WAVELET TRANSFORMS FOR NONLINEAR SIGNAL PROCESSING

Robert Nowak

Michigan State University
East Lansing, MI 48824-1226
Email: rnowak@egr.msu.edu
WWW: http://www.egr.msu.edu/spc/

Richard Baraniuk

Rice University
Houston, TX 77251-1892
Email: richb@rice.edu
WWW:http://www-dsp.rice.edu

ABSTRACT

In this paper we describe two new structures for nonlinear signal processing. The new structures simplify the analysis, design, and implementation of nonlinear filters and can be applied to obtain more reliable estimates of higher-order statistics. Both structures are based on a two-step decomposition consisting of a linear orthogonal signal expansion followed by scalar polynomial transformations of the resulting signal coefficients. While most existing approaches to nonlinear signal processing characterize the nonlinearity in the time domain or frequency domain; in our framework any orthogonal signal expansion can be employed. In fact, there are good reasons for characterizing nonlinearity using more general signal representations like the wavelet transform. Wavelet expansions often provide very concise signal representation and thereby can simplify subsequent nonlinear analysis and processing. Moreover, we show that the wavelet domain offers significant theoretical advantages over classical time or frequency domain approaches to nonlinear signal analysis and processing.

1. INTRODUCTION

Nonlinear signal processing techniques are commonly applied in signal analysis, detection and estimation, image enhancement and restoration, and filtering. In this paper, we develop a new approach to nonlinear signal processing based on the ***nonlinear signal transformation*** (NST) depicted in Figure 1. Here, a length-m signal vector $\mathbf{x}$ is first expanded onto an orthonormal signal basis $\{\mathbf{b}_1, \ldots, \mathbf{b}_m\}$ to produce the vector of coefficients $[\beta_1, \ldots, \beta_m]^T$. These signal coefficients are then combined in nonlinear processing nodes η, which are simple n-th order polynomial operations, to form the ***n-th order nonlinear coefficients*** of the signal $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_N]^T$. Concisely, we denote the NST of Figure 1 by the operator $F_n : \mathbf{x} \mapsto \boldsymbol{\theta}$.

The NST framework encompasses two new structures, each corresponding to a different choice for the scalar processing nodes η in Figure 1. *Product nodes* compute different n-fold products of the signal coefficients at each node:

$$\eta(\beta_1, \ldots, \beta_m) = \beta_{i_1}\beta_{i_2}\cdots\beta_{i_n}. \qquad (1)$$

This work was supported by the National Science Foundation, grant no. MIP–9701692 and grant no. MIP–9457438, and the Office of Naval Research, grant no. N00014–95–1–0849

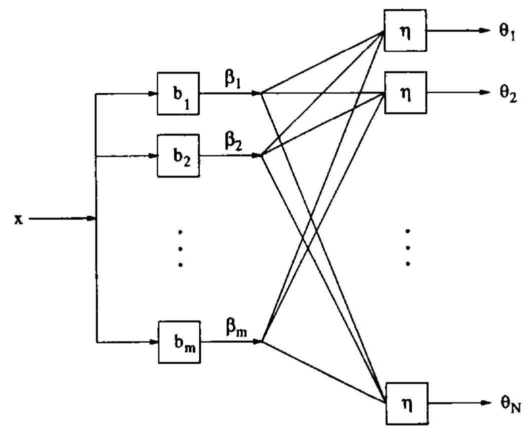

Figure 1: *Nonlinear signal transformation (NST) $F_n : \mathbf{x} \mapsto \boldsymbol{\theta}$. The front end processing (projection onto the basis $\{\mathbf{b}_j\}$) is linear; the back end processing (by η from* (1) *or* (2)*) is nonlinear.*

Summing nodes raise linear combinations of the coefficients to the n-th power:

$$\eta(\beta_1, \ldots, \beta_m) = \left(\sum_{j=1}^{m} a_j\, \beta_j\right)^n. \qquad (2)$$

(Although the outputs of the product and summing nodes are not equivalent, we will see that they both produce similar NSTs.)

We will prove that an NST architecture with $N = \binom{m+n-1}{n}$ processing nodes[1] can generate ***all possible*** n-th order nonlinear interactions between the various signal components, with the strengths of these interactions reflected in the nonlinear signal coefficients $\boldsymbol{\theta}$. Therefore, these coefficients can be used for efficient nonlinear filter implementations, robust statistical estimation, and nonlinear signal analysis.

The NST framework is flexible, because it does not rely on a particular choice of basis $\{\mathbf{b}_j\}$. Traditionally, nonlinear signal analysis has been carried out in the time or frequency domains. For example, if the $\{\mathbf{b}_j\}$ are the canonical unit vectors, or delta basis, then the components of $\boldsymbol{\theta}$ represent

[1] $\binom{m+n-1}{n}$ denotes the binomial coefficient $\frac{(n+m-1)!}{n!\,(m-1)!}$.

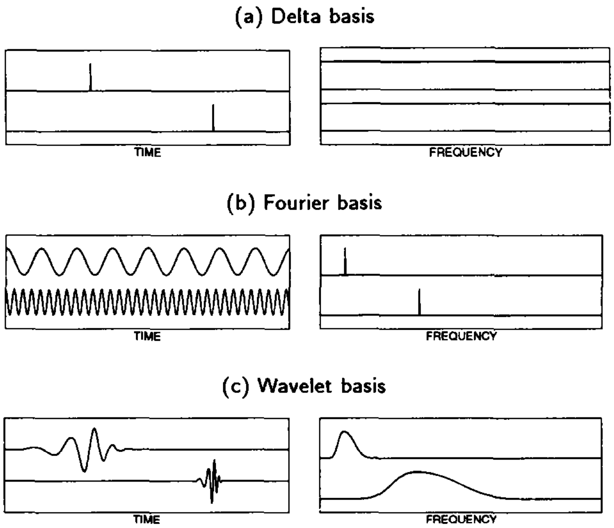

Figure 2: *Comparison of different bases $\{\mathbf{b}_j\}$ for nonlinear signal processing. The choice of basis employed in the linear front end of the NST of Figure 1 determines in which domain we represent signal interactions. Consider a second-order NST, which generates squares β_j^2 and cross-products $\beta_i\beta_j$ of the signal coefficients. We illustrate two basis elements $\mathbf{b}_i$ and $\mathbf{b}_j$ from three different bases, in both the time domain and the frequency (squared magnitude) domain. In the delta basis, each $\mathbf{b}_j$ is a unit pulse, so β_j is simply a sample of the signal. The corresponding NST represents coupling between different time lags of the signal. In the Fourier basis, each $\mathbf{b}_j$ is a sinusoid, so β_j is a Fourier coefficient of the signal. The corresponding NST represents intermodulations between different frequencies. In the wavelet basis, each $\mathbf{b}_j$ is localized in both time and frequency simultaneously, so β_j measures the time-frequency content of the signal. The corresponding NST represents coupling between different localized wavelet atoms.*

n-th order interactions between different time lags of the signal $\mathbf{x}$ (see Figure 2(a)). If the $\{\mathbf{b}_j\}$ make up the Fourier basis, then $\boldsymbol{\theta}$ represents the n-th order frequency intermodulations (see Figure 2(b)).

In this paper, we will emphasize the *wavelet basis* [4], whose elements are localized in both time and frequency. Wavelet-based NSTs represent the local n-th order interactions between signal components at different times *and* frequencies (see Figure 2(c)). From a practical perspective, this can be advantageous in problems involving non-stationary data, such as machinery monitoring [3]. From a theoretical perspective, we will show that the wavelet domain provides an optimal framework for studying nonlinear signals and systems.

We will consider several applications of NSTs in this paper. NSTs provide an elegant structure for the *Volterra filter* that simplifies filter analysis, design, and implementation. Applications of Volterra filters include signal detection and estimation, adaptive filtering, and system identification [8]. The output of a Volterra filter applied to a signal $\mathbf{x}$ consists of a polynomial combination of the samples of $\mathbf{x}$. We will show that every n-th order Volterra filter can be represented by simple linear combinations of the nonlinear signal coefficients $\boldsymbol{\theta}$. NSTs are also naturally suited for performing *higher-order statistical signal analysis* [9]. For example, in the time or frequency domains, the nonlinear signal coefficients $\boldsymbol{\theta}$ are simply values of a higher-order moment or higher-order spectrum [6]. The wavelet domain provides an alternative, and optimal, representation for higher-order statistical analysis.

The paper is organized as follows. In Section 2, show that both the product and summing node NSTs provide a complete representation of all possible n-th order nonlinear signal interactions. We show in Section 3 that wavelet bases are optimal for NST signal analysis and processing. Section 4 applies the theory to Volterra filtering. Setion 5 offers a discussion and conclusions.

2. COMPLETE NSTS

2.1. Criterion for Completeness

In this section, we show that the transformation $F_n : \mathbf{x} \mapsto \boldsymbol{\theta}$ pictured in Figure 1 provides a complete representation of all n-th order nonlinear signal interactions. The notion of a complete transformation is defined as follows.

Definition 1 *Let F_n be fixed. If for every signal $\mathbf{x} = [x(1), \ldots, x(m)]^T \in \mathbb{R}^m$ and every multidimensional array $h \in \underbrace{\mathbb{R}^m \times \cdots \times \mathbb{R}^m}_{n\text{-times}}$ there exists a collection of real numbers $\{\alpha_k\}_{k=1}^N$ such that*

$$\sum_{i_1,\ldots,i_n=1}^{m} h_{i_1,\ldots,i_n} x(i_1)\cdots x(i_n) = \sum_{k=1}^{N} \alpha_k\, \theta_k \qquad (3)$$

where $\boldsymbol{\theta} = F_n(\mathbf{x})$, then the transformation F_n is said to be a complete n-th order signal transformation.

Definition 1 states that every n-th order multilinear functional of $\mathbf{x}$ may be computed by a linear functional of $\boldsymbol{\theta}$. Therefore, a complete n-th order nonlinear signal transformation allows us to study all possible n-th order nonlinear signal interactions of $\mathbf{x}$ in terms of simple linear operations on $\boldsymbol{\theta}$. This implies that a complete NST is capable of realizing every possible n-th order Volterra filter of $\mathbf{x}$. Furthermore, a complete NST captures all possible n-th order signal interactions necessary to compute higher order statistical quantities such as the moments and cumulants of $\mathbf{x}$.

We now show that both the product node and summing node transformations are complete. The NSTs can be interpreted as a linear mapping on an appropriate tensor space. Consequently, the notion of completeness can be formulated as a spanning condition in a tensor space.

First, we need some simple tensor notation. Given a collection of m-dimensional, real-valued vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, with $\mathbf{v}_k = [v_{1,k}, \ldots, v_{m,k}]^T$, the n-fold *tensor* or *Kronecker product* [1] $\tau = \bigotimes_{j=1}^n \mathbf{v}_j$ produces a vector composed of all possible n-fold products of the elements in $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$.

We can also interpret the tensor τ as an amorphous n-dimensional array with elements $\tau_{i_1,\ldots,i_n} = v_{i_1,1} \cdots v_{i_n,n}$. For example, the kernel h in (3) is a tensor. The n-fold tensor product of the vector $\mathbf{v}$ with itself is denoted by $\mathbf{v}^{(n)}$ and contains all n-fold products of the elements in $\mathbf{v}$.

2.2. Product Node Transformation

In the product node transformation, different n-fold products of the signal coefficients are computed at each node according to

$$\eta(\beta_1, \ldots, \beta_m) = \beta_{i_1}\beta_{i_2}\cdots\beta_{i_n}$$

Completeness of this transformation is easily established by noting that this structure is related to a tensor space.

Tensor products simplify the description of the product node NST. First note that products of the form $\beta_{i_1}\cdots\beta_{i_n}$ can be expressed, using standard tensor product identities [1], as

$$\beta_{i_1}\cdots\beta_{i_n} = \langle \mathbf{b}_{i_1}, \mathbf{x}\rangle \cdots \langle \mathbf{b}_{i_n}, \mathbf{x}\rangle = \left\langle \bigotimes_{j=1}^{n} \mathbf{b}_{i_j}, \mathbf{x}^{(n)} \right\rangle, \tag{4}$$

where $\langle\cdot,\cdot\rangle$ denotes the inner product. In tensor notation, the linear combination $\sum_{k=1}^{N} \alpha_k \theta_k$ of Definition 1 is given by

$$\sum_{1\le i_1 \le \cdots \le i_n \le m} \alpha_{i_1,\ldots,i_n} \left\langle \left(\bigotimes_{j=1}^{n} \mathbf{b}_{i_j} \right), \mathbf{x}^{(n)} \right\rangle, \tag{5}$$

where we have used a multi-indexing scheme on the $\{\alpha_k\}$ for notational convenience. Similarly, we can rewrite the multilinear function on the left side of (3) as the inner product

$$\sum_{1\le i_1 \le \cdots \le i_n \le m} h_{i_1,\ldots,i_n} x_{i_1} \cdots x_{i_n} = \langle \mathbf{h}, \mathbf{x}^{(n)} \rangle \tag{6}$$

where $\mathbf{h}$ is a vectorized version of the kernel h. Comparing this expression to (3) and (6), we make the identification

$$\mathbf{h} = \sum_{1\le i_1 \le \cdots \le i_n \le m} \alpha_{i_1,\ldots,i_n} \left(\bigotimes_{j=1}^{n} \mathbf{b}_{i_j} \right). \tag{7}$$

It follows from (7) and Definition 1 that the product node NST is complete if the following condition is satisfied:

$$\text{Span}\left\{ \left(\bigotimes_{j=1}^{n} \mathbf{b}_{i_j} \right) : 1 \le i_1 \le \ldots \le i_n \le m \right\} = \text{Span}\{n\text{-th order kernels}\}.$$

This is condition is easily established using some results from tensor theory [6]:

Theorem 1 *Let $\{\mathbf{b}_1, \ldots, \mathbf{b}_m\}$ be a basis (orthonormal basis) for $\mathbb{R}^m$. Then the NST with $\binom{m+n-1}{n}$ product nodes forming all unique n-fold products of $\beta_1, \ldots, \beta_m$ is complete.*

2.3. Summing node transformation

Recall that the summing node nonlinearities (2) raise linear combinations of the $\{\beta_1, \ldots, \beta_m\}$ to the n-th power. For the k-th output θ_k, we can write

$$\theta_k \triangleq \left(\sum_{j=1}^{m} a_{j,k}\, \beta_j \right)^n = \left(\sum_{j=1}^{m} a_{j,k} \langle \mathbf{b}_j, \mathbf{x} \rangle \right)^n, \tag{8}$$

We can interpret (8) as weighting the connection between the j-th basis element and the k-th summing node with the gain $a_{j,k}$ (see Figure 1).

We can also write (8) as

$$\theta_k = \left\langle \sum_{j=1}^{m} a_{i,k}\, \mathbf{b}_i, \mathbf{x} \right\rangle^n = \langle \mathbf{f}_k, \mathbf{x} \rangle^n, \tag{9}$$

with

$$\mathbf{f}_k \triangleq \sum_{j=1}^{m} a_{i,k}\, \mathbf{b}_i, \qquad k = 1, \ldots, N \tag{10}$$

a linear combination of the original basis vectors. Equivalently, by stacking the basis (column) vectors into the matrix $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_m]$ and defining $\mathbf{a}_k = [a_{1,k}, \ldots, a_{m,k}]^T$, we can write

$$\mathbf{f}_k = \mathbf{B}\,\mathbf{a}_k, \qquad k = 1, \ldots, N. \tag{11}$$

If the basis vectors $\{\mathbf{b}_i\}$ are viewed as functions with a single "bump" (for example, the delta basis in the time domain, the Fourier basis in the frequency domain, or the wavelet basis in either domain — see Figure 2), then the vectors $\{\mathbf{f}_k\}$ will be functions with multiple "bumps." In this alternative representation, the summing node NST provides an extremely simple structure for generating arbitrary n-th order nonlinear signal interactions. This representation consists of two decoupled subsystems:

1. an overcomplete set of $N = \binom{m+n-1}{n}$ linear filters $\{\mathbf{f}_k\}_{k=1}^{N}$ that control both the system dynamics and component mixing, followed by
2. a set of trivial monomial nonlinearities $(\cdot)^n$.

In Section 4, we will apply this powerful representation of the summing node NST to the Volterra filter implementation problem. The filter bank representation not only leads to a simple and effective representation for the computation of a filter output, but also provides insight into the dynamics of the filter.

Similar to the analysis leading to (8), we make the identification

$$\mathbf{h} = \sum_{k=1}^{N} \alpha_k\, \mathbf{f}_k^{(n)}, \tag{12}$$

It follows that the summing node NST is complete if

$$\text{Span}\left\{ \mathbf{f}_k^{(n)} \right\}_{k=1}^{N} = \text{Span}\{n\text{-th order kernels}\}. \tag{13}$$

Several different constructions for the filters $\{\mathbf{f}_k\}$ provide complete summing node NSTs. Here we state the most general construction. For details see [6].

Theorem 2 *Fix $\rho \in \mathbb{R}$, $|\rho| \neq 1$, $\rho \neq 0$. Set $\gamma_r = \rho^r$, $r = 0, \ldots, n$. Form the collection of $N = \binom{m+n-1}{n}$ length-m vectors $\{\mathbf{a}_k\}_{k=1}^N$ according to*

$$\{\mathbf{a}_k\}_{k=1}^n = \left\{ [\gamma_{l_1}, \ldots, \gamma_{l_m}]^T : \sum_{j=1}^m l_j = n,\ l_j \in \{0, \ldots, n\} \right\}. \tag{14}$$

Then, with $\{\mathbf{a}_k\}_{k=1}^N$ employed in (8) or (11), the condition (13) holds, and the corresponding summing node NST is complete.

3. NSTS IN THE WAVELET DOMAIN

The previous section has shown that complete nonlinear signal transformations can be derived from any orthonormal signal basis $\mathbf{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_m\}$. For example, $\mathbf{B}$ may be a time, Fourier, or wavelet domain basis. We will now show that wavelet-based NSTs offer a significant theoretical advantage. The motivation for wavelet-based NSTs is developed for infinite-dimensional (continuous) spaces (note that until now we have focused on finite-dimensional signal spaces). The properties of wavelet-based NSTs in the infinite-dimensional setting carry over to high-dimensional sampled spaces.

It has been shown that noise removal, compression, and signal recovery methods based on wavelet coefficient shrinkage or wavelet series truncation enjoy asymptotic minimax performance characteristics and do not introduce excessive artifacts in the signal reconstruction [5]. The theoretical justification for the exceptional performance of wavelet-based processing is the fact that wavelet bases are *unconditional bases* for many signal spaces.

It is well-known that wavelet bases derived from multiresolution transformations are unconditional bases for a diverse variety of signal spaces. However, for the NSTs of interest, tensor spaces are the natural framework to consider. Hence, we would like to establish the unconditionality of tensor product wavelet bases. It should be noted that the tensor wavelet basis, also referred to as the "rectangular wavelet decomposition" [7], is quite different from the usual multidimensional wavelet basis obtained via a multiresolution analysis.

The theorem below, proved in [6], shows that the tensor product of a wavelet basis is an unconditional basis for the tensor space $L_p(\mathbb{R}) \otimes_{\Delta_p} L_p(\mathbb{R})$, which is isometric to the space of 2-d L_p functions.

Theorem 3 *If $\{\phi_i\}$ is an unconditional wavelet basis for $L_p(\mathbb{R})$, $1 < p < \infty$, then $\{\phi_i \otimes \phi_j\}$ is an unconditional basis for $L_p(\mathbb{R}) \otimes_{\Delta_p} L_p(\mathbb{R})$.*

This result can easily be extended to arbitrary n-th order tensor spaces, and shows that wavelet-based NSTs correspond to an unconditional basis expansion of the nonlinear signal coefficients. It should be possible to extend this result to more general spaces, including various smoothness spaces. One possible starting point for the general problem may be found in [11].

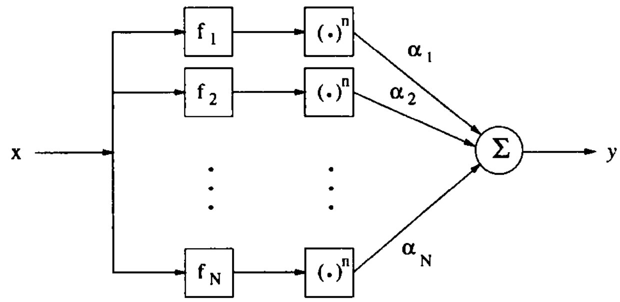

Figure 3: *Volterra filter realization using a summing node NST.*

4. APPLICATION TO VOLTERRA FILTERING

In this Section, we consider Volterra filter realizations based on the NST. We show that a complete n-th order NST is capable of realizing every n-th order Volterra filter. In particular, the summing node transformation leads to an elegant filter bank representation.

The output of a homogeneous n-th order Volterra filter applied to a signal $\mathbf{x} = [x_1, \ldots, x_m]^T$ is given by [8]

$$y = \sum_{1 \le i_1 \le \cdots \le i_n \le m} h_{i_1, \ldots, i_n} x_{i_1} \cdots x_{i_n}. \tag{15}$$

The filter output y is simply an n-th order multilinear combination of the samples $x_1, \ldots, x_m$. The set of weights h is called the n-th order *Volterra kernel*. Note that while (15) computes only a single output value given m input values, the extension to online processing of infinite-length signals is straightforward. To treat the input signal x_l, we simply set $\mathbf{x}_l = [x_l, \ldots, x_{l-m+1}]^T$, with m the *memory length* of the filter. The output of (15) is then y_l, a nonlinearly filtered version of x_l.

Since (15) is identical to the multilinear functional (3) appearing in Definition 1, it follows that every n-th order Volterra filter can be computed as a linear combination of the nonlinear signal coefficients $\boldsymbol{\theta} = F_n(\mathbf{x})$. As shown in Section 2, both the product node and summing node structures are capable of computing a complete n-th order signal transformation. The summing node structure is particularly interesting in this application, because it allows us to represent every n-th order Volterra filter using the simple filter bank of Figure 3. Key to this scheme is that *the overcomplete linear transformation, rather than the nonlinearities, manage the signal coupling prescribed by the overall Volterra filter.* Therefore, this new representation greatly simplifies the analysis, synthesis, and implementation of Volterra filters.[2]

Volterra filter realizations of this type are often referred to as *parallel-cascade realizations* [10]. Previous work on parallel-cascade designs has relied on complicated numerical optimizations to construct kernel-specific sets of linear filters and hence a separate parallel-cascade structure for each distinct Volterra filter [2, 10]. In contrast, the summing

[2]The canonical representation of the Volterra filter (15) is of limited utility, due to the inherent difficulty in interpreting the multidimensional kernel h (particularly when $n > 2$).

node NST can represent every n-th order Volterra filter simply by adjusting the output weights $\{\alpha_k\}_{k=1}^N$. The linear filters $\{\mathbf{f}_k\}_{k=1}^N$ of the summing node structure remain the same for every Volterra kernel. Hence, the summing node structure is a ***universal structure*** for homogeneous Volterra filtering. Nonhomogeneous Volterra filters can also be implemented with the summing node structure by following each linear filter with an n-th degree polynomial nonlinearity instead of the homogeneous n-th order monomial.

The weights $\{\alpha_k\}$ corresponding to a specific Volterra filter with kernel h can be computed by solving a system of linear equations. Let $\mathbf{h}$ be a vectorized version of h ordered to correspond to the Kronecker product in (6). According to (12), the Volterra kernel generated by the summing node NST is given by $\sum_{k=1}^N \alpha_k \mathbf{f}_k^{(n)}$. Therefore, to represent the Volterra filter with kernel $\mathbf{h}$ we choose the weights $\{\alpha_k\}$ so that $\sum_{k=1}^N \alpha_k \mathbf{f}_k^{(n)} = \mathbf{h}$. The proper weights are readily obtained by solving this system of linear equations.

As an example, consider the implementation of a homogeneous third-order ($n = 3$) Volterra filter using the summing node NST. Let $\mathbf{B} = \{\mathbf{b}_j\}_{j=1}^m$ be an orthonormal basis for $\mathbb{R}^m$. For example, $\mathbf{B}$ could be the delta, Fourier, or wavelet basis. We design the filters $\mathbf{f}_1, \ldots, \mathbf{f}_N$, $N = \binom{m+2}{3}$, for the summing node transformation using the construction of Theorem 2. Referring to the Theorem, we take $\rho = 2$ and hence $\gamma_0 = 1, \gamma_1 = 2, \gamma_2 = 4, \gamma_3 = 8$. Each filter $\mathbf{f}_k$, $k = 1, \ldots, N$, is a linear combination of the basis vectors: $\mathbf{f}_k = \mathbf{B}\mathbf{a}_k$, with $\mathbf{a}_k$ a vector with elements in the set $\{1, 2, 4, 8\}$. Each $\mathbf{a}_k$ consists of all 1s except for either a single 8, a 2 paired with a 4, or three 2s. Raising the output of each filter to the third power generates third-order interactions between the different distinct components of the input signal represented by the basis vectors. Taken together, these filters collaborate to generate all possible third-order nonlinear interactions of the signal.

Different types of interactions are produced depending on the choice of basis. The delta basis produces interactions between different time samples of the signal. The Fourier basis yields frequency intermodulations, whereas the wavelet basis produces interactions between wavelet atoms localized in both time and frequency. The fact that wavelet tensor bases are unconditional bases for many tensor spaces suggests that wavelets may provide a more parsimonious representation for Volterra filters than time- or frequency-domain representations.

5. CONCLUSIONS

In this paper, we have developed two new structures for computing n-th order NSTs. The product and summing node NSTs, while simple, can represent all n-th order nonlinear signal interactions. Both transformations have an elegant interpretation in terms of tensor spaces. The summing node NST results in a redundant filter bank structure natural both for analyzing and interpreting nonlinear interactions and for designing efficient implementations. Not only does the summing node architecture suggest new, efficient algorithms for nonlinear processing, it also decouples the processing into linear dynamics and static nonlinearities.

NSTs are not constrained to a fixed choice of basis. However, we have shown that wavelet bases provide an optimal framework for NSTs in the sense that wavelet tensor bases are unconditional for many important tensor spaces. Because the wavelet basis provides a more concise representation of many real-world signals, Volterra kernels and higher-order moments and kernels can be more efficiently represented with wavelets as compared to time- or frequency-domain approaches. We have focused on the classical L_p tensor spaces in our theoretical analysis of wavelet-domain nonlinear processing. However, new results in the statistical literature suggest that more general spaces such as Besov and Triebel spaces are extremely useful for characterizing real-world signals [5]. Therefore, an important avenue for future work will be to extend the results of this paper to these more general settings, possibly using the results of [11].

6. REFERENCES

[1] J. W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Trans. Circuits Syst.*, Vol. 25, pp. 772-781, 1978.

[2] H.-H. Chiang, C. L. Nikias, and A. N. Venetsanopoulos, "Efficient implementations of quadratic filters," *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. 34, pp. 1511-1528, 1986.

[3] K. C. Chou and L. P. Heck, "A multiscale stochastic modelling approach to the monitoring of mechanical systems," *IEEE Int. Symp. Time-Frequency and Time-Scale Analysis*, Philadelphia, pp. 25-27, 1994.

[4] I. Daubechies, *Ten Lectures on Wavelets*, SIAM, 1992.

[5] D. L. Donoho, "Unconditional bases are optimal bases for data compression and for statistical estimation," *Applied Comp. Harm. Anal.*, pp. 100-115, vol. 1, no. 1, Dec. 1993.

[6] R. D. Nowak and R. G. Baraniuk, "Wavelet-based transformations fo r nonlinear signal processing," submitted to *IEEE Trans. Signal Proc.*, February, 1997.

[7] B. Jawerth and W. Sweldens, "An Overview of Wavelet Based Multiresolution Analysis," *SIAM Rev.*, vol.36, no. 3, pp.377-412, 1994.

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

[9] C. L. Nikias and A. P. Petropulu, *Higher-Order Spectra Analysis. A Nonlinear Signal Processing Framework*, Prentice-Hall, 1993.

[10] T. M. Panicker and V.J. Mathews, "Parallel-cascade realizations and approximations of truncated Volterra systems," *Proceedings of the 1996 IEEE Intl. Conf. Acoustics, Speech, and Signal Proc.*, pp. 1589-1592, Atlanta, 1996.

[11] W. Wang, "A discrete transform and Triebel-Lizorkin spaces on the bidisc," *Trans. Amer. Math. Soc.*, vol. 347, no. 4, Apr. 1996.