FACTA UNIVERSITATIS (NIŠ)
Series: Electronics and Energetics vol.13, No.3, December 2000, 365-377

# MORE ON FILTER-GENERATING SYSTEM

**Sead Samadi and Akinori Nishihara**

**Abstract.** It is known that some families of digital filters may be represented by single multidimensional transfer functions called filter-generating functions. Previous examples of such filter-generating systems were related to the class of maximally flat FIR digital filters. The purpose of this paper is to show that the Chebyshev kernel used in the McClellan transformation of 1-D filters, and the Sylvester-type Hadamard transform, also known as the Walsh-Hadamard transform also possess simple multidimensional generating functions. Explicit formulas for the generating functions are derived and filter-generating systems for their implementation are discussed.

**Key words:** IIR digital filter, McClellan transformation, Hadamard transformation.

## 1. Introduction

An important distinction of the field of digital signal processing from other related engineering and mathematical disciplines lies in its heavy usage of the $\mathcal{Z}$ transform for representation and manipulation of signals and filters. Traditionally, engineers have used this important tool, known to mathematicians through the method of generating functions, to describe single signals and filter transfer functions. In the field of combinatorics, mathematicians make extensive use of generating functions for enumeration purposes [1]. In the theory of special functions, generating functions of important families of polynomials have been known for a long time. A classic textbook on the $\mathcal{Z}$ transform is that of Jury [2], and a recent work on generating functions is the interesting book of Wilf [3].

In [4] the authors showed that the notions of the $\mathcal{Z}$ transform and generating functions can be extended to provide concise representations of

Manuscript received August 8, 2000.
Manuscript revised October 12, 2000.
S. Samadi is with Shibaura Institute of Technology, Tokyo, Japan. A. Nishihara is with Tokyo Institute of Technology, Tokyo, Japan (e-mail: aki@ec.ss.titech.ac.jp).

whole families of some classes of interrelated transfer functions. In addition to the usual discrete time indices, the usage in [4] involves a new group of indices that specify and enumerate the members of the family. We coined the term filter-generating function to signify the resulting mathematical entity. The system implementation of a filter-generating function was then called a filter-generating system. As a concrete example, we derived a 3-D IIR filter-generating function that could generate the entire family of maximally flat FIR filters of linear-phase type. It was also shown that a filter-generating function may be used to generate modular structures for implementation of the members of the family. This gives a systematic mathematical method for multiplexing a single low-order digital filter from the family to generate the entire members by trading processing time for less number of adders and multipliers. Figure 1 illustrates the three main applications of filter-generating systems.

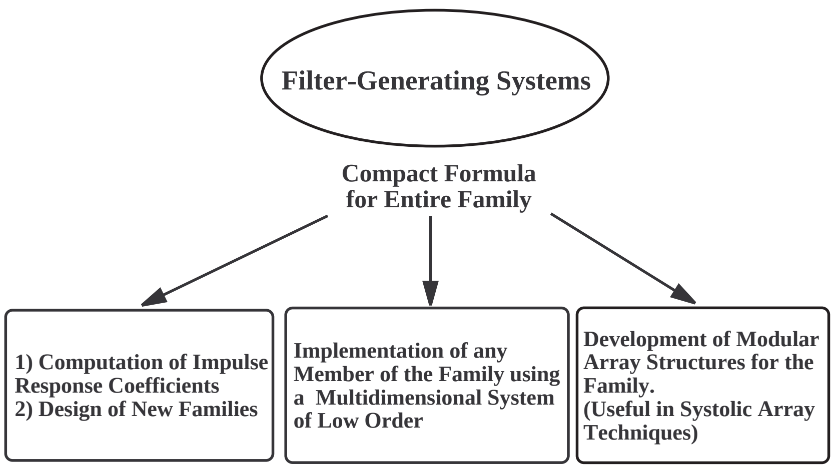



*Fig. 1. Possible applications of filter-generating systems.*

In [5] we showed that the more general class of nonlinear-phase maximally flat filters also possess a generating function. Though we only worked out maximally flat examples in [4] and [5], this does not mean that the use of filter-generating functions is limited to maximally flat filters. In this paper we see that it is possible to derive generating functions for some classical filter kernels and linear transforms as well. Specifically, the Chebyshev kernels and Sylvester-type Hadamard transforms are discussed. The Chebyshev kernels are well known in the literature on multidimensional filter design in connection with the McClellan transformation. The Hadamard transforms have found various applications in communications and signal processing [10]. By deriving generating systems for these two fundamental tools we will be able to gain a strict and unified mathematical view of their properties, and algorithms and structures for their implementation. We also develop structures based on the newly derived filter-generating systems.

This paper is organized as follows. After a brief review of the concept of filter-generating systems in Section 2, we derive a generating system for the family of filters designed using the McClellan transformation in Section 3. Then we give a thought-provoking generating system for the Sylvester-type Hadamard transform in Section 4. Finally, we identify some open problems and draw conclusions in Section 5.

## 2. The Concept

In this section we review the concept of filter generating systems. A more detailed exposition can be found in [4]. Consider the family of $\mathcal{Z}$ transforms $\{H_i(z),\, i = 0, 1, \dots\}$, where each member of the family is being identified by its index $i$. The family may contain a finite or infinite number of members. For instance, we can think of a family of lowpass digital filters of a fixed order with varying group delay characteristics. The formal power series

$$G(x) = \sum_{i \geq 0} H_i(z)\, x^i \tag{1}$$

is defined to be the filter-generating function for the family. In the case that a closed-form rational representation for $G(x)$ exists, we can write

$$G(x) = \frac{\sum_{i=0}^{P} A_i(z) x^i}{1 - \sum_{i=0}^{Q} B_i(z) x^i}. \tag{2}$$

The rational functions $A_i(z)$ and $B_i(z)$ may be viewed as the coefficients of a discrete-time system whose transfer function is $G(x)$ and has $x$ as its delay operator. Such system is called a filter-generating system for the family $\{H_i(z)\}$.

If the filter-generating system is excited by an impulse signal under the condition of zero initial states, then the $i$th sample $g(i)$ of the system impulse response equals $H_i(z)$, i.e., $g(i) = H_i(z)$, where $g(i)$ denotes the $i$th impulse response sample of the system $G(x)$. Thus the transfer functions of the entire members of the family $\{H_i(z)\}$ are obtained.

Using the well-known recursive structure for realization of IIR systems, the filter-generating system (2) may be realized as a recursive system as shown in [4]. The system can be used for three different purposes. First, to compute the transfer functions of the members $H_i(z)$ successively by applying the impulse sequence

$$\delta(n) = \{1, 0, 0, \dots\}, \tag{3}$$

and observing the output signals that are rational functions in $z$. Here $G(x)$ is considered to be a 1-D transfer function with coefficients that are rational functions of $z$. Second, we can use it to obtain the convolution of a 1-D signal $X(z)$ and the $i$th member of the family using the temporal realization approach detailed in [4]. Third, it is possible to obtain a spatio-temporal realization in the form of a modular structure that can implement an arbitrary member of the family [4].

The definition of filter-generating functions can be naturally extended to cover the family of doubly-indexed $\mathcal{Z}$ transforms as

$$G(x, y) = \sum_{i\geq 0}\sum_{j\geq 0} H_{i,j}(z)\, x^i\, y^j. \tag{4}$$

Again if a closed-form rational expression of the form

$$G(x, y) = \frac{\sum_{i=0}^{P}\sum_{j=0}^{Q} A_{i,j}(z)x^i y^j}{1 - \sum_{i=0}^{P}\sum_{j=0_{(i+j\neq 0)}}^{Q} B_{i,j}(z)x^i y^j}, \tag{5}$$

exists, then $G(x, y)$ corresponds to a filter-generating system. The coefficients $A_{i,j}(z)$ and $B_{i,j}(z)$ are rational functions in $z$.

Why the concept of filter-generating systems is beneficial in developing algorithms, architectures, and designs? The answer lies in their power to integrate all those aspects. That is, they integrate the implementation-related properties and the design-related aspects into a single higher dimensional transfer function. A super transfer function that tells us what the basic design parameters of the family are. A mathematical entity that shows how we can take advantage of the intimate relationship among the members of the family to implement them in a modular and regular fashion. Examples of the following two chapters show that filter-generating functions need not be plain rational functions of $x$ and $z$. They can have more complex forms involving convolution of polynomials, as shown in Section 3. Alternatively, they may even have an equivalent product-type expression that is a cascade of low-order multidimensional systems as shown in Section 4. In both of those cases the associating filter-generating systems offer the same advantages as their plain IIR counterparts.

## 3. Filter-Generating System for McClellan Transformation

A reader familiar with the so-called Chebyshev structure [7], proposed for implementation of multidimensional digital filters designed via the method of McClellan transformation [8], may appreciate the power of filter-generating systems after reading this section. Here we show that the transformation method of McClellan, the associated Chebyshev structure, and the whole family of 2-D digital filters designed by that approach can be represented using a very compact filter-generating function.

### 3.1 Derivation of generating function

We use the notation developed in [9]. Let

$$H(z) = \sum_{-N \leq n \leq N} h(n)\, z^n$$

denote a 1-D zero-phase filter, where $h(n) = h(-n)$, $n \geq 1$. Then we can write

$$H(z) = \sum_{0 \leq n \leq N} a(n)\, \frac{z^n + z^{-n}}{2},$$

where $a(0) = h(0)$, and $a(n) = 2h(n)$, for $n \geq 1$. Using the Chebyshev polynomials of the first kind, the filter can be expressed as

$$H(z) = \sum_{0 \leq n \leq N} a(n)\, T_n(\frac{z + z^{-1}}{2}).$$

Now let $F(z_1, z_2)$ denote the zero-phase transformation function that plays the role of mapping function. It transforms the 1-D frequency response of $H(z)$ to the desired 2-D characteristics. The 2-D digital filter obtained by applying $F(z_1, z_2)$ can be expressed as

$$H_N(z_1, z_2) = \sum_{0 \leq n \leq N} a(n)\, T_n\big(F(z_1, z_2)\big).$$

Note that we used the slightly modified notation $H_N(z_1,z_2)$ by taking the value of $N$ into account. This is a necessary step in order to enhance our ability to handle filters of any given order. Now we write

$$G(x) = \sum_{N\geq 0} H_N(z_1,z_2)\,x^N. \tag{6}$$

The real power of $G(x)$ as the filter-generating function for the family of 2-D filters designed via the McClellan transformation, can be exploited if we can find a closed-form expression for it. Substitution of the summation form of $H_N(z_1,z_2)$ into (6) results in

$$G(x) = \sum_{N\geq 0}\sum_{0\leq n\leq N} a(n)\,T_n\big(F(z_1,z_2)\big)\,x^N.$$

From which we have

$$\begin{aligned}(1-x)G(x) =& a(0)\\ &+a(1)\,T_1\big(F(z_1,z_2)\big)\,x\\ &+a(2)\,T_2\big(F(z_1,z_2)\big)\,x^2+\cdots,\end{aligned}$$

and thus we can write

$$(1-x)G(x) = \sum_{N\geq 0} a(N)\,T_N\big(F(z_1,z_2)\big)\,x^N.$$

Now note that the right-hand side of the above expression is a power series whose terms are the product of two independent sequences, viz., $a(N)$ and $T_N\big(F(z_1,z_2)\big)$. Thus it is equal to the convolution of the two power series

$$A(x) = \sum_{N\geq 0} a(N)\,x^N,$$

and

$$C(x) = \sum_{N\geq 0} T_N\big(F(z_1,z_2)\big)\,x^N.$$

Applying the three-term recurrence formula of Chebyshev polynomials [6]

$$\begin{aligned}&T_N(F) = 2F\,T_{N-1}(F) - T_{N-2}(F), \qquad N=2,3,\ldots,\\ &T_0(F)=1, \quad T_1(F)=F,\end{aligned}$$

and using the definition of $C(x)$, we get

$$C(x) = 1 + F(z_1, z_2)\, x + \sum_{N \geq 2} \Big( 2F(z_1, z_2)\, T_{N-1}\big(F(z_1, z_2)\big) - T_{N-2}\big(F(z_1, z_2)\big) \Big)\, x^N.$$

From the above, we can easily verify that the simple closed-form expression

$$C(x) = \frac{1 - F(z_1, z_2)\, x}{1 - 2F(z_1, z_2)\, x + x^2}$$

is the generating function for the Chebyshev polynomials of the first kind. Consequently, the overall generating function $G(x)$ becomes

$$G(x) = \frac{\dfrac{1 - F(z_1, z_2)\, x}{1 - 2F(z_1, z_2)\, x + x^2} * A(x)}{1 - x}. \tag{7}$$

Unlike the generating functions derived so far by the authors, the above expression is not a plain rational function. We will shortly see that this will not affect its usefulness.

### 3.2 Structure

A spatio-temporal implementation of (7) is developed here. Consider the 1-D impulse signal $\delta(n_x)$, where $n_x$ are indices along the direction associated with $x$. The impulse response of $G(x)$ can be computed by the following four steps.

1. Obtain the impulse response of $C(x)$, denoted $y_1(n_x)$, using $y_1(n_x) = 2F(z_1, z_2) y_1(n_x - 1) - y_1(n_x - 2) + \delta(n_x) - F(z_1, z_2)\delta(n_x - 1)$. The first two values of $y_1(n_x)$ are $y_1(0) = 1, \quad y_1(1) = F(z_1, z_2)$. For $n_x \geq 2$ we have $y_1(n_x) = 2F(z_1, z_2)\, y_1(n_x - 1) - y_1(n_x - 2)$.
2. Compute $y_2(n_x) = y_1(n_x)\, a(n_x)$. This is equal to the impulse response of $C(x) \star A(x)$.
3. To get the overall impulse response, the signal $y_2(n_x)$ is fed to the system with transfer function $1/(1 - x)$. This results in $y_3(n_x)$ that is formed according to the recurrence $y_3(n_x) = y_3(n_x - 1) + y_2(n_x)$.

A signal flow-graph for the structure generated by $G(x)$ with outputs up to $n_x = 4$ is given in Fig. 2. This structure is identical to what is called the Chebyshev structure in the literature [7], [9].

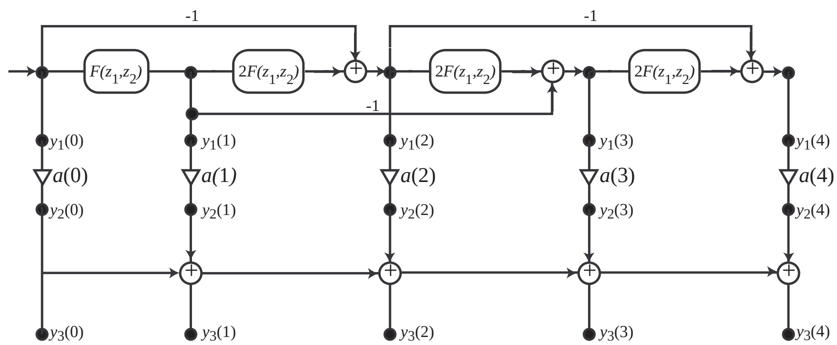



*Fig. 2.The Chebyshev structure generated from the filter-generating system (7) for the McClellan transformation.*

## 4. Filter-Generating System for Hadamard Transform

This section introduces another field of application for filter-generating systems. In multirate signal processing it is known that the so-called block orthogonal transforms are in fact multirate filter banks with filters that have the same order $N$ as the rate conversion factor. A number of researchers have investigated this connection, and we know that the celebrated Sylvester-type Hadamard transform can be implemented using a tree-structured filter bank with only two types of order-1 digital filters, namely, $H_0(z) = 1 + z^{-1}$, and $H_1(z) = -1 + z^{-1}$ on its branches. So there is a highly regular filter bank for this block orthogonal transform that can be extended to any desirable order in a straight-forward manner. See Fig. 3 for an example of such tree-structured filter bank.

### 4.1 Derivation of generating function

Our aim here is to show that the generating system approach yields very interesting results on the Sylvester-type Hadamard transform that enhance our current knowledge, and open up new implementation possibilities. Note that the Hadamard transform considered in this paper is also known as the Walsh-Hadamard transform in the literature.

Let us define a generating-function for the $2^N$th-order Hadamard transform. The function should put together all $2^N$ digital filters of the filter

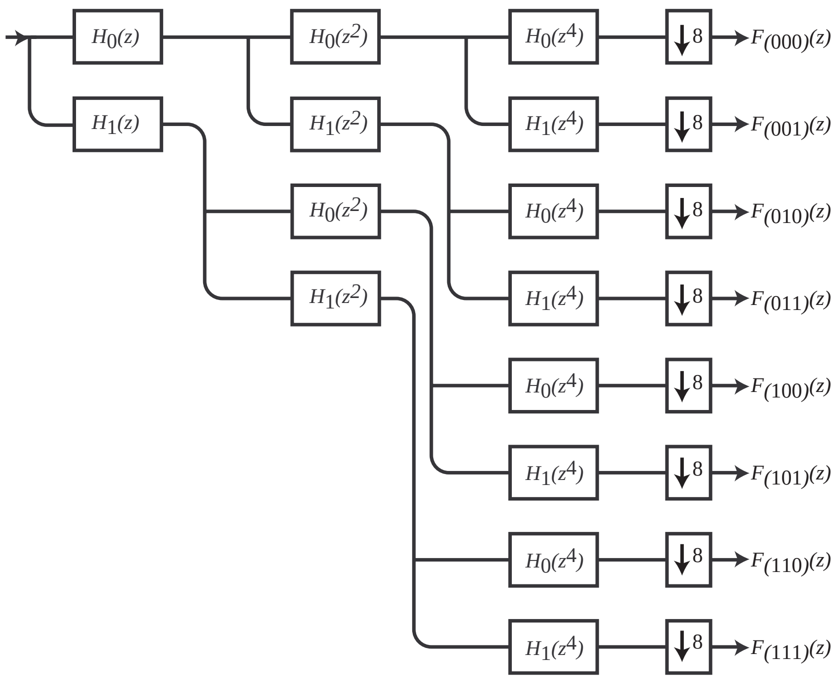



*Figure 3. Tree-structured Hadamard filter bank.*

bank tree into a single power series. Thus we define

$$G_N(x) = \sum_{0 \leq k \leq 2^N - 1} F_k(z) x^k. \tag{8}$$

The transfer functions $F_k(z)$ are given by [12], [13]

$$F_k(z) = H_{i_0}(z^{2^0}) H_{i_1}(z^{2^1}) \cdots H_{i_{N-1}}(z^{2^{N-1}}), \tag{9}$$

where $\{i_0, i_1, \dots, i_{N-1}\}$ is the base-2 representation of integer $k$, i.e.,

$$k = i_0 + 2i_1 + \cdots + 2^{N-1} i_{N-1}.$$

Thus we can write

$$G_N(x) = \sum_{\{i_0, i_1, \dots, i_{N-1}\} \in \{0,1\}} H_{i_0}(z^{2^0})\, H_{i_1}(z^{2^1})\, \cdots H_{i_{N-1}}(z^{2^{N-1}})\, x^{i_0 + 2i_1 + \cdots + 2^{N-1} i_{N-1}}.$$

Needless to mention that there are several possible ways to extend $G_N(x)$ to an infinite-length power series and facilitate the derivation of an IIR type of filter-generating system. However our approach here is to keep $G_N(x)$ in its present form and obtain a factored form for it. We will see that this leads to a modular filter-generating system that can be implemented in the cascade form. Let us write

$$\begin{aligned} G_N(x) = \\ H_0(z^{2^{N-1}}) \sum_{\{i_0,\dots,i_{N-2}\}\in\{0,1\}} H_{i_0}(z^{2^0})H_{i_1}(z^{2^1})\cdots H_{i_{N-2}}(z^{2^{N-2}})x^{i_0+2i_1+\cdots+2^{N-2}i_{N-2}} \\ + x^{2^{N-1}}H_1(z^{2^{N-1}})\sum_{\{i_0,\dots,i_{N-2}\}\in\{0,1\}} H_{i_0}(z^{2^0})H_{i_1}(z^{2^1})\cdots H_{i_{N-2}}(z^{2^{N-2}})\, x^{i_0+2i_1+\cdots+2^{N-2}i_{N-2}}. \end{aligned}$$

Thus the common summation term can be factored out as

$$\begin{aligned} G_N(x) = \Big[H_0(z^{2^{N-1}}) + x^{2^{N-1}}\, H_1(z^{2^{N-1}})\Big] \\ \times \sum_{\{i_0,\dots,i_{N-2}\}\in\{0,1\}} H_{i_0}(z^{2^0})H_{i_1}(z^{2^1})\cdots H_{i_{N-2}}(z^{2^{N-2}})x^{i_0+2i_1+\cdots+2^{N-2}i_{N-2}}. \end{aligned}$$

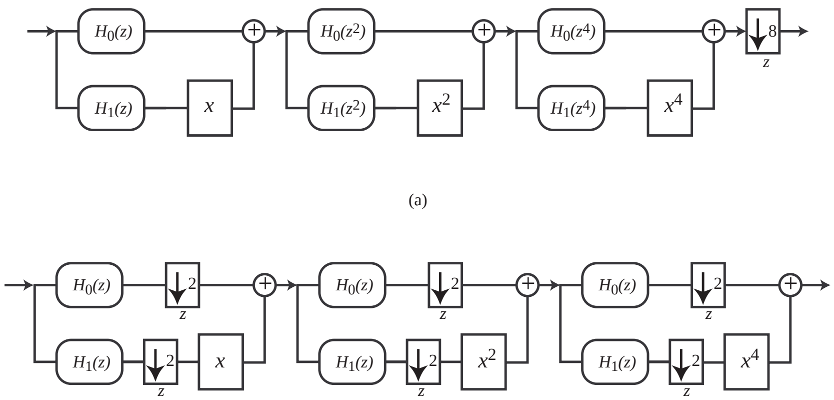
Now, repetitive application of the above procedure results in nothing but

$$G_N(x) = \prod_{0\le k\le N-1} \Big[H_0(z^{2^k}) + x^{2^k}\, H_1(z^{2^k})\Big]. \tag{10}$$

The above is a product-form filter-generating system for the Hadamard transform of order $2^N$.

### 4.2 Structure

A direct implementation of (10) for $N = 3$ is illustrated in Fig. 4(a). To save on the number of additions, we should move the down-sampler left towards the input front of the structure. This can be done in the usual manner, using the noble identities, and the result is the 2-D multirate filter generating system of Fig. 4(b). This is a more modular structure compared to the one given in Fig. 4(a). It can be verified that the number of arithmetic operations per output sample for Fig. 4(b) is the same as that of a conventional fast Hadamard transform. The structure makes repetitive use of a single filter pair at each stage through the multiplexing power of the




(b)

*Figure 4. Filter-generating system for Hadamard filter bank.*

delay element $x$. This means that in a direct special-purpose hardware implementation, the number of adders may be kept to the minimum possible number at the cost of a two-dimensional delaying mechanism.

It seems that the structure of Fig. 4(b) is equivalent to the decoder hardware of Mariner'69 mission invented by Green at NASA Jet Propulsion Laboratory [14]. Though this structure is not mentioned in the signal processing literature, [14] gives a schematic of Green's fast Hadamard transform machine, the Green machine, that may be thought of as an implementation of the structure of Fig. 4(b). It is interesting that the technique of filter-generating systems can elegantly provide a system theoretic point of view for an ingenious hardware invention.

## 5. Conclusion

Filter-generating functions have the power to furnish all information needed to synthesize and implement an entire family of discrete-time systems by a single multidimensional transfer function. This formula may have different forms. The most common form is a low-order multidimensional IIR transfer function that is the product of simple FIR and IIR transfer functions. This paper provided two alternative forms. One involves convolution of transfer functions rather than the usual products. It was shown that filters designed through the McClellan transformation method may be compactly represented by this type of filter-generating systems. Another possibility is a generating function expressible as infinite products of simple multidimensional transfer functions. The family of filter banks that realize the Sylvester type Hadamard transforms has a generating function that belongs to this latter form.

We developed structures based on the two newly-derived filter-generating systems. For the McClellan transformation the structure turned out to be equivalent to the Chebyshev structure. For the Hadamard filter banks we conjectured that the structure is equivalent to the Green machine, a fast Hadamard transformer that has not been mentioned in the signal processing literature. In spite of the fact that the filter-generating systems of this paper are slightly different from the ones developed in our previous papers, both structures still yield cellular realizations of the filter members through regularly interconnected identical cells of low orders. The ability to produce such cellular structures through a system theoretic approach is interesting from both theoretical and practical considerations.

Some open problems include development of filter-generating systems for the discrete Fourier and Cosine transforms and investigation of their relation to structures for fast implementation of the said transforms.

## REFERENCES


1. G. E. Andrews: *The Theory of Partitions, Encyclopedia of Mathematics and its Applications, vol. 2.* Reading, MA: Addison-Wesley, 1976.
2. E. I. Jury: *Theory and Application of the Z Transform Method.* Huntington, NY, R. E. Krieger, 1973.
3. H. S. Wilf: *Generatingfunctionology.* 2nd ed., San Diego, CA: Academic Press, 1994.
4. S. Samadi, A. Nishihara and H. Iwakura: *Filter-generating systems.* IEEE Trans. Circuits Syst. II, vol. 47, March 2000, pp. 214–221.
5. S. Samadi, A. Nishihara and H. Iwakura: *Universal maximally flat lowpass FIR systems.* IEEE Trans. Signal Processing, vol. 48, July 2000.
6. T. J. Rivlin: *Chebyshev Polynomials: From Approximation Theory to Algebra and Number Theory.* New York: John Wiley, 1990.
7. J. H. McClellan and D. K. Chan: *A 2-D FIR filter structure derived from the Chebyshev recursion.* IEEE Trans. Circuits Syst., vol. 24, July 1977, pp. 372–378.
8. R. M. Mersereau, W. F. G. Mecklenbrauker and T. F. Quatieri: *McClellan transformations for 2-D digital filtering: I–Design.* IEEE Trans. Circuits Syst., vol. 23, July 1976, pp. 405–414.
9. D. E. Dudgeon and R. M. Mersereau: *Multidimensional Digital Signal Processing.* Prentice-Hall, 1984.

10. R. K. Yarlagadda and J. E. Hershey: *Hadamard Matrix Analysis and Synthesis, with Applications to Communications and Signal/Image Processing.* Boston: Kluwer, 1996.

11. M. Vetterli: *Tree structures for orthogonal transforms and application to the Hadamard Transform.* Signal Proceesing, vol. 5, 1983, pp. 473–484.

12. A. E. Cetin, O. N. Gerek, and S. Ulukus: *Block wavelet transforms for image coding.* IEEE Trans. Circuits Syst. Video Technol., vol. 3, Dec. 1993, pp. 433–435

13. A. Makur: *BOT's based on nonuniform filter banks.* IEEE Trans. Signal Processing, vol. 44, Aug. 1996, pp. 1971–1981.

14. V. S. Pless and W. C. Huffman: *Handbook of Coding Theory: Volume II.* Amsterdam: Elsevier, 1998.