用快速多项式变换(FPT)计算二维离散富里叶变换(DFT)

蒋 增 荣

提 要 本文利用快速多项式变换(FPT)计算$N\times M$型二维DFT($M=2^m$, $N=2^{m-r+1}$, $1\leqslant r\leqslant m$), 所需的乘法及加法次数（复乘及复加）分别为

$$M_u=\frac{1}{2}NM\log_2 M-\frac{3}{2}NM+N^2+N(1+\log_2 M-\log_2 N)$$

$$A_d=NM\log_2 NM,$$

与通常的以2为基的二维FFT比较，加法次数相同，乘法次数减少约30—40%，从而提高了计算精度。本算法还适用于并行算法。

一、引 言

H.J.Nussbaumer 和 P.Quandalle 在 [1] 中引入了多项式变换并用它计算数字卷积和离散富里叶变换[2],[3]。我们在[4]中详细的研究了模$M(z)$是可约多项式时变换存在的条件，并用它来计算过两个多项式的乘积[5]。本文研究用FPT计算二維DFT的方法，结果表明，当$N=2^{m-r+1}$, $M=2^m(1\leqslant r\leqslant m)$时，所用的加法次数与通常以2为基的二維FFT所需的加法次数相同，乘法次数减少了30—40%，并且本方法还适用于并行算法。

设

$$[x]=\begin{pmatrix} x_{00} & x_{01} & \cdots & x_{0,M-1} \\ x_{10} & x_{11} & & x_{1,M-1} \\ \vdots & \vdots & & \vdots \\ x_{N-1,0} & x_{N-1,1} & \cdots & x_{N-1,M-1} \end{pmatrix} \tag{1.1}$$

其二維DFT定义为

$$X_{k_1,k_2}=\sum_{n_1=0}^{N-1}\sum_{n_2=0}^{M-1}x_{n_1,n_2}W_N^{n_1k_1}\,W_M^{n_2k_2}, \quad \begin{pmatrix} k_1=0,1,\cdots,N-1 \\ k_2=0,1,\cdots,M-1 \end{pmatrix} \tag{1.2}$$

其中$W_N=e^{-i2\pi/N}$, $W_M=e^{-i2\pi/M}$。记$[X]$为

$$[X]=\begin{pmatrix} X_{00} & X_{01} & \cdots & X_{0,M-1} \\ X_{10} & X_{11} & & X_{1,M-1} \\ \vdots & \vdots & & \vdots \\ X_{N-1,0} & X_{N-1,1} & \cdots & X_{N-1,M-1} \end{pmatrix} \tag{1.3}$$

本文1981年12月收到

[2]中证明了如下结论：设

$$\overline{X}_{k_1}(z) \equiv \sum_{n_1=0}^{N-1} X_{n_1}(z) W_N^{n_1 k_1} \bmod (z^M - 1), (k_1 = 0, 1, \cdots, N-1) \tag{1.4}$$

则

$$X_{k_1,k_2} \equiv \overline{X}_{k_1}(z) \bmod (z - W_M^{k_2}), (k_1 = 0, 1, \cdots, N-1; k_2 = 0, 1, \cdots, M-1) \tag{1.5}$$

其中

$$X_{n_1}(z) = \sum_{n_2=0}^{N-1} x_{n_1,n_2} z^{n_2}, \quad (n_1 = 0, 1, \cdots, N-1) \tag{1.6}$$

是[x]的按行形成的母多项式序列。

本文利用上述结论，研究当 $N = 2^{m-r+1}$，$M = 2^m$（$1 \leqslant r \leqslant m$）时(1.2)式的计算。首先研究最常见也最重要的情况：$N = M = 2^m$.

二、N=M=2^m 的 情 况

[x]的二維 DFT 为

$$X_{k_1,k_2} = \sum_{n_1=0}^{N-1} \sum_{n_2=0}^{N-1} x_{n_1,n_2} W^{n_1 k_1 + n_2 k_2}, (k_1, k_2 = 0, 1, \cdots, N-1) \tag{2.1}$$

其中$W = e^{-i2\pi/N}$，$N = 2^m$. 这时有

$$X_{n_1}(z) = \sum_{n_2=0}^{N-1} x_{n_1,n_2} z^{n_2}, \quad (n_1 = 0, 1, \cdots, N-1) \tag{2.2}$$

$$\overline{X}_{k_1}(z) \equiv \sum_{n_1=0}^{N-1} X_{n_1}(z) W^{n_1 k_1} \bmod (z^N - 1), (k_1 = 0, 1, \cdots, N-1) \tag{2.3}$$

$$X_{k_1,k_2} \equiv \overline{X}_{k_1}(z) \bmod (z - W^{k_2}), \quad (k_1, k_2 = 0, 1, \cdots, N-1) \tag{2.4}$$

由于

$$z^N - 1 = \left(z^{\frac{N}{2}} + 1\right)\left(z^{\frac{N}{2}} - 1\right) \tag{2.5}$$

且$\left(z^{\frac{N}{2}} + 1\right)$和$\left(z^{\frac{N}{2}} - 1\right)$互素，因此如设

$$\overline{X}_{k_1}^{(1)}(z) \equiv \overline{X}_{k_1}(z) \bmod \left(z^{\frac{N}{2}} + 1\right)$$

$$\overline{X}_{k_1}^{(2)}(z) \equiv \overline{X}_{k_1}(z) \bmod \left(z^{\frac{N}{2}} - 1\right)$$

则由孙子定理有

$$\overline{X}_{k_1}(z) \equiv \frac{1}{2}\left[\left(1 - z^{\frac{N}{2}}\right)\overline{X}_{k_1}^{(1)}(z) + \left(1 + z^{\frac{N}{2}}\right)\overline{X}_{k_1}^{(2)}(z)\right] \bmod (z^N - 1) \tag{2.6}$$

由(2.4)式并注意到W是复数域中的N阶单位根，有

$$X_{k_1,k_2} \equiv \overline{X}_{k_1}^{(1)}(z) \bmod (z - W^{k_2}), (k_2 = 1, 3, \cdots, N-1) \tag{2.7}$$

$$(k_1 = 0, 1, \cdots, N-1)$$

$$X_{k_1,k_2} \equiv \overline{X}_{k_1}^{(2)}(z) \bmod (z - W^{k_2}), (k_2 = 0, 2, \cdots, N-2) \tag{2.8}$$

由$\overline{X}_{k_1}(z)$，$\overline{X}^{(1)}_{k_1}(z)$，$\overline{X}^{(2)}_{k_1}(z)$的定义有

$$\overline{X}^{(1)}_{k_1}(z)\equiv\sum_{n_1=0}^{N-1}X^{(1)}_{n_1}(z)W^{n_1k_1}\bmod\left(z^{\frac{N}{2}}+1\right),(k_1=0,1,\cdots,N-1) \tag{2.9}$$

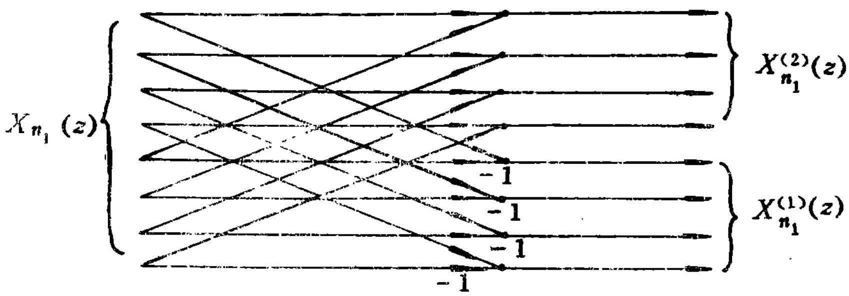
$$\overline{X}^{(2)}_{k_1}(z)\equiv\sum_{n_1=0}^{N-1}X^{(2)}_{n_1}(z)W^{n_1k_1}\bmod\left(z^{\frac{N}{2}}-1\right),(k_1=0,1,\cdots,N-1) \tag{2.10}$$

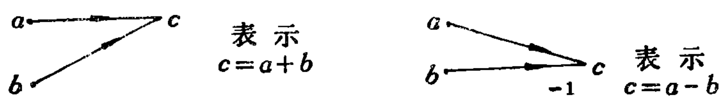
其中
$$X^{(1)}_{n_1}(z)\equiv X_{n_1}(z)\bmod\left(z^{\frac{N}{2}}+1\right) \tag{2.11}$$

$$(n_1=0,1,\cdots,N-1)$$

$$X^{(2)}_{n_1}(z)\equiv X_{n_1}(z)\bmod\left(z^{\frac{N}{2}}-1\right) \tag{2.12}$$

母多项式序列$\{X_{n_1}(z)\}$由给定数阵$[x]$确定，因此根据（2.11）和（2.12）式容易求得$\left\{X^{(1)}_{n_1}(z)\right\}$和$\left\{X^{(2)}_{n_1}(z)\right\}$. 注意到 $z^{\frac{N}{2}}\equiv-1\bmod\left(z^{\frac{N}{2}}+1\right)$和 $z^{\frac{N}{2}}\equiv1\bmod\left(z^{\frac{N}{2}}-1\right)$，还可以用FFT类型的快速算法计算。以$N=8$为例，这时求$\left\{X^{(1)}_{n_1}(z)\right\}$和$\left\{X^{(2)}_{n_1}(z)\right\}$的快速算法流程为





上图左边八个点是$X_{n_1}(z)$的八个系数，右边上面四个点是$X^{(2)}_{n_1}(z)$的四个系数，下面四个点是$X^{(1)}_{n_1}(z)$的四个系数。

由上图知，由$\{X_{n_1}(z)\}$计算出$\left\{X^{(i)}_{n_1}(z)\right\}(i=1,2)$ 共需

$$A_1=N^2 \tag{2.13}$$

次加法。

下面研究(2.9)式的计算。显然有

$$\overline{X}^{(1)}_{\langle k_1k_2\rangle}(z)\equiv\sum_{n_1=0}^{N-1}X^{(1)}_{n_1}(z)W^{n_1k_1k_2}\bmod\left(z^{\frac{N}{2}}+1\right)$$

由(2.7)式有

$$X_{\langle k_1k_2\rangle,k_2}\equiv\sum_{n_1=0}^{N-1}X_{n_1}^{(1)}(z)W^{n_1k_1k_2}\equiv\sum_{n_1=0}^{N-1}X_{n_1}^{(1)}(z)z^{n_1k_1}\ \mathrm{mod}\left(z^{\frac{N}{2}}+1\right),\left(z-W^{k_2}\right)$$

为方便，仍记

$$\bar{X}_{\langle k_1k_2\rangle}^{(1)}(z)\equiv\sum_{n_1=0}^{N-1}X_{n_1}^{(1)}(z)z^{n_1k_1}\ \mathrm{mod}\left(z^{\frac{N}{2}}+1\right),\qquad(2.14)$$

$$(k_1=0,1,\cdots,N-1;k_2=1,3,\cdots,N-1)$$

于是有

$$X_{\langle k_1k_2\rangle,k_2}\equiv\bar{X}_{\langle k_1k_2\rangle}^{(1)}(z)\mathrm{mod}(z-W^{k_2}),\qquad(2.15)$$

$$(k_1=0,1,\cdots,N-1;k_2=1,3,\cdots,N-1)$$

(2.14) 及 (2.15) 式中的$\langle k_1k_2\rangle=k_1k_2\mathrm{mod}N$，由于$k_2=1,3,\cdots,N-1$，$N=2^m$，故$(k_2,N)=1$，所以$\langle k_1k_2\rangle$是$0,1,2,\cdots,N-1$的一个排列 $(k_1=0,1,\cdots,N-1)$，从而(2.15)式左边是$[X]$的k_2列的全部元素。不难看出[4]，(2.14)式是以不可约多项式$z^{\frac{N}{2}}+1$为模、z为根、长度为N的多项式变换。由于$N=2^m$，(2.14)式可用FFT类型的快速算法（称为快速多项式变换—FPT）计算。注意到(2.14)式右端与k_2无关，所以计算出$\bar{X}_{\langle k_1k_2\rangle}^{(1)}(z)$共需要$N\log_2N$对次数为$\frac{N}{2}-1$的多项式加法，全部算出$\bar{X}_{\langle k_1k_2\rangle}^{(1)}(z)$共需

$$A_2=\frac{N}{2}\cdot N\log_2N=\frac{1}{2}N^2\log_2N\qquad(2.16)$$

次加法(不需乘法)。由于模是$z^{\frac{N}{2}}+1$，故有

$$\bar{X}_{\langle k_1k_2\rangle}^{(1)}(z)=\sum_{l=0}^{\frac{N}{2}-1}y_{k_1,l}^{(1)}z^l,\ (k_1=0,1,\cdots,N-1;k_2=1,3,\cdots,N-1)\qquad(2.17)$$

于是由(2.15)式便有

$$X_{\langle k_1k_2\rangle,k_2}=\sum_{l=0}^{\frac{N}{2}-1}y_{k_1,l}^{(1)}W^{k_2l},\ (k_1=0,1,\cdots,N-1;k_2=1,3,\cdots,N-1)\qquad(2.18)$$

(2.18)式可看作$\frac{N}{2}$点的简化DFT，它可用FFT类型的快速算法计算，为说明起见，以$N=16$为例说明，这时(2.18)式的矩阵表示为

$$\begin{pmatrix}X_{\langle k_1\rangle,1}\\X_{\langle 3k_1\rangle,3}\\X_{\langle 5k_1\rangle,5}\\X_{\langle 7k_1\rangle,7}\\\hdashline X_{\langle 9k_1\rangle,9}\\X_{\langle 11k_1\rangle,11}\\X_{\langle 13k_1\rangle,13}\\X_{\langle 15k_1\rangle,15}\end{pmatrix}=\left(\begin{array}{cccc:cccc}1&W&W^2&W^3&W^4&W^5&W^6&W^7\\1&W^3&W^6&-W&-W^4&-W^7&W^2&W^5\\1&W^5&-W^2&-W^7&W^4&-W&-W^6&W^3\\1&W^7&-W^6&W^5&-W^4&W^3&-W^2&W\\\hdashline 1&-W&W^2&-W^3&W^4&-W^5&W^6&-W^7\\1&-W^3&W^6&W&-W^4&W^7&W^2&-W^5\\1&-W^5&-W^2&W^7&W^4&W&-W^6&-W^3\\1&-W^7&-W^6&-W^5&-W^4&-W^3&-W^2&-W\end{array}\right)\begin{pmatrix}y_{k_1,0}^{(1)}\\y_{k_1,1}^{(1)}\\y_{k_1,2}^{(1)}\\y_{k_1,3}^{(1)}\\\hdashline y_{k_1,4}^{(1)}\\y_{k_1,5}^{(1)}\\y_{k_1,6}^{(1)}\\y_{k_1,7}^{(1)}\end{pmatrix}$$

$$(W=e^{-i2\pi/16})$$

将第0，1, 2, 3, 4, 5, 6, 7行按其二进制逆序0，4, 2, 6, 1, 5, 3, 7排列，并将所得矩阵分解，就有

$$
\begin{pmatrix}
X_{<k_1>,1}\\ X_{<9k_1>,9}\\ X_{<5k_1>,5}\\ X_{<13k_1>,13}\\ \hdashline X_{<3k_1>,3}\\ X_{<11k_1>,11}\\ X_{<7k_1>,7}\\ X_{<15k_1>,15}
\end{pmatrix}
=
\left(\begin{array}{cccc:cccc}
1 & W & W^2 & W^3 & W^4 & W^5 & W^6 & W^7\\
1 & -W & W^2 & -W^3 & W^4 & -W^5 & W^6 & -W^7\\
1 & W^5 & -W^2 & -W^7 & W^4 & -W & -W^6 & W^3\\
1 & -W^5 & -W^2 & W^7 & W^4 & W & -W^6 & -W^3\\ \hdashline
1 & W^3 & W^6 & -W & -W^4 & -W^7 & W^2 & W^5\\
1 & -W^3 & W^6 & W & -W^4 & W^7 & W^2 & -W^5\\
1 & W^7 & -W^6 & W^5 & -W^4 & W^3 & -W^2 & W\\
1 & -W^7 & -W^6 & -W^5 & -W^4 & -W^3 & -W^2 & -W
\end{array}\right)
\begin{pmatrix}
y^{(1)}_{k_1,0}\\ y^{(1)}_{k_1,1}\\ y^{(1)}_{k_1,2}\\ y^{(1)}_{k_1,3}\\ \hdashline y^{(1)}_{k_1,4}\\ y^{(1)}_{k_1,5}\\ y^{(1)}_{k_1,6}\\ y^{(1)}_{k_1,7}
\end{pmatrix}
$$

$$
=\left(\begin{array}{cccc:cccc}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1\\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1\\
1 & W^4 & -1 & -W^4 & 1 & W^4 & -1 & -W^4\\
1 & -W^4 & -1 & W^4 & 1 & -W^4 & -1 & W^4\\ \hdashline
1 & W^2 & W^4 & W^6 & -1 & -W^2 & -W^4 & -W^6\\
1 & -W^2 & W^4 & -W^6 & -1 & W^2 & -W^4 & W^6\\
1 & W^6 & -W^4 & W^2 & -1 & -W^6 & W^4 & -W^2\\
1 & -W^6 & -W^4 & -W^2 & -1 & W^6 & W^4 & W^2
\end{array}\right)
\left(\begin{array}{cccc:cccc}
1 &&&&&&&\\
& W &&&&&&\\
&& W^2 &&&&&\\
&&& W^3 &&&&\\ \hdashline
&&&& W^4 &&&\\
&&&&& W^5 &&\\
&&&&&& W^6 &\\
&&&&&&& W^7
\end{array}\right)
\begin{pmatrix}
y^{(1)}_{k_1,0}\\ y^{(1)}_{k_1,1}\\ y^{(1)}_{k_1,2}\\ y^{(1)}_{k_1,3}\\ \hdashline y^{(1)}_{k_1,4}\\ y^{(1)}_{k_1,5}\\ y^{(1)}_{k_1,6}\\ y^{(1)}_{k_1,7}
\end{pmatrix}
$$

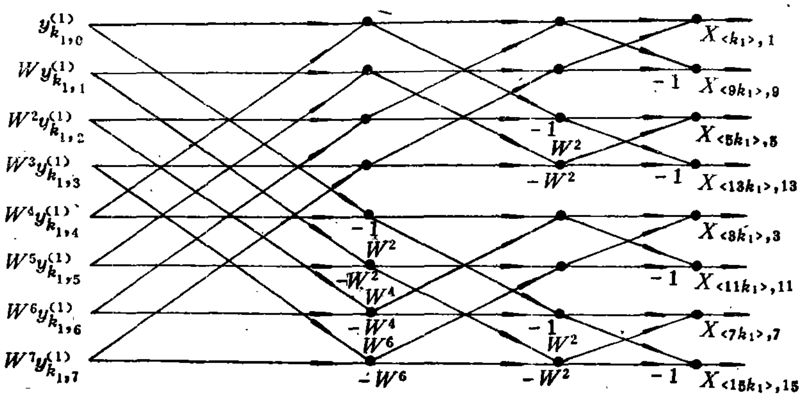
$$
=\left(\begin{array}{cccc:cccc}
1 & 1 & 1 & 1 &&&&\\
1 & -1 & 1 & -1 &&&&\\
1 & W^4 & -1 & -W^4 &&&&\\
1 & -W^4 & -1 & W^4 &&&&\\ \hdashline
&&&& 1 & 1 & 1 & 1\\
&&&& 1 & -1 & 1 & -1\\
&&&& 1 & W^4 & -1 & -W^4\\
&&&& 1 & -W^4 & -1 & W^4
\end{array}\right)
\left(\begin{array}{cccc:cccc}
1 &&&& 1 &&&\\
& 1 &&&& 1 &&\\
&& 1 &&&& 1 &\\
&&& 1 &&&& 1\\ \hdashline
1 &&&& -1 &&&\\
& W^2 &&&& -W^2 &&\\
&& W^4 &&&& -W^4 &\\
&&& W^6 &&&& -W^6
\end{array}\right)
$$

$$
\cdot\left(\begin{array}{cccc:cccc}
1 &&&&&&& \\
& W &&&&&& \\
&& W^{2} &&&&& \\
&&& W^{3} &&&& \\
\hdashline
&&&& W^{4} &&& \\
&&&&& W^{5} && \\
&&&&&& W^{6} & \\
&&&&&&& W^{7}
\end{array}\right)
\left(\begin{array}{c}
y_{k_1,0}^{(1)} \\ y_{k_1,1}^{(1)} \\ y_{k_1,2}^{(1)} \\ y_{k_1,3}^{(1)} \\ \hdashline y_{k_1,4}^{(1)} \\ y_{k_1,5}^{(1)} \\ y_{k_1,6}^{(1)} \\ y_{k_1,7}^{(1)}
\end{array}\right)
$$

$$
=\left(\begin{array}{cccc:cccc}
1 & 1 &&&&&& \\
1 & -1 &&&&&& \\
&& 1 & 1 &&&& \\
&& 1 & -1 &&&& \\
\hdashline
&&&& 1 & 1 && \\
&&&& 1 & -1 && \\
&&&&&& 1 & 1 \\
&&&&&& 1 & -1
\end{array}\right)
\left(\begin{array}{cccc:cccc}
1 && 1 &&&&& \\
& 1 && 1 &&&& \\
1 && -1 &&&&& \\
& W^{2} && -W^{2} &&&& \\
\hdashline
&&&& 1 && 1 & \\
&&&&& 1 && 1 \\
&&&& 1 && -1 & \\
&&&&& W^{2} && -W^{2}
\end{array}\right)
$$

$$
\cdot\left(\begin{array}{cccc:cccc}
1 &&&& 1 &&& \\
& 1 &&&& 1 && \\
&& 1 &&&& 1 & \\
&&& 1 &&&& 1 \\
\hdashline
1 &&&& -1 &&& \\
& W^{2} &&&& -W^{2} && \\
&& W^{4} &&&& -W^{4} & \\
&&& W^{6} &&&& -W^{6}
\end{array}\right)
\left(\begin{array}{cccc:cccc}
1 &&&&&&& \\
& W &&&&&& \\
&& W^{2} &&&&& \\
&&& W^{3} &&&& \\
\hdashline
&&&& W^{4} &&& \\
&&&&& W^{5} && \\
&&&&&& W^{6} & \\
&&&&&&& W^{7}
\end{array}\right)
\left(\begin{array}{c}
y_{k_1,0}^{(1)} \\ y_{k_1,1}^{(1)} \\ y_{k_1,2}^{(1)} \\ y_{k_1,3}^{(1)} \\ \hdashline y_{k_1,4}^{(1)} \\ y_{k_1,5}^{(1)} \\ y_{k_1,6}^{(1)} \\ y_{k_1,7}^{(1)}
\end{array}\right)
$$

这样就得到 $N=16$ 时(2.18)式的快速算法，其流程图为：



由于 $W^2=W_8$，故由上图不难看出，只需以

$$\left(y^{(1)}_{k_1,0}, Wy^{(1)}_{k_1,1}, W^2y^{(1)}_{k_1,2}, W^3y^{(1)}_{k_1,3}, W^4y^{(1)}_{k_1,4}, W^5y^{(1)}_{k_1,5}, W^6y^{(1)}_{k_1,6}, W^7y^{(1)}_{k_1,7}\right)^T$$

作为输入序列，用八点的 FFT 便得 $N=16$ 时(2.18)式的快速算法。

对于 $N=2^m$，(2.18)式可像上例那样进行快速计算，这只需将下列$\dfrac{N}{2}$个点

$$\left(y^{(1)}_{k_1,0}, Wy^{(1)}_{k_1,1}, W^2y^{(1)}_{k_1,2}, \cdots\cdots, W^{\frac{N}{2}-1}y^{(1)}_{k_1,\frac{N}{2}-1}\right)^T \tag{2.19}$$

作为输入序列，用$\dfrac{N}{2}=2^{m-1}$点的 FFT 算法便得到(2.18)式的快速算法。以 2 为基的 N 点 FFT 所需的复乘和复加次数分别为

$$\left.\begin{aligned} M_0 &= \frac{N}{2}\log_2 N-\frac{3}{2}N+2 \\ A_0 &= N\log_2 N \end{aligned}\right\} \tag{2.20}$$

因此用上述快速算法计算 (2.18) 式中的 $X_{\langle k_1k_2\rangle,k_2}$ ($k_1=0,1,\cdots,N-1;k_2=1,3,\cdots,N-1$) 所需的复乘和复加次数分别为

$$M_1=N\left(\frac{N}{4}\log_2\frac{N}{2}-\frac{3}{4}N+2\right)+N\left(\frac{N}{2}-1\right)=\frac{1}{4}N^2\log_2 N-\frac{1}{2}N^2+N \tag{2.21}$$

$$A_3=N\cdot\frac{1}{2}N\log_2\frac{N}{2}=\frac{1}{2}N^2\log_2 N-\frac{1}{2}N^2. \tag{2.22}$$

下面再研究(2.10)式的计算。作变换

$$z=Wu, W=e^{-i2\pi/N} \tag{2.23}$$

并记

$$\widetilde{X}_{n_1}(u)=X_{n_1}(Wu), \widetilde{X}^{(2)}_{n_1}(u)=X^{(2)}_{n_1}(Wu), \widetilde{\overline{X}}{}^{(2)}_{k_1}(u)=\overline{X}^{(2)}_{k_1}(Wu) \tag{2.24}$$

(2.10)式就成为

$$\widetilde{\widetilde{X}}_{k_1}^{(2)}(u)\equiv\sum_{n_1=0}^{N-1}\widetilde{X}_{n_1}^{(2)}(u)W^{n_1k_1}\ \mathrm{mod}(u^{\frac{N}{2}}+1),(k_1=0,1,\cdots,N-1)\tag{2.25}$$

$$\widetilde{X}_{n_1}^{(2)}(u)\equiv\widetilde{X}_{n_1}(u)\mathrm{mod}(u^{\frac{N}{2}}+1).$$

再由 $z\equiv W^{k_2}\mathrm{mod}(z-W^{k_2})$ 就有

$$u\equiv W^{k_2-1}\mathrm{mod}(u-W^{k_2-1})\tag{2.26}$$

这样就有

$$\widetilde{\widetilde{X}}_{\langle k_1(k_2-1)\rangle}^{(2)}(u)\equiv\sum_{n_1=0}^{N-1}\widetilde{X}_{n_1}^{(2)}(u)u^{n_1k_1}\ \mathrm{mod}(u^{\frac{N}{2}}+1)\tag{2.27}$$

$$(k_1=0,1,\cdots,N-1;k_2=0,2,\cdots,N-2)$$

$$X_{\langle k_1(k_2-1)\rangle,k_2}\equiv\widetilde{\widetilde{X}}_{\langle k_1(k_2-1)\rangle}^{(2)}(u)\mathrm{mod}(u-W^{k_2-1})\tag{2.28}$$

$$(k_1=0,1,\cdots,N-1;k_2=0,2,\cdots,N-2)$$

上两式中 $\langle k_1(k_2-1)\rangle=\langle k_1(k_2-1)\rangle\mathrm{mod}N$。当 $k_1=0,1,\cdots,N-1,k_2=0,2,\cdots,N-2$ 时，$\langle k_1(k_2-1)\rangle$ 也是 $0,1,\cdots,N-1$ 的一个排列。(2.27)式是以不可约多项式 $u^{\frac{N}{2}}+1$ 为模、u 为根、长度为N的多项式变换，由于 $N=2^m$，故可用FPT计算，所需的加法次数如(2.16)式所示。由于模是 $u^{\frac{N}{2}}+1$，故有

$$\widetilde{\widetilde{X}}_{\langle k_1(k_2-1)\rangle}^{(2)}(u)=\sum_{l=0}^{\frac{N}{2}-1}y_{k_1,l}^{(2)}u^l,(k_1=0,1,\cdots,N;k_2=0,2,\cdots,N-2)\tag{2.29}$$

于是由(2.28)式有

$$X_{\langle k_1(k_2-1)\rangle,k_2}=\sum_{l=0}^{\frac{N}{2}-1}y_{k_1,l}^{(2)}W^{l(k_2-1)},(k_1=0,1,\cdots,N-1;k_2=0,2,\cdots,N-2)\tag{2.30}$$

如果注意到$W^{-1}=W^{N-1}$，那么(2.30)式就可看作$\frac{N}{2}$点的简化DFT，如以如下$\frac{N}{2}$点序列

$$\boldsymbol{y}_{k_1}^{(2)}=(y_{k_1,0}^{(2)},W^{N-1}y_{k_1,1}^{(2)},W^{N-2}y_{k_1,2}^{(2)},\cdots,W^{\frac{N}{2}+1}y_{k_1,\frac{N}{2}-1}^{(2)})^T\tag{2.31}$$

作为输入序列，那么(2.30)式的矩阵表示式就为

$$\boldsymbol{X}_{k_1}=[W_{\frac{N}{2}}]\,\boldsymbol{y}_{k_1}^{(2)}\tag{2.32}$$

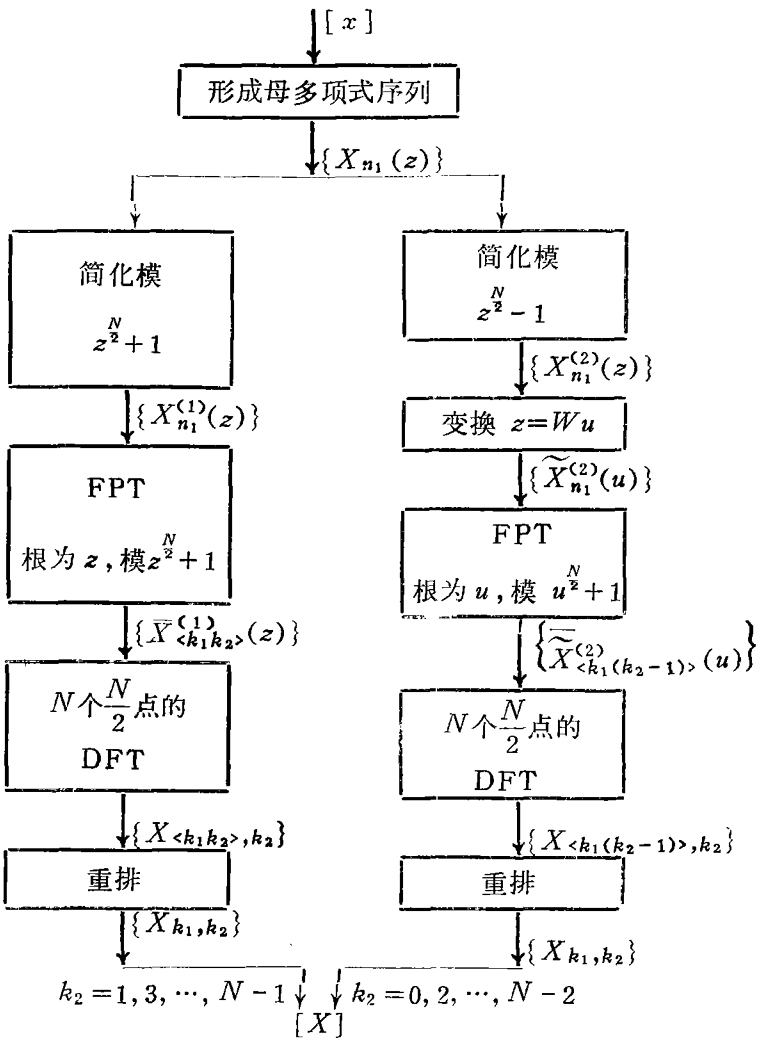
其中

$$\boldsymbol{X}_{k_1}=(X_{\langle-k_1\rangle,0},X_{\langle k_1\rangle,2},X_{\langle 3k_1\rangle,4},\cdots,X_{\langle k_1(N-3)\rangle,N-2})^T\tag{2.33}$$

$$\left[W_{\frac{N}{2}}\right]=\begin{pmatrix}1&1&\cdots&1\\1&W^2&\cdots&W^{2(\frac{N}{2}-1)}\\\vdots&\vdots&&\vdots\\1&W^{2(\frac{N}{2}-1)}&\cdots&W^{2(\frac{N}{2}-1)^2}\end{pmatrix}=\begin{pmatrix}1&1&\cdots&1\\1&W_{\frac{N}{2}}&\cdots&W_{\frac{N}{2}}^{\frac{N}{2}-1}\\\vdots&\vdots&&\vdots\\1&W_{\frac{N}{2}}^{\frac{N}{2}-1}&\cdots&W_{\frac{N}{2}}^{(\frac{N}{2}-1)^2}\end{pmatrix}\tag{2.34}$$

因此，(2.30)式的快速算法流程图同于(2.18)式的快速算法流程图，只需以(2.31)式为输入序列，输出则为$\boldsymbol{X}_{k_1}$的一个排列（下标按二进制逆序）。利用这个快速算法，所需乘法和加法次数如(2.21)，(2.22)式所示。

综上所述，利用 FPT 计算$N\times N(N=2^m)$型二維 DFT 总的流程图为



下面来分析所需的运算量。

由于多项式变换不需要乘法，所以整个算法需要乘法的地方是计算 $2N$ 个$\frac{N}{2}$点的简化一維 DFT 及用代换由$\{X_{n_1}^{(2)}(z)\}$获得$\{\widetilde{X}_{n_1}^{(2)}(u)\}$，前者需要 $2M_1$ 次复乘，后者需要$N\left(\frac{N}{2}-1\right)$次复乘，因此共需

$$M_u=2M_1+N\left(\frac{N}{2}-1\right)=2\left(\frac{1}{4}N^2\log_2 N-\frac{1}{2}N^2+N\right)+N\left(\frac{N}{2}-1\right)$$
$$=\frac{N^2}{2}\log_2 N-\frac{1}{2}N^2+N \tag{2.35}$$

次复乘，所需加法次数（复加）为：

$$A_d=A_1+2A_2+2A_3=N^2+2\cdot\frac{1}{2}N^2\log_2 N+2\cdot\frac{1}{2}N^2\log_2\frac{N}{2}$$
$$=2N^2\log_2 N. \tag{2.36}$$

通常的以 2 为基的二维 FFT 计算 $N\times N$ 型二維 DFT 所需的复乘及复加次数分别为

$$M_u'=2N\left(\frac{1}{2}N\log_2 N-\frac{3}{2}N+2\right)=N^2\log_2 N-3N^2+4N \tag{2.37}$$

$$A_d'=2N^2\log_2 N \tag{2.38}$$

对于一些$N\times N$二維 DFT 所需的运算量，列于表 1 。由表 1 可知，FPT 及FFT 所需的加法次数相同，当 $N\geqslant 32$ 时，FPT 比 FFT 所需的乘法次数减少约 30%左右，并且随N的增加减少得愈多。从而提高了计算精度。

表 1

二維DFT 大 小	FPT		FFT		$\left(1-\frac{M_u}{M_u'}\right)\cdot 100\%$
	乘法次数	加法次数	乘法次数	加法次数	
$2^2\times 2^2$	12	64	0	64	——
$2^3\times 2^3$	72	384	32	384	——
$2^4\times 2^4$	400	2048	320	2048	——
$2^5\times 2^5$	2080	10240	2176	10240	4.41
$2^6\times 2^6$	10304	49152	12544	49152	17.85
$2^7\times 2^7$	49280	229376	66048	229376	25.38
$2^8\times 2^8$	229632	1048576	328704	1048576	30.14
$2^9\times 2^9$	1049088	4718592	1574912	4718592	33.38
$2^{10}\times 2^{10}$	4719616	20971520	7344128	20971520	35.73

三、一 般 情 况

用(1.4)，(1.5)和(1.6)式计算$[X]$，这时模

$$z^M-1=(z^{\frac{M}{2^r}}-1)\prod_{i=1}^{r}(z^{\frac{M}{2^i}}+1) \tag{3.1}$$

如记

$$\overline{X}_{k_1}^{(i)}(z)\equiv\overline{X}_{k_1}(z)\bmod(z^{\frac{M}{2^i}}+1),(i=1,2,\cdots,r) \tag{3.2}$$

$$\overline{X}_{k_1}^{*}(z)\equiv\overline{X}_{k_1}(z)\bmod(z^{\frac{M}{2^r}}-1) \tag{3.3}$$

那么由孙子定理有

$$\overline{X}_{k_1}(z)\equiv\sum_{i=1}^{r}\left(-\frac{1}{2^i}\right)\frac{z^M-1}{z^{M/2^i}+1}\overline{X}_{k_1}^{(i)}(z)+\frac{1}{2^r}\frac{z^M-1}{z^{M/2^r}-1}\overline{X}_{k_1}^{*}(z)\bmod(z^M-1) \tag{3.4}$$

利用(1.5)式得到

$$X_{k_1,k_2}\equiv\overline{X}_{k_1}^{(i)}(z)\bmod(z-W_M^{k_2}),k_2=2^{i-1},3\cdot2^{i-1},\cdots,M-2^{i-1}\left(共\frac{M}{2^i}个\right) \tag{3.5}$$

$$(i=1,2,\cdots,r;k_1=0,1,\cdots,N-1).$$

$$X_{k_1,k_2}\equiv\overline{X}_{k_1}^{*}(z)\bmod(z-W_M^{k_2}),k_2=0,2^r,\cdots,M-2^r\left(共\frac{M}{2^r}个\right) \tag{3.6}$$

$$(k_1=0,1,\cdots,N-1).$$

又由 $\overline{X}_{k_1}(z)$和$\overline{X}_{k_1}^{(i)}(z)$, $\overline{X}_{k_1}^{*}(z)$的定义有

$$\overline{X}_{k_1}^{(i)}(z)\equiv\sum_{n_1=0}^{N-1}X_{n_1}^{(i)}(z)W_N^{n_1k_1}\bmod(z^{M/2^i}+1),(i=1,2,\cdots,r) \tag{3.7}$$

$$\overline{X}_{k_1}^{*}(z)\equiv\sum_{n_1=0}^{N-1}X_{n_1}^{*}(z)W_N^{n_1k_1}\bmod(z^{M/2^r}-1),(k_1=0,1,\cdots,N-1) \tag{3.8}$$

其中

$$X_{n_1}^{(i)}(z)\equiv X_{n_1}(z)\bmod(z^{M/2^i}+1),(i=1,2,\cdots,r) \tag{3.9}$$

$$X_{n_1}^{*}(z)\equiv X_{n_1}(z)\bmod(z^{M/2^r}-1),(n_1=0,1,\cdots,N-1) \tag{3.10}$$

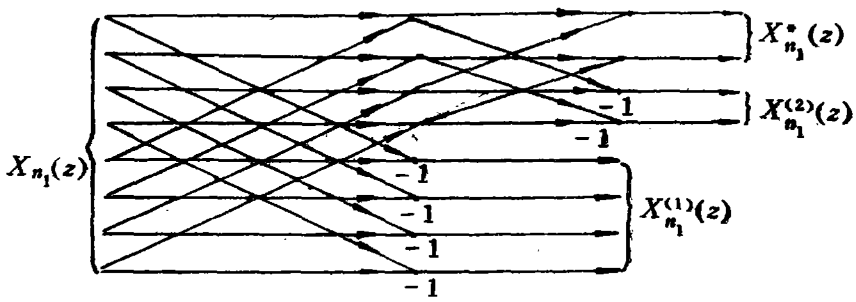
$\{X_{n_1}(z)\}$是所给数阵$[x]$的母多项式序列，是已知的，从而由上两式可求出$\{X_{n_1}^{(i)}(z)\}$和$\{X_{n_1}^{*}(z)\}$。并且可用 FFT 类型的快速算法计算。例如，$M=8,N=4\,(r=2)$，就有

$$X_{n_1}^{(1)}(z)\equiv X_{n_1}(z)\bmod(z^4+1)$$

$$X_{n_1}^{(2)}(z)\equiv X_{n_1}(z)\bmod(z^2+1)\qquad n_1=0,1,2,3.$$

$$X_{n_1}^{*}(z)\equiv X_{n_1}(z)\bmod(z^2-1)$$

其快速算法流程图为



由此知，计算$X^{(i)}_{n_1}(z)$和$X^*_{n_1}(z)$不需乘法只需加法，对于固定的n_1共需

$$M+\frac{M}{2}+\cdots+\frac{M}{2^{r-1}}=2M\left(1-\frac{1}{2^r}\right)$$

次加法，因此计算出所有$X^{(i)}_{n_1}(z)$，$X^*_{n_1}(z)$ （$n_1=0,1,\cdots,N-1$）共需

$$A_1=2NM\left(1-\frac{1}{2^r}\right) \tag{3.11}$$

次加法。

下面来计算(3.5)式。由(3.7)式有

$$\bar{X}^{(i)}_{\langle\frac{1}{2^{i-1}}k_1k_2\rangle}(z)\equiv\sum_{n_1=0}^{N-1}X^{(i)}_{n_1}(z)W_N^{\frac{1}{2^{i-1}}n_1k_1k_2}\equiv\sum_{n_1=0}^{N-1}X^{(i)}_{n_1}(z)W_M^{2^{r-i}n_1k_1k_2}\bmod(z^{M/2^i}+1)$$

注意到 $z\equiv W_M^{k_2}\bmod(z-W_M^{k_2})$和(3.5)式，仍记

$$\bar{X}^{(i)}_{\langle\frac{1}{2^{i-1}}k_1k_2\rangle}(z)\equiv\sum_{n_1=0}^{N-1}X^{(i)}_{n_1}(z)\left(z^{2^{r-i}}\right)^{n_1k_1}\equiv\sum_{n_1=0}^{N-1}X^{(i)}_{n_1}(z)\tilde{z}^{n_1k_1}\bmod(z^{M/2^i}+1)$$

$$(i=1,2,\cdots,r;k_1=0,1,\cdots,N-1;k_2=2^{i-1},3\cdot2^{i-1},\cdots,M-2^{i-1}). \tag{3.12}$$

上式中$\tilde{z}=z^{2^{r-i}}$是不可约多项式 $z^{M/2^i}+1$为模的N阶单位根，事实上

$$\tilde{z}^N=(z^{2^{r-i}})^{m-r+1}=(z^{M/2^i})^2\equiv(-1)^2=1 \quad \bmod(z^{M/2^i}+1)$$

$$\tilde{z}^{\frac{N}{2}}=z^{M/2^i}\equiv-1\bmod(z^{M/2^i}+1),$$

由此由[4]的定理6知(3.12)式是以($z^{M/2^i}+1$)为模，$\tilde{z}$为根、长度为N的多项式变换。由于$\left(\frac{1}{2^{i-1}}k_2,N\right)=1$，故$\left\langle\frac{1}{2^{i-1}}k_1k_2\right\rangle=\frac{1}{2^{i-1}}k_1k_2\bmod N$是$0,1,\cdots,N-1$的一个排列（$k_1=0,1,\cdots,N-1$）。又因$N$是2的幂，所以(3.12)式可用FPT计算。对于某个i，需$N\log_2N$对次数为$M/2^i-1$的多项式加法，所以共需

$$\frac{M}{2^i}N\log_2N$$

次加法。(3.12)式的右端与 k_2 无关，所以用 FPT 全部算出 $\overline{X}^{(i)}_{\langle\frac{1}{2^{i-1}}k_1k_2\rangle}(z)$ $(i=1, 2, \cdots, r)$ 共需

$$A_2=\sum_{i=1}^{r}\frac{1}{2^i}MN\log_2 N=MN\left(1-\frac{1}{2^r}\right)\log_2 N \tag{3.13}$$

次加法。

由于模是$M/2^i$ 次的，故由(3.12)式有

$$\overline{X}^{(i)}_{\langle\frac{1}{2^{i-1}}k_1k_2\rangle}(z)=\sum_{l=0}^{M/2^i-1}y^{(i)}_{k_1,l}z^l, \ (k_1=0, 1, \cdots, N-1; k_2=2^{i-1}, 3\cdot 2^{i-1}, \cdots, M-2^{i-1}) \tag{3.14}$$

于是就有

$$X_{\langle\frac{1}{2^{i-1}}k_1k_2\rangle, k_2}=\sum_{l=0}^{M/2^i-1}y^{(i)}_{k_1,l}W_M^{lk_2}, \ (i=1, 2, \cdots, r; k_1=0, 1, \cdots, N-1;$$
$$k_2=2^{i-1}, 3\cdot 2^{i-1}, \cdots, M-2^{i-1}) \tag{3.15}$$

(3.15)式可看作$M/2^i$ $(i=1, 2, \cdots, r)$点的一維简化 DFT，由于$M/2^i=2^{m-i}$，故有 FFT 类型的快速算法，这只要将

$$(y^{(i)}_{k_1,0}, W_M^{2^{i-1}}y^{(i)}_{k_1,1}, W_M^{2^i}y^{(i)}_{k_1,2}, \cdots\cdots, W_M^{2^{i-1}(M/2^i-1)}y^{(i)}_{k_1,M/2^i-1})^T \tag{3.16}$$

作为输入序列用 $M/2^i=2^{m-i}$ 点的 FFT 算法便得到 (3.15) 式的快速算法。对于固定的 i，需

$$M_{1,i}=N\left(\frac{M}{2^{i+1}}\log_2\frac{M}{2^i}-\frac{3}{2}\cdot\frac{M}{2^i}+2\right)+N\left(\frac{M}{2^i}-1\right)$$

$$=\frac{N}{2}\cdot\frac{M}{2^i}\log_2\frac{M}{2^i}-\frac{1}{2}MN\cdot\frac{1}{2^i}+N$$

次复乘和

$$A_{3,i}=N\cdot\frac{M}{2^i}\log_2\frac{M}{2^i}$$

次复加，因此算出所有的$X_{\langle\frac{1}{2^{i-1}}k_1k_2\rangle, k_2}$ $(i=1, 2, \cdots, r)$共需

$$M_1=\frac{1}{2}N\sum_{i=1}^{r}\frac{M}{2^i}\log_2\frac{M}{2^i}-\frac{1}{2}MN\left(1-\frac{1}{2^r}\right)+Nr \tag{3.17}$$

次复乘和

$$A_3=N\sum_{i=1}^{r}\frac{M}{2^i}\log_2\frac{M}{2^i} \tag{3.18}$$

次复加。

再计算(3.6)式。由(3.8)式，$M/2^r=\frac{N}{2}$，所以有

$$\overline{X}^*_{k_1}(z) \equiv \sum_{n_1=0}^{N-1} X^*_{n_1}(z) W_N^{n_1 k_1} \mod(z^{\frac{N}{2}}-1) \tag{3.19}$$

$$(k_1=0,1,\cdots,N-1; k_2=0,2^r,\cdots,M-2^r)$$

$$X_{k_1,k_2} \equiv \overline{X}^*_{k_1}(z) \mod(z-W_M^{k_2}). \tag{3.20}$$

作变换

$$z=W_N u, \quad W_N=e^{-i2\pi/N} \tag{3.21}$$

并记 $\widetilde{X}_{n_1}(u)=X_{n_1}(W_N u)$, $\widetilde{X}^*_{n_1}(u)=X^*_{n_1}(W_N u)$, $\widetilde{\overline{X}}^*_{k_1}(u)=\overline{X}^*_{k_1}(W_N u)$ (3.22)

于是有

$$\widetilde{\overline{X}}^*_{k_1}(u) \equiv \sum_{n_1=0}^{N-1} \widetilde{X}^*_{n_1}(u) W_N^{n_1 k_1} \mod(u^{\frac{N}{2}}+1) \tag{3.23}$$

设$t=\frac{1}{2^{r-1}}k_2$, $k_2=0,2^r,\cdots,M-2^r$; $t=0,2,\cdots,N-2$。则由 $z\equiv W_M^{k_2}\equiv W_N^t \mod(z-W_M^{k_2})$有

$$u\equiv W_N^{t-1} \mod(u-W_N^{t-1}) \tag{3.24}$$

因此由(3.23)和(3.20)式，若仍记

$$\widetilde{\overline{X}}^*_{\langle k_1(t-1)\rangle}(u) \equiv \sum_{n_1=0}^{N-1} \widetilde{X}^*_{n_1}(u) u^{n_1 k_1} \mod(u^{\frac{N}{2}}+1) \tag{3.25}$$

则有

$$X_{\langle k_1(t-1)\rangle,k_2} \equiv \widetilde{\overline{X}}^*_{\langle k_1(t-1)\rangle}(u) \mod(u-W_N^{t-1}) \tag{3.26}$$

$$(k_1=0,1,\cdots,N-1; t=0,2,\cdots,N-2; \ k_2=0,2^r,\cdots,M-2^r)$$

(3.25)式是以不可约多项式 $u^{\frac{N}{2}}+1$ 为模，u 为根，长度为N的多项式变换，$N=2^{m-r+1}$，故可用FPT计算。于是有

$$\widetilde{\overline{X}}^*_{\langle k_1(t-1)\rangle}(u)=\sum_{l=0}^{\frac{N}{2}-1} y^*_{k_1,l} u^l, (k_1=0,1,\cdots,N-1; t=0,2,\cdots,N-2) \tag{3.27}$$

所以

$$X_{\langle k_1(t-1)\rangle,k_2}=\sum_{l=0}^{\frac{N}{2}-1} y^*_{k_1,l} W_N^{l(t-1)}, (k_1=0,1,\cdots,N-1; t=0,2,\cdots,N-2; \ k_2=0,2^r,\cdots,M-2^r) \tag{3.28}$$

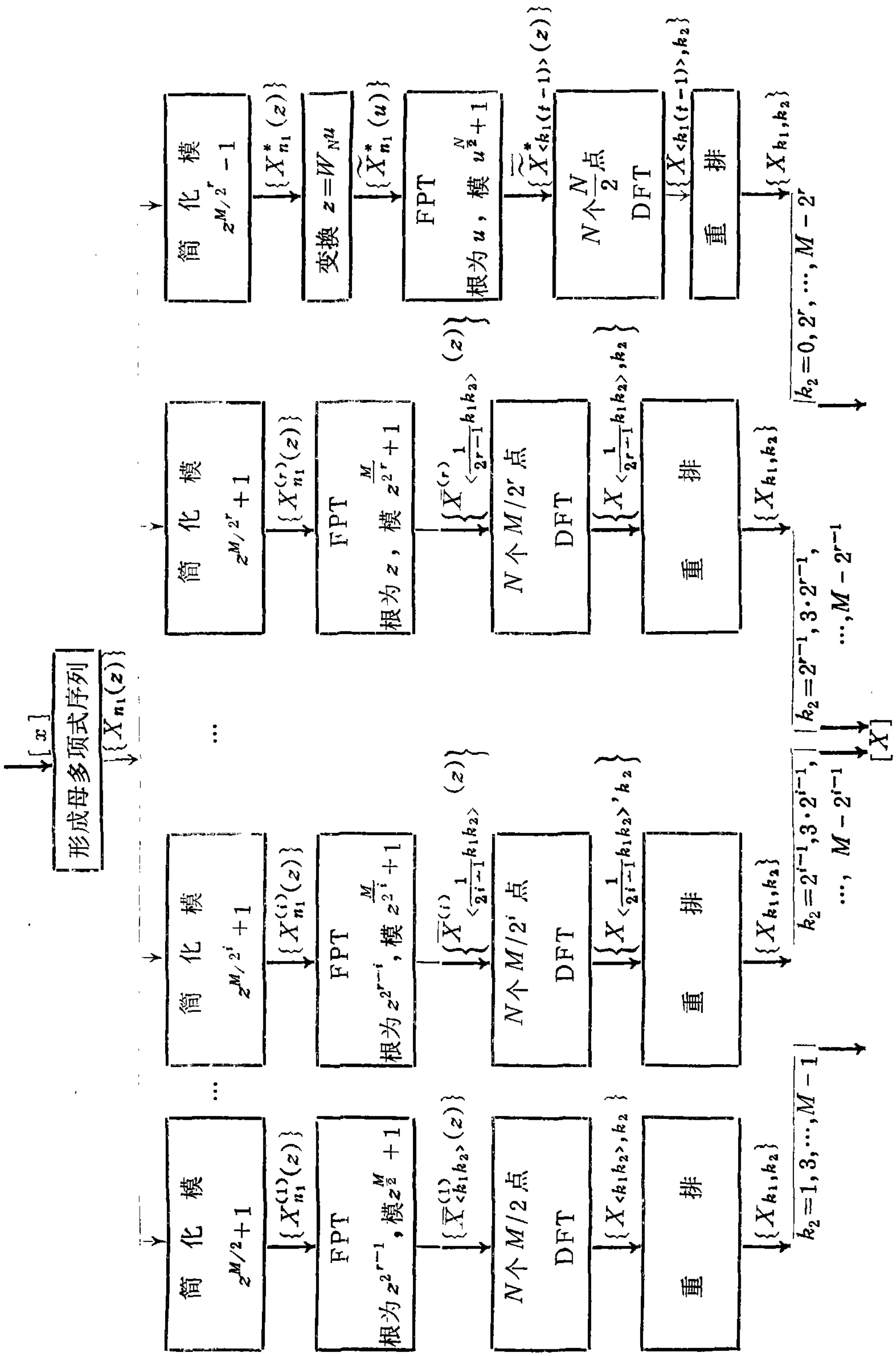
(3.25)式、(3.27)式和(3.28)式完全同于(2.27)式、(2.29)式和(2.30)式，所需的运算量也如(2.16)式、(2.21)式和(2.22)式所示，即

$$A_4=\frac{1}{2}N^2\log_2 N \tag{3.29}$$

$$M_2=\frac{1}{4}N^2\log_2 N-\frac{1}{2}N^2+N \tag{3.30}$$

$$A_5=\frac{1}{2}N^2\log_2\frac{N}{2} \tag{3.31}$$

综上所述，用FPT计算$N\times M$二維DFT总的流程图为


形成母多项式序列
简化模
FPT
DFT
重排
变换 $z=W_N^u$
$[X]$

下面讨论本算法所需的运算量。

N个$\frac{M}{2^i}$ $(i=1, 2, \cdots, r-1)$ 点 DFT 和 $2N$个$\frac{N}{2}$点 DFT 以及用变换 $z=W_N u$ 由 $\{X^*_{n_1}(z)\}$得到$\{\widetilde{X}^*_{n_1}(u)\}$需要乘法，共需

$$M_u=M_1+M_2+N\left(\frac{N}{2}-1\right)=\frac{1}{2}N\sum_{i=1}^{r}\frac{M}{2^i}\log_2\frac{M}{2^i}+\frac{1}{4}N^2\log_2 N-\frac{1}{2}NM\left(1-\frac{1}{2^r}\right)+rN$$

次乘法，加法次数为

$$A_d=A_1+A_2+A_3+A_4+A_5=N\sum_{i=1}^{r}\frac{M}{2^i}\log_2\frac{M}{2^i}+N^2\log_2 N-\frac{1}{2}N^2$$

$$+2NM\left(1-\frac{1}{2^r}\right)+NM\left(1-\frac{1}{2^r}\right)\log_2 N.$$

注意到 $2^r=\frac{2M}{N}$，$\sum_{i=1}^{r}\frac{M}{2^i}\log_2\frac{M}{2^i}=M\left(1-\frac{1}{2^r}\right)\log_2 M-M\left(2-\frac{r+2}{2^r}\right)$，有

$$M_u=\frac{1}{2}NM\log_2 M-\frac{3}{2}NM+N^2+N(1+\log_2 M-\log_2 N) \tag{3.32}$$

$$A_d=NM\log_2 NM \tag{3.33}$$

通常的以2为基的二維 FFT（$N\times M$型）所需的乘法和加法次数分别为

$$M'_u=\frac{1}{2}NM\log_2 NM-3NM+2(N+M) \tag{3.34}$$

$$A'_d=NM\log_2 NM \tag{3.35}$$

对于一些$N\times M$型二維 DFT，用 FPT 和 FFT 计算所需的乘法和加法次数列于表2。

由表2知，两者加法次数相同。当$N\geqslant 2^4$，$M\geqslant 2^5$时，FPT 所需乘法次数比 FFT 所需的减少约 30—40%。

表 2

二維DFT	FPT		FFT		$\left(1-\frac{M_u}{M'_u}\right)\cdot 100\%$
大小	乘法次数	加法次数	乘法次数	加法次数	
$2^2\times 2^4$	60	384	40	384	—
$2^3\times 2^4$	144	896	128	896	—
$2^3\times 2^5$	344	2048	336	2048	—
$2^4\times 2^5$	800	4608	864	4608	7.40
$2^4\times 2^6$	1840	10240	2208	10240	16.66
$2^5\times 2^6$	4160	22528	5312	22528	21.68
$2^5\times 2^7$	9312	49152	12608	49152	26.14
$2^6\times 2^7$	20608	106496	29056	106496	29.07
$2^6\times 2^8$	45248	229376	66176	229376	31.62
$2^7\times 2^8$	98560	491520	148224	491520	33.50
$2^7\times 2^9$	213376	1048576	328960	1048576	35.13
$2^8\times 2^9$	459264	2228224	722432	2228224	36.42
$2^8\times 2^{10}$	983808	4718592	1575424	4718592	37.55
$2^9\times 2^{10}$	2098176	9961472	3410944	9961472	38.48

四、结 束 语

本文研究了用 FPT 计算 $N\times M$ 二維 DFT ($M=2^m$, $N=2^{m-r+1}$, $1\leqslant r\leqslant m$) 的方法，与一般二維 FFT 比较，乘法次数减少了 30—40%，加法次数相同，从而减少了由舍入引起的积累误差，提高了计算精度，又由于本算法是对$[X]$不同的列分别计算的，因此适用于并行算法，提高了计算速度。

参 考 文 献

[1] H.J.Nussbaumer and P.Quandalle,"Computation of Convolutions and Discrete Fourier Transforms by Polynomial Transforms,"IBM.J.Res. Develop., Vol.22, №.2, March 1978.

[2] H.J.Nussbaumer and P. Quandalle, "Fast Computation of Discrete Fourier Transforms Using Polynomial Transforms," IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP—27, №.2, April 1979.

[3] H.J. Nussbaumer, "Fast Polynomial Transform Algorithms for Digital Convolution," IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP—28, №.2, April 1980.

[4] 蒋增荣，"多项式变换及其在卷积计算中的应用"，国防科技大学学报，1980 年第三期.

[5] 蒋增荣，"用多项式变换计算多项式乘积"，湖南数学年刊，1982年第二期.

Computation of Two Dimensional Discrete Fourier Transforms (DFT) Using Fast Polynomial Transforms (FPT)

Jiang Zeng-rong

Abstract

In this paper, a fast algorithm is developed to compute two-dimensional Discrete Fourier Transform (DFT) of an array of $N\times M$ complex number points using FPT, where $M=2^m$, $N=2^{m-r+1}$, $1\leqslant r\leqslant m$. As compared with the conventional radix-2 two-dimensional Fast Fourier Transforms, this new algorithm requires less number of multiplications (decrease by 30—40%) and same number of additions, so that elevates the computing accuracy.

The number of multiplications and additions are respectively

$$M_u = \frac{1}{2} MN \log_2 M - \frac{3}{2} MN + N^2 + N(1 + \log_2 M - \log_2 N)$$

$$A_d = NM \log_2 MN.$$

This algorithm also has the advantage to adopt the parallel algorithm.