

Compensation of biased excitation effects for MLS-based nonlinear systems' identification

Virginia Yagüe-Jiménez , José Luis Blanco-Murillo,
Francisco Javier Casajús-Quirós

A B S T R A C T

MLS-based identification of nonlinear systems is largely affected by deviations in the excitation signal amenable to the combined effect of DC-offset and an arbitrary gain. These induce orthogonality loss in the MLS filter bank output, thus invalidating the underlying identification construction. In this paper we present a correction algorithm to derive the corrected Volterra kernels from the biased estimations provided by the standard MLS-based procedure.

1. Introduction

The use of Volterra kernels as a means of modeling non-linear systems arises whenever a general input–output relationship is needed. For this purpose, authors have introduced a nonlinear operator $\mathcal{T}(x)$ to which a Taylor series expansion can be applied [1,2]. Let this expansion be

$$y(t) = \mathcal{T}(u)u(t) = \{\mathcal{V}_0 + \mathcal{V}_1[u(t)] + \dots + \mathcal{V}_N[u(t)]\}u(t) = \mathcal{V}[u(t)]u(t) \quad (1)$$

where \mathcal{V}_n denotes an n th order nonlinear operator. Term \mathcal{V}_0 represents the part of the output signal $y(t)$ which is independent of input signal $u(t)$. \mathcal{V}_1 is the linear part, \mathcal{V}_2 the quadratic part and so on.

In Volterra's approach [3], the set of operators $\{\mathcal{V}_r(x)\}$ completely characterizes a nonlinear system with memory. From the Volterra representation, the output of a nonlinear system of order N can be approximated by a finite summation, involving discrete-variable Volterra kernels with finite memory, M —provided that Volterra's integrals converge. Unfortunately, this does not imply that the problem is numerically tractable, but merely that it is consistently formulated. Even so, if we assume that the Volterra series exists for a causal system with memory of length M , a series of discrete Volterra kernels of orders 0 – N can be defined, and the sampled output signal $y[n]$ be written as

$$y[n] = h_0 + \sum_{r=1}^N \sum_{k_1=0}^{M-1} \dots \sum_{k_r=0}^{M-1} h_r[k_1, \dots, k_r] u[n-k_1] \dots u[n-k_r] \quad (2)$$

This linear, multidimensional formulation may also be written in matrix form. To do so, the first term h_0 is eliminated from Eq. (2), since it is an input independent term that can be added directly to the output. A row vector $\mathbf{u}_n = [u(n), u(n-1), \dots, u(n-M+1)]$ is

then built from the set of excitations used, while kernels $h_r[k_1, \dots, k_r]$ are also cast into a single vector by means of an ordering system that is explained later on. In this manner, the generalized discrete convolution can be expressed as a Kronecker product of these two vectors. Using $\mathbf{u}_n \otimes \mathbf{u}_n \otimes \dots \otimes \mathbf{u}_n = \mathbf{u}_n^{[r] \otimes}$ for the sequential application of r -Kronecker products on the same vector, Eq. (2) can be written in compact form as

$$y[n] = \sum_{r=1}^N \mathbf{u}_n^{[r] \otimes} \mathbf{h}_r \quad (3)$$

Similarly, the system output may also be expressed in terms of Wiener's model and the corresponding Wiener coefficients. We explore this representation in Section 2 for an appropriate excitation and an specific construction of Wiener's model based on Maximum Length Sequences (MLS). One may derive a convenient formulation from these, to exploit noticeable redundancies during the identification process, as we describe in Section 2.1. The identification process based on MLS sequences is formulated in Section 2.2 in matrix form. MLS-based identification is widely used in audio [4,5], as well as in general purpose systems identification [6].

Little has been done to analyze how DC level of the excitation affects the identification process. Direct current is present whenever polarization of the nonlinear system is required, altering its behavior. Most frequently this is used to select a convenient operation point for the system, for instance, to approach the most desirable linear behavior. From the perspective of identification, DC presence causes the orthogonality assumption to fail, which is required by MLS identification techniques as explained in Section 3. The result is a set of biased Volterra kernels' estimates. As an example, in [7] authors constrain their analysis to zero-mean excitations; forcing the operation point of the system.

One may overcome this situation and derive the set of corrections needed in Section 4, from which the actual non-biased, non-parametric representations may be extracted. The correction algorithm on the Volterra kernels is proposed based on the matrix form of the identification problem. In Section 5 we assess the performance of the correction algorithm on a digital non-linear system. Finally, in Section 6 we discuss the results of our experiment, while in Section 7 we present our conclusions.

2. Non-parametric identification of non-linear systems

A wide variety of physical phenomena match the definition of a nonlinear system. Such a variety implies that different methods must exist for description and identification, depending on the underlying physical processes. Broadly, these are classified into two categories: parametric and non-parametric.

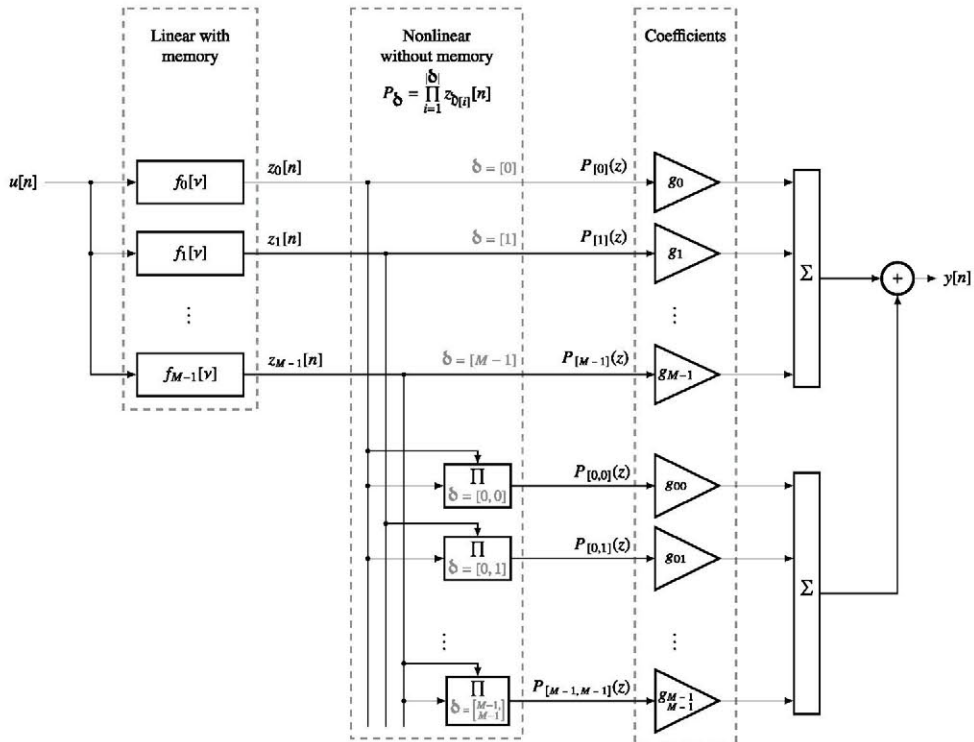


Fig. 1. General Wiener model for an arbitrary non-linear system with memory.

Generally speaking, the main drawback of non-parametric techniques is the larger number of coefficients that must be computed to entirely specify the abstract kernels [8]; compared to the more limited number in a physically-informed model. Consequently, probe signals are often much longer and computational complexity is higher; while on the other hand, these are particularly suitable techniques for black-box approaches and systems' inversion [9].

Early non-parametric methods for system identification were based on fixed block structures, and used random excitations. Attending to Wiener's model, a general formulation is driven from a single-input, multi-output linear filter bank, followed by a nonlinear network. Orthonormal basis was introduced at the linear filter bank, which along with the discussion in [1] allows resulting Wiener coefficients computation, from which the Volterra kernels can be derived via a basis change (see [10]). The structure depicted in Fig. 1 matches the Wiener model description: a filter bank based on Laguerre functions is followed by a nonlinear network in terms of Hermite polynomials.

Since this approach requires very long input random signals, several authors have evaluated pseudorandom sequences instead. In [6] authors introduce the Kronecker product to write Eq. (2) in a compact matrix form, and a least-squares (LS) algorithm is then applied, yielding estimates on the values of the Volterra kernels. Combining Schetzen's method [1] with this matrix form, Reed and Hawksford [11] derived an identification method that employs linear combinations of MLS. Such signals had already been used in characterizing the distortion generated by nonlinearities inside a system [12].

In Reed and Hawksford's approach, MLS are used to build a convenient orthogonal filter bank. The individual impulse responses of which are the time-reversed and time-shifted versions of the same MLS. The subsequent polynomial, nonlinear network considers every possible multiplicative, partial, permutation of the signals at the output of the filter bank to model all possible cross-effects between the filter bank output branches. The system is then excited itself with a set of MLS inputs that result from the combinations of these same MLS. This approach aims at minimizing the length of the probe signal.

For the system in Fig. 1, output $y[n]$ is the result of as many non-linear operators \mathcal{Z}_r as the maximum order N needed, applied on a set of M individual linear, MLS, impulse responses $\{f_r[\nu]\}_{r=0,\dots,M-1}$ to a MLS-like excitation \mathbf{u} , and combined in the memoryless polynomial (product) network to implement all combinations of the filter bank individual outputs all variations of M elements from 1 to the maximum order N . Full details are provided in Section 2.1, while Eq. (4) provides a full mathematical formulation for the system output:

$$y[n] = \sum_{r=0}^N \mathcal{Z}_r[u[n]] = \sum_{r=1}^N \mathbf{g}_{i_1,\dots,i_r} \sum_{i_1=0}^{M-1} \dots \sum_{i_r=0}^{M-1} \prod_{j=1}^r z_{i_j}[n] \quad (4)$$

Inputs are the outputs from the filter bank, while based on parallel connection rules for nonlinear systems [10], we may compute individual outputs:

$$P_{\mathfrak{b}}(z[n]) = \prod_{i=1}^{|\mathfrak{b}|} z_{\mathfrak{b}[i]}[n] = \sum_{\nu_1=0}^{M-1} \dots \sum_{\nu_r=0}^{M-1} \prod_{s=1}^r f_{\mathfrak{b}[s]}[\nu_s] u[n-\nu_s] \quad \text{where } |\mathfrak{b}| = r \quad (5)$$

Here $P_{\mathfrak{b}}(z[n])$ is a polynomial built from the filter outputs $z[n]$, and \mathfrak{b} denotes the sequence of z variables used to compute a particular output branch in the nonlinear, memoryless network. The cardinality of \mathfrak{b} , the number of nonlinear combinations introduced, is $|\mathfrak{b}|$.

The filter bank effect here expressed in terms of linear, discrete convolutions can be formulated as Toeplitz matrix products, $\mathbf{u}_n \mathbf{F}_r$ (see [13]). Considering now the set of polynomial functions generated by sequences \mathfrak{b} , the output of the model can be written as follows, attending to (i) the time-shifted construction of the proposed filter bank and (ii) the Toeplitz structure of the discrete convolution matrix:

$$y[n] = \sum_{r=1}^N \mathbf{u}_n^{[\mathfrak{b}] \otimes} \mathbf{F}^{[r] \otimes} \mathbf{g}_r \quad (6)$$

Comparing Eqs. (3) and (6) we identify the tight relationship between coefficient vectors \mathbf{g}_r and the Volterra kernels, \mathbf{h}_r . This arises from the existence of an unambiguous transformation of a set of, easily computable, Wiener coefficients into a set of, physically-interpretable, symmetric Volterra kernels.

The use of Kronecker products in formulas involving generalized convolutions provides compact expressions. But they are not parsimonious, since coefficients \mathbf{g}_r are highly redundant. The very nature of Kronecker products and commutativity of scalar products create multiple symmetries that can be exploited to simplify the process of kernel identification. Those symmetries are closely related to the combinatorial nature of the polynomial network, and the partial permutations and combinations there implied.

2.1. Reformulation of the problem: symmetry of Volterra kernels in matrix form

In this section we reformulate the identification problem to take advantage of notorious redundancies that are present in Volterra's and Wiener's representations, due to the product network's input combinations [14].

2.1.1. Indexation of the elements in kernel-based representations

Let us define the set $\mathcal{M} = \{0, \dots, M-1\}$, where M is the length in samples of the memory of the system to be identified. The number of outgoing branches in the filter bank is also M , as indicated in Fig. 1, so \mathcal{M} may be used to index these.

Considering the Volterra model for this system, the r -order kernel is a set of M^r coefficients, corresponding to the r -permutations with repetition of the elements of set \mathcal{M} , i.e. the individual filter bank outputs. This provides the maximum number of branches for the product network that matches the number of all possible interproducts. The kernel coefficients may then be indexed using the r -tuples $\mathbf{e} = (e[0], e[1], \dots, e[r-1])$ belonging to \mathcal{M}^r , the r -order product set of \mathcal{M} , i.e., $\mathcal{M}^r = \mathcal{M} \times \dots \times \mathcal{M}$. We now focus on coefficients indexation and ordination.

The index r -tuples can be univocally and naturally ordered by simply translating them into numbers written in an M -ary system. Consequently, the ordinal number for a r -tuple \mathbf{e} is set to

$$\left\{ S(\mathbf{e}) = \sum_{i=0}^{r-1} e[i]M^{r-i-1} \mid |\mathbf{e}| = r \right\} \quad \text{where we know that } S(\mathbf{e}_1) > S(\mathbf{e}_2) \Leftrightarrow \mathbf{e}_1 > \mathbf{e}_2 \quad (7)$$

In what follows we refer to this ordering criterium as \mathcal{O} . One may then write the ordered set of all r -permutations with repetition of the elements in \mathcal{M} as $\mathfrak{D}_r(\mathcal{M}) := \mathcal{O}(\mathcal{M}^r)$.

Next, we define $\mathfrak{D}(\mathcal{M})$ as the ordered sequence of all possible partial permutations with repetition, involving 1– N elements. For this purpose the concatenation operator $\#$ is used. $\mathfrak{D}(\mathcal{M}) := [\mathfrak{D}_1(\mathcal{M}) \# \mathfrak{D}_2(\mathcal{M}) \dots \# \mathfrak{D}_N(\mathcal{M})]$. Each sequence $\mathfrak{d} \in \mathfrak{D}(\mathcal{M})$ in this set has one and only one associated ordinal index within the set $\mathcal{J} := \{1, 2, \dots, \sum_{r=1}^N M^r\}$. According to Eq. (7), this index is precisely the numerical value of the sequence in M -ary number system. We define

$$\{\mathfrak{d}_j := [\mathfrak{D}(\mathcal{M})]_j \mid \mathfrak{d}_j \in \mathfrak{D}(\mathcal{M}) \wedge j \in \mathcal{J}\} \quad (8)$$

Set \mathcal{J} can be split into N subsets containing sequences that have length r , and which may be associated to $\mathfrak{D}_r(\mathcal{M})$. So \mathcal{J} can also be defined by ordered concatenation as $\mathcal{J} := [\mathcal{J}_1 \# \mathcal{J}_2 \# \dots \# \mathcal{J}_N]$, where $\{j \in \mathcal{J}_r \mid \mathfrak{d}_j \in \mathfrak{D}_r(\mathcal{M}) \Rightarrow |\mathfrak{d}_j| = r\}$. It is only natural to introduce now the set $\mathcal{K}_r := \{1, 2, \dots, M^r\}$ as the indices for sets $\mathfrak{D}_r(\mathcal{M})$. An illustration on this double indexing structure is provided in Fig. 2 to the left. Both \mathcal{J} and the individual \mathcal{K}_r indexing structures may be used to reference any sequence in $\mathfrak{D}(\mathcal{M})$ —between square brackets.

The proposed indexation systems for $\mathfrak{D}(\mathcal{M})$ provide alternative ways to accurately and orderly refer to the combinations of the filter bank outputs given by \mathfrak{d} . Interestingly, Kronecker products combined with iterative summations as in Eq. (6) cause numerous redundant computations. This is intimately related with the well-known property that any nonlinear system amenable to the Volterra series representation is uniquely represented on the symmetric Volterra kernels. Substantial redundancy reduction may be introduced in the computation process, provided that redundancy terms are properly identified. To benefit from this phenomenon our index system is now adapted to exploit the redundancy.

Let $\tilde{\mathfrak{D}}_r(\mathcal{M})$ be defined as the set of all combinations with repetition of size r (or r -multicombinations) of set \mathcal{M} —instead of the partial permutations, $\mathfrak{D}_r(\mathcal{M})$, at which we looked before. This set may also be ordered by means of criterium \mathcal{O} in Eq. (7), yielding $\mathfrak{Q}_r(\mathcal{M}) := \mathcal{O}(\tilde{\mathfrak{D}}_r(\mathcal{M}))$.

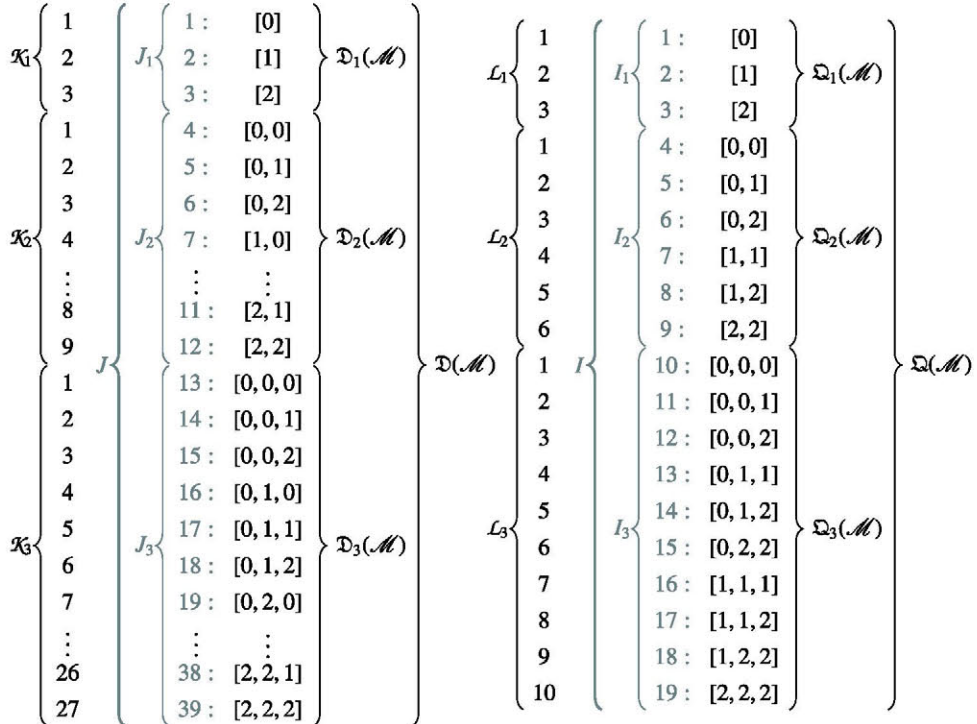


Fig. 2. Ordered sets of sequences belonging to $\mathfrak{D}(\mathcal{M})$ and $\mathfrak{Q}(\mathcal{M})$ and their indices for $M=3$ and $N=3$.

Concatenation of those sets allows one to define the set of multicombinations with sizes ranging from 1 to N :

$$\mathfrak{Q}(\mathcal{M}) := [\mathfrak{Q}_1(\mathcal{M}) \# \mathfrak{Q}_2(\mathcal{M}) \cdots \# \mathfrak{Q}_N(\mathcal{M})] \quad (9)$$

Similar to what was done before for $\mathfrak{D}(\mathcal{M})$, this set of sequences will be indexed by an adequate index set such as $\mathcal{I} := \{1, 2, \dots, \sum_{s=1}^N \binom{M+s-1}{s}\}$, the elements of which serve as indices for the sequences \mathfrak{q} :

$$\{\mathfrak{q}_i := [\mathfrak{Q}(\mathcal{M})]_i \mid \mathfrak{q}_i \in \mathfrak{Q}(\mathcal{M}) \wedge i \in \mathcal{I}\} \quad (10)$$

Note that the cardinality of \mathcal{I} is different from that of \mathcal{J} . The latter indexation was oriented to partial permutations, while the current one relates to the combinations with repetition, the number of which is smaller. As in the preceding case, the index set \mathcal{I} can be split into subsets attending to the length r , i.e., $\mathcal{I}_r := [\mathcal{I}_1 \# \mathcal{I}_2 \# \dots \# \mathcal{I}_N]$, where

$$\{i \in \mathcal{I}_r \mid \mathfrak{q}_i \in \mathfrak{Q}_r(\mathcal{M}) \Rightarrow |\mathfrak{q}_i| = r\} \quad (11)$$

Set \mathcal{L}_r plays a similar role for sequences $\mathfrak{Q}_r(\mathcal{M})$ as \mathcal{K}_r does for $\mathfrak{D}_r(\mathcal{M})$ indexation. This is evidenced in Fig. 2 where the two indexation structures proposed for $\mathfrak{Q}(\mathcal{M})$ are depicted to the right, and compared to those proposed for $\mathfrak{D}(\mathcal{M})$ —to the left. The indexation systems for sequences \mathfrak{d} and \mathfrak{q} we have developed are used hereafter to handle sequence products in an orderly manner. This is easily amenable to the matrix notation used to formulate the nonlinear, discrete convolution.

As an example for sequences \mathfrak{d} —used for partial permutations—one may like to focus on Eq. (5). This sequence product involves a partial permutation of r variables $z_i[n]$ selected by sequence \mathfrak{d} that contains r elements—i.e. $|\mathfrak{d}| = r$. From the Volterra series representation in Eq. (2), one may rewrite this as follows:

$$y[n] = h_0 + \sum_{r=1}^N \left(\sum_{\mathfrak{d} \in \mathfrak{D}_r} h_r[\mathfrak{d}] u[n - \mathfrak{d}[1]] \cdots u[n - \mathfrak{d}[r]] \right) \quad (12)$$

The resulting formation of the Volterra series convolution includes the summation of all r -order Volterra kernels up to order N . On the r th one, we have a sequence product of \mathfrak{d} -shifted vectors, times the r -order Volterra kernel.

2.1.2. Symmetrical Volterra kernels and non-linear convolution redundancy

Sequences $\mathfrak{d} \in \mathfrak{D}(\mathcal{M})$ indexation can be used to condense the system output representation by rearranging Volterra kernels in matrix notation as one single vector. This is required for Eqs. (3) and (6), where the output of a nonlinear system is formulated in matrix form.

Applying order criterion \mathcal{O} , defined by Eq. (7), the Volterra kernels can be stacked on a single column vector as follows. First, we define vector \mathbf{h}_r , which is the r -order Volterra kernel, for which the ζ -th position is

$$\{[\mathbf{h}_r]_\zeta = h_r[\mathfrak{d}_\zeta] \mid \mathfrak{d}_\zeta \in \mathfrak{D}_r(\mathcal{M}) \wedge \zeta \in \mathcal{K}_r\} \quad (13)$$

Then, every kernel \mathbf{h}_r must be stacked with the others so as to form the characteristic kernel vector:

$$\mathbf{h}^\top \triangleq [\mathbf{h}_1^\top \mid \mathbf{h}_2^\top \mid \cdots \mid \mathbf{h}_N^\top], \quad \text{where the } j\text{-th element is } [[\mathbf{h}]_j = h_r[\mathfrak{d}_j] \mid \mathfrak{d}_j \in \mathfrak{D}(\mathcal{M}) \wedge j \in \mathcal{J}_r] \quad (14)$$

Sequences $\mathfrak{q} \in \mathfrak{Q}(\mathcal{M})$ are useful when we try to eliminate redundancy in the matrix operations we are concerned about. Other authors have evaluated less-redundant, looser representations as in [15]. Attending to these, Volterra's representation in Eq. (2), may be written either in terms of \mathfrak{d} as in Eq. (12), or be indexed based on \mathfrak{q} :

$$y[n] = h_0 + \sum_{r=1}^N \left(\sum_{\mathfrak{q} \in \mathfrak{Q}_r} |\mathfrak{G}(\mathfrak{q})| h_r[\mathfrak{q}] u[n - \mathfrak{q}[1]] \cdots u[n - \mathfrak{q}[r]] \right) \quad (15)$$

If the Volterra kernels are given in symmetrical form, the order in which shifted input coefficients appear is irrelevant—product commutative property. Consequently, the summation of the products for the r th kernel including all r -permutations (indexed by \mathfrak{d}) includes multiple repetitions of a reduced number of different products, which are given by the combinations with repetition indexed by \mathfrak{q} . While computing the summation, one may benefit from these redundancies, count the number of repeated products and compute only these. Hence, we may formally state that

$$\{\forall \mathfrak{d} \in \mathfrak{G}(\mathfrak{q}_\ell) \Rightarrow h_r[\mathfrak{d}_\zeta] = h_r[\mathfrak{q}_\ell] \mid \mathfrak{d}_\zeta \in \mathfrak{D}_r(\mathcal{M}) \wedge \mathfrak{q}_\ell \in \mathfrak{Q}_r(\mathcal{M}) \wedge \zeta \in \mathcal{K}_r \wedge \ell \in \mathcal{L}_r\} \quad (16)$$

where $\mathfrak{G}(\mathfrak{q}_\ell)$ are all possible permutations of sequence \mathfrak{q}_ℓ .

If we collapse Volterra kernels and conveniently reformulate this representation, instead of having $|\mathfrak{D}_r(\mathcal{M})|$ coefficients only $|\mathfrak{Q}_r(\mathcal{M})|$ appear to be needed to compute the r -order kernel contribution. Each of these elements of the reduced list of coefficients appears $|\mathfrak{G}(\mathfrak{q}_\ell)|$ times in the summation. We may then define the individual, non-redundant, triangular, r -order Volterra kernel in matrix form, as a one-dimensional vector that takes at its ℓ -th position the value

$$[\mathbf{h}_r]_\ell = |\mathfrak{G}(\mathfrak{q}_\ell)| \cdot [\mathbf{h}_r]_\zeta \mid \mathfrak{d}_\zeta \in \mathfrak{G}(\mathfrak{q}_\ell) \quad (17)$$

The resulting triangular Volterra kernels display the same structure as in Eq. (14) and its length depends on $|\mathfrak{Q}(\mathcal{M})|$:

$$\mathbf{h}^\top \triangleq [\mathbf{h}_1^\top \mid \mathbf{h}_2^\top \mid \cdots \mid \mathbf{h}_N^\top], \quad |\mathbf{h}| = \sum_{r=1}^N |\mathfrak{Q}_r(\mathcal{M})| = \sum_{r=1}^N \binom{M+r-1}{r} \quad (18)$$

2.1.3. Expansion and compression of indexation sequences

In cases such as those we have expounded, it is useful to define the expansion and reduction matrices, which enable index changes from \mathfrak{q} sequences in $\mathfrak{Q}(\mathcal{M})$ to \mathfrak{d} sequences in $\mathfrak{D}(\mathcal{M})$, and vice versa.

Let reduction matrix \mathbf{Q}_r for order r be defined as

$$\exists! \mathbf{Q}_r \left| q_{\ell, \kappa} = \begin{cases} 1 & : \mathfrak{d}_{\kappa} \in \mathfrak{G}(\mathfrak{q}_{\ell}) \\ 0 & : \mathfrak{d}_{\kappa} \notin \mathfrak{G}(\mathfrak{q}_{\ell}) \end{cases} \quad \forall \ell \in \mathcal{L}_r \wedge \forall \kappa \in \mathcal{K}_r \quad (19)$$

This definition allows us to express triangular Volterra kernel vectors as reductions of the symmetric kernels. Thus

$$\left\{ \mathbf{Q}_r: \mathbb{R}^{|\mathfrak{D}_r(\mathcal{M})|} \rightarrow \mathbb{R}^{|\mathfrak{Q}_r(\mathcal{M})|} \left| \hat{\mathbf{h}}_r = \mathbf{Q}_r \mathbf{h}_r \right. \right\} \quad (20)$$

Conversely, vector reductions can be undone by expansion matrices, \mathbf{P}_r :

$$\exists! \mathbf{P}_r \left| p_{\kappa, \ell} = \begin{cases} \frac{1}{|\mathfrak{G}(\mathfrak{q}_{\ell})|} & : \mathfrak{d}_{\kappa} \in \mathfrak{G}(\mathfrak{q}_{\ell}) \\ 0 & : \text{otherwise} \end{cases} \quad \forall \kappa \in \mathcal{K}_r \wedge \forall \ell \in \mathcal{L}_r \quad (21)$$

$$\left\{ \mathbf{P}_r: \mathbb{R}^{|\mathfrak{Q}_r(\mathcal{M})|} \rightarrow \mathbb{R}^{|\mathfrak{D}_r(\mathcal{M})|} \left| \mathbf{h}_r = \mathbf{P}_r \hat{\mathbf{h}}_r \right. \right\} \quad (22)$$

As an example reduction— \mathbf{Q} —and expansion— \mathbf{P} —matrices for $r=3$ and $M=2$ are shown next:

$$\mathbf{Q}_3|_{M=2} = \left. \begin{array}{c} \overbrace{\begin{matrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{matrix}}^{\mathfrak{D}_3(\mathcal{M})} \\ \left[\begin{array}{cccccccc|ccc} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{1} \end{array} \right] \end{array} \right\}_{\mathfrak{Q}_3(\mathcal{M})} \quad \mathbf{P}_3|_{M=2} = \left. \begin{array}{c} \overbrace{\begin{matrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{matrix}}^{\mathfrak{D}_3(\mathcal{M})} \\ \left[\begin{array}{cccc|ccc} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array} \right] \end{array} \right\}_{\mathfrak{P}_3(\mathcal{M})} \quad (23)$$

The defined compression and expansion matrices \mathbf{Q}_r and \mathbf{P}_r display the following properties:

$$\mathbf{Q}_r \mathbf{P}_r = \mathbf{I}_{|\mathfrak{Q}_r(\mathcal{M})|}, \quad \mathbf{P}_r \mathbf{Q}_r = \mathbf{S}_r, \quad \mathbf{S}_r \mathbf{h}_r = \mathbf{h}_r, \quad \mathbf{S}_r \mathbf{g}_r = \mathbf{g}_r \quad (24)$$

where $\mathbf{I}_{|\mathfrak{Q}_r(\mathcal{M})|}$ is an $|\mathfrak{Q}_r(\mathcal{M})| \times |\mathfrak{Q}_r(\mathcal{M})|$ identity matrix and \mathbf{S}_r the following symmetry-defining matrix:

$$\exists! \mathbf{S}_r \left| s_{\kappa, \ell} = \begin{cases} \frac{1}{|\mathfrak{G}(\mathfrak{d}_{\ell})|} & : \mathfrak{d}_{\kappa} \in \mathfrak{G}(\mathfrak{d}_{\ell}) \\ 0 & : \text{otherwise} \end{cases} \quad \forall \kappa, \ell \in \mathcal{K}_r \quad (25)$$

For the example matrices $\mathbf{Q}_3|_{M=2}$ and $\mathbf{P}_3|_{M=2}$ in Eq. (23), the identity matrix corresponding to $\mathbf{Q}_3|_{M=2} \cdot \mathbf{P}_3|_{M=2}$ is \mathbf{I}_4 , whereas the symmetry matrix obtained for $\mathbf{P}_3|_{M=2} \cdot \mathbf{Q}_3|_{M=2}$ is

$$\mathbf{S}_3|_{M=2} = \left. \begin{array}{c} \overbrace{\begin{matrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{matrix}}^{\mathfrak{D}_3(\mathcal{M})} \\ \left[\begin{array}{cccccccc|ccc} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \frac{\mathbf{1}}{\mathbf{3}} & \frac{\mathbf{1}}{\mathbf{3}} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array} \right] \end{array} \right\}_{\mathfrak{S}_3(\mathcal{M})} \quad (26)$$

These same matrices may also be used when removing redundancy from a matrix that results of a repeated Kronecker product. Consider any square matrix \mathbf{B} , with elements indexed by a pair (i, j) . On matrix $\mathbf{Z} = \mathbf{B}^{[r] \otimes}$, elements may be indexed by sequences \mathfrak{b} . Compounded by the indices of matrix \mathbf{B} contributing to the product that generates a given element of \mathbf{Z} , one

may write the (i,j) element of this as

$$z(i,j) = b(d_i[1], d_j[1]) \cdot b(d_i[2], d_j[2]) \cdot \dots \cdot b(d_i[r], d_j[r]) = b(\mathbf{v}_1) \cdot b(\mathbf{v}_2) \cdot \dots \cdot b(\mathbf{v}_r) \quad (27)$$

where \mathbf{v}_x is the pair $(d_i[x], d_j[x])$ and $i, j \in \mathcal{J}_r$. \mathbf{Z} can then be indexed by those pairs, as in $z(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r)$. Interested readers may find this and other properties of the Kronecker product in [16]. We then conclude that there is a symmetry in \mathbf{Z} that is similar to the one among Volterra kernels:

$$\{z(\mathbf{u}) = z(\mathbf{u}) | \mathbf{u} = (\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r) \wedge \mathbf{u} \in \mathfrak{G}(\mathbf{u})\} \quad (28)$$

Based on this formulation of redundancy in repeated Kronecker products, we may now evaluate the relationship between Volterra and Wiener formulations for the system output in Eqs. (3) and (6). This happens to be the following for the r -th kernel:

$$\mathbf{h}_r = \mathbf{F}^{[r] \otimes} \mathbf{g}_r \quad (29)$$

When filter matrix \mathbf{F} is orthonormal, as it happens for identification based on MLS sequences, its associated repeated Kronecker product, $\mathbf{F}^{[r] \otimes}$, is orthonormal too. Hence, its matrix inverse is simply its transpose:

$$\mathbf{g}_r = (\mathbf{F}^{[r] \otimes})^\top \mathbf{h}_r \quad (30)$$

Since symmetry condition Eq. (28) still holds after transposition, we conclude that symmetries in the Volterra's coefficients have their counterpart on Wiener's representation; provided that the filter bank is orthonormal. We can then write for Wiener's coefficients an analogous expression to Eq. (16) for Volterra's:

$$\{\forall \mathbf{d} \in \mathfrak{G}(\mathbf{q}_i) \Rightarrow \mathbf{g}_r[\mathbf{d}_\ell] = \mathbf{g}_r[\mathbf{q}_\ell] | \mathbf{d}_\ell \in \mathfrak{D}_r(\mathcal{M}) \wedge \mathbf{q}_\ell \in \mathfrak{Q}_r(\mathcal{M}) \wedge \ell \in \mathcal{K}_r \wedge \ell \in \mathcal{L}_r\} \quad (31)$$

This leads to a single, triangular vector form with no redundancy for Wiener's representation, similar to one proposed for the triangular Volterra kernel vector $\hat{\mathbf{h}}$ in Eq. (18) and that displays a similar structure to \mathbf{h} —Eq. (14):

$$\hat{\mathbf{g}}^\top \triangleq [\hat{\mathbf{g}}_1^\top | \hat{\mathbf{g}}_2^\top | \dots | \hat{\mathbf{g}}_N^\top] \quad \text{and} \quad [\hat{\mathbf{g}}_i] = |\mathfrak{G}(\mathbf{q}_i)| [\mathbf{g}_i] | \mathbf{d}_j \in \mathfrak{G}(\mathbf{q}_i) \quad (32)$$

2.2. System identification based on MLS sequences in matrix form

The proposed matrix forms for Volterra's and Wiener's coefficients, \mathbf{h} and \mathbf{g} , and their reduced formulations, $\hat{\mathbf{h}}$ and $\hat{\mathbf{g}}$, along with our previous notation, are now used to formulate the nonlinear system identification problem based on MLS sequences [17]. This is essentially a matrix inversion problem when the specific probe inputs \mathbf{u} are used and the symmetry of the Volterra coefficients is exploited.

Assuming that a discrete, FIR representation is possible for the filter bank, and that those sequences of length M generate an orthonormal basis, we may specify the entire, discrete filter bank by row-matrix:

$$\mathbf{M} = \begin{bmatrix} \mathbf{m}_0 & \mathbf{m}_1 & \dots & \mathbf{m}_{M-1} \\ m_0[0] & m_0[1] & \dots & m_0[M-1] \\ m_0[1] & \dots & m_0[M-1] & m_0[0] \\ \dots & \dots & \dots & \dots \\ m_0[M-1] & m_0[0] & \dots & m_0[M-2] \end{bmatrix} \begin{matrix} \mathbf{m}_0 \\ \mathbf{m}_1 \\ \dots \\ \mathbf{m}_{M-1} \end{matrix}, \quad \mathbf{M}\mathbf{M}^\top = \mathbf{I}_M \quad (33)$$

According to the construction of the filter bank proposed in [18] and used in [11], based on successive one-sample circular shifts, \mathbf{M} displays Toeplitz structure. Starting from one single MLS sequence \mathbf{m}_0 , we then apply circular shifts to generate all other rows; the elements of which are $m_i[n] = m_0[\langle n+i \rangle]$. This causes matrix \mathbf{M} to be orthonormal [19], provided that the appropriate normalization of the sequences is introduced to ensure that $\mathbf{m}_i \mathbf{m}_j^\top = 1$ for $i=j$. Other sequences may exist that satisfy this condition [20], the evaluation of which is out of the scope of this paper.

The identification of the system under test just requires the computation of $|\mathfrak{Q}(\mathcal{M})|$ coefficients of vector \mathbf{g} from the Wiener representation: exactly those contained in $\hat{\mathbf{g}}$. To solve the identification problem for those coefficients, we then need the same number of output samples $y[n]$ after Eq. (6):

$$\mathbf{y} = \mathbf{A} \hat{\mathbf{g}} \quad (34)$$

For ease of explanation, no gain will be applied to the input signal. This possibility is covered in [11]. To derive Eq. (34) we start from Eq. (6) and use the expansion and reduction matrices defined in Eqs. (19) and (21) to reformulate the generalized convolution:

$$y[n] = \sum_{r=1}^N \mathbf{u}_i^{[r] \otimes} \underbrace{\mathbf{M}^{[r] \otimes} \mathbf{P}_r \mathbf{Q}_r}_{\mathbf{h}_r} \quad (35)$$

where

$$\mathbf{h}_r = \mathbf{M}^{[r]} \otimes \mathbf{g}_r \quad (36)$$

Matrix \mathbf{A} is then

$$\mathbf{A} = \begin{bmatrix} \mathbf{u}_0 \mathbf{M} \mathbf{P}_1 & (\mathbf{u}_0 \mathbf{M})^{[2]} \otimes \mathbf{P}_2 & \dots & (\mathbf{u}_0 \mathbf{M})^{[N]} \otimes \mathbf{P}_N \\ \mathbf{u}_1 \mathbf{M} \mathbf{P}_1 & (\mathbf{u}_1 \mathbf{M})^{[2]} \otimes \mathbf{P}_2 & \dots & (\mathbf{u}_1 \mathbf{M})^{[N]} \otimes \mathbf{P}_N \\ \vdots & \vdots & \dots & \vdots \\ \mathbf{u}_{|\mathcal{Q}(\mathcal{M})|-1} \mathbf{M} \mathbf{P}_1 & (\mathbf{u}_{|\mathcal{Q}(\mathcal{M})|-1} \mathbf{M})^{[2]} \otimes \mathbf{P}_2 & \dots & (\mathbf{u}_{|\mathcal{Q}(\mathcal{M})|-1} \mathbf{M})^{[N]} \otimes \mathbf{P}_N \end{bmatrix} \quad (37)$$

and can be used to rewrite Eq. (35) as the matrix product in Eq. (34). The $|\mathcal{Q}(\mathcal{M})|$ sequences \mathbf{u}_i make up the probe input signal. A simple probe matrix \mathbf{X} is assembled from MLS sequences $\{\mathbf{m}_i\}$ so as to form $|\mathcal{Q}(\mathcal{M})|$ lines with M samples per line. The individual sequences \mathbf{m}_i are generated from one single MLS sequence \mathbf{m}_0 as we did before for matrix \mathbf{M} . The resulting matrix \mathbf{X} is built row-wise from sequences \mathbf{m} in the way the following equation states. The probe sequence that is finally used to excite the system is extracted from matrix \mathbf{X} as proposed in [11]. Authors described there how to shorten the excitation length attending to repeated patterns arising in \mathbf{X} , and reconstruct from the respective system output the output that corresponds to \mathbf{X} :

$$\mathbf{x}_i = \sum_{k=1}^{|\mathbf{q}_i|} \mathbf{m}_{\mathbf{q}_i[k]}^T \quad \text{and} \quad \mathbf{X}^T = [\mathbf{x}_1^T | \mathbf{x}_2^T | \dots | \mathbf{x}_{|\mathcal{Q}(\mathcal{M})}|}^T] \quad (38)$$

Matrix \mathbf{A} , as defined by Eq. (37), can be indexed by means of set \mathcal{I} :

$$a_{ij} = \mathbf{x}_i^{[r]} \otimes_{s=1}^r \mathbf{m}_{\mathbf{q}_j[s]} \Big|_{i,j \in \mathcal{I} \wedge r = |\mathbf{q}_j|} \quad (39)$$

The mixed product rule can then be applied to arbitrary column vectors \mathbf{b} and \mathbf{c}_i of the same size, with the result $(\mathbf{b}^T)^{[r]} \otimes (\mathbf{c}_1 \otimes \mathbf{c}_2 \otimes \dots \otimes \mathbf{c}_r) = \mathbf{b}^T \mathbf{c}_1 \otimes \mathbf{b}^T \mathbf{c}_2 \otimes \dots \otimes \mathbf{b}^T \mathbf{c}_r$ [16]. Consequently, the preceding equation reduces to

$$a_{ij} = \prod_{s=1}^{|\mathbf{q}_j|} \sum_{k=1}^{|\mathbf{q}_i|} \delta(\mathbf{q}_i[k] - \mathbf{q}_j[s]) \quad (40)$$

Where $\delta(\cdot)$ is Kronecker's delta-function. From this we know that matrix \mathbf{A} is sparse and its many null elements ease the process of inversion to calculate the coefficients of vector $\hat{\mathbf{g}}$ based on \mathbf{y} . Further considerations on MLS-based measuring are provided in [21], while an extensive analysis on their properties and examples of other sequences that could be used are provided in [22]. Once these are computed, the coefficients from the discrete Volterra kernels may be computed from Eq. (36).

3. Orthogonality loss due to DC level

In most applications involving nonlinear systems, average input level (e.g. the direct current level or polarization level for electrical and electronic devices) as well as the input dynamic range need to be appropriately arranged. Operation point and input excursion determine the system behavior; so their values must be informedly chosen.

Dynamic range adjustments in MLS-based nonlinear systems' identification had already been addressed in [11] where an adjustable gain parameter α was introduced to the input matrix \mathbf{X} . On the contrary, no prior contribution has evaluated the impact of DC-level. Hereafter we evidence that a non-zero DC-level in the system excitation dissipates the required orthogonality between the input and the filter bank impulse responses, invalidating the MLS identification method, unless offset effect is somehow compensated.

Let us assume that a gain α is applied to the input signal, as well as a DC-offset with a normalized value ϵ in relation to α . The output of the system is then

$$y[n] = \sum_{r=1}^N (1/\alpha)^r (\mathbf{u}_n + \epsilon)^{[r]} \mathbf{h}_r \quad (41)$$

where ϵ is a vector form resulting from the expansion of ϵ by a single-valued 1-vector, and the adjustable gain α must be introduced to compensate the effect to the amplified probe defined by matrix \mathbf{M} .

Despite remodeling, one may write output $y[n]$ as a matrix transformation of \mathbf{g} , provided that the definition of \mathbf{A} in Eq. (37) is changed accordingly to reflect the input offset. Let now $\hat{\mathbf{A}}$ be the biased matrix. In this case, $\mathbf{y} = \hat{\mathbf{A}} \hat{\mathbf{g}}$.

Solving this equation for $\hat{\mathbf{g}}$ will provide values for $\hat{\mathbf{A}}$, but not for the non-biased matrix \mathbf{A} in which we are interested. To focus on \mathbf{A} , a biased set of coefficients $\hat{\mathbf{g}}$ may be obtained instead. These must account for the aforementioned biases in the excitation signal, and when projected by means of Eq. (36), shall result in a set of biased Volterra kernels $\hat{\mathbf{h}}_r$, instead of the

expected \mathbf{h}_r . Therefore, when considering the output signal $y[n]$, the following equivalence holds:

$$\mathbf{y} = \tilde{\mathbf{A}} \hat{\mathbf{g}} = \tilde{\mathbf{A}} \hat{\mathbf{g}} \quad (42)$$

When $\tilde{\mathbf{A}}$ is inverted and multiplied by \mathbf{y} , instead of using \mathbf{A} , the biased vector $\tilde{\mathbf{h}}_r$ of Volterra kernels are noticeably perturbed. The relationship between \mathbf{h} and $\tilde{\mathbf{h}}$ can be established if the nature of the bias is taken into account.

3.1. Biased coefficients and true coefficients

The input matrix \mathbf{X} , defined in Eq. (38), changes due to offset, resulting in matrix $\tilde{\mathbf{X}}$, the rows of which are

$$\tilde{\mathbf{x}}_i = \alpha \left(\boldsymbol{\varepsilon}^\top + \sum_{k=1}^{|\mathbf{q}_i|} \mathbf{m}_{\mathbf{q}_i[k]}^\top \right) \quad (43)$$

The effect of the DC offset, with normalized value ε with respect to α , is here introduced by $\boldsymbol{\varepsilon}$, the single-valued, M -dimensional column vector extended from constant value ε . Factor α accounts for the input signal amplitude.

Arranging the new input sequences Eq. (43) attending to the definition of \mathbf{A} in Eq. (39), which contains the information about the system input, the elements of the resulting biased matrix $\tilde{\mathbf{A}}$ happen to be

$$\tilde{a}_{ij} = \tilde{\mathbf{x}}_i^{[r]} \otimes_{s=1}^r \mathbf{m}_{\mathbf{q}_j[s]} = \alpha^r \left(\boldsymbol{\varepsilon}^\top + \sum_{k=1}^{|\mathbf{q}_i|} \mathbf{m}_{\mathbf{q}_i[k]}^\top \right)^{[r]} \otimes_{s=1}^r \mathbf{m}_{\mathbf{q}_j[s]} \Big|_{i,j \in \mathcal{I} \wedge r = |\mathbf{q}_j|} \quad (44)$$

The properties of mixed Kronecker product can now be applied, as we did to derive Eq. (40). Taking into account that $\mathbf{1}_M^\top \mathbf{M} = \mathbf{1}_M^\top$, where $\mathbf{1}_M$ is a column vector with M unitary elements, matrix $\tilde{\mathbf{A}}$ elements may be rewritten as

$$\tilde{a}_{ij} = \alpha^r \prod_{s=1}^r \left(\varepsilon + \sum_{k=1}^{|\mathbf{q}_i|} \delta(\mathbf{q}_i[k] - \mathbf{q}_j[s]) \right) \Big|_{i,j \in \mathcal{I} \wedge r = |\mathbf{q}_j|} \quad (45)$$

According to the connection between matrices \mathbf{A} and $\tilde{\mathbf{A}}$ in Eq. (42), and exploiting the definition of compression matrices \mathbf{Q}_r in Eq. (20), the connection between coefficients \mathbf{g} and $\tilde{\mathbf{g}}$ may be formulated as follows, provided that system-characteristic matrix \mathbf{A} is not singular—see [11] for a proof on \mathbf{A} 's invertibility:

$$\begin{bmatrix} \mathbf{Q}_1 \tilde{\mathbf{g}}_1 \\ \mathbf{Q}_2 \tilde{\mathbf{g}}_2 \\ \dots \\ \mathbf{Q}_N \tilde{\mathbf{g}}_N \end{bmatrix} = \mathbf{A}^{-1} \tilde{\mathbf{A}} \begin{bmatrix} \mathbf{Q}_1 \mathbf{g}_1 \\ \mathbf{Q}_2 \mathbf{g}_2 \\ \dots \\ \mathbf{Q}_N \mathbf{g}_N \end{bmatrix} \quad (46)$$

From now on, subindex $[\mathbf{X}]_{\mathcal{Y}\mathcal{Z}}$ indicates a selection of elements of matrix \mathbf{X} . The selection includes those rows indexed by \mathcal{Y} and columns indexed by \mathcal{Z} . Whenever the whole set of lines or columns is selected, a dot \bullet will be used. For instance,

$$\mathbf{X} = \begin{array}{c} \mathbf{X}_{\bullet\mathcal{Y}} \quad \mathbf{X}_{\mathcal{Y}\mathcal{Z}} \\ \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{bmatrix} \end{array} \mathbf{X}_{\mathcal{W}\bullet}, \quad \text{where } \mathcal{W} = \{2, 3\} \\ \mathcal{Y} = \{1, 2\} \\ \mathcal{Z} = \{2, 3, 4\} \quad (47)$$

With this notation and the set of partial matrix products in Eq. (46), one may write each of these partial products as

$$\mathbf{Q}_r \tilde{\mathbf{g}}_r = [[\mathbf{A}^{-1} \tilde{\mathbf{A}}]_{\mathcal{I}_1} \mathbf{Q}_1 \mathbf{g}_1] [[\mathbf{A}^{-1} \tilde{\mathbf{A}}]_{\mathcal{I}_2} \mathbf{Q}_2 \mathbf{g}_2] \dots [[\mathbf{A}^{-1} \tilde{\mathbf{A}}]_{\mathcal{I}_N} \mathbf{Q}_N \mathbf{g}_N]_{\mathcal{I}_r} \quad (48)$$

where \mathcal{I}_r are the indices connected to the r -multicombinations as defined in Eq. (11). Given the fact that coefficients $\tilde{\mathbf{g}}$ and $\tilde{\mathbf{h}}$ share the same symmetry, properties in Eq. (24) ought to be exploited so that when Eqs. (48) and (36) are combined we get

$$\tilde{\mathbf{h}}_r = \mathbf{M}^{[r]} \otimes \mathbf{P}_r [[\mathbf{A}^{-1} \tilde{\mathbf{A}}]_{\mathcal{I}_1} \mathbf{Q}_1 \mathbf{M}^\top [[\mathbf{A}^{-1} \tilde{\mathbf{A}}]_{\mathcal{I}_2} \mathbf{Q}_2 (\mathbf{M}^\top)^{[2]} \dots [[\mathbf{A}^{-1} \tilde{\mathbf{A}}]_{\mathcal{I}_N} \mathbf{Q}_N (\mathbf{M}^\top)^{[N]}]_{\mathcal{I}_r}]_{\mathcal{I}_r} \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \dots \\ \mathbf{h}_N \end{bmatrix} \quad (49)$$

We shall now shed some light on matrix product $\mathbf{A}^{-1} \tilde{\mathbf{A}}$, bearing in mind that both matrices result from entrywise operations. Matrix \mathbf{A} is formed from matrices $\mathbf{Y}_t^{x\mathcal{Z}}$, where z indicates the order of the coefficients, x is the sequences' length with $x \leq z$, and t is one of those sequences that results from combinations without repetition of length x and z elements. Column-wise indexation is done upon indices \mathcal{L}_r (see Fig. 2). The set of combinations without repetition of size x and

elements from set \mathcal{Z} is hereafter noted as $\mathfrak{B}_x(\mathcal{Z})$. The result is the following definition for the matrices:

$$\exists! \mathbf{Y}_t^{\alpha z} \left| v_t^{\alpha z}(i, l) = \prod_{n=1}^x \sum_{k=1}^{|\mathcal{q}_l|} \delta(q_i[k] - q_l[\mathfrak{t}[n]]) \quad \forall q_i \in \mathfrak{Q}(\mathcal{M}), \forall q_l \in \mathfrak{Q}_z(\mathcal{M}) \right. \\ \left. \mathfrak{t} \in \mathfrak{B}_x(\mathcal{Z}), \text{ siendo } \mathcal{Z} = \{1, \dots, \mathcal{Z}\} \right. \quad (50)$$

As an example, according to Eq. (50) for $M=3, N=3$ and $\mathfrak{t} = \{1, 3\}$ we have that

$$\mathbf{Y}_{\{13\}}^{23} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 2 \\ 0 & 0 & 0 & 1 & 1 & 2 & 1 & 1 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 2 & 1 & 2 & 2 & 2 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \\ 9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 2 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 \\ 4 & 0 & 2 & 0 & 2 & 2 & 0 & 0 & 0 & 1 \\ 1 & 2 & 0 & 2 & 0 & 0 & 4 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 2 & 0 & 2 & 2 & 0 & 0 & 0 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & 2 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 2 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 9 \end{bmatrix} \begin{matrix} 0 \\ 1 \\ 2 \\ 00 \\ 01 \\ 02 \\ 11 \\ 12 \\ 22 \\ 000 \\ 001 \\ 002 \\ 011 \\ 012 \\ 022 \\ 111 \\ 112 \\ 122 \\ 222 \end{matrix}$$

From Eq. (9) and our previous definition of matrices $\mathbf{Y}_t^{\alpha z}$ one may write \mathbf{A} —the unperturbed matrix—as

$$\mathbf{A} = [\mathbf{Y}_{\{1\}}^{\alpha 11}; \mathbf{Y}_{\{1,2\}}^{\alpha 22}; \mathbf{Y}_{\{1,2,3\}}^{\alpha 33}; \dots] \quad (51)$$

Similarly, matrix $\tilde{\mathbf{A}}$ including the perturbations induced by DC-offset term ε and arbitrary input gain α may also be decomposed in terms of $\mathbf{Y}_t^{\alpha z}$, though the resulting form is slightly different:

$$\tilde{\mathbf{A}} = \left[\begin{array}{c|c|c|c} \alpha \mathbf{Y}_{\{1\}}^{\alpha 11} + \alpha \mathbf{E}_1 & & & \\ & \alpha^2 \mathbf{Y}_{\{1,2\}}^{\alpha 22} + & & \\ & + \alpha^2 \varepsilon \mathbf{Y}_{\{1\}}^{\alpha 12} + \alpha^2 \varepsilon \mathbf{Y}_{\{2\}}^{\alpha 12} + & & \\ & + \alpha^2 \mathbf{E}_2^{[2]o} & & \\ & & \alpha^3 \mathbf{Y}_{\{1,2,3\}}^{\alpha 33} + & \\ & & + \alpha^3 \varepsilon \mathbf{Y}_{\{1,2\}}^{\alpha 23} + \alpha^3 \varepsilon \mathbf{Y}_{\{1,3\}}^{\alpha 23} + \alpha^3 \varepsilon \mathbf{Y}_{\{2,3\}}^{\alpha 23} + & \\ & & + \alpha^3 \varepsilon^2 \mathbf{Y}_{\{1\}}^{\alpha 13} + \alpha^3 \varepsilon^2 \mathbf{Y}_{\{2\}}^{\alpha 13} + \alpha^3 \varepsilon^2 \mathbf{Y}_{\{3\}}^{\alpha 13} + & \\ & & + \alpha^3 \mathbf{E}_3^{[3]o} & \\ & & & \dots \end{array} \right] \quad (52)$$

where \mathbf{E}_z is a single-valued, constant matrix of $|\mathfrak{Q}(\mathcal{M})|$ rows and $|\mathfrak{Q}_z(\mathcal{M})|$ columns, with values equal to constant value ε . A z times Hadamard (entrywise) product on this matrix is noted by $\mathbf{E}^{[z]o}$.

Hereby we write the $\mathbf{A}^{-1} \tilde{\mathbf{A}}$ product as the subsequent matrices' summation:

$$\mathbf{A}^{-1} \tilde{\mathbf{A}} = \underbrace{\begin{bmatrix} \overbrace{\alpha \mathbf{I}_{|\mathcal{J}_1|}}^{\alpha \mathbf{1}} & \mathbf{0}_{|\mathcal{J}_1|, |\mathcal{J}_2|} & \mathbf{0}_{|\mathcal{J}_1|, |\mathcal{J}_3|} & \dots & \mathbf{0}_{|\mathcal{J}_1|, |\mathcal{J}_N|} \\ \mathbf{0}_{|\mathcal{J}_2|, |\mathcal{J}_1|} & \alpha^2 \mathbf{I}_{|\mathcal{J}_2|} & \mathbf{0}_{|\mathcal{J}_2|, |\mathcal{J}_3|} & \dots & \mathbf{0}_{|\mathcal{J}_2|, |\mathcal{J}_N|} \\ \mathbf{0}_{|\mathcal{J}_3|, |\mathcal{J}_1|} & \mathbf{0}_{|\mathcal{J}_3|, |\mathcal{J}_2|} & \alpha^3 \mathbf{I}_{|\mathcal{J}_3|} & \dots & \mathbf{0}_{|\mathcal{J}_3|, |\mathcal{J}_N|} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{|\mathcal{J}_N|, |\mathcal{J}_1|} & \mathbf{0}_{|\mathcal{J}_N|, |\mathcal{J}_2|} & \dots & \dots & \alpha^N \mathbf{I}_{|\mathcal{J}_N|} \end{bmatrix}}_{\text{Cross-order interferences}} + \underbrace{\mathbf{A}^{-1} [\alpha \mathbf{E}_1; \alpha^2 \mathbf{E}_2^{[2]o}; \alpha^3 \mathbf{E}_3^{[3]o}; \dots]}_{\text{DC+gain interferences}} + \mathbf{A}^{-1} \left[\begin{array}{c|c|c|c} \mathbf{0}_{|\mathfrak{Q}(\mathcal{M}), M} & \alpha^2 \varepsilon (\mathbf{Y}_{\{1\}}^{\alpha 12} + \mathbf{Y}_{\{2\}}^{\alpha 12}) & \alpha^3 \varepsilon (\mathbf{Y}_{\{1,2\}}^{\alpha 23} + \mathbf{Y}_{\{1,3\}}^{\alpha 23} + \mathbf{Y}_{\{2,3\}}^{\alpha 23}) + & \\ & & + \alpha^3 \varepsilon^2 (\mathbf{Y}_{\{1\}}^{\alpha 13} + \mathbf{Y}_{\{2\}}^{\alpha 13} + \mathbf{Y}_{\{3\}}^{\alpha 13}) & \dots \end{array} \right] \quad (53)$$

From Eqs. (46) and (53) one may realize that perturbed coefficients $\tilde{\mathbf{g}}$ convey the effect of the original \mathbf{g} coefficients weighted by arbitrary gain factor α , plus two additional, spurious effects.

Hereafter we denote by $\overbrace{\alpha \mathbf{I}}^{\alpha \mathbf{1}}$ the identity matrix of size $|\mathcal{I}| \times |\mathcal{I}|$ weighted by α^r —the first term in Eq. (53)—, where r is the order associated to the indices subset \mathcal{I}_r . Similar to Eq. (53), the analogous matrix indexed by \mathcal{J} and referring to the permutations with repetition—expanded formulation—of size $|\mathcal{J}| \times |\mathcal{J}|$ will be noted as $\alpha \mathbf{I}$.

The two additional, additive terms have significantly different natures. The first of these, being the second addend in Eq. (53), corresponds in the Volterra domain to the projection on coefficients $\tilde{\mathbf{h}}_r$ of the true coefficients \mathbf{h}_r according to a set of projection matrices, $\{\mathbf{K}_{E_r}\}$; that is essentially a set of constant matrices relating the DC-offset level ε and the input gain α .

The second (third addend) results in projection according to matrices $\{\mathbf{K}_{Y_r}\}$, and accounts for the interferences introduced by incomplete-order matrices \mathbf{Y}_t^{xz} with $x < z$.

3.2. Analysis on DC and gain interference: \mathbf{K}_E

From Eq. (49), and noticing that row-wise summation of elements from matrices \mathbf{M} (in Eq. (33)), \mathbf{Q}_r (in Eq. (20)) and \mathbf{P}_r (in Eq. (22)) make one, we may reformulate sub-matrices \mathbf{K}_{E_r} and \mathbf{K}_{Y_r} in the Volterra domain.

Matrices \mathbf{K}_{E_r} associated to DC-offset level and input gain may be written as

$$\begin{aligned} \mathbf{K}_{E_r} &= \mathbf{M}^{[r]} \circ \mathbf{P}_r [\alpha \varepsilon \mathbf{A}^{-1} \mathbf{1}_{|\mathcal{I}|} \mathbf{1}_{|\mathcal{J}_1|}^\top \mathbf{Q}_1 \mathbf{M}^\top \dots \alpha^N \varepsilon^N \mathbf{A}^{-1} \mathbf{1}_{|\mathcal{I}|} \mathbf{1}_{|\mathcal{J}_N|}^\top \mathbf{Q}_r (\mathbf{M}^\top)^{[r]}]_{\mathcal{I}_r} \\ &= \mathbf{M}^{[r]} \circ \mathbf{P}_r [\alpha \varepsilon \mathbf{A}^{-1} \mathbf{U}_{|\mathcal{I}|, |\mathcal{J}_1|} \dots \alpha^N \varepsilon^N \mathbf{A}^{-1} \mathbf{U}_{|\mathcal{I}|, |\mathcal{J}_N|}]_{\mathcal{I}_r} \end{aligned} \quad (54)$$

where $\mathbf{1}_x$ is the all-ones, column vector of length x and $\mathbf{U}_{x,z}$ is an all-ones, constant matrix of size $x \times z$. Thus, from matrices \mathbf{K}_{E_r} we may build

$$\mathbf{K}_E = \begin{bmatrix} \mathbf{M} \mathbf{P}_1 [\mathbf{A}^{-1}]_{\mathcal{I}_1} \\ \mathbf{M}^{[2]} \circ \mathbf{P}_2 [\mathbf{A}^{-1}]_{\mathcal{I}_2} \\ \vdots \\ \mathbf{M}^{[N]} \circ \mathbf{P}_N [\mathbf{A}^{-1}]_{\mathcal{I}_N} \end{bmatrix} [\alpha \mathbf{E}_1 \mid \alpha^2 \mathbf{E}_2^{[2]_0} \mid \alpha^3 \mathbf{E}_3^{[3]_0} \mid \dots] \quad (55)$$

where now \mathbf{E}_x is a single-valued, constant matrix of $|\mathcal{I}|$ rows and $|\mathcal{J}_x|$ columns, and elements equal to ε .

To compute the contribution of elements in rows \mathbf{K}_{E_r} defined in Eq. (54), it may seem necessary to compute inverse matrix \mathbf{A}^{-1} from the highly-dimensional, sparse matrix \mathbf{A} . Nevertheless, we may avoid this attending to

$$[\mathbf{A}_{\bullet \mathcal{J}_1} \mathbf{Q}_1 \mid \mathbf{A}_{\bullet \mathcal{J}_2} \mathbf{Q}_2 \mid \dots \mid \mathbf{A}_{\bullet \mathcal{J}_N} \mathbf{Q}_N] \cdot \left[\left(\mathbf{P}_1 [\mathbf{A}^{-1}]_{\mathcal{I}_1} \right)^\top \mid \left(\mathbf{P}_2 [\mathbf{A}^{-1}]_{\mathcal{I}_2} \right)^\top \mid \dots \mid \left(\mathbf{P}_N [\mathbf{A}^{-1}]_{\mathcal{I}_N} \right)^\top \right]^\top = \mathbf{I}_{|\mathcal{I}|} \quad (56)$$

By simply multiplying Eq. (56) on both sides by matrix $[\alpha \mathbf{E}_1 \mid \alpha^2 \mathbf{E}_2^{[2]_0} \mid \alpha^3 \mathbf{E}_3^{[3]_0} \mid \dots]$ we observe that

$$\underbrace{[\mathbf{A}_{\bullet \mathcal{J}_1} \mathbf{Q}_1 \mid \dots \mid \mathbf{A}_{\bullet \mathcal{J}_N} \mathbf{Q}_N]}_{\text{Matrix 1}} \underbrace{\begin{bmatrix} \mathbf{M} \mathbf{P}_1 [\mathbf{A}^{-1}]_{\mathcal{I}_1} \\ \mathbf{M}^{[2]} \circ \mathbf{P}_2 [\mathbf{A}^{-1}]_{\mathcal{I}_2} \\ \vdots \\ \mathbf{M}^{[N]} \circ \mathbf{P}_N [\mathbf{A}^{-1}]_{\mathcal{I}_N} \end{bmatrix}}_{\text{Matrix 2}} [\alpha \mathbf{E}_1 \mid \dots \mid \alpha^N \mathbf{E}_N^{[N]_0}] = [\alpha \mathbf{E}_1 \mid \dots \mid \alpha^N \mathbf{E}_N^{[N]_0}] \quad (57)$$

For Matrix 1 in Eq. (57) row-wise addition of the elements corresponding to the r -th column-wise partition of the matrix $-\mathbf{A}_{\bullet \mathcal{J}_r} \mathbf{Q}_r$ —are all equal along rows associated to the same order s , since row-wise addition along Eq. (40) results in

$$\sum_{j \in \mathcal{J}_r} \prod_{l=1}^r \sum_{k=1}^{|\mathbf{q}_l|} \delta(\mathbf{q}_i[k] - \mathbf{d}_j[l]) = |\mathcal{D}_r(\mathfrak{S})| = |\mathbf{q}_i|^r \quad \text{where } \mathfrak{S} = \{1, \dots, |\mathbf{q}_i|\} = \{1, \dots, s\} \quad (58)$$

Moreover, despite lacking any prior knowledge on the values for the first component of Matrix 2 in Eq. (57), we deduce that this must display the same constant values row-wise. Now based on Eqs. (57) and (58), matrix \mathbf{K}_E is completely defined by

$$\mathbf{K}_{E_r} = [\varepsilon_{r1} \mathbf{U}_{|\mathcal{I}_r|, |\mathcal{J}_1|} \mid \varepsilon_{r2} \mathbf{U}_{|\mathcal{I}_r|, |\mathcal{J}_2|} \mid \dots \mid \varepsilon_{rN} \mathbf{U}_{|\mathcal{I}_r|, |\mathcal{J}_N|}] \quad (59)$$

where the only unknowns are $\{\varepsilon_{rx}\}$. Hence, the inversion problem is reduced to a system of N^2 linear equations in ε :

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ 2 & 2^2 & \dots & 2^N \\ \vdots & \vdots & \ddots & \vdots \\ N & N^2 & \dots & N^N \end{bmatrix} \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \dots & \varepsilon_{1N} \\ \varepsilon_{21} & \varepsilon_{22} & \dots & \varepsilon_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon_{N1} & \varepsilon_{N2} & \dots & \varepsilon_{NN} \end{bmatrix} = \begin{bmatrix} \alpha \varepsilon & \alpha^2 \varepsilon^2 & \dots & \alpha^N \varepsilon^N \\ \alpha \varepsilon & \alpha^2 \varepsilon^2 & \dots & \alpha^N \varepsilon^N \\ \vdots & \vdots & \ddots & \vdots \\ \alpha \varepsilon & \alpha^2 \varepsilon^2 & \dots & \alpha^N \varepsilon^N \end{bmatrix} \quad (60)$$

3.3. Analysis on induced cross-order interference: \mathbf{K}_Y

Having disentangled the form of matrix \mathbf{K}_{E_r} , we move on to matrix \mathbf{K}_{Y_r} . To do so we first define binary matrix $\overset{\Delta}{\Gamma}_t^{xz}$. Elements from this matrix are all null, except for pairs (i, l) for which elements indexed by rows \mathbf{q}_i are all equal to those indexed by \mathbf{q}_l , which we select through sequence \mathbf{t} :

$$\exists! \overset{\Delta}{\Gamma}_t^{xz} \Big| \overset{\Delta}{\gamma}_t^{xz}(i, l) = \begin{cases} 0 & \text{si } |\mathbf{q}_i| \neq x & \forall \mathbf{q}_i \in \mathfrak{Q}(\mathcal{M}) \\ \prod_{n=1}^x \delta(\mathbf{q}_i[n] - \mathbf{q}_l[\mathbf{t}[n]]) & \text{si } |\mathbf{q}_i| = x & \forall l \in \mathcal{L}_2(\mathcal{M}) \mid \mathbf{q}_l \in \mathfrak{Q}_2(\mathcal{M}) \\ & & \mathbf{t} \in \mathfrak{B}_x(\mathcal{Z}) \text{ siendo } \mathcal{Z} = \{1, \dots, z\} \end{cases} \quad (61)$$

As an example, for a sequence $\mathbf{t} = \{1, 3\}$, with $x=2$ and $z=3$ for a system with memory length $M=3$, matrix $\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz}$ corresponds to

$$\hat{\Gamma}_{\{1,3\}}^{\Delta, 23} = \begin{bmatrix} \mathbf{0}_{3,10} & 0,1,2 \\ \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 2 \\ 0 & 0 & 0 & 1 & 1 & 2 & 1 & 1 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 2 \end{matrix} & \begin{matrix} 00 \\ 01 \\ 02 \\ 11 \\ 12 \\ 22 \\ 000,001,\dots,222 \end{matrix} \end{bmatrix}$$

Considering now the product $\mathbf{A}\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz}$, we observe that this equates $\mathbf{Y}_{\mathbf{t}}^{xz}$ since

$$\begin{aligned} [\mathbf{A}\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz}]_{i,l} &= \sum_{\forall \mathbf{q}_q \in \mathfrak{D}_x(\mathcal{M})} a(i, q) \gamma_{\mathbf{t}}^{\Delta, xz}(q, l) \\ &= \sum_{\forall \mathbf{q}_q \in \mathfrak{D}_x(\mathcal{M})} \left[\left(\prod_{n=1}^x \sum_{k=1}^s \delta(q_i[k] - q_q[x]) \right) \prod_{m=1}^z \delta(q_q[m] - q_l[\mathbf{t}[m]]) \right] \\ &= \prod_{n=1}^x \sum_{k=1}^{|\mathbf{q}_i|} \delta(q_i[k] - q_l[\mathbf{t}[m]]) = v_{\mathbf{t}}^{xz}(i, l) \end{aligned} \quad (62)$$

Consequently, one may write the third addend in Eq. (53) as a combination of matrices $\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz} = \mathbf{A}^{-1} \mathbf{Y}_{\mathbf{t}}^{xz}$ defined by Eq. (61), avoiding \mathbf{A}^{-1} computation:

$$\begin{aligned} \mathbf{A}^{-1} \left[\begin{array}{c|c|c|c} \mathbf{0}_{\mathfrak{D}(\mathcal{M}), M} & \alpha^2 \varepsilon(\mathbf{Y}_{(1)}^{12} + \mathbf{Y}_{(2)}^{12}) & \alpha^3 \varepsilon(\mathbf{Y}_{(1,2)}^{23} + \mathbf{Y}_{(1,3)}^{23} + \mathbf{Y}_{(2,3)}^{23}) + \alpha^3 \varepsilon^2(\mathbf{Y}_{(1)}^{13} + \mathbf{Y}_{(2)}^{13} + \mathbf{Y}_{(3)}^{13}) & \dots \end{array} \right] = \\ = \left[\begin{array}{c|c|c|c} \mathbf{0}_{\mathcal{I}, M} & \alpha^2 \varepsilon(\hat{\Gamma}_{(1)}^{12} + \hat{\Gamma}_{(2)}^{12}) & \alpha^3 \varepsilon(\hat{\Gamma}_{(1,2)}^{23} + \hat{\Gamma}_{(1,3)}^{23} + \hat{\Gamma}_{(2,3)}^{23}) + \alpha^3 \varepsilon^2(\hat{\Gamma}_{(1)}^{13} + \hat{\Gamma}_{(2)}^{13} + \hat{\Gamma}_{(3)}^{13}) & \dots \end{array} \right] \end{aligned} \quad (63)$$

Once the influence of coefficients $\tilde{\mathbf{g}}_r$ on \mathbf{g}_r has been described, we deduce how this reflects in the Volterra domain.

Starting from Eq. (61), and bearing in mind that matrices $\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz}$ only display non-null values in elements from rows where $|\mathbf{q}_i| = x$ holds, to derive the projections on Volterra's coefficients one must expand these matrices to index on $\mathfrak{d} \in \mathfrak{D}(\mathcal{M})$ —partial permutations with repetition—, rather than $\mathbf{q}_i \in \mathfrak{D}(\mathcal{M})$ —multicombinations.

Matrix $\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz}$ defined in Eq. (61) is now re-indexed so that $\gamma_{\mathbf{t}}^{\Delta, xz}(i, l)$ corresponds to elements $\forall i \in \mathcal{I}_x$ —indexing sequences $\mathfrak{d}_i \in \mathfrak{D}_x(\mathcal{M})$ —and $\forall l \in \mathcal{I}_z$ —indexing $\mathfrak{d}_l \in \mathfrak{D}_z(\mathcal{M})$. Such re-indexation of matrix $\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz}$ is equivalent to compression and expansion

involving the reduced matrix: $\mathbf{P}_x \hat{\Gamma}_{\mathbf{t}}^{\Delta, xz} \mathbf{I}_x \bullet \mathbf{Q}_z$. Noticing that the following property holds for matrix $\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz}$:

$$\mathbf{M}^{[r_1] \otimes} \sum_{\forall \mathbf{t} \in \mathfrak{B}_x(\mathcal{Z})} \hat{\Gamma}_{\mathbf{t}}^{r_1 r_2} (\mathbf{M}^\top)^{[r_2] \otimes} = \sum_{\forall \mathbf{t} \in \mathfrak{B}_x(\mathcal{Z})} \hat{\Gamma}_{\mathbf{t}}^{r_1 r_2} \quad (64)$$

and that matrices $\hat{\Gamma}_{\mathbf{t}}^{\Delta, xz}$ only display non-null values at positions i, l corresponding respectively to orders x, z ; one may follow the same rationale used to derive \mathbf{K}_{E_r} from Eq. (49), to rewrite the third addend in Eq. (53) attending to Eqs. (63) and (64):

$$\mathbf{K}_{Y_r} = \left[\begin{array}{c|c|c|c|c} \mathbf{0} & \alpha^{r+1} \varepsilon \sum_{\mathfrak{B}_r(\mathcal{R}_{r+1})} \hat{\Gamma}_{\mathbf{t}}^{r, r+1} & \alpha^{r+2} \varepsilon^2 \sum_{\mathfrak{B}_r(\mathcal{R}_{r+2})} \hat{\Gamma}_{\mathbf{t}}^{r, r+2} & \dots & \alpha^N \varepsilon^{N-r} \sum_{\mathfrak{B}_r(\mathcal{R}_N)} \hat{\Gamma}_{\mathbf{t}}^{r, N} \end{array} \right] \quad (65)$$

where $\mathcal{R}_x = \{1, 2, \dots, x\}$ and $\mathfrak{B}_r(\mathcal{R}_x)$ is the set of combinations without repetition of size r built from \mathcal{R}_x .

Once matrices \mathbf{K}_{E_r} and \mathbf{K}_{Y_r} are completely defined, we may conclude that the connection between Volterra's biased ($\tilde{\mathbf{h}}$) and true coefficients (\mathbf{h}) is given by

$$\begin{bmatrix} \tilde{\mathbf{h}}_1 \\ \tilde{\mathbf{h}}_2 \\ \vdots \\ \tilde{\mathbf{h}}_N \end{bmatrix} = \left(\mathbf{I} + \begin{bmatrix} \mathbf{K}_{E1} + \mathbf{K}_{Y1} \\ \mathbf{K}_{E2} + \mathbf{K}_{Y2} \\ \vdots \\ \mathbf{K}_{EN} + \mathbf{K}_{YN} \end{bmatrix} \right) \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_N \end{bmatrix} \quad (66)$$

4. Correction of the deviations induced in triangular Volterra kernels

The general expression obtained for the projection of \mathbf{h} on $\tilde{\mathbf{h}}$ in Eq. (66) accounts for the deviations introduced to system identification due to DC-offset and arbitrary gain in the signals used to excite the system under test.

Unfortunately, the matrix connecting \mathbf{h} and $\tilde{\mathbf{h}}$ is intractable from a computational point of view, even for a short-memory, weakly nonlinear system. One may be able to overcome this situation by using the expansion and compression matrices we previously introduced, and focusing on the reduced, triangular Volterra kernels $\hat{\mathbf{h}}$ and $\hat{\tilde{\mathbf{h}}}$:

$$\begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix} = \left(\hat{\alpha} \mathbf{I} + \begin{bmatrix} \mathbf{Q}_1(\mathbf{K}_{E1} + \mathbf{K}_{Y1}) \\ \mathbf{Q}_2(\mathbf{K}_{E2} + \mathbf{K}_{Y2}) \\ \vdots \\ \mathbf{Q}_N(\mathbf{K}_{EN} + \mathbf{K}_{YN}) \end{bmatrix} \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \vdots \\ \mathbf{P}_N \end{bmatrix} \right) \begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix} \quad (67)$$

By focusing on this we obtain the projection matrix connecting $\hat{\mathbf{h}}$ and $\hat{\tilde{\mathbf{h}}}$, and observe that our previous inversion problem involving a large matrix (size $|\mathfrak{D}(\mathcal{M})| \times |\mathfrak{D}(\mathcal{M})|$) is reduced to the inversion problem of a $|\mathfrak{Q}(\mathcal{M})| \times |\mathfrak{Q}(\mathcal{M})|$ matrix. Thus, a much more efficient correction algorithm is defined when considering the triangular Volterra kernels:

$$\begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix} = \left(\hat{\alpha} \mathbf{I} + \begin{bmatrix} \hat{\mathbf{K}}_{E1} \\ \hat{\mathbf{K}}_{E2} \\ \vdots \\ \hat{\mathbf{K}}_{EN} \end{bmatrix} + \begin{bmatrix} \hat{\mathbf{K}}_{Y1} \\ \hat{\mathbf{K}}_{Y2} \\ \vdots \\ \hat{\mathbf{K}}_{YN} \end{bmatrix} \right) \begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix} \quad (68)$$

which we write in the more compact form as $\hat{\tilde{\mathbf{h}}} = (\hat{\alpha} \mathbf{I} + \hat{\mathbf{K}}_E + \hat{\mathbf{K}}_Y) \hat{\mathbf{h}}$

Matrix $\hat{\mathbf{K}}_E$ results from the compactation and triangularization of Eq. (54); stacking the following rows:

$$\hat{\mathbf{K}}_{Er} = [\epsilon_{r1} \mathbf{p}_r \mathbf{1}_{|\mathcal{L}_1}^\top; \epsilon_{r2} \mathbf{p}_r \mathbf{1}_{|\mathcal{L}_2}^\top; \dots; \epsilon_{rN} \mathbf{p}_r \mathbf{1}_{|\mathcal{L}_N}^\top] \quad (69)$$

constant values ϵ_{rx} are computed according to Eq. (60), while vectors \mathbf{p}_r encode the number of sequences resulting from the r -permutations with repetition generated from a sequence \mathbf{q}_ℓ that is indexed by $\ell \in \mathcal{L}_r$:

$$\{\exists! \mathbf{p}_r | p_\ell = |\mathfrak{G}(\mathbf{q}_\ell)|, \ell \in \mathcal{L}_r \wedge \mathbf{q}_\ell \in \mathfrak{Q}_r(\mathcal{M})\} \quad (70)$$

On the other hand, to derive the general form for $\hat{\mathbf{K}}_{Yr}$ we look back to the compacted definition of $\hat{\Gamma}_t^{\Delta, xz}$, the elements of which are defined in Eq. (61):

$$\hat{\mathbf{K}}_{Yr} = \begin{bmatrix} \mathbf{0} & \alpha^{r+1} \epsilon \sum_{\substack{\forall t \in \\ \mathfrak{B}_r(\mathcal{L}_{r+1})}} \hat{\Gamma}_t^{r, r+1} & \alpha^{r+2} \epsilon^2 \sum_{\substack{\forall t \in \\ \mathfrak{B}_r(\mathcal{L}_{r+2})}} \hat{\Gamma}_t^{r, r+2} & \dots & \alpha^N \epsilon^{N-r} \sum_{\substack{\forall t \in \\ \mathfrak{B}_r(\mathcal{L}_N)}} \hat{\Gamma}_t^{r, N} \end{bmatrix} \quad (71)$$

Partitioning matrix $\hat{\mathbf{K}}_{Yr}$, which had been obtained by stacking together the N -orders defined in Eq. (71), into $N \times N$ submatrices, each of these corresponds to

$$\hat{\mathbf{K}}_{Yr,s} = \begin{cases} \mathbf{0}_{\mathcal{I}_r, \mathcal{I}_s} & \text{if } r \geq s \\ \alpha^s \epsilon^{s-r} \sum_{\substack{\forall t \in \\ \mathfrak{B}_r(\mathcal{L}_s)}} \hat{\Gamma}_t^{rs} & \text{otherwise} \end{cases} \quad (72)$$

In efficiently solving the inversion problem we look for a set of coefficients $\hat{\mathbf{h}}$ that minimize the difference $|\hat{\mathbf{h}} - \hat{\tilde{\mathbf{h}}}|$, without the need to invert the projection matrix in Eq. (68). The proposed correction algorithm involves two steps.

Step 1: Subtraction of aberrations due to $\hat{\mathbf{K}}_E$. First, we estimate the average values for the real coefficients estimated from the biased ones. We know from Volterra kernels symmetry and the properties of the compression matrix \mathbf{Q}_r that $\mathbf{1}^\top \hat{\mathbf{h}}_r = \mathbf{1}^\top \mathbf{h}_r$. Hence, from Eq. (68) one may estimate the average values for the true Volterra coefficients based on

$$\begin{bmatrix} \mathbf{1}_{|\mathcal{L}_1}^\top \hat{\mathbf{h}}_1 \\ \mathbf{1}_{|\mathcal{L}_2}^\top \hat{\mathbf{h}}_2 \\ \vdots \\ \mathbf{1}_{|\mathcal{L}_N}^\top \hat{\mathbf{h}}_N \end{bmatrix} = \left(\begin{bmatrix} \mathbf{1}_{|\mathcal{L}_1}^\top \hat{\mathbf{K}}_{E1} \\ \mathbf{1}_{|\mathcal{L}_2}^\top \hat{\mathbf{K}}_{E2} \\ \vdots \\ \mathbf{1}_{|\mathcal{L}_N}^\top \hat{\mathbf{K}}_{EN} \end{bmatrix} + \begin{bmatrix} \mathbf{1}_{|\mathcal{L}_1}^\top (\alpha \mathbf{1}_{\mathcal{I}_1} + \hat{\mathbf{K}}_{Y1}) \\ \mathbf{1}_{|\mathcal{L}_2}^\top (\alpha^2 \mathbf{1}_{\mathcal{I}_2} + \hat{\mathbf{K}}_{Y2}) \\ \vdots \\ \mathbf{1}_{|\mathcal{L}_N}^\top (\alpha^N \mathbf{1}_{\mathcal{I}_N} + \hat{\mathbf{K}}_{YN}) \end{bmatrix} \right) \begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix} \quad (73)$$

as follows:

$$\begin{aligned}
 \begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix} &= \begin{bmatrix} \mathbf{w} \\ \mathbf{v} \end{bmatrix} + \begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix} \\
 &= \begin{bmatrix} \epsilon_{11}|\mathcal{L}_1|\bar{\mathbf{p}}_1 & \epsilon_{12}|\mathcal{L}_2|\bar{\mathbf{p}}_1 & \dots & \epsilon_{1N}|\mathcal{L}_N|\bar{\mathbf{p}}_1 \\ \epsilon_{21}|\mathcal{L}_1|\bar{\mathbf{p}}_2 & \epsilon_{22}|\mathcal{L}_2|\bar{\mathbf{p}}_2 & & \\ \vdots & & \ddots & \vdots \\ \epsilon_{N1}|\mathcal{L}_1|\bar{\mathbf{p}}_N & & \dots & \epsilon_{NN}|\mathcal{L}_N|\bar{\mathbf{p}}_N \end{bmatrix} + \\
 &+ \begin{bmatrix} \alpha & \alpha^2 \epsilon \sum_{s=0}^1 |\mathcal{Q}_s(\mathcal{R}_M)| & \alpha^3 \epsilon^2 \sum_{s=0}^2 |\mathcal{Q}_s(\mathcal{R}_M)| & \dots & \alpha^N \epsilon^{N-1} \sum_{s=0}^{N-1} |\mathcal{Q}_s(\mathcal{R}_M)| \\ 0 & \alpha^2 & \alpha^3 \epsilon \sum_{s=0}^1 |\mathcal{Q}_s(\mathcal{R}_{M+1})| & \dots & \alpha^N \epsilon^{N-2} \sum_{s=0}^{N-2} |\mathcal{Q}_s(\mathcal{R}_{M+1})| \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & \alpha^N \end{bmatrix} \begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix}
 \end{aligned} \tag{74}$$

Denoting by $\bar{\mathbf{h}}$, $\hat{\mathbf{h}}$, and $\hat{\mathbf{h}}$ the N -long column vectors containing the average values for the true, biased and the computed estimates for the true Volterra kernels, respectively, the following approximation may be used, involving the inversion of a single $N \times N$ matrix:

$$\bar{\hat{\mathbf{h}}} \simeq \hat{\mathbf{h}} = (\mathbf{V} + \mathbf{W})^{-1} \bar{\hat{\mathbf{h}}} \tag{75}$$

From this one may subtract the influence of $\hat{\mathbf{K}}_E$ from Eq. (68), bearing in mind Eq. (69), and having the estimated average values for the true Volterra kernels obtained from Eq. (75):

$$\mathbf{b} = \hat{\mathbf{h}} - \left(\mathbf{W} \begin{bmatrix} \frac{1}{\bar{\mathbf{p}}_{\mathcal{L}_1}} \hat{\mathbf{h}}_1 & \frac{1}{\bar{\mathbf{p}}_{\mathcal{L}_2}} \hat{\mathbf{h}}_2 & \dots & \frac{1}{\bar{\mathbf{p}}_{\mathcal{L}_N}} \hat{\mathbf{h}}_N \end{bmatrix} \right) \otimes [\mathbf{p}_1^\top | \mathbf{p}_2^\top | \dots | \mathbf{p}_N^\top]^\top = \left(\alpha \mathbf{I} + \hat{\mathbf{K}}_Y \right) \hat{\mathbf{h}} \tag{76}$$

Step 2: Subtraction of aberrations due to $\hat{\mathbf{K}}_Y$. Considering the expression in Eq. (68), only the influence of $\hat{\mathbf{K}}_Y$ is left. Matrix $\alpha \mathbf{I} + \hat{\mathbf{K}}_Y$ is known to be upper-triangular, with elements on the diagonal being weighted by powers of α . The N -order Volterra kernel is associated with the elements of \mathbf{b} indexed by \mathcal{I}_N .

Hence, \mathbf{b} can be expressed as a function of the expected estimation result as

$$\mathbf{b} = \begin{bmatrix} \alpha \mathbf{I}_{|\mathcal{I}_1|} & \hat{\mathbf{K}}_{Y1,2} & \hat{\mathbf{K}}_{Y1,3} & \dots & \hat{\mathbf{K}}_{Y1,N} \\ \mathbf{0}_{|\mathcal{I}_2|,|\mathcal{I}_1|} & \alpha^2 \mathbf{I}_{|\mathcal{I}_2|} & \hat{\mathbf{K}}_{Y2,3} & \dots & \hat{\mathbf{K}}_{Y2,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{|\mathcal{I}_N|,|\mathcal{I}_1|} & \mathbf{0}_{|\mathcal{I}_N|,|\mathcal{I}_2|} & \dots & \dots & \alpha^N \mathbf{I}_{|\mathcal{I}_N|} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{h}}_1 \\ \hat{\mathbf{h}}_2 \\ \vdots \\ \hat{\mathbf{h}}_N \end{bmatrix} \tag{77}$$

Considering the inner triangular structure of this matrix, the solution is obtained through recurrent pivoting (starting from the lower-right submatrix, corresponding to order N) and substitution, moving towards the upper rows—lower orders.

Correction algorithm summary: Consequently, one may compute estimates of the Volterra kernels based on the proposed algorithm, despite deviations introduced on the system excitation that are amenable to the combination of a DC-offset and an arbitrary gain, attending to the following algorithm.

Algorithm 1. Volterra kernels correction.

$$\begin{aligned}
 &\hat{\mathbf{h}} \leftarrow \mathbf{Q} \hat{\mathbf{h}} && \triangleright \text{Triangularization of each order based on Eq. (20), stack according to Eq. (18)} \\
 &\hat{\mathbf{h}} \leftarrow (\mathbf{V} + \mathbf{W})^{-1} \hat{\mathbf{h}} && \triangleright N \times N\text{-matrix inversion: good condition number} \\
 &\mathbf{b} \leftarrow \hat{\mathbf{h}} - \hat{\mathbf{K}}_E \hat{\mathbf{h}} && \triangleright \text{From kernel's estimated average values and Eq. (76)} \\
 &\hat{\mathbf{h}}_N \leftarrow \frac{1}{\alpha^N} \mathbf{b}_N \\
 &\text{for } n \leftarrow N, 2 \text{ do} \\
 &\quad \mathbf{b}_{n-1} \leftarrow \mathbf{b}_{n-1} - [\hat{\mathbf{K}}_{Y1,n} | \hat{\mathbf{K}}_{Y2,n} | \dots | \hat{\mathbf{K}}_{Yn-1,n}]^\top \hat{\mathbf{h}}_n && \triangleright \hat{\mathbf{K}}_{Yr,n} \text{ computed row-wise if } M \text{ or } N \text{ demand it} \\
 &\quad \hat{\mathbf{h}}_{n-1} \leftarrow \frac{1}{\alpha^{n-1}} \mathbf{b}_{n-1} \\
 &\text{end for}
 \end{aligned}$$

In spite of our considerations on the correction problem and the algorithm proposed, one may choose to compute the true Volterra coefficients directly from Eq. (46). Such decision would imply that a $|\mathfrak{Q}(\mathcal{M})| \times |\mathfrak{Q}(\mathcal{M})|$ -matrix needs to be inverted. In most cases this is an intractable problem for most real applications, unless the system order and memory length are exceedingly short and uninteresting.

The proposed solution only requires the inversion of an $N \times N$ -matrix, and to generate the appropriate matrices $\overset{\Delta}{\mathbf{K}}_{\gamma,s,r}$. These matrices are not too complex and can be easily computed through a series of simple, logical operations.

5. Evaluation of the correction algorithm performance

Multiple examples may be found among nonlinear systems involved in real signal processing applications, which need to be identified. In particular, the number of those which require nonzero average inputs may seem small at first, but include some very representative examples. In [23] authors list a number of these, including AM (amplitude modulation) non-coherent demodulators and piezoelectric filters; all displaying some kind of asymmetry in their input voltage range.

To illustrate our analysis we now consider an example of a digital, nonlinear system matching Wiener's model. The selected parameters for this model highlight the effect of orthogonality loss due to DC-level in the input excitation, and allow us to assess the performance of the correction algorithm within a controlled scenario.

While considering the identification of a digital, nonlinear, Wiener system, one may rely on (fully or quasi) zero-mean MLS excitations as proposed in [11]; and obtain good estimates of the parameters in Wiener's modeling of the system. However, as we mathematically proved in Section 2.2, the results of the identification procedure dramatically change when the excitations are non-zero mean—e.g. DC-offset required to polarize the circuit.

Fig. 3 displays the block diagram corresponding to the digital, nonlinear, Wiener system used in this example. The system exhibits a multi-branch construction, involving linear transformations as delays (i.e. z^{-10}) and level attenuations (i.e. -3 dB), affecting the responses of a set of three linear subsystems (i.e. $c_1[m]$, $c_2[m]$ and $c_3[m]$). For our example, the latter are low-pass ($c_1[m]$) and band-stop ($c_2[m]$ and $c_3[m]$), finite impulse response (FIR) linear filters, the time (a) and frequency (b) forms of which are depicted in Fig. 4. Their outputs are then combined (added up), only after traversing a set of second-order, memoryless nonlinearities, to form the overall system output.

By simple inspection of Fig. 3 and comparing it to the general form of the Wiener model in Fig. 1, one may observe that our example matches a third-order Wiener construction, regardless of the individual impulse responses of the linear subsystems. Thus, we may test the well-known MLS-based identification scheme; and to do so we shall consider a nonzero average, MLS-based excitation and check the identification system performance.

We start with a binary, MLS sequence 63 samples long (shift register order 6, i.e. the impulse responses of the linear filters are assumed hereafter to be shorter than 63 samples) and binary levels -0.1111 and $+0.1389$, which had been adjusted based on the sequence length. From this we generate the discrete filter bank impulse responses matrix \mathbf{M} —see Eq. (33)—and prove matrix \mathbf{X} that is used to excite the system—see Eq. (38). To generate the normalized, biased input signal the values for the alpha-normalized DC-offset was set to $\varepsilon = 0.3333$ and the gain to $\alpha = 13.3333$. The input average value for the MLS sequence was 0.0159, which along with the DC-offset level ε produces (i) a nonzero average prove and (ii) a set of quasi-orthogonal filter bank responses (no influence of the input DC-offset; only affected by the nonzero MLS sequence average value), the outputs of which for the biased prove excitation are not orthogonal.

The results of the identification and correction algorithms for the example are displayed in Fig. 5. One may realize on these the existence of an underlying connection between the biased and corrected results, just as described by Eq. (66) for their respective symmetric forms.

Bearing in mind the correction order imposed by the proposed algorithm, we first correct the third order kernel. In Figs. 5(f) and (g) the values for \hat{h}_3 and \hat{h}_3 that are close to their respective modes are not displayed for ease of representation. According to Eq. (66) the only differences between the two are an additive constant value and a larger excursion for \hat{h}_3 due to the system gain (see Eq. (77)). This is perhaps more evident as we focus on the diagonals of these third order kernels \hat{h}_3 and \hat{h}_3 (see Fig. 5 (c)). The delay introduced on one multiplicative branches with respect to the other two (see the upper branch in the third-order subsystem of the block diagram in Fig. 3) causes most of the kernel information to lay outside the main diagonal.

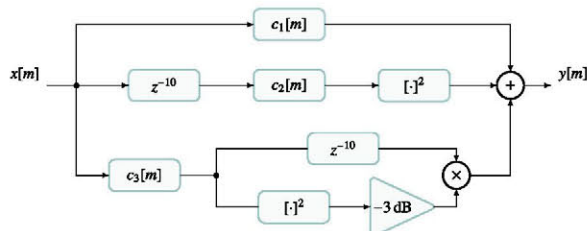


Fig. 3. Block diagram corresponding to the digital, Wiener system used in the example.

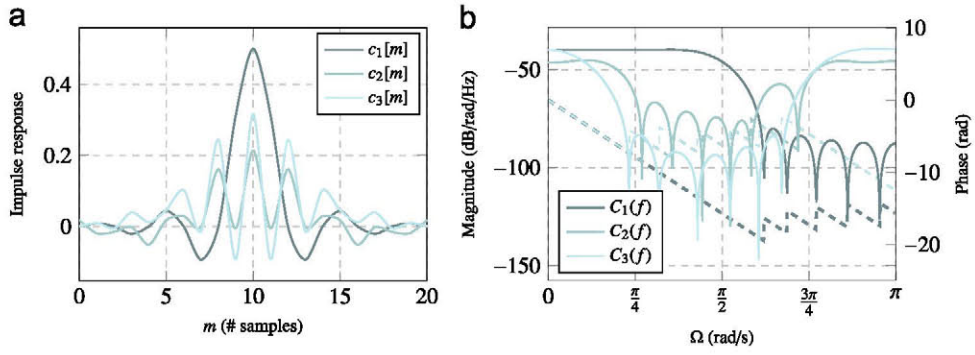


Fig. 4. Time and frequency responses of the three FIR linear subsystems $c_1[m]$, $c_2[m]$ and $c_3[m]$, appearing in the model depicted in Fig. 3.

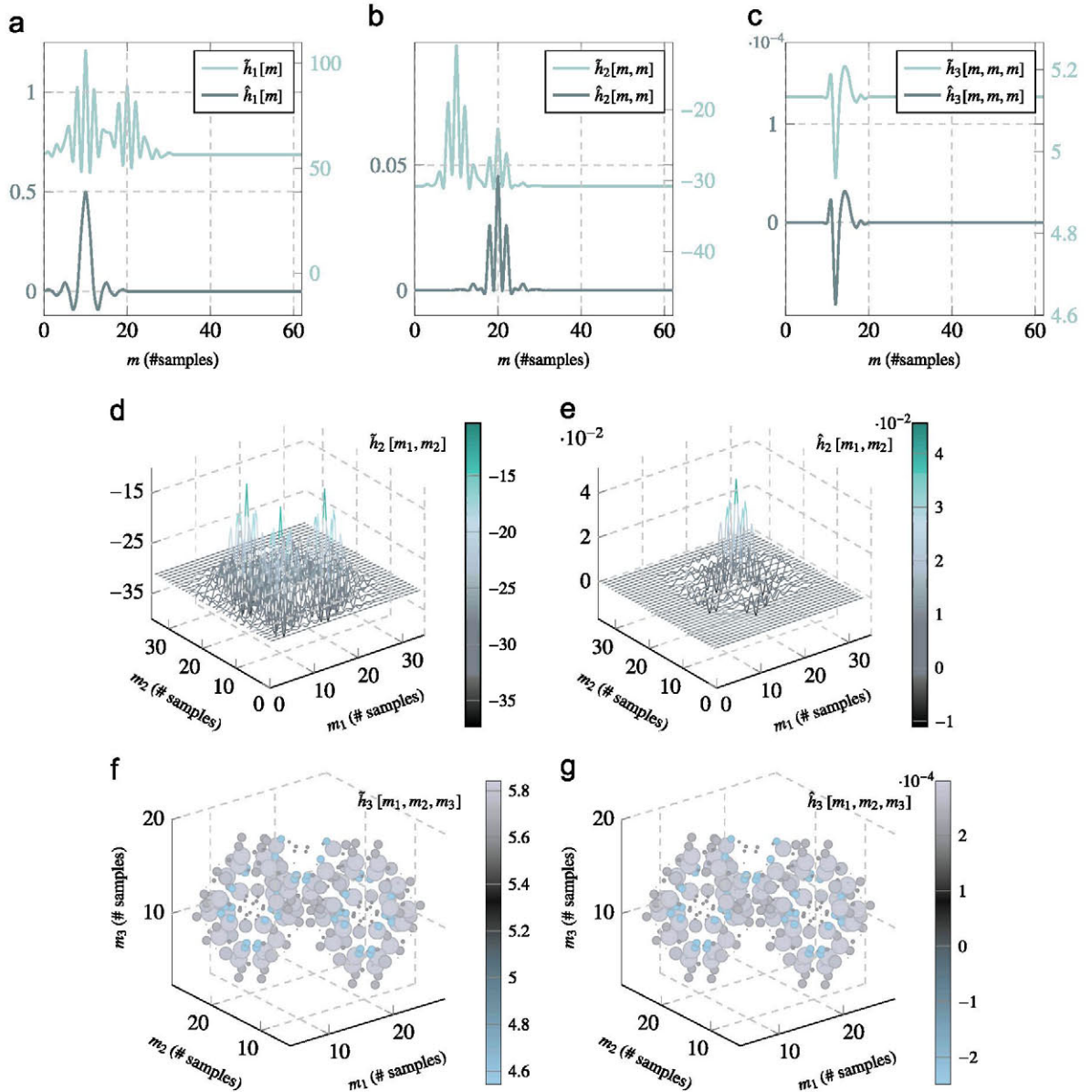


Fig. 5. Estimates of the symmetric, Volterra kernels, biased and corrected: (a) first order, (b) second order diagonal, (c) third order diagonal. The biased and corrected 2D and 3D kernels are displayed in (d), (e) second order; (f), (g) third order—only values outside the mode are plotted.

Results for the second-order kernel are depicted in Fig. 5(d), (e) and (b). Prior to correction, the biased kernel is affected by a constant value and the projection of the third order kernel onto the second order (see the projection matrix in the right bottom corner of Fig. 6). This is perhaps more evident on the kernel diagonal (see Fig. 5(b)). Noticeably, $\hat{h}_2[m, m]$ contains information from $c_3[m]$. Its contribution is different from that of $c_2[m]$ due to the delay introduced in the third-order subsystem compared to the second-order one.

Finally, on the first-order kernel (see Fig. 5(a)) one may observe the deviations introduced by the DC-offset and the second- and third-orders, explained by Eq. (66) (projection matrices are depicted in Fig. 6: top left and right for second and third orders, respectively); and the result of the correction algorithm. The resulting estimate of kernel $\hat{h}_1[m]$ is very similar to $c_1[m]$, the impulse response of the linear subsystem in our example, just as expected.

To evaluate the performance of the correction algorithm combined with the MLS-based identification procedure, we excited the system with a 10,000 samples long exponential, swept sine covering the frequency range $\Omega_s/10$ rad/s to $2\Omega_s/5$ rad/s, and used this same signal to excite the identified Wiener's system, with and without kernels' correction. Observing Fig. 7(a) one may find a negligible reconstruction error on the corrected kernels. The individual contributions of the different orders to the reconstructed signal are depicted in Fig. 7(b).

The need for the proposed correction algorithm and genuineness of our solution is self-evident attending to Fig. 7(c). Comparing the system output (on the x -axis) to its reconstructions (y -axes) on the biased ($\hat{y}[m]$) and corrected ($\check{y}[m]$) kernels, we observe (i) nonlinear relations for the biased kernel reconstruction, and (ii) an accurate reconstruction for the corrected kernels producing a twin output—i.e. driving a quasi-identical linear mapping.

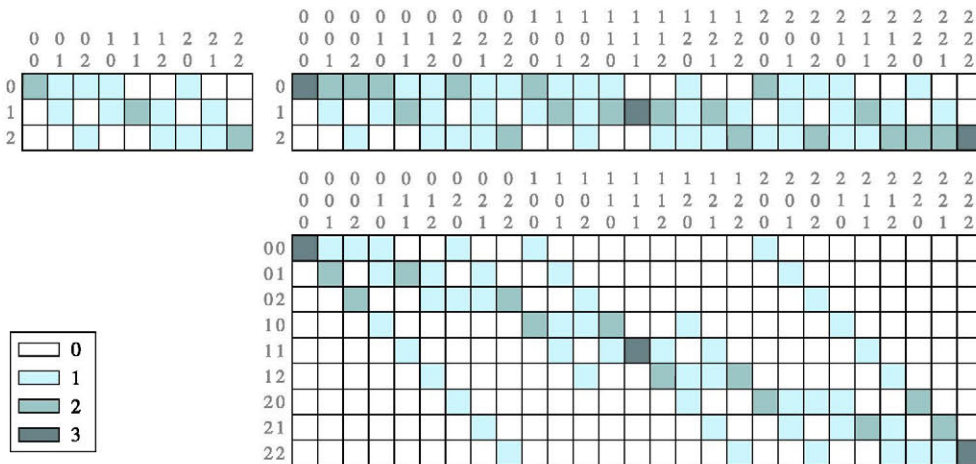


Fig. 6. Computed values for matrices $K_{\gamma_{1,2}}$ (top left), $K_{\gamma_{1,3}}$ (top right) and $K_{\gamma_{2,3}}$ (bottom) causing the order-dependent aberrations.

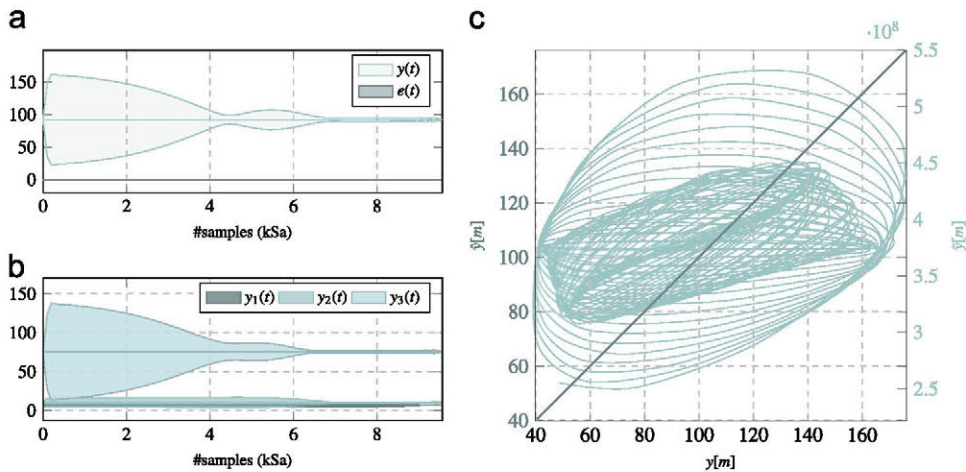


Fig. 7. Low time resolution envelopes for the system output reconstructions. (a) Reconstruction of the system output obtained on the corrected kernels, and reconstruction error. (b) Individual contributions of the different orders to the reconstructed signal. (c) Output to reconstructed-output graphs for the biased and corrected kernels.

6. Discussion

The results obtained on the example evidence that model parameters estimated without taking into consideration the presence of DC-offset are of little use on their own. The reconstruction of the system output based on these is noticeably different from the expected, matching the premise of this work.

When the correction is introduced, the diagonals of the triangular, Volterra kernels become very similar to the linear subsystems' impulse responses. An effect for the structure of the system in our example marks a successful identification. The results on the reconstruction of the system output on the corrected kernels reinforce this idea.

When performing the correction, in addition to the estimated, biased, triangular, Volterra kernels, and the assumptions introduced for the identification of the nonlinear system, only the values for the input's DC-offset and gain are required. The matrices required for these calculations may all be precomputed in advance, provided that the order of the nonlinearity had been wisely identified, and that the values of the gain and DC-offset are accurately established.

7. Conclusions

Probe's average value and dynamic range have a dramatic influence on MLS-based identification, causing orthogonality loss and invalidating the identification method premises. These produce systematic biases on the estimated Volterra kernels, which we have formulated in compact, matrix form.

We have developed a correction algorithm for MLS-based system identification under such excitation constraints. This is suitable for highly nonlinear, finite-order, invariant systems and may also be used on nonlinear system's representations based on Wiener's descriptors.

The resulting corrected kernels provide an accurate representation of the system, regardless of the probe input power and mean. The fact that the correction algorithm is suitable for non-parametric modeling suggests that the corrected identification results may be used for reliably inversion or correction of non-linear effects under constrained excitation conditions.

We demonstrate the accuracy of the proposed method through a comparative analysis of the results of MLS-based identification on a digital, Wiener system, given adequate estimations of the system's order and memory. The correction algorithm is remarkably less demanding than the identification procedure. For this reason, and the galling presence of non-zero average in MLS sequences, the proposed correction algorithm should always be used as a complement to the standard identification method.

Acknowledgments

This work is funded by the Spanish Ministry of Economy and Competitiveness as part of the TEC2012-38402-C04-1 Project.

References

- [1] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*, Wiley, New York, 1980.
- [2] M. Schetzen, *Orthogonal Development of the Wiener α -Functionals*, Wiley, New York, 1980, p. xvi, 531 pp. (Chapter 7).
- [3] S. Boyd, L.O. Chua, C.A. Desoer, Analytical foundations of Volterra series, *IMA J. Math. Control Inf.* 1 (3) (1984) 243–282, <http://dx.doi.org/10.1093/imamci/1.3.243>.
- [4] M. Tyril, J.A. Pedersen, P. Rubak, Identification of discrete Volterra series solving for one kernel at a time, in: *Audio Engineering Society Convention*, vol. 108, 2000, pp. 1–12.
- [5] D. Havelock, S. Kuwano, M. Vorländer, *Handbook of Signal Processing in Acoustics*, vol. 1, Springer, New York, 2008.
- [6] R. Nowak, B. Van Veen, Random and pseudorandom inputs for Volterra filter identification, *IEEE Trans. Signal Process.* 42 (8) (1994) 2124–2135, <http://dx.doi.org/10.1109/78.301847>.
- [7] C.-H. Cheng, E. Powers, Optimal Volterra kernel estimation algorithms for a nonlinear communication system for PSK and QAM inputs, *IEEE Trans. Signal Process.* 49 (1) (2001) 147–163, <http://dx.doi.org/10.1109/78.890357>.
- [8] R.D. Nowak, B.D.V. Veen, Efficient methods for identification of Volterra filter models, *Signal Process.* 38 (3) (1994) 417–428, [http://dx.doi.org/10.1016/0165-1684\(94\)90157-0](http://dx.doi.org/10.1016/0165-1684(94)90157-0).
- [9] J.-M.L. Caillec, Spectral inversion of second order Volterra models based on the blind identification of Wiener models, *Signal Process.* 91 (11) (2011) 2541–2555, <http://dx.doi.org/10.1016/j.sigpro.2011.05.007>.
- [10] W. Rugh, *Nonlinear system theory: the Volterra/Wiener approach*, *Johns Hopkins Series in Information Sciences and Systems*, Johns Hopkins University Press, Baltimore, Maryland, 1981.
- [11] M. Reed, M. Hawksford, Identification of discrete Volterra series using maximum length sequences, *IEE Proc.—Circuits Dev. Syst.* 143 (5) (1996) 241–248, <http://dx.doi.org/10.1049/ip-cds:19960726>.
- [12] M.J. Dunn, Chris, Hawksford, Distortion immunity of MLS-derived impulse response measurements, *J. Audio Eng. Soc.* 41 (5) (1993) 314–335.
- [13] G. Golub, C. Van Loan, *Matrix Computations*, *Johns Hopkins Studies in the Mathematical Sciences*, Johns Hopkins University Press, Baltimore, Maryland, 1996.
- [14] L. Comtet, *Advanced Combinatorics: The Art of Finite and Infinite Expansions*, D. Reidel Publishing Company, Dordrecht, Boston, 1974.
- [15] C. Crespo-Cadenas, P. Aguilera-Bonet, J. Becerra-Gonzalez, S. Cruces, On nonlinear amplifier modeling and identification using baseband Volterra-Parafac models, *Signal Process.* 96 (Part B) (2014) 401–405, <http://dx.doi.org/10.1016/j.sigpro.2013.09.028>.
- [16] J. Brewer, Kronecker products and matrix calculus in system theory, *IEEE Trans. Circuits Syst.* 25 (9) (1978) 772–781, <http://dx.doi.org/10.1109/TCS.1978.1084534>.
- [17] S. Golomb, G. Gong, *Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar*, Cambridge University Press, New York, 2005.

- [18] M. Cohn, A. Lempel, On fast M-sequence transforms (Corresp.), *IEEE Trans. Inf. Theory* 23 (1) (1977) 135–137, <http://dx.doi.org/10.1109/TIT.1977.1055666>.
- [19] M. Schroeder, Spectral properties of Galois sequences, in: *Number Theory in Science and Communication: With Applications in Cryptography, Physics, Digital Information, Computing and Self-similarity*, 5th ed., Springer Series in Information Sciences, Springer, Berlin Heidelberg 1984, pp. 331–333, 335–336 (Chapter 28).
- [20] H. Luke, H. Schotten, Odd-perfect, almost binary correlation sequences, *IEEE Trans. Aerosp. Electron. Syst.* 31 (1) (1995) 495–498, <http://dx.doi.org/10.1109/7.366335>.
- [21] J. Vanderkooy, Aspects of MLS measuring systems, *J. Audio Eng. Soc.* 42 (4) (1994) 219–231.
- [22] R. Pintelon, J. Schoukens, Design of excitation signals, in: *System Identification: A Frequency Domain Approach*, 2nd ed., Wiley, Hoboken, NJ, USA, 2012, pp. 151–175 (Chapter 5).
- [23] Y. Shmaliy, Nonlinear time invariant systems, in: *Continuous-Time Systems, Signals and Communication Technology*, Springer, Dordrecht, The Netherlands, 2007, pp. 497–502 (Chapter 7).