PARALLEL-CASCADE REALIZATIONS AND APPROXIMATIONS OF TRUNCATED VOLTERRA SYSTEMS

T. M. Panicker and V. John Mathews

Department of Electrical Engineering, University of Utah, Salt Lake City, UT 84112.

ABSTRACT

This paper introduces parallel-cascade realizations of truncated Volterra systems with arbitrary, but finite order of nonlinearity. Parallel-cascade realizations implement higher-order Volterra systems using parallel and multiplicative combinations of lower-order Volterra systems. Such realizations are very modular and therefore well-suited for VLSI implementation. A systematic way of approximating higher-order Volterra systems in parallel-cascade form using a reduced number of branches and a bound on the mean-square error in the output signals caused by such approximate realizations are derived in this paper. A variation of the parallel-cascade structure in which a pth order Volterra filter is implemented as a parallel combination of linear filters whose outputs are raised to the pth power is also described in this paper.

1. INTRODUCTION

Truncated Volterra system models have found applications in a large variety of signal processing problems including signal detection and estimation [4], adaptive filtering [3], and image processing [7]. The output of a homogeneous and causal pth order Volterra system with N-sample memory is related to its input as [8]

$$y(n) = \sum_{m_p=m_{p-1}}^{N-1} \cdots \sum_{m_1=0}^{N-1} h_p(m_1,\cdots,m_p) \times x(n-m_1)\cdots x(n-m_p), \quad (1)$$

where $h_r(m_1,\cdots,m_r)$ represents the rth order Volterra kernel of the system and we have explicitly made use of the invariance of the coefficients with respect to permutations of their indices. The objective of this paper is to introduce alternatives to direct form realizations of truncated Volterra systems. We show in this paper that any system with input-output relationship as in (1) can be represented exactly using a parallel combination of components as shown in Figure 1. The components in the figure are made of lower-order Volterra systems combined in a multiplicative manner. It is well-known that higher-order Volterra systems can be synthesized using lower-order components [8]. It is also known that parallel-cascade combinations similar to that in Figure 1 can approximate a large class of nonlinear systems [2]. Efficient parallel-cascade realization of quadratic filters has been presented in [1]. However, the authors are not aware of any work that involves the *exact* realization of Volterra kernels using parallel-cascade structures for kernels of order higher than two.

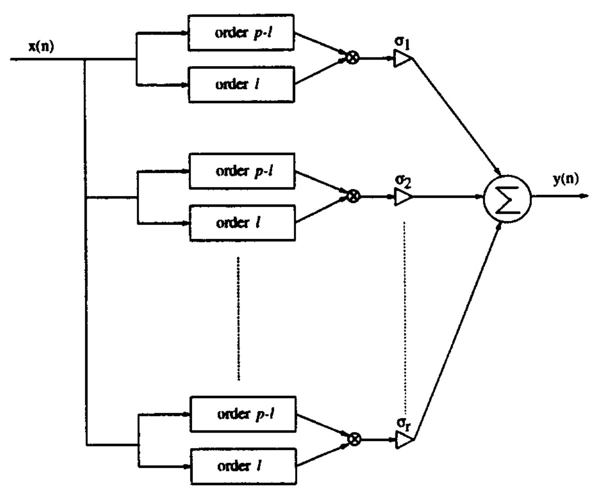

Figure 1. A parallel-cascade realization of a pth order Volterra kernel. Each block represents a Volterra system of the order shown within.

2. PARALLEL-CASCADE REALIZATIONS

We consider the problem of obtaining a parallel-cascade realization of a homogeneous pth order Volterra system using lth order and $(p-l)$th order components as in Figure 1. The results are applicable to any p and $l < p$, and each component can be further decomposed using lower-order components. The input-output relationship in (1) can be compactly written using matrix notation as

$$y(n) = \mathbf{X}_{N,l}^T(n)\mathbf{H}_{N,l,p-l}\mathbf{X}_{N,p-l}(n), \quad (2)$$

where $\mathbf{H}_{N,p_1,p_2}$ is a matrix of dimension $\begin{pmatrix} N+p_1-1 \\ p_1 \end{pmatrix} \times \begin{pmatrix} N+p_2-1 \\ p_2 \end{pmatrix}$ elements in which the coefficients in (1) are arranged in some orderly manner. The vector $\mathbf{X}_{N,p_1}(n)$ has $\begin{pmatrix} N+p_1-1 \\ p_1 \end{pmatrix}$ elements and contains all possible p_1th order product signals of the form $x(n-k_1)x(n-k_2)\cdots x(n-k_{p_1})$, where $0\leq k_1,k_2,\cdots,k_{p_1} \leq N-1$. Since

$\mathbf{H}_{N,p_1,p_2}$ contains more elements than there are independent coefficients, several entries of the coefficient matrix may be zero. Let the rank of $\mathbf{H}_{N,l,p-l}$ be r. Then, it is well known that we can express $\mathbf{H}_{N,l,p-l}$ as

$$\mathbf{H}_{N,l,p-l} = \sum_{i=1}^{r} \sigma_i \mathbf{U}_i \mathbf{V}_i^T, \tag{3}$$

using singular value decomposition. In the above equations, σ_i's are the non-zero singular values of $\mathbf{H}_{N,l,p-l}$ and $\mathbf{U}_i$'s and $\mathbf{V}_i$'s are the left and right singular vectors, respectively, of the matrix. Note also that the sets $\{\mathbf{U}_i; i = 1, 2, \cdots, r\}$ and $\{\mathbf{V}_i; i = 1, 2, \cdots, r\}$ consists of orthonormal vectors. Substituting (3) in (2), we get

$$\begin{aligned} y(n) &= \sum_{i=1}^{r} \sigma_i [\mathbf{X}_{N,l}^T \mathbf{U}_i][\mathbf{V}_i^T \mathbf{X}_{N,p-l}] \\ &= \sum_{i=1}^{r} \sigma_i y_{l,i}(n) y_{p-l,i}(n), \end{aligned} \tag{4}$$

where we have defined $y_{l,i}(n)$ as the output of a homogeneous lth order Volterra system given by

$$y_{l,i}(n) = \mathbf{X}_{l,i}^T \mathbf{U}_i. \tag{5}$$

The signal $y_{p-l,i}(n)$ is also defined in a similar manner. From the above analysis, it should be clear that the left and right singular vectors define the coefficients of the lower-order components used in the decomposition shown in Figure 1. Furthermore, The output of the system is the sum of products of outputs of lth and $(p-l)$th order systems weighted by the singular values.

The decomposition in (3) immediately results in a systematic method for approximating Volterra systems using simpler structures. It is well-known that the best k-rank approximation for $\mathbf{H}_{N,l,p-l}$ is

$$\tilde{\mathbf{H}}_{N,l,p-l} = \sum_{i=1}^{k} \sigma_i \mathbf{U}_i \mathbf{V}_i^T, \tag{6}$$

where we assumed that $\sigma_1, \sigma_2, \cdots, \sigma_k$ are the k largest singular values of $H_{N,l,p-l}$. Furthermore, we can characterize the error in the approximation using

$$\| \mathbf{H}_{N,l,p-l} - \tilde{\mathbf{H}}_{N,l,p-l} \|^2 = \sum_{i=k+1}^{r} \sigma_i^2, \tag{7}$$

where the squared-norm represents the sum of the squared values of the error matrix. Consequently, we can approximate the system using a lower-complexity system by removing the branches corresponding to low singular values.

3. A BOUND ON THE MEAN-SQUARE OUTPUT ERROR

Previous work on approximation of Volterra kernels may be found in [6]. The derivation presented in this section is similar to the work in [6]. Let $\mathbf{h}_e$ denote a vector containing the coefficient errors caused by the approximation in the previous section and let $\tilde{y}(n)$ represent the output of the approximate realization. The mean-square error at the output of the approximated filter can then be written as

$$E[(y(n) - \tilde{y}(n))^2] = \mathbf{h}_e^T E[\mathbf{X}_{N,p}(n)\mathbf{X}_{N,p}^T(n)]\mathbf{h}_e. \tag{8}$$

The correlation matrix $E[\mathbf{X}_{N,p}(n)\mathbf{X}_{N,p}^T(n)]$ is symmetric and can therefore be decomposed as

$$E[\mathbf{X}_{N,p}(n)\mathbf{X}_{N,p}^T(n)] = \mathbf{U}\mathbf{S}\mathbf{U}^T, \tag{9}$$

where $\mathbf{U}$ is the matrix of the eigenvectors and $\mathbf{S}$ is a diagonal matrix with the eigenvalues as the diagonal entries. Substituting (9) in (8), we get

$$E[(y(n) - \tilde{y}(n))^2] = \mathbf{h}_{eu}^T \mathbf{S} \mathbf{h}_{eu}, \tag{10}$$

where $\mathbf{h}_{eu} = \mathbf{U}^T \mathbf{h}_e$. Let S_{max} be the maximum of all the diagonal entries in the matrix $\mathbf{S}$. Then, the above expression can be bounded by

$$\begin{aligned} E[(y(n) - \tilde{y}(n))^2] &\leq S_{max} \| \mathbf{h}_{eu} \|^2 \\ &= S_{max} \sum_{i=k+1}^{r} \sigma_i^2, \end{aligned} \tag{11}$$

where k is the number of branches retained in the parallel-cascade realization. Note that

$$\| \mathbf{h}_{eu} \|^2 = \| \mathbf{h}_e \|^2 \tag{12}$$

since the columns of $\mathbf{U}$ are orthogonal.

4. VARIATIONS OF THE GENERAL STRUCTURE

We now consider three variations of the parallel-cascade realization. The first structure realizes the higher order filter with lower order components without any matrix decomposition. The second structure utilizes the standard singular value decomposition. The last structure expresses a homogeneous higher order filter of order p as a parallel combination of first order filters whose outputs are raised to the pth power. Unfortunately, there is no systematic way of approximating the filter using structures 1 and 3.

4.1. Structure 1

This structure is similar to the decomposition in [5], where the matrix $\mathbf{H}_{N,l,p-l}$ is written as

$$\mathbf{H}_{N,l,p-l} = \mathbf{H}_{N,l,p-l} \mathbf{I}_{p-l,p-l}, \tag{13}$$

where $\mathbf{I}_{p-l,p-l}$ is an identity matrix of appropriate dimensions. Now we can rewrite (2) using (13) as

$$y(n) = \sum_{i=1}^{m} (\mathbf{X}_{N,l}^T \mathbf{H}_{i,N,l,p-l}) x_{i,N,p-l}(n) \tag{14}$$

where $\mathbf{H}_{i,N,l,p-l}$ is the ith column of the coefficient matrix $\mathbf{H}_{N,l,p-l}$, $x_{i,N,p-l}(n)$ is the ith element of the vector $\mathbf{X}_{N,p-l}(n)$ and m is the number of columns in $\mathbf{H}_{N,l,p-l}$.

4.2. Structure 2

The second structure uses the standard singular value decomposition to obtain the vectors $\mathbf{U}_i$ and $\mathbf{V}_i$. However, when p is even, we can choose l as $p/2$ which implies symmetry of the coefficient matrix $\mathbf{H}_{N,l,p-l}$. Symmetry of the matrix $\mathbf{H}_{N,l,p-l}$ makes the vector $\mathbf{U}_i$ and $\mathbf{V}_i$ identical. The output of the system can therefore be written as

$$y(n) = \sum_{i=1}^{r} \sigma_i y_{p/2,i}^2(n), \qquad (15)$$

where r is the rank of the matrix $\mathbf{H}_{N,l,p-l}$.

A Simulation Example. We consider the realization of a fourth-order system with twelve-sample memory using second-order components. The decomposition based on Structure 3 was performed on the coefficient matrix $\mathbf{H}_{12,2,2}$ whose coefficients are given by

$$h(k_1,k_2,k_3,k_4) = \frac{100}{2\pi[1.5^2 + a_1^4 + a_2^4 + a_3^4 + a_4^4]^{3/4}},$$

where $a_i = (k_i - 6)$ and $0 \le k_1, k_2, k_3, k_4 \le 11$. The maximum number of branches for this particular case is 78. Figure 2 displays the squared-norm of the coefficient error as a function of the number of discarded branches in the approximate realization. From the above result, we can conclude

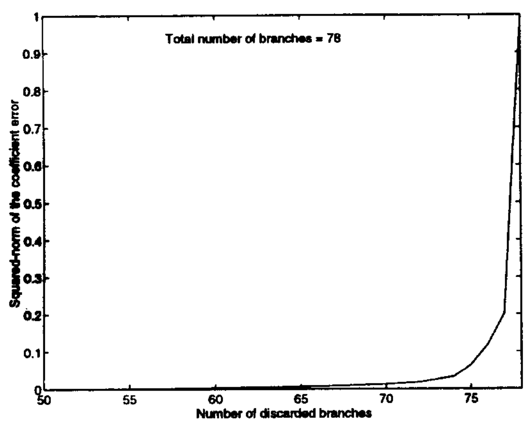

Figure 2. Squared-norm of the coefficient error

that very good approximations can be obtained by retaining less than ten branches of the parallel-cascade structure in this example.

4.3. Structure 3

This structure realizes a homogeneous pth order truncated Volterra filter with N-sample memory as

$$y(n) = \sum_{i=1}^{k_{p,N}} a_{i,p,N}(\mathbf{W}_{i,p,N}^T \mathbf{X}_{N,1}(n))^p, \qquad (16)$$

where $\mathbf{W}_{i,p,N}$ is a linear filter, $k_{p,N}$ represents the total number of branches for the pth order Volterra kernel with N-sample memory, and $a_{i,p,N}$ is a constant multiplier. Note that if $p=2$, we can derive this structure using (15). Such a structure can be derived for the general case in an iterative manner as follows.

For this derivation, we assume that the pth order Volterra kernel is symmetric, *i.e.*, the coefficients do not change with a permutation of its indices. Assume that we can express a $(p-1)$th order filter using the above structure, *i.e*, we can express the output of a $(p-1)$th order Volterra filter can be expressed as

$$y_{p-1}(n) = \sum_{i=1}^{k_{p-1,N}} a_{i,p-1,N}(\mathbf{W}_{i,p-1,N}^T \mathbf{X}_{N,1}(n))^{p-1}. \qquad (17)$$

We will show using this assumption that we can express the output of the pth order Volterra filter also as in (16). Since such a structure exists for homogeneous quadratic filters, it follows by induction that the structure in (16) exists for all homogeneous truncated Volterra filters with finite memory.

Now, consider a pth order kernel $h_p(k_1,k_2,\cdots,k_p)$. We can express $h_p(N-1,k_2,\cdots,k_p)$ using (17) as

$$\begin{aligned} h_p(N-1,k_2,\cdots,k_p) &= \sum_{i=1}^{k_{p-1,N}} a_{i,p-1,N} w_{i,p-1,N}(k_2) \\ &\quad \times \cdots \times w_{i,p-1,N}(k_p), \end{aligned} \qquad (18)$$

where $w_{i,r,L}(k)$ represents the kth element of the coefficient vector $\mathbf{W}_{i,r,L}$. Let us define $\tilde{h}_{p,N}(k_1,k_2,\cdots,k_p)$ as

$$\begin{aligned} \tilde{h}_{p,N}(k_1,k_2,\cdots k_p) &= \sum_{i=1}^{k_{p-1,N}} \frac{a_{i,p-1,N}}{w_{i,p-1,N}(N-1)} \\ &\quad \times w_{i,p-1,N}(k_1) w_{i,p-1,N}(k_2) \\ &\quad \cdots \times w_{i,p-1,N}(k_p) \end{aligned} \qquad (19)$$

We have assumed that $w_{i,p-1,N}(N-1) \neq 0$ for the ith filter in the above step. It is easy to see that

$$h_p(N-1,k_2,\cdots k_p) = \tilde{h}_{p,N}(N-1,k_2,\cdots,k_p). \qquad (20)$$

Because of the symmetry of the coefficients, h_p and $\tilde{h}_{p,N}$ are identical whenever at least one of the indices $k_1,k_2,\cdots,k_p$ is $N-1$. Thus, the realization in (19) obtained using $k_{p-1,N}$ parallel branches, with each branch containing a linear filter followed by a memoryless $(\cdot)^p$ operator and a constant multiplier provides an exact realization of all the coefficients for which at least one of the indices is $N-1$.

In order to achieve the exact representation of all the coefficients, we proceed in a similar manner. That is, we find additional branches for representing the coefficients for which at least one index is $N-2$ exactly, and then find more branches that can represent coefficients having $N-3$ in its indices, and so on until we find one branch that provides the exact representation for $h_p(0,0,\cdots,0)$. Note that the coefficient errors given by

$$\begin{aligned} he_{p,N-1}(k_1,k_2,\cdots,k_p) &= h_p(k_1,k_2,\cdots k_p) \\ &\quad - \tilde{h}_{p,N}(k_1,k_2,\cdots,k_p) \end{aligned} \qquad (21)$$

has non-zero values in the range $0 \le k_1,k_2,\cdots,k_p \le N-2$. We can express all the entries of $he_{p,N-1}(N-2,k_2,\cdots,k_p)$

using a parallel-cascade structure with $k_{p-1,N-1}$ branches that contain linear filters with $(N-1)$-sample memory using the technique described above. That is, we can express $\tilde{he}(N-2, k_2, \cdots k_p)$ as

$$\tilde{he}(N-2, k_2, \cdots k_p) = \sum_{i=1}^{k_{p-1,N-1}} a_{i,p-1,N-1} \times w_{i,p-1,N-1}(k_1) w_{i,p-1,N-1}(k_2) \cdots \times w_{i,p-1,N-1}(k_p). \quad (22)$$

Now, define $\tilde{he}(k_1, k_2, \cdots k_p)$ as

$$\tilde{he}(k_1, k_2, \cdots, k_p) = \sum_{i=1}^{k_{p-1,N-1}} \frac{a_{i,p-1,N-1}}{w_{i,p-1,N-1}(N-2)} \times w_{i,p-1,N-1}(k_1) w_{i,p-1,N-1}(k_2) \cdots \times w_{i,p-1,N-1}(k_p). \quad (23)$$

It follows immediately that

$$h_p(k_1, k_2, \cdots k_p) = \tilde{h}_{p,N}(k_1, k_2, \cdots k_p) + \tilde{he}_{p,N-1}(k_1, k_2, \cdots k_p) \quad (24)$$

for all $k_2, \cdots, k_p$, and $k_1 = (N-1)$ and $(N-2)$. The symmetry of the coefficients implies that all the coefficients with at least one of $k_1, k_2, \cdots, k_p$ as $N-1$ or $N-2$ can be represented by a parallel cascade structure having $k_{p-1,N} + k_{p-1,N-1}$ branches.

The above process may be iterated till we express all the coefficients exactly in a similar manner. At the first level of iteration, the linear filters have a maximum memory span of N samples. The memory span of the filters obtained from the second level of iteration is at most $N-1$. The maximum memory requirement reduces by one sample for each additional level of iteration.

An Example. We consider a homogeneous third-order Volterra filter with three-sample memory with coefficients

$$h_3(k_1, k_2, k_3) = \frac{1}{1 + k_1 + k_2 + k_3} \; ; \; 0 \le k_1, k_2, k_3 \le 2. \quad (25)$$

Using the steps described above we can obtain a realization of the filter using six branches. The decomposition of the second-order filters were achieved using an *LU* decomposition of the corresponding symmetric coefficient matrices. We obtain three branches with coefficients given by

$$\mathbf{W}_{1,3,3} = [1.67 \;\; 1.25 \;\; 1]^T$$
$$\mathbf{W}_{2,3,3} = [0 \;\; 0.75 \;\; 1]^T$$

and

$$\mathbf{W}_{3,3,3} = [0 \;\; 0 \;\; 1]^T,$$

where $(\cdot)^T$ denotes the transpose of $(\cdot)$. The constant multipliers after the $(\cdot)^3$ operators are given by $a_{1,3,3} = 0.12$, $a_{2,3,3} = 0.02$ and $a_{3,3,3} = 6.35 \times 10^{-4}$, respectively. Two branches are required at the second level of the iteration and one branch is required at the third and final level of iteration. The three linear filters for the additional branches are given by

$$\mathbf{W}_{4,3,3} = [4 \;\; 1 \;\; 0]^T$$
$$\mathbf{W}_{5,3,3} = [0 \;\; 1 \;\; 0]^T$$

and

$$\mathbf{W}_{6,3,3} = [1 \;\; 0 \;\; 0]^T.$$

The multipliers for these branches are given by $a_{4,3,3} = 0.0052$, $a_{5,3,3} = 0.001$ and $a_{6,3,3} = 0.11$, respectively.

5. CONCLUDING REMARKS

This paper presented a method for realizing higher-order Volterra systems using parallel-cascade structures. It was also shown that approximate structures with reduced complexity can be derived from the parallel-cascade realization. A bound on the mean-square error at the output of the approximated filter was also derived. The realization of homogeneous pth order Volterra filters using parallel combinations of linear filters whose outputs are raised to the pth power was also presented in this paper. The results presented in this paper are believed to be novel for Volterra filters of nonlinearity order greater than two. The applicability of such filters in practical applications should be significantly enhanced because of the implementational simplicity provided by our structures.

Acknowledgment. The authors thank Rob Nowak for his careful reading of a previous version of this paper. His comments resulted in substantial changes in the paper.

REFERENCES

[1] H. H. Chang, C. L. Nikias, and A. N. Venetsanopoulos, "Efficient implementations of quadratic filters," *IEEE Trans. on Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 1511-1528, Dec. 1986.

[2] M. J. Korenberg, "Functional expansions, parallel cascades, and nonlinear difference equations," *Advanced Methods of Physiological System Modeling*, vol. 1, V. Z. Marmarelis, Ed., U.S.C. Biomedical Simulation Resource, Los Angeles, 1987

[3] V. J. Mathews, "Adaptive polynomial filters," *IEEE Sig. Proc. Mag.*, vol. 8, no. 3, pp. 10-26, July 1991.

[4] B. Picinbono, P. Duvant, "Optimal linear-quadratic systems for detection and estimation," *IEEE Trans. on Inform. Theory*, vol. 34, pp 304-311, 1988.

[5] B. G. Mertzios "Parallel modeling and structure od nonlinear Volterra discrete systems," *IEEE Trans. on Circuits and Systems-I*, vol. 41, no. 5, May 1994.

[6] R. Nowak and B. Van Veen, "Tensor product basis approximations for Volterra filters," to appear in *IEEE Trans. Sig. Proc.*, vol. 44, no. 1, January 1996.

[7] G. Ramponi, "Bi-Impulse response design of isotropic quadratic filters," *Proc. IEEE*, vol. 78, no.4, April 1990.

[8] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*, Reprint Edition, Krieger, 1989.