第 3 卷 第 3 期 光学 精密工程 Vol. 3, No. 3
1995 年 6 月 OPTICS AND PRECISION ENGINEERING June, 1995

时域抽取基 2 快速傅里叶变换(FFT)算法

庞景先 温 坚* 赵明晶*
(长春大学,长春 130022)
(*长春大学机械工程学院,长春 130022)

摘要 本文阐述了时域抽取基 2 快速傅里叶变换算法。内容包括公式推导、蝶与蝶块的计算方法、快速傅里叶变换算法的信号流程图与计算机框图以及计算实例。

关键词:快速傅里叶变换;蝶;对偶结点;时域抽取

1 引 言

数学变换往往使问题的分析求解得到简化,傅里叶变换就是利用这种变换分析技巧,广泛用于科学研究的许多领域之中,成为现代科学分析的一个重要工具。傅里叶变换的一些典型应用领域是频谱分析、线性系统与随机过程的分析、随机变量的特征函数与偏微分方程以求解以及天线方向图与量子理论中测不准原理的确定等。

傅里叶变换的定义可表述如下:

$$H(f)=\int_{-\infty}^{\infty}h(t)e^{-j2\pi ft}dt \tag{1}$$

逆变换

$$h(t)=\int_{-\infty}^{\infty}H(f)e^{j2\pi ft}df \tag{2}$$

其实质就是时间函数在频域上的表示,也就是把一个时域波形分解成许多不同频率的正弦波之和。由于傅里叶变换的应用领域广泛,因此,很自然地希望把傅里叶变换能用数字电子计算机来计算。然而,连续傅里叶变换对数字电子计算机来说是无法接受的,因而必须运用数字分析技术,对连续傅里叶变换的时间函数或频率函数进行抽样和截断处理,使之成为离散傅里叶变换(DFT),以便数字电子计算机能够接受。

离散傅里叶变换的数学表达式为:

$$X_n=\sum_{k=0}^{N-1}x_ke^{-j2\pi nk/N} \tag{3}$$

逆变换

收稿日期:1995 年 3 月 10 日

$$x_k = \frac{1}{N}\sum_{n=0}^{N-1} X_n e^{j2\pi nk/N} \tag{4}$$

由(3)式可以看出，它表征 N 个方程的计算，需要 N^2 次复数乘法和 $N(N-1)$ 次复数加法计算。显然，计算的次数太繁多了，即便是使用计算速度惊人的电子计算机来计算，当 N 的数值很大时也会发现，花费的时间是太多了。为此，1965年美国 *IBM* 研究中心的 *Cooley-Tukey* 提出了快速傅里叶变换算法，给离散傅里叶变换提供了快速的计算方法。

快速傅里叶变换有许多算法，如时域抽取和频域抽取的基2、基4、基8、基16等算法。其基数越高，总运算量愈少，但基数越增加，算法变得越复杂，其中基2算法最为简单实用。基于此，本文将阐述时域抽取基2快速傅里叶变换算法。

2 公式推导

离散傅里叶变换的另一种形式为：

$$X_n = \frac{1}{N}\sum_{k=0}^{N-1} x_k e^{-j2\pi nk/N} \tag{5}$$

逆变换

$$x_k = \sum_{n=0}^{N-1} X_n e^{j2\pi nk/N} \tag{6}$$

离散傅里叶变换的复函数形式为：

$$X_n = R_n + jI_n \tag{7}$$

$$A_n = \sqrt{R_n^2 + I_n^2} \tag{8}$$

$$\varphi_n = \mathrm{arctg}\,\frac{I_n}{R_n} \tag{9}$$

式中 X_n:频率函数;x_k:时间函数;A_n:时间函数的振幅谱或傅里叶频谱;φ_n:傅里叶变换的相角;R_n:傅里叶变换的实部;I_n:傅里叶变换的虚部;N:抽样点数或序列项数;j:虚数单位，$j=\sqrt{-1}$。

设$[x_k]$为 N 项数据或 N 个抽样构成的时间序列，且 $N=2^M$(M 为正整数)，则其离散傅里叶变换可按(5)式重写为：

$$X_n = \frac{1}{N}\sum_{k=0}^{N-1} x_k e^{-j2\pi nk/N}$$

$$n = 0,1,\cdots\cdots,N-1$$

令 $W=e^{-j2\pi/N}$

则

$$X_n = \frac{1}{N}\sum_{k=0}^{N-1} x_k W^{nk} \tag{10}$$

$n = 0,1,\cdots\cdots,N-1$ 现将$[x_k]$序列按下标 k 的偶奇项分解成两个子序列$[y_k]$和$[z_k]$，则：

$$\left.\begin{array}{l} y_k=x_{2k} \\ z_k=x_{2k+1} \\ k=0,1,\cdots\cdots,\dfrac{N}{2}-1 \end{array}\right\} \tag{11}$$

根据(10)式，y_k 和 z_k 的离散傅里叶变换可写成

$$\left.\begin{aligned}&Y_n=\frac{2}{N}\sum_{k=0}^{\frac{N}{2}-1}y_kW^{2nk}\\&Z_n=\frac{2}{N}\sum_{n=0}^{\frac{N}{2}-1}z_kW^{2nk}n=0,1,\cdots\cdots,\frac{N}{2}-1\\&n=0,1,\cdots\cdots,\frac{N}{2}-1\end{aligned}\right\}\tag{12}$$

由于$[x_k]=[x_0,0,x_2,0,\cdots\cdots\cdots,X_{N-2}]+[0,x_1,0,x_3,\cdots\cdots\cdots,X_{N-1}]$

所以(1)式的离散傅里叶变换可写成：

$$\begin{aligned}X_n&=\frac{1}{N}\sum_{k=0}^{N-1}x_kW^{nk}\\&=\frac{2}{N}\sum_{k=0}^{\frac{N}{2}-1}x_{2k}W^{n(2k)}+\frac{2}{N}\sum_{k=0}^{\frac{N}{2}-1}x_{2k+1}W^{n(2k+1)}\\&=\frac{2}{N}\sum_{k=0}^{\frac{N}{2}-1}y_kW^{2nk}+W^n\frac{2}{N}\sum_{k=0}^{\frac{N}{2}-1}z_kW^{2nk}\end{aligned}\tag{13}$$

$$n=0,1,\cdots\cdots\cdots,N-1$$

将(12)式代入(13)式则得：

$$X_n=Y_n+W^nZ_n\tag{14}$$

$$n=0,1,\cdots\cdots\cdots,\frac{N}{2}-1$$

必须说明，$[X_n]$对 $0\leqslant n\leqslant N-1$ 均有定义，而$[Y_n]$与$[Z_n]$只对对 $0\leqslant n\leqslant\frac{N}{2}-1$ 有定义，所以，(14)式只表示$[X_n]$的前半序列，而后半序列，则当 $n=\frac{N}{2},\frac{N}{2}+1,\cdots\cdots,N-1$ 时，则$[Y_n]$和$[Z_n]$都是以$\frac{N}{2}$为周期的，即 $Y_{n+\frac{N}{2}}=Y_n,Z_{n+\frac{N}{2}}=Z_n$，所以$[X_n]$的后半序列应写成：

$$X_{n+\frac{N}{2}}=Y_{n+\frac{N}{2}}+W^{n+\frac{N}{2}}\cdot Z_{n+\frac{N}{2}}$$

根据尤拉公式：

$$e^{-j\varphi}=\cos\varphi-j\sin\varphi$$

$$e^{-j\pi}=\cos\pi-j\sin\pi=-1$$

$$W^{\frac{N}{2}}=e^{\frac{-j2\pi}{N}\cdot\frac{N}{2}}=-1$$

$$W^{n+\frac{N}{2}}=W^n\cdot W^{\frac{N}{2}}=-W^n$$

代入得

$$X_{n+\frac{N}{2}}=Y_n-W^nZ_n\tag{15}$$

$$n=0,1,\cdots\cdots\cdots,\frac{N}{2}-1$$

以上(14)、(15)两式即为基2快速傅里叶变换算法的计算公式，显然，其后项相同，只差一个正负号，故可使复数乘法的次数减少一半。

3　计算方法

(14)与(15)式的FFT算法可用图1所示的蝶图表示。图中左列结点代表子序列〔Y_n〕、〔Z_n〕的量值；右列结点代表合成序列〔X_n〕的量值，每个结点均有两个箭头进入，每个简头表示上列结点的量值与W^n相乘之积，当没有W^n时，则意味着$W^n=1$，两个箭头各自所代表的乘积之和即代表该结点的量值，即合成序列〔X_n〕的量值。图中X_n、$X_{n+\frac{N}{2}}$两结点称为对偶结点，即相隔$\frac{N}{2}$项的结点。这里需要说明，只有对偶结点才能形成蝶，只有在同一蝶上的结点才能运用两个*DFT*子序列合成的*FFT*(14)、(15)。

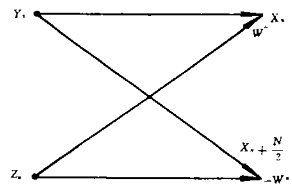

图1　蝶的计算方法

对于两个*DFT*子序列合成的*FFT*算法，实际上需要若干个蝶组成的蝶块来计算。图2给出了N=8的两个DFT子序列合成的蝶块计算方法，其计算过程如下：

$$X_0=Y_0+W^0Z_0$$
$$X_1=Y_1+W^1Z_1$$
$$X_2=Y_2+W^2Z_2$$
$$X_3=Y_3+W^3Z_3$$
$$X_4=Y_0-W^0Z_0$$
$$X_5=Y_1-W^1Z_1$$
$$X_6=Y_2-W^2Z_2$$
$$X_7=Y_3-W^3Z_3$$

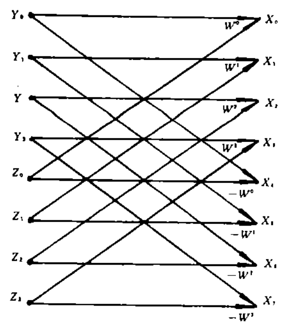

图2　$N=8$的蝶块计算方法

必须注意，在运用公式(14)、(15)进行FFT合成计算和确定W^n时，式中的N是代表每个蝶块的序列项数，N值不同，W值也不同。为计算方便，列出以下各式：

$$N=1\qquad W=e^{-j2\pi/1}=1$$
$$N=2\qquad W=e^{-j2\pi/2}=-1$$
$$N=3\qquad W=e^{-j2\pi/4}=-j$$
$$N=8\qquad W=e^{-j2\pi/8}=\cos\frac{\pi}{4}-j\sin\frac{\pi}{4}$$
$$N=N\qquad W=e^{-j2\pi/N}=\cos\frac{2\pi}{N}-j\sin\frac{2\pi}{N}$$

$$W^0=1$$
$$W^{\frac{N}{2}}=-1$$
$$W^N=1$$
$$W^{N+\frac{N}{2}}=-W^n$$

$$W^{2N}=1$$

$$W^{nk}=W^{nk\bmod(N)}\quad(\text{用}\,N\,\text{去除}\,nk\,\text{的余数})$$

图 3 给出了 $N=8$ 的 FFT 算法全过程的信号流程图。其计算步骤如下：

(1)将具有 N 项数据的序列$[x_k]$($N=8$)按偶奇项分为两个子序列，每个子序列具有 $N/2$ 项数据($\frac{N}{2}=4$)。

(2)将每个子序列再按偶奇项分为两个子序列，依此类推，直至每个子序列具有一项数据为止，则得一重新排列的具有 N 项数据的新序列($N=8$)。

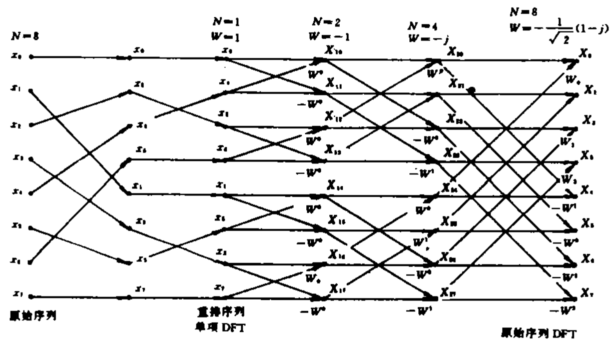

图 3　$N=8$ 的 FFT 算法信号流程图

(3)序列重排方法见表 1。

(4)将新序列的每个单项数据各自进行 DFT 计算，则得各单项数据的 DFT 序列，而各单项数据的 DFT 实际上就是各单项数据自身，因为单项数据 N=1，根据(5)式则有：

$$X_0=\frac{1}{N}\sum_{k=0}^{0}x_0e^{-j2\pi 00/1}=x_0$$

(5)利用公式(14)、(15)，将各单项 DFT 序列的相邻两项进行 FFT 合成计算，则得$\frac{N}{2}$个 DFT 子序列($\frac{N}{2}=4$)。

(6)再将相邻两个子序列继续进行 FFT 合成计算，则得$\frac{N}{4}$个 DFT 子序列($\frac{N}{4}=2$)，依此类推，直至得到 N/N 个 DFT 序列，则此序列即为原始序列的 DFT，其计算结果如下：

$$X_{10}=\frac{1}{N}(x_0+x_4)$$

$$X_{11}=\frac{1}{N}(x_0-x_4)$$

$$X_{12}=\frac{1}{N}(x_2+x_6)$$

$$X_{13}=\frac{1}{N}(x_2-x_6)$$

$$X_{14}=\frac{1}{N}(x_1+x_5)$$

$$X_{15}=\frac{1}{N}(x_1-x_5)$$

$$X_{16}=\frac{1}{N}(x_3+x_7)$$

$$X_{17}=\frac{1}{N}(x_3-x_7)$$

$$X_{20}=\frac{1}{N}[x_0+x_4+(x_2+x_6)]$$

$$X_{21}=\frac{1}{N}[x_0-x_4-j(x_2-x_6)]$$

$$X_{22}=\frac{1}{N}[x_0+x_4-(x_2+x_6)]$$

$$X_{23}=\frac{1}{N}[x_0-x_4+j(x_2-x_6)]$$

$$X_{24}=\frac{1}{N}[x_1+x_5+(x_3+x_7)]$$

$$X_{25}=\frac{1}{N}[x_1-x_5-j(x_3-x_7)]$$

$$X_{26}=\frac{1}{N}[x_1+x_5-(x_3+x_7)]$$

$$X_{27}=\frac{1}{N}[x_1-x_5+j(x_3-x_7)]$$

$$\left.\begin{aligned}
X_0&=\frac{1}{N}[x_0+x_4+x_2+x_6+x_1+x_5+x_3+x_7]\\
X_1&=\frac{1}{N}[x_0-x_4-j(x_2-x_6)+(x_1-x_5-jx_3+jx_7)(\frac{1}{\sqrt{2}}-j\frac{1}{\sqrt{2}})]\\
X_2&=\frac{1}{N}[x_0+x_4-(x_2+x_6)+(x_1+x_5-x_3-x_7)(\frac{1}{\sqrt{2}}-j\frac{1}{\sqrt{2}})^2]\\
X_3&=\frac{1}{N}[x_0-x_4+j(x_2-x_6)+(x_1-x_5+jx_3-jx_7)(\frac{1}{\sqrt{2}}-j\frac{1}{\sqrt{2}})^3]\\
X_4&=\frac{1}{N}[x_0+x_4+x_2+x_6-x_1-x_5-x_3-x_7]\\
X_5&=\frac{1}{N}[x_0-x_4-j(x_2-x_6)-(x_1-x_5-jx_3+jx_7)(\frac{1}{\sqrt{2}}-j\frac{1}{\sqrt{2}})]\\
X_6&=\frac{1}{N}[x_0+x_4-(x_2+x_6)-(x_1+x_5-x_3-x_7)(\frac{1}{\sqrt{2}}-j\frac{1}{\sqrt{2}})^2]\\
X_7&=\frac{1}{N}[x_0-x_4+j(x_2-x_6)-(x_1-x_5+jx_3-jx_7)(\frac{1}{\sqrt{2}}-j\frac{1}{\sqrt{2}})^3]
\end{aligned}\right\}\qquad(16)$$

4 原始序列重排方法

(1)将原始序列下标按十进制排列(表1);
(2)将十进制数码改用二进制表示;
(3)将二进制数码左右对称翻转;

表1 原始序列的重排

序	内 容	进 制	原始序列下标							
1	写出下标K	十进制	0	1	2	3	4	5	6	7
2	改换进制	二进制	000	001	010	011	100	101	110	111
3	对称翻转	二进制	000	100	010	110	001	101	011	111
4	改回进制得下标J	十进制	0	4	2	6	1	5	3	7

(4)将翻转后的二进制数码再改用十进制数码表示，此数码即为重排序列的下标；
(5)二进制数码的位数应为 $M=\log N/\log 2$。

5 计算实例

有一序列$[x_k]=1,2,1,1,3,2,1,2$，求其DFT频谱A_n和相角φ_n。
将$[x_k]$序列的各项数值代入(16)式可得其DFT序列为：

$$
\begin{aligned}
X_0 &= 1.62500\\
X_1 &= -0.16161+0.08839j\\
X_2 &= 0.25000-0.12500j\\
X_3 &= -0.33838+0.08839j\\
X_4 &= -0.12500\\
X_5 &= -0.33838+0.08839j\\
X_6 &= 0.25000-0.12500j\\
X_7 &= -0.16161+0.08839j
\end{aligned}
$$

利用(8)、(9)式可得其DFT频谱A_n和相角φ_n

$$
\begin{aligned}
A_0 &= \sqrt{1.62500^2}=1.62500\\
A_1 &= \sqrt{(-0.16161)^2+(0.08839)^2}=0.18420\\
A_2 &= \sqrt{(0.25000)^2+(-0.12500)^2}=0.27951\\
A_3 &= \sqrt{(-0.33838)^2+(0.08839)^2}=0.34973\\
A_4 &= \sqrt{(-0.12500)^2}=0.12500\\
A_5 &= \sqrt{(-0.33838)^2+(-0.08839)^2}=0.34973\\
A_6 &= \sqrt{(0.25000)^2+(0.12500)^2}=0.27951\\
A_7 &= \sqrt{(0.16161)^2+(-0.08839)^2}=0.18420\\
\varphi_0 &= arctg\,\frac{0}{1.62500}=0^\circ\\
\varphi_1 &= arctg\,\frac{0.08839}{-0.16161}=151^\circ 19'28''
\end{aligned}
$$

$$\varphi_2=arctg\,\frac{-0.12500}{0.25000}=-26°33'54''$$

$$\varphi_3=arctg\,\frac{0.08839}{-0.33838}=165°21'44''$$

$$\varphi_4=arctg\,\frac{0}{-0.12500}=0°$$

$$\varphi_5=arctg\,\frac{-0.08839}{-0.33838}=-165°21'44''$$

$$\varphi_6=arctg\,\frac{0.12500}{0.25000}=26°33'54''$$

$$\varphi_7=arctg\,\frac{-0.08839}{-0.16161}=-151°19'28''$$

6　FFT 算法的计算机框图

FFT 算法的计算机框图如图 4 所示。图中符号的意义如下：

N——序列项数即抽样点数；

M——幂指数即二进制位数或 DFT 计算的列数；

K——原始序列下标即序列重排的循环变量，取值 $k=0\sim N-1$，步长 1，初值 $k=0$；

J——重排序列下标；

I——控制二进制位数的循环变量，取值 $I=1\sim M$，步长 1，初值 $I=1$；

K_1——计算重排序列下标的中间变量；

K_2——计算重排序列下标的中间变量；

T——序列重排换置的中间存贮单元，即计算每个蝶的后对偶结点 DFT 值的中间存贮单元；

L——列转移的循环变量（即列号计数器），取值 $L=1\sim M$，步长 1，初值 $L=1$；

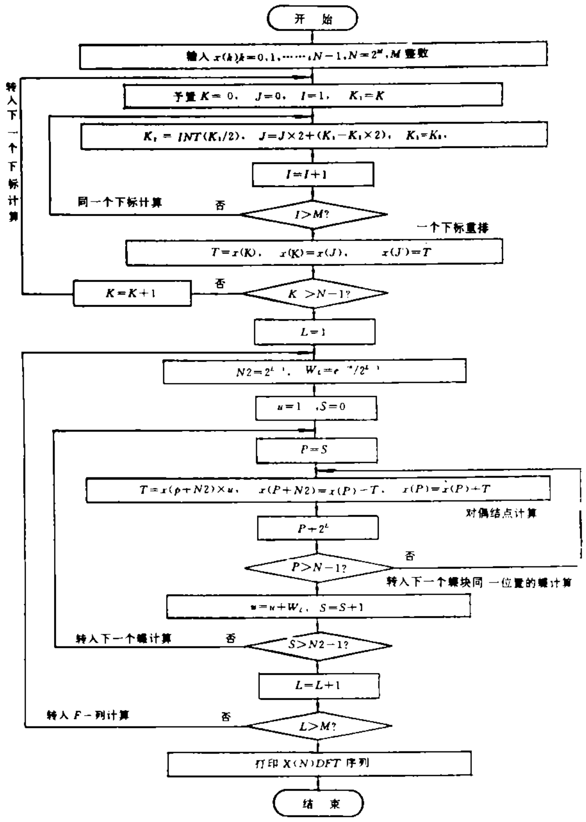

图 4　FFT 算法的计算机框图

U——各蝶块中的公共因子，初值 $u=w_L^0=1$；

W_L——第 L 列的公共因子，$W_L=e^{-j\pi/2^{L-1}}=\cos\frac{\pi}{2^{L-1}}-j\sin\frac{\pi}{2^{L-1}}$；

$N2$——第 L 列蝶块中蝶的个数，$N2=2^{L-1}$

S——蝶块中蝶转移的循环变量（即蝶的序号计数器），取值 $S=0\sim N2-1$，步长 1，第一个

蝶 $S=0$；

P——蝶块转移的循环变量，取值 $P=S\sim N-1$，步长 2^L；

$X(P)$——同一蝶中前对偶结点的 DFT 值；

$X(P+N2)$——同一蝶中后对偶结点的 DFT 值。

参 考 文 献

[1] E·O 布赖姆，快速傅里叶变换. 上海：上海科技出版社，1979

[2] 雷继尧，数字信号的处理原理及应用. 重庆：重庆大学出版社，1983

[3] BASIC subroutine runs a slow FFT. EDN，1978，**23**(18)

The $N=2^M$ FFT Algorithm for Decimation in Time

Pang Jingxian, Wen Jian and Zhao Mingjing

(*Changchun University, Changchun* 130022)

Abstract

This Paper describes the $N=2^M$ FFT algorithm for decimation in time. It contains the demonstration of formulas, the FFT butterfly operation, the signal flow chart and the computer block diagram of the FFT algorithm and the calculation example.

Key words:FFT, Butterfly, Pair node, Decimation in time.