增刊 第2期 自动化学报
1985年12月 ACTA AUTOMATICA SINICA 11卷

递推快速傅立叶变换(RFFT)[1)]

朱飞虹
（北京钢铁学院）

摘　要

本文提出了一种新的基于快速傅立叶变换的递推算法，给出了算法的推导和证明。此算法输入数据的数目和递推次数可任意选择，所以运算灵活，变换结果更接近实际频谱，且易于工程实现。算法可推广至多维情况。

快速傅立叶变换（FFT）已在许多领域内广泛使用。但在应用 FFT 时只能对一定数量的采样数据进行变换，而且必须在全部数据得到后才能开始计算。本文针对上述两个问题提出一种递推快速傅立叶变换的综合算法，基本思想是对每次输入的 N个数据进行快速傅立叶变换，根据递推公式重复，直至全部数据处理完毕。由于大部分计算可利用采样间隔时间进行，所以可提高计算机的利用率，加快变换速度，这种方法适用于时变谱分析、自适应控制等。可应用微计算机进行在线分析。

一、算　法

定义. $x(t)=\begin{cases} X(t), & 0\leqslant t<N\Delta t \\ 0, & t<0,\ t>N\Delta t, \end{cases}$

有
$$x^*(t)=\sum_{m=-\infty}^{+\infty}X(m\Delta t)\,\delta(t-m\Delta t)$$
$$=\sum_{m=0}^{N-1}X(m\Delta t)\,\delta(t-m\Delta t).$$

对 $x^*(t)$ 取傅氏变换，　$F_{x^*}(\omega)=\sum_{m=0}^{N-1}X(m\Delta t)e^{-jm\omega\Delta t}.$

在一组指定频率处

$$F_{x^*}(\omega_K)=\sum_{m=0}^{N-1}X(m\Delta t)e^{-jm\omega_k\Delta t},\qquad k=0,\ 1,\ 2,\cdots,A-1 \tag{1}$$

对 $x^*(t)$ 的镜象函数　$\hat{x}^*(t)=\sum_{m=0}^{N-1}X((N-1-m)\Delta t)\,\delta(t-m\Delta t)$

本文于1983年12月16日收到。

1）RFFT为Recursive Fast Fourier Transformation.

有
$$F\hat{x}^*(\omega_k)=\sum_{m=0}^{N-1}X((N-1-m)\Delta t)e^{-i m\omega_k\Delta t}. \tag{2}$$

可以证明文献〔1〕中$Fx^*(\omega_k)$ 和$F\hat{x}^*(\omega_k)$ 有关系式
$$|Fx^*(\omega_k)|^2=|F\hat{x}^*(\omega_k)|^2, \tag{3}$$
$$\psi x^*(\omega_k)=-[\psi\hat{x}^*(\omega_k)+(N-1)\Delta t\omega_k]. \tag{4}$$

引用递推关系$\mathbf{z}(\omega_k, m)=L(\omega_k)\mathbf{z}(\omega_k, m-1)+\mathbf{b}X(m)$, $m=0, 1, \cdots, N-1$, 其中
$$\mathbf{z}(\omega_k, m)=\begin{bmatrix}Z_1(\omega_k, m)\\ Z_2(\omega_k, m)\end{bmatrix}, \quad \mathbf{z}(\omega_k, -1)=\begin{bmatrix}0\\0\end{bmatrix},$$
$$\mathbf{b}=\begin{bmatrix}1\\0\end{bmatrix}, \qquad L(\omega_k)=\begin{bmatrix}\cos(\omega_k\Delta t) & -\sin(\omega_k\Delta t)\\ \sin(\omega_k\Delta t) & \cos(\omega_k\Delta t)\end{bmatrix}.$$

$L(\omega_k)$具有正交性质 $L(\omega_k)^m=\begin{bmatrix}\cos(m\omega_k\Delta t) & -\sin(m\omega_k\Delta t)\\ \sin(m\omega_k\Delta t) & \cos(m\omega_k\Delta t)\end{bmatrix}.$

用数学归纳法可证明
$$\mathbf{z}(\omega_k, N-1)=\sum_{m=0}^{N-1}L(\omega_k)^m\mathbf{b}X(N-1-m)$$
$$=\begin{bmatrix}\sum_{m=0}^{N-1}X(N-1-m)\cos(m\omega_k\Delta t)\\ \sum_{m=0}^{N-1}X(N-1-m)\sin(m\omega_k\Delta t)\end{bmatrix}=\begin{bmatrix}Z_1(\omega_k, N-1)\\ Z_2(\omega_k, N-1)\end{bmatrix}.$$

由（2）式有 $F\hat{x}^*(\omega_k)=[Z_1(\omega_k, N-1)-iZ_2(\omega_k, N-1)]$.

每次输入N个数据，可证其递推关系式为
$$\mathbf{z}(\omega_k, hN-1)=HX_{hN-1,\ (h-1)N}+L(\omega_k)^N\mathbf{z}(\omega_k, (h-1)N-1), \tag{5}$$
其中 $h=1, 2, \cdots$为递推次数。

$H=[L(\omega_k)^0\mathbf{b}\ L(\omega_k)b\cdots L(\omega_k)^{N-1}b]$, $X_{hN-1,\ (h-1)N}=[X(hN-1)\ X(hN-2)\ \cdots X((h-1)N)]^T$.

式（5）中后一项是2×2矩阵和一个二维向量相乘，是修正项，对第一项采用FFT计算。

令 $[u(0)u(1)\cdots u(N-1)]^T=[X(hN-1)X(hN-2)\cdots X((h-1)N)]^T$,

则
$$HX_{hN-1,\ (h-1)N}=\begin{bmatrix}\sum_{m=0}^{N-1}u(m)\cos(m\omega_k\Delta t)\\ \sum_{m=0}^{N-1}u(m)\sin(m\omega_k\Delta t)\end{bmatrix}.$$

用复数表示
$$C(\omega_k)=\sum_{m=0}^{N-1}u(m)e^{-im\omega_k\Delta t}$$
$$=\sum_{m=0}^{N-1}u(m)e^{-i2\pi mk/A}, \quad \omega_k=K\frac{2\pi}{\Delta t\cdot A}.$$

上式是在N个数据下求取A个频谱点，与通常的FFT不同，下面举例说明。

令 $$W = e^{-j2\pi/A},$$

则 $$C(k) = \sum_{m=0}^{N-1} u(m)\, w^{km}.$$

将m和k用二进制形式表示

$$m = m_{n-1}2^{n-1} + m_{n-2}2^{n-2} + \cdots + m_0,\quad m_i \in \{0, 1\},\quad i = 0, 1, \cdots, n-1,\quad n = \log_2 N$$

$$K = K_{a-1}2^{a-1} + K_{a-2}2^{a-2} + \cdots + K_0,\quad K_i \in \{0, 1\},\quad i = 0, 1, \cdots, a-1,\quad a = \log_2 A,$$

$$C(k) = \sum_{m=0}^{N-1} u(m) W^{km}$$

$$= \sum_{m_0}\sum_{m_1}\cdots\sum_{m_{n-1}} u(m_{n-1}, m_{n-2}, \cdots, m_0) W^{k(m_{n-1}2^{n-1}+m_{n-2}2^{n-2}+\cdots+m_0))}$$

当N = 4，A = 8时

$$C(k) = \sum_{m_1}\sum_{m_0} u(m_1, m_0) W^{k(m_1 2+m_0)}.$$

令 $$M_1 = \sum_{m_1} u(m_1, m_0) W^{2m_1 k} = \sum_{m_1} u(m_1, m_0) W^{2m_1(2k_1+k_0)},$$

M_1是k_0，k_1，m_0的函数，记为$\tilde{X}_1(k_0, k_1, m_0)$。

$$C(k) = \sum_{m_0} \tilde{X}_1(k_0, k_1, m_0) W^{km_0}) W^{km_0}$$

$$= \sum_{m_0} \tilde{X}_1(k_0, k_1, m_0) W^{km_0(2^2k_2+2k_1+k_0)}.$$

同理，令$M_0 = \sum_{m_0} \tilde{X}_1(k_0, k_1, m_0) W^{m_0(2^2k_2+2k_1+k_0)}$，记 为 $\tilde{X}_2(k_0, k_1, k_2)$。对$\tilde{X}_1(k_0, k_1, m_0)$ 和$\tilde{X}_2(k_0, k_1, k_2)$ 展开得计算流程图，如图 1 所示。

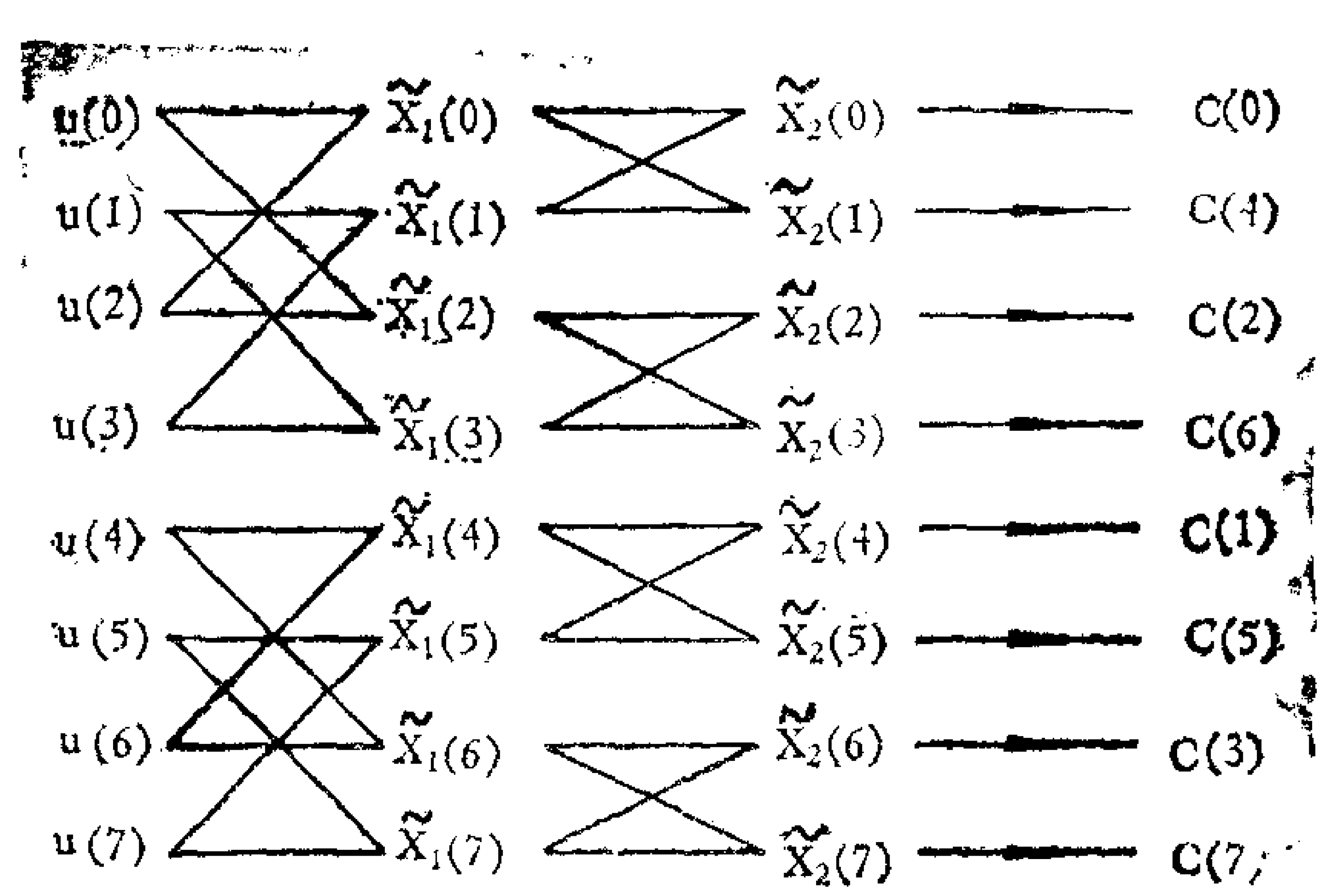

图 1

把流程图和通常的FFT流程图[2]相比较，可以看到RFFT中的快速傅氏变换：1）数据重复排列使用，重复次数$j = A/N$；2）第一次迭代相应于通常FFT中第 $a-n+1$ 次迭代（$a = \log_2 A$，$n = \log_2 N$）。其余与FFT计算流程图相同。

最后经h次递推后，得振幅及相位谱

$x^*(t)$：　　$$|F x^*(\omega_k)|^2 = \| Z(\omega_k, hN-1) \|^2,$$

$$\psi_{\hat{x}*}(\omega_K) = tg^{-1}\frac{-Z_2(\omega_K,hN-1)}{Z_1(\omega_K,hN-1)}.$$

$x*(t)$：

$$|F_{x*}(\omega_K)|^2 = \| Z(\omega_K,hN-1) \|^2,$$

$$\psi_{x*}(\omega_K) = -\left\{tg^{-1}\left[\frac{-Z_2(\omega_K,hN-1)}{Z_1(\omega_K,hN-1)}\right] + (hN-1)\omega_K\Delta t\right\}.$$

计算框图如图2所示。

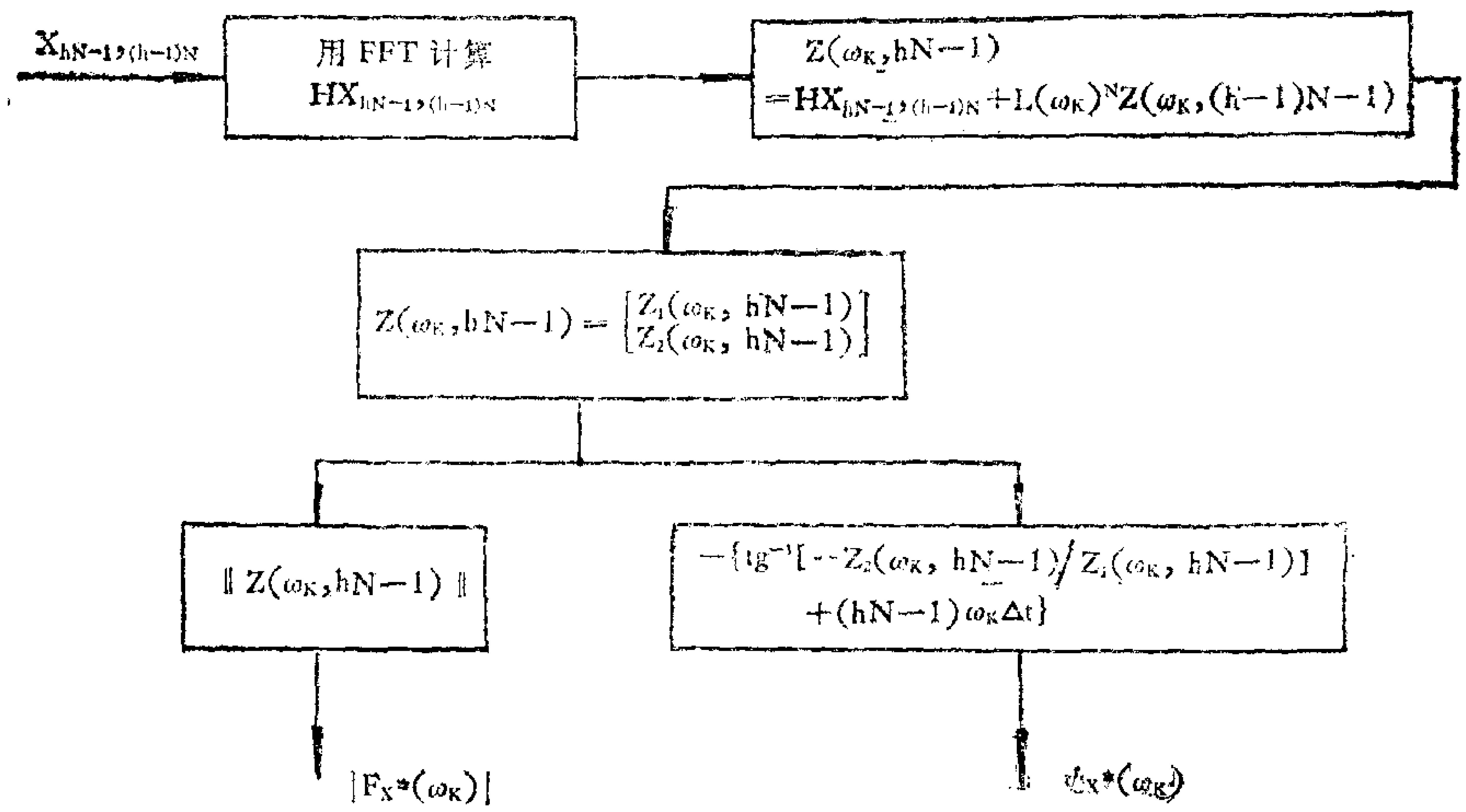

图2

在计算机上对RFFT算法进行仿真实验的结果表明：1）不管每次输入数据N为多少，只要数据总数和频率点数相等，则和普通FFT变换的结果完全相同。这是因为RFFT是从式（2）推出，FFT是从式（1）推出，而式（1）和（2）有确定关系即式（3）和式（4），所以当 $N=A$ 时二者变换结果相同；2）当输入数据的数目大于频率点数时，变换结果更接近连续频谱。图3是对一阶系统 $G(S)=1/S+1$ 的脉冲响应 $h(t)=e^{-t}$ 采用RFFT得到的频谱，曲线1是连续函数 e^{-t} 的频谱，曲线2是对 e^{-t} 采样，每次输入数据 $N=16$ 递推4次得到的频谱，曲线3是 $N=16$ 递推2次得到的频谱（频率点数均为 $A=32$）。图4是 $h(t)=2e^{-t}+3e^{-2t}$ 的变换结果。

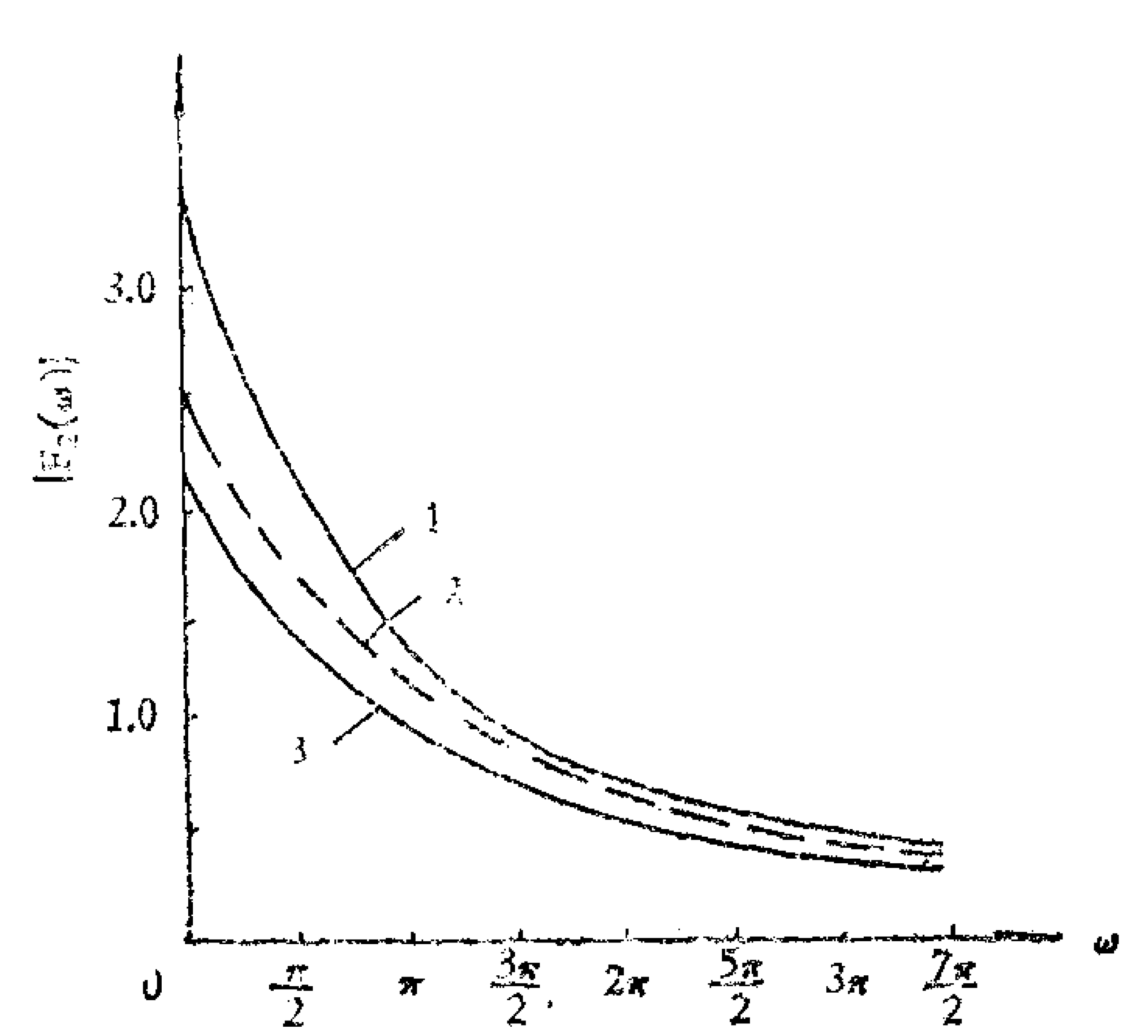

图3

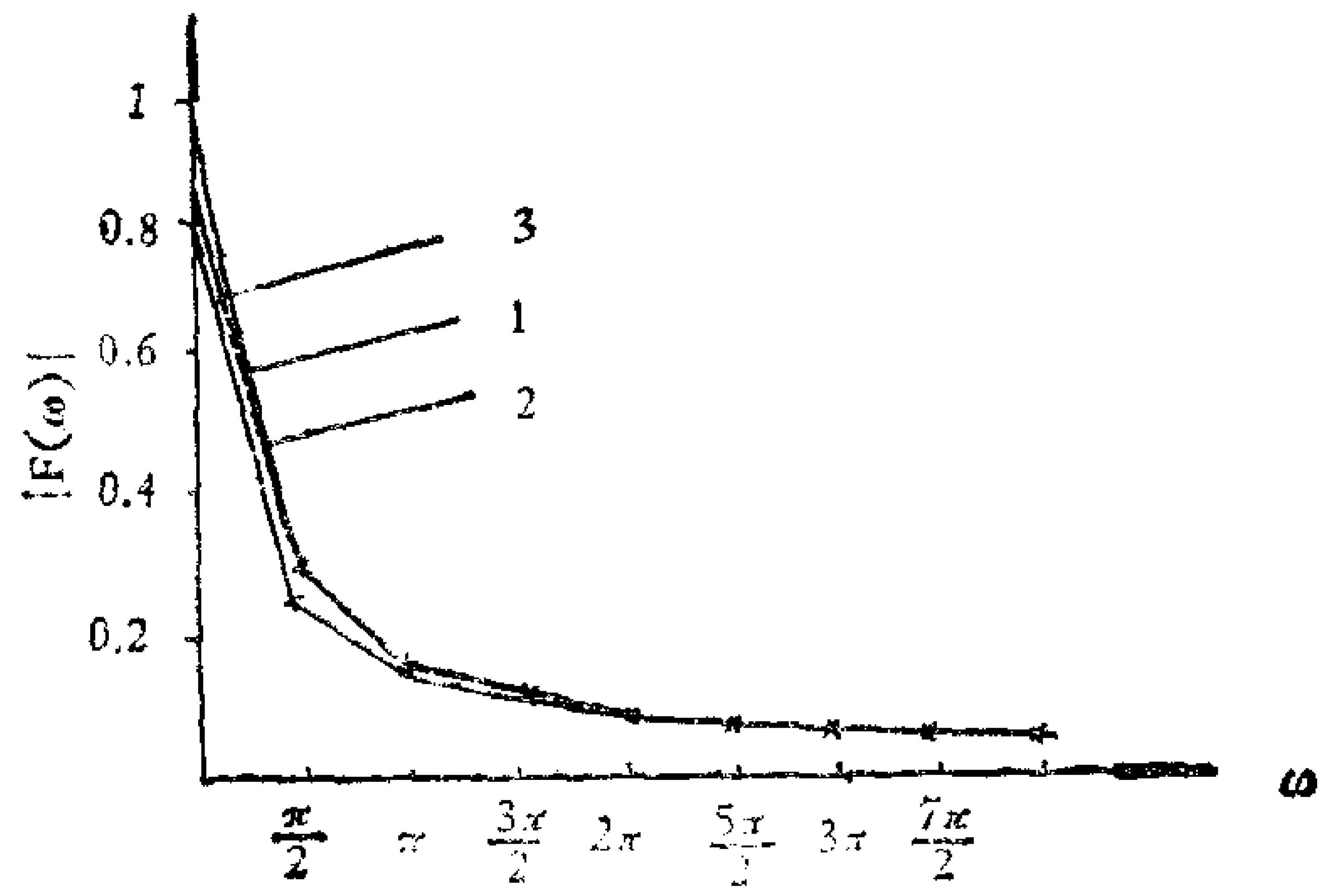

图 4

二、结　　论

理论推导和实际计算结果证明，随着数据的增加，（N·h），RFFT变换结果更接近实际频谱。在相同数据下和FFT是一致的，因而RFFT对实际谱的收敛问题等价于FFT对实际谱的收敛问题。由于数据量和递推次数不受限制，可对平稳随机过程的样本函数在足够长时间内进行变换，获得功率谱密度函数。RFFT总运算次数为$hA\log_2 N$，但对缓变信号(例如控制系统)，（$h-1$）$A\log_2 N$次运算可在递推中利用采样间隔完成。对时变谱分析、自适应控制及内存和处理能力有限的微计算机，RFFT是一种有效而且实用的方法。

感谢涂序彦老师审阅了此算法并提出了宝贵意见。

参 考 文 献

〔1〕 Ahmed, N., Rao, K.R., Orthogonal Transforms for Digital Signal Processing, New York, 1975.
〔2〕 孙仲康，快速傅立叶变换及其应用，人民邮电出版社，1982，

RECURSIVE FAST FOURIER TRANSFORMATION (RFFT)

Zhu Feihong
(Beijing Institute of Iron and Steel)

Abstract

A new recursive algorithm for Fast Fourier Transformation is proposed, which is derived and proven mathematically, Because of the arbitrariness in choosing the sections of input sequence and the times of recursion, this method is more flexible and leads to more precise results when applied to practical problems. The algorithm is easy to implement and can be extended to multi-dimensional cases.