

数字信号处理程序库

数字信号处理程序库

中国电子学会信号处理学会
仪器仪表委员会 编译
《数字信号处理程序》编译组

清华

清华大学出版社

及社

数字信号处理程序库

中国电子学会信号处理学会
仪器仪表分会编译
“数字信号处理程序库”编译组

清华大学出版社

内 容 简 介

本书是根据美国电气工程学会出版的《数字信号处理程序》编译而成。全书共八章，内容包括：快速付里叶变换子程序、功率谱分析与相关函数、快速卷积、语言信号的线性预测分析、有限冲激响应数字滤波器的设计与综合、无限冲激响应数字滤波器的设计与综合、倒谱分析、插入法和抽取法等，以及编制这些程序的用途、原理和应用这些程序的说明。书中 28 个程序，也已在具有 CDOS CP/M 磁盘操作系统的微型机上通过，并已制成 8 吋软磁盘，方便用户使用。该程序库的建立，为部分微型机用户减免了大量通用性的编程工作，为计算机的资源共享作出了积极的贡献。

本书适用电子技术、自动化技术、计算机用户、航天、地震、气象、语言识别等科研及工程部门从事数字信号处理工作的广大科技人员，高等院校有关专业师生。

JS430/11

数 字 信 号 处 理 程 序 库

中国电子学会信号处理学会
仪器仪表学会 编译
“数字信号处理程序库”编译组

★

清华大学出版社出版

北 京 清 华 园

清华大学印刷厂印刷

新华书店北京发行所发行 · 各地新华书店经售

开本：787×1092 1/16 印张：44 字数：1070 千字

1983 年 7 月第一版

1983 年 7 月第一次印刷

印数 1~35000

统一书号：15235·72

精装定价：8.50 元

平装定价：7.00 元

编 者 的 话

一九八一年七月在青岛召开的中国电子学会、中国仪器仪表学会信号处理学会成立大会上,根据目前国内数字信号处理科研的急需,决定编译出版数字信号处理程序汇编,并且以提供文本和磁盘的方式发行,为使具有同类微型机的用户可以减免大量通用性的编程工作,便于把精力更集中于自己所开发的必需从事的信号处理研究上来。这一决定受到了与会全体委员的一致赞同,并在会上推选了由北方交通大学、清华大学、科学院声学所与北京自动化技术研究所组成编译小组。会上还决定编译工作所需经费由电子学会与电子工业部科技司联系从科研项目拨款,由北方交通大学信息研究所负责召集和组织全部工作。在工作进行中,南京工学院无线电系也参加了部分工作。

经过一年的努力,在信号处理学会的领导下,在电子工业部的支持下,28个程序已经全部在国内微型机(CROMEMCO-Ⅲ)
(BCM-Ⅱ)上通过,并将程序与其解释打印成文本。目前编译工作已告结束,文本由清华大学出版社出版。

书中28个程序已复制成软盘,由北方交通大学信息所发行。

我们选定的全部程序是根据美国电气工程学会出版的“Programs for Digital Signal Processing”所作的。因为这里所介绍的程序经过使用比较可靠,为将此书介绍的程序适用于微型计算机,程序中某些内容作了修改。其中第六章第一节,由于占内存太多,微型机无法运算,因此对这一节仅做了翻译工作。目前介绍的全部程序,已可直接用于教学及科研。所有程序的名称及符号全部引用原书,这主要为了今后便于相互交流。

编译工作的分工为:

北方交通大学负责:2.1, 2.2, 2.3, 4.3, 5.1, 5.2, 7.1, 7.2节;清华大学负责:1.4, 1.5, 1.6, 1.8, 1.9, 3.1, 6.1, 6.2, 6.3, 6.4节;科学院声学所负责:1.1, 1.2, 4.1, 4.2节;北京自动化技术研究所负责:5.3, 5.4, 8.1, 8.2, 8.3节;南京工学院负责:1.3, 1.7节。

参加此项工作的有:北方交通大学袁保宗, 姜乃英, 徐祥军, 马秀莲, 林碧琴, 尹作友, 陆鹏飞;清华大学茅于杭, 江泽同, 贺美英, 王文渊, 王普, 李兆玉, 杨存荣, 杨德发, 张大力, 董晓雪, 程丰;科学院声学所李昌立, 莫福源;北京自动化技术研究所王端新, 郑梦龄, 韩宝玲, 王荣凤, 赵玲;南京工学院沈祝平, 邹子英。

本书由袁保宗, 茅于杭, 姜乃英审校。

由于程序库汇编工作是我们第一次搞,没有经验,另外对这些程序虽然都做了校验,但还没有被我们广泛地使用过,因此在推广之中可能有许多不妥之处,希望读者批评指正。

最后要强调的是:这项工作得到中国电子学会学术委员会主席罗沛霖总工程师、中国仪器仪表学会一机部仪表总局陈令同志的大力支持,信号处理学会主任常邈教授和副主任北京自动化研究所潘甦民所长的直接关心,在工作人员的辛勤劳动下才得以顺利完成的。在此表示衷心的感谢。另外我们出版发行这本程序库,也对原书的编者及程序作者表示感谢。

编译者 于北京

一九八二年十月

目 录

前言	1
第一章 快速付里叶变换子程序	3
1.1 FOUREA——一个 FFT 简要说明的范本	17
1.2 快速付里叶变换算法	25
1.3 具有特殊性质序列的 FFT 子程序	50
1.4 混合基快速付里叶变换	77
1.5 最优海量存储 FFT 程序	101
1.6 Chirp Z-变换算法程序	142
1.7 复数通用 N Winograd 付里叶变换程序 (WFTA)	160
1.8 基 4 高效快速付里叶变换	188
1.9 二维混合基海量存储付里叶变换	204
第二章 功率谱分析与相关函数	221
2.1 功率谱估计的周期图方法	222
2.2 功率谱估计的相关方法	235
2.3 相关谱估计和互谱估计程序	253
第三章 快速卷积	281
3.1 FASTFILT——一个用快速付里叶变换的滤波程序	282
第四章 语言信号的线性预测分析	314
4.1 线性预测分析程序 (AUTO—COVAR)	316
4.2 线性预测的有效格点法	324
4.3 线性预测系数变换子程序 LPTRN	339
第五章 FIR 滤波器设计及综合	348
5.1 FIR 线性相位滤波器设计程序	350
5.2 FIR 开窗滤波器设计程序——WINDOW	367
5.3 设计具有最平通带与阻带的对称 FIR 低通数字滤波器子程序 MXFLAT	391
5.4 用于有限字长有限冲击响应滤波器 (FIR) 设计的子程序	398
第六章 IIR 滤波器的设计和综合	423
6.1 递归数字滤波器的设计程序	425
6.2 用最小 P 误差准则综合递归数字滤波器的程序	535
6.3 设计数字滤波器转移函数的最优化程序	589
6.4 设计有限字长 IIR 数字滤波器的程序	617
第七章 倒谱分析	639
7.1 复倒谱的计算	641
7.2 实倒谱的计算和最小相位重建	657
第八章 插入法和抽取法	667
8.1 用于数字插入器设计的计算机程序	669
8.2 用有理比率法实现数据采样率转换的通用程序	675
8.3 多级抽取插入和窄带滤波的程序	683

前 言

在过去的十五年里, 数字信号处理已极其活跃。随着集成电路技术和处理机结构的发展大大地扩大了数字信号处理技术的应用范围。基本信号处理技术和算法的研究促使信号处理系统的效率获得很大的改善。

数字信号处理取得进展的一个重要方面是算法的发展和这些算法的计算机程序的具体化, 包括信号处理的实现和信号处理滤波器以及系统的设计。本书的目的在于, 以实用的形式, 成为能广泛用于数字信号处理方面的一本计算机程序的综合资料。此外, 本书可作为在程序方面精心努力的结果, 并且本书能使编写程序的作者由于他们的工作而赢得信誉。

本程序经过细致的选择, 包括了数字信号处理的应用及其设计技术一个很宽的范围。本程序分为八章, 每章都有专门的提要。在第一章中重点叙述了离散付里叶变换 (DFT) 并且给出了各种快速付里叶变换 (FFT) 及其有关的算法。在第二章中包括了功率谱估计的周期图和相关法算法以及相关谱估计和互谱估计的算法。快速 FFT 为基础的卷积程序放在第三章中叙述。在第四章中给出了有关线性预测信号处理技术的几种算法, 包括自相关法, 协方差法和格点法。在第五章中主要阐述了有限冲激响应 (FIR) 数字滤波器的设计与综合。在第五章中还包括了最优化的, 窗函数的和最平滤波器设计的算法以及考虑有限字长影响的设计程序。在第六章中给出了有关无限冲激响应 (IIR) 数字滤波器设计程序的综合资料。其中第一个程序包括了大部分典型的滤波器设计技术以及对有限字长问题的考虑, 如按配对、按单一顺序而不是模块。其它程序着重叙述了有限字长设计的问题或迭代最优化滤波器的设计。在第七章里叙述了倒谱和同态算法, 侧重于同态系统相位计算的难点。最后, 在第八章里给出了有关内插和抽取的几个程序。在数字系统里, 为改变取样率所需的基本运算, 包括多级实现和按有理比率的取样率变换。

本书是在 1976 年初, 由 IEEE 声学, 语言学以及信号处理学会所组成的数字信号处理委员会, 着手进行的一项计划的结晶。本书与以前在 IEEE 出版社再版的丛书相比有所不同, 本书包括了很多以前从未发表过的材料。此外, 关于出版计算机程序的一些特殊问题必须认真对待。这些问题是: (1) 文件和程序说明的明瞭和一致性; (2) 单独校核和检验全部程序; (3) 可移植性以及和机器无关的编码; (4) 计算机可读源程序的有效性。因此, 这项计划对委员会来说, 对编写程序作者和校核者来说是一项十分紧迫而耗费时间的工作。

为了实现这项计划, 委员会在 1976 年初, 给数字信号处理学会一份初步计划和一份所需程序的建议。这项工作既通过直接写信方式进行, 又利用在 1976 年声学, 语言学以及信号处理国际会议上分派下去。在短时间内, 由于很多作者愿意编写程序, 使委员会坚信, 本书不仅能实现, 而且能为数字信号处理迈出新的一步。此外, 把这些材料初步编排为本书的八章。

由于编写程序作者和校对者的辛勤努力, 我们可以根据所提供的材料确定每一程序, 并选

出适合于我们所需的那些程序。为防止实现同一函数出现两种程序，通常，这种选择是根据所选的程序更加符合于文件所规定的出版格式，可移植性，通过预先使用所证实，或者其它类似的标准。

每一程序选定之后，给作者发出请贴，要求作者提出程序的全部编码和文件。要求作者的所有程序符合下列条件：

- (1) 必须包括全部用户文件。
- (2) 程序必须用 FORTRAN 语言，并且符合 ANSI FORTRAN 标准。
- (3) 编码应全部注释并已经排除错误，尽可能地清晰，明瞭。
- (4) 编码应能传送，并且与机器无关。
- (5) 应提供有关输入，输出综合调试实例。

当然，除打印之外，计算机可读全部编码的源程序只得由作者提供。这带来更繁重的任务，即集中所提供的磁带和程序卡片，把它们读入中心档案系统，并且以几种可能的形式之一，把这些磁带和卡片分给校对者。

本委员会认为计算机可读源程序(这些程序适用于本书读者)是这项计划的基本部分。显然，把全部程序集中到中心档案对完成此项计划是必需的。此外，由于 IEEE 出版社 Reed Crone 和 IEEE 服务中心 Jack Fraum 的合作，在需求方面，在以微薄的成本制作本书全部程序的磁带方面，我们都能安排 IEEE 来进行翻印和销售业务。对于，如何订购这种磁带，可与 IEEE 出版社，345 East 47th st, 纽约，纽约 10017 联系。值得强调的是，这是 IEEE 过去从未搞过的一项新业务，也许，其它项目中，也能开展类似业务。

作者提出程序后，校对工作才能进行。对每个校对者的要求，其一：充当一名能严厉批评程序的用户。其二：充当一名象编辑杂志一类技术性资料的校对者。要求校对者能阅读程序，编译程序，实际运行所给的调试实例以及其它类似的调试实例。校对工作是大量的，校对者需要频繁地与作者磋商一些难点。多数情况下，作者的程序是好的，校对工作相当顺利。然而，在某些情况下，特别是程序的可移植性方面，一连串的困难和问题就暴露出来，需要在校对过程中改正。

程序的可移植性在这项计划里是很重要的目的。一次成功的校对不会完全保证达到这个目的。因此，在程序标准化方面，进行了广泛的研究工作。

特别注意了保证精度，明瞭以及打印材料的一致性。所有程序编码的列表，直接从中央收集计算机可读源材料中打印出。特别注意选择一种清楚的字体。避免直接使用作者原稿上的各式各样字体照相复制，避免采用誊写，排版列表，而造成不可避免的错误。作者的文件手稿打入计算机系统之后，进行最后的编辑和校正。再由计算机带动照相排版机得到书中全部打印材料。

尽管这是整个数字信号处理委员会的工作，在这里，一些做出杰出贡献的个人值得专门提及。1976 年初，由 Ronald Crochiere 向委员会首次提出这项工作的想法。这项工作由委员会主席 Alan.v.Oppenheim 指导下开始工作。开始时，Crochiere 负责从数字信号处理专业同仁中征求、收集程序。1976 年 9 月 Clifford Weinstein 担任委员会主席。并且从那时起，为本书担任总的协调人。

第一章 快速付里叶变换子程序

J.W.Cooley and M.T.Dolan

引 言

下面是包含在本书中许多优秀的离散付氏变换 (DFT) 的一个小样例。本书所选用的每个程序都有某些特殊的优点, 借助这些优点使程序易于使用和理解, 并在特殊应用中更为有效。每个子程序前所包含的资料将更好地使用户应用它。子程序清单中的评论语句给出参数的简略描述, 在某种程度上也描述了子程序的内部运算。详细的描写是完整的, 描写本身又是充分的, 所以在这导论中仅作一点评述, 以使用户在许多子程序中选出适合他自己特殊应用的子程序。对用户更进一步的考虑, 下面表 1 至表 5 的详细描述, 给出了存贮量的要求、执行时间和所有子程序调试计算的精度。

FOUREA——FFT 的一个简明范本 (1.1)

C.Rader 和文献^[1]中的其他人初始所写的子程序 FOUREA 描述和解释了 FFT 算法, 且可用作写专用程序或微处理器程序一个模式。通常这样的程序并不倾向于将它用作运算程序。然而, 在某些终端操作系统和解释方面, 计算机运行时间可较少考虑, 由于这类程序简短易懂, 因此在这方面可以有效地使用它。

快速付氏变换法 (1.2)

G.D.Bergland 和 M.T.Dolan 在有效的基 2—4—8 子程序上作出了贡献, 他们写出了完全不同目的的子程序。这产生了十分快的程序, 其代价是大量的指令存贮和程序的复杂性。第一个离散付氏变换—逆离散付氏变换 (DFT—IDFT) 对 FAST—FSST 计算 1024 点实数据变换所需的时间大约是前面 FOUREA 程序的十分之一。但在 Honeywell 6080N 计算机上需存贮 1940 个指令字, 而 FOUREA 仅需 260 个。程序 FFA—FFS 做 FAST—FSST 同样的工作, 但它包含一个基 8 运算, 因此运行更快, 对 1024 点变换, 用 134 毫秒代替 FAST—FSST 的 156 毫秒。其代价是指令存贮字是 3210 个而不是 1940 个。在此软件包中的第三个程序是 FFT842, 它计算复数输入的 DFT, 对 1024 点复数变换需要 250 毫秒, 这大约是 FAST 所需时间的二倍。然而, 应该记住 FAST 和 FFA 计算实数输入而不是复数输入, 因而它们的计算量比 FOUREA 和 FFT842 的一半略为多一些。

特殊性质序列的 FFT 子程序 (1.3)

L.R.Rabiner 所写的对特殊对称性序列的 FFT 子程序在取数据对称性以节省计算量这优点方面是特别有用的。那样的对称性出自数据的固有性质是经常出现的。例如, 自相关函数总是实数和偶函数。在某些切比雪夫 (Chebyshev) 近似中出现仅有奇次谐波的余弦序

列。在解边界条件一端为零值、另一端导数为零的线性偏微分方程时，有奇次谐波的正弦序列是有用的。这些程序是用 FOUREA 那样的方式来写的，亦即它是简短和易于阅读的。正弦和余弦值是需要计算的，而不是查表方式。计算是用没有简化特殊情况的“DO”循环语句来进行，这样做的原因是相对大量的时间是花在(1.3)的子程序中，其中大部分时间是花在内部调用 DFT 子程序上。这里给出的程序是调用 FAST 和 FSST 子程序，但要强调的是这些子程序可以被其他有效的 DFT 子程序代替。例如，假如 N 不是 2 的幂次或数据是在某些群存贮单元中，可以用 Singleton 的(1.4)或 Fraser 的(1.5) FFT 子程序。

前四个程序 FFTSYM、IFTSYM、FFTASM 和 IFTASM 调用 FAST 和 FSST 来计算真正的余弦和正弦变换。通常那样的程序在程序文献中是十分有用的。某些最新贡献的 DFT 文献提供了详细的描写，表明如何进一步取对称性的优点，这对称性相应于在 DFT 中仅有奇次谐波。这杰出的想法结合到子程序中用带有“OH”的字母来命名，意即“奇次谐波”(odd harmonics)。例如，子程序 FFTOHM 有双重多余度的实数据的 DFT；它的前一半是后一半的负数，因此仅 $N/2$ 个数据需要存贮和仅 $N/2$ 个实数点需要计算。贮存量节省了一半，计算结果也节省了差不多一半。由于数据对零的对称和反对称性，最后四个以“SOH”和“AOH”命名的子程序，用了附加的双重多余度。这些结果进一步减少了存贮量和计算时间。表 2a 和 2b 给出的计算时间表明：随着 N 增加，对每个双重多余度数据，节省约一半。假如 N 不是 2 的幂次，子程序调用可以改变为下一部分的混基子程序。但是，假如 N 不是 2 的幂次，对前 6 个子程序，它必须是 4 的倍数，对后 4 个子程序它必须是 8 的倍数。

混基快速付氏变换 (1.4)

所有上面的子程序要求 N 是 2、4 或 8 的幂次，因为它是 FFT 方法最简单和最有效的形式。R.C.Singleton 所作的混基方法允许 N 除包含 2 以外的其他因子。一般来说， N 的质因子越大，从运算数目的观点看，对每个输出点计算的有效性越小。

虽然上面描述的程序可用于多维变换，这是简单的计算每一维中每个矢量点的变换，但大量的子程序调用使这方法收效甚微。混基 FFT 法的第二个重要特点是对此情况它可有效地用于多维复数或实数变换。对每一维调用子程序一次，这子程序计算这一维中这些点的 DFT，但是在最内层循环，子程序对所有其他维进行重复计算。表 2a 表明，对 2 的幂次，程序 FFT 要花和程序 FFT842 差不多的时间。增加灵活性的代价是 FFT 需要 230₀ 个指令字，而 FFT842 只需 1430 个。

最佳群存贮 FFT 程序 (1.5)

上面子程序所用数据阵的大小，当然取决于有效存贮磁芯的数量。机器大小和数据阵大小的取决，有时可以用如磁盘存贮器这样的随机寻址器件来保留数据阵。D.Fraser 的最佳群存贮 FFT 使这尽可能容易和有效。用户定义一个有效的缓冲存贮区大小，当数据驻留在里面时，提供给子程序来寻址这存贮区。按照作者设计和发表的方法，子程序计算最佳块的大小，安排磁芯存贮以及群存贮体间数据传输处于一种最佳方式。用户应该自己去熟悉全部文献和应该花些气力去建立读、写子程序。但这对十分大的付氏变换，将形成一个十分有

效的程序。假如 N 不是 2 的幂次或假如是一个二维矩阵, 则 Singleton 的 2 维群存贮混基 FFT (1.9) 可能更为合适。

线性调频 (Chirp) Z 变换算法程序 (1.6)

L.Rabiner 的线性调频 Z 变换程序可以放在这一部分中, 因为在某种意义上, 它是一个一般化的 DFT。当 DFT 考虑作为在单位圆上 N 个等距离点估算 Z 变换方法时, 线性调频 Z 变换给出在 Z 平面上任意螺线上许多等空间点的 Z 变换值。在某种意义上, 这是一个一般化的离散信号谱分析, 对离散信号, 它起着和在连续信号情况下作拉普拉斯 (Laplace) 变换同样的作用。这里给出的程序版本是用 FFT842, 但如同本书中许多其他程序一样, FFT842 也可为其他适当的 DFT 子程序代替。

一般复数 N 的 Winograd 付氏变换算法 (WFTA) (1.7)

J.H.McClellan 和 H.Nawab 的一般复数 N 的 Winograd 算法十分不同于 FFT。就使用来说, 假如 N 是 2 的幂次, 应该使用 FFT 子程序, 但假如另有重要原因, N 中有别的因子, 对某些 N 值, 用混基 DFT (1.4) 运行时间也许可改善 40%。没有一个一致的规律, 对描述的 N 值范围, 使用某个程序或某种方法更好, 所以必须考虑列出一张详细写出的表来。WFTA 子程序和它所要求的子程序同任何 FFT 子程序比较, 它需要非常大的指令存贮。在鉴别合理的运行时间效果方面, 在计算 DFT 前, 人们也必须考虑运行 N 值初始化程序所需的时间。因而, 当进行同样 N 的计算数目时, WFTA 子程序只是在普遍意义上更可取些。

时间有效基 4 快速付氏变换 (1.8)

当许多具有同样 N 值的 DFT 进行运算时, 对于合理的时间效果, L.R.Morris 的时间有效基 4 快速付氏变换是调整存贮器的极端例子, 因为这种情况在数字信号处理中特别典型, 将它考虑为对本书有价值的贡献。这里所用的程序是用文献中描述的 Morris 的自生技术产生的。本质上, 这程序取了给定程序的所有独立数据部分, 在此情况下, 上面所述的 Singleton 程序的基 4 部分将这些独立数据部分放在预处理阶段, 并有效地使一些内循环不缠绕起来, 得到某些简单运算的特殊情况, 对特殊场合, 这产生通常的最佳码。看完这个例子之后, 读者也许感兴趣于和作者联系使程序进一步最佳化, 以产生使用自生技术的信号处理程序的其他最佳形式。

二维混基群存贮付氏变换 (1.9)

R.C.Singleton 群存贮混基 2 维 DFT 本质上是上面所描述的他的混基程序的自适应化, 来完成 Fraser 程序所作的工作。主要的不同是这程序略为易于使用, 它不要求 N 是 2 的幂次, 和它要求较少的指令存贮。然而, 它局限于应用在 2 维矩阵上, 并且可能不象 Fraser 程序那样是最佳的。

存贮量和时间

表 4 包含了用户调用子程序的清单, 以及每个子程序所需要子程序的清单。给出的存贮

量要求是每个用户调用子程序的总存贮量,当然,这不包括调用子程序对数据和其他变量的存贮。在 Honeywell 6080N 计算机上,运行总存贮量(run—totals)包括作者提供的采样程序空间、数据矩阵、FFT 子程序和系统子程序。

表 2a 到 2d 以毫秒为单位的变换和逆变换平均计算时间,变换长度为 $N = 2^m$, $m = 1, 2, \dots, 15$ 的随机序列,其中某些长度相当于用户长度数量级的子程序。表 3a 至 3d 包含了每个程序平均运行时间的标准偏差,这可以解释为:计算时间和给定时间的差应有 68% 的概率小于标准偏差。

精 度

在几次运算 DFT 和 IDFT 时,初始随机序列之间会有差异,表 4 给出了平均值。表 5 给出了使用表 4 时误差的标准偏差。按 Kaneko 和 Liu^[1]的分析,在 DFT 中误差的偏差为:

$$\sigma_{\delta a}^2 = H(m)\sigma_a^2 \quad (1)$$

这里 σ_a^2 是元素 $a(n)$, $n = 0, 1, \dots, N-1$ 的均方值。

$$H(m) = 0.5(5m-6)\sigma_e^2 + 0.25(25m^2 - 51m + 18)\mu^2 \quad (2)$$

这里 $m = \log_2 N$, μ 和 σ_e^2 分别是平均偏差和舍入误差。从本书中 DFT 的定义,我们有 $\sigma_a^2 = N\sigma_x^2$ 在 IDFT 中误差是二个误差的和,一个是计算 DFT 的误差,另一个是计算 IDFT 的误差。前一个误差是 DFT 误差 $\delta_a(n)$ 的 IDFT,按 Parseval 定理,这得到附加的贡献 $\sigma_{\delta a}^2/N$ 加到误差偏差上。对 IDFT 计算本身产生的误差,除了 a 为 x 代替外,为上面给定的公式给出。假定所有的误差是独立的,对形成 $x(j)$ 的偏差,我们得到:

$$\sigma_{\delta x}^2 = 2H(m)\sigma_x^2 \quad (3)$$

$\sigma_x^2 = 1/3$ 给出:

$$\sigma_{\delta x}^2 = \left(\frac{2}{3}\right)H(m) \quad (4)$$

假定 m 适当大,机器是适当舍入的,即 $\mu = 0$,将产生一个标准偏差近似为下式的误差:

$$\sigma_{\delta x} = \sqrt{5m/3} \sigma_e \quad (5)$$

在机器中截断代替舍入,这里 $\mu \neq 0$,在 $H(m)$ 中带有 μ 项起支配地位,所以近似得到:

$$\sigma_{\delta x} = \frac{5}{\sqrt{6}} m_\mu \approx 2m_\mu \quad (6)$$

从表 4 和表 5 可以看出,误差粗略地和这估算符合。

参 考 文 献

1. J.W-Cooley, P. A. W. Lewis, and P. D. Welch, "The Fast Fourier Transform Algorithm and its Application", *IEEE Trans. on Education*, Vol. E-12, No.1, PP. 27-34,

March 1969.

2. T. Kaneko and B. Liu, "Accumulation of Round-Off Error in Fast Fourier Transforms", *J. Assoc. Comp. Mach.*, Vol. 17, No. 4, PP. 637-654, October 1970.

表 1

带有 FORTRAN-Y 编译器的 Honeywell 6080N 计算机对存贮量的要求

"Run—Total" 包括所有 FFT 子程序、作者提出的主测试程序和 Honeywell 6080N 计算机上运行所需的系统子程序。

"total" 在软件包中所有 FFT 子程序所要求的存贮量。

"Subtotals" 指独立的子软件包所需的存贮量。

1.1 FOUREA—FFT 的一个简明范本

C.M.Rader

子程序名称	字
FOUREA	260
Run—Total = 10K	Total = 260

1.2 快速付氏变换算法

G.D.Bergland and M.T.Dolan

子程序名称	字
FAST	220
FSST	220
FR2TR	50
FR4TR	570
FR4SYN	570
FORD1	80
FORD2	230
FFA	260
FFS	260
R2TR	50
R4TR	90
R8TR	1060
R4SYN	90
R8SYN	1090
ORD1	80
ORD2	230
FFT842	500
R2TX	80

001

R4TX	150
R8TX	700

Run - Total = 17K

Total = 6580

子软件包	Subtotals
FAST,FSST,FR2TR,FR4TR,FR4SYN,FORD1,FORD2	1940
FFA,FFS,R2TR,R4TR,R8TR,R4SYN,R8SYN,ORD1,ORD2	3210
FFT842,R2TX,R4TX,R8TX	1430
FAST,FR2TR,FR4TR,FORD1,FORD2	1150
FSST,FR2TR,FR4SYN,FORD1,FORD2	1150
FFA,R2TR,R4TR,R8TR,ORD1,ORD2	1770
FFS,R2TR,R4SYN,R8SYN,ORD1,ORD2	1800

1.3 具有特殊性质序列的 FFT 子程序

L.R.Rabiner

子程序名称	字
FFTSYM	240
IFTSYM	300
FFTASM	210
IFTASM	220
FFTOHM	250
IFTOHM	200
FFTSOH	180
IFTSOH	270
FFTAOH	180
IFTAOH	240
FAST,FSST—见 1.2子软件包	1940

Run - Total = 24K

Total = 4230

子软件包	Subtotals
FFTSYM,FAST 子软件包	1390
IFTSYM,FSST 子软件包	1450
FFTSYM,IFTSYM,FAST—FSST 子软件包	2480
FFTASM,FAST 子软件包	1360
IFTASM,FSST 子软件包	1370
FFTASM,IFTASM,FAST—FSST 子软件包	2370
FFTOHM,FAST 子软件包	1400
IFTOHM,FSST 子软件包	1350

FFTOHM,IFTOHM,FAST—FSST 子软件包	2390
FFTSOH,FFTOHM,FAST 子软件包	1580
IFTSOH,IFTOHM,FSST 子软件包	1620
FFTSOH,IFTSOH,FFTOHM,IFTOHM,FAST—FSST 子软件包	2840
FFTAOH,FFTOHM,FAST 子软件包	1580
IFTAOH,IFTOHM,FSST 子软件包	1590
FFTAOH,IFTAOH,FFTOHM,IFTOHM,FAST—FSST 子软件包	2810

1.4 混基快速付氏变换

R.C.Singleton

子程序名称	字
FFT	350
FFTMX	1690
REALS	260
REALT	370
SORTG	340
NORMAL	60
RMS	110
ISTKGT	110
ISTKRL	150

Run - Total = 35K

Total = 3440

子软件包	Subtotals
FFT,FFTMX,ISTKGT,ISTKRL 子软件包	2300
FFT,FFTMX,ISTKGT,ISTKRL,REALS 子软件包	2560
FFT,FFTMX,ISTKGT,ISTKRL,REALT 子软件包	2670
FFT,FFTMX,ISTKGT,ISTKRL,REALS,REALT 子软件包	2930

1.5 最佳群存贮 FFT

D.Fraser

子程序名称	字
RMFFT	120
CMFFT	270
MFCOMP	380
MFSORT	230
MFREV	320
MFLOAD	150

MFINDX	170
FMSUM	90
MFRCMP	830
MFRLOD	280
MFPAR	340
DMPERM	140
RANMF	60
NAIVE	350
MFREAD	50
MFWRIT	60
Run - Total = 19K	Total = 3840

1.6 线性调频 Z 变换算法程序

L.R.Rabiner

子程序名称	字
CZT	760
RECUR	140
DECUR	130
FFT842 - 见 1.2 子软件包	1430
Run - Total = 17K	Total = 2460

1.7 一般复数 N Winograd 付氏变换算法 (WFTA)

J.H.McClellan and H.Nawab

子程序名称	字
INISHL	600
CONST	600
WFTA	42470
WE1AVE	1280
WEAVE2	1480
Run - total = 65K	Total = 46430

1.8 时间有效基 4 快速付氏变换

L.R.Morris

子程序名称	字
RADIX4	6900
RAD4SB	90
Run - Total = 21K	Total = 6990

1.9 二维混基群存贮付氏变换

子程序名称	字
FFT2T	350
FFT21	350
XFR	40
TRNSP	80
EXCH	50
MFREAD	50
MFWRIT	40
Run - Total = 71K	total = 960
子软件包	Subtotals
FFT2T,XFR,TRNSP,MFWRIT,MFREAD,EXCH 子软件包	610
FFT2T,XFR,TRNSP,MFWRIT,MFREAD,EXCH 子软件包	610

表 2a
平均时间 (以毫秒为单位)

N	FOUREA	FAST—FSST	FFA—FFS	FFT842	FFT
2	0.767	1.020	.998	0.820	0.888
4	1.510	1.510	1.100	1.060	1.100
8	3.190	1.810	1.740	2.040	2.370
16	6.790	2.960	2.360	3.370	3.110
32	14.700	4.820	4.110	6.000	6.230
64	31.200	8.770	7.370	12.000	11.800
128	66.900	17.400	14.500	25.600	28.800
256	143.000	35.200	29.200	52.600	54.700
512	304.000	75.000	60.900	110.000	122.000
1024	649.000	156.000	134.000	250.000	261.000
2048	1380.000	339.000	282.000	533.000	629.000
4096	2900.000	706.000	595.000	1120.000	1240.000
8192	6090.000	1520.000	1290.000	2470.000	2740.000
16384	12900.000	3180.000	2680.000	5140.000	5680.000
32768	27200.000	6790.000	5680.000	10800.000	12800.000

表 2b
平均时间 (以毫秒为单位)
有特殊性质的序列

N	FFTSYM— IFTSYM	FFTASM— IFTASM	FFTOHM— IFTOHM	FFTSOH— IFTSOH	FFTAOH— IFTAOH
2	0.225	0.213	0.202	0.211	0.208
4	1.560	1.450	1.480	0.225	0.219
8	2.170	1.960	2.080	2.100	1.930
16	2.770	2.500	2.760	2.910	2.680
32	4.380	4.000	4.190	3.780	3.480
64	7.230	6.450	6.950	5.840	5.430
128	13.300	12.000	12.600	9.880	9.060
256	26.100	23.200	24.700	17.900	16.700
512	52.300	46.500	49.500	35.100	32.400
1024	109.000	97.400	103.000	69.800	64.800
2048	223.000	200.000	212.000	143.000	133.000
4096	467.000	421.000	445.000	286.000	266.000
8192	971.000	882.000	933.000	607.000	569.000
16384	2040.000	1860.000	1980.000	1250.000	1170.000
32768	4230.000	3870.000	4080.000	2600.000	2440.000

表 2c
平均时间 (以毫秒为单位)

N	WFTA	
	初始化所需时间	继续运算时间
4	1.540	0.838
8	2.210	1.380
16	3.980	2.450
30	8.490	5.330
60	16.400	9.890
120	34.500	20.900
240	81.700	47.700
504	173.000	108.000
1008	416.000	252.000
2520	1040.000	728.000
5040	2360.000	1670.000