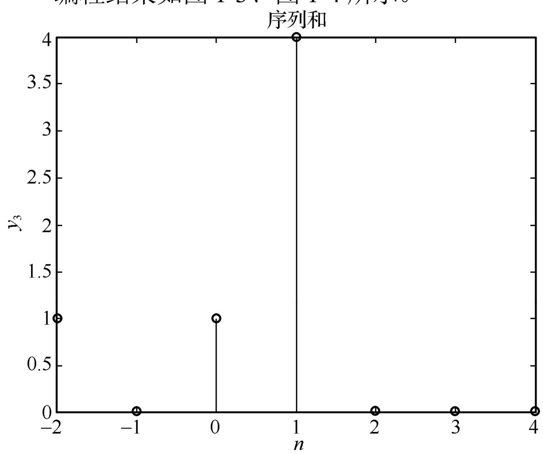


图 1-3　序列和

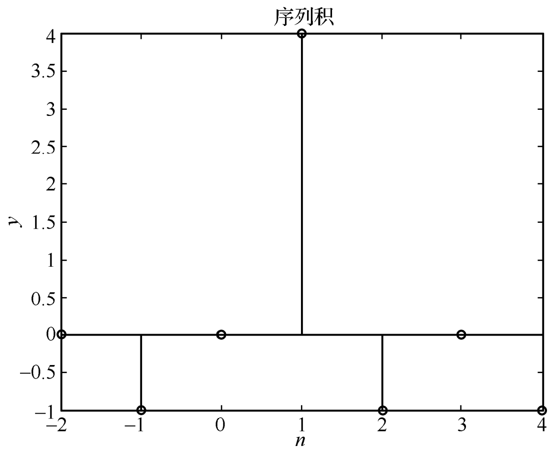


图 1-4　序列积

3. 累加和

序列的累加和为该序列的前 n 项值的和的序列，定义式为

$$z(n)=\sum_{m=-\infty}^{n}x(m) \tag{1-4}$$

序列的累加和运算类似于连续信号的积分运算。

【例 1-3】 计算如图 1-2 所示的序列 $x(n)$ 的累加和序列 $z(n)$ 。

解： $z(-1)=1$ ，即 $n=-1$ 的序列值与 $n=-1$ 以前的所有序列值的和；

$z(0)=2$ ，即 $n=0$ 的序列值与 $n=0$ 以前的所有序列值的和；

$$\vdots$$

$z(4)=4$ ，即 $n=4$ 的序列值与 $n=4$ 以前的所有序列值的和。

当 n 取其他值时，累加和序列的所有值都为 4。

其累加和的序列波形如图 1-5 所示。

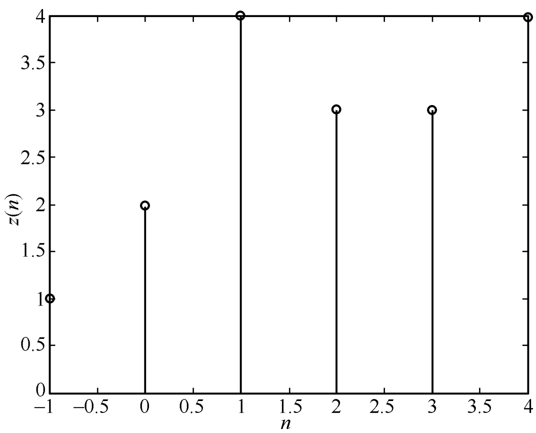


图 1-5　序列累加和

要点四：计算序列的累加和得到一个新的序列，新序列的第 n 项序列值为原序列的前 n 项序列值的和。

4. 移位

序列的移位序列可以表示序列的延时或超前序列。延时序列表示为序列右移，当 $m>0$ 时，$x(n-m)$ 表示序列 $x(n)$ 的右移序列（或延时序列）；超前序列表示为序列左移，当 $m>0$ 时， $x(n+m)$ 表示序列 $x(n)$ 的左移序列（或超前序列）。

【例 1-4】 计算如图 1-2 所示的序列 $x(n)$ 的移位序列 $x(n-1)$ 与 $x(n+1)$ 。

解：

$$x(n-1)=\begin{cases}1 & n=0\\ 1 & n=1\\ 2 & n=2\\ -1 & n=3\\ 0 & n=4\\ 1 & n=5\\ 0 & \text{其他}\end{cases}\qquad x(n+1)=\begin{cases}1 & n=-2\\ 1 & n=-1\\ 2 & n=0\\ -1 & n=1\\ 0 & n=2\\ 1 & n=3\\ 0 & \text{其他}\end{cases}$$

波形如图 1-6 所示。

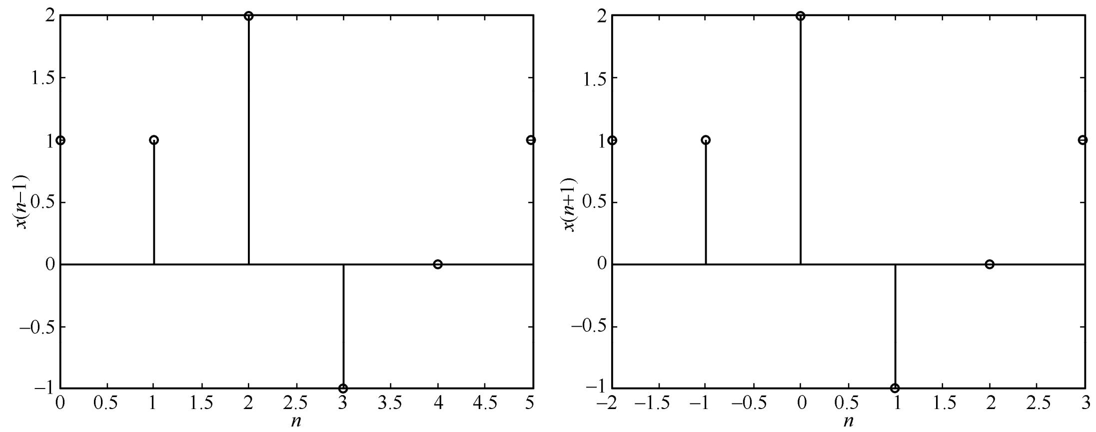


图 1-6　序列的移位

要点五： *序列右移得到原序列的延时序列，表示为* $x(n-m)$ *；序列左移得到原序列的超前序列，表示为* $x(n+m)$ 。

5．反褶（反转）

序列 $x(n)$ 以 $n=0$ 为纵对称轴进行翻褶，形成的序列称为序列 $x(n)$ 的反褶序列或反转序列，表示为 $x(-n)$ 。

【例 1-5】 计算如图 1-2 所示的序列 $x(n)$ 的反褶序列 $x(-n)$ 。

解：

$$x(-n)=\begin{cases} 1 & n=-4 \\ 0 & n=-3 \\ -1 & n=-2 \\ 2 & n=-1 \\ 1 & n=0 \\ 1 & n=1 \\ 0 & 其他 \end{cases}$$

波形如图 1-7 所示。

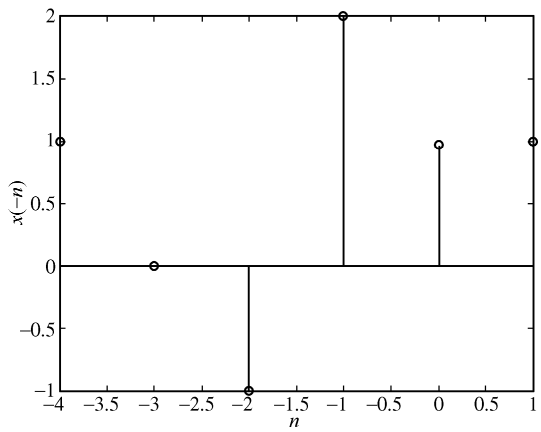


图 1-7　序列的反褶（反转）

要点六：序列 $x(n)$ 的反褶（反转）序列 $x(-n)$ 是以 $n=0$ 为纵对称轴对 $x(n)$ 进行反褶得到的序列。

6．差分

前向差分

$$\Delta x(n)=x(n+1)-x(n) \tag{1-5}$$

后向差分

$$\nabla x(n)=x(n)-x(n-1) \tag{1-6}$$

由定义式可得出

$$\nabla x(n)=\Delta x(n-1) \tag{1-7}$$

离散信号的差分运算与连续信号的微分运算相对应，如连续信号 $\delta(t)=\dfrac{\mathrm{d}u(t)}{\mathrm{d}t}$；离散信号单位冲激序列可以由单位阶跃序列的差分运算得到，即 $\delta(n)=u(n)-u(n-1)$ 。

【例 1-6】 计算如图 1-2 所示的序列 $x(n)$ 的前向差分 $\Delta x(n)$ 和后向差分 $\nabla x(n)$ 。

解：

$$\Delta x(n)=\begin{cases}1 & n=-2\\ 0 & n=-1\\ 1 & n=0\\ -3 & n=1\\ 1 & n=2\\ 1 & n=3\\ -1 & n=4\\ 0 & \text{其他}\end{cases}\qquad \nabla x(n)=\begin{cases}1 & n=-1\\ 0 & n=0\\ 1 & n=1\\ -3 & n=2\\ 1 & n=3\\ 1 & n=4\\ -1 & n=5\\ 0 & \text{其他}\end{cases}$$

波形如图 1-8 所示。

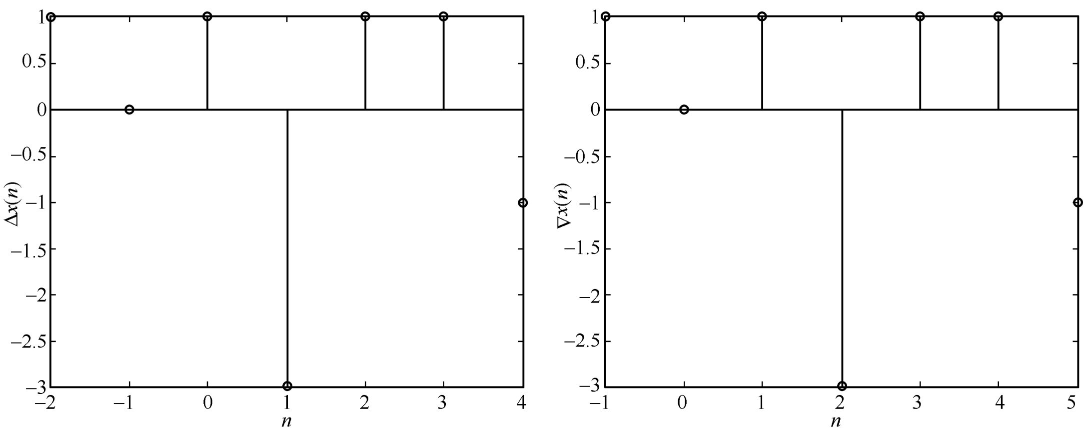


图 1-8　序列的差分

要点七：序列 $x(n)$ 的前向差分为其超前序列 $x(n+1)$ 与原序列的差序列；序列 $x(n)$ 的后向差分为该序列与其延时序列 $x(n-1)$ 的差序列。

7．尺度变换（抽取和插值）

序列 $x(n)$ 的尺度变换包括抽取和插值两种运算，分别对应于序列波形的压缩和扩展。

抽取是将序列 $x(n)$ 的自变量 n 换成 nm（$m\geqslant 2$，为正整数），得到一个新序列 $x(mn)$ 的变换。$x(mn)$

是对 $x(n)$ 以 $n=0$ 为起点，分别向左、向右每隔 $m-1$ 个点抽取一个点得到的。如果 $x(n)$ 是对连续时间信号 $x_{\mathrm{a}}(t)$ 以抽样周期 T 的抽样，则 $x(mn)$ 就相当于对 $x_{\mathrm{a}}(t)$ 以抽样周期 mT 的抽样，即

$$x(n)=x_{\mathrm{a}}(t)\big|_{t=nT} \tag{1-8}$$

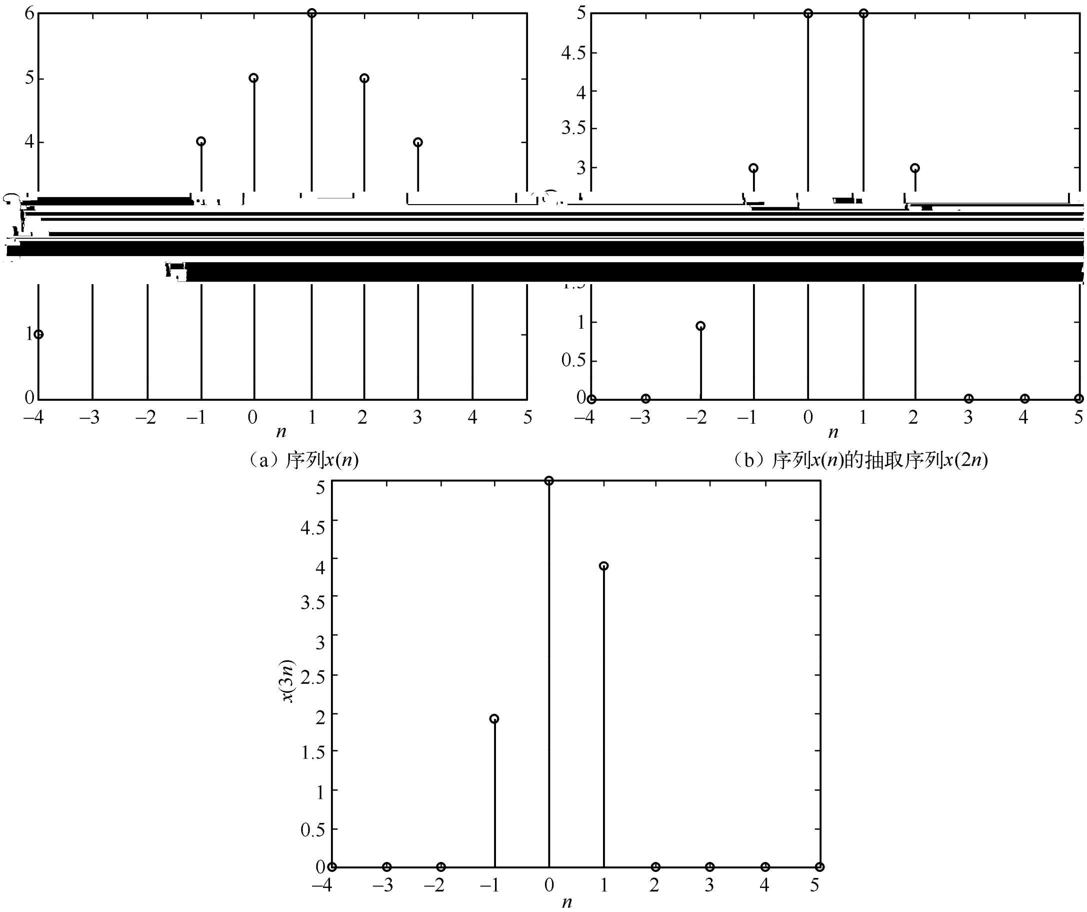
$$x(mn)=x_{\mathrm{a}}(t)\big|_{t=mnT} \tag{1-9}$$

例如，若序列 $x(n)$ 的波形如图 1-9（a）所示，则序列 $x(2n)$ 是对序列 $x(n)$ 以 $n=0$ 为起点，分别向左、向右每隔 1 个点抽取一个点得到的；序列 $x(3n)$ 是对序列 $x(n)$ 以 $n=0$ 为起点，分别向左、向右每隔 2 个点抽取一个点得到的。序列 $x(2n)$ 的波形如图 1-9（b）所示，序列 $x(3n)$ 的波形如图 1-9（c）所示。

插值是将序列 $x(n)$ 的自变量 n 换成 n/m（$m\geqslant 2$，为正整数）得到一个新序列 $x(n/m)$ 的变换。插值可以视为抽取的反过程，$x(n/m)$ 是在 $x(n)$ 的相邻序列值之间插入 $m-1$ 个“0”得到的，相当于对 $x_{\mathrm{a}}(t)$ 以抽样周期 T/m 进行抽样，即

$$x(mn)=x_{\mathrm{a}}(t)\big|_{t=nT/m} \tag{1-10}$$

例如，如图 1-9（a）所示的序列 $x(n)$ 的插值序列 $x(n/2)$ 如图 1-10 所示。



（a）序列$x(n)$　　（b）序列$x(n)$的抽取序列$x(2n)$

（c）序列$x(n)$的抽取序列$x(3n)$

图 1-9　序列抽取实例

要点八： 序列 $x(n)$ 的抽取序列 $x(mn)$ 是对序列 $x(n)$ 以 $n=0$ 为起点，分别向左、向右每隔 $m-1$ 个点抽取一个点得到的序列（$m\geqslant 2$，为正整数）。插值可以视为抽取的反过程，$x(n/m)$ 是在 $x(n)$ 的相邻序列值之间插入 $m-1$ 个“0”得到的序列。

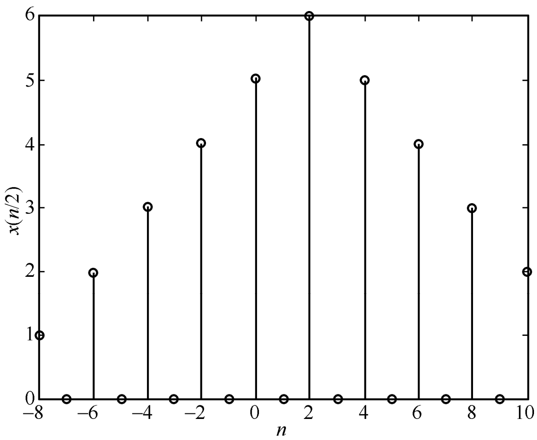


图 1-10　序列插值序列 $x(n/2)$

8．卷积和

序列 $x(n)$ 和序列 $y(n)$ 的卷积和定义为

$$z(n)=x(n)*y(n)=\sum_{m=-\infty}^{\infty}x(m)y(n-m) \tag{1-11}$$

式中，“*”表示卷积和运算。

类似于时域连续线性移不变系统输出响应（零状态响应）的卷积积分求解方法，卷积和是求解时域离散线性移不变系统输出响应（零状态响应）的主要方法。

卷积和的求解方法有图解法、对角线列表法、相乘对位相加法、解析法等多种求解方法。这里选择几种比较简便的求解方法进行介绍。

（1）图解法

根据卷积和定义式，卷积和运算可以采用图解法计算，通过以下步骤完成。

第一步，变量代换，将自变量 n 变为 m，得到 $x(m)$ 和 $y(m)$。

第二步，反褶，将 $y(m)$ 反褶，得到 $y(-m)$（其中两个序列任意一个均可反褶，一般选简单者）。

第三步，每移一位，给定一个 n 值，将 $y(-m)$ 移位得到 $y(n-m)$，$\begin{cases} n>0 & \text{向右平移} \\ n<0 & \text{向左平移} \end{cases}$。

第四步，取 n 的初值为 $-\infty$，逐渐增大 n 值，向右移动序列 $y(n-m)$，使不动序列 $x(m)$ 的最左边值和移动序列 $y(n-m)$ 的最右边值对齐。

第五步，将 $x(m)$ 与 $y(n-m)$ 的对应序列值相乘，并将相乘产生的全部结果值相加，取得的结果作为序列 $z(n)$ 的当前值。

第六步，将 n 值增加 1，向右平移 $y(n-m)$，并重复第五步。

第七步，重复第六步，直到不动序列的最右边值和移动序列的最左边值对齐，便产生序列 $z(n)$ 的全部序列值。此时，产生的序列 $z(n)$ 即为序列 $x(n)$ 和序列 $y(n)$ 的卷积和。

卷积和 $z(n)$ 序列的长度为 $L=M+N-1$，其中，M、N 分别为 $x(m)$ 和 $y(m)$ 的序列长度。

【例 1-7】 用图解法求下列两个序列的卷积和。

$$x(n)=\begin{cases}1 & 0\leqslant n\leqslant 3\\ 0 & \text{其他}\end{cases}\qquad y(n)=\begin{cases}2^{-n} & -1\leqslant n\leqslant 2\\ 0 & \text{其他}\end{cases}$$

解：由定义可知，卷积和为

$$z(n) = x(n) * y(n) = \sum_{m=-\infty}^{\infty} x(m)y(n-m)$$

图解过程和结果如图 1-11 所示。

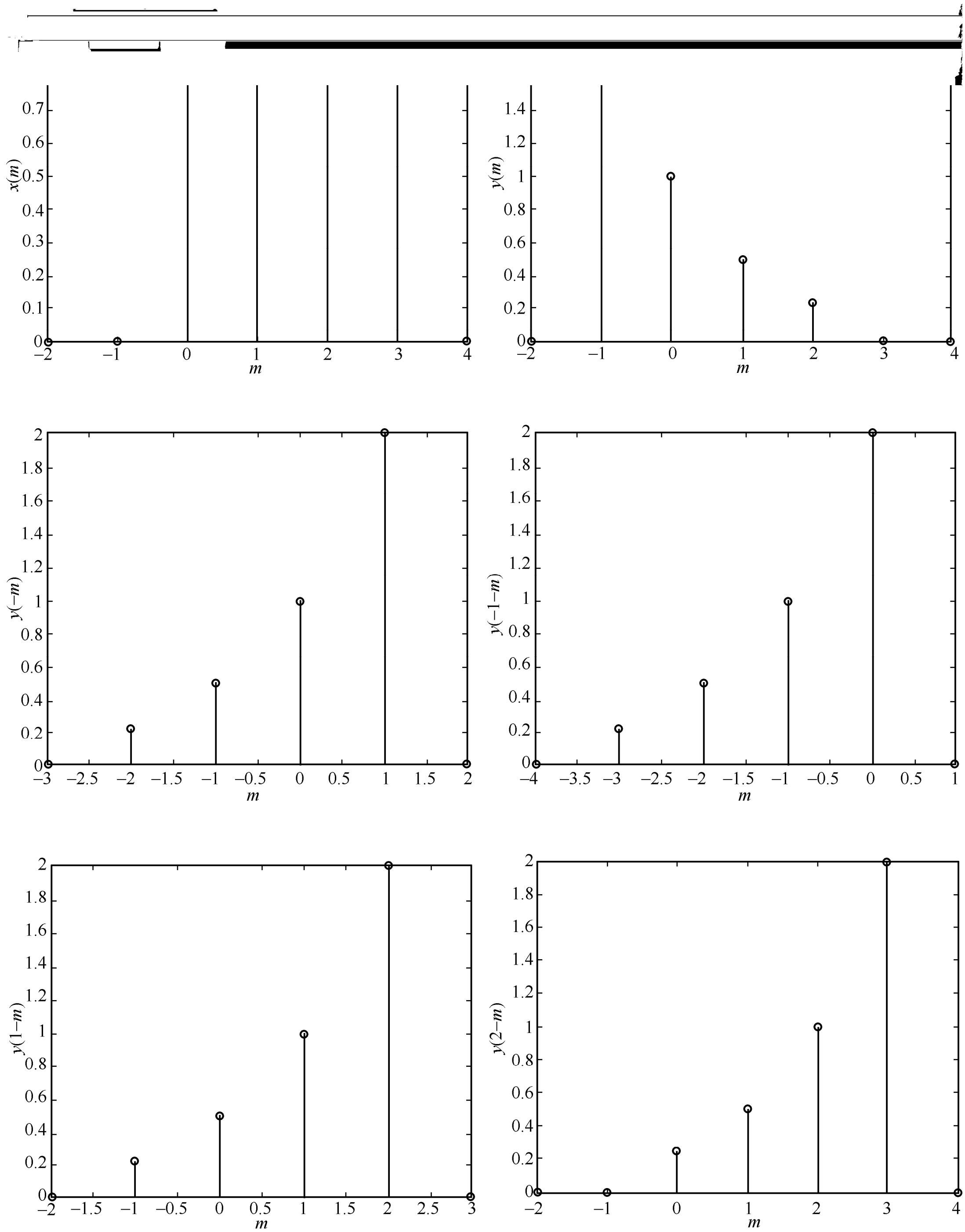


图 1-11　图解法求卷积和

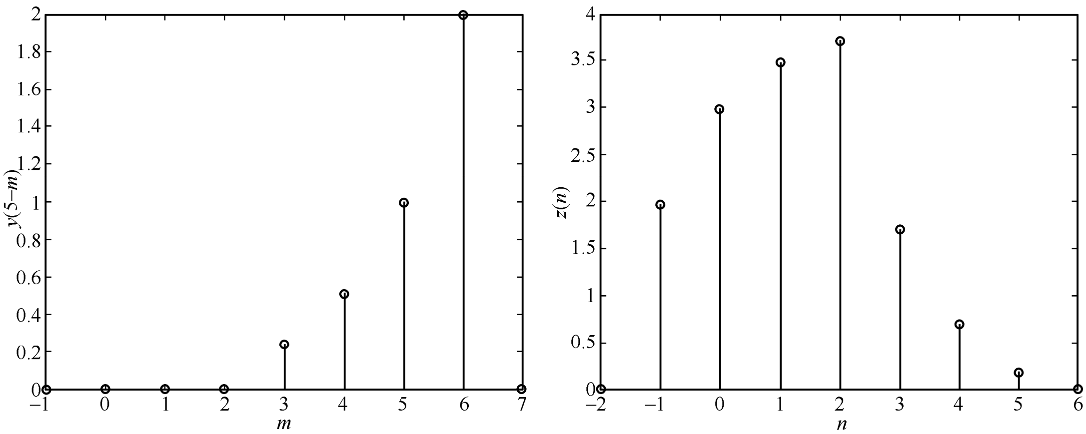


图 1-11　图解法求卷积和（续）

将 $x(n)$ 和 $y(n)$ 表示为 $x(m)$ 和 $y(m)$，并将 $y(m)$ 反褶成序列 $y(-m)$。$y(-m)$ 的移位序列 $y(n-m)$ 和 $x(m)$ 对应序列值对位相乘之后相加得到的结果就是卷积和 $z(n)$ 的值，可以分为以下三个区间。

① 当 $n<-1$ 时，移位序列 $y(n-m)$ 与 $x(m)$ 的非零值没有重叠部分，$z(n)$ 为 0。

② 当 $-1\leqslant n\leqslant 5$ 时，移位序列 $y(n-m)$ 与 $x(m)$ 的非零值部分重叠或全部重叠，此时卷积和为两个序列的对应序列值相乘之后求和的结果。

例如，$z(-1)=x(m)y(-1-m)=1\times 2=2$，这时 $x(m)$ 最左边序列值与 $y(-1-m)$ 最右边序列值刚好对齐，波形如图 1-11 中的 $x(m)$ 与 $y(-1-m)$。此时，卷积开始，结果 $z(-1)=1\times 2=2$。

接着，将 $y(-1-m)$ 右移一位，为 $y(0-m)$，对应波形为图 1-11 中的 $y(-m)$，与 $x(m)$ 相乘、求和得到卷积和序列的下一个序列值为

$$z(0)=\sum_{m=0}^{1}x(m)y(0-m)=1\times 1+1\times 2=3$$

同理，每右移一位，就进行对应序列值相乘、求和，得到一个对应的卷积和序列值，直到 $n=5$ 时序列 $y(n-m)$ 的最左边序列值正好与序列 $x(m)$ 的最右边序列值对齐。此时 $y(n-m)$ 对应图 1-11 中 $y(5-m)$，计算得到 $-1\leqslant n\leqslant 5$ 区间内序列 $z(n)$ 的值，即卷积和。序列 $z(n)$ 波形如图 1-11 中的 $z(n)$ 波形所示。

卷积和结果为

$$z(n)=\{\underset{\uparrow}{2},3,7/2,15/4,7/4,3/4,1/4\}$$

③ 当 $n>5$ 时，移位序列 $y(n-m)$ 与 $x(m)$ 的非零值没有重叠部分，卷积结束。

综上可得：

$$z(n)=\begin{cases}2 & n=-1\\ 3 & n=0\\ 7/2 & n=1\\ 15/4 & n=2\\ 7/4 & n=3\\ 3/4 & n=4\\ 1/4 & n=5\\ 0 & \text{其他}\end{cases}$$

（2）对角线列表法

根据卷积和定义

$$z(n)=x(n)*y(n)=\sum_{m=-\infty}^{\infty}x(m)y(n-m)$$

可以将序列 $x(n)$ 和序列 $y(n)$ 表示为

$$x(n)=\{x_1(n),x_2(n),x_3(n),\cdots\}$$
$$y(n)=\{y_1(n),y_2(n),y_3(n),\cdots\}$$

通过以下步骤求得序列 $x(n)$ 和序列 $y(n)$ 的卷积和。

第一步，以 $x_1(n),x_2(n),x_3(n),\cdots$ 为行标题，以 $y_1(n),y_2(n),y_3(n),\cdots$ 为列标题，列出如图 1-12 所示的表格，将序列 $x(n)$ 的每个序列值和序列 $y(n)$ 的每个序列值分别相乘，并将结果填入表中。

第二步，在表中画出如图 1-12 所示的对角线。

第三步，将对角线上的元素相加，得到

$$z(n)=\{x_1(n)y_1(n),[x_1(n)y_2(n)+x_2(n)y_1(n)],[x_1(n)y_3(n)+x_2(n)y_2(n)+x_3(n)y_1(n)],\cdots\}$$

由于卷积具有可交换卷积顺序的性质，参加卷积运算的两个序列的顺序是可以交换的。即可以选择任意一个序列作为行标题，将另外一个序列作为列标题。

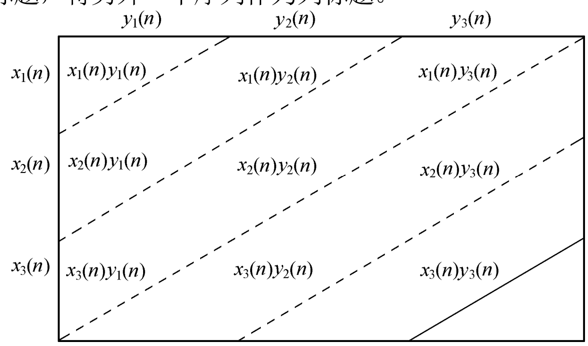


图 1-12 对角线列表法卷积图

【例 1-8】 已知 $x(n)=\{1,2,3\}$， $y(n)=\{1,1,1\}$，用对角线列表法求序列 $x(n)$ 和序列 $y(n)$ 的卷积和。

解：卷积和如图 1-13 所示。

$$x(n)*y(n)=\{1,1+2,1+2+3,2+3,3\}=\{1,3,6,5,3\}$$

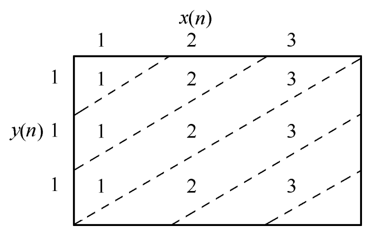


图 1-13 对角线列表法卷积和实例图

（3）相乘对位相加法

相乘对位相加法的计算过程类似于竖式乘法，通过例 1-9 说明其求解步骤。

【例 1-9】 用相乘对位相加法，求解例 1-7 中序列 $x(n)$ 和序列 $y(n)$ 的卷积和。

解：相乘对位相加法只适合于两个有限长序列的卷积和。

如图 1-14 所示，将两个序列的序列值分两行排列，逐位竖式相乘得到 4 行；从左到右将乘积各项按竖式对位相加，结果就是卷积和。

由例 1-9 可以看出，序列 $x(n)$ 的非零值区间为 $0\leqslant n\leqslant 3$，长度为 4；序列 $y(n)$ 的非零值区间为 $-1\leqslant n\leqslant 2$，长度为 4；卷积和 $z(n)=x(n)*y(n)$ 的非零值区间为 7，等于两个卷积因子序列长度之和减 1。事实上，若序列 $x(n)$ 的非零区间为 $n_1\leqslant n\leqslant n_2$，长度 $N_1=n_2-n_1+1$；序列 $y(n)$ 的非零值区间为 $n_3\leqslant n\leqslant n_4$，长度 $N_2=n_4-n_3+1$；则卷积和 $z(n)=x(n)*y(n)$ 的非零值区间为 $n_1+n_3\leqslant n\leqslant$

n_2+n_4，长度 $N=N_1+N_2-1$。

可见线性卷积长度为

$$N=N_1+N_2-1$$

其中，N_1、N_2 为两个参加卷积序列的长度。

卷积和运算满足交换律、结合律和分配律，如图 1-15 所示。

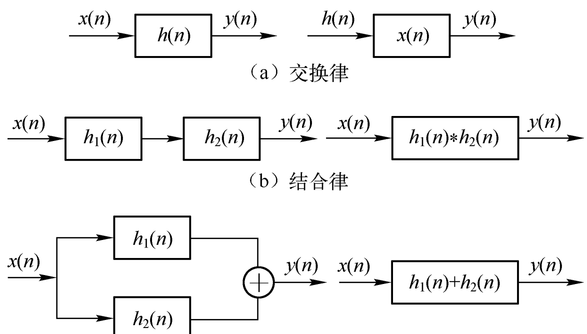
交换律：$x(n)*h(n)=h(n)*x(n)$

结合律：$x(n)*h_1(n)*h_2(n)=x(n)*[h_1(n)*h_2(n)]$

分配律：$x(n)*[h_1(n)+h_2(n)]=x(n)*h_1(n)+x(n)*h_2(n)$

	1	1	1	1			
×	2	1	1/2	1/4			
	2	2	2	2			
		1	1	1	1		
			1/2	1/2	1/2	1/2	
+				1/4	1/4	1/4	1/4
	2	3	7/2	15/4	7/4	3/4	1/4

图 1-14　相乘对位相加法



（a）交换律

（b）结合律

（c）分配律

图 1-15　卷积和运算的性质

要点九：序列 $x(n)$ 和序列 $y(n)$ 的卷积和定义为

$$z(n)=x(n)*y(n)=\sum_{m=-\infty}^{\infty}x(m)y(n-m)$$

求卷积和可以采用图解法、对角线列表法和相乘对位相加法。

卷积和序列的长度为两个卷积因子序列长度之和减 1。

卷积和运算满足交换律、分配律和结合律。

用 MATLAB 表示离散序列时，自变量的取值和序列值分别用两个向量表示。例如，$x(n)=\{2,2,\underset{\uparrow}{1},-1,0,3,1\}$ 可以表示为 $\mathrm{n}=-2:4$，$\mathrm{x(n)}=\{2,2,1,-1,3,1\}$，使用 stem(n,x)画序列波形。

离散卷积是数字信号处理中的一个基本运算，系统输出信号 $y(n)$ 是系统输入信号 $x(n)$ 与系统单位冲激响应 $h(n)$ 的卷积。MATLAB 提供了卷积函数 conv，卷积函数的调用方式为 $\mathrm{y=conv(x,h)}$。

【例 1-10】 计算序列 $x(n)$ 与序列 $y(n)$ 的卷积和。

$$x(n)=\begin{cases}1 & 0\leqslant n\leqslant 3\\ 0 & \text{其他}\end{cases}$$

$$y(n)=\begin{cases}2^{-n} & -1\leqslant n\leqslant 2\\ 0 & \text{其他}\end{cases}$$

解：计算序列卷积和的程序如下。

```
nx=0:3;
x=[1,1,1,1];
ny=-1:2;
y=2.^(-ny);
z=conv(x,y);卷积和计算
```

```
    n=nx(1)+ny(1):nx(end)+ny(end);卷积和区间范围计算
    stem(n,z);
    xlabel('n');ylabel('z');
```

程序运行的结果如图 1-16 所示。

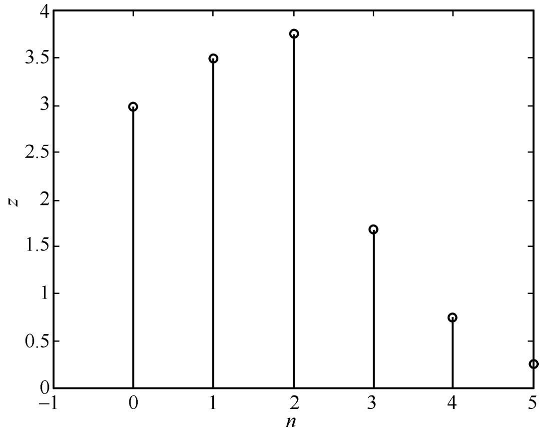


图 1-16　例 1-10 序列卷积和计算结果

1.2.3　序列的能量、周期性及常用典型序列

1．序列的能量

序列 $x(n)$ 的能量 E 定义为序列各抽样值的平方和

$$E=\sum_{n=-\infty}^{\infty}\left|x(n)\right|^2 \tag{1-12}$$

2．序列的周期性

如果存在正整数 N，使序列 $x(n)$ 对所有的 n 满足

$$x(n)=x(n+N) \tag{1-13}$$

则称 $x(n)$ 为周期性序列，使序列满足周期性的最小的正整数 N 称为序列 $x(n)$ 的周期。

3．常用典型序列

（1）单位抽样序列（单位冲激序列）$\delta(n)$

$$\delta(n)=\begin{cases}1 & n=0\\0 & n\neq 0\end{cases} \tag{1-14}$$

单位抽样序列类似于时域连续信号中的单位冲激函数 $\delta(t)$，但是 $\delta(n)$ 的定义是在 $n=0$ 处有确定取值的，而 $\delta(t)$ 是具有极限概念的信号。$\delta(n)$ 在时域离散信号与系统中的作用等同于 $\delta(t)$ 在时域连续信号与系统中的作用，但是计算要容易得多。单位抽样序列如图 1-17 所示。

要点十：单位抽样序列

$$\delta(n)=\begin{cases}1 & n=0\\0 & n\neq 0\end{cases}$$

（2）单位阶跃序列 $u(n)$

$$u(n)=\begin{cases}1 & n\geqslant 0\\0 & n<0\end{cases} \tag{1-15}$$

$u(n)$ 类似于连续时间信号中的单位阶跃函数 $u(t)$，所不同的是，$u(t)$ 在 $t=0$ 处没有确定值，而 $u(n)$ 在 $n=0$ 处定义为 1。

分析 $\delta(n)$ 与 $u(n)$ 的定义，可以得出两者的关系，$\delta(n)$ 是 $u(n)$ 的后向差分 $\nabla u(n)$，如下

$$\delta(n)=\nabla u(n)=u(n)-u(n-1) \tag{1-16}$$

如果将 $\delta(n)$ 移位以后再累加，就可以得到 $u(n)$，$u(n)$ 是 $\delta(n)$ 的移位累加和，如下

$$u(n)=\sum_{m=0}^{\infty}\delta(n-m)=\delta(n)+\delta(n-1)+\delta(n-2)+\cdots \tag{1-17}$$

令 $n-m=k$，得

$$u(n)=\sum_{k=-\infty}^{n}\delta(k) \tag{1-18}$$

单位阶跃序列如图 1-18 所示。

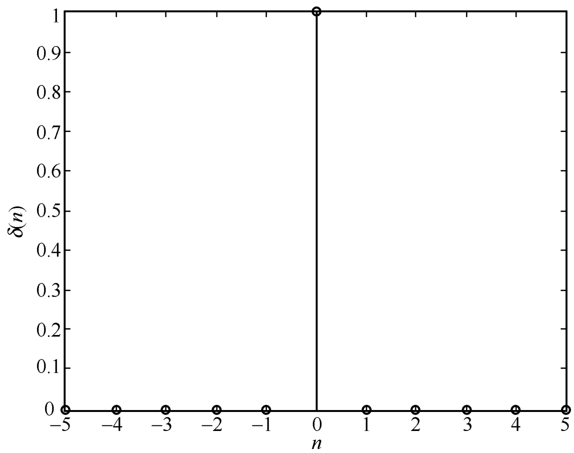


图 1-17　单位抽样序列

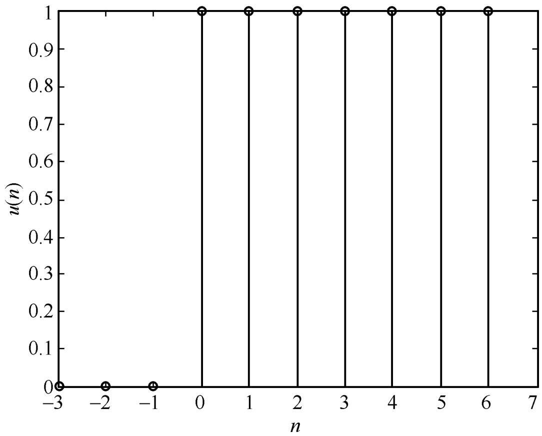


图 1-18　单位阶跃序列

要点十一：单位阶跃序列

$$u(n)=\begin{cases}1 & n\geqslant 0\\ 0 & n<0\end{cases}$$

要点十二：单位抽样序列与单位阶跃序列的关系：

① $\delta(n)=\nabla u(n)=u(n)-u(n-1)$

② $u(n)=\sum_{m=0}^{\infty}\delta(n-m)=\delta(n)+\delta(n-1)+\delta(n-2)+\cdots$

（3）矩形序列

矩形序列常用来进行长序列的截断处理。矩形序列的定义为

$$R_N(n)=\begin{cases}1 & 0\leqslant n\leqslant N-1\\ 0 & 其他\end{cases} \tag{1-19}$$

称 N 为矩形序列的长度。如序列

$$R_4(n)=\begin{cases}1 & 0\leqslant n\leqslant 3\\ 0 & 其他\end{cases}$$

为矩形序列，波形如图 1-19 所示。

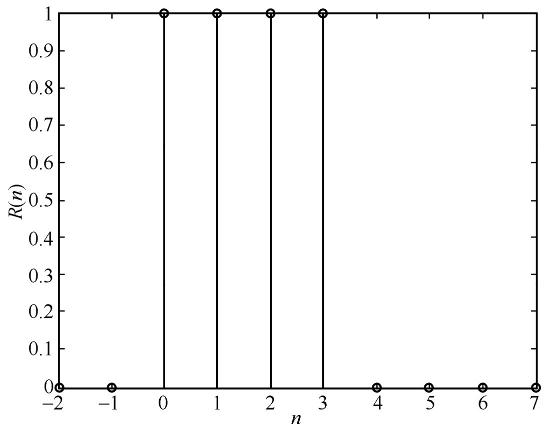


图 1-19　矩形序列实例

$R_N(n)$ 可以用 $u(n)$ 表示，也可以用 $\delta(n)$ 表示。$R_N(n)$ 与 $u(n)$ 的关系如式（1-20）所示，$R_N(n)$ 与 $\delta(n)$ 的关系如式（1-21）所示。

$$R_N(n)=u(n)-u(n-N) \tag{1-20}$$

$$R_N(n)=\sum_{m=0}^{N-1}\delta(n-m) \tag{1-21}$$

要点十三：矩形序列

$$R_N(n)=\begin{cases}1 & 0\leqslant n\leqslant N-1\\ 0 & \text{其他}\end{cases}$$

要点十四：矩形序列与单位抽样序列的关系

$$R_N(n)=\sum_{m=0}^{N-1}\delta(n-m)$$

要点十五：矩形序列与单位阶跃序列的关系

$$R_N(n)=u(n)-u(n-N)$$

（4）单边实指数序列

单边实指数序列的定义为

$$x(n)=a^n u(n) \tag{1-22}$$

式中，a 为实数。当 $|a|<1$ 时，序列 $x(n)$ 是收敛的；当 $|a|>1$ 时，序列 $x(n)$ 是发散的。单边实指数序列如图 1-20 所示，图 1-20（a）所示为 $a>1$ 的单边实指数序列 $x(n)$，图 1-20（b）所示为 $0<a<1$ 的单边实指数序列 $x(n)$，图 1-20（c）所示为 $a<-1$ 的单边实指数序列 $x(n)$，图 1-20（d）所示为 $-1<a<0$ 的单边实指数序列 $x(n)$。其中图 1-20（a）和图 1-20（c）为 $|a|>1$ 的情况，序列 $x(n)$ 是发散的；图 1-20（b）和图 1-20（d）为 $|a|<1$ 的情况，序列 $x(n)$ 是收敛的。

（5）复指数序列

复指数序列的定义为

$$x(n)=\mathrm{e}^{(\sigma+\mathrm{j}\omega_0)n} \tag{1-23}$$

式中，ω_0 称为数字角频率。复指数序列可以表示为实部与虚部

$$x(n)=\mathrm{e}^{(\sigma+\mathrm{j}\omega_0)n}=\mathrm{e}^{\sigma n}(\cos\omega_0 n+\mathrm{j}\sin\omega_0 n)=\mathrm{e}^{\sigma n}\cos\omega_0 n+\mathrm{j}\mathrm{e}^{\sigma n}\sin\omega_0 n$$

复指数序列还可以表示为模与相角

$$x(n)=\mathrm{e}^{(\sigma+\mathrm{j}\omega_0)n}=|x(n)|\mathrm{e}^{\mathrm{j}\arg[x(n)]}=\mathrm{e}^{\sigma n}\cdot\mathrm{e}^{\mathrm{j}\omega_0 n}$$