Nonlinear homogeneous order separation for Volterra Series Identification

Damien Bouvier, Thomas Hélie, David Roze

NONLINEAR HOMOGENEOUS ORDER SEPARATION FOR VOLTERRA SERIES IDENTIFICATION

Damien Bouvier, Thomas Hélie, David Roze

S3AM team, IRCAM-CNRS-UPMC UMR 9912
1, place Igor Stravinsky, 75004 Paris, France
damien.bouvier@ircam.fr

ABSTRACT

This article addresses identification of nonlinear systems represented by Volterra series. To improve the robustness of some existing methods, we propose a pre-processing stage that separates nonlinear homogeneous order contributions from which Volterra kernels can be identified independently. The proposed separation method exploits phase relations between test signals rather than amplitude relations that are usually used. This method is compared with standard separation process. Its contribution to identification is illustrated on a simulated loudspeaker with nonlinear suspension.

1. INTRODUCTION

Volterra series are a representation formalism for dynamical system with memory and weak nonlinearities. In audio, it has been widely used in various applications such as simulation of nonlinear resonators (brass instruments [1], Moog ladder filter [2], etc), distortion effect [3,4], audio transducer analysis [5]. For any purpose (analysis, simulation or even system control), the use of Volterra series has to begin with the computation of the Volterra kernels, either directly from the system's dynamical equation [2], or by identification from inputs/outputs measurement [6].

This paper focuses on the identification problem in blind context, that is, without assuming any particular structures (Hammerstein, Wiener-Hammerstein, etc) [7,8] or parametric model [9]. To this end, we propose a pre-processing stage that separates nonlinear homogeneous order contributions from which Volterra kernels can be identified independently. Such an approach is commonly used (see [10–14]), but the separation process relies on amplitude discrimination, that rapidly leads to ill-conditioned problems. To improve robustness, a new method is proposed that also exploits phase relations. This pre-processing method is embedded into a kernel identification method proposed in [6].

This paper is organized as follows. Section 2 gives an overview of Volterra Series and standard order separation. In Section 3, the proposed separation method is presented, and its advantages and disadvantages are discussed. Finally, in Section 4, simulation of a loudspeaker with nonlinear suspension is used to compare the separation methods and their contribution to identification.

2. RECALLS ON VOLTERRA SERIES AND ORDER SEPARATION

2.1. Overview on Volterra Series

An overview of the Volterra formalism is given here; further and more thorough explanations can be found in [15, 16], among the vast literature on Volterra series.

Definition 1 (Volterra series)**.** A nonlinear causal time-invariant system is described by a Volterra series $\{h_n\}_{n\geq\mathbb{N}^*}$ if, for all input signals u such that $\|u\|_\infty < \rho$, the output signal y is given by the following Volterra operator:

$$y = V[u] = \sum_{n=1}^{\infty} y_n \tag{1}$$

where, for continuous-time systems:

$$y_n(t) = \int_{\mathbb{R}_+^n} h_n(\tau_1, \dots, \tau_n) \prod_{i=1}^{n} u(t-\tau_i)\mathrm{d}\tau_i \tag{2}$$

and for discrete-time systems:

$$y_n[l] = \sum_{\mathbb{N}^n} h_n[m_1, \dots, m_n] \prod_{i=1}^{n} u[l-m_i] \tag{3}$$

and with ρ the convergence radius of the characterising function $\Phi_h(x) = \sum_{n=1}^{+\infty} \|h_n\|_1 x^n$. The terms h_n are called the *Volterra kernels* of the system, and terms y_n the *nonlinear homogeneous order contributions* (or more simply the nonlinear orders).

In the following, for sake of notation, continuous-time signals and systems will be used; if not specified otherwise, results are also valid for their discrete-time counterparts.

Remark 1 (Convergence)**.** It has been shown in [17, 18] that there exists a large class of well-posed systems for which we know how to compute the convergence radius of the Volterra series. In this work, we will always assume that convergence's conditions are met.

Remark 2 (Order and memory truncation)**.** In numerical implementation for simulation or identification, it will be necessary to truncate both infinite sums (i.e. for nonlinear orders and memory). Thus, in practice, Volterra series can only be used to approximate systems with small nonlinearities (limited to the first few nonlinear orders) and with finite memory [19].

Remark 3 (Non-unicity of kernels)**.** It can easily be seen from (2) that kernels are not uniquely defined. To circumvent this problem for identification purpose, uniquely-defined forms can be specified, such as the triangular or symmetric kernels (where the kernel is invariant to any permutation of its arguments).

Remark 4 (Frequency domain kernels)**.** As it is common for linear filters, it is possible to work in the frequency domain by means of a Laplace or Fourier transform.

2.2. Volterra operator properties

In the following definition, $L^p(\mathbb{E},\mathbb{F})$ denotes the standard Lebesgue-space of functions from vector spaces $\mathbb{E}$ to $\mathbb{F}$ with p-norm.

Definition 2 (Volterra operator of order n)**.** Let $h_n \in L^1(\mathbb{R}_+^n,\mathbb{R})$ be a Volterra kernel, for $n \in \mathbb{N}^*$. Then we introduce the multilinear operator $V_n : L^\infty(\mathbb{R},\mathbb{R}) \times \ldots \times L^\infty(\mathbb{R},\mathbb{R}) \mapsto L^\infty(\mathbb{R},\mathbb{R})$ such that function $V_n[u_1,\ldots,u_n]$ is defined $\forall t \in \mathbb{R}$ by

$$V_n[u_1,\ldots,u_n](t) = \int_{\mathbb{R}_+^n} h_n(\tau_1,\ldots,\tau_n) \prod_{i=1}^{n} u_i(t-\tau_i)\mathrm{d}\tau_i \quad (4)$$

If $u_1 = \ldots = u_n = u$, $V_n[u,\ldots,u] = y_n$.

Property 1 (Symmetry)**.** *Given a symmetric kernel h_n, the corresponding Volterra operator V_n is also symmetric, meaning*

$$V_n\big[u_{\pi(1)},\ldots,u_{\pi(n)}\big](t) = V_n[u_1,\ldots,u_n](t) \quad (5)$$

for any permutations π and $\forall t \in \mathbb{R}$.

In the following, symmetry of h_n and V_n will be supposed.

Property 2 (Multilinearity and homogeneity)**.** *Volterra operator V_n is multilinear, i.e. for any signals u_1, u_2, and any scalars λ, μ,*

$$V_n[\lambda u_1 + \mu u_2,\ldots,\lambda u_1 + \mu u_2](t) = \sum_{q=0}^{n} \binom{n}{q} \lambda^{n-q}\mu^q V_n\big[\underbrace{u_1,\ldots,u_1}_{n-q}\underbrace{u_2,\ldots,u_2}_{q}\big](t) \quad (6)$$

This also implies that V_n is a homogeneous operator of degree n, i.e. for any signal u and scalar α,

$$V_n[\alpha u,\ldots,\alpha u](t) = \alpha^n V_n\big[u,\ldots,u\big](t) \quad (7)$$

2.3. State-of-the-art order separation

Nonlinear homogeneous order separation implies the ability to recover signals y_n from the output y of a system described by a Volterra series truncated to order N.

From (7), $V[\alpha u](t) = \sum_{n=1}^{N} \alpha^n y_n(t)$. Consider a collection of input signals $u_k(t) = \alpha_k u(t)$, with $\alpha_k \in \mathbb{R}^*, k = 1,\ldots,N$ and note $z_k(t) = V[u_k](t)$ their corresponding output through the system; then, for all time t:

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix}(t) = \begin{bmatrix} \alpha_1 & \alpha_1^2 & \ldots & \alpha_1^N \\ \alpha_2 & \alpha_2^2 & \ldots & \alpha_2^N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_N & \alpha_N^2 & \ldots & \alpha_N^N \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}(t) \ , \ \alpha_k \in \mathbb{R}^*$$
$$\boldsymbol{Z}(t) = \boldsymbol{A} \cdot \boldsymbol{Y}(t) \quad (8)$$

Since $\boldsymbol{A}$ is a Vandermonde matrix, it is invertible if and only if all α_k are different; hence it is possible to recover terms y_n.

But for real-valued α_k, this type of matrix is known for becoming rapidly ill-conditioned when its size grows (meaning small noise in the measured outputs would become a large error in the estimation); so robustness decreases rapidly when the truncation order N increases. In order to circumvent that, it is possible to solve the linear problem $\boldsymbol{Z} = \boldsymbol{A}\boldsymbol{Y}$ by using a Newton Recursive or Lagrange Recursive method [20, Algorithm 4.6.1 and 4.6.2]. But in practice, this approach is still very sensitive to the choice of amplitude factors α_k. Indeed, for small amplitudes, higher orders will be hidden in measurement noise, while high values of α_k will potentially overload the system, or simply lead it out of its Volterra convergence radius.

Despite those disadvantages, this separation method has been used intensively for simplifying identification process [13]; it is generally used in frequency domain (the previous equations and remarks remains valid) and jointly with frequency probing methods [10,11]; recently, a maximum order truncation estimation method has been constructed from it [14]. In the following, this method will be referred to as the *Amplitude Separation* (AS) method.

3. PHASE-BASED HOMOGENEOUS SEPARATION METHOD

The starting point of the proposed separation method are the following remarks:

- using the AS method with factor 1 and -1, it is possible to separate odd and even orders by inverting a mixing matrix $\boldsymbol{A}$ with optimum condition number;
- multiplying a signal by amplitude factor -1 is equivalent to taking its opposite phase;
- thus, for a system truncated to order $N = 2$, there exists a robust separation method that relies only on *phase* deconstruction and reconstruction between tests signals.

The main idea of this paper is to generalize the use of phase for robust separation method to Volterra systems with truncation $N > 2$.

3.1. Method for complex-valued input

This section proposes a theoretical separation method relying on the use of complex signals $u(t) \in \mathbb{C}$ as system inputs.

3.1.1. Principle

Using complex signals, factors α_k in the AS method are not limited to real scalar. So it would be possible to choose values which only differs by their phase (e.g. are on the unit circle instead of the real axis). Noticing that 1 and -1 are the two square root of unity, a natural extension of the toy-method proposed for order-2 systems would be to take the N-th roots of unity as factors α_k. Choosing $\alpha_k = w_N^k$ with $w_N = e^{j\frac{2\pi}{N}}$, (8) becomes, for all time t:

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix}(t) = \begin{bmatrix} w_N & w_N^2 & \ldots & w_N^N \\ w_N^2 & w_N^4 & \ldots & w_N^{2N} \\ \vdots & \vdots & \ddots & \vdots \\ w_N^N & w_N^{2N} & \ldots & w_N^{N^2} \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}(t) \ , \ w_N = e^{j\frac{2\pi}{N}}$$
$$\boldsymbol{Z}(t) = \boldsymbol{W_N} \cdot \boldsymbol{Y}(t) \quad (9)$$

where $\boldsymbol{W_N}$ is the Discrete Fourier Transform (DFT) matrix of order N (after a column and row permutation[1]). It is important to note that here the DFT does not apply on time but on the homogeneous nonlinear orders.

Since the DFT is invertible, order separation is possible. Furthermore, the DFT matrix is well-conditioned[2], and the solution

[1] It suffices to consider vectors $\widehat{\boldsymbol{Z}}(t) = [z_N, z_1, \ldots, z_{N-1}]^T$ and $\widehat{\boldsymbol{Y}} = [y_N, y_1, \ldots, y_{N-1}]^T$ to recover the usual DFT matrix.

[2] Its condition number is even optimum, since it is 1 for any order N.

can be computed using the Fast Fourier Transform algorithm[3]. In the following, this method will be referred to as the *Phase Separation* (PS) method.

3.1.2. Nonlinear order aliasing and rejection factor

Given the N-periodicity of N-th root of unity w_n, the ouptut of a Volterra system with no order truncation is:

$$V[w_N u](t) = \sum_{n=1}^{N} w_N^n \sum_{r=0}^{\infty} y_{n+rN}(t) \tag{10}$$

By applying the PS method, estimation $\widetilde{\boldsymbol{Y}}$ of nonlinear orders 1 to N yields:

$$\widetilde{\boldsymbol{Y}}(t) = \begin{bmatrix} y_1 + \sum_{r=1} y_{1+rN} \\ y_2 + \sum_{r=1} y_{2+rN} \\ \vdots \\ y_N + \sum_{r=1} y_{N+rN} \end{bmatrix}(t) \tag{11}$$

Equation (11) reveals that estimation $\widetilde{y}_n$ is perturbed by a residual term $\sum_{r=1} y_{n+rN}$, which is structured as an aliasing with respect to the nonlinear order: we[4] call this effect the *nonlinear order aliasing*.

For a band-limited input signal, presence of term y_{n+k} in the estimated signal $\widetilde{y}_n$ means presence of higher-frequency components than expected; therefore, this artefact can help detect a wrong truncation order N. But moreover, (11) permits to create higher-order rejection by using amplitude as a *contrast factor*. Taking $\alpha_k = \rho w_N^k$, where ρ is a positive real number less than 1, estimation $\widetilde{\boldsymbol{Y}}$ using PS method becomes

$$\widetilde{\boldsymbol{Y}}(t) = \begin{bmatrix} \rho & & 0 \\ & \rho^2 & \\ & \ddots & \\ 0 & & \rho^N \end{bmatrix} \begin{bmatrix} y_1 + \sum_{r=1} \rho^{rN} y_{1+rN} \\ y_2 + \sum_{r=1} \rho^{rN} y_{2+rN} \\ \vdots \\ y_N + \sum_{r=1} \rho^{rN} y_{N+rN} \end{bmatrix}(t), \tag{12}$$

creating a ρ^N ratio between desired signal $y_n(t)$ and the first perturbation $y_{n+N}(t)$. Thus parameters N and ρ enables to reach a required Signal-to-Noise Ratio (SNR).

However, the need of complex input (and output) signals prevents the use of PS method in practice.

3.2. Application to real-valued input

Consider the real signal $v(t)$ constructed as follows:

$$v(t) = w\,u(t) + \overline{w\,u(t)} = 2\,\mathrm{Re}[w\,u(t)] \tag{13}$$

where w is a complex scalar on the unit circle (such that $\overline{w} = w^{-1}$) and $u(t)$ a complex signal. Therefore, using property (6), and assuming symmetry for the operator V_n (meaning symmetry for kernel h_n), the order n contribution is:

$$\begin{aligned} y_n(t) &= V_n[v, \dots, v](t) \\ &= V_n\big[w\,u + \overline{w\,u}, \dots, w\,u + \overline{w\,u}\big](t) \\ &= \sum_{q=0}^{n} \binom{n}{q} w^{n-q} \overline{w}^q V_n\big[\underbrace{u, \dots, u}_{(n-q)\text{ times}}, \underbrace{\overline{u}, \dots, \overline{u}}_{q\text{ times}}\big](t) \\ &= \sum_{q=0}^{n} \binom{n}{q} w^{n-2q} M_{n,q}(t) \end{aligned} \tag{14}$$

with $M_{n,q}(t) = V_n\big[\underbrace{u, \dots, u}_{n-q}, \underbrace{\overline{u}, \dots, \overline{u}}_{q}\big](t) \in \mathbb{C}$. This term represents the homogeneous contribution of order n for an equivalent multi-input system excited by combinations of u and $\overline{u}$.

Remark 5. By symmetry of V_n, there is $M_{n,q}(t) = \overline{M_{n,n-q}(t)}$ and term $M_{n,n/2}(t)$, for even n, is real. Therefore, from sum over q in (14), realness of $y_n(t)$ is recovered.

Remark 6. By their definition, terms $M_{n,0}(t)$ (respectively $M_{n,n}(t)$) are homogeneous contribution of order n of the system excited by the complex signal $u(t)$ (resp. $\overline{u}(t)$).

Equation (14) shows that, in the output term y_n, there is more than one characterising phase factor w. So PS method is not directly exploitable to separate terms y_n.

3.3. Method for real-valued input

Difficulty analysis on an example: Consider a system truncated to $N = 3$ with input the real signal described in (13); then, omitting temporal dependency, its nonlinear orders are:

$$\begin{aligned} y_1 = &\quad w M_{1,0} &+ &\quad w^{-1} M_{1,1} \\ y_2 = &\quad w^2 M_{2,0} &+ &\quad 2\,M_{2,1} &+ &\quad w^{-2} M_{2,2} \\ y_3 = w^3 M_{3,0} &+ &\quad 3\,w M_{3,1} &+ &\quad 3\,w^{-1} M_{3,2} &+ &\quad w^{-3} M_{3,3} \end{aligned} \tag{15}$$

Only 7 different phase terms appears (from w^{-3} to w^3). Consider a collection of $K = 7$ real signals $v_k(t) = w_K^k\,u(t) + \overline{w_K^k\,u(t)}$, where w_K is the first K-th root of unity, and $z_k(t) = V[v_k](t)$ their corresponding output through the system. Then:

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{bmatrix}(t) = \boldsymbol{W_K} \cdot \begin{bmatrix} M_{2,1} \\ M_{1,0} + 3M_{3,1} \\ M_{2,0} \\ M_{3,0} \\ M_{3,3} \\ M_{2,2} \\ M_{1,1} + 3M_{3,2} \end{bmatrix}(t)\,, \tag{16}$$

where $\boldsymbol{W_K}$ is the DFT matrix[5] of order K.

Therefore, by application of the PS algorithm (i.e. inversion of $\boldsymbol{W_K}$) on this of signals z_k, the right-hand side vector in (16) is recovered. Further separation could be made using two different amplitudes to discriminate between $M_{1,0}$ and $M_{3,1}$, and thus be able to reconstruct nonlinear orders y_n using (14).

[3] Even if the gain in time computation is not significant since generally N is not a power of 2 and is not very high.

[4] We and the anonymous reviewer that proposed this relevant expression.

[5] It is important to notice that the right-hand side vector has hermitian symmetry, due to its Fourier Transform (the left-hand side) being real.

Generalization: Using (14), it is possible to generalize phase grouping shown in (15) as follows (see Appendix 7 for detailed computation):

$$y(t) = \sum_{\substack{-N\leq p\leq N\\ p\text{ even}}} w^p \sum_{\substack{1<|p|\leq n\leq N\\ n\text{ even}}} \binom{n}{\frac{n-p}{2}} M_{n,\frac{n-p}{2}}(t) + \sum_{\substack{-N\leq p\leq N\\ p\text{ odd}}} w^p \sum_{\substack{|p|\leq n\leq N\\ n\text{ odd}}} \binom{n}{\frac{n-p}{2}} M_{n,\frac{n-p}{2}}(t) \quad (17)$$

So, by applying a PS algorithm of order $K = 2N + 1$, we can separate the terms

$$Q_p(t) = \sum_{\substack{|p|\leq n\leq N\\ n\equiv p \mod 2}} \binom{n}{\frac{n-p}{2}} M_{n,\frac{n-p}{2}}(t) \quad (18)$$

with $-N \leq p \leq N$. Then, application of the AS algorithm on each Q_p (with $\lfloor N/2 \rfloor$ amplitudes) gives all $M_{n,q}$ terms; terms y_n are reconstructed using (14).

This concatenation of PS and AS algorithm constitutes the proposed phase-based method, which will be referred to as *Phase-Amplitude Separation* (PAS) method. As will be pointed out in Section 4.3, it can be more interesting to use directly terms $M_{n,q}$ for identification, instead of nonlinear orders y_n; this alternative process will be referred to as raw-PAS (rPAS) method.

3.4. Condition number

In numerical analysis, condition number κ measures the method's sensibility to noise in the measured data; it only depends on the solving method itself (the matrix to invert in linear problems), and not the data it is applied to. In this section, condition number is used to compare AS and PAS robustness[6].

For AS method, amplitude factors α_k are chosen equally-spaced in dB scale, with alternating signs:

$$\alpha_{2p} = \gamma^{p-1} \text{ and } \alpha_{2p+1} = -\alpha_{2p} \quad (19)$$

where γ is a chosen spacing gain.

Because the DFT matrix is optimally conditioned, PAS method overall conditioning only depends on the K applications of AS method. Amplitude factors are also chosen equally-spaced in dB scale; the need to separate terms with same order-parity prevents us from using alternating signs. Only the worst condition number for all K sub-problems is reported.

Figure 1 presents conditioning for AS and PAS methods. For all maximum order truncation and gain spacing, an improvement from amplitude-based to phase-based method is visible. Furthermore, for negative gain spacing, optimum conditioning is divided by a factor 2 between AS and PAS. But, for both methods, the same behaviour when truncation order N increases is remarked (which is unsurprising due to the fact that PAS relies partly on AS).

Those results indicates that PAS robustness to noisy measurements should be better than AS.

[6] As PAS and rPAS methods share the same steps, their condition number are equal.

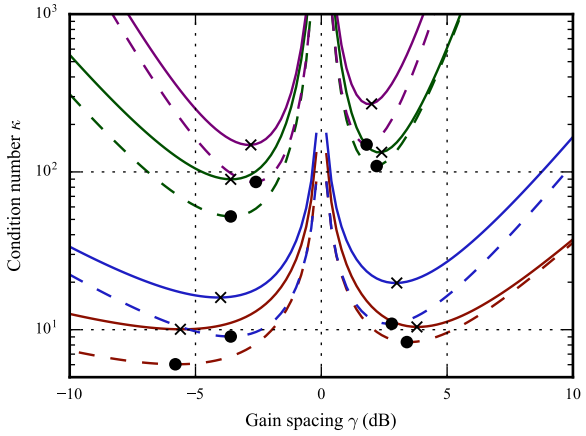


Figure 1: Condition number $\kappa = \|\boldsymbol{A}\|\|\boldsymbol{A^{-1}}\|$ (Frobenius norm): evolution w.r.t gain spacing γ and truncation order N (3 to 6, from bottom to top) for AS (solid line, $\times$ indicating minima) and PAS (dashed line, $\bullet$ indicating minima) methods.

4. APPLICATIONS

In order to test and compare the different separation method, data from simulation of a nonlinear device were used.

4.1. Simulation of a loudspeaker with nonlinear suspension

The simulated system was a loudspeaker with nonlinear suspension described by the modified Thiele & Small following equations:

$$u(t) = R_\mathrm{e} i(t) + L\frac{\mathrm{d}i(t)}{\mathrm{d}t} + Bl\frac{\mathrm{d}\ell(t)}{\mathrm{d}t} \quad (20a)$$

$$M\frac{\mathrm{d}^2\ell(t)}{\mathrm{d}t^2} = Bl\, i(t) - R_\mathrm{m}\frac{\mathrm{d}\ell(t)}{\mathrm{d}t} - \sum_{n=1}^{3} k_n \ell^n(t) \quad (20b)$$

where u is the voltage at the loudspeakers terminals, i the current flowing through it and ℓ the position of the diaphragm. The term $\sum_{n=1}^{3} k_n \ell^n(t)$ corresponds to the nonlinear force that the suspension applies on the diaphragm.

Consider the state vector $\boldsymbol{x} = \left[i \;\; \ell \;\; \mathrm{d}(\ell)/\mathrm{d}t\right]^T$. Then, using state-space formalism, the system of input $u(t)$ and output $i(t)$ is written:

$$\begin{cases} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u + \boldsymbol{K}_2(\boldsymbol{x},\boldsymbol{x}) + \boldsymbol{K}_3(\boldsymbol{x},\boldsymbol{x},\boldsymbol{x}) \\ i &= \boldsymbol{C}\boldsymbol{x} \end{cases} \quad (21)$$

where $\dot{\boldsymbol{x}}$ indicates the temporal derivative of $\boldsymbol{x}$, and:

$$\boldsymbol{A} = \begin{bmatrix} -\frac{R_\mathrm{e}}{L} & 0 & -\frac{Bl}{L} \\ 0 & 0 & 1 \\ \frac{Bl}{M} & -\frac{k_1}{M} & -\frac{R_\mathrm{m}}{M} \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} \frac{1}{L} \\ 0 \\ 0 \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

and where $\boldsymbol{K}_2$ et $\boldsymbol{K}_3$ are multilinear functions of $\boldsymbol{x}$ such that:

$$\boldsymbol{K}_2(\boldsymbol{a},\boldsymbol{b}) = \begin{bmatrix} 0 \\ 0 \\ -\frac{k_2}{M}a_2 b_2 \end{bmatrix}, \text{ and } \boldsymbol{K}_3(\boldsymbol{a},\boldsymbol{b},\boldsymbol{c}) = \begin{bmatrix} 0 \\ 0 \\ -\frac{k_3}{M}a_2 b_2 c_2 \end{bmatrix}$$

R_e	$5,7\ \Omega$	R_m	$4,06 \cdot 10^{-1}\ \mathrm{Nm^{-1}s}$
L	$1,1 \cdot 10^{-1}\ \mathrm{H}$	k_1	$912,3\ \mathrm{kg \cdot s^{-2}}$
Bl	$2,99\ \mathrm{NA^{-1}}$	k_2	$611.5\ \mathrm{kg \cdot m^{-1}s^{-2}}$
M	$1,9 \cdot 10^{-3}\ \mathrm{kg}$	k_3	$8,0 \cdot 10^{7}\ \mathrm{kg \cdot m^{-2}s^{-2}}$

Table 1: Thiele & Small parameters of SICA loudspeaker model Z000900, given by constructor [21] (apart from Bl and the k_n, which were measured experimentally).

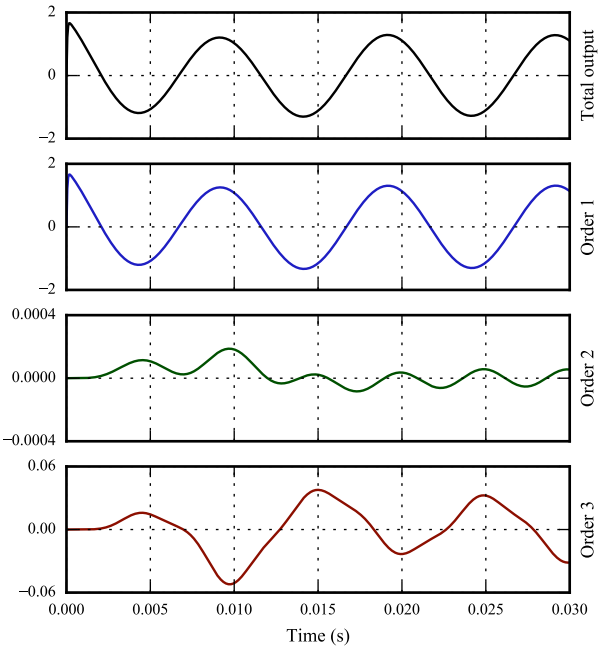


Figure 2: Total output and first three nonlinear orders y_n of the simulated loudspeaker for a cosinusoidal input of frequency 100 Hz and amplitude 10 V.

Simulations were made using numerical methods described in Appendix 8, with parameters given in Table 1.

4.2. Error separation

Order separation method are compared using the relative error ϵ_n of each order n for noisy measurements of system outputs, where

$$\epsilon_n = \frac{\mathrm{RMS}(\widetilde{y}_n[k] - y_n[k])}{\mathrm{RMS}(y_n[k])} \tag{22}$$

with y_n the true nonlinear homogeneous contribution, $\widetilde{y}_n$ their estimation, and RMS the standard Root-Mean Square measure.

Simulation were made at a sampling frequency of 20000 Hz, and with a truncation order $N = 5$. All signals were 1 second long.

Figure 2 shows the total output and first three nonlinear orders of the system. Its linear behaviour is predominant: y_1 is one (respectively four) magnitude order above y_3 (resp. y_2). This large difference of amplitude between signals y (or equivalently y_1) and y_2 means that the quadratic order could be hidden by noise in relatively high SNR.

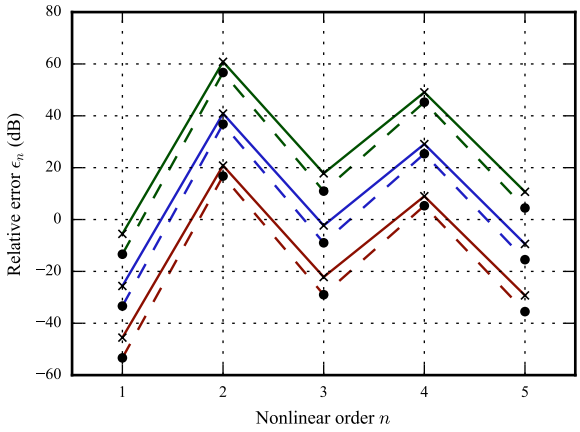


Figure 3: Relative error ϵ_n (in dB) of separation w.r.t. nonlinear order n and SNR (80 dB, 60 dB and 40 dB from bottom to top) for AS (solid line with $\times$) and PAS (dashed line with $\bullet$) methods. Input signals are linear sweep of amplitude 10 V going from 30 Hz to 200 Hz.

Figure 3 compares the separation measure ϵ_n for both AS and PAS methods applied to noisy output data with different SNR. Estimation on clean data output is not shown here; in this case, both methods perform similarly and give the true homogeneous contribution y_n (within machine accuracy). The high values of relative errors ϵ_2 and ϵ_4 is due to the smaller amplitude of signals y_2 and y_4 in respect to the other orders, which makes their estimation more sensitive to measurement noise.

Furthermore, Figure 3 shows that PAS outperforms AS method in presence of noise. For same order n and SNR, the gain in ϵ_n is around 6 dB.

4.3. Kernel identification using order separation

4.3.1. Identification algorithms

The standard Volterra identification method used in this paper is the Korenberg's algorithm for Least-Squares problem (KLS) as described in [6].

It consists of solving the following linear-in-the-parameters problem:

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{f} \tag{23}$$

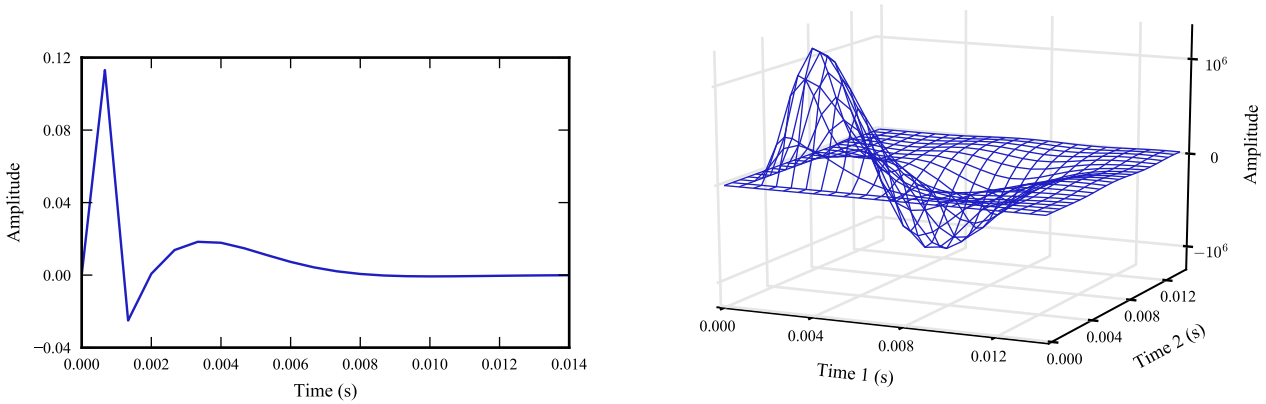
where:

- vector $\boldsymbol{y} = [y[0], \ldots, y[L-1]]^T$ is the concatenation of all output samples, with L the signal length (in samples);
- $\boldsymbol{\Phi} = [\boldsymbol{\phi}[0], \ldots, \boldsymbol{\phi}[L-1]]^T$ is the input combinatorial matrix, where vector $\boldsymbol{\phi}[k]$ regroups all cross-product $u[k-k_1] \ldots u[k-k_n]$ of the input signal at time k (for all orders n);
- vector $\boldsymbol{f}$ is the concatenation of all kernels values (for all orders n).

Using order separation (AS or PAS method), (23) is separated into N smaller problems:

$$\boldsymbol{y}_n = \boldsymbol{\Phi}_n \boldsymbol{f}_n \tag{24}$$

Identification is then carried on separately for each kernel; this leads to AS-KLS and PAS-KLS algorithm.



(a) Order 1

(b) Order 2

Figure 4: First and second-order kernels computed from (30) and (31) in Appendix 8.2.

Furthermore, using rPAS method, (23) is separated into the following problems

$$\boldsymbol{M}_{n,q} = \boldsymbol{\Phi}_{n,q} \boldsymbol{f}_n \tag{25}$$

for $n = 1, \ldots, N$ and $q = 1, \ldots, \lfloor n/2 \rfloor$[7]. Identification is then carried on separately for each couple (n, q); kernels coefficients are taken as the mean of the $\lfloor n/2 \rfloor$ estimations. This gives the proposed rPAS-KLS identification algorithm.

4.3.2. Identification results

The aforementioned identification methods were tested on the nonlinear loudspeaker, truncated to order $N = 3$.

Simulation were made at a sampling frequency of 1500 Hz. Gaussian noise signals of amplitude 10 V and 2 second length were used for input test signals. Kernel memory length was supposed to be equal to $M = 22$ samples (equivalent to 14 ms memory).

First two kernels of the nonlinear loudspeaker are given in Figure 4. Both linear and quadratic kernels shows exponential-like decay; the chosen memory length proves to be adequate at this sampling rate.

For all algorithms, relative identification error (computed as $\xi_n = \text{RMS}\left(\widetilde{h}_n - h_n\right)/\text{RMS}\left(h_n\right)$) are given in Figure 5.

When using clean output data, the addition of a prior order separation stage before identification improves greatly overall estimation process: error is reduced of more than 30 dB for the first order and 50 dB for the second. In this case, AS-KLS and PAS-KLS performs similarly; this is due to the similar separation results of AS and PAS methods with clean data.

Likewise, the better separation robustness of PAS over AS (around 6 dB gain) induces PAS-KSL to be more robust to noise than AS-KLS (around 2 dB gain).

[7] We consider only one case for the complex conjugates couples (n, q) and $(n, n - q)$.

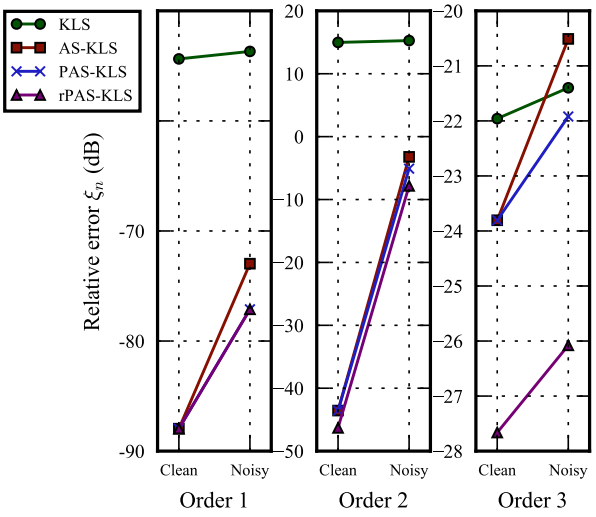
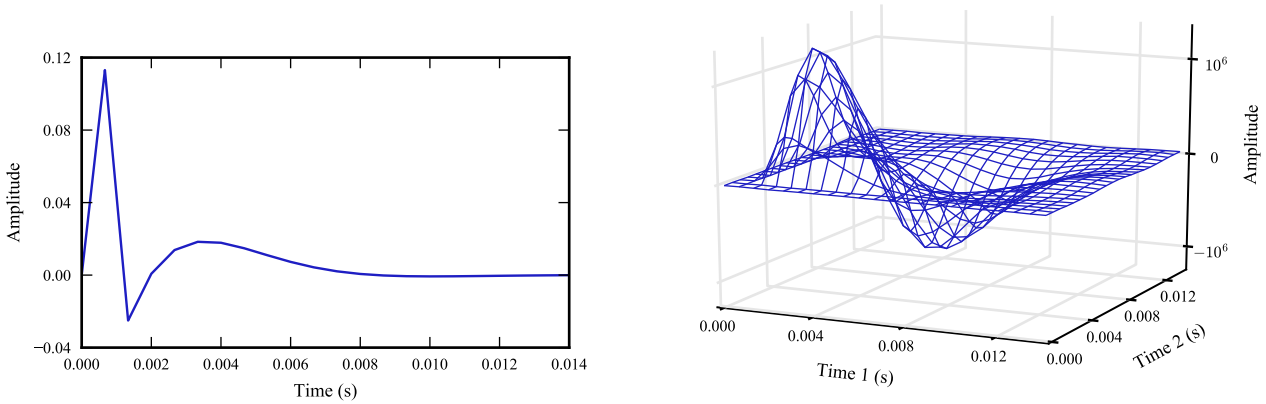


Figure 5: Relative error ξ_n (in dB) of identification w.r.t. order n for all identification methods using clean and noisy data (SNR $=$ 80 dB)

Furthermore, for second and third-order, rPAS-KLS algorithm offers a little improvement over PAS-KLS. This amelioration is the direct consequence of using terms $M_{n,q}$ rather than nonlinear orders y_n directly; indeed, averaging over all estimated $f_{n,q}$ for identification of kernel h_n can be seen as a way to artificially increase the data size used for identification, thus leading to better estimates.

The first two estimated kernels using rPAS-KLS on clean data are shown in Figure 6. We can see that the second-order kernel, which has a relative error $\xi_2 = -46$ dB, is well-identified using this method.



(a) Order 1

(b) Order 2

Figure 6: First and second-order kernels estimated with rPAS-KLS.

5. CONCLUSIONS

In this paper, a new method of nonlinear homogeneous order separation for Volterra series was proposed, based on phase dissimilarity. This method has been shown to be better conditioned, and thus more robust to noise, than amplitude-based separation algorithm.

Furthermore, two Volterra kernels identification methods were constructed combining the proposed separation process and a Least Squares-based identification algorithm from literature. Those methods have been shown to greatly improves kernel estimation.

Future works will be on the reproduction, with real signals, of the product's complex behaviour that enables usage of the theoretical PS method. An other interest would be to derive another identification method from Korenberg's algorithm that could be used directly on the phase-grouped terms Q_p, without needing to use AS method for further separation.

6. REFERENCES

[1] Thomas Hélie and Vanessa Smet, "Simulation of the weakly nonlinear propagation in a straight pipe: application to a real-time brassy audio effect," in *16th Mediterranean Conference on Control and Automation*. IEEE, 2008, pp. 1580–1585.

[2] Thomas Hélie, "Volterra series and state transformation for real-time simulations of audio circuits including saturations: Application to the Moog ladder filter," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 4, pp. 747–759, 2010.

[3] Finn T. Agerkvist, "Volterra Series Based Distortion Effect," in *Audio Engineering Society Convention 129*. Audio Engineering Society, 2010.

[4] Lamberto Tronchin, "The emulation of nonlinear time-invariant audio systems with memory by means of Volterra series," *Journal of the Audio Engineering Society*, vol. 60, no. 12, pp. 984–996, 2013.

[5] Arie J.M. Kaizer, "Modeling of the nonlinear response of an electrodynamic loudspeaker by a Volterra series expansion," *Journal of the Audio Engineering Society*, vol. 35, no. 6, pp. 421–433, 1987.

[6] Yves Goussard, William C Krenz, Lawrence Stark, and Guy Demoment, "Practical identification of functional expansions of nonlinear systems submitted to non-Gaussian inputs," *Annals of biomedical engineering*, vol. 19, no. 4, pp. 401–427, 1991.

[7] Antonin Novak, Laurent Simon, Pierrick Lotton, and Frantisek Kadlec, "Modeling of nonlinear audio systems using swept-sine signals: Application to audio effects," in *Proc. of the 12th Int. Conference on Digital Audio Effects (DAFx-09)*, 2009, pp. 1–4.

[8] Marc Rébillat, Romain Hennequin, Etienne Corteel, and Brian F. G. Katz, "Identification of cascade of Hammerstein models for the description of nonlinearities in vibrating devices," *Journal of Sound and Vibration*, vol. 330, no. 5, pp. 1018–1038, 2011.

[9] Stephen A. Billings, *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains*, John Wiley & Sons, 2013.

[10] Stephen P. Boyd, Y. S. Tang, and Leon O. Chua, "Measuring volterra kernels," *Circuits and Systems, IEEE Transactions on*, vol. 30, no. 8, pp. 571–577, 1983.

[11] Delphine Bard and Göran Sandberg, "Modeling of Nonlinearities in Electrodynamic Loudspeakers," in *Audio Engineering Society Convention 123*. Audio Engineering Society, 2007.

[12] Finn Agerkvist, Antoni Torras-Rosell, and Richard McWalter, "Eliminating transducer distortion in acoustic measurements," in *Audio Engineering Society Convention 137*. Audio Engineering Society, 2014.

[13] Russell H. Lambert, "Vandermonde Method for Separation of Nonlinear Orders and Measurement of Linear Response,"

in *Audio Engineering Society Convention 141*. Audio Engineering Society, 2016.

[14] B. Zhang and S.A. Billings, "Volterra series truncation and kernel estimation of nonlinear systems in the frequency domain," *Mechanical Systems and Signal Processing*, vol. 84, pp. 39–57, 2017.

[15] Wilson J. Rugh, *Nonlinear system theory*, Johns Hopkins University Press Baltimore, 1981.

[16] Stephen P. Boyd, Leon O. Chua, and Charles A. Desoer, "Analytical foundations of Volterra series," *IMA Journal of Mathematical Control and Information*, vol. 1, no. 3, pp. 243–282, 1984.

[17] Thomas Hélie and Béatrice Laroche, "Computation of convergence bounds for Volterra series of linear-analytic single-input systems," *Automatic Control, IEEE Transactions on*, vol. 56, no. 9, pp. 2062–2072, 2011.

[18] Thomas Hélie and Béatrice Laroche, "Computable convergence bounds of series expansions for infinite dimensional linear-analytic systems and application," *Automatica*, vol. 50, no. 9, pp. 2334–2340, 2014.

[19] Stephen Boyd and Leon Chua, "Fading memory and the problem of approximating nonlinear operators with Volterra series," *IEEE Transactions on circuits and systems*, vol. 32, no. 11, pp. 1150–1161, 1985.

[20] Gene H. Golub and Charles F. Van Loan, *Matrix computations*, vol. 3, JHU Press, 2012.

[21] SICA, "Constructor datasheet for SICA loudpseaker model Z000900," `https://www.sica.it/media/Z000900.pdf`, Accessed: 2017-04-07.

7. APPENDIX: DETAILED COMPUTATION OF (17)

Consider a system truncated to $N = 3$ with real input signal described in (13). From (17), we have:

$$y_n(t) = \sum_{q=0}^{n} \binom{n}{q} w^{n-2q} M_{n,q}(t) \tag{26}$$

So, using (1), the overall output of the system is:

$$\begin{aligned}
y(t) &= \sum_{n=1}^{N} y_n(t) \\
&= \sum_{n=1}^{N} \sum_{q=0}^{n} \binom{n}{q} w^{n-2q} M_{n,q}(t) \\
&= \sum_{\substack{1\le n\le N \\ n \text{ even}}} \sum_{q=0}^{n} \binom{n}{q} w^{n-2q} M_{n,q}(t) \\
&\quad + \sum_{\substack{1\le n\le N \\ n \text{ odd}}} \sum_{q=0}^{n} \binom{n}{q} w^{n-2q} M_{n,q}(t) \\
&= \sum_{\substack{1\le n\le N \\ n \text{ even}}} \sum_{\substack{-n\le p\le n \\ p \text{ even}}} \binom{n}{(n-p)/2} w^{p} M_{n,(n-p)/2}(t) \\
&\quad + \sum_{\substack{1\le n\le N \\ n \text{ odd}}} \sum_{\substack{-n\le p\le n \\ p \text{ odd}}} \binom{n}{(n-p)/2} w^{p} M_{n,(n-p)/2}(t)
\end{aligned}$$

by posing $p = n - 2q$. Taking into account the fact that both sum are finite, it is possible to inverse their orders, and thus:

$$\begin{aligned}
y(t) = &\sum_{\substack{-N\le p\le N \\ p \text{ even}}} w^{p} \left(\sum_{\substack{1<|p|\le n\le N \\ n \text{ even}}} \binom{n}{(n-p)/2} M_{n,(n-p)/2}(t) \right) \\
&+ \sum_{\substack{-N\le p\le N \\ p \text{ odd}}} w^{p} \left(\sum_{\substack{|p|\le n\le N \\ n \text{ odd}}} \binom{n}{(n-p)/2} M_{n,(n-p)/2}(t) \right)
\end{aligned} \tag{27}$$

8. APPENDIX: SYSTEM NUMERICAL APPROXIMATION

8.1. Numerical simulation

The input signal $u(t)$ is approximated at sampling rate f_s with a zero-order holder; then, following the state-space representation in (21), output current i at sample time l is given by:

$$i[l] = \sum_{n=1}^{N} \boldsymbol{C}\boldsymbol{x}_n[l] \tag{28}$$

where $\boldsymbol{x}_n$ are the states of nonlinear homogeneous order n; first three orders are given by the following recursive equations:

$$\boldsymbol{x}_1[l+1] = e^{\boldsymbol{A}T_s}\boldsymbol{x}_1[l] + \boldsymbol{\Delta_0}\boldsymbol{B}u[l] \tag{29a}$$

$$\boldsymbol{x}_2[l+1] = e^{\boldsymbol{A}T_s}\boldsymbol{x}_2[l] + \boldsymbol{\Delta_0}\boldsymbol{K}_2(\boldsymbol{x}_1[l], \boldsymbol{x}_1[l]) \tag{29b}$$

$$\begin{aligned}
\boldsymbol{x}_3[l+1] = e^{\boldsymbol{A}T_s}\boldsymbol{x}_3[l] &+ \boldsymbol{\Delta_0}\boldsymbol{K}_3(\boldsymbol{x}_1[l], \boldsymbol{x}_1[l], \boldsymbol{x}_1[l]) \\
&+ 2\boldsymbol{\Delta_0}\boldsymbol{K}_2(\boldsymbol{x}_1[l], \boldsymbol{x}_2[l])
\end{aligned} \tag{29c}$$

where $T_s = 1/f_s$ is the sampling time and $\boldsymbol{\Delta_0} = \boldsymbol{A}^{-1}\left(e^{\boldsymbol{A}T_s} - \boldsymbol{I}\right)$ is a bias term due to sampling.

8.2. Discrete-time kernel computation

Discrete-time kernels h_n corresponding to the loudspeaker simulation using (28) and (29) are given by:

$$h_n[m_1, \ldots, m_n] = \sum_{n=1}^{N} \boldsymbol{C}\boldsymbol{g}_n[m_1, \ldots, m_n] \tag{30}$$

where $\boldsymbol{g}_n$ are the input-to-states kernels of order n; first three orders are given by:

$$\boldsymbol{g}_1[m] = (1 - \delta_{m,0}) e^{\boldsymbol{A}T_e(m-1)}\boldsymbol{\Delta_0} \tag{31a}$$

$$\boldsymbol{g}_2[m_1, m_2] = \sum_{l=0}^{\max\{m_1,m_2\}} e^{\boldsymbol{A}T_e l}\boldsymbol{\Delta_0}\boldsymbol{K}_2(\boldsymbol{g}_1[m_1 - l], \boldsymbol{g}_1[m_2 - l]) \tag{31b}$$

$$\begin{aligned}
\boldsymbol{g}_3[m_1, m_2, m_3] = \sum_{l=0}^{\max\{m_1,m_2,m_3\}} e^{\boldsymbol{A}T_e l}\boldsymbol{\Delta_0} \Big(&\\
&\boldsymbol{K}_3(\boldsymbol{g}_1[m_1 - l], \boldsymbol{g}_1[m_2 - l], \boldsymbol{g}_1[m_3 - l]) \\
&+ 2\boldsymbol{K}_2(\boldsymbol{g}_1[m_1 - l], \boldsymbol{g}_2[m_2 - l, m_3 - l]) \Big)
\end{aligned} \tag{31c}$$

with $\delta_{m,n}$ the Kronecker delta.