

# 数字 Walsh 滤波器设计\*

G. G. Murray

## 引言

在 1970 年 Walsh 函数应用讨论会上,提出了几篇模拟 Walsh 滤波器方面的论文报告,其中突出的有 H. F. Harmuth<sup>[1]</sup>, E. F. Vandivere<sup>[2]</sup>, T. Lee<sup>[3]</sup> 和 D. Roth<sup>[4]</sup> 的文章。与模拟相反的数字 Walsh 滤波器的实现是令人感兴趣的,原因是它具有灵活性,采用现有的大规模集成电路,以及有好的物理性能。甚至采用 MSI (中规模集成电路)的成熟的设计都是可能的。此外,快速 Walsh 变换能导致有效的、然而比较简单的滤波器。

数字 Walsh 滤波器应用的一个有趣例子是在图象处理方面的应用,如由 H. C. Andrews 和 W. K. Pratt<sup>[5]</sup> 所给出的。图象的改进和数据压缩可以用合适的滤波器来达到。他们的研究依赖一个数字计算机。用一个小的电子柜即一个 Walsh 滤波器来代替计算机的可能性是有吸引力的。那时,可更有效地遥测从卫星和飞机送回到地面的气象摄影和地面资源的信息。此外也可以改进传真装备。

## Hadamard 矩阵

了解函数  $f$  的 Walsh 变换的一个方便的方法是利用 Hadamard 矩阵。各阶的 Hadamard 矩阵可以如下法递归地定义,即一开始用

$$H_2 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix}$$

以后使用矩阵的 Kronecker 乘积。举例说明:

$$H_8 = H_4 \times H_4 = \begin{bmatrix} H_4 & H_4 \\ H_4 & -H_4 \end{bmatrix}$$

于是给定  $N$  阶的 Hadamard 矩阵,用  $H_2$  当作第二个因子,其 Kronecker 乘积产生  $2N$  阶的 Hadamard 矩阵。

下面考虑按列构成的具有分量  $f_0, f_1, \dots, f_{N-1}$  的矢量  $f$ , 其中  $N = 2^n$ 。在有些情况下,  $f_i$  可看作是输入函数  $f(t)$  的离散实值采样;或者可以看作是一张照片的一条水平线上等间隔点的光强度。无论哪种情况,矢量  $f$  的 Walsh 变换可定义为:

$$F = Hf$$

变换所得矢量的分量可视为频谱的系数。这是因为,  $H_N$  的行是  $N$  阶的 Walsh 函数。  $f(t)$  可

\* 译自 IEEE vol. EMC-13, No.3(1971)。

展开成

$$f(t) = \sum F_i w_i(t)$$

它表示基于  $N$  个正交函数即 Walsh 函数  $w_i(t)$  的一系列近似值。用离散的代表法表示时，

$$f_j = \sum F_i w_{i,j/N}$$

式中  $w_{i,j}$  是在第  $j$  个区间上第  $i$  个 Walsh 函数之值。 $f(t)$  的范围总是取有限的，且通常是归一的。

一个重要的观察结果是， $H$  是一个对称矩阵，亦即  $H$  是其自身的逆：

$$HH = NI$$

此处假定， $H$  是  $N$  阶矩阵， $I$  是单位矩阵。因此，对矢量  $f$  应用两次  $H$  可恢复原来的函数：

$$f = (1/N)HHf = (N/N)If = f$$

在滤波器设计中，可以应用上述运算。首先，进行 Walsh 变换以进入“序列”（“sequency”）或频率域，这样就能够有选择地衰减或放大频率。例如，若  $D$  是对角线矩阵，那么它可当作一个滤波器来用：

$$f^* = (1/N)HDHf$$

滤波后  $f$  的形式  $f^*$  是由序列域恢复的，其中  $D$  的实值对角线的数值不是提高就是衰减不同的序列。如此，可得到低通、高通和混合的滤波器。

## 滤波的类型

在 Andrews 和 Pratt 的表示法中，用对角线矩阵  $D$  来定义的滤波器不但是线性的，而且是标量的。由于它不取决于系数的振幅，所以说它是线性的；又因为每个新的系数仅取决于该系数先前的值，所以说它是标量的。在他们的表示法中，当  $D$  含有不为 0 的非对角线元素时，就有与标量滤波相反的矢量滤波。

在非线性滤波中，系数既可作为序列的函数又可作为振幅的函数来变化。语言处理中的一个例子是谱频谱（即频谱的系数取对数）。Andrews 和 Pratt 的另一种方法是门限采样，即丢掉绝对值小的系数。

如上面所指出的，数字滤波器优点之一是它的灵活性。一旦获得了频谱的系数，就可以以各种方式对其进行运算。数字技术既准确又比较容易实现。

下面的例子具体说明一种可叫作择多逻辑 (majority logic) 的两值函数滤波方法。这完全是为了说明问题而提出的。定义  $f$  为：

$$f = 01000001000101110001110000110001$$

其 32 个 Walsh 频谱系数的分布如下：4 个幅度为 12，4 个幅度为 8，12 个幅度为 4，12 个等于 0。如果除了五个系数外都丢弃，且采用择多逻辑——为了解决问题这种逻辑需要一个奇数——则得到的是滤波后的信号。取四个最强的系数，并用幅度 8 中的一个 ( $w_5(t)$ ) 对其增广，则所用的项是  $w_5(t)$ ,  $w_9(t)$ ,  $w_{12}(t)$ ,  $w_{21}(t)$  和  $w_{31}(t)$ 。滤波后的函数与原来的函数  $f$  相比只是第 0 位和第 25 位不相同。

当然，通常  $f$  不会在两值的，而是在某个最大和最小值之间变化的。处理前必须抽样和量化。在大多数应用中，特别是在图象处理中，目的是由比较少量的频谱系数抽样还原函数。在这种情况下，就能作到数据压缩，从而使通信带宽缩减。

## 直接的数字 Walsh 滤波器

图 1 具体说明了实现数字 Walsh 滤波器的直接方法。图左边是带有抽样  $f_i$  的输入母线。A/S 是一般的加减器，R 是累加寄存器。如图所示，滤波器是 8 级的。

在母线上，抽样  $f_i$  是相继引入的。对所有加减器来说，每个抽样是同时可得到的。在一个只读存储器 (ROM) 的控制下，每个 A/S 依照 Hadamard 矩阵相应行的符号执行一次加法或减法；根据抽样是第一个，还是第二个，第三个，...，控制符号由行中第一个，第二个，第三个，... 元素取出。由于该行是 +1 和 -1 的一个 Walsh 函数，于是构成一个内积。第一、第二个累加器最终持有

$$F_0 = f_0 + f_1 + f_2 + \dots + f_7$$

$$F_1 = f_0 - f_1 + f_2 - \dots - f_7$$

一旦频谱的系数  $F_i$  是在累加器里，就能实现滤波。对一个门限采样滤波器来说，阈值应置在母线上，使其能够和不同的系数进行比较。阈值以下的那些系数调到 0。

图 1 也示出，怎样利用频谱的系数反馈到输入母线使其能实现第二次变换来恢复滤波后的信号。

对这种设置的一种变动是使寄存器排列在输入母线上与加减器 (A/S) 单元相对应的位置上。在这种排列中，每个 A/S 只存取在自己的寄存器中的抽样。例如，在抽样  $f_0$  和  $f_1$  进入后，第一个 A/S 将取到两个，而第二个 A/S 只看到  $f_0$ 。这样就得到一种交错的运算。这种方案的优点是在  $f_7$  进入之后，就可得到  $F_0$ ，且滤波后它立即可放在母线上紧接在  $f_7$  后面。用这种方法，以步进的方式可以方便地恢复滤波后的信号。

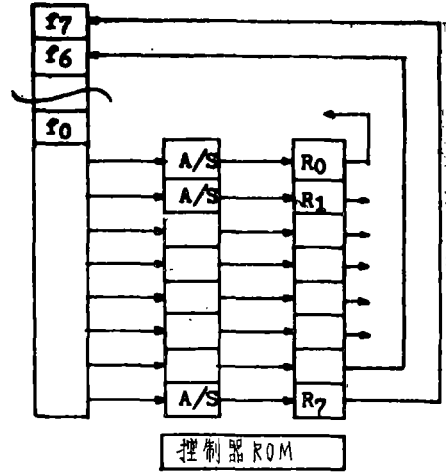


图 1 数字滤波器

## 采用 FWT 的滤波器

由于图 1 的加减器是比较简单的设备，而且能用 LSI MOSFET (大规模集成金属氧化物半导体场效应晶体管) 集成电路来实现，因此整个系统可推广到更高级数的情形。存储器也使用 MOSFET 电路作成移位寄存器 (SR) 的形式。FWT (快速 Walsh 变换) 的利用大大地改进了滤波器的性能。这种改进可以在速度或简单性这两方面实现。就目前而言，重点在前者；但是对某些应用来说，硬设备的简单性是优先的。

FFT (快速付里叶变换) 或 FWT 的应用，使计算的步数由  $N^2$  次减为  $N \log_2 N$  次，M. C. Pease<sup>[6]</sup> 建立了对并行处理有重要意义的另一种 FFT 算法形式。C.K. Rushforth<sup>[7]</sup> 修改了这种算法使之适用于 FWT。在 Pease 的方法中，避免了原来算法对于级联排列 (每级包括有输入数据的一个不同的“排列次序” (“shuffling”)) 的要求。与此相代替的是，总数为  $\log_2 N$  的各级是完全相同的。因此就有可能采用一个单级并把数据从输出到输入循环  $\log_2 N$  次。

这种简化的关键是一个理想的“排列”(shuffle)操作器P,当P加到一个偶数维矢量上时,它能将其分量分成前半和后半两组,然后交替地先从前一半取出第一个,再从后一半取出一个, ..., 就象洗纸牌时的情形一样。概括 Pease 的推导: 当  $N=2^n$ ,

$$H_N = (H_2 \times I_2 \times \dots \times I_2) \cdot (I_2 \times H_2 \times \dots \times I_2) \cdot \dots \cdot (I_2 \times I_2 \times \dots \times H_2)$$

式中右边有  $n$  项。引进缩写

$$C = H_2 \times I_2 \times \dots \times I_2$$

和理想的“排列”矩阵P, 于是就能证明

$$I_2 \times H_2 \times \dots \times I_2 = PCP^{-1}$$

$$I_2 \times I_2 \times H_2 \times \dots \times I_2 = P^2CP^{-2}$$

等等。因此得到

$$H_N = C(PCP^{-1})(P^2CP^{-2}) \dots (P^{n-1}CP^{-(n-1)}) = (CP)^n$$

这就是所要的结果。

上式表明了, FWT 可由理想的排列P后面作用算子C的  $n$  次重复来获得。通过研究  $C_8$ , 可确定C的性质。

$$C_8 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

因此C加到矢量上的作用是, 由相邻对来取分量且构成和与差。图2具体说明了这一点。

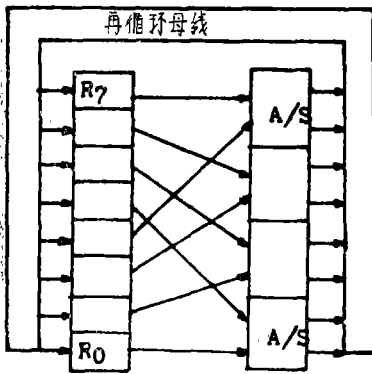


图2 并行的 FWT

图中假定了输入抽样  $f_i$  是同时可得到的。每个 A/S 是一个加减法运算单元。相邻分量在通过相应的输入交叉线接受理想的排列之后, 由 A/S 构成和与差。其输出再循环到输入端。 $n$  步后, 在寄存器里可得到 FWT。假定抽样确实是同时可得到的, 那么这种机构比图1的装置快  $N/n$  倍。但是, 硬设备没有节省。

不过, 对于串行处理是能够采用 Pease 的算法的, 它可导得一种比较简单的滤波器。如图3中描写的, 一个单个加减器和两个移位寄存器构成了完整的滤波器。利用上面的移位寄存器的中心抽头, 对进入 A/S 的数据实现理想的排列。成对数据点既作加又作减。在下次通过时, 对下面的移位寄存器中数据的处理和第一次通过时对上面的移位寄存器中数据的处理一样。(图3中没有具体说明这些连接。)

$n$  次通过后, 可得到 FWT 和具有自然次序的频谱系数。H. S. Stone<sup>[6]</sup> 指出, 理想的排列不仅对 FFT 和 FWT, 而且对分类和多项式求值都是很适用的。对门限抽样来说,

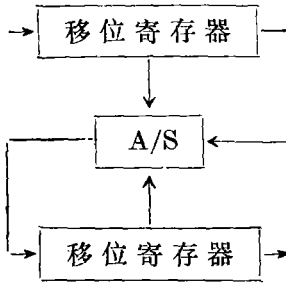


图3 串行的 FWT 滤波器

分类在图 3 中能够完成,其中电平是自适应的。若门限是固定的数值,要恢复滤波过的信号,则可将某些系数调到 0,且再做一次 FWT。使用两个移位寄存器时,这些装置适于用 LSI 制造。

## 图 象 处 理

Andrews 和 Pratt 在用计算机研究图象处理的变换方法时指出,二维的 FWT 是十分有效的。当应用于图象时,经过作变换、滤波后通过一个通道送到接收站,在该处利用第二次变换再构成原来的图象。有两种可能得到的好处。一是数据压缩,其次是通信通道中的抗扰性。实验证明,用 10:1 或更高的数据压缩,照片中的细节仍能观察。这一特点正是主要的兴趣所在。

考虑由  $x$  和  $y$  方向格栅点上的许多强度采样表示的一帧图象。分辨力选择为,对某个  $n$  值在每个方向是  $N=2^n$ 。在 Andrews 和 Pratt 的很大一部分工作中,他们使用  $256 \times 256$  点的图象。若  $f(x, y)$  代表在格栅中某一点的强度,则用一维的变换类推,二维的 Walsh 变换可定义成

$$F(u, v) = Hf(x, y)H$$

如前所述,当再次应用这个变换时,原来的函数除了一个比例因子外可恢复原状:

$$f(x, y) = (1/N^2)HF(u, v)H$$

从整个运算次数来看,这种方法是具有吸引力的,因为采用 FWT 只需要  $2N \log_2 N$  次运算。不过对二维的变换来说,所需的存储器可能是过量的。在一个方向进行变换之后,为了在另一个方向作变换,就需要存储用作输入的  $256 \times 256 = 65,000$  个字,亦即,在构成  $HfH$  前,  $Hf$  必须是存储的。

### 存储器简化

进行第一次变换前,图象是存放在照片中的。利用这一优点, H. D. Wishner<sup>[9]</sup> 提出了一种节省存储器的方法。图象按需要反复地扫描。对图象的每列向下扫描时可显示出图象,不过这时对所有的列来说仅仅构成(比如说)头 8 个频谱系数。这样,只有  $8 \times 256 = 4096$  个字需要存储,因为相当于在行上运算的第二个变换是立即进行的,并同时发送所得的变换系数。

现在图象再重复扫描,并对各列计算下 8 个频谱系数。按前所述,变换在另一个方向进行,并发送下组二维变换系数。对  $n=3$  的情形,图 4 示出怎样构成对一帧  $8 \times 8$  图象的一列每次扫描只产生两个频谱系数的处理器。

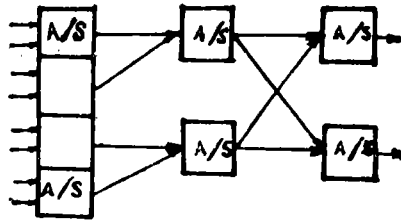


图4 部分编码器

现在来考察存储器简化的另外两种方法。这两种方法不包含多次扫描（多次扫描并不总是可能作到的）。头一种方法比较直接了当，它将  $256 \times 256$  个抽样组成的图象分割成比如说 16 块。在这种情形中，要测量  $64 \times 64$  项，且二维 FWT 按前面所述的方法进行。虽然只需要做图象的单独一次扫描，但一个问题是扫描的方法不是通常的用整列或整行的那一种方法。

第二种方法采用常规的比如说用行的图象扫描，不过与其它方法不同的是，只进行一次一维的变换。假定它能给出良好的性能，则它在运算速度和硬设备量方面具有明显的优点。实际上，图 3 的设备是可取的。值得注意的一点是，移位寄存器存储器对一维的变换来说是理想的，但对二维的变换就不是。这是因为后者要求在  $x$  和  $y$  方向都存取数据。

在第二种方法中，图象是沿着每第 16 条线水平切断的，同时输入到 FWT 滤波器的是具有  $16 \times 256 = 4096$  个分量的单个矢量。按图 3 构成滤波器时，每个移位寄存器要 4096 个字长。对 8 位字来说，当然可以并行地设置 8 个移位寄存器。对每个矢量，要形成 FWT 就需要 12 次通过。

问题是，用所得的一维 FWT 能获得多大的数据压缩。人们可以希望缩减  $1/3$ ，这恰好是根据单个 256 单元线编码而来的。由依次扫描 16 条这样的线能获得什么样的附加压缩呢？在 16 条线的范围内，典型的图象应该有好的相关。假定是一种理想的情形，即 16 条线都是相同的，此时可容易地看出，附加的压缩将是  $1/16$ 。如果仅仅实现  $1/3$ ，则这将是令人满意的，因为全部压缩将是  $1/3 \times 1/3 = 1/9$ 。由于节省了硬设备，比二维变换可能达到的数据压缩比低的压缩比是可以接受的。

## 小 结

本文介绍了数字 Walsh 滤波器的几种不同的设备。重点放在能采用 MOSFET 型大规模集成电路（特别是移位寄存器）经济地进行设计的那些设备。本文说明了，可以用相同的实现 Walsh 变换的加减器，以不同的几种方式来获得频谱系数，并由此产生了许多不同的滤波作用。FWT 对减小计算量特别有效，并且可以应用 Pease 算法。

虽然 Pease 算法是为了便于实现高速并行处理的，但对高效率的串行逻辑滤波器也适用。对于由一点传到另一点的数据压缩的图象而言，这两种设备是有用的。在二维变换处理方法中，存储器需要量是一个问题。本文对简化存储器提出了几种可能的途径。其中最简单的一种基于采用移位寄存器的一维 FWT。这种方法是吸引人的，不过需要通过计算机模拟来作出鉴定。

（单荣华译）

## 参 考 文 献

- [ 1 ] H. F. Harmuth, "Survey of analog sequency filters based on Walsh functions", Proc. of the Symposium on Applications of Walsh Functions, p. 208-219, 1970. 4.
- [ 2 ] E. F. Vandivere, "A flexible Walsh filter design for signals of moderately low sequency", *ibid*, p. 3-6.
- [ 3 ] T. Lee, "Hardware approach to Walsh function sequency filters", *ibid*, p. 7-11.
- [ 4 ] D. Roth, "Special filters based on Walsh functions", *ibid*, p. 12-16.
- [ 5 ] H. C. Andrews & W. K. Pratt, "Digital image processing", *ibid*, p. 183-194.
- [ 6 ] M. C. Pease, "An adaptation of the Fast Fourier Transform for parallel processing", *J. of the ACM*, vol. 15, p.252-264, 1968.4.
- [ 7 ] C. D. Rushforth, "Fast Fourier-Hadamard decoding of orthogonal codes", *Information and Control*, vol. 15, p. 33-37, 1969. 7.
- [ 8 ] H. S. Stone, "Parallel processing with the perfect shuffle", *IEEE* vol. C-20, p. 153-161, 1971.
- [ 9 ] H. D. Wishner, Aerospace Corp., private communication.