IDENTIFICATION OF PARAFAC-VOLTERRA CUBIC MODELS USING AN ALTERNATING RECURSIVE LEAST SQUARES ALGORITHM

A. Khouaja[†] *and G. Favier*[†]

[†]Laboratoire d'Informatique, Signaux et Systèmes (I3S), CNRS-UNSA, 2000, route des Lucioles, Les Algorithmes - Euclide B, B.P.121, 06903 Sophia-Antipolis, France

(khouaja,favier)@i3s.unice.fr

ABSTRACT

A broad class of nonlinear systems can be modelled by the Volterra series representation. However, its practical use in nonlinear system identification is sometimes limited due to the large number of parameters associated with the Volterra filters structure. This paper is concerned with the problem of identification of third-order Volterra kernels. A tensorial decomposition called PARAFAC is used to represent such a kernel. A new algorithm called the Alternating Recursive Least Squares (ARLS) algorithm is applied to identify this decomposition for estimating the Volterra kernels of cubic systems. This method significantly reduces the computational complexity of Volterra kernel estimation. Simulation results show the ability of the proposed method to achieve a good identification and an important complexity reduction, i.e. representation of Volterra cubic kernels with few parameters.

Keywords : Nonlinear system identification, Volterra models, tensors, PARAFAC models.

1. INTRODUCTION

The identification of nonlinear dynamical systems from a given input output data set has attracted considerable interest since many physical systems exhibit nonlinear characteristics. The Volterra model structure can be used to represent a broad class of nonlinearities. The output of a third-order, homogeneous, discrete-time, time invariant, truncated Volterra cubic filter with input sequence $u(k)$ is given by :

$$\begin{aligned} y(n) &= \sum_{n_1=1}^{\infty}\sum_{n_2=1}^{\infty}\sum_{n_3=1}^{\infty} h_{n_1n_2n_3}u(n-n_1) \\ &\quad \times u(n-n_2)u(n-n_3) \end{aligned} \qquad (1)$$

where $\{h_{n_1n_2n_3}\}$ are the coefficients of the Volterra cubic kernel. In practice, the infinite sum in (1) may be truncated to a finite number if the system has fading memory [1]. It has been shown in [1] that any time-invariant nonlinear system with fading memory can be well approximated by a finite Volterra series representation to any precision. Hence, the class of truncated Volterra series model is attractive to use in nonlinear system identification.

Volterra filters are very simple to use and have nice properties. For instance, they are linear in their parameters and hence standard and well-behaved parameter estimation techniques can be used. However, the large number of parameters associated with the Volterra models limit their practical utility to problems involving only modest values of memory. This limitation arises because the identification of the large number of parameters may be problematic, but also design procedures based upon such models may be cumbersome. To eliminate this drawback, two ways can be followed :

- to arrange the kernel coefficients in matrices that are decomposed in applying a reduced order Singular Value Decomposition (SVD) which leads to a low complexity parallel-cascade realization of the Volterra filter [9],
- to expand the kernel on an orthonormal basis such as the Laguerre functions basis ([2],[4]) or Generalized Orthonormal Bases (GOB) ([5],[7]).

The purpose of this paper is to develop reduced complexity third-order Volterra models identified by means of the Alternating Least Squares (ALS) and the Alternating Recursive Least Squares (ARLS) methods.

Third-order Volterra kernels can be considered as third-order tensors. So, we apply a tensor decomposition called PARAFAC to represent Volterra cubic kernels. The corresponding reduced complexity Volterra model called PARAFAC-Volterra model is presented in section 2. A new Alternating Recursive Least Squares (ARLS) algorithm to estimate the parameters of such Volterra models is presented in section 3. In section 4, we evaluate the performance of this new approach by means of simulations before concluding in section 5.

2. THE PARAFAC-VOLTERRA CUBIC MODEL

The PARAFAC (PARAllel FACtor) also called CANDECOMP (CANonical DECOMPosition) was introduced by Harshman (1970) [6] and by Caroll and Chang (1970) [3] in order to reduce the complexity of an N^{th} order tensor. This decomposition entirely preserves the information contained in the original tensor.

Before defining the PARAFAC model of a third-order $(N_1 \times N_2 \times N_3)$ tensor $\mathbb{H}$, we define the following matrices :

- $H_{n_1..}$ $(n_1 = 1,\ldots,N_1)$ are $(N_2 \times N_3)$ matrices such as $H_{n_1..}(n_2,n_3) = \mathbb{H}(n_1,n_2,n_3)$.
- $H_{.n_2.}$ $(n_2 = 1,\ldots,N_2)$ are $(N_3 \times N_1)$ matrices such as $H_{.n_2.}(n_3,n_1) = \mathbb{H}(n_1,n_2,n_3)$.
- $H_{..n_3}$ $(n_3 = 1,\ldots,N_3)$ are $(N_1 \times N_2)$ matrices such as $H_{..n_3}(n_1,n_2) = \mathbb{H}(n_1,n_2,n_3)$. □

The construction of the matrices $H_{n_1..}$, $H_{.n_2.}$ and $H_{..n_3}$ is described in figure 1.

We also define the unfolded matrices $H_{N_1\times N_2N_3}$, $H_{N_2\times N_1N_3}$ and $H_{N_3\times N_1N_2}$ of the tensor $\mathbb{H}$ as :

$$H_{N_1\times N_2N_3} = \begin{bmatrix} H_{..1} & H_{..2} & \cdots & H_{..N_3} \end{bmatrix} \qquad (2)$$

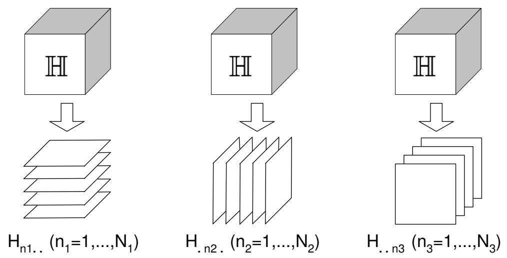

Figure 1: $H_{n_1..}$, $H_{.n_2.}$ and $H_{..n_3}$ construction

$$H_{N_2\times N_1N_3} = [\ H_{1..} \quad H_{2..} \quad \cdots \quad H_{N_1..}\ ] \quad (3)$$
$$H_{N_3\times N_1N_2} = [\ H_{.1.} \quad H_{.2.} \quad \cdots \quad H_{.N_2.}\ ] \quad (4)$$

The PARAFAC model of a third-order $(N_1 \times N_2 \times N_3)$ tensor $\mathbb{H}$ is defined by three matrices A, B and C with respective dimensions $(N_1 \times P)$, $(N_2 \times P)$ and $(N_3 \times P)$. The scalar representation of the model is written as :

$$h_{n_1n_2n_3} = \sum_{p=1}^{P} a_{n_1p}b_{n_2p}c_{n_3p}\ ; \quad \begin{array}{l} n_1 = 1,\ldots,N_1 \\ n_2 = 1,\ldots,N_2 \\ n_3 = 1,\ldots,N_3 \end{array} \quad (5)$$

where $h_{n_1n_2n_3}$ is the element (n_1,n_2,n_3) of the tensor $\mathbb{H}$, a_{n_1p}, the element (n_1,p) of the matrix A, b_{n_2p}, the element (n_2,p) of the matrix B, c_{n_3p}, the element (n_3,p) of the matrix C and P is the number of the PARAFAC model factors.

The determination of P is directly related to the rank of the tensor $\mathbb{H}$. Kruskal [8] defined the number of the PARAFAC model factors P as :

$$P \geq \mathbf{rank}(\mathbb{H}) = \max \left\{ \begin{array}{l} \mathrm{rank}(H_{N_1\times N_2N_3}), \\ \mathrm{rank}(H_{N_2\times N_1N_3}), \\ \mathrm{rank}(H_{N_3\times N_1N_2}) \end{array} \right\} \quad (6)$$

From the scalar representation (5) of the PARAFAC model, we can write its matrix representation using the matrices $H_{n_1..}$, $H_{.n_2.}$ and $H_{..n_3}$:

$$H_{n_1..} = BD^A_{n_1}C^T \text{ with } D^A_{n_1} = \mathbf{diag}\,(A_{n_1.}) \quad (7)$$
$$H_{.n_2.} = CD^B_{n_2}A^T \text{ with } D^B_{n_2} = \mathbf{diag}\,(B_{n_2.}) \quad (8)$$
$$H_{..n_3} = AD^C_{n_3}B^T \text{ with } D^C_{n_3} = \mathbf{diag}\,(C_{n_3.}) \quad (9)$$

where, $(A_{n_1.})$, $(B_{n_2.})$ and $(C_{n_3.})$ are respectively the n_1^{th}, the n_2^{th} and the n_3^{th} row of the matrices A, B and C. $D^C_{n_3}$ is the $(P \times P)$ diagonal square matrix the diagonal elements of which are the elements of the row vector $(C_{n_3.})$.

Using the scalar representation of the PARAFAC model, the output of the truncated Volterra model in equation (1) can be written as follows :

$$\begin{aligned} \widehat{y}(n) &= \sum_{n_1=1}^{M}\sum_{n_2=1}^{M}\sum_{n_3=1}^{M}\left(\sum_{p=1}^{P} a_{n_1p}b_{n_2p}c_{n_3p}\right)\times \\ &\quad u(n-n_1)u(n-n_2)u(n-n_3) \quad (10) \\ &= \sum_{p=1}^{P}\left(\sum_{n_1=1}^{M} a_{n_1p}\,u(n-n_1)\right) \\ &\quad \times\left(\sum_{n_2=1}^{M} b_{n_2p}\,u(n-n_2)\right) \\ &\quad \times\left(\sum_{n_3=1}^{M} c_{n_3p}\,u(n-n_3)\right) \quad (11) \end{aligned}$$

The input/output relation (11) can be implemented in using a parallel-cascade structure, as shown in figure 2.

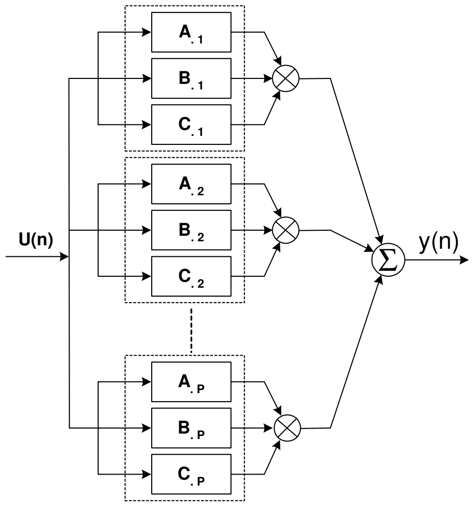

Figure 2: A parallel-cascade structure realization of the PARAFAC-Volterra cubic model

In figure 2, $U(n) = [\ u(n-1) \quad \cdots \quad u(n-M)\ ]^T$, $A_{.i}$, $B_{.j}$ and $C_{.k}$ represent respectively the i^{th} column of A, the j^{th} column of B and the k^{th} column of C, and the boxes of stage p design the convolution operations $U^T(n)A_{.p}$, $U^T(n)B_{.p}$ and $U^T(n)C_{.p}$.

The PARAFAC decomposition has the properties to be unique and applicable to any tensor. The Volterra cubic kernel in (1) can be viewed as a third-order $(M \times M \times M)$ tensor with a complexity $C_{standard} = M^3$ in terms of its coefficients number.

The PARAFAC-Volterra complexity is $C_{parafac} = 3MP$. So, the Ratio of Complexity Reduction (RCR) using PARAFAC with respect to the standard Volterra cubic model is $RCR = \frac{3MP}{M^3} = \frac{3P}{M^2}$. When $P \ll M$, a significant complexity reduction can be achieved.

From the RCR, we can conclude that a complexity reduction is possible when $P < \frac{M^2}{3}$.

3. THE ARLS ALGORITHM

This algorithm uses the scalar representation of the PARAFAC model given by (5) and is based on an alternating procedure. It updates the matrices A, B and C by minimizing the following cost function η :

$$\eta(n) = \sum_{t=1}^{n}\left(y(t) - \widehat{y}(t)\right)^2 \quad (12)$$

where $y(n)$ denotes the output of the Volterra system and $\widehat{y}(n)$ denotes the output of the model based on the PARAFAC decomposition and defined in (11). Let :

$$\varphi_p^A(n) = \sum_{n_2=1}^{M} b_{n_2 p} u(n-n_2) \sum_{n_3=1}^{M} c_{n_3 p} u(n-n_3) \quad (13)$$

$$\varphi_p^B(n) = \sum_{n_1=1}^{M} a_{n_1 p} u(n-n_1) \sum_{n_3=1}^{M} c_{n_3 p} u(n-n_3) \quad (14)$$

$$\varphi_p^C(n) = \sum_{n_1=1}^{M} a_{n_1 p} u(n-n_1) \sum_{n_2=1}^{M} b_{n_2 p} u(n-n_2) \quad (15)$$

$$\Phi_A(n) = \begin{bmatrix} \varphi_1^A(n) & \varphi_2^A(n) & \cdots & \varphi_P^A(n) \end{bmatrix}^T \quad (16)$$

$$\Phi_B(n) = \begin{bmatrix} \varphi_1^B(n) & \varphi_2^B(n) & \cdots & \varphi_P^B(n) \end{bmatrix}^T \quad (17)$$

$$\Phi_C(n) = \begin{bmatrix} \varphi_1^C(n) & \varphi_2^C(n) & \cdots & \varphi_P^C(n) \end{bmatrix}^T \quad (18)$$

and

$$U(n) = \begin{bmatrix} u(n-1) & \cdots & u(n-M) \end{bmatrix}^T \quad (19)$$

By supposing that the matrices B and C are known, equation (11) is then written as follows :

$$\begin{aligned} \widehat{y}(n) &= \sum_{p=1}^{P} \sum_{n_1=1}^{M} a_{n_1 p} u(n-n_1) \varphi_p^A(n) && (20) \\ &= \sum_{p=1}^{P} (a_{1p} u(n-1) + a_{2p} u(n-2) + \cdots \\ &\quad \cdots + a_{Mp} u(n-M)) \varphi_p^A(n) && (21) \\ &= (a_{11} u(n-1) + \cdots + a_{M1} u(n-M)) \varphi_1^A(n) \\ &+ \cdots && (22) \\ &+ (a_{1P} u(n-1) + \cdots + a_{MP} u(n-M)) \varphi_P^A(n) \\ &= (\mathbf{vec}(A))^T \underbrace{\Phi_A(n) \otimes U(n)}_{P_A(n)} && (23) \\ &= P_A^T(n)\, \mathbf{vec}(A) && (24) \end{aligned}$$

and the cost function $\eta(n)$ becomes :

$$\eta_A(n) = \sum_{t=1}^{n} \left[y(t) - P_A^T(t)\, \mathbf{vec}(A) \right]^2 \quad (25)$$

By minimizing $\eta_A(n)$ with respect to the matrix A we get the following estimated solution in the recursive least squares (RLS) sense :

$$\mathbf{vec}(\widehat{A}(n)) = \mathbf{vec}(\widehat{A}(n-1)) + K_A(n) \varepsilon_A(n) \quad (26)$$

where

$$\varepsilon_A(n) = y(n) - P_A^T(n)\, \mathbf{vec}(\widehat{A}(n-1)) \quad (27)$$

$$K_A(n) = \frac{Q_A(n-1) P_A(n)}{1 + P_A^T(n) Q_A(n-1) P_A(n)} \quad (28)$$

$$Q_A(n) = [I - K_A(n) P_A^T(n)] Q_A(n-1) \quad (29)$$

Similarly, by supposing known respectively the matrices A and C and the matrices A and B, the cost function $\eta(n)$ is written :

$$\eta_B(n) = \sum_{t=1}^{n} \left[y(t) - P_B^T(t)\, \mathbf{vec}(B) \right]^2 \quad (30)$$

$$\eta_C(n) = \sum_{t=1}^{n} \left[y(t) - P_C^T(t)\, \mathbf{vec}(C) \right]^2 \quad (31)$$

1. **Initialization**
   - $\widehat{A}(0)$, $\widehat{B}(0)$ et $\widehat{C}(0)$.
   - $Q_A(0) = Q_B(0) = Q_C(0) = I_{MP}$.
2. **Updating of the PARAFAC components**
   - Calculate $P_A(n)$, $P_B(n)$ et $P_C(n)$
   - $\begin{cases} \varepsilon_A(n) = y(n) - P_A^T(n)\mathbf{vec}(\widehat{A}(n-1)) \\ \varepsilon_B(n) = y(n) - P_B^T(n)\mathbf{vec}(\widehat{B}(n-1)) \\ \varepsilon_C(n) = y(n) - P_C^T(n)\mathbf{vec}(\widehat{C}(n-1)) \end{cases}$
   - $\begin{cases} K_A(n) = \frac{Q_A(n-1)P_A(n)}{1+P_A^T(n)Q_A(n-1)P_A(n)} \\ K_B(n) = \frac{Q_B(n-1)P_B(n)}{1+P_B^T(n)Q_B(n-1)P_B(n)} \\ K_C(n) = \frac{Q_C(n-1)P_C(n)}{1+P_C^T(n)Q_C(n-1)P_C(n)} \end{cases}$
   - $\begin{cases} Q_A(n) = [I - K_A(n)P_A^T(n)]Q_A(n-1) \\ Q_B(n) = [I - K_B(n)P_B^T(n)]Q_B(n-1) \\ Q_C(n) = [I - K_C(n)P_C^T(n)]Q_C(n-1) \end{cases}$
   - $\begin{cases} \mathbf{vec}(\widehat{A}(n)) = \mathbf{vec}(\widehat{A}(n-1)) + K_A(n)\varepsilon_A(n) \\ \mathbf{vec}(\widehat{B}(n)) = \mathbf{vec}(\widehat{B}(n-1)) + K_B(n)\varepsilon_B(n) \\ \mathbf{vec}(\widehat{C}(n)) = \mathbf{vec}(\widehat{C}(n-1)) + K_C(n)\varepsilon_C(n) \end{cases}$
3. **Reconstruction of the cubic kernel**
   - $\widehat{h}_{n_1 n_2 n_3} = \sum_{p=1}^{P} \widehat{a}_{n_1 p} \widehat{b}_{n_2 p} \widehat{c}_{n_3 p}\ ; \quad \begin{matrix} n_1 = 1, \ldots, M \\ n_2 = 1, \ldots, M \\ n_3 = 1, \ldots, M \end{matrix}$
4. **Go back to step 2 until convergence of the algorithm**

Table 1: The ARLS algorithm

where $P_B(t)$ and $P_C(t)$ are constructed in a same manner as $P_A(t)$.

By alternatively minimizing the cost functions η_A, η_B and η_C, we update the estimated matrices A, B and C that are used to represent the PARAFAC-Volterra cubic model. The equations of the corresponding algorithm called Alternating Recursive Least Squares algorithm (ARLS), are summarized in table 1.

4. SIMULATION RESULTS

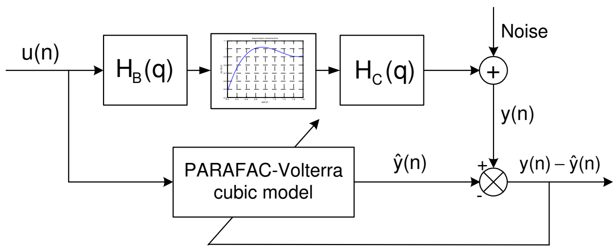

Figure 3: Application of PARAFAC-Volterra cubic model to identify a nonlinear satellite channel

The simulated system is a simplified model of a nonlinear satellite channel [10] represented in figure 3. The channel filters the input signal $u(k)$ by a low-pass linear filter denoted by $H_B(z)$, then the signal passes through a memoryless nonlinear device defined by its input-output characteristic $A(r)$ represented in figure 3 and defined as a third order polynomial. In the last stage the signal passes through another low-pass filter $H_C(z)$. In this example, $H_B(z)$ is a Butterworth filter and $H_C(z)$ a Chebychev filter. Both are fourth order and respectively defined in equations (32) and (33).

$$H_B(z) = \frac{(0.078+0.1559z^{-1}+0.078z^{-2})(0.0619+0.1238z^{-1}+0.0619z^{-2})}{(1-1.3209z^{-1}+0.6327z^{-2})(1-1.0486z^{-1}+0.2961z^{-2})} \quad (32)$$

$$H_C(z) = \frac{(0.4638-0.4942z^{-1}+0.4638z^{-2})(0.183+0.1024z^{-1}+0.183z^{-2})}{(1-1.2556z^{-1}+0.6891z^{-2})(1-0.7204z^{-1}+0.1888z^{-2})} \quad (33)$$

The input signal is a gaussian white noise sequence of length $N = 20000$ with zero mean and unit variance. The simulation results were obtained using the Monte Carlo method with 50 different additive noise sequences.

The Normalized Mean Square Error (NMSE_{output}) between the system output $y(n)$ and the output of the PARAFAC-Volterra model $\widehat{y}(n)$ is calculated as follows :

$$NMSE_{output}(n) = \frac{\sum_{i=1}^{n}(y(i)-\widehat{y}(i))^2}{\sum_{i=1}^{n} y^2(i)} \quad (34)$$

Table 2 contains the NMSE_{output} values between the system output $y(n)$ and the output $\widehat{y}(n)$ of the PARAFAC-Volterra model as a function of the Signal to Noise Ratio (SNR) and the PARAFAC factors number P. The evolution of the NMSE_{output} for different P values is plotted in figure 4. The PARAFAC-Volterra method allows to modelize the nonlinear satellite channel with relatively small modeling error.

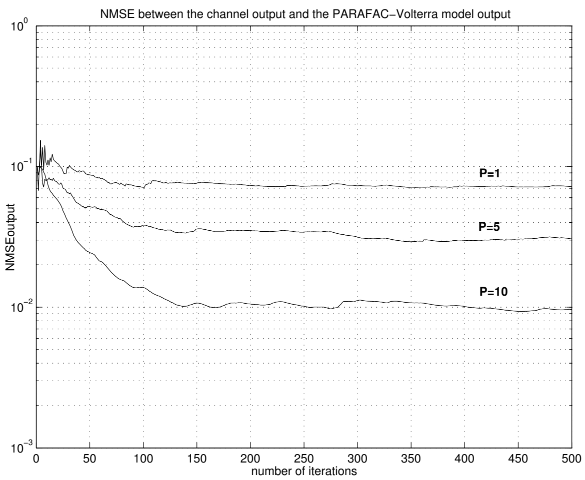

Figure 4: NMSE_{output} obtained with the PARAFAC-Volterra model for three different factors numbers P (SNR=30dB)

	SNR=60	SNR=30	SNR=5
P=1	$5.895\ 10^{-2}$	$7.191\ 10^{-2}$	$2.812\ 10^{-1}$
P=5	$2.329\ 10^{-2}$	$3.060\ 10^{-2}$	$6.974\ 10^{-2}$
P=10	$8.283\ 10^{-3}$	$9.641\ 10^{-3}$	$2.856\ 10^{-2}$

Table 2: NMSE_{output} obtained with the PARAFAC-Volterra model

5. CONCLUSION

In this paper, we have presented a new approach to represent and identify third-order Volterra kernels using the PARAFAC decomposition, which significantly reduces the parametric complexity of such kernels, especially when they are separable. The ARLS algorithm is used to identify such a decomposition.

Extension of this work to Volterra kernels of order higher than three is under study.

REFERENCES

[1] S. Boyd and L.O. Chua, "Fading memory and the problem of approximating nonlinear operators with Volterra series", *IEEE Tr. Circuits and Systems*, **32**, 11, pp. 1150-1171, **1985.**

[2] R. J. G. B. Campello, G. Favier and W. C. Amaral, "Optimal expansions of discrete-time Volterra models using Laguerre functions", *IFAC Symp. SYSID*, pp. 1844-1849, Rotterdam, The Netherlands, **2003.**

[3] J.D. Carroll and J. J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition", *Psychometrika*, **35**, pp. 283-319, **1970.**

[4] G. A. Dumont and Y. Fu, "Nonlinear adaptive control via Laguerre expansion of Volterra kernels", *Int. J. Adaptive Control and Signal Processing*, **7**, pp. 367-382, **1993.**

[5] G. Favier, A. Kibangou and R. J. G. B. Campello, "Nonlinear systems modelling by means of generalized orthonormal basis functions", *Invited paper, IEEE Conference on Signals, Systems, Decision and information technology, SSD'03*, Sousse, Tunisia, **2003.**

[6] R. A. Harshman, "Foundation of the PARAFAC procedure : Models and conditions for an "explanatory" multimodal factor analysis", *UCLA working papers in phonetics*, **16**, pp. 1-84, **1970.**

[7] A. Kibangou, G. Favier and M. M. Hassani, "A growing approach for selecting generalized orthonormal basis functions in the context of system modeling", *Proc. IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing, NSIP'03*, Grado, Italy, **2003.**

[8] J. B. Kruskal, "Three-way arrays : Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics", *Linear algebra and its applications*, **18**, pp. 95-138, **1977.**

[9] T. M. Panicker and V. J. Mathews, "Parallel-cascade realizations and approximations of truncated Volterra systems", *IEEE Trans. Signal Processing*, **46**, 10, pp. 2829-2832, **1998.**

[10] A. A. M. Saleh, "Frequency independent and frequency dependent nonlinear models of TWT amplifiers", *IEEE Tr. on Communications*, **29**, 11, pp. 1715-1720, **1981.**