

Walsh 矩阵的复制生成及其计算机图像*

陈宁涛¹⁺, 王能超², 施保昌²

¹(华中科技大学 计算机科学与技术学院,湖北 武汉 430074)

²(华中科技大学 并行计算研究所,湖北 武汉 430074)

Walsh Matrix's Copy Generation and Its Computer Images

CHEN Ning-Tao¹⁺, WANG Neng-Chao², SHI Bao-Chang²

¹(School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China)

²(Parallel Computing Institute, Huazhong University of Science and Technology, Wuhan 430074, China)

+ Corresponding author: Phn: +86-27-87543531, E-mail: nt_chen@263.net, http://www.hust.edu.cn

Chen NT, Wang NC, Shi BC. Walsh matrix's copy generation and its computer images. *Journal of Software*, 2006,17(5):1241-1250. <http://www.jos.org.cn/1000-9825/17/1241.htm>

Abstract: Walsh functions are widely used in many areas such as signal and image processing, digital communication etc. Walsh function system is orthogonal and completed. Among many methods to generate Walsh functions, Swick's copy theory is famous. This method takes Walsh function's order numbers as the copy information and can generate an arbitrary function given certain order number, which in fact is the vector processing method and is not applicable for such 2D signal processing applications as fast transforms. Walsh function system can be expressed by Walsh matrix W_k . In this paper, row copy and block copy methods are put forward based on W_k . Copy operators are designed based on symmetries and a new ordering is discovered (named Walsh-Like ordering). After the iterative formulas of 6 ordering Walsh matrices are deduced by Kronecker product, the computer images of these matrices are illustrated. It is proved that the proposed method is more advanced than the former. The later can achieve higher performance and is applicable to the fast transforms designing. The fourth symmetric ordering of Walsh function system is discovered (Walsh-Like ordering), and the conjecture that the new ordering is likely to be the reverse form of Walsh ordering is made by analyzing and comparing with these images.

Key words: Walsh function; copy; evolution; symmetry; fractal

摘要: Walsh 函数在信号处理、图像处理、通信等众多领域有着广泛的应用。Walsh 函数系是一个正交而完备的函数系,可以通过多种方法生成这一函数系。其中,Swick 提出的复制方法应用最为广泛,该方法以 Walsh 函数的序码作为复制信息,可以复制出任意给定序数的 Walsh 函数。其本质是基于向量的处理,不适于类似快速变换等二维信号的处理。Walsh 函数系可用 Walsh 方阵 W_k 表示。提出了基于 W_k 的行复制和块复制方法。基于对称性引入复制算子,并发现了一种新序(类 Walsh 序)。利用 Kronecker 积推导了 6 种序的 Walsh 方阵的递推公式并绘制了它们的计算机图像,发现这些图像具有分形意义上的自相似结构。结果表明,基于矩阵的复制是比基于序码

* 本文为 2005 年中国计算机大会推荐优秀论文。Supported by the National Natural Science Foundation of China under Grant No.60473015 (国家自然科学基金)

Received 2005-06-15; Accepted 2006-01-18

的复制更先进的复制方法.前者性能更优,适于快速变换的设计.而且,利用它发现了 Walsh 函数系的第 4 种对称的序:类 Walsh 序.通过分析和比较各种序的计算机图像,得出类 Walsh 序更适合作为 Walsh 序的逆反形式的猜想.

关键词: Walsh 函数;复制;演化;对称;分形

中图法分类号: TP391 文献标识码: A

Walsh 函数在信号处理、图像处理、通信等众多领域有着广泛的应用. Walsh 函数系是一个在二分集 E_k 的每个子段上取值 ± 1 的正交函数系. 在每个二分集 E_k 上, Walsh 函数的值组成一个 2^k 阶方阵 W_k , 称为 Walsh 方阵. 这样, Walsh 函数的演化过程可表述为 Walsh 方阵 W_k 的加工过程, 而加工手续则是某种对称性复制^[1]. Walsh 函数复制理论的奠基人 Swick^[2]于 1969 年提出用镜像复制方法产生 Walsh 序的 Walsh 函数. 张其善^[3]扩展了这一理论, 推导了 Paley 序和 Hadamard 序的 Walsh 函数, 还根据这种复制方法的逻辑关系发现了 Walsh 函数的一种新的排序方式, 并命名为 X 序. Swick 与张其善所归纳出的复制方法不需要从低序号到高序号递推, 以 Walsh 函数的序号作为复制信息, 因而运用这种方法可以复制出任意给定序数的 Walsh 函数. 该方法的优点是易于在计算机上实现, 也易于硬件实现. 而本文介绍的复制方法是与上述复制方法完全不同的一种技术. 该方法以矩阵为考察对象, 以二分技术^[4]为手段, 以演化法则^[5]为基础, 从对称性角度分析了 6 种序的 Walsh 函数的复制过程.

历史上, Walsh 函数有 3 种序: Paley 序、Hadamard 序和 Walsh 序. 其中: Paley 序和 Hadamard 序分别由自然码与反自然码生成, 因此可以说 Paley 序和 Hadamard 序是互反的; Walsh 序由 Gray 码生成. 而对 Walsh 序的逆反形式的研究, 目前还没有一个统一的答案, 这使得对 Walsh 函数系统的研究不是那么对称、和谐. 文献[6]分析了 Gray 码的两种逆反形式: 反 Gray 码和逆 Gray 码. 反 Gray 码生成反 Walsh 序的 Walsh 函数; 逆 Gray 码生成逆 Walsh 序的 Walsh 函数, 而逆 Gray 码其实就是反自然码的 Gray 码^[7](这种码生成 X 序). 因此, X 序即为逆 Walsh 序. 相应地, 本文称反 Walsh 序为 X^T 序. 文献[1]中所谓的反 Walsh 序其实是逆 Walsh 序, 这是一个名字上的错误. 文献[8,9]详细分析了这两种序及其快速变换算法. 本文从对称性出发分析了各种序的复制关系, 并导出了另一种序, 暂且称其为类 Walsh 序. 最后, 绘制了 Walsh 方阵的计算机图像, 以从宏观上把握各种 Walsh 方阵的特点.

本文第 1 节介绍 Walsh 函数系的定义、基于序号的复制技术、对称性定义以及基于矩阵的行复制和块复制技术. 第 2 节从对称性出发设计 4 种复制算子, 复制生成 6 种序的 Walsh 方阵. 第 3 节给出 3 种复制算法的 C 语言描述和性能测试实验, 绘制 6 种序的 Walsh 方阵的计算机图像. 最后是全文的结论.

1 相关理论

1.1 Walsh 函数系定义

1923 年, 美国数学家 J.L. Walsh 提出的 Walsh 函数系定义如下:

$$W_i(x) = \prod_{r=0}^{k-1} \text{sgn}[\cos j_r 2^r \pi x], 0 \leq x < 1, i = 0, 1, 2, \dots, 2^k - 1, j_r = \sum_{r=0}^{k-1} j_r 2^r.$$

其中: k 表示层数, k 层 Walsh 函数系含有 2^k 个 Walsh 函数; sgn 是符号函数, $\text{sgn}[x]$ 当 $x > 0$ 时取值 $+1$, $x < 0$ 时取值 -1 ; 又 j_r 取值 0 或 1 是序号 j 的二进制码, j 为序号 i 的某种编码. j 的不同编码将产生不同序的 Walsh 函数系. 目前各种序的 Walsh 函数系定义如下:

- (1) 如果 j 取自然码, 即 $j_{k-r-1} = i_{k-r-1}, r = 0 \sim k-1$, 则得到 Paley 序的 Walsh 函数.
- (2) 如果 j 取 Gray 码, 即 $j_{k-1} = i_{k-1}, j_{k-r-1} = i_{k-r-1} \oplus i_{k-r}, r = 0 \sim k-1$, 则得到 Walsh 序的 Walsh 函数. J.L. Walsh 最早提出的 Walsh 函数正是这种排序方式.
- (3) 如果 j 取反自然码, 即 $j_{k-r-1} = i_r, r = 0 \sim k-1$, 则得到反 Paley 序的 Walsh 函数. 由于这种排序方式和 Hadamard 矩阵有紧密的关系, 故也称为 Hadamard 序.
- (4) 如果 j 取反 Gray 码, 即是先取序号 i 的 Gray 码, 然后取该结果的反自然码得到的关于 i 的编码, 即 $j_0 = i_{k-1}, j_r = i_{k-r} \oplus i_{k-r-1}, r = 1 \sim k-1$, 则得到反 Walsh 序的 Walsh 函数系.

(5) 如果 j 取逆 Gray 码,即是先取序号 i 的反自然码,然后取该结果的 Gray 码得到的关于 i 的编码,即 $j_{k-1}=i_0, j_{k-r-1}=i_{\oplus r-1}, r=1 \sim k-1$,则得到逆 Walsh 序的 Walsh 函数系,又称为 X 序.

图 1 列举了 $k=3$ 时 Walsh 序的 8 个 Walsh 函数的波形.此时 j 取 Gray 码.中心轴上方表示取值+1,下方表示取值为-1.可以看出,Walsh 函数尽管取值简单,但其波形却很复杂,似乎比三角函数复杂得多,以至于根据定义很难作出它们的图形.Swick 的序码复制技术较好地解决了这一问题.

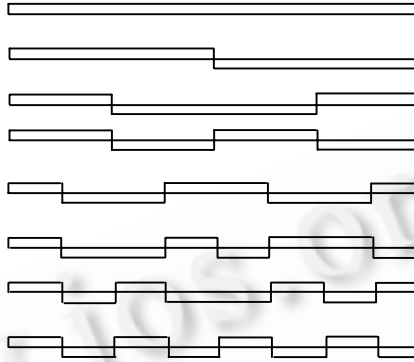


Fig.1 Eight Walsh ordering functions

图 1 Walsh 序的 8 个 Walsh 函数

1.2 基于序码的复制技术

1969 年,D.A. Swick 提出了运用镜像复制技术逐个生成 Walsh 序的 Walsh 函数的方法,揭开了利用复制技术生成 Walsh 函数的序幕.1983 年,张其善、李植华将 Swick 的复制技术(包括平移复制技术)推广到其他序的 Walsh 函数.该方法可以不借助其他函数,直接从 Walsh 函数的序号 i 的编码出发,逐个复制,生成任意给定序的 Walsh 函数.表 1 利用序码复制技术生成 Walsh 序的 8 个 Walsh 函数,其中复制信息采用 j 的 Gray 码,复制技术采用平移复制(“+,-”分别表示“+1,-1”).

Table 1 Order code copy technique

表 1 序码复制技术

Order number	Gray code ($j_2j_1j_0$)	Initial	Step 1 (j_2)	Step 2 (j_1)	Step 3 (j_0)
0	000	+	+	++	++++
1	001	+	+	++	----
2	011	+	+	--	--++
3	010	+	+	--	+-+-
4	110	+	-	-+	+--+
5	111	+	-	-+	-+-+
6	101	+	-	+-	-+++
7	100	+	-	+-	+--+

1.3 Walsh 方阵

Walsh 函数虽然只取 ± 1 两个值,但其取值方式却很复杂.Walsh 函数频繁地在这两个值之间跃变,因而运用传统的数学方法研究 Walsh 函数是困难的.为了便于描述 Walsh 函数的跃变特征,文献[1]引进二分集的概念.

设将区间 $E_0=[0,1]$ 对半二分,其左右两个子段合成为集 E_1 :

$$E_1=E_1^{(0)} \cup E_1^{(1)}, E_1^{(0)}=[0,1/2], E_1^{(1)}=[1/2,1],$$

再将 E_1 的每个子段对半二分,又得到区间集 E_2 :

$$E_2=[0,1/4] \cup [1/4,1/2] \cup [1/2,3/4] \cup [3/4,1],$$

如此二分下去,二分 k 次所得出的 E_k 包含 2^k 个子段(如图 2 所示).

在二分集的每个子段上取定值的函数称作二分集上的阶跃函数.阶跃函数在某一子段上的函数值称作该

子段的阶跃值.不言而喻,二分集 E_k 上任一函数的阶跃值构成一个 2^k 维向量,而 E_k 上含有 2^k 个正交函数的阶跃函数组,对应于某个 2^k 阶方阵 W_k ,称为 Walsh 方阵.

$$E_k = \bigcup_{i=0}^{2^k-1} E_k^{(i)}, E_k^{(i)} = \left[\frac{i}{2^k}, \frac{i+1}{2^k} \right)$$

Fig.2 Bisection sets

图 2 二分集

1.4 对称性

文献[5]中 Walsh 序 Walsh 函数的生成引入了奇对称与偶对称两个概念; Paley 序 Walsh 函数的生成引入了平移对称与反射对称两个概念.为了区别它们,我们把对称分为两种:镜像对称与平移对称.奇对称与偶对称属于镜像对称,平移对称与反射对称属于平移对称.镜像对称和平移对称的定义如下:

定义 1(镜像对称). 称函数 $f(x)$ 在时基 $[0,1)$ 左右两侧呈偶对称,如果 $f(1/2-x)=f(1/2+x)$, $0 \leq x < 1/2$; 反之,如果 $f(1/2-x)=-f(1/2+x)$, $0 \leq x < 1/2$, 则称函数 $f(x)$ 的左右两侧呈奇对称.

定义 2(平移对称). 称函数 $f(x)$ 在时基 $[0,1)$ 左右两侧呈正对称,如果其右侧是左侧的平移,即有 $f(x)=f(x-1/2)$, $1/2 \leq x < 1$; 反之,如果其右侧是左侧的反射平移,即 $f(x)=-f(x-1/2)$ 成立, $1/2 \leq x < 1$, 则称函数 $f(x)$ 的左右两侧呈反对称.

1.5 基于矩阵的复制技术

基于镜像对称性与平移对称性,文献[1]引入了奇偶复制和平移复制的概念.其中, Walsh 序与逆 Walsh 序通过奇偶复制生成, Paley 序与反 Paley 序通过平移复制生成.考察演化法则的合成手续^[1](0-1 法则),其中逆 Walsh 序与反 Paley 序通过前后接排合成; Walsh 序与 Paley 序通过奇偶混排合成.在前后接排手续下, W_k 的生成是以整块 W_{k-1} 作为复制对象,不妨称其为“块复制”.在文献[5]的图 12 与图 13 中, Hadamard 序 Walsh 函数的生成原理及生成过程体现了块复制思想.在奇偶混排手续下, W_k 的生成是以 W_{k-1} 的每一行作为复制对象,不妨称之为“行复制”.文献[1]中, Walsh 序与 Paley 序方阵的生成体现了行复制思想.由块复制与行复制的特征可知,从 W_{k-1} 到 W_k 矩阵的行数与列数都加倍.因此, W_k 的元素数是 W_{k-1} 的 4 倍.

注意奇偶复制、平移复制的概念和块复制、行复制的概念的区别,两者是从不同角度引入的复制技术.根据文献[1,5]中的演化法则,奇偶复制、平移复制是针对分解法则的(0 法则或 1 法则),而块复制、行复制是针对合成法则的(0-1 法则).必须强调指出的是,文献[5]中的方法完全是一个几何方法,是对波形的操作,对于类似变换算法的数字化的运算并不适合.文献[1]在文献[5]的基础上提出了代数方法,是对矩阵进行操作,适合于数字化运算.容易看出:若合成法则为块复制,则块与块之间可以施行奇偶复制或平移复制或两者混合;若合成法则为行复制,则每一行只能施行奇偶复制或平移复制其中之一.已经知道,按照行复制+奇偶复制产生 Walsh 序方阵;按照行复制+平移复制产生 Paley 序方阵;按照块复制+平移复制产生反 Paley 序方阵;按照块复制+矩阵块上下平移复制、左右奇偶复制产生逆 Walsh 序方阵.根据排列组合理论,关于块复制尚有两种组合方式:块复制+奇偶复制与块复制+矩阵块左右平移复制、上下奇偶复制.前者产生一种新序,暂且称为类 Walsh 序;关于后者,文献[6]根据反 Gray 码按照 Swick 的复制方法可以生成反 Walsh 序.至此, Walsh 序的逆反形式有 3 种:逆 Walsh 序、反 Walsh 序和类 Walsh 序.它们从不同角度反映了 Walsh 序的反,但哪种序是 Walsh 序的反呢?如果从 Paley 序与反 Paley 序的关系来看,好像类 Walsh 序“更像” Walsh 序的反,但这仅仅是一个猜想.

2 Walsh 方阵的复制生成

2.1 复制算子

文献[1,8]采用 Kronecker 积推导低阶 Walsh 阵 W_{k-1} 到高阶 Walsh 阵 W_k 的生成过程.文献[1]引入了平移复制算子 H 和奇偶复制算子 G 如下:

$$H = \begin{bmatrix} e & e \\ e & -e \end{bmatrix}, G = \begin{bmatrix} e & \bar{e} \\ e & \bar{e} \end{bmatrix}.$$

H, G 分别包含 4 个元素,每个元素表示一种复制符, $e, -e, \bar{e}$ 和 \bar{e} 分别称为平移复制符、反平移复制符、偶复制符和奇复制符.设 $\bar{\otimes}$ 与 \otimes 分别代表行复制与块复制的 Kronecker 积运算符,对于任意矩阵 $A, H\bar{\otimes}A$ 与 $H\otimes A$ 表示矩阵 A 在 H 的作用下分别进行了一次行复制与块复制操作.设矩阵 A 为 $(m+1)\times(n+1)$ 的矩阵,则

$$H\bar{\otimes}A = \begin{bmatrix} A_{00} & \dots & A_{0n} & A_{00} & \dots & A_{0n} \\ A_{00} & \dots & A_{0n} & -A_{00} & \dots & -A_{0n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{m0} & \dots & A_{mn} & A_{m0} & \dots & A_{mn} \\ A_{m0} & \dots & A_{mn} & -A_{m0} & \dots & -A_{mn} \end{bmatrix}, G\bar{\otimes}A = \begin{bmatrix} A_{00} & \dots & A_{0n} & A_{0n} & \dots & A_{00} \\ A_{00} & \dots & A_{0n} & -A_{0n} & \dots & -A_{00} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{m0} & \dots & A_{mn} & A_{mn} & \dots & A_{m0} \\ A_{m0} & \dots & A_{mn} & -A_{mn} & \dots & -A_{m0} \end{bmatrix},$$

$$H\otimes A = \begin{bmatrix} A_{00} & \dots & A_{0n} & A_{00} & \dots & A_{0n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{m0} & \dots & A_{mn} & A_{m0} & \dots & A_{mn} \\ A_{00} & \dots & A_{0n} & -A_{00} & \dots & -A_{0n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{m0} & \dots & A_{mn} & -A_{m0} & \dots & -A_{mn} \end{bmatrix}, G\otimes A = \begin{bmatrix} A_{00} & \dots & A_{0n} & A_{0n} & \dots & A_{00} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{m0} & \dots & A_{mn} & A_{mn} & \dots & A_{m0} \\ A_{00} & \dots & A_{0n} & -A_{0n} & \dots & -A_{00} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{m0} & \dots & A_{mn} & -A_{mn} & \dots & -A_{m0} \end{bmatrix}.$$

观察奇偶复制算子 G :对于块复制来说, G 的上下为平移复制,左右为奇偶复制.由前述可知,利用 G 可以复制生成逆 Walsh 阵.但对于反 Walsh 阵的生成, G 就不能胜任,需引入新的复制算子.同理,类 Walsh 阵的生成也需引入新的复制算子.设生成反 Walsh 阵的复制算子为 G' ,生成类 Walsh 阵的复制算子为 G'' .根据 G ,很容易推导出 G' 与 G'' 的矩阵形式如下:

$$G' = \begin{bmatrix} e & e \\ e & \bar{e} \end{bmatrix}, G'' = \begin{bmatrix} e & \bar{e} \\ e & \bar{e} \end{bmatrix}.$$

其中, e, \bar{e} 和 \bar{e} 分别称为上下偶复制符、上下奇复制符和上下左右奇复制符.这样,

$$G'\otimes A = \begin{bmatrix} A_{00} & \dots & A_{0n} & A_{00} & \dots & A_{0n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{m0} & \dots & A_{mn} & A_{m0} & \dots & A_{mn} \\ A_{m0} & \dots & A_{mn} & -A_{m0} & \dots & -A_{mn} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{00} & \dots & A_{0n} & -A_{00} & \dots & -A_{0n} \end{bmatrix}, G''\otimes A = \begin{bmatrix} A_{00} & \dots & A_{0n} & A_{0n} & \dots & A_{00} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{m0} & \dots & A_{mn} & A_{mn} & \dots & A_{m0} \\ A_{m0} & \dots & A_{mn} & -A_{mn} & \dots & -A_{m0} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ A_{00} & \dots & A_{0n} & -A_{0n} & \dots & -A_{00} \end{bmatrix}.$$

复制算子的引入为 Walsh 方阵的复制生成奠定了基础.设 $H^{\otimes k}$ 表示 k 个算子 H 按 Kronecker 积累乘,下面分别介绍 6 种序 Walsh 方阵的递推公式.

2.2 6种序Walsh方阵的复制生成

(1) 反 Paley 阵:块复制+平移复制产生反 Paley 阵. $W_0 \rightarrow W_k$ 的演化步可表述为如下递推关系式: $W_k = H\otimes W_{k-1} = H^{\otimes 2}\otimes W_{k-2} = \dots = H^{\otimes k}\otimes W_0$.其中, $W_0 = [+]$,表示初始矩阵,此时只有一个元素+.此公式表明,从 W_0 出发,运用复制算子 H 对矩阵 W_0 运算 Kronecker 积 k 次,就可以得到反 Paley 阵 W_k .反 Paley 阵 W_3 的复制生成如

图 3(a)所示,以下类推.

- (2) 逆 Walsh 阵: $W_k = G \otimes W_{k-1} = G^{\otimes 2} \otimes W_{k-2} = \dots = G^{\otimes k} \otimes W_0$, 如图 3(b)所示.
- (3) 反 Walsh 阵: $W_k = G' \otimes W_{k-1} = G'^{\otimes 2} \otimes W_{k-2} = \dots = G'^{\otimes k} \otimes W_0$, 如图 3(c)所示.
- (4) 类 Walsh 阵: $W_k = G^n \otimes W_{k-1} = G^{n \otimes 2} \otimes W_{k-2} = \dots = G^{n \otimes k} \otimes W_0$, 如图 3(d)所示.
- (5) Walsh 阵: $W_k = G \bar{\otimes} W_{k-1} = G^{\bar{\otimes} 2} \bar{\otimes} W_{k-2} = \dots = G^{\bar{\otimes} k} \bar{\otimes} W_0$, 如图 3(e)所示.
- (6) Paley 阵: $W_k = H \bar{\otimes} W_{k-1} = H^{\bar{\otimes} 2} \bar{\otimes} W_{k-2} = \dots = H^{\bar{\otimes} k} \bar{\otimes} W_0$, 如图 3(f)所示.

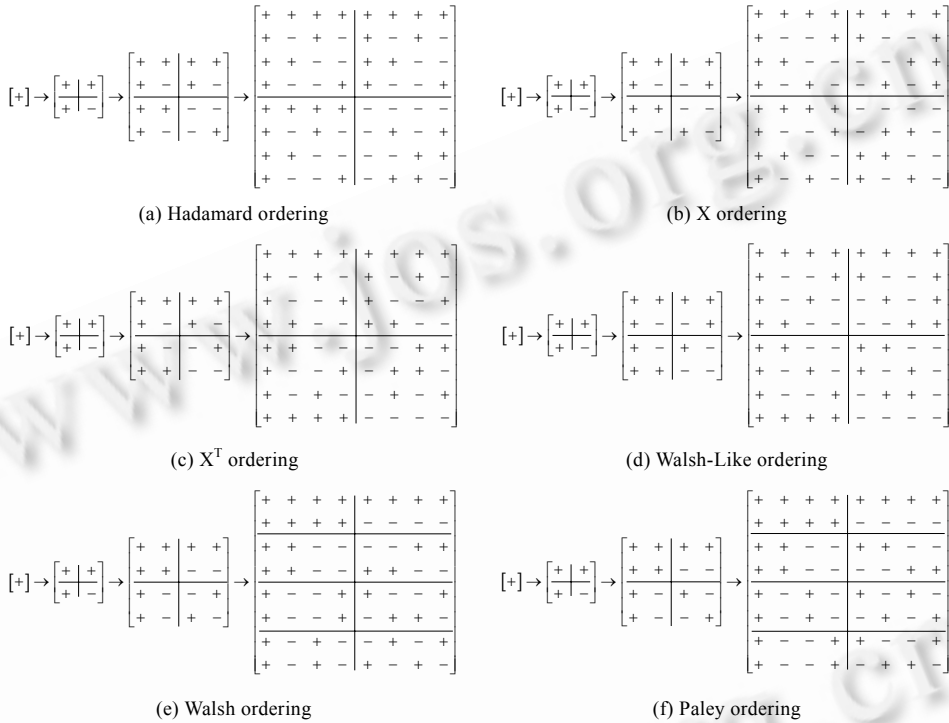


Fig.3 Copy generation of six orderings of Walsh matrix

图 3 6 种序 Walsh 矩阵的复制生成

3 Walsh 方阵的计算机图像

3.1 矩阵生成算法

分别用序码复制和矩阵复制两种方法生成矩阵.实验环境:OS 为 Windows 2000 Professional,CPU 为赛扬 2.4GHz,256M 内存,VC++6.0.

1. 序码复制算法

- (1) 分配二维数组 $A[2^k][2^k]$ 用于保存 Walsh 矩阵,每一行的第 1 个元素初始化为+1;
- (2) 对于某种序的 Walsh 函数,首先生成序码表,保存在二维数组 $B[2^k][k]$ 中, B 中每一行元素对应某种序的序码,用二进制串表示;
- (3) 扫描 B 中的每一行元素,调用镜像或平移复制规则生成 A 的一行元素.

序码复制算法(以 Paley 矩阵的生成为例,其他序的矩阵生成可以类似得到)的 C 语言描述如下(其中,变量 vector_width 表示当前向量的宽度):

```
for (i=0; i<2k; i++) A[i][0]=1; /*初始化数组 A 的每一行第 1 个元素为 1*/
生成二进制码表,保存在数组 B 中;
```

```

for (i=0;i<2k;i++){
    vector_width=1; /*对于每一行,向量宽度初始化为 1*/
    for (j=0;j<k;j++){
        start=0; /*用于数组 A 的每一列的滑动游标*/
        for (step=0;step<vector_width;step++){ /*step 表示数组 A 的当前向量滑动的起始地址*/
            if (B[i][k-j-1]==0)/*根据不同的序,执行镜像复制或者平移复制*/
                A[i][vector_width+step]=A[i][start++]; /*偶复制或者正平移复制*/
            else
                A[i][vector_width+step]=-A[i][start++]; /*奇复制或者反平移复制*/
        }vector_width<<=1; /*扫描数组 B 每一个元素后,向量宽度加倍*/
    }
}

```

2. 矩阵复制算法(包括行复制和块复制两种算法)

- (1) 分配二维数组 $A[2^k][2^k]$,用于保存 Walsh 矩阵,初始化 $A[0][0]=+1$;
- (2) 根据行复制或块复制规则迭代地填充 A 的所有其他元素(如图 3 所示)。

行复制和块复制算法的 C 语言描述如下(其中变量 $matrix_width$ 表示当前方阵的宽度):

- (a) 行复制算法(以 Walsh 矩阵的生成为例)。

$A[0][0]=1$; /*初始化 A 的第 1 个元素为 1*/

$matrix_width=1$; /*初始化 A 的当前宽度为 1*/

```

for (i=0;i<k;i++){
    /*将当前欲填充的方阵分为左右两部分,首先生成左半部分的元素*/
    for (row=matrix_width-1;row>=0;row--)
        for (col=0;col<matrix_width;col++)
            A[2*row+1][col]=A[2*row][col]=A[row][col];
    /*利用左边生成的元素信息,再根据镜像复制或者平移复制的两条对偶规则生成右半部分元素*/
    for (row=0;row<matrix_width;row+=2)
        for (col=0;col<matrix_width;col++){
            A[row][col+matrix_width]=A[row][matrix_width-col-1];
            A[row+1][col+matrix_width]=-A[row][matrix_width-col-1];
        }matrix_width<<=1; /*每迭代一次,当前方阵的宽度加倍*/
}

```

- (b) 块复制算法(以类 Walsh 矩阵的生成为例):

$A[0][0]=1$; $matrix_width=1$; /*初始化 A 的第 1 个元素为 1,初始化 A 的当前宽度为 1*/

```

for (i=0;i<k;i++){
    /*将当前欲填充的方阵均匀分为 4 个小方阵,其中左上角元素是上一层矩阵元素,不必再计算*/
    for (row=0;row<matrix_width;row++)
        for (col=0;col<matrix_width;col++){
            A[row][col+matrix_width]=A[row][matrix_width-col-1]; /*生成右上角元素*/
            A[row+matrix_width][col]=A[matrix_width-row-1][col]; /*生成左下角元素*/
        }
    /*利用左下角元素,生成右下角元素*/
    for (row=0;row<matrix_width;row++)

```

```

for (col=0;col<matrix_width;col++)
    A[row+matrix_width][col+matrix_width]=-A[row+matrix_width][matrix_width-col-1];
matrix_width<<=1; /*每迭代一次,当前方阵的宽度加倍*/
}

```

易知,相对于序码复制法,矩阵复制法不需要计算和保存序码信息表,也消除了扫描序码表时不必要的判断.前者是基于“一步到位”的策略,而后者基于迭代技术.虽然最终生成 A 的实际计算量是相同的,但前者的辅助操作明显大于后者.图 4 是 3 种算法的性能曲线比较.可以看出,块复制方法是最快的方法,行复制介于二者之间. k 每增加 1,矩阵元素数就增加为原来的 4 倍,计算量也增加为原来的 4 倍.

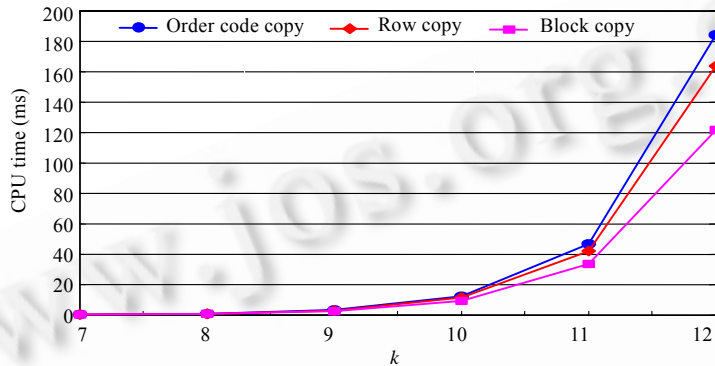


Fig.4 Performance comparison of the three copy methods

图 4 3 种复制方法性能比较

3.2 图像生成

Walsh 矩阵生成后,通过将矩阵元素映射到计算机屏幕上,就可以得到其图像.文献[10]给出了 Hadamard 矩阵(设为 H)的图像,并指出从 H 矩阵的性质不难得到其图像的几何性质,诸如对称性、分块的相似性;如果 H_2^T 图像占据单位正方形,面积为 1,那么,其中黑色块的面积为 $0.5+1/2^{r+1}$ 等.由于分块过程反复运用了嵌入手续,因此,矩阵 H_2^T 的图像具有无限精细的自相似结构($r \rightarrow \infty$).这就给出了一类有意义的分形.图 5 为 6 种序的 5 阶和 8 阶 Walsh 方阵的图像,其中,5 阶图像为放大图像,使每个 8×8 的像素点区域对应一个矩阵元素;8 阶图像为每个像素点对应一个矩阵元素.

从 8 阶图像可以看出整个图像的自相似结构及对称性,从 5 阶图像可以看出其细节.图 5(1)和图 5(2)即为 Hadamard 矩阵的图像.文献[11]中出现了 Hadamard 阵图像(图 5(1)), Walsh 阵图像(图 5(9))以及 Paley 阵图像(图 5(11)),并指出:Hadamard 阵图像具有嵌套结构(nested);Walsh 阵图像的每一行的颜色变化都比前一行多一次;Paley 阵图像的每一行对 Walsh 阵图像的相应行进行了重新排序.可以看出,Paley 阵图像具有明显的断裂现象(图 5(11)和图 5(12)),而 Walsh 阵图像比较平滑(图 5(9)和图 5(10)).图像的对称性如下:除了逆 Walsh 序与反 Walsh 序以外,其他序的图像都关于主对角线对称;而逆 Walsh 序与反 Walsh 序的图像互为转置.逆 Walsh 序与反 Walsh 序的这种不对称性是否就意味着类 Walsh 序更“适合”作为 Walsh 序的反呢?这显然需要进一步的研究.不过,引进类 Walsh 阵以后,Walsh 方阵的排序方式增至表 2 中列出的 4 种.实际上,Walsh 方阵仅有表 2 所列的 4 种对称的形式.

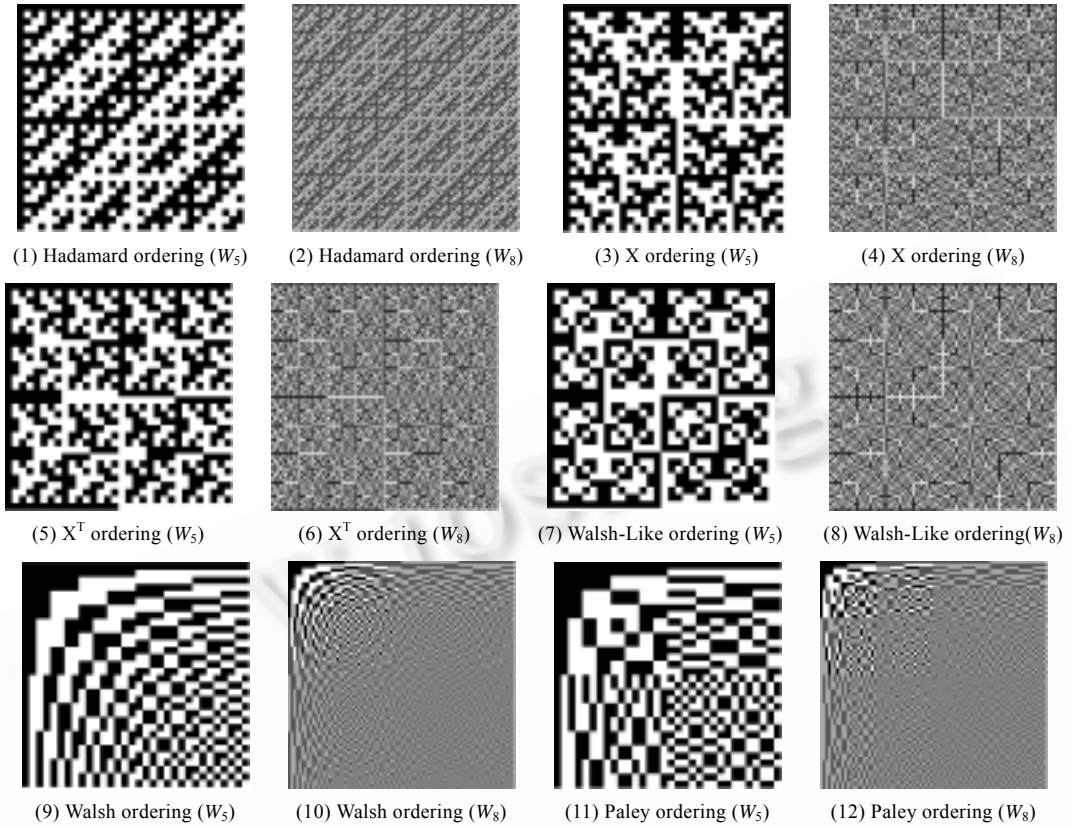


Fig.5 Images of six orderings of Walsh matrix

图 5 6 种序 Walsh 矩阵的图像

Table 2 Four symmetric Walsh matrices

表 2 4 种对称的 Walsh 方阵

Symmetry	Shift symmetry	Mirror image symmetry
Copy mode		
Row copy	Paly ordering	Walsh ordering
Block copy	Hadamard ordering	Walsh-Like ordering

4 结束语

本文通过对 Walsh 序的逆反形式的研究,从对称性角度发现了一种新序(类 Walsh 序),绘制了 6 种序的计算机图像,发现 Walsh 函数矩阵具有分形意义上的自相似结构,并猜想类 Walsh 序更适合作为 Walsh 序的逆反形式.类 Walsh 序和历史上的 3 种对称的 Walsh 序组成了一个非常对称、和谐的 Walsh 函数系统(见表 2).

相对于 Swick 和张其善的基于序码信息的复制技术,基于矩阵的复制技术是一种更先进的技术.因为前者的复制是向量复制,是基于一维的复制;而后者是矩阵复制,是基于二维的复制.显然,本文的方法更适合于 Walsh 函数在二维信号处理领域中的应用,如快速 Walsh 变换(FWT).Walsh 方阵的复制生成成为 FWT 的设计奠定了基础,因为这两者互为逆过程.类 Walsh 序的快速变换算法可以仿照 Hadamard 序很方便地进行设计.对称的 Walsh 函数有一个重要的优点,即由其设计的快速 Walsh 变换的正变换和反变换是互逆的,而逆 Walsh 变换和反 Walsh 变换的正反变换则不互逆^[8,9].我们推荐类 Walsh 序作为 Walsh 函数系统的第 4 种序,正是基于这一点来考虑的.

References:

- [1] Wang NC. The mathematical beauty of Walsh functions (I), (II). Journal of Yunnan University (Natural Science), 1997, 19(Supplement):306–323 (in Chinese with English abstract).
- [2] Swick DA. Walsh function generation. IEEE Trans. on Information Theory, 1969, IT-15(1):167–167.
- [3] Li ZH, Zhang QS. Ordering of Walsh function. IEEE Trans. on Electromagnetic Compatibility, 1983, EMC-25(2):115–119.
- [4] Wang NC. Bisection techniques of synchronous parallel algorithms design. Science in China (Series A), 1995, 25(2):207–211 (in Chinese with English abstract).
- [5] Wang NC. Walsh functions generated with evolution. China Journal of Image and Graphics, 1996, 1(3):225–231 (in Chinese with English abstract).
- [6] Lu XP, He NZ, Wang NC. The reverse forms of Gray-code. Journal of Yunnan University (Natural Science), 1997, 19(Supplement): 297–299 (in Chinese with English abstract).
- [7] Zhang QS. Signal's Copy Theory and Its Applications. Beijing: People's Posts & Telecommunications Publishing House, 2001 (in Chinese).
- [8] Guo WB, Wang NC, Shi BC. Fast algorithms for the Walsh transform in M-ordering. Mini-Micro System, 2003, 24(2):295–298 (in Chinese with English abstract).
- [9] Li Q, Zhou ML, Shi BC, Wang NC. Fast algorithms for the Walsh transform in contrary Walsh-ordering. Science General Journal, 1998, 43(5):482–488 (in Chinese with English abstract).
- [10] Qi DX. Fractal and its Computer Generation. Beijing: Science Press, 1996 (in Chinese).
- [11] Wolfram S. A New Kind of Science. Champaign: Wolfram Media, Inc., 2002.

附中文参考文献:

- [1] 王能超. Walsh 函数的数学美(I)、(II). 云南大学学报(自然科学版), 1997, 19(增刊):306–323.
- [4] 王能超. 同步并行算法设计的二分技术. 中国科学(A 辑), 1995, 25(2):207–211.
- [5] 王能超. Walsh 函数的演化生成. 中国图像图形学报, 1996, 1(3):225–231.
- [6] 陆晓朋, 何南忠, 王能超. Gray 码的逆反形式. 云南大学学报(自然科学版), 1997, 19(增刊):297–299.
- [7] 张其善. 信号的复制生成理论及应用. 北京: 人民邮电出版社, 2001.
- [8] 郭卫斌, 王能超, 施保昌. M 序 Walsh 变换的快速算法设计. 小型微型计算机系统, 2003, 24(2):295–298.
- [9] 李青, 周美莲, 施保昌, 王能超. 反 Walsh 序 Walsh 变换的快速算法设计. 科学通报, 1998, 43(5):482–488.
- [10] 齐东旭. 分形及其计算机生成. 北京: 科学出版社, 1996.



陈宁涛(1976 -),男,河南长垣人,博士生,主要研究领域为分形与混沌理论及可视化,快速算法.



施保昌(1959 -),男,教授,博士生导师,主要研究领域为快速算法,并行算法.



王能超(1937 -),男,教授,博士生导师,主要研究领域为快速算法,并行算法.