

数字信号处理的

理论与应用下

何振亚
编著

数字信号处理的理论与应用

下 册

何 振 亚 编著

人 民 邮 电 出 版 社

内 容 提 要

本书分为上、下两册,讨论用数字序列表示信号、用数字运算方法处理信号,以及把信号作各种离散变换、排除干扰,使信号容易识别等数字信号处理技术的基本理论和这门新技术的广泛应用。上册共六章,依次讲数字信号处理的基本原理、离散系统与线性变换、数字网络的理论基础、数字滤波器原理、数字滤波器综合及数字滤波器中的有限字长效应。下册共四章,依次讨论各种快速变换方法、随机信号分析、二维数字信号处理及数字信号处理的各种应用。本书内容丰富、便于阅读,是一本密切联系实际的基础理论著作,对从事数字通信、数字信号和图象信号处理的通信工程技术人员和通信专业的师生,均有参考价值。

数字信号处理的理论与应用

下 册

何 振 亚 编 著

*

人 民 邮 电 出 版 社 出 版

北京东长安街27号

河 南 省 邮 电 印 刷 厂 印 刷

新华书店北京发行所发行

各 地 新 华 书 店 经 售

*

开本: 850×1168 1/32 1983年12月第一版

印张: 17 12/32页数: 278 1987年8月河南第二次印刷

字数: 459 千字 印数: 9,001—11,480 册

统一书号: 15045·总2786-无6259

定价: 3.80 元

目 录

第七章 快速变换	1
第一节 引言	1
第二节 快速付里叶变换方法	4
一、FFT 算法的基本概念	4
二、时间抽取算法	7
三、频率抽取算法	15
四、IDFT 的运算方法	19
第三节 任意数的 FFT 算法	20
一、N 为高合成数的 FFT 算法——布鲁斯坦算法	21
二、N 为素数的 FFT 算法——Chirp-Z 变换算法	23
第四节 FFT 卷积与相关	37
一、重迭相加法	39
二、重迭保留法	41
第五节 FFT 的实现	44
一、通用机实现 FFT 算法的程序流程图	44
二、蝶形公式的实数运算法	47
三、反序重排算法	50
四、用 BASIC 算法语言实现 FFT	51
第六节 FFT 运算的误差分析	53
一、定点实现法中的误差	53
二、浮点实现法中的误差	62
第七节 维诺格兰算法 (WFTA)	66
一、基本概念	67
二、短序列的 WFTA	76
三、长序列的 WFTA	87
四、WFTA 程序流程图	98

第八节 沃尔什函数与变换	99
一、沃尔什函数	100
二、沃尔什函数的基本性质	106
三、离散沃尔什函数和变换	109
第九节 哈达玛矩阵和变换	119
一、哈达玛矩阵	119
二、哈达玛变换	121
第十节 快速哈达玛—沃尔什变换	123
一、快速哈达玛变换	124
二、快速沃尔什变换	128
三、序率滤波	132
第十一节 数论变换	136
第十二节 多项式变换	146
一、N 等于素数的 DFT 计算	148
二、 $N=2^m$ 的 DFT 计算	152
三、一维循环卷积的计算	155
四、二维循环卷积的计算	158
附录 7 A 新 FFT 算法 (Rader-Brenner 算法)	162
参考文献	164
第八章 谱分析与谱估计	167
第一节 引言	167
第二节 规则信号的谱分析	169
一、FFT 型谱分析器	171
二、滑动滤波型谱分析器	173
第三节 离散随机信号及其通过线性系统的响应	180
一、随机信号的基本特性	183
二、随机信号通过线性系统的响应	194
第四节 随机信号的谱估计	203
一、谱估计的基本概念	203
二、谱估计的质量评定	206
三、随机信号模型和谱估计方法	208

第五节 相关估计法	212
一、自相关估计原理	212
二、用 FFT 计算相关	215
三、快速沃尔什变换估算法	220
第六节 周期图法	223
一、周期图的基本概念	223
二、平均周期图法	227
三、平滑周期图法 (窗口函数法)	230
四、平滑周期图的平均法	234
第七节 最大似然估计法	237
一、预备知识	237
二、最大似然估计法	239
三、用自回归模型估算功率谱	242
第八节 最大熵谱分析法	245
一、熵的含义	246
二、平稳随机信号的熵	247
三、最大熵谱估计	248
四、用预测误差滤波法估算最大熵谱	252
五、最大熵谱的快速算法	256
六、最大熵谱公式阶数的确定	267
第九节 最小交叉熵谱分析法	269
一、最小交叉熵原理	270
二、最小交叉熵谱分析	272
三、线性滤波器输入与输出之间的交叉熵	275
四、最小交叉熵谱分析法与最大熵谱分析法的关系	276
第十节 自回归滑动平均信号模型法	281
一、ARMA 信号模型	282
二、ARMA 信号模型系数的确定	286
参考文献	288
第九章 二维数字信号处理	290
第一节 引言	290

第二节 二维序列与系统	291
一、二维序列	291
二、二维线性非移变系统	293
三、二维抽样定理	297
第三节 二维离散变换	302
一、二维 z 变换	302
二、二维离散付里叶变换	309
三、二维离散卡一洛变换	314
四、二维离散余弦变换	316
第四节 二维数字滤波	329
一、二维滤波的一般原理	329
二、二维数字滤波	334
(一) 二维信号的离散数学模型	335
(二) 二维数字滤波方法	338
(三) 二维约束解离散卷积滤波	341
第五节 二维非递归滤波器的设计	346
一、一般概念	346
二、二维窗函数设计法	350
三、二维频率抽样设计法	357
四、频率变换法	363
第六节 二维递归滤波器的稳定性	368
第七节 二维递归型滤波器的设计	378
一、空间域设计法	379
二、频率域设计法	383
第八节 二维递归滤波器幅频响应与群延迟的兼优设计	389
第九节 二维同态滤波处理	395
一、同态的基本概念	395
二、相乘信号的同态滤波处理	398
三、卷积信号的同态滤波处理	403
四、复倒谱与解卷积算法	406
(一) 复倒谱的性质	408

(二) 复倒谱算法	409
(三) 解卷积算法	412
第十节 二维卡尔曼滤波	414
一、卡尔曼滤波的基本概念	415
二、二维卡尔曼滤波的递推算法	418
参考文献	419
第十章 数字信号处理的应用	421
第一节 引 言	421
第二节 快速付里叶分析仪	422
一、仪器工作原理	423
(一) 奇型离散付里叶变换 (ODFT)	423
(二) 双奇型离散付里叶变换 (O^2DFT)	427
(三) 用 ODFT 计算实序列的卷积	430
(四) 用 ODFT 求相关函数	433
二、硬件实现方法	436
三、设计考虑	443
(一) FFT 运算误差	443
(二) 数据字长	445
(三) 动态范围	445
(四) 基数选择	446
四、高分辨率付里叶分析	446
第三节 数字处理接收机	449
第四节 PCM 通信中的数字处理与变换	453
一、数据调制解调	454
(一) 用数字电路置换模拟电路的方法	454
(二) 将输出波形写入 ROM 的方法	455
(三) 采用微程序信号处理的方法	456
二、数字压缩与扩展	458
(一) 压扩的数学法则	458
(二) 数字扩展电路	460
(三) 数字压缩电路	463

三、 ΔM 与 PCM 和 DPCM 的变换	465
(一) ΔM 变换为 PCM (DPCM)	465
(二) PCM (DPCM) 变换为 ΔM	468
四、数字 PCM 终端	469
第五节 时分与频分多路通信系统的数字化接口	471
一、用数字滤波器实现 TDM/FDM 变换	472
二、用 DFT 与多相数字网络级联实现 TDM/FDM 变换	476
(一) 基本原理	476
(二) 实输入的 DFT 与多相数字网络实现多路变换	480
(三) 用 WFTA 与多相数字网络实现多路变换	488
第六节 语音信号处理	488
一、语音产生的模型	489
二、语音信号的短时间频谱分析	490
三、线性预测声码器	494
四、同态声码器	503
第七节 雷达信号的数字处理	505
一、数字匹配滤波器	506
二、线性调频信号的数字式脉冲压缩	508
(一) 数字模糊函数	509
(二) 线性调频信号的压缩处理	511
三、相位编码信号的数字式脉冲压缩	513
(一) 二位相位编码信号	513
(二) 用数字相关法作精确测距	513
(三) 用数字相关法作自动距离跟踪	515
第八节 生物医学工程上的信息处理	515
一、医学检查和分析的数字处理	517
(一) 呼吸功能检查装置	517
(二) 心血管病诊断系统	518
二、医用图象处理	520
(一) 染色体的自动分类系统	520
(二) 计算机 X 射线断层摄影装置	521

参考文献	523
附录 III 数字信号处理快速算法的实用程序	525
附录III.1 FFT 算法的 BASIC语言程序.....	525
附录III.2 FFT算法的 FORTRAN语言程序.....	528
附录III.3 WFTA 算法的 BASIC 语言程序	531
附录 IV 谱分析的实用程序	535
附录IV.1 用周期图法计算功率谱的 BASIC-II语言程序.....	535
附录IV.2 实倒谱计算的 BASIC 语言程序	541
参考文献	545

第七章 快速变换

第一节 引言

从前几章的讨论中,我们已经清楚地看到在数字滤波和数字信号处理系统中,广泛地应用了卷积和相关的运算。如果直接应用时域法进行处理,计算量将随着抽样点数 N 的平方而增加。对于大的 N 值,就使计算机的负担加重,计算也很费时,难达到实时处理的要求。因此,一般都采用变换方法,就是把输入信号先进行变换,再进行处理,然后进行反变换,恢复为时域的信号。用计算机对变换后的信号进行处理,往往比直接处理更加方便,还有可能大大减少计算次数,提高处理速度。所以,在数字信号处理中要研究快速变换的算法。

在信号处理中,最常用的变换方法是离散付里叶变换(DFT),在数学解析作用方面,它与付里叶变换有着相类似的作用和性质,因而在离散信号分析与数字系统的信号处理中占有极重要的地位。它不仅建立了离散时域与离散频域之间的联系,而且由于离散付里叶变换存在周期性,它还兼有连续时域中付里叶级数的作用,与离散付里叶级数有着密切联系。此外,在计算速度方面,因为有计算DFT的快速算法,从而使离散付里叶变换的应用更十分普遍。它在实现各种数字信号处理系统中扮演着核心的角色。例如通过计算信号序列的DFT可以直接分析它的数字频谱,又在FIR滤波器的设计中,经常要从冲激响应 $h(n)$ 求频率抽样值 $H(k)$,以及它们之间的反演算。而且数字滤波器设计好之后,一般都要计算DFT,通过数字频谱的响应来检验设计效果,等等。

1965年柯利-杜开(Cooley-Tukey)提出了快速付里叶变换

(FFT)方法,这是一种计算 DFT 的快速程序。计算机使用了这种程序,可以使原来需要 N^2 次运算的算法变为只需 $\frac{N}{2}\log_2 N$ 次运算的算法。图 7.1 绘出了 DFT 直接算法和 FFT 算法所需乘法次数的比较,被变换的抽样点 N 越多,FFT 算法计算量的节省就越显著。例如在水声学中涉及频段在 $0\sim 10\text{ kHz}$ 的信号,较理想的谱分

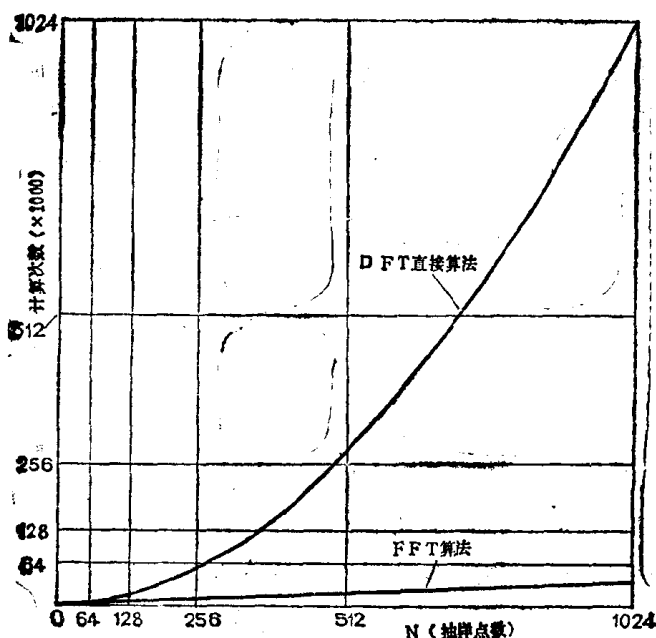


图 7.1 FFT 算法与 DFT 直接算法所需乘法次数的比较

辨率为 1 Hz , 这样就必须求得 10^4 个谱点, 如果直接运用 DFT 算法, 要求计算机进行 10^8 次运算, 就需要算几小时, 但是采用 FFT 算法就可快 1000 多倍, 大大缩短了计算时间。

FFT 算法包含有三种情况:

- (1) N 为 2 的整数幂的算法;
- (2) N 为高复合数时的算法;
- (3) N 为素数时的算法。

在本章中将分别进行讨论。

1976年维诺格兰(Winograd)发表了快速计算DFT的新算法(称为WFTA)*,它要求的乘法次数仅为FFT乘法次数的1/3,加法次数与FFT的几乎相同。他还证明当 N 为一素数时, WFTA算法所需要的乘法次数达到了理论上最小值。

理论上,除了离散付里叶变换的快速算法之外,还有一些其他的算法,就是用别的正交函数系来代替付里叶变换所用的三角函数系。我们知道,付里叶变换的基础是由正、余弦函数构成完备的正交函数系,使任何绝对可积的函数都可以表示为正、余弦函数,但是它要进行复数运算就比较费时。如果有其它合适的完备的正交函数系能避免这种复数运算,一定可以提高变换的运算速度。沃尔什(Walsh)函数系就是能够达到上述目的的完备的正交函数系,每个函数只取+1与-1两个值,就是所谓二值正交函数。它的特点是图形呈矩形,有跳变特性,在跳变点间仅取上述两个数值,这与数字逻辑电路中的两种状态相对应,同时满足了完备和正交特性,从而可以进行级数展开与函数变换。因此,以沃尔什函数为基础所构成的变换,是实数加减运算,可以想到其运算速度,一定会比付里叶变换为快。通常为了便于与所熟知的频域法相区别,把沃尔什变换这类解析方法称为序域法。用这种变换法实现数字滤波的系统称为序率滤波器,又叫做列率滤波器。

我们还要介绍数论变换及其类似算法的多项式变换,这都是最近几年发展起来的一些新的变换算法。例如费马数变换,它所用的数学工具是数论。这种变换只用到整数,运算是按模整数的同余数式规则进行的,计算速度也比FFT为快,而且计算中没有舍入误差。多项式变换是以模多项式代替模整数,不象数论变换有字长的限制,可以在实数或复数的通用运算器内完成算术操作。

* WFTA是Winograd Fourier Transform Algorithm的缩写

第二节 快速付里叶变换方法

我们知道,应用电子数字计算机来计算付里叶变换必须是离散付里叶变换(DFT),它的正变换是

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad k=0,1,\dots,N-1 \quad (7.1)$$

逆变换(IDFT)是

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}, \quad n=0,1,\dots,N-1 \quad (7.2)$$

其中 $W_N = e^{-j\frac{2\pi}{N}}$ $x(n)$ 和 $X(k)$ 可以是实数,也可以是复数。由上式可见,要计算一个抽样数据序列就需要做 N 次复数乘法, $N-1$ 次复数加法,但是后面可以知道,采用快速付里叶变换(FFT)算法可以减少计算量,尤其当 $N=2^M$ 时更是如此。这节就先讨论基 2 (即 $N=2^M$) 的算法。

一、FFT 算法的基本概念

为了说明计算过程,现以 $N=2^2=4$ 为例,列出计算方程如下:

$$X(k) = \sum_{n=0}^3 x(n) W_4^{kn} \quad (7.3)$$

或展成

$$X(0) = x(0)W_4^0 + x(1)W_4^0 + x(2)W_4^0 + x(3)W_4^0$$

$$X(1) = x(0)W_4^0 + x(1)W_4^1 + x(2)W_4^2 + x(3)W_4^3$$

$$X(2) = x(0)W_4^0 + x(1)W_4^2 + x(2)W_4^4 + x(3)W_4^6$$

$$X(3) = x(0)W_4^0 + x(1)W_4^3 + x(2)W_4^6 + x(3)W_4^9$$

写成矩阵形式

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} \quad (7.4)$$

利用 W_N^{kn} 的两个特点就可以提高计算效率。第一个特点是 $W_N^{k(N-n)} = (W_N^{kn})^*$ 的对称特性，因为 $W_N^{k(N-n)} = W_N^{-kn} = (W_N^{kn})^*$ 。

第二个特点是 $W_N^{kn} = W_N^{k(n+N)} = W_N^{(k+N)n}$ 的周期特性，在上例中 $N=4$ 。

$$W_4^4 = W_4^0 = 1; W_4^6 = W_4^{2+4} = W_4^2, W_4^9 = W_4^{1+2 \times 4} = W_4^1$$

把这些关系代入上面矩阵方程中，并重排其次序，得到

$$\begin{bmatrix} X(0) \\ X(2) \\ X(1) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_4^2 & 1 & W_4^2 \\ 1 & W_4^1 & W_4^2 & W_4^3 \\ 1 & W_4^3 & W_4^2 & W_4^1 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix}$$

把上式中的系数矩阵用虚线分成四个子矩阵，可见左上角和右上角两个子矩阵完全相同，左下角的子矩阵是

$$\begin{bmatrix} 1 & W_4^1 \\ 1 & W_4^3 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & W_4^2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & W_4^1 \end{bmatrix}$$

在右下角的子矩阵中，考虑到 $W_4^2 = (e^{-j\frac{2\pi}{4}})^2 = e^{-j\pi} = -1$ ，并把 -1 乘到一个子矩阵内的各元素上去，于是简化为

$$\begin{bmatrix} W_4^2 & W_4^3 \\ W_4^2 & W_4^1 \end{bmatrix} = W_4^2 \begin{bmatrix} 1 & W_4^1 \\ 1 & W_4^3 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & W_4^2 \end{bmatrix} \begin{bmatrix} -1 & 0 \\ 0 & -W_4^1 \end{bmatrix}$$

因此，可得

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_4^2 & 1 & W_4^2 \\ 1 & W_4^1 & W_4^2 & W_4^3 \\ 1 & W_4^3 & W_4^2 & W_4^1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & & \\ 1 & W_4^2 & 0 & \\ & & 1 & 1 \\ 0 & & 1 & W_4^2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & W_4^1 & 0 & -W_4^1 \end{bmatrix}$$

令

$$\begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ \hline 1 & 0 & -1 & 0 \\ 0 & W_4^1 & 0 & -W_4^1 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} = \begin{bmatrix} x(0) + x(2) \\ x(1) + x(3) \\ x(0) - x(2) \\ [x(1) - x(3)]W_4^1 \end{bmatrix} \quad (7.5)$$

再注意到 $W_4^2 = W_2 = -1$, 于是

$$\begin{aligned} \begin{bmatrix} X(0) \\ X(2) \\ X(1) \\ X(3) \end{bmatrix} &= \begin{bmatrix} 1 & 1 & & \\ & 1 & W_4^2 & \\ \hline & 0 & 1 & 1 \\ & 1 & W_4^2 & \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \end{bmatrix} = \\ &= \begin{bmatrix} 1 & 1 & & \\ & 1 & -1 & \\ \hline & 0 & 1 & 1 \\ & 1 & -1 & \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ x_1(3) \end{bmatrix} \end{aligned}$$

或写成

$$\begin{bmatrix} X(0) \\ X(2) \\ X(1) \\ X(3) \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix} x_1(0) \\ \begin{bmatrix} 1 & -1 \end{bmatrix} x_1(1) \\ \begin{bmatrix} 1 & 1 \end{bmatrix} x_1(2) \\ \begin{bmatrix} 1 & -1 \end{bmatrix} x_1(3) \end{bmatrix} = \begin{bmatrix} x_1(0) + x_1(1) \\ x_1(0) - x_1(1) \\ x_1(2) + x_1(3) \\ x_1(2) - x_1(3) \end{bmatrix} \quad (7.6)$$

因此, 按照(7.5)式和(7.6)式分两步来计算 DFT, 就是一种快速付里叶变换算法, 它只需用一次乘法和 8 次加法, 而不是象直接计算 DFT 需要用 16 次乘法和 12 次加法。其所以能这样减少乘法运算, 除了上述正交函数的两个特点之外, 主要因为使用矩阵析因法, 可以在矩阵中填充若干零值。由此可看出, 采用矩阵析因法能够大大提高 FFT 的计算效率。

虽然上面例子是对 DFT 正变换进行演算的, 但是, 对比

(7.1)和(7.2)式可以看出, FFT 算法在 IDFT 的计算中, 同样存在这种优越性, 下面我们以一般化表达式来分析 FFT 算法。

二、时间抽取算法

FFT 的原始柯利-杜开形式相当于按时间抽取算法, 后来桑德(Sande)提出的一种改进形式是按频率抽取算法, 称为桑德-杜开方法。

(一) 基本原理

为了大幅度地提高计算效率, 把 DFT 的计算分成小步来进行, 并在计算过程中, 充分利用复指数正交函数系的对称性和周期性。因为 $N=2^M$ 是偶数, 所以可以将序列 $\{x(n)\}$ 分解为两部分, 一是偶数部分 $x(2n)$, 一是奇数部分 $x(2n+1)$, 这里 $n=0, 1, 2, \dots, \left(\frac{N}{2}-1\right)$, 也就是序列 $\{x(n)\}$ 是由 $N/2$ 个偶数样本点和 $N/2$ 个奇数样本点所组成。这就是说, 序列 $\{x(n)\}$ 的 DFT 可以用两个 $N/2$ 抽样点的 DFT 表示和计算, 因此, (7.1)式可以写成

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} [x(2n)W_N^{2nk} + x(2n+1)W_N^{(2n+1)k}],$$

$$k=0, 1, 2, \dots, N-1. \quad (7.7)$$

因为 $W_N^2 = e^{-2j\frac{2\pi}{N}} = e^{-j\frac{2\pi}{N/2}} = W_{N/2}$, 所以上式可写成

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} x(2n)W_{\frac{N}{2}}^{nk} + W_N^k \sum_{n=0}^{\frac{N}{2}-1} x(2n+1)W_{\frac{N}{2}}^{nk}$$

$$= G(k) + W_N^k H(k) \quad (7.8)$$