各类离散 W 变换的快速 Hartley 变换算法

钟广军,成礼智,陈火旺
(国防科技大学,长沙 410073)

摘　要: 本文通过建立各类 N 阶离散 W 变换(DWTs)到 N 阶离散 Hartley 变换(DHT)的转换,得到了一种利用 DHT 统一计算各类 DWTs 的非常简单的快速算法. 该算法结构简单,且每一种转换过程总的运算量均低于 $5N$.

关键词: 离散 Hartley 变换; 离散 W 变换; 快速算法
中图分类号: TN917　**文献标识码**: A　**文章编号**: 0372-2112 (2001) 02-0190-02

Fast Hartley Transform Algorithm for Various Discrete W Transforms

ZHONG Guang-jun, CHENG Li-zhi, CHEN Huo-wang
(*National University of Defense Technology, Changsha 410073, China*)

Abstract: A simple fast algorithm by DHT to utilize various DWTs, based on structuring a transform between N-order DWTs and N-order DHT, is proposed here. The algorithm is simple in computational structure and the total number of operations for each transform is less than $5N$.

Key words: discrete Hartley transform; discrete W transform; fast algorithm

1　引言

作为离散 Hartley 变换(DHT) 的推广,由 Z. Wang[1]提出的各类离散 W 变换(DWTs)已成为信号与图像处理中一种较为有用的工具. 人们对于各种离散 W 变换提出了许多快速计算方法[1~4]. 它们大致上可以分为两大类:一类是先作分解再利用其它离散变换计算 DWT,如文[1]用多个离散余弦变换(DCT)计算 DWT 而文[3]则用快速多项式变换(FPT)计算 DWT-II,-III;另一类则对 DWT 作全过程的直接分解,例如文[5]. 上述两类算法具有下列特点:(1) DWT 的长度必须为 2 的幂;(2) 全过程直接分解算法的算法结构比较复杂,不便于程序实现,例如:利用文[5]中算法计算 DWT-IV 时需将 N 点 DWT-IV 分解为一个 $N/2$ 点的 DWT-II 与一个 $N/2$ 点的 DWT-III,而 $N/2$ 点的 DWT-II,或 - III 再分解为 $N/4$ 的 DWT-III 与 DWT-IV;不难看出,直接分解算法在计算某一类 DWT 时需要其它类 DWT,且这种需要是递归式的,这给算法以及程序实现的优化带来不少困难. 而另一方面,被称之为实数域上的 DFT 的离散 Hartley 变换(DHT)在各种长度上都存在许多高效且算法结构优良的快速算法,如文[6]. 基于上述讨论,本文建立了用 DHT 计算等长各类 DWTs 的快速算法.

2　算法推导

为了便于算法描述,首先回顾四类离散 W 变换的定义.

定义 1　实序列$\{x(n)\}$的离散 W 变换定义为

$$X_I(k)=\sum_{n=0}^{N-1}x(n)\,\mathrm{cas}\frac{2nk\pi}{N},\quad X_{II}(k)=\sum_{n=0}^{N-1}x(n)\,\mathrm{cas}\frac{n(2k+1)\pi}{N}\tag{1}$$

$$X_{III}(k)=\sum_{n=0}^{N-1}x(n)\,\mathrm{cas}\frac{(2n+1)k\pi}{N},\quad X_{IV}(k)=\sum_{n=0}^{N-1}x(n)\,\mathrm{cas}\frac{(2n+1)(2k+1)\pi}{2N}\tag{2}$$

其中,$\mathrm{cas}(z)=\cos(z)+\sin(z)$, $k=0,1,\cdots,N-1$,下标 I,II,III 与 IV 分别表示第一,第二,第三与第四类离散 W 变换并分别记为 DWT-I,-II,-III 与-IV. 显然,DWT-I 正是在信号处理中有广泛应用的离散 Hartley 变换(DHT)[5,6]. 为了与已有的常见记号相一致,本文仍将 DWT-I 称之为 DHT.

算法的推导基于下面的三角恒等式:

$$2\mathrm{cas}\left(n(2k+1)\frac{\pi}{N}\right)\cos\frac{n\pi}{N}=\mathrm{cas}\left(n(k+1)\frac{2\pi}{N}\right)+\mathrm{cas}\left(2nk\frac{\pi}{N}\right)\tag{3}$$

$$2\mathrm{cas}\left((2n+1)(2k+1)\frac{\pi}{N}\right)\sin\frac{(2n+1)\pi}{2N}=\mathrm{cas}\frac{(2n+1)(N-k-1)\pi}{N}-\mathrm{cas}\frac{(2n+1)(N-k)\pi}{N}\tag{4}$$

2.1　N 为奇数时的算法推导

直接应用式(1)与式(3),可以得到

$$X_{II}(k)=\sum_{n=0}^{N-1}\frac{x(n)}{2\cos(n\pi/N)}\left[\mathrm{cas}\frac{2n(k+1)\pi}{N}+\mathrm{cas}\frac{2nk\pi}{N}\right]\tag{5}$$

如果定义 $y_1(n)=\dfrac{x(n)}{2\cos(n\pi/N)}$, $0\le n\le N-1$ 并称 $Y_I^{(1)}(k)$为序列 $y_1(n)$的 DHT,则式(5)可以通过下列 DHT 来计算

$$X_{II}(k)=Y_I^{(1)}(k)+Y_I^{(1)}(k+1),\quad k=0,1,\cdots,N-1\tag{6}$$

其中 $Y_I^{(1)}(0)=Y_I^{(1)}(N)$.

收稿日期:2000-03-13;修回日期:2000-07-05

类似地,如果定义 $y_2(0) = x(0) + x(N-1)$, $y_2(n) = x(n-1) + x(n)$, $n = 1,2,\cdots,N-1$ 并称 $Y_I^{(2)}(k)$ 为序列 $y_2(n)$ 的 DHT,则序列 $x(n)$ 的 DWT-III $X_{III}(k)$ 可以表示为

$$X_{III}(k) = \sum_{n=0}^{N-1} x(n)\,\mathrm{cas}\frac{(2n+1)k\pi}{N} = \frac{1}{2\cos(k\pi/N)}\cdot\left[\sum_{n=0}^{N-1} x(n)\left(\mathrm{cas}\frac{(n+1)2k\pi}{N} + \mathrm{cas}\frac{2nk\pi}{N}\right)\right] = Y_I^{(2)}(k)/(2\cos(k\pi/N)),\quad k = 0,1,\cdots,N-1 \tag{7}$$

为了计算式(2)的 $X_{IV}(k)$,还需要引入下列序列

$$x_2(n) = \frac{x(n)}{2\sin[(2n+1)\pi/2N]},\ y_3(0) = x_2(0) + x_2(N-1),$$

$$y_3(n) = x_2(n-1) + x_2(n),\quad n = 1,2,\cdots,N-1$$

如果用 $Y_{III}(k)$ 表示序列 $x_2(n)$ 的 DWT-III,利用式(2)与(4)可以将序列 $x(n)$ 的 DWT-IV $X_{IV}(k)$ 表示为

$$X_{IV}(k) = \sum_{n=0}^{N-1} x_2(n)\left[\mathrm{cas}\frac{(2n+1)(N-k-1)\pi}{N} - \mathrm{cas}\frac{(2n+1)(N-k)\pi}{N}\right] = Y_{III}(N-k-1) - Y_{III}(N-k),\quad k = 0,1,\cdots,N-1 \tag{8}$$

其中,$Y_{III}(N) = Y_{III}(0)$.

又若定义 $Y_I^{(3)}(k)$ 为序列 $y_3(n)$ 的 DHT,实施类似于式(7)中所作的运算,可以得到 $Y_{III}(k) = Y_I^{(3)}(k)/(2\cos(k\pi/N))$. 将上式代入式(8)之右端,可以得到

$$X_{IV}(k) = \frac{Y_I^{(3)}(N-k)}{2\cos(k\pi/N)} - \frac{Y_I^{(3)}(N-k-1)}{2\cos[(k+1)\pi/N]},\ k = 0,1,\cdots N-1 \tag{9}$$

其中,$Y_I^{(3)}(N) = Y_I^{(3)}(0)$.

当变换长度 N 为奇数时,式(6),(7)与(9)分别给出了 DWT-II,-III 以及 DWT-IV 的 DHT 算法.

2.2　N 为偶数时的算法推导

由于当 N 为偶数时有 $\cos(k\pi/N) = 0$ 对于 $k = N/2$ 成立,为了得到各类 DWTs 的 DHT 算法,需要对上述 DWT 算法作适当修改. 此时定义序列 $y_1'(n)$ 满足:当 $n \neq N/2$ 时,$y_1'(n) = y_1(n)$,而 $n = N/2$ 时,$y_1'(n) = 0$. 并记 $Y_I^{(1*)}(k)$ 为序列 $y_1'(n)$ 的 DHT,则式(6)可以用下式来代替

$$X_{II}(k) = Y_I^{(1*)}(k) + Y_I^{(1*)}(k+1) + (-1)^k x(N/2),\quad k = 0,1,\cdots,N-1 \tag{10}$$

而式(7)则可以按下式另外给出

$$X_{III}\left(\frac{N}{2}\right) = \sum_{n=0}^{N-1}(-1)^n x(n),$$

$$X_{III}(k) = \frac{Y_I^{(2)}(k)}{2(\cos k\pi/N)},\ 0 \le k \le N-1,\ k \neq N/2. \tag{11}$$

至于式(2)中定义的 DWT-IV 的计算,利用式(8)以及 $k = N/2$ 时有 $\cos(k\pi/N) = 0$ 等结论,可以得到下列计算公式

$$Y_{III}\left(\frac{N}{2}\right) = \sum_{n=0}^{N-1}(-1)^n x_2(n),$$

$$Y_{III}(k) = \frac{Y_I^{(3)}(k)}{2\cos(k\pi/N)},\ 0 \le k \le N-1,\ k \neq \frac{N}{2} \tag{12}$$

$$Y_{III}(N) = Y_{III}(0),\ X_{IV}(k) = Y_{III}(N-k-1) - Y_{III}(N-k),\quad k = 0,1,\cdots,N-1 \tag{13}$$

当变换长度 N 为偶数时,式(10),(11)以及(12),(13)分别给出了 DWT-II,-III,以及 DWT-IV 的 DHT 算法.

3　计算复杂性讨论

设长度为 N 的 DWT-L ($L = I, II, III, IV$) 所需乘法与加法次数分别记为 $M[L(N)]$ 与 $A[L(N)]$. 由式(6)～(13)得到

$$M[II(N)] = M[I(N)] + FM_2(N),\quad A[II(N)] = A[I(N)] + FA_2(N); \tag{14}$$

$$M[III(N)] = M[I(N)] + FM_3(N),\quad A[III(N)] = A[I(N)] + FA_3(N); \tag{15}$$

$$M[IV(N)] = M[I(N)] + FM_4(N),\quad A[IV(N)] = A[I(N)] + FA_4(N); \tag{16}$$

其中,当变换长度 N 为奇数时,附加运算量满足:$FM_2(N) = FM_3(N) = N-1$, $FM_4(N) = 2N$, $AM_2(N) = AM_3(N) = N$, $AM_4(N) = 2N$. 而当 N 为偶数时,其附加运算量满足:$FM_2(N) = FM_3(N) = N-2$, $FM_4(N) = 2N-1$, $AM_2(N) = 2N$, $AM_3(N) = 2N-1$, $AM_4(N) = 3N-1$. 这表明,一个长 N 的 DWT-II 或 DWT-III 可以通过一个长 N 的 DHT 及最多不超过 $N-2$ 次乘法,$2N-1$ 次加法得到,而长 N 的 DWT-IV 可以通过一个长 N 的 DHT 及最多不超过 $2N-1$ 次乘法,$3N-1$ 次加法得到.

4　结论

通过提出一种从长为 N 的各类 DWTs 到长为 N 的 DHT 的简单转换,得到了一种基于 DHT 的各类 DWTs 快速算法. 从式(14)～(16)可以看到,转换过程中的附加运算总数均低于 $5N$,因此计算 DWT 的主要运算是由 DHT 的运算量决定的. 由于对各种不同长度的 DHT,均存在高效率的快速 DHT 算法,因此利用本文提供的方法可以建立各类 DWTs 的快速算法.

参考文献:

[1] Z. Wang. Fast algorithms for the discrete W transform and for the discrete cosine transform [J]. IEEE Trans., 1984, ASSP-32(4): 803-816.

[2] 王中德. 快速 W 变换 ——算法与程序 [J]. 中国科学(A), 1988, (5): 549-560.

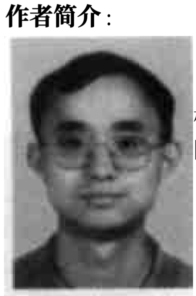
[3] 曾泳泓,蒋增荣. 二维离散 W 变换的多项式变换算法 [J]. 电子学报, 1997, 25(8): 63-66.

[4] 成礼智. 用递归和 W 变换计算一维循环卷积 [J]. 通信学报, 1996, 17(3): 47-50.

[5] G. Bi. On computation of the discrete W transform [J]. IEEE Trans. 1999, SP-47(5): 1450-1453.

[6] R. Baghaie, V. Dimitrov. DHT algorithm based on encoding algebraic integers [J]. Electron Letters, 1999, 35(16): 1303-1304.

作者简介:



钟广军　1974 年生. 现在国防科技大学计算机学院攻读博士学位. 目前感兴趣的研究领域有图像处理,形式化软件开发方法.