FAST ORTHOGONAL DECOMPOSITION OF VOLTERRA CUBIC KERNELS USING OBLIQUE UNFOLDING

Rémy Boyer

Université Paris XI (UPS)
LSS-Supélec, CNRS
remy.boyer@lss.supelec.fr

Roland Badeau

Télécom ParisTech
CNRS LTCI
rbadeau@enst.fr

Gérard Favier

Université Nice Sophia Antipolis
I3S, CNRS
favier@i3s.unice.fr

ABSTRACT

Discrete-time Volterra modeling is a central topic in many application areas and a large class of nonlinear systems can be modeled using high-order Volterra series. The problem with Volterra series is that the number of parameters grows very rapidly with the order of the nonlinearity and the memory in the system. In order to efficiently implement this model, kernel eigen-decomposition can be used in the context of a Parallel-Cascade realization of a Volterra system. So, using the multilinear SVD (HOSVD) for decomposing high-order Volterra kernels seems natural. In this paper, we propose to drastically reduce the computational cost of the HOSVD by (1) considering the symmetrized Volterra kernel and (2) exploiting the column-redundancy of the associated mode by using an oblique unfolding of the Volterra kernel. Keeping in mind that the complexity of the full HOSVD for a cubic $(I \times I \times I)$ unstructured Volterra kernel needs $12I^4$ flops, our solution allows reducing the complexity to $2I^4$ flops, which leads to a gain equal to six for a sufficiently large size I.

Keywords: Volterra kernel, fast HOSVD, oblique unfolding

1. INTRODUCTION

Specific applications that need nonlinear structures are encountered in many different areas, in particular in mobile communication, image processing, in geophysical and biomedical signal processing. Volterra series [1] can be used to represent a broad class of nonlinearities. There exists a plethora of identification techniques adapted to the Volterra model (see [2–5]). But the main drawback is the huge number of parameters needed to characterize the Volterra kernel. There are different ways to reduce the parametric complexity of Volterra models and a major challenge is to decompose the Volterra kernel efficiently. A first approach is to expand this kernel using orthonormal basis functions like Laguerre functions. Another approach consists in representing the Volterra model in a parallel-cascade form resulting from the singular value decomposition of an unfolded matrix representation of the kernel. In this class of methods, we can find two families: the matrix-based approach [6] and the tensor-based approaches [7,8]. In the second family, tensor-based decompositions are used as for instance the PARAFAC-Volterra [8] and the multilinear SVD (HOSVD) [7, 9]. More precisely, we extend the work initiated in references [7,8] by exploiting the column-redundancies existing in the mode of symmetric or symmetrized Volterra kernels [10]. This permits to strongly reduce the computational cost of the parallel-cascade realization (PCR) of the Volterra model.

2. VOLTERRA MODELS AND EIGEN DYNAMIC MODES (EDM)

2.1. Definition of the model

Volterra series constitute a model for systems which yield generalized Taylor series expansions. The input/output relationship for a discrete-time time-invariant nonlinear causal system can be expressed as

$$y(n) = \sum_{m=1}^{M} y_m(n) = \sum_{m=1}^{M} \langle \mathbb{H}_m^*, \mathbb{X}(n) \rangle \quad (1)$$

where $*$ denotes complex conjugation, $y(n)$ is the system's output for the n-th discrete observation, M is the order of the Volterra model and the nonlinearity degree, $\langle ., . \rangle$ stands for the inner product. The data tensor for the n-th observation is given by $\mathbb{X}(n) = \mathbf{x}(n) \circ \ldots \circ \mathbf{x}(n)$ where $\circ$ stands for the outer product, and $\mathbf{x}(n) = [x(n)\, x(n-1) \ldots x(n-I+1)]^T$ in which ^{T} stands for matrix transposition. In model (1), the $(I \times \ldots \times I)$ tensor $\mathbb{H}_m$ is called the m-dimensional Volterra kernel and describes the dynamics of the system. I is the memory length of the m-th order homogeneous term $y_m(n)$. Expanding the inner product, we obtain

$$
\begin{aligned}
\langle \mathbb{H}_m^*, \mathbb{X}(n) \rangle &= \sum_{k_1, \ldots, k_m = 0}^{I-1} [\mathbb{H}_m]_{k_1 \ldots k_m} [\mathbb{X}(n)]_{k_1 \ldots k_m} \\
&= \sum_{k_1, \ldots, k_m = 0}^{I-1} [\mathbb{H}_m]_{k_1 \ldots k_m} x(n-k_1) \ldots x(n-k_m)
\end{aligned}
$$

which represents a multidimensional convolution of the input signal $x(n)$ with m-th order Volterra kernel. When $M = 3$, the above model is called a cubic Volterra model. In the sequel, we often consider this case but the presented results can be straightforwardly generalized to $M > 3$.

2.2. Multilinear SVD (HOSVD) for cubic Volterra models

2.2.1. Definition of the HOSVD

Every complex $(I \times I \times I)$-tensor $\mathbb{H}_3$ can be written as the product [9]

$$\mathbb{H}_3 = \mathbb{Q} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3, \quad (2)$$

in which $\times_m$ denotes the m-mode product [9], $\mathbf{U}_1$, $\mathbf{U}_2$ and $\mathbf{U}_3$ are unitary matrices and $\mathbb{Q}$ is an all-orthogonal and ordered complex tensor. This decomposition is a generalization of the matrix SVD because the diagonality of the matrix containing the singular

values, in the matrix case, is a special case of all-orthogonality. Also, the HOSVD of a second-order tensor (matrix) yields the matrix SVD, up to trivial indeterminacies. For the third-order tensor $\mathbb{H}_3$, the $I \times I^2$ unfolded matrix representations can be obtained as

$$
\begin{aligned}
[\mathbf{H}_1]_{k_1, k_3 I + k_2} &= [\mathbb{H}_3]_{k_1 k_2 k_3}, \\
[\mathbf{H}_2]_{k_2, k_3 I + k_1} &= [\mathbb{H}_3]_{k_1 k_2 k_3}, \\
[\mathbf{H}_3]_{k_3, k_1 I + k_2} &= [\mathbb{H}_3]_{k_1 k_2 k_3}.
\end{aligned}
$$

An alternative (but equivalent) definition with respect to the k-th frontal slice, denoted by $\boldsymbol{\mathcal{H}}_{k_3} = [\mathbb{H}_3]_{:,:,k_3}$, is

$$
\begin{aligned}
\mathbf{H}_1 = \text{unfold}_1\{\mathbb{H}_3\} &= \begin{bmatrix} \boldsymbol{\mathcal{H}}_0 & \dots & \boldsymbol{\mathcal{H}}_{I-1} \end{bmatrix}, \\
\mathbf{H}_2 = \text{unfold}_2\{\mathbb{H}_3\} &= \begin{bmatrix} \boldsymbol{\mathcal{H}}_0^T & \dots & \boldsymbol{\mathcal{H}}_{I-1}^T \end{bmatrix}, \\
\mathbf{H}_3 = \text{unfold}_3\{\mathbb{H}_3\} &= \begin{bmatrix} \text{vec}\left(\boldsymbol{\mathcal{H}}_0^T\right) & \dots & \text{vec}\left(\boldsymbol{\mathcal{H}}_{I-1}^T\right) \end{bmatrix}^T
\end{aligned}
$$

where $\text{vec}(.)$ creates a $I^2 \times 1$ vector from a $I \times I$ matrix by stacking the column vectors of this matrix below one another. For a general definition of unfolded matrices in case of higher dimensions, the interested reader can see [9] for instance. The matrix of m-mode singular vectors, $\mathbf{U}_m$, can be found as the matrix of left singular vectors of the unfolded matrix representation $\mathbf{H}_m$.

Based on the unitary matrices $\mathbf{U}_1$, $\mathbf{U}_2$ and $\mathbf{U}_3$, the core tensor is given by

$$
\mathbb{Q} = \mathbb{H}_3 \times_1 \mathbf{U}_1^H \times_2 \mathbf{U}_2^H \times_3 \mathbf{U}_3^H \tag{3}
$$

where ^{H} denotes the conjugate transpose of a matrix.

2.2.2. Complexity in flops

The computational costs presented in this paper are related to the flop (*floating point operation*) count. For example, a dot product of I-dimensional vectors approximately involves $2I$ flops (I multiplications plus $I-1$ additions). Using the GR-SVD method [11], the complexity of the full HOSVD of a $(I \times I \times I)$-tensor is evaluated to $12I^4$ flops [10].

2.3. Eigen dynamic modes (EDM) and Parallel-Cascade realization

Plugging the HOSVD given in (2) into the model (1), we obtain

$$
y_3(n) = \sum_{k_1, k_2, k_3 = 0}^{I-1} \underbrace{q_{k_1 k_2 k_3} \, w_{k_1}(n) w_{k_2}(n) w_{k_3}(n)}_{y_{k_1 k_2 k_3}(n)}
$$

in which $q_{k_1 k_2 k_3} = [\mathbb{Q}]_{k_1 k_2 k_3}$, $w_{k_m}(n) = \langle \mathbf{u}_{k_m}^*, \mathbf{x}(n) \rangle = \mathbf{u}_{k_m}^T \mathbf{x}(n)$ where $\mathbf{u}_{k_m} = [\mathbf{U}_m]_{k_m}$ is the k_m-th column of matrix $\mathbf{U}_m$. So, according to this expression a PCR of the Volterra model is obtained as the parallelization and sum of the following block diagram:
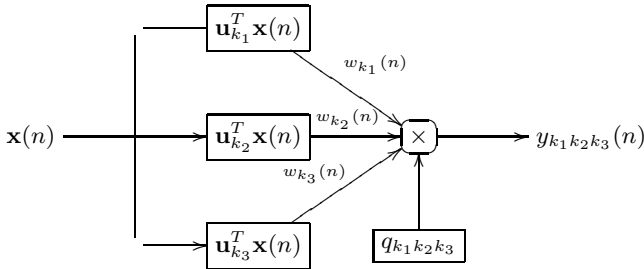
3. FAST HOSVD FOR THE CUBIC VOLTERRA KERNEL

Our method is based on the exploitation of the symmetry of the Volterra kernel. A kernel is said to be symmetric if the indices can be interchanged without affecting its value. More precisely, a m-th order tensor $\mathbb{S}$ which is unchanged by any permutation π is called a *symmetric* tensor: $\forall k_1, \dots, k_m \in \{0, \dots, I-1\}$, $[\mathbb{S}]_{k_{\pi(1)}, \dots, k_{\pi(m)}} = [\mathbb{S}]_{k_1 \dots k_m}$. In practice, we have two situations explained in the next section.

3.1. On the symmetry for the Volterra kernel

3.1.1. Example of a symmetric kernel

The kernel of the Volterra model associated with the Wiener-Hammerstein model is already symmetric. Specifically, this model is formed of a polynomial defined by the coefficients $\{c_1, \dots, c_m\}$, enclosed between two linear filters of impulse responses $r(.)$ and $g(.)$, and memories M_r and M_g respectively. The associated Volterra kernel has been derived in [12]:

$$
[\mathbb{H}_m]_{k_1, \dots, k_m} = c_m \sum_{i=0}^{M_g - 1} g(i) \prod_{u=1}^{m} r(k_u - i),
$$

where $k_1, \dots, k_m \in \{0, \dots, M_v\}$ and $M_v = M_r + M_g - 1$ is the memory of the nonlinear plant, which is assumed to be known. If $M_r = 1$, a Hammerstein model is selected whereas $M_g = 1$ corresponds to a Wiener model. It is clear that $\mathbb{H}_m$ is a symmetric tensor since $r(k_1 - i)r(k_2 - i) \dots r(k_m - i)$ is invariant under permutation.

3.1.2. Existence of a symmetrized kernel

If the kernel is general in the sense that there is no symmetric relations between its entries, there always exists an associated symmetrized kernel computed according to

$$
[\mathbb{S}_m]_{k_1 \dots k_m} = \frac{\alpha}{m!} \sum_{\pi \in \mathcal{P}} [\mathbb{H}_m]_{k_{\pi(1)} \dots k_{\pi(m)}} \tag{4}
$$

where ! stands for the factorial notation, *ie.*, $m! = m(m-1)\dots 2.1$, and $\mathcal{P}$ is the permutation set of cardinal $m!/\alpha$ with $\alpha = n_1! \dots n_r!$, where r is the number of distinct values in the set $\{k_1, \dots, k_m\}$ and $n_1 \dots n_r$ is the occurrence of each index value. It follows that for any tensor $\mathbb{X}(n)$, $\langle \mathbb{H}_m^*, \mathbb{X}(n) \rangle = \langle \mathbb{S}_m^*, \mathbb{X}(n) \rangle$. The complexity of this operation is $(1 + m!/\alpha)I^m$ flops.

3.2. Orthogonal tensor decomposition for a symmetric tensor

In the sequel, we consider the third-order Volterra model ($M = 3$) but the proposed method can be easily extended to higher orders. In the following developments, we no longer mention index m in order to simplify the notations.

3.2.1. Multilinear SVD (HOSVD) for symmetric tensors

As a special case of the general definition of the HOSVD given in section 2.2.1, every complex *symmetric* $(I \times I \times I)$-tensor $\mathbb{S}$ can be written as the product:

$$
\mathbb{S} = \mathbb{Q} \times_1 \mathbf{U} \times_2 \mathbf{U} \times_3 \mathbf{U}, \tag{5}
$$

in which $\mathbf{U}$ is an unitary matrix and $\mathbb{Q}$ is an all-orthogonal and ordered complex tensor.

3.2.2. Unfolding of a third-order symmetrized/symmetric tensor

The three modes of a third-order symmetric tensor are all equal and are given by

$$\mathbf{S} = \text{unfold}_1\{\mathbb{S}\} = \text{unfold}_2\{\mathbb{S}\} = \text{unfold}_3\{\mathbb{S}\}. \quad (6)$$

Thus, $\mathbf{S}$ is a fat $I \times I^2$ matrix (more columns than rows) and, as we show in the next section, it is *column redundant* [10].

3.3. Column-redundancy of the mode for a symmetrized or symmetric tensor

#### 3.3.1. Definition of the compressed mode

Let us begin by an example. Let $\mathbb{H}$ be a $2 \times 2 \times 2$ tensor defined as
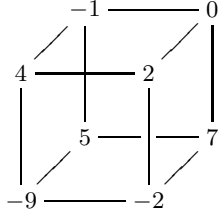
The modes (unfolded matrices) are given by

$$\mathbf{H}_1 = \text{unfold}_1\{\mathbb{H}\} = \left[\begin{array}{cc|cc} 4 & 2 & -1 & 0 \\ -9 & -2 & 5 & 7 \end{array} \right],$$

$$\mathbf{H}_2 = \text{unfold}_2\{\mathbb{H}\} = \left[\begin{array}{cc|cc} 4 & -9 & -1 & 5 \\ 2 & -2 & 0 & 7 \end{array} \right],$$

$$\mathbf{H}_3 = \text{unfold}_3\{\mathbb{H}\} = \left[\begin{array}{cc|cc} 4 & 2 & -9 & -2 \\ -1 & 0 & 5 & 7 \end{array} \right].$$

Note that tensor $\mathbb{H}$ has no particular structure. This is also true for the modes. Now, compute the symmetrized associated tensor $\mathbb{S}$ defined according to (4) where the cardinal of the permutation group is $3!/(1!2!) = 3$ according to
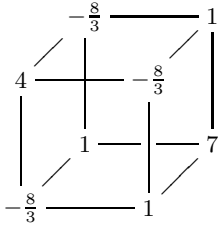
Then, its single mode is given by

$$\mathbf{S} = \left[\begin{array}{cc|cc} 4 & -\frac{8}{3} & -\frac{8}{3} & 1 \\ -\frac{8}{3} & 1 & 1 & 7 \end{array} \right]. \quad (7)$$

Let us now consider the general case of symmetric tensors of arbitrary dimension. The mode associated to a symmetric tensor admits an axial blockwise symmetry [10] and thus $\mathbf{S}$ is column redundant. This is a consequence of the symmetry of tensor $\mathbb{S}$. Specifically, some columns in $\mathbf{S}$ are repeated twice.

Let $\mathbf{S}' \triangleq \mathbf{SJ}$ define the $I \times J$ compressed mode where $\mathbf{J}$ is a selection matrix which cancels the column-redundancy in mode

$\mathbf{S}$ (symbol $\triangleq$ stands for definition). The number of columns of the compressed mode is now reduced to

$$J \triangleq \frac{(I+1)I}{2} < I^2. \quad (8)$$

The derivation of the selection matrix is not straightforward for any size I but a systematic computation of the compressed mode can be obtained following an oblique unfolding, denoted by o-unfold(.), of tensor $\mathbb{S}$, described in Fig. 1, according to

$$\mathbf{S}' = \text{o-unfold}(\mathbb{S}) = \begin{bmatrix} \mathbf{T}_0 & \cdots & \mathbf{T}_{I-1} \end{bmatrix} \quad (9)$$

where $[\mathbf{T}_{k_3}]_{k_1,k_2} = [\mathbb{S}]_{k_1,k_2,k_3+k_2}$ ($\forall i \in \{0,\dots,I-1\}$, the dimensions of $\mathbf{T}_i$ are $I \times (I-i)$).

3.3.2. Weighted compressed mode

To compensate the missing columns in the compressed mode, we introduce a diagonal matrix

$$\mathbf{D} = \text{diag}\{d_0,\dots,d_{J-1}\} \quad (10)$$

which takes the redundancy of each column in the mode into account. The weighting factors are given by

$$d_k = \begin{cases} 1 \text{ if } 0 \le k < I, \\ 2 \text{ if } I \le k < J. \end{cases} \quad (11)$$

This means that if the k-th column is repeated twice then $d_k = 2$. In case of no repetition, $d_k = 1$. We call matrix $\mathbf{S}'\mathbf{D}^{1/2}$ the weighted compressed mode.

Let us comeback to the example. Reorganize the columns of the mode according to

$$\mathbf{S}^{(\pi)} = \left[\begin{array}{cc|c|c} 4 & 1 & -\frac{8}{3} & -\frac{8}{3} \\ -\frac{8}{3} & 7 & 1 & 1 \end{array} \right]. \quad (12)$$

where $^{(\pi)}$ means that the columns of $\mathbf{S}$ have been permuted. It is clear that the last columns are repeated. To mitigate this problem, consider the oblique unfolding introduced in (9) and compute the compressed mode $\mathbf{S}'$ with $J = 3$ according to

$$\mathbf{S}' = \text{o-unfold}(\mathbb{S}) = \begin{bmatrix} \mathbf{T}_0 & \mathbf{T}_1 \end{bmatrix} \quad (13)$$

where $\mathbf{T}_0 = \begin{bmatrix} 4 & 1 \\ -\frac{8}{3} & 7 \end{bmatrix}$ and $\mathbf{T}_1 = \begin{bmatrix} -\frac{8}{3} \\ 1 \end{bmatrix}$. As expected, the repeated columns are removed and the remaining columns are scaled according to the weighting matrix given by $\mathbf{D} = \text{diag}\{1,1,2\}$.

3.3.3. SVD of the compressed mode

The SVD of the compressed mode is given by

$$\mathbf{S}'\mathbf{D}^{1/2} = \mathbf{UGV}^H \quad (14)$$

where $\mathbf{G}$ is a diagonal matrix with non-negative coefficients and $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices. Consequently, the sample covariance of the mode $\mathbf{S}$ verifies

$$\mathbf{R} \triangleq \mathbf{SS}^H = \mathbf{S}'\mathbf{D}^{1/2}(\mathbf{S}'\mathbf{D}^{1/2})^H = \mathbf{UG}^2\mathbf{U}^H. \quad (15)$$

This means that the mode $\mathbf{S}$ and the weighted compressed mode $\mathbf{S}'\mathbf{D}^{1/2}$ share the same unitary basis[1] $\mathbf{U}$. In this way, the

---

[1] The detailed proof can be found in [10].

cost of the HOSVD is reduced to that of the SVD of $\mathbf{S'D}^{1/2}$. Using the GR-SVD method [11], the complexity of the computation of the SVD of an $I \times J$ matrix with $J \gg I$ needs $4JI^2$ flops. For large I, we have $4JI^2 \approx 2I^4$. In particular, it can be noted that the compression and weighting of the modes lead to a complexity approximately 6 times as low as that of the full HOSVD computation, and twice as low as that the HOSVD for symmetric tensors. To illustrate this result, the number of flops is plotted in Fig. 2 with respect to the size of the kernel. In addition, the different complexities are summarized in Table 1. Remark that the computational cost of the symmetrization of the Volterra kernel is dominated by the computational cost of the SVD.
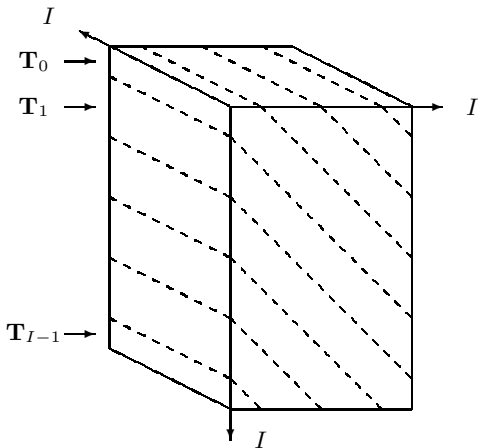

Fig. 1. Oblique unfolding of tensor $\mathbb{S}$

Table 1. Cost of the HOSVD

Operation	Cost per iteration
HOSVD	$12I^4$
Symmetrization (optional)	$7I^3$
HOSVD (symmetric tensor)	$4I^4$
HOSVD (symmetric + compressed mode)	$2I^4$

4. CONCLUSION

Reducing the number of parameters toward a parallel-cascade realization of a high-order Volterra kernel is a hot topic. In this paper, we propose a fast computation of the multilinear SVD (HOSVD) which fully exploits the redundancies of symmetric/symmetrized Volterra kernels. First, for such kernels and without loss of generality, the HOSVD needs the computation of a single SVD of its unique mode (unfolded matrix). This step leads to reduce the computational cost by a factor three in the case of a third-order tensor. Next, a second improvement, which is the main contribution of the paper, is that this mode is column redundant, meaning that some
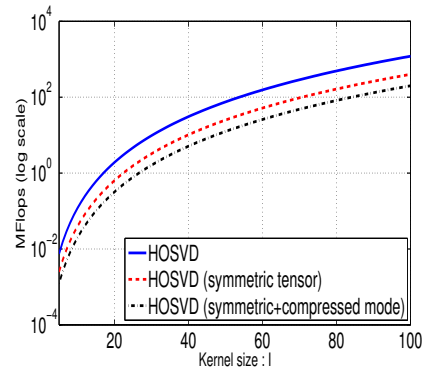

Fig. 2. Number of flops with respect to the Kernel size

columns are repeated twice. Consequently, we present a technique for deriving an SVD of a mode which has less columns than the initial mode since the repeated columns are removed. Finally, we show that this compressed mode, before convenient scaling, is obtained by an oblique unfolding of the symmetric/symmetrized Volterra kernel. This last point allows decreasing the computational cost by a factor two and thus by a final factor equal to six with respect to the direct computation of the HOSVD for the initial Volterra kernel. For the interested reader, further improvements for efficiently computing the HOSVD of structured tensors are presented in [10].

5. REFERENCES

[1] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*, John Wiley and Sons, New York, NY, USA, 1980.

[2] Y.W. Lee and M. Schetzen, "Measurement of the Wiener kernels of a nonlinear system by crosscorrelation," *Int. Journal of Control*, vol. 2, no. 3, pp. 237–254, Sept. 1965.

[3] V. J. Mathews and G. L. Sicuranza, *Polynomial Signal Processing*, John Wiley and Sons, New York, NY, USA, 2000.

[4] G. B. Giannakis and E. Serpedin, "A bibliography on nonlinear system identification," *Signal Processing*, vol. 81, no. 3, pp. 533–580, Mar. 2001.

[5] A. Khouaja and G. Favier, "Identification of Parafac-Volterra cubic models using an alternating recursive least squares algorithm," in *European Signal Processing Conference (EUSIPCO)*, Vienna, Austria, Sept. 2004.

[6] T. M. Panicker and V. J. Mathews, "Parallel-cascade realizations and approximations of truncated Volterra systems," *IEEE Trans. Signal Processing*, vol. 46, no. 10, pp. 2829–2832, Oct. 1998.

[7] E. Seagraves, B. Walcott, and D. Feinauer, "Efficient implementation of Volterra systems using a multilinear SVD," in *International IEEE Symposium on Intelligent Signal Processing and Communication System (ISPACS)*, Xiamen, China, Nov. 2007, pp. 762–765.

[8] G. Favier and T. Bouilloc, "Parametric complexity reduction of Volterra models using tensor decompositions," in *European Signal Processing Conference (EUSIPCO)*, Glasgow, Scotland, Aug. 2009.

[9] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, Apr. 2000.

[10] R. Badeau and R. Boyer, "Fast multilinear singular value decomposition for structured tensors," *SIAM. J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1008–1021, Sept. 2008.

[11] G. H. Golub and C.F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 3rd edition, 1996.

[12] A.Y. Kibangou and G. Favier, "Wiener-Hammerstein systems modeling using diagonal Volterra kernels coefficients," *IEEE Signal Proc. Letters*, vol. 13, no. 6, pp. 381–384, June 2006.