

文章编号:1001-9081(2010)10-2777-04

频域抽取多维向量基快速傅里叶变换

徐妮妮,于海艳,肖志涛

(天津工业大学 信息与通信工程学院,天津 300160)

(xunini81240124@gmail.com)

摘要:给出了频域抽取(DIF)多维向量基快速傅里叶变换(FFT)算法。对多维频域信号的每一维,采用向量基2频域抽取法,导出了快速算法蝶形运算的一般形式。该FFT算法适合于维数为任意整数的情况,当维数为1时,算法退化为著名的频域抽取向量基2FFT算法。为了便于编程实现,以频域抽取3维向量基FFT算法为例,给出了快速算法实现流程,该流程易于向任意整数维推广。计算量比较结果显示,频域抽取多维向量基FFT算法比多维分离式FFT算法计算量低。

关键词:多维离散傅里叶变换;频域抽取;多维向量基;快速傅里叶变换;多维分离式FFT算法

中图分类号: TP301.6 **文献标志码:** A

Multi-dimensional vector radix fast Fourier transform with decimation in frequency domain

XU Ni-ni, YU Hai-yan, XIAO Zhi-tao

(School of Information and Communication Engineering, Tianjin Polytechnic University, Tianjin 300160, China)

Abstract: A multi-dimensional vector radix Fast Fourier Transform (FFT) algorithm with Decimation In Frequency (DIF) was proposed. Using vector radix 2 DIF for each dimension of the multi-dimensional frequency domain signal, the general form of butterfly computation of the FFT algorithm was obtained. This FFT algorithm can be used with arbitrary integer dimensions. For 1-dimension, the algorithm will be simplified as the well-known DIF vector radix 2 FFT. To facilitate the programming, flow chart with DIF 3-dimension vector radix FFT was given. And this flow chart can be extended to any integer dimensions easily. The comparison results show that, compared with multi-dimensional separable FFT, the DIF multi-dimensional vector radix FFT algorithm has lower calculation load.

Key words: multi-dimensional discrete Fourier transform; Decimation in Frequency (DIF); multi-dimensional vector radix; Fast Fourier Transform (FFT); multi-dimensional separable FFT algorithm

0 引言

离散傅里叶变换(Direct Fourier Transformer, DFT)为数字信号频域分析提供了有利的变换工具,但其直接实现运算量大,运算时间长。快速傅里叶变换(Fast Fourier Transform, FFT)的提出使得DFT在数字信号处理领域的应用成为现实。

国内外学者对向量基和分裂基FFT算法做了深入地研究^[7-10],但都是关于1维或2维数据的快速算法。文献[11]对时域抽取多维向量基FFT算法进行了详细的研究,首次实现了3维以上时域抽取向量基FFT算法,但未给出具体的编程实现框图。实际上,快速算法中的蝶形运算到具体程序实现还有一段距离。文献[12]采用奇偶抽取法对时域抽取3维向量基FFT算法进行了研究,但编程实现框图中输入数据码位编址不能用同一组存储单元通过数据交换完成。由于向量基算法在理论推导、运算量和软硬件实现方面的综合特性较其他算法优越,因此本文提出了频域抽取多维向量基FFT算法并解决了码位编址问题,且以3维为例,给出了程序实现框图;当维数为1时,该算法即蜕化为频域抽取向量基2FFT算法。

1 频域抽取多维向量基FFT算法

1.1 多维信号的离散傅里叶变换定义

对于 S (S 为任意整数)维有限区间离散时间信号 $x(n_1, n_2, \dots, n_S)$ ($n_i = 0 \sim N_i - 1, i = 1, 2, \dots, S$),它的DFT定义为:

$$X(k_1, k_2, \dots, k_S) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \dots \sum_{n_S=0}^{N_S-1} x(n_1, n_2, \dots, n_S) W_{N_1}^{k_1 n_1} W_{N_2}^{k_2 n_2} \dots W_{N_S}^{k_S n_S} \quad (1)$$

其中: $k_i = 0 \sim N_i - 1, W_{N_i}^{k_i n_i} = \exp(-j \frac{2\pi k_i n_i}{N_i}), j = \sqrt{-1}, i = 1, 2, \dots, S$ 。

为了使问题简化,下文假定各维采样点数相同,即 $N_i = N = 2^M$ (M 为非负整数, $i = 1, 2, \dots, S$)。

1.2 频域抽取多维向量基FFT算法实现

S 维信号 $x(n_1, n_2, \dots, n_S)$ ($n_i = 0 \sim N - 1, i = 1, 2, \dots, S$),根据是否沿第 i 维空间变量 n_i 平移,可以把原 (N, N, \dots, N) 点的 S 维序列分割成 2^S 组 $(N/2, N/2, \dots, N/2)$ 点的 S 维子序列,即:

$$x_{c_1 c_2 \dots c_S}(n_1, n_2, \dots, n_S) =$$

收稿日期:2010-04-14;**修回日期:**2010-06-12。 **基金项目:**国家青年自然科学基金资助项目(60602036);国家自然科学基金资助项目(20676100);天津工业大学校基金资助项目(029470)。

作者简介:徐妮妮(1974-),女,山东莱州人,副教授,博士,主要研究方向:现代信息处理、图像及视频编码;于海艳(1986-),女,吉林长春人,硕士研究生,主要研究方向:数字图像处理;肖志涛(1971-),男,河北邯郸人,教授,博士,主要研究方向:图像处理、模式识别。

$$x\left(n_1 + c_1 \frac{N}{2}, n_2 + c_2 \frac{N}{2}, \dots, n_S + c_S \frac{N}{2}\right);$$

$$n_i = 0 \sim \frac{N}{2} - 1, c_i = 0 \text{ 或 } 1, i = 1, 2, \dots, S \quad (2)$$

其中: $c_i = 1$ 表示沿 n_i 移动 $N/2$, $c_i = 0$ 不移动。例如, 当 $S = 3$ 时, 按二进制数 $c_1 c_2 \dots c_S$ 由小到大的组合, 对应的 8 组平移分割子序列的具体表达形式见表 1。考虑到 $W_N^{k_i N/2} = (-1)^{k_i}$, 因此, S 维 (N, N, \dots, N) 点离散时间信号 $x(n_1, n_2, \dots, n_S)$ 的 DFT 可以用子序列表示为:

$$X(k_1, k_2, \dots, k_S) = \sum_{n_1=0}^{N/2-1} \sum_{n_2=0}^{N/2-1} \dots \sum_{n_S=0}^{N/2-1} \left[\sum_{c_1=0}^1 \sum_{c_2=0}^1 \dots \sum_{c_S=0}^1 (-1)^{c_1 k_1 + c_2 k_2 + \dots + c_S k_S} x_{c_1 c_2 \dots c_S}(n_1, n_2, \dots, n_S) \right]$$

$$W_N^{k_1 n_1} W_N^{k_2 n_2} \dots W_N^{k_S n_S}; k_i = 0 \sim N-1, i = 1, 2, \dots, S \quad (3)$$

表 1 (N, N, N) 点 3 维序列的平移分割子序列

c_i 组合	平移分割子序列	等价子序列
$c_1 c_2 c_3$	$x_{c_1 c_2 c_3}(n_1, n_2, n_3)$	$x(n_1 + c_1 N/2, n_2 + c_2 N/2, n_3 + c_3 N/2)$
000	$x_{000}(n_1, n_2, n_3)$	$x(n_1, n_2, n_3)$
001	$x_{001}(n_1, n_2, n_3)$	$x(n_1, n_2, n_3 + N/2)$
010	$x_{010}(n_1, n_2, n_3)$	$x(n_1, n_2 + N/2, n_3)$
011	$x_{011}(n_1, n_2, n_3)$	$x(n_1, n_2 + N/2, n_3 + N/2)$
100	$x_{100}(n_1, n_2, n_3)$	$x(n_1 + N/2, n_2, n_3)$
101	$x_{101}(n_1, n_2, n_3)$	$x(n_1 + N/2, n_2, n_3 + N/2)$
110	$x_{110}(n_1, n_2, n_3)$	$x(n_1 + N/2, n_2 + N/2, n_3)$
111	$x_{111}(n_1, n_2, n_3)$	$x(n_1 + N/2, n_2 + N/2, n_3 + N/2)$

若把 k_i 的奇偶性记为 p_i (k_i 为偶数, $p_i = 0$; k_i 为奇数, $p_i = 1$), 式(3)可以改写为:

$$X(2k_1 + p_1, 2k_2 + p_2, \dots, 2k_S + p_S) = \sum_{n_1=0}^{N/2-1} \sum_{n_2=0}^{N/2-1} \dots \sum_{n_S=0}^{N/2-1} \left[\sum_{c_1=0}^1 \sum_{c_2=0}^1 \dots \sum_{c_S=0}^1 (-1)^{[c_1 c_2 \dots c_S] [2k_1 + p_1, 2k_2 + p_2, \dots, 2k_S + p_S]^T} \right.$$

$$X_{c_1 c_2 \dots c_S}(n_1, n_2, \dots, n_S) \left. W_N^{(2k_1 + p_1)n_1} W_N^{(2k_2 + p_2)n_2} \dots W_N^{(2k_S + p_S)n_S}; \right.$$

$$k_i = 0 \sim \frac{N}{2} - 1, p_i = 0 \text{ 或 } 1, i = 1, 2, \dots, S \quad (4)$$

把式(4)简记为:

$$X_{p_1 p_2 \dots p_S}(k_1, k_2, \dots, k_S) = \sum_{n_1=0}^{N/2-1} \sum_{n_2=0}^{N/2-1} \dots \sum_{n_S=0}^{N/2-1} [u_{p_1 p_2 \dots p_S}(n_1, n_2, \dots, n_S) W_N^{k_1 n_1} W_N^{k_2 n_2} \dots W_N^{k_S n_S};$$

$$k_i = 0 \sim \frac{N}{2} - 1, p_i = 0 \text{ 或 } 1, i = 1, 2, \dots, S \quad (5)$$

其中:

$$X_{p_1 p_2 \dots p_S}(k_1, k_2, \dots, k_S) =$$

$$X(2k_1 + p_1, 2k_2 + p_2, \dots, 2k_S + p_S) \quad (6)$$

$$u_{p_1 p_2 \dots p_S}(n_1, n_2, \dots, n_S) =$$

$$v_{p_1 p_2 \dots p_S}(n_1, n_2, \dots, n_S) W_N^{p_1 n_1} W_N^{p_2 n_2} \dots W_N^{p_S n_S} \quad (7)$$

$$v_{p_1 p_2 \dots p_S}(n_1, n_2, \dots, n_S) =$$

$$\sum_{c_1=0}^1 \sum_{c_2=0}^1 \dots \sum_{c_S=0}^1 (-1)^{\tau [c_1 c_2 \dots c_S]} x_{c_1 c_2 \dots c_S}(n_1, n_2, \dots, n_S) \quad (8)$$

$$\tau = [c_1 \ c_2 \ \dots \ c_S] [p_1 \ p_2 \ \dots \ p_S]^T \quad (9)$$

可见, (N, N, \dots, N) 点的 S 维 DFT 可以按 k_i 的奇偶性抽取分解为 2^S 个 $(N/2, N/2, \dots, N/2)$ 点的 S 维短序列 $u_{p_1 p_2 \dots p_S}(n_1, n_2, \dots, n_S)$ 的 DFT。频域数据按 k_i

的奇偶性抽取分解, 即为向量基 2 频域抽取法名字的来源。式(7)~(9)则构成了频域抽取 S 维向量基 FFT 变换的基本蝶形运算, 为了进一步减少基本蝶形运算的加法次数, 本文将在 1.3 节详细给出其运算实现公式和流程。

当 $S = 3$ 时, 按二进制数 $p_1 p_2 p_3$ 由小到大的组合, 3 维基本蝶形运算涉及的序列, 如频域抽取子序列 $X_{p_1 p_2 p_3}(k_1, k_2, k_3)$ 、旋转因子 $W_N^{p_1 n_1} W_N^{p_2 n_2} W_N^{p_3 n_3}$ 和线性组合序列 $v_{p_1 p_2 p_3}(n_1, n_2, n_3)$ 的具体形式见表 2。

表 2 3 维基本蝶形运算涉及的序列

p_i 组合	频域抽取子序列	旋转因子	线性组合序列
$p_1 p_2 p_3$	$X_{p_1 p_2 p_3}(k_1, k_2, k_3)$	$W_N^{p_1 n_1} W_N^{p_2 n_2} W_N^{p_3 n_3}$	$v_{p_1 p_2 p_3}(n_1, n_2, n_3)$
000	$X_{000}(k_1, k_2, k_3)$	$(W_N^{n_1})^0 (W_N^{n_2})^0 (W_N^{n_3})^0$	$v_{000}(n_1, n_2, n_3)$
001	$X_{001}(k_1, k_2, k_3)$	$(W_N^{n_1})^0 (W_N^{n_2})^0 (W_N^{n_3})^1$	$v_{001}(n_1, n_2, n_3)$
010	$X_{010}(k_1, k_2, k_3)$	$(W_N^{n_1})^0 (W_N^{n_2})^1 (W_N^{n_3})^0$	$v_{010}(n_1, n_2, n_3)$
011	$X_{011}(k_1, k_2, k_3)$	$(W_N^{n_1})^0 (W_N^{n_2})^1 (W_N^{n_3})^1$	$v_{011}(n_1, n_2, n_3)$
100	$X_{100}(k_1, k_2, k_3)$	$(W_N^{n_1})^1 (W_N^{n_2})^0 (W_N^{n_3})^0$	$v_{100}(n_1, n_2, n_3)$
101	$X_{101}(k_1, k_2, k_3)$	$(W_N^{n_1})^1 (W_N^{n_2})^0 (W_N^{n_3})^1$	$v_{101}(n_1, n_2, n_3)$
110	$X_{110}(k_1, k_2, k_3)$	$(W_N^{n_1})^1 (W_N^{n_2})^1 (W_N^{n_3})^0$	$v_{110}(n_1, n_2, n_3)$
111	$X_{111}(k_1, k_2, k_3)$	$(W_N^{n_1})^1 (W_N^{n_2})^1 (W_N^{n_3})^1$	$v_{111}(n_1, n_2, n_3)$

1.3 S 维基本蝶形运算的实现

令 τ 为 2^S 行 2^S 列符号矩阵, 其第 i 行第 j 列元素为:

$$\tau_{ij} = (-1)^{[c_1 \ c_2 \ \dots \ c_S] [p_1 \ p_2 \ \dots \ p_S]^T} \quad (10)$$

其中: $c_k = 0$ 或 1 , $p_k = 0$ 或 1 , $1 \leq k \leq S$, $i = c_1 2^{S-1} + c_2 2^{S-2} + \dots + c_S 2^0$, $j = p_1 2^{S-1} + p_2 2^{S-2} + \dots + p_S 2^0$ 。

若记:

$$U = \begin{bmatrix} u_{00\dots 0}(n_1, n_2, \dots, n_S) \\ u_{00\dots 1}(n_1, n_2, \dots, n_S) \\ \vdots \\ u_{11\dots 1}(n_1, n_2, \dots, n_S) \end{bmatrix}, V = \begin{bmatrix} v_{00\dots 0}(n_1, n_2, \dots, n_S) \\ v_{00\dots 1}(n_1, n_2, \dots, n_S) \\ \vdots \\ v_{11\dots 1}(n_1, n_2, \dots, n_S) \end{bmatrix}$$

$$X = \begin{bmatrix} x_{00\dots 0}(n_1, n_2, \dots, n_S) \\ x_{00\dots 1}(n_1, n_2, \dots, n_S) \\ \vdots \\ x_{11\dots 1}(n_1, n_2, \dots, n_S) \end{bmatrix},$$

$$W = \text{diag} \begin{bmatrix} (W_N^{n_1})^0 (W_N^{n_2})^0 \dots (W_N^{n_S})^0 \\ (W_N^{n_1})^0 (W_N^{n_2})^0 \dots (W_N^{n_S})^1 \\ \vdots \\ (W_N^{n_1})^1 (W_N^{n_2})^1 \dots (W_N^{n_S})^1 \end{bmatrix}$$

(11)

则式(7)~(9)构成的基本蝶形运算可以用上述矩阵表示为:

$$U = WV = W\tau X \quad (12)$$

$$\begin{bmatrix} u_{000}(n_1, n_2, n_3) \\ u_{001}(n_1, n_2, n_3) \\ u_{010}(n_1, n_2, n_3) \\ u_{011}(n_1, n_2, n_3) \\ u_{100}(n_1, n_2, n_3) \\ u_{101}(n_1, n_2, n_3) \\ u_{110}(n_1, n_2, n_3) \\ u_{111}(n_1, n_2, n_3) \end{bmatrix} = \text{diag} \begin{bmatrix} (W_N^{n_1})^0 (W_N^{n_2})^0 (W_N^{n_3})^0 \\ (W_N^{n_1})^0 (W_N^{n_2})^0 (W_N^{n_3})^1 \\ (W_N^{n_1})^0 (W_N^{n_2})^1 (W_N^{n_3})^0 \\ (W_N^{n_1})^0 (W_N^{n_2})^1 (W_N^{n_3})^1 \\ (W_N^{n_1})^1 (W_N^{n_2})^0 (W_N^{n_3})^0 \\ (W_N^{n_1})^1 (W_N^{n_2})^0 (W_N^{n_3})^1 \\ (W_N^{n_1})^1 (W_N^{n_2})^1 (W_N^{n_3})^0 \\ (W_N^{n_1})^1 (W_N^{n_2})^1 (W_N^{n_3})^1 \end{bmatrix} \begin{bmatrix} v_{000}(n_1, n_2, n_3) \\ v_{001}(n_1, n_2, n_3) \\ v_{010}(n_1, n_2, n_3) \\ v_{011}(n_1, n_2, n_3) \\ v_{100}(n_1, n_2, n_3) \\ v_{101}(n_1, n_2, n_3) \\ v_{110}(n_1, n_2, n_3) \\ v_{111}(n_1, n_2, n_3) \end{bmatrix} =$$

当 $S = 3$ 时,式(12)表示为:

$$\text{diag} \begin{bmatrix} 1 \\ W_N^{n_3} \\ W_N^{n_2} \\ W_N^{n_1+n_3} \\ W_N^{n_1+n_2} \\ W_N^{n_1+n_2+n_3} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} x_{000}(n_1, n_2, n_3) \\ x_{001}(n_1, n_2, n_3) \\ x_{010}(n_1, n_2, n_3) \\ x_{011}(n_1, n_2, n_3) \\ x_{100}(n_1, n_2, n_3) \\ x_{101}(n_1, n_2, n_3) \\ x_{110}(n_1, n_2, n_3) \\ x_{111}(n_1, n_2, n_3) \end{bmatrix};$$

$$n_i = 0 \sim \frac{N}{2} - 1, i = 1, 2, 3$$

符号矩阵 τ 可以分解为 S 个相同维数的稀疏矩阵 τ_i 的乘积,即:

$$\tau = \prod_{i=1}^S \tau_i = \tau_1 \cdot \tau_2 \cdot \cdots \cdot \tau_S \quad (14)$$

τ_i 表示基本蝶形运算中节距为 2^{i-1} 的 2^{S-1} 个二输入二输出蝶形。符号矩阵 τ 改写后,基本蝶形复数加法运算量由 $2^S(2^S - 1)$ 次减小为 $2^S \cdot S$ 次,复数乘法运算量 $2^S - 1$ 次(因为每个基本蝶形的第一个旋转因子恒为“1”)保持不变。

当 $S = 3$ 时:

$\tau =$

$$\begin{bmatrix} J & 0 & 0 & 0 \\ 0 & J & 0 & 0 \\ 0 & 0 & J & 0 \\ 0 & 0 & 0 & J \end{bmatrix} \begin{bmatrix} I_{2 \times 2} & I_{2 \times 2} & 0 & 0 \\ I_{2 \times 2} & -I_{2 \times 2} & 0 & 0 \\ 0 & 0 & I_{2 \times 2} & I_{2 \times 2} \\ 0 & 0 & I_{2 \times 2} & -I_{2 \times 2} \end{bmatrix} \begin{bmatrix} I_{4 \times 4} & I_{4 \times 4} \\ I_{4 \times 4} & -I_{4 \times 4} \end{bmatrix} \quad (15)$$

其中: $J = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$, $I_{n \times n}$ 是 $n \times n$ 维单位阵。可见基本蝶形

复数加法运算量由 56 次减小为 24 次,复数乘法运算量 7 次保持不变。

$$\text{令 } U = W \left(\prod_{i=1}^S \tau_i \right) \quad (16)$$

式(16)即为频域抽取 S 维向量基基本蝶形运算实现公式,当 $S = 3$ 时,其运算流程图如图1所示。

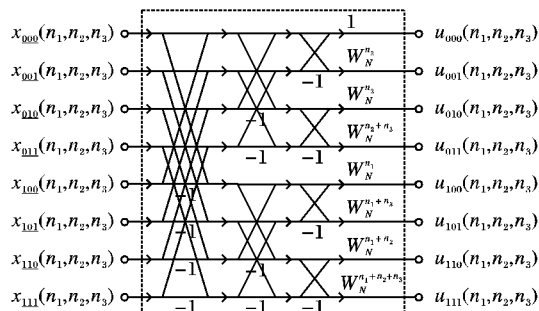


图1 频域抽取3维向量基基本蝶形运算流程

由式(7)~(9)可知,按频域抽取方法, S 维 (N, N, \dots, N) 点 DFT $X(k_1, k_2, \dots, k_S)$ 可以分解为 $2^{S(M-1)}$ 个基本蝶形和 2^S 个 S 维 $(N/2, N/2, \dots, N/2)$ 点的子序列 $u_{p_1 p_2 \dots p_S}(n_1, n_2, \dots, n_S)$ 的 DFT, 同样 S 维 $(N/2, N/2, \dots, N/2)$ 点子序列的 DFT 可以分解为 $2^{S(M-2)}$ 个基本蝶形和 2^S 个 S 维 $(N/4, N/4, \dots, N/4)$ 点子序列的 DFT, 这样不断分解可以得到 S 维 $(1, 1, \dots, 1)$ 点的 DFT, S 维 $(1, 1, \dots, 1)$ 点的 DFT 就是其本身。因为 $N = 2^M$, 因此这样的分解共有 M 级, 每一级由 $2^{S(M-1)}$ 个基本

蝶形构成。例如, $(4, 4, 4)$ 点 3 维 DFT 的频域抽取 3 维向量

基 FFT 运算流程如图2所示。为使图面简明, 图中只画了第一级蝶形具有代表性节点的信号流图(即一个完整的基本蝶形, 本级中这样的基本蝶形共有 8 个), 画出的基本蝶形恰好是图1中输入输出序列标号 $n_1 = 0, n_2 = 0, n_3 = 0$ 时的情况。图2中 $(2, 2, 2)$ 点 DFT 流程恰好是图1矩形虚线框中的一个完整的基本蝶形运算流程, 只是此时 8 个旋转因子的取值都为 1。

1.4 输入输出序列的排序

由于 FFT 流程图输入序列 $x(n_1, n_2, \dots, n_S)$ 与其输出序列 $X(k_1, k_2, \dots, k_S)$ 都是 S 维序列, 而 FFT 流程图中输入和输出都是按图2的形式在平面上排成一维序列, 下面分别讨论二者的排序规律, 排序规律对于频域抽取 S 维向量基 FFT 运算用统一程序实现是有意义的。

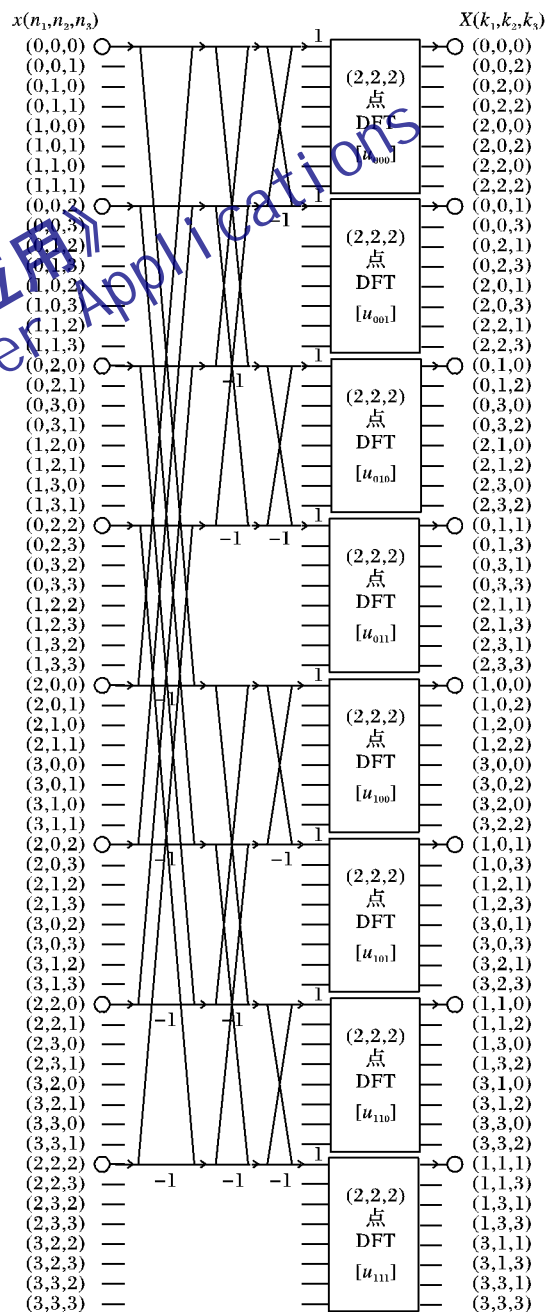


图2 $(4, 4, 4)$ 点频域抽取3维向量基FFT运算流程
(为使图面简明, 图中只画了具有代表性节点的信号流图)

输出序列 $X(k_1, k_2, \dots, k_S)$ 的一维排序规律如图3。图3中第 $l-1$ ($l = M, \dots, 1$) 级的 2^S 个 S 维 $(2^{l-1}, 2^{l-1}, \dots, 2^{l-1})$ 点序列 $X_{p_1 p_2 \dots p_S}^{l-1}(k_1, k_2, \dots, k_S)$ 表示对上一级每个 S 维 $(2^l, 2^l, \dots, 2^l)$ 点序列 $X_{p_1 p_2 \dots p_S}^l(k_1, k_2, \dots, k_S)$ 按 k_i 维坐标奇偶抽取分解的结果, k_i 的奇偶性记为 p_i , k_i 为偶数, $p_i = 0$, k_i 为奇数,

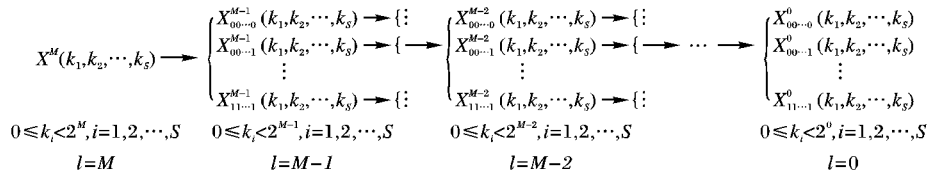


图3 频域抽取 S 维向量基 FFT 流程图中输出序列的一维排序

频域抽取 S 维向量基 FFT 流程图中同一水平线上, 输入数据 $x(n_1, n_2, \dots, n_S)$ 跟输出数据 $X(k_1, k_2, \dots, k_S)$ 的第 i 维坐标标号 k_i 和 n_i 具有严格的二进制码位倒序的性质, 若 k_i 用二进制数 $(b_{M-1}b_{M-2} \dots b_0)_2$ 表示, 则 n_i 用二进制数 $(b_0b_1 \dots b_{M-1})_2$ 表示, 数制转换的数学表达式为 $k_i = b_{M-1}2^{M-1} + b_{M-2}2^{M-2} + \dots + b_12^1 + b_02^0$, $n_i = b_02^{M-1} + b_12^{M-2} + \dots + b_{M-2}2^1 + b_{M-1}2^0$ 。因此, 若输出序列 $X(k_1, k_2, \dots, k_S)$ 的一维排序已知, 则按上述对应关系, 容易得到 FFT 流程图中输入序列 $x(n_1, n_2, \dots, n_S)$ 的一维排序。

2 编程实现

2.1 同址运算

S 维 (N, N, \dots, N) 点序列 $x(n_1, n_2, \dots, n_S)$ 做频域抽取向量基 FFT 运算, 共需进行 M 级蝶形运算。因为每级蝶形输入数据和输出数据数目相同, 而且每级中的蝶形输入在蝶形运算完成后对后序运算不发生任何影响, 故在一级蝶形运算完成后, 可以直接把这级蝶形输出信号覆盖输入信号的存储空间, 然后再进行下一级蝶形运算, 这样的存储方式称为同址运算, 每一级内的基本蝶形同样可以采用同址运算的方式实现, 同址运算可以大大节省内存, 降低设备成本。

2.2 程序框图

根据式(5)~(9)、(17), 同时考虑到蝶形运算的级数与同址运算, 得出频域抽取 S 维向量基 FFT 变换程序框图如图4、图5和图6所示, 该程序框图可以推广到 S 为任意维的情形, 方便了工程技术人员对本文提出的快速算法在工程领域的应用。

图5所示的基本蝶形运算比严格同址运算需要额外增加2个临时存储单元, 当维数增大时, 基本蝶形运算所需的临时存储单元数目不变, 当维数较高时, 编程实现可以大大减少程序代码, 其优越性比较明显。

图6所示的倒序程序框图是必需的, 由于采用同址运算, 存储 $x(n_1, n_2, \dots, n_S)$ 的存储单元在 M 级蝶形运算完成后实际上存储的是非自然排序的 $X(k_1, k_2, \dots, k_S)$, 由本文1.4节可知, n_i 和 k_i 具有严格的二进制码位倒序性质, 因此为了得到自然排序输出序列, 需要对输出序列倒序。

本文给出的编程实现方法简洁, 实用性强, 适合维数 S 取确切值的情况, 根据频域抽取 S 维向量基 FFT 运算流程图的特点, 也可以用统一的程序实现任意维频域抽取向量基 FFT 变换, 篇幅所限, 另文再述。

3 计算量的比较

表3给出了 (N, N, \dots, N) 点 S 维信号的4种离散傅里叶变换的计算量, 即各种变换的复数乘法和复数加法的次数与 N 的关系。通过比较, 可以清晰地看到, 频域抽取 S 维向量基

$p_i = 1$ ($i = 1, 2, \dots, S$) $\circ X^M(k_1, k_2, \dots, k_S)$ 初始化为 $X(k_1, k_2, \dots, k_S)$, $0 \leq k_i < 2^M$, $i = 1, 2, \dots, S$,

经过 M 级分解, S 维数据块排成一系列数据, 这列数据即为频域抽取 S 维向量基 FFT 流程图中的输出数据。当 $S = 3$, $M = 2$ 时, 输出序列 $X(k_1, k_2, k_3)$ 的一维排序如图3所示。

FFT 变换的运算量最小。与 (N, N, \dots, N) 点时域抽取 S 维向量基 FFT 算法^[11]相比, 二者的运算量相同。

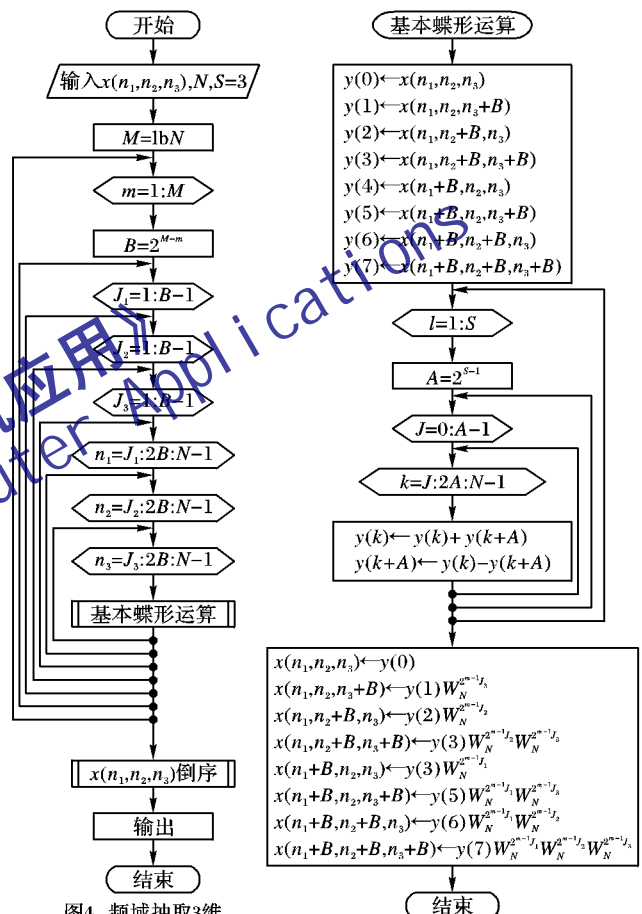


图4 频域抽取3维向量基FFT流程

图5 基本蝶形运算流程

表3 S 维信号各种离散傅里叶变换的计算量

方法	复数加法次数	复数乘法次数
直接计算 S 维 DFT	$N^S(N^S - 1)$	N^{2S}
S 维分离式 DFT	$SN^S(N - 1)$	SN^{S+1}
S 维分离式 FFT (每维采用频域抽取 1 维向量基 FFT)	$SN^S \lg N$	$(SN^S \lg N)/2$
频域抽取 S 维向量基 FFT	$SN^S \lg N$	$[(2^S - 1)N^S \lg N]/2^S$

4 结语

本文给出了频域抽取 S 维向量基 FFT 的理论推导, 并给出了编程实现的程序框图, 该框图可以推广到 S 为任意维的情况, 快速算法运算量低, 比 S 维分离式 FFT 更具高效性。在 S 维信号的频域分析、信号去噪、模式识别、数据压缩等信号

(下转第 2818 页)

易于实现。实验结果证明,本文算法提高了去噪图像的PSNR值,对于噪声强度具有很强的适应性,且较好地保持了图像的边与纹理的细节信息,可以适用于处理受不同程度椒盐脉冲噪声污染的灰度级图像。

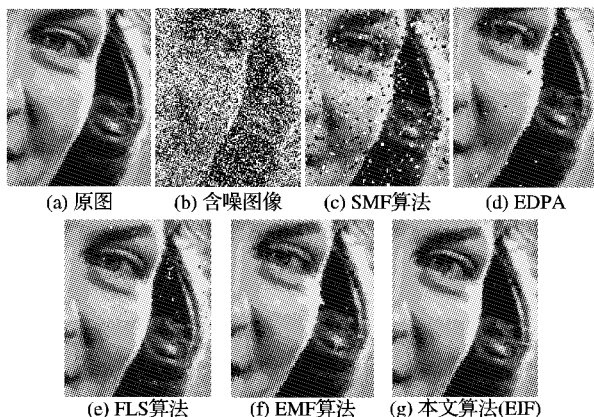


图3 噪声密度为0.5时的滤波效果比较

参考文献:

- [1] BAR L, KIRYATI N, SOCHEN N. Image deblurring in the presence of impulsive noise [J]. International Journal of Computer Vision, 2006, 70(3): 279–298.
- [2] KO S J, LEE Y H. Center weighted median filters and their applications to image enhancement [J]. IEEE Transactions on Circuits and Systems, 1991, 38(9): 984–993.
- [3] WANG Z, ZHANG D. Progressive switching median filter for the removal of impulse noise from highly corrupted images [J]. IEEE Transactions on Circuits and Systems, II: Analog and Digital Signal Processing, 1999, 46(1): 78–80.
- [4] ARAKAWA K. Median filters based on fuzzy rules and its applica-

tion to image restoration [J]. Fuzzy Sets and Systems, 1996, 77(1): 3–13.

- [5] CHEN T, MA K K, CHEN L H. Tri-state median filter for image denoising [J]. IEEE Transactions on Image Processing, 1999, 8(12): 1834–1838.
- [6] SRINIVASAN K S, EBENEZER D. A new fast and efficient decision-based algorithm for removal of high-density impulse noises [J]. IEEE Signal Processing Letters, 2007, 14(3): 189–192.
- [7] LUO WENBIN. Efficient removal of impulse noise from digital images [J]. IEEE Transactions on Consumer Electronics, 2006, 52(2): 523–527.
- [8] ZHANG XUMING, XIONG YOULUN. Impulse noise removal using directional difference based noise detector and adaptive weighted mean filter [J]. IEEE Signal Processing Letters, 2009, 16(4): 295–298.
- [9] AIZENBERG I, BUTAKOFF C. Effective impulse detector based on rank-order criteria [J]. IEEE Signal Processing Letters, 2004, 11(3): 363–366.
- [10] CHEN PEI-YIN, LIEN CHIH-YUAN. An efficient edge-preserving algorithm for removal of salt-and-pepper noise [J]. IEEE Signal Processing Letters, 2008, 15(2): 833–836.
- [11] LUO WENBIN. An efficient detail-preserving approach for removing impulse noise in images [J]. IEEE Signal Processing Letters, 2006, 13(7): 413–416.
- [12] 李岳阳, 王士同, 胡德文. 基于区间类型2模糊系统的高斯噪声检测滤波器[J]. 计算机研究与发展, 2004, 41(9): 1507–1513.
- [13] 曲延峰, 徐建, 李卫军, 等. 有效去除图像中脉冲噪声的新型滤波算法[J]. 计算机辅助设计与图形学学报, 2003, 15(4): 397–401.
- [14] 满利红, 王士同. 彩色图像脉冲噪声的模糊检测和滤波[J]. 计算机应用研究, 2009, 26(3): 1198–1201.

(上接第2780页)

处理领域有乐观的应用前景。

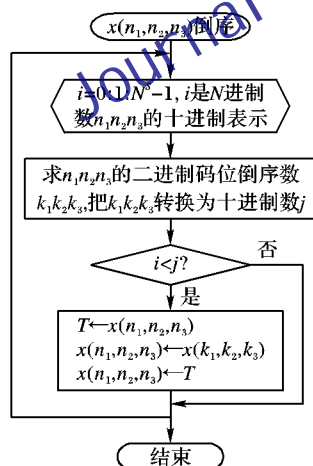


图6 $x(n_1, n_2, n_3)$ 倒序

参考文献:

- [1] COOLEY J W, TUKEY J W. An algorithm for the machine calculation of complex Fourier series [J]. Mathematics of Computation, 1965, 19(4): 297–301.
- [2] OPPENHEIM A V, SCHAFER R W, BUCK J R. Discrete-time signal processing [M]. 2nd ed. Beijing: Tsinghua University Press, 2005: 629–677.
- [3] DUHAMEL P, VETTERLI M. Fast Fourier transforms: A tutorial review and a state of the art [J]. Signal Processing, 1990, 19(4): 259–299.

- [4] HARRIS D B, MCCLELLAN J H, CHAN D S K, et al. Vector radix fast Fourier transform [C]// IEEE International Conference on Acoustics, Speech, and Signal Processing. Washington, DC: IEEE, 1977, 548–551.
- [5] HONG P P. Fast two-dimensional Fourier transform [C]// Proceedings of the Third Hawaii International Conference on System Science. Honolulu, Hawaii, USA: [s. n.], 1970, 990–993.
- [6] MOU Z J, DUHAMEL P. In-place butterfly-style FFT of 2-D real sequences [J]. IEEE Transactions on Acoustics Speech and Signal Processing, 1988, 36(10): 1642–1650.
- [7] WU H R, PAOLOI F J. On the two-dimensional vector split-radix FFT algorithm [J]. IEEE Transactions on Acoustics Speech and Signal Processing, 1989, 37(8): 1302–1304.
- [8] WU H R, PAOLONI F J. The structure of vector radix fast Fourier transform [J]. IEEE Transactions on Acoustics Speech and Signal Processing, 1989, 37(9): 1415–1424.
- [9] JOHNSON S G, FRIGO M. A modified split-radix FFT with fewer arithmetic operations [J]. IEEE Transactions on Signal Processing, 2007, 55(1): 111–119.
- [10] 徐妮妮, 吴云峰, 肖志涛. 频域抽取二维向量基快速傅里叶变换 [J]. 天津工业大学学报, 2008, 27(6): 47–50.
- [11] CHEN ZHAODOU, ZHANG LIJING. Vector coding algorithms for multidimensional discrete Fourier transform [J]. Journal of Computational and Applied Mathematics, 2008, 212(1): 63–74.
- [12] 吴云峰, 徐妮妮, 肖志涛, 等. 三维向量基快速傅里叶算法 [J]. 计算机应用, 2009, 29(2): 618–621.