Diagonal Kernel Point Estimation of nth-Order Discrete Volterra-Wiener Systems

Massimiliano Pirani

Dipartimento di Elettronica, Intelligenza artificiale e Telecomunicazioni, Università Politecnica delle Marche,
Via Brecce Bianche 12, 60131 Ancona, Italy
Email: m.pirani@deit.univpm.it

Simone Orcioni

Dipartimento di Elettronica, Intelligenza artificiale e Telecomunicazioni, Università Politecnica delle Marche,
Via Brecce Bianche 12, 60131 Ancona, Italy
Email: sim@deit.univpm.it

Claudio Turchetti

Dipartimento di Elettronica, Intelligenza artificiale e Telecomunicazioni, Università Politecnica delle Marche,
Via Brecce Bianche 12, 60131 Ancona, Italy
Email: turchetti@deit.univpm.it

The estimation of diagonal elements of a Wiener model kernel is a well-known problem. The new operators and notations proposed here aim at the implementation of efficient and accurate nonparametric algorithms for the identification of diagonal points. The formulas presented here allow a direct implementation of Wiener kernel identification up to the nth order. Their efficiency is demonstrated by simulations conducted on discrete Volterra systems up to fifth order.

Keywords and phrases: nonlinear system identification, Wiener kernels, Volterra filtering.

1. INTRODUCTION

Among the identification techniques based on input-output correlations, the one proposed by Lee and Schetzen [1] is the most widely adopted due to its versatility, even if more recent techniques and up-to-date insights on these arguments can be found in [2] and more references in [3]. The application of the Lee-Schetzen technique on discrete nonlinear systems is straightforward and also gains some validity advantages versus the continuous time version, as stated rigorously in [4] and in [5]. In [6], the authors describe some characteristic behaviors of the Lee-Schetzen method for discrete systems and propose practical suggestions on its use.

The estimation of diagonal elements of a Wiener model kernel is a well-known problem. Such problem can be found documented in [6, 7, 8]. It arises from the higher estimation error variance exhibited by the estimation process of the kernel points having at least two equal coordinates. In [6], some explanations for this phenomenon, which augments increasing the number of equal coordinates, are given. The original Lee-Schetzen identification technique was particularly subject to this kind of errors. Goussard et al., in [9], made a ma-

jor contribution to the solution of the diagonal point estimation problem, although their work contains explicit solutions and proofs only up to the third order.

Koukoulas and Kalouptsidis, in [10], using the results on the calculation of cumulants due to the work of Leonov and Shiryaev [11], proposed a proof of the nth-order case valid also for inputs drawn from nonwhite Gaussian distributions. In the white Gaussian input case, the general formulas in [10] can be shown to reduce to Goussard's method. Other formulas using cumulants to estimate Wiener kernels directly have been proposed in [12]. Unfortunately, no implementation problems or any simulated efficiency tests were considered in [10, 12] because they were not among the purposes of the authors.

In this paper, we propose alternative formulas for the identification of nth-order Wiener kernels in the case of white Gaussian inputs, which avoid the explicit use of cumulants and are a useful shortcut to the proof of Goussard's method for higher orders. Moreover, the proposed formulas constitute an efficient way for the automatic generation of algorithm code for every order kernel identification, whereas the writing of efficient computer code is a very difficult task

as the kernel order increases. Some results on implementation tests are supplied to show the efficiency of the proposed method.

2. THE LEE-SCHETZEN METHOD

The Volterra series constitute a model for systems which yield generalized Taylor series expansions [1]. Under appropriate system class requirements [1, 2, 4, 13, 14, 15, 16, 17, 18], the input/output relationship for a discrete-time causal time-invariant nonlinear system can be expressed as

$$y(n) = h_0 + \sum_{m=1}^{\infty} \sum_{\tau_1,\ldots,\tau_m \geq 0}^{\infty} h_m(\tau_1,\ldots,\tau_m) \prod_{j=1}^{m} x(n-\tau_j). \quad (1)$$

To enhance model convergence and to allow identification by Lee-Schetzen method, the series (1) must be rearranged in terms of nonhomogeneous G operators [1, 2]. An operator is said to be a Wiener G operator if it satisfies the following definitions and conditions [1, 2]:

$$G_p[k_{p(p)}, k_{p-1(p)},\ldots,k_{0(p)}; x(n)]$$
$$= k_{0(p)} + \sum_{r=1}^{p} \sum_{\tau_1=0}^{\infty} \cdots \sum_{\tau_r=0}^{\infty} k_{r(p)}(\tau_1,\ldots,\tau_r) \quad (2)$$
$$\times x(n-\tau_1)\cdots x(n-\tau_r),$$

$$E\{H_m[h_m; x(n)] G_r[k_{r(r)}, k_{r-1(r)},\ldots,k_{0(r)}; x(n)]\} = 0, \quad (3)$$

for $m < r$, $n = 0, 1, 2,\ldots$; $k_p \triangleq k_{p(p)}$ is the Wiener kernel of pth order; H_m is a homogeneous mth-order Volterra operator, defined as

$$H_m[h_m; x(n)]$$
$$= \sum_{\tau_1,\ldots,\tau_m=0}^{\infty} h_m(\tau_1,\ldots,\tau_m) x(n-\tau_1)\cdots x(n-\tau_m), \quad (4)$$

where $x(n)$ must be a zero-mean Gaussian white process (i.e., an independent identically distributed (i.i.d.) sequence from a zero-mean Gaussian distribution) with $E\{x(n)x(n+t)\} = A\delta(t)$, where $E\{\cdot\}$ is the statistical expectation operator, $\delta(t)$ is the unitary impulse sequence, and A is the second-order moment of the input x.

The Lee-Schetzen method for nondiagonal point estimation of a pth-order Wiener kernel is described by [1, 6]:

$$k_p(\sigma_1,\ldots,\sigma_p)$$
$$= \frac{1}{p!A^p} E\{y(n)x(n-\sigma_1)\cdots x(n-\sigma_p)\}. \quad (5)$$

For the diagonal point case, a more complicated form is needed to account for the lower-order kernel contributions. The exact expressions for the second- and third-order Wiener kernels are [1]

$$2!A^2 k_2(\sigma_1,\sigma_2) = E\{y(n)x(n-\sigma_1)x(n-\sigma_2)\} - Ak_0\delta_{\sigma_1\sigma_2},$$
$$3!A^3 k_3(\sigma_1,\sigma_2,\sigma_3) = E\{y(n)x(n-\sigma_1)x(n-\sigma_2)x(n-\sigma_3)\}$$
$$- A^2(k_1(\sigma_1)\delta_{\sigma_2\sigma_3} + k_1(\sigma_2)\delta_{\sigma_1\sigma_3} + k_1(\sigma_3)\delta_{\sigma_1\sigma_2}), \quad (6)$$

where $\delta_{\sigma_i\sigma_j} \triangleq \delta(\sigma_i - \sigma_j)$ is the unitary impulse sequence delayed by $\sigma_i - \sigma_j$. For higher orders, this kind of explicit expression becomes unwieldy, due to the great number of correction terms in the diagonal point case. To overcome this difficulty, Lee and Schetzen [1] proposed the general identification formula [1, 6]

$$k_p(\sigma_1,\ldots,\sigma_p) = \frac{1}{p!A^p} E\left\{ \left(y(n) - \sum_{m=0}^{p-1} G_m[k_m; x(n)] \right) \right.$$
$$\left. \times x(n-\sigma_1)\cdots x(n-\sigma_p) \right\}, \quad (7)$$

where $G_m[k_m; x(n)]$ is the mth G-functional of the white Gaussian input $x(n)$ [1, 2, 6]. Unfortunately, this way of proceeding results in poor performances of the identification algorithm. In a practical situation, the limitations due to the finite length of input sequences and the departure from ideal statistical properties bias the identification procedure. In the implementation of (7), the identification errors of every point of the lower-order identified kernels are summed up by the G_m operator and they all contribute to the output error. On the contrary, only pointwise lower-order kernel errors affect expressions like (6). Indeed, we found that the development of nth-order compact expressions of the form (6) leads to some implementation advantages, while the numerical results remain the same with respect to the method of Goussard et al. in [9] which featured a similar kind of improvement of the original Lee-Schetzen method.

3. EFFICIENT nTH-ORDER FORMULAS FOR THE IDENTIFICATION OF DIAGONAL POINTS

In the major literature concerning the identification of Volterra systems, the examples supplied often do not exceed the third order. This is due to the fact that the identification algorithms become very cumbersome for higher orders. To extend the identification algorithms to higher orders in an easy way, we introduced new notations and operators which permit to handle, in a short and recursive form, the complicated expressions involved by algorithm generation. Actually, a manual generation of the code may be a very tedious and difficult task still for relatively low-order problems.

3.1. Preliminaries

Let M be a set of m distinct naturals, $Q \subseteq M$, and $q = |Q|$ and $m = |M|$ the cardinalities of Q and M, respectively. If $\mathbf{P}(M)$ is the power set of M (i.e., the set of all subsets of M) and $\mathbf{M}$ is the set of all n-tuples of formal variables of integer values, a relationship between the elements of $\mathbf{P}(M)$ and $\mathbf{M}$ follows:

$$\boldsymbol{\sigma} : \mathbf{P}(M) \longrightarrow \mathbf{M}, \quad (8)$$

such that $\boldsymbol{\sigma}(Q) = (\sigma_{i_1},\ldots,\sigma_{i_q}) \in \mathbf{M}$, where $Q \subseteq \mathbf{P}(M)$, $i_j \in Q$, $j = 1,\ldots,q$.

Furthermore, it will come in handy to define $\overline{\boldsymbol{\sigma}}^M(Q) = \boldsymbol{\sigma}(M-Q)$ as the function $\boldsymbol{\sigma}$ applied to the complementary set of Q with respect to M. Also, it will be necessary to generalize the definition of a qth-order Wiener kernel in the following way:

$$k(\boldsymbol{\sigma}(Q)) \triangleq k_q(\sigma_{i_1}, \ldots, \sigma_{i_q}), \qquad (9)$$

with $Q \neq \varnothing$ and $k(\boldsymbol{\sigma}(\varnothing)) \triangleq k_0$, where k_0 is the Wiener zeroth-order kernel. Moreover, we define

$$D[x(n); \boldsymbol{\sigma}(Q)] = x(n - \sigma_{i_1}) \cdots x(n - \sigma_{i_q}), \qquad (10)$$

with $Q \neq \varnothing$, and $D[x(n); \boldsymbol{\sigma}(\varnothing)] = 1$.

We now give a definition analogous to that given by Schetzen for the homonym $\sum \prod$ operator in [1]. For our purposes, the operator $\sum \prod$ will be redefined as

$$\sum \prod \boldsymbol{\sigma}(Q) \triangleq A^{-q/2} \sum \prod {}_{\text{Lee-Schetzen}}$$
$$= A^{-q/2} E\{D[x(n); \boldsymbol{\sigma}(Q)]\}. \qquad (11)$$

In [1], Schetzen reported that when $x(n)$ is a stationary zero-mean jointly Gaussian random sequence $E\{D[x(n); \boldsymbol{\sigma}(Q)]\} = 1$, when $q = 0$, $E\{D[x(n); \boldsymbol{\sigma}(Q)]\} = 0$ for odd q and it is equal to the sum of products of factors $E\{x(n - \sigma_i)x(n - \sigma_j)\}$ with $i, j \in Q$ for even q, resulting from all completely distinct ways of partitioning the set $\{x(n - \sigma_h) : h \in Q\}$ into pairs. If $x(n)$ is white Gaussian, under the ergodicity hypothesis, it holds that $E\{x(n - \sigma_i)x(n - \sigma_j)\} = A\delta_{\sigma_i \sigma_j}$.

In particular, for $q = 0$, we have $\sum \prod(\varnothing) = 1$, and for $q = 2$ and $q = 4$, we have, respectively,

$$\sum \prod (\sigma_{i_1}, \sigma_{i_2}) \triangleq \frac{1}{A} E\{D[x(n); \boldsymbol{\sigma}(\{i_1, i_2\})]\} = \delta_{\sigma_{i_1} \sigma_{i_2}},$$

$$\sum \prod (\sigma_{i_1}, \sigma_{i_2}, \sigma_{i_3}, \sigma_{i_4}) \triangleq \frac{1}{A^2} E\{D[x(n); \boldsymbol{\sigma}(\{i_1, i_2, i_3, i_4\})]\}$$
$$= \delta_{\sigma_{i_1} \sigma_{i_2}} \delta_{\sigma_{i_3} \sigma_{i_4}} + \delta_{\sigma_{i_1} \sigma_{i_3}} \delta_{\sigma_{i_2} \sigma_{i_4}} + \delta_{\sigma_{i_1} \sigma_{i_4}} \delta_{\sigma_{i_2} \sigma_{i_3}}. \qquad (12)$$

A new operator $\sum_\Pi$ will now be introduced as

$$\sum_\Pi^M f(\boldsymbol{\sigma}(Q); \cdot) \triangleq \sum_{i=1}^r \left[f(\boldsymbol{\sigma}(Q_i); \cdot) \sum \prod \overline{\boldsymbol{\sigma}}^M(Q_i) \right], \qquad (13)$$

where $r = \binom{m}{q}$, $Q \subseteq M$, Q_i are all the subsets generated by the combinations of q elements chosen from M, and f is a symmetrical mapping with respect to $\boldsymbol{\sigma}(Q)$. In particular, it holds that

$$\sum_\Pi^M f(\boldsymbol{\sigma}(Q); \cdot) = \sum_\Pi^M f(\boldsymbol{\sigma}(Q_i); \cdot), \quad 1 \leq i \leq r, \qquad (14)$$

$$\sum_\Pi^M f(\boldsymbol{\sigma}(\varnothing); \cdot) = f(\boldsymbol{\sigma}(\varnothing); \cdot) \sum \prod \boldsymbol{\sigma}(M), \qquad (15)$$

$$\sum_\Pi^M f(\boldsymbol{\sigma}(M); \cdot) = f(\boldsymbol{\sigma}(M); \cdot). \qquad (16)$$

The properties (14), (15), and (16) are trivially verified using definitions (13) and (11).

3.2. Formulas for *m*th-order Wiener kernel estimation

From the above definitions, we have derived the following general formulas for the *m*th-order kernel estimates:

$$E\{y(n)D[x(n); \boldsymbol{\sigma}(M)]\}$$
$$= \sum_{h=0}^{\lfloor m/2 \rfloor} (m - 2h)! A^{m-h} \sum_\Pi^M k(\boldsymbol{\sigma}(\{i_1, \ldots, i_{m-2h}\})), \qquad (17)$$

from which

$$m! A^m k(\boldsymbol{\sigma}(M))$$
$$= E\{y(n)D[x(n); \boldsymbol{\sigma}(M)]\}$$
$$- \sum_{h=1}^{\lfloor m/2 \rfloor} (m - 2h)! A^{m-h} \sum_\Pi^M k(\boldsymbol{\sigma}(\{i_1, \ldots, i_{m-2h}\})), \qquad (18)$$

where $\lfloor (\cdot) \rfloor$ denotes the integer part of $(\cdot)$. The formulas just presented allow the *m*th-order Wiener kernel to be identified. Note that for $m = 2, 3$ they reduce to (6). In the diagonal points, the estimation will be improved with respect to the classical Schetzen technique referred to here by (7). It must be noted that a real improvement is obtained only when the expectations are assessed by averages on finite-length sequences, as it is unavoidable in practice. A proof for (18) can be found in Appendix A.

3.3. Explicit generalization of Goussard's method to *m*th order

As previously pointed out, an improvement in the estimation of diagonal elements was also obtained by Goussard et al. [9]. Although they proposed a method for the improvement of the diagonal points estimation, which is in principle analogous to the idea which resides behind the development of (17) and (18), in [9] they demonstrated only the expressions up to the third order. Actually, we aimed at the generalization of those formulas and proofs for higher orders in a compact and handy way.

It can be proved (see Appendix B) that the *m*th-order version of the original Goussard formulas assumes the following form:

$$m! A^m k(\boldsymbol{\sigma}(M)) = E\{y(n)\Psi[x(n); \boldsymbol{\sigma}(M)]\}, \qquad (19)$$

where the operator Ψ is defined as

$$\Psi[x(n); \boldsymbol{\sigma}(\varnothing)] = 1, \qquad (20)$$

$$\Psi[x(n); \boldsymbol{\sigma}(M)] = \sum_{h=0}^{\lfloor m/2 \rfloor} (-1)^h A^h \sum_\Pi^M D[x(n); \boldsymbol{\sigma}(\{i_1, \ldots, i_{m-2h}\})]. \qquad (21)$$

We also propose a recursive form of formula (21) which can be useful for generating the code which computes Ψ for higher orders:

$$\Psi[x(n); \boldsymbol{\sigma}(M)] = D[x(n); \boldsymbol{\sigma}(M)]$$
$$- \sum_{h=1}^{\lfloor m/2 \rfloor} A^h \sum_\Pi^M \Psi[x(n); \boldsymbol{\sigma}(\{i_1, \ldots, i_{m-2h}\})]. \qquad (22)$$

The preceding formula can also be given in a more compact implicit form:

$$D[x(n); \boldsymbol{\sigma}(M)] = \sum_{h=0}^{\lfloor m/2 \rfloor} A^h \sum_{\Pi}^{M} \Psi[x(n); \boldsymbol{\sigma}(\{i_1, \ldots, i_{m-2h}\})].$$

(23)

A proof for (19), (20), (21), (22), and (23) has been supplied in Appendix B.

Interesting higher-order formulas for identification in a nonparametric approach can also be found in [10] or referenced in [2], where they are based on cross-cumulants rather than crossmoments. These formulas generalize the identification method avoiding an explicit Wiener-to-Volterra series conversion and they hold also for nonwhite Gaussian inputs. If the input is white, they simplify in a form equivalent to (19). Actually, the use of (19) and (21) can be found to be equivalent to the formula using the cumulant definitions in the white Gaussian case.

In [12] and references therein, a useful formula can be found which directly relates Wiener kernels to cumulants. The use of (18), after some manipulations, is equivalent to the formulation proposed in [12]. The computation of the joint cumulants of mth-order requires, in principle, the knowledge of all the joint moments up to mth-order [2]. In (18), only the mth moment is needed because the lower-order moments are implicitly stored in lower-order previously estimated kernels. So the notations and formulas proposed here constitute mainly a handy tool for the straightforward implementation of cumulant calculus in the particular case of white Gaussian input. The implementation efficiency of (18) resides in the way the storing of lower-order moments is accomplished by accounting for similar terms generated by the symmetry properties of the lower-order moments (or cumulants themselves).

The main differences between the method related to (18) and the method of (19) and (21) reside in the application point of view: while the first needs the storage of lower-order kernels, the plain implementation of (19) permits to identify any kernel without knowing the others. This second technique obviously causes additional computation time in the complete estimation of a model, as will be shown in the next section.

The use of (18) gives also the most efficient way to access the lower-order moments needed by a smart implementation of (19) and (21). In [9], those general-order implementation issues were not covered.

4. IMPLEMENTATION TESTS

In the following, the results obtained by the implementation of (18) (which will be referred to as the *straight method*) are compared with the ones obtained by the formulas of Schetzen [1] (which will be referred to as the *classic method*[1]) and

TABLE 1: Mean values, over 100 independent systems, of percentage of kernel points which have an identification relative error less than threshold (these points are referred to as valid points[2]). Simulations with 1 input (left) and 10 inputs (right). Every input is a 10^5 sample sequence from a zero-mean independent white Gaussian distribution.

Kernel order	Classic	Straight	Off-diagonal
2nd order	86.10/93.20	86.10/93.20	87.09/94.56
3rd order	62.31/62.67	91.22/95.72	94.67/97.80
4th order	25.44/31.95	40.11/54.44	41.20/57.60
5th order	23.57/27.34	52.01/65.47	55.86/71.74

TABLE 2: Mean computation time (seconds) over 10 identifications of a test system versus model order, for each of the three methods implemented.

Methods	2nd	3rd	4th	5th
Classic	0.58	9.56	61.65	223.68
Goussard's	0.57	13.23	97.93	643.97
Straight	0.51	7.24	40.54	155.49

the ones by Goussard et al. [9] (referred to here as *Goussard's method*), which, in Section 3.3, have been extended to higher orders explicitly. The formulas have been tested identifying 100 discrete Volterra systems of the fifth order.

For a significant implementation test, we needed a quite general set of test systems. The most general Volterra discrete causal system could have been created drawing the values of the kernels from a Gaussian distribution. Here, for the sake of simplicity of the implementation and of the exposition, only the constituent FIR filter taps have been drawn from a Gaussian distribution (independent from the input sequences). Indeed, the nth-order kernel was constituted by the cascade of an FIR filter and an nth-power nonlinear block. The system memory length for each order results from the ten taps of the FIR kernel generators. Besides this restriction, we retain that the test so conducted still maintained enough generality.

It must be noted that the results coming from the *straight method* and Goussard's one differed only by round-off errors, so in Table 1 and Figure 1, only the results for the *straight method* will be reported but they hold for *Goussard's method* as well. Besides, the two methods differ considerably in computing times: Table 2 shows that the *straight method* is faster than *Goussard's method* (computation times are almost four times shorter for the fifth-order case). This happens because the *straight method* avoids some redundant computation of the moments of the input and output vectors by trading it for the storage of lower-order kernel values.

---

[1]The implementation of (7) has actually been done subtracting only the lower-order G-functionals which had the same parity with the order of the kernel being identified, as suggested by Schetzen in [1].

[2]The reported quantities are obtained by an average over 100 independent systems estimate of the quantity $100 \times N_{pv}/N_p$, where N_p is the number of necessary points (taking account of symmetries) for the estimation of k_p and N_{pv} is the number of the valid points defined as follows. Let $k_p(\overline{\tau}_1, \overline{\tau}_2, \ldots, \overline{\tau}_p)$ be a point of k_p and $\hat{k}_p(\overline{\tau}_1, \overline{\tau}_2, \ldots, \overline{\tau}_p)$ its estimate, then a point is considered valid if $|\hat{k}_p(\overline{\tau}_1, \overline{\tau}_2, \ldots, \overline{\tau}_p) - k_p(\overline{\tau}_1, \overline{\tau}_2, \ldots, \overline{\tau}_p)|/|k_p(\overline{\tau}_1, \overline{\tau}_2, \ldots, \overline{\tau}_p)| \leq 10$.
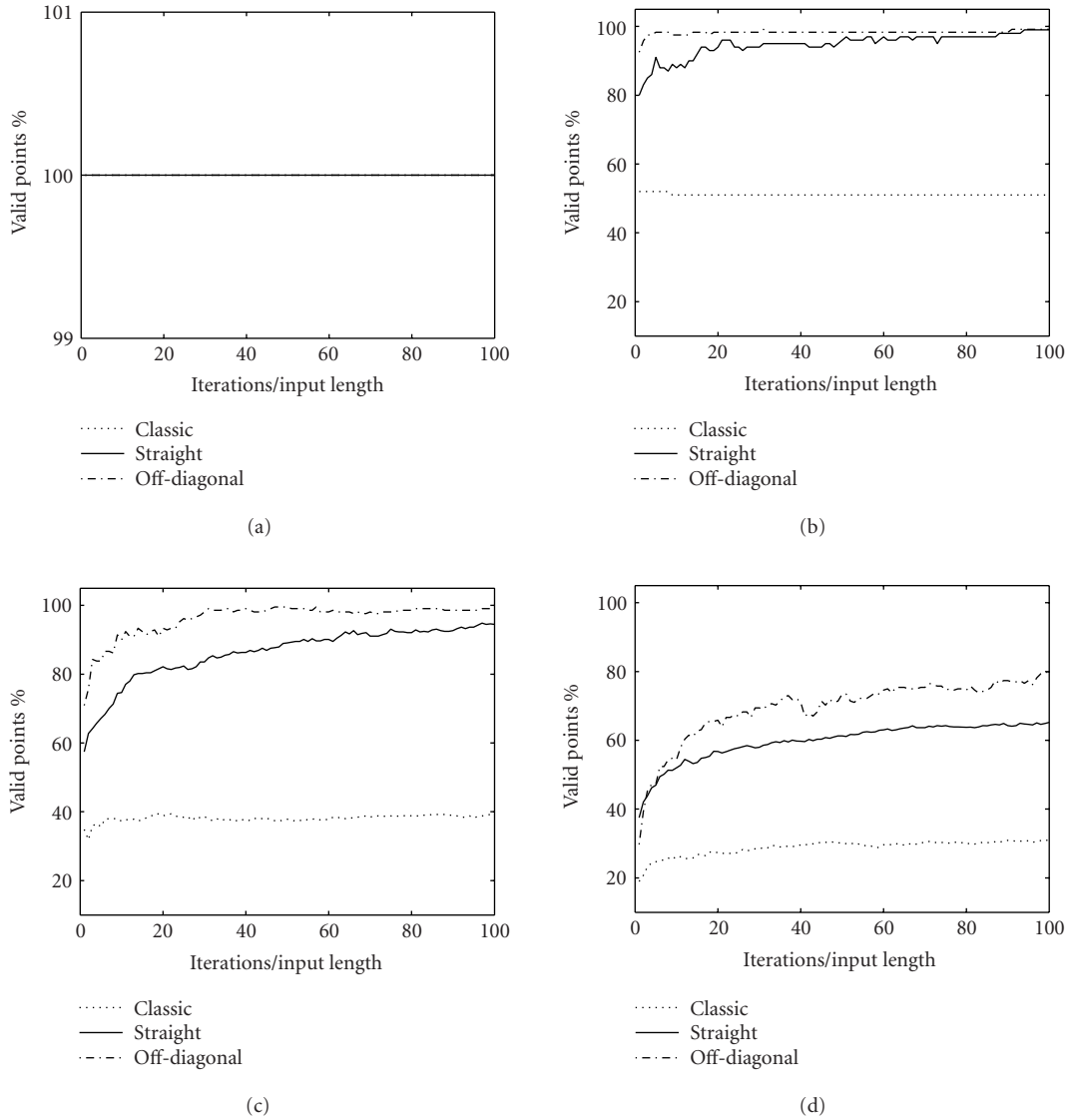
FIGURE 1: Percentage of valid points (see footnote a) versus number (or length) of input signals for only one of the test systems. An abscissa unit corresponds to 10^5 independent input samples. (a) 2nd order, (b) 3rd order, (c) 4th order, and (d) 5th order.

The first test for the estimation efficiency has been performed using ten white Gaussian inputs of 10^5 samples for each of the 100 systems. The results of this test are shown in Table 1. Each cell of the table reports two values: the first one refers to one input of 10^5 samples. The second one is the value obtained with a mean on ten kernel identifications, with ten independent input sequences of the same length. Under the assumption of the ergodicity of the identification process, this procedure corresponds to a single experiment with an input length ten times longer than the first one. When the value of the desired kernel is nearly zero, the relative error tends to infinity. As a consequence, we established an arbitrary threshold for the relative error value equal to 10. Only the points with a relative error under threshold are consid-

ered as valid points. Table 1 shows the percentage of valid points.

Results show, for all the methods, an improvement of identification accuracy as the input length increases. In the *classic method*, such improvement is less than in the *straight* one. This first test was a consistency test for the algorithms.

In a subsequent simulation, the percentage of acceptance of kernel points has been calculated increasing further the number of the input signals for only one of the test systems arbitrarily chosen. Figure 1 shows such results, evidencing how the *straight method* works better than the *classic* one. The off-diagonal estimates have been reported both in Table 1 and in Figure 1 as a reference, because they represent the best case, which is equivalent for all the methods considered so far.

5. CONCLUSION

The formulas proposed here with the use of well-suited notations permit to handle, in an efficient way, the nth-order identification of Wiener kernels. The proof of the formulas has been supplied and simulation has demonstrated their efficiency with respect to the classic Lee-Schetzen method. The alternate method proposed here and referred to as the *straight method* has been shown to be considerably faster than previous improvements of the Lee-Schetzen method known in literature [9], especially as the order and the size of the kernels increase.

APPENDICES

A. PROOF OF FORMULA (17)

We have to prove that when $x(n)$ is a sequence of white Gaussian random variables (an i.i.d. Gaussian process), it holds that

$$
\begin{aligned}
&E\{y(n)D[x(n); \boldsymbol{\sigma}(M)]\} \\
&\quad = \sum_{h=0}^{\lfloor m/2 \rfloor} (m-2h)! A^{m-h} \sum_{\Pi}^{M} k(\boldsymbol{\sigma}(\{i_1, \dots, i_{m-2h}\})),
\end{aligned}
\tag{A.1}
$$

with $M = \{1, 2, \dots, m\}$, $m \in \mathbb{N}$, $m < \infty$.

When the Wiener series expansion exists, we can write

$$
y(n) = \sum_{h=0}^{\infty} G_h[k_h; x(n)].
\tag{A.2}
$$

If we multiply the left and right members of (A.2) by $D[x(n); \boldsymbol{\sigma}(M)]$ and apply the expectation operator, then, exploiting the orthogonality of the G and D operators defined in (2) and (10) (it can be easily proved that D operators are a particular case of G operators [1, 9]), it holds that

$$
\begin{aligned}
&E\{y(n)D[x(n); \boldsymbol{\sigma}(M)]\} \\
&\quad = \sum_{h=0}^{m} E\{G_h[k_h; x(n)]D[x(n); \boldsymbol{\sigma}(M)]\}.
\end{aligned}
\tag{A.3}
$$

From the properties of the expectation operator and of the operators G and D, it follows that in the sum of (A.3) for even (odd) m, the terms with indices h odd (even) are identically zero, then (A.3) can be simplified as follows:

$$
\begin{aligned}
&E\{y(n)D[x(n); \boldsymbol{\sigma}(M)]\} \\
&\quad = \sum_{h=0}^{\lfloor m/2 \rfloor} E\{G_{m-2h}[k_{m-2h}; x(n)]D[x(n); \boldsymbol{\sigma}(M)]\}.
\end{aligned}
\tag{A.4}
$$

So (A.1) holds if the validity of the following can be verified:

$$
\begin{aligned}
&E\{G_{m-2h}[k_{m-2h}; x(n)]D[x(n); \boldsymbol{\sigma}(M)]\} \\
&\quad = (m-2h)! A^{m-h} \sum_{\Pi}^{M} k(\boldsymbol{\sigma}(\{i_1, \dots, i_{m-2h}\})),
\end{aligned}
\tag{A.5}
$$

for all $m, h \in \mathbb{N}$, $h \le \lfloor m/2 \rfloor$.

To prove (A.5), we have to consider the explicit general expression of a G operator in the discrete-time case [1, 2]:

$$
\begin{aligned}
G_p[k_p; x(n)] &= \sum_{s=0}^{\lfloor p/2 \rfloor} \sum_{\tau_1} \cdots \sum_{\tau_{p-2s}} \frac{(-1)^s p! A^s}{(p-2s)! s! 2^s} \\
&\quad \times \sum_{\xi_1} \cdots \sum_{\xi_s} k_p(\xi_1, \xi_1, \dots, \xi_s, \xi_s, \tau_1, \dots, \tau_{p-2s}) \\
&\quad \times x(n - \tau_1) \cdots x(n - \tau_{p-2s}).
\end{aligned}
\tag{A.6}
$$

Using (A.6) and the definition of the $\sum \prod$ operator and letting $p = m - 2h$, (A.5) can be simplified as

$$
\begin{aligned}
\sum_{s=0}^{\lfloor p/2 \rfloor} &C_s \sum_{\tau_1} \cdots \sum_{\tau_{p-2s}} \sum_{\xi_1} \cdots \sum_{\xi_s} k_p(\xi_1, \xi_1, \dots, \xi_s, \xi_s, \tau_1, \dots, \tau_{p-2s}) \\
&\times \sum \prod (\tau_1, \dots, \tau_{p-2s}, \sigma_1, \dots, \sigma_m) \\
&= \sum_{\Pi}^{M} k(\boldsymbol{\sigma}(\{i_1, \dots, i_p\})),
\end{aligned}
\tag{A.7}
$$

with

$$
C_s = \frac{(-1)^s}{(p-2s)! s! 2^s}.
\tag{A.8}
$$

Now consider the general expression of a term of the sum over s in the first member of (A.7):

$$
\begin{aligned}
C_s \sum_{\tau_1} \cdots &\sum_{\tau_{p-2s}} \sum_{\xi_1} \cdots \sum_{\xi_s} k_p(\xi_1, \xi_1, \dots, \xi_s, \xi_s, \tau_1, \dots, \tau_{p-2s}) \\
&\times \sum \prod (\tau_1, \dots, \tau_{p-2s}, \sigma_1, \dots, \sigma_m).
\end{aligned}
\tag{A.9}
$$

We will show hereafter how the terms deriving from the expansion of (A.9) can be grouped in $\lfloor p/2 \rfloor$ term typologies. We define as *ν-type term* the following expression:

$$
\begin{aligned}
C_s \sum_{\xi_1} \cdots &\sum_{\xi_\nu} k_p(\xi_1, \xi_1, \dots, \xi_\nu, \xi_\nu, \sigma_1, \dots, \sigma_{p-2\nu}) \\
&\times \sum \prod (\sigma_{p-2\nu+1}, \sigma_{p-2\nu+2}, \dots, \sigma_{m-1}, \sigma_m).
\end{aligned}
\tag{A.10}
$$

Note that in the expression (A.10), there are ν pairs of identical formal variables in the argument of k_p and a corresponding number ν of sum operators, which explains the choice of the *ν-type term* name. The term (A.10) collects a group of addenda of (A.9), as we are going to point out next.

Recalling that an expression like $E\{x(n - \sigma_i)x(n - \sigma_j)\}$ generates the sequence $\delta(\sigma_i - \sigma_j)$, consider one of the products of unity pulse δ-sequences deriving from the complete expansion of (A.9):

$$
\begin{aligned}
\delta_{\tau_1 \tau_2} \cdots &\delta_{\tau_{(2N-1)} \tau_{(2N)}} \delta_{\tau_{(2N+1)} \sigma_1} \cdots \delta_{\tau_{(p-2s)} \sigma_{(p-2s-2N)}} \\
&\times \delta_{\sigma_{(p-2s-2N+1)} \sigma_{(p-2s-2N+2)}} \cdots \delta_{\sigma_{m-1} \sigma_m},
\end{aligned}
\tag{A.11}
$$

with N chosen such that $0 \le N \le \lfloor (p-2s)/2 \rfloor$. The term (A.11) is compounded by two types of factors: we will dub *homogeneous* those factors of the form $\delta_{\tau_i \tau_j}$ and *mixed* those factors of the form $\delta_{\tau_i \sigma_j}$, for all $i, j \in \mathbb{N}$. Note that the product

between the first part of (A.9),

$$C_s \sum_{\tau_1} \cdots \sum_{\tau_{p-2s}} \sum_{\xi_1} \cdots \sum_{\xi_s} k_p(\xi_1, \xi_1, \ldots, \xi_s, \xi_s, \tau_1, \ldots, \tau_{p-2s}),$$

(A.12)

and a *homogeneous* factor $\delta_{\tau_i \tau_j}$ collapses the two sums $\sum_{\tau_i}$ and $\sum_{\tau_j}$ into one, and at the same time makes indistinguishable the arguments of k_p in the *i*th and *j*th positions. On the other hand, the product between (A.12) and a *mixed* factor $\delta_{\tau_i \sigma_j}$ cancels the sum $\sum_{\tau_i}$ and substitutes τ_i with σ_j in the *i*th position of the argument of k_p. Using the sifting property of the δ-sequences and letting $\xi_{s+i} = \tau_i$, from the product between (A.12) and (A.11), we obtain the following simplified expression of a group of addenda of (A.9):

$$C_s \sum_{\tau_1} \cdots \sum_{\tau_N} \sum_{\xi_1} \cdots$$
$$\times \sum_{\xi_s} k_p(\xi_1, \xi_1, \ldots, \xi_s, \xi_s, \tau_1, \tau_1, \ldots, \tau_N, \tau_N, \sigma_1, \ldots, \sigma_{p-2s-2N})$$
$$\times \delta_{\sigma_{(p-2s-2N+1)}\sigma_{(p-2s-2N+2)}} \cdots \delta_{\sigma_{m-1}\sigma_m}.$$

(A.13)

The expression (A.13) constitutes a part of a *ν-type term* like (A.10) with $\nu = s + N$, and it is easy to show that all the terms obtained from $\sum \prod (\tau_1, \ldots, \tau_{p-2s}, \sigma_1, \ldots, \sigma_m)$, having their first $p - 2s$ factors in common with the term (A.11), can be collected by the following expression:

$$\delta_{\tau_1 \tau_2} \cdots \delta_{\tau_{(2N-1)}\tau_{(2N)}} \delta_{\tau_{(2N+1)}\sigma_1} \cdots \delta_{\tau_{(p-2s)}\sigma_{(p-2s-2N)}}$$
$$\times \sum \prod (\sigma_{p-2s-2N+1}, \sigma_{p-2s-2N+2}, \ldots, \sigma_{m-1}\sigma_m).$$

(A.14)

Hence, the term of type $s + N$

$$C_s \sum_{\xi_1} \cdots \sum_{\xi_s} \sum_{\tau_1} \cdots$$
$$\times \sum_{\tau_N} k_p(\xi_1, \xi_1, \ldots, \xi_s, \xi_s, \tau_1, \tau_1, \ldots, \tau_N, \tau_N, \sigma_1, \ldots, \sigma_{p-2(s+N)})$$
$$\times \sum \prod (\sigma_{p-2s-2N+1}, \sigma_{p-2s-2N+2}, \ldots, \sigma_{m-1}\sigma_m),$$

(A.15)

collects all the addenda of the complete expansion of (A.9) which have in common the following $p - 2s$ factors:

$$\delta_{\tau_1 \tau_2} \cdots \delta_{\tau_{(2N-1)}\tau_{(2N)}} \delta_{\tau_{(2N+1)}\sigma_1} \cdots \delta_{\tau_{(p-2s)}\sigma_{(p-2s-2N)}}.$$

(A.16)

Now, it can be observed that in the expansion of (A.9), there are other terms of the same kind of (A.15) which have the expression

$$\sum \prod (\sigma_{p-2s-2N+1}, \sigma_{p-2s-2N+2}, \ldots, \sigma_{m-1}, \sigma_m)$$

(A.17)

in common, but the argument of k_p different for a permutation of the group of variables

$$(\tau_1, \tau_1, \ldots, \tau_N, \tau_N, \sigma_1, \ldots, \sigma_{p-2(s+N)}).$$

(A.18)

Using the symmetry hypothesis[3] on k_p, those terms become similar to (A.15). Hence we now aim at obtaining the coefficient to be multiplied by (A.15) which accounts for all those similar terms. This coefficient is actually the number of completely distinct permutations, in the sense of the definition of $\sum \prod$, among the $P = (p - 2s)!/2^N$ permutations of the group of variables (A.18) with N pairs of repeated elements. Indeed, note that a position exchange of the variable σ from *i*th to *j*th position in the argument of k_p corresponds to a distinct permutation. In fact, that position exchange derives from two distinct factor products of $\sum \prod (\cdot)$ which differ at least in the mixed factors $\delta_{\sigma \tau_i}$ and $\delta_{\sigma \tau_j}$. On the other hand, a position exchange between the variable pairs (τ_i, τ_i) and (τ_j, τ_j) corresponds to a change of the order of factors in product terms. The product terms, where the homogeneous factors $\delta_{\tau_i \tau_i}$ and $\delta_{\tau_j \tau_j}$ differ only in position, will be indistinguishable for $\sum \prod$. N being the number of pairs of the τ variables in the argument of k_p, for each of the allowed P permutations, there will be a group of $N!$ indistinguishable corresponding permutations. So, in the expansion of (A.9), the number of indistinguishable terms from the term (A.15), due to the symmetry of k_p, will be equal to

$$U_{s+N,s} = \frac{(p-2s)!}{N!2^N}.$$

(A.19)

The first subscript of U denotes the type of the term to which the coefficient is associated and the second is the index of the outer sum of (A.7).

Up to this point, we have focused our attention on the fact that, s, N, and the *n*-tuple $(\sigma_1, \sigma_2, \ldots, \sigma_{p-2(s+N)})$ being chosen, the term (A.15) is a representative of $U_{s+N,s}$ similar terms of (A.9). Now, we observe that, for the symmetry of k_p and the definition of $\sum \prod (\tau_1, \ldots, \sigma_m)$, we have $N_C = \binom{m}{p-2(s+N)}$ equivalence classes which have a term like (A.15) as a representative. Those N_C classes constitute the quotient set of the terms of (A.9) under the symmetry of k_p and the distinguishability rules of $\sum \prod$. Actually, each equivalence class corresponds to an unordered choice of $(p - 2(s + N))$ from a total of m σ-variables.

Henceforth, the definition (13) of the $\sum_\Pi$ operator comes in handy to define the term:

$$T_{s+N} = \sum_{\Pi}^{M} \sum_{\xi_1} \cdots$$
$$\times \sum_{\xi_{s+N}} k_p(\xi_1, \xi_1, \ldots, \xi_{s+N}, \xi_{s+N}; \sigma(\{i_1, \ldots, i_{p-2(s+N)}\})).$$

(A.20)

This term collects all the representatives of the equivalence classes we can obtain from the set of terms of (A.9) for a certain choice of s and N.

---

[3]The kernel that is derived for a system need not be symmetric but computations are greatly simplified if only symmetric kernels are considered. A simple procedure exists by which a nonsymmetric kernel can be symmetrized so that we are able to consider only symmetric kernels without any loss of generality [1].

Now, by the previous arguments and definitions, it is straightforward to rewrite (A.7) in the following equivalent form:

$$\sum_{s=0}^{[p/2]} C_s \sum_{N=0}^{[(p-2s)/2]} U_{s+N,s} T_{s+N} = T_0. \qquad (A.21)$$

For the validity of (A.21), it suffices to verify $\lfloor p/2 \rfloor + 1$ equations, the first of which is

$$C_0 U_{0,0} T_0 = T_0, \qquad (A.22)$$

and it is trivially verified as $C_0 = 1/p!$ and $U_{0,0} = p!$. The remaining $\lfloor p/2 \rfloor$ equations will be verified if it holds that

$$\sum_{s=0}^{j} C_s U_{j,s} = 0. \qquad (A.23)$$

Using definitions (A.19) and (A.8), we can write

$$\sum_{s=0}^{j} C_s U_{j,s} = \sum_{s=0}^{j} (-1)^s (1)^{j-s} \binom{j}{s} \frac{1}{j! 2^j} = 0. \qquad (A.24)$$

Then with the use of Newton's binomial formula $\sum_{s=0}^{j} \binom{j}{s} a^{j-s} b^s = (a+b)^j$, for all $a, b \in \mathbb{R}$, (A.23) follows immediately.

Finally, (A.22) and (A.23) imply (A.21) which is equivalent to (A.7) and (A.5). The validity of (A.5), for all $m, h \in \mathbb{N}$, $h \leq \lfloor m/2 \rfloor$, implies (A.1).

B. PROOF OF FORMULAS (19), (20), (21), (22), AND (23)

Exploiting (18), we will prove that formulas (19), (20), (21), (22), and (23) are valid for every finite set of distinct naturals M with cardinality m.

With the use of (18), the verification of (19) is equivalent to the verification of the following equation:

$$
\begin{aligned}
& E\{y(n)\Psi[x(n); \sigma(M)]\} \\
& = E\{y(n)D[x(n); \sigma(M)]\} \\
& \quad - \sum_{h=1}^{\lfloor m/2 \rfloor} (m-2h)! A^{m-h} {\sum_{\Pi}^{M}} k(\sigma(\{i_1, \ldots, i_{m-2h}\})).
\end{aligned} \qquad (B.1)
$$

We have to show that the definition of the Ψ operator given in (20), (21), (22), and (23) implies (B.1). This will be demonstrated using induction separately for the odd and even m cases. If we let the induction index equal to $\nu = \lfloor m/2 \rfloor + 1$, then the cases $\nu = 1, 2$ correspond to $m = 0, 2$ in the even m case and to $m = 1, 3$ in the odd m case. The cases with $m = 0, 1, 2, 3$ are verified by (18) or, alternatively, can be found proved in [9] or [10] in a different way. Hence, we consider $m = 2\nu - 2$ for the even case and $m = 2\nu - 1$ for the odd case, and suppose that (22) satisfies (19) when the induction index is $\nu - 1$. For $m > 3$ and for $1 \leq h \leq \lfloor m/2 \rfloor$, this is equivalent to supposing the following equation valid:

$$
\begin{aligned}
& (m-2h)! A^{m-2h} k(\sigma(\{i_1, \ldots, i_{m-2h}\})) \\
& \quad = E\{y(n)\Psi[x(n); \sigma(\{i_1, \ldots, i_{m-2h}\})]\}.
\end{aligned} \qquad (B.2)
$$

Using (B.2), (B.1) can be rewritten as

$$
\begin{aligned}
& E\{y(n)\Psi[x(n); \sigma(M)]\} \\
& = E\{y(n)D[x(n); \sigma(M)]\} \\
& \quad - \sum_{h=1}^{\lfloor m/2 \rfloor} A^h {\sum_{\Pi}^{M}} E\{y(n)\Psi[x(n); \sigma(\{i_1, \ldots, i_{m-2h}\})]\}.
\end{aligned} \qquad (B.3)
$$

Further, for the properties of the expectation and the $\sum_{\Pi}$ operators, (B.3) can be rewritten as follows:

$$
\begin{aligned}
& E\{y(n)\Psi[x(n); \sigma(M)]\} \\
& = E\Bigg\{ y(n) \Bigg[D[x(n); \sigma(M)] \\
& \quad - \sum_{h=1}^{\lfloor m/2 \rfloor} A^h {\sum_{\Pi}^{M}} \Psi[x(n); \sigma(\{i_1, \ldots, i_{m-2h}\})] \Bigg] \Bigg\}.
\end{aligned} \qquad (B.4)
$$

Due to the arbitrary choice of $y(n)$, (22) and (23) guarantee a recursive definition of Ψ which is a solution for the ν-index case of the induction. So (22) or (23) is a solution for (B.1).

It is left to prove that (21) fits formulas (22) and (23), so (21) will be the explicit operative solution for (19). Exploiting (21) and (14) in the right member of (23), we get

$$
\begin{aligned}
D[x(n); \sigma(M)] & = \sum_{h=0}^{\lfloor m/2 \rfloor} A^h {\sum_{\Pi}^{M}} \sum_{\ell=0}^{\lfloor m/2-h \rfloor} (-1)^\ell A^\ell \\
& \quad \times {\sum_{\Pi}^{\{i_1, \ldots, i_{m-2h}\}}} D[x(n); \sigma(\{i_1, \ldots, i_{m-2h-2\ell}\})].
\end{aligned} \qquad (B.5)
$$

From the definition of the $\sum_{\Pi}$ operator and from the property proved in Appendix C, it is easy to derive the rules that allow to rewrite (B.5) in this way:

$$
\begin{aligned}
& D[x(n); \sigma(M)] \\
& = \sum_{h=0}^{\lfloor m/2 \rfloor} \sum_{\ell=0}^{\lfloor m/2-h \rfloor} (-1)^\ell A^{\ell+h} C(m, m-2h, m-2h-2\ell) \\
& \quad \times {\sum_{\Pi}^{M}} D[x(n); \sigma(\{i_1, \ldots, i_{m-2h-2\ell}\})].
\end{aligned} \qquad (B.6)
$$

After collecting similar terms in (B.6) (i.e., the terms with equal $\ell + h$), it can be stated that the validity of the following equation:

$$\sum_{j=0}^{h+\ell} (-1)^{h+\ell-j} C(m, m-2h-2\ell+2j, m-2h-2\ell) = 0 \qquad (B.7)$$

suffices for the validity of (B.6). Using definition (C.2) and after some trivial manipulations, (B.7) becomes

$$\sum_{j=0}^{h+\ell} (-1)^{(h+\ell-j)} (1)^j \binom{h+\ell}{j} = 0. \qquad (B.8)$$

With Newton's binomial formula the verification of (B.8) is straightforward. Hence also (B.6) holds and so does (B.5). This suffices to state that (21) is a solution for (22) and then for (19).

C. A PROPERTY OF THE $\sum_{\Pi}$ OPERATOR

Let M be a finite set of positive distinct integers and R, Q such that $R \subseteq Q \subseteq M$ with cardinalities m, r, and q, respectively, m, r, and q must be all odd or all even integers. Let f be a symmetrical mapping with respect to the argument $\boldsymbol{\sigma}(R)$. Under this hypothesis, it holds that

$$\sum_{\Pi}^{M} \sum_{\Pi}^{Q} f(\cdot; \boldsymbol{\sigma}(R)) = C(m,q,r) \sum_{\Pi}^{M} f(\cdot; \boldsymbol{\sigma}(R)), \quad (C.1)$$

with

$$C(m,q,r) = \frac{((m-r)/2)!}{((m-q)/2)!((q-r)/2)!}. \quad (C.2)$$

To prove this, let $\mathbf{S}$ be a mapping which associates a sum of terms with the set of the terms being summed. It must be observed that the first and the second member of (C.1) are actually made of sums of terms. So we can associate two sets to the sums in the left and right members of (C.1) in this way:

$$
\begin{aligned}
A &= \left\{ a : a \in \mathbf{S}\left(\sum_{\Pi}^{M} \sum_{\Pi}^{Q} f(\cdot; \boldsymbol{\sigma}(R)) \right) \right\}, \\
B &= \left\{ b : b \in \mathbf{S}\left(\sum_{\Pi}^{M} f(\cdot; \boldsymbol{\sigma}(R)) \right) \right\}.
\end{aligned}
\quad (C.3)
$$

Now, the proof of (C.1) can be made by proving that

(1) for all $a \in A$, there exists $b \in B$ such that $b \equiv a$;
(2) for all $b \in B$, there exists $A_b = \{a \in A \mid a \equiv b\} \neq \varnothing$, $|A_b| = C(m,q,r)$.

We consider first the item (1). Using definition (13) of $\sum_{\Pi}$, the left and the right member of (C.1) can be made more explicit (it could be done also in the former definitions of A and B):

$$\sum_{j=1}^{\binom{m}{q}} \sum_{i=1}^{\binom{q}{r}} f(\cdot; \boldsymbol{\sigma}(R_i)) \sum \prod \boldsymbol{\sigma}(Q_j - R_i) \sum \prod \boldsymbol{\sigma}(M - Q_j),$$
$$(C.4)$$

$$\sum_{h=1}^{\binom{m}{r}} f(\cdot; \boldsymbol{\sigma}(R_h)) \sum \prod \boldsymbol{\sigma}(M - R_h), \quad (C.5)$$

Q_j being a combination of q elements chosen from m elements of M, R_i a combination of r elements chosen from q elements of Q_j, and R_h a combination of r elements chosen from m elements of M. We also need the definition of the sets of addenda associated to a particular choice of R_i, Q_j and R_h in this way:

$$
\begin{aligned}
A_{ij} &= \mathbf{S}\Big(f(\cdot; \boldsymbol{\sigma}(R_i)) \sum \prod \overline{\boldsymbol{\sigma}}^{Q_j}(R_i) \sum \prod \overline{\boldsymbol{\sigma}}^{M}(Q_j) \Big), \\
B_h &= \mathbf{S}\Big(f(\cdot; \boldsymbol{\sigma}(R_h)) \sum \prod \overline{\boldsymbol{\sigma}}^{M}(R_h) \Big).
\end{aligned}
\quad (C.6)
$$

It obviously holds that $A = \bigcup_{i,j} A_{ij}$ and $B = \bigcup_h B_h$. If now we consider two sets of distinct positive integers $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, exploiting the properties deriving from definition (11) of $\sum \prod$, it is easy to prove that

$$\mathbf{S}\Big(\sum \prod \boldsymbol{\sigma}(\boldsymbol{\alpha}) \times \sum \prod \boldsymbol{\sigma}(\boldsymbol{\beta}) \Big) \subseteq \mathbf{S}\Big(\sum \prod \boldsymbol{\sigma}(\boldsymbol{\alpha} \cup \boldsymbol{\beta}) \Big). \quad (C.7)$$

Then, noting that for every i, j allowed by (C.4), R_i is a combination of elements of Q_j, $Q_j \subseteq M$ implies that R_i is also a combination of elements of M. Hence there exists at least one h (among the ones allowed by (C.5)) such that $R_h = R_i$. With these arguments, it can be said that $M - R_h = (Q_j - R_i) \cup (M - Q_j)$ holds. From the preceding expression and from (C.7), it trivially follows that $A_{ij} \subseteq B_h$, and then using (C.6), it follows that, for all $a_{ij} \in A_{ij}$, there exists $b_h \in B_h$ such that $a_{ij} \equiv b_h$. Item (1) has been proved.

Now we will focus on item (2). If we choose a set R_h allowed by (C.7) and the corresponding B_h, an arbitrary element $b \in B_h$ would be described by the following expression:

$$b = f(\cdot; \boldsymbol{\sigma}(R_h)) \delta_{\sigma_{i_1} \sigma_{i_2}} \cdots \delta_{\sigma_{i_{p-r-1}} \sigma_{i_{p-r}}}, \quad (C.8)$$

with $\{i_1, \ldots, i_{m-r}\} = (M - R_h)$. Note that to every factor δ, a pair of subscripts is associated. The number of subscript pairs for the term b is equal to $|M - R_h|/2 = (m-r)/2$. Now we choose a two-set partition of the factors of b, with $(q-r)/2$ and $(m-q)/2$ elements, respectively. To the two-set of factors just obtained will be associated the two corresponding sets having, as elements, the indices of the σ-variables in the subscripts $I' = \{i'_1, \ldots, i'_{q-r}\}$, $I'' = \{i''_1, \ldots, i''_{m-q}\}$, $I' \cup I'' = (M - R_h)$, and $I' \cap I'' = \varnothing$. Now we will pick up only the part of b having the factors belonging to the indices set I' to form the term b':

$$b' = f(\cdot; \boldsymbol{\sigma}(R_h)) \delta_{\sigma_{i'_1} \sigma_{i'_2}} \cdots \delta_{\sigma_{i'_{q-r-1}}} \sigma_{i'_{q-r}}. \quad (C.9)$$

It must be observed that the term b' can be generated only by the inner sum of (C.4). In particular, it is generated only when $Q_j = I' \cup R_h$. Q_j is an allowed choice of q elements among the m elements of M, and it also holds that $I'' = M - Q_j$. From this, it follows that in the expansion of (C.4), there exists only one group of addenda as follows:

$$f(\cdot; \boldsymbol{\sigma}(R_h)) \delta_{\sigma_{i'_1} \sigma_{i'_2}} \cdots \delta_{\sigma_{i'_{q-r-1}} \sigma_{i'_{q-r}}} \sum \prod \boldsymbol{\sigma}(M - Q_j). \quad (C.10)$$

This group, by the definition of $\sum \prod$, will surely contain once the addend (C.8). So we showed that for all $b \in B$ and for all partitions of the factors of b in two groups of $(q-r)/2$ and of $(m-q)/2$ elements, there exists a choice for Q_j which guarantees that there exists one and only one element of A which is congruent with b, and so $A_b \neq \varnothing$. Moreover, $|A_b|$ is equal to the number of all possible such partitions of the factors of b. This number is obviously equal to the number of permutations of $(m-r)/2$ elements with the repetition of two elements $(q-r)/2$ and $(m-q)/2$ times, respectively. Hence we get $|A_b| = ((m-r)/2)!/((q-r)/2)!((m-q)/2)!$. This proves item (2).

ACKNOWLEDGMENTS

REFERENCES

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

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

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

[4] G. Palm and T. Poggio, "The Volterra representation and the Wiener expansion: validity and pitfalls," *SIAM J. Appl. Math.*, vol. 33, no. 2, pp. 195–216, 1977.

[5] G. Palm and T. Poggio, "Stochastic identification methods for nonlinear systems: an extension of the Wiener theory," *SIAM J. Appl. Math.*, vol. 34, no. 3, pp. 524–534, 1978.

[6] S. Orcioni, M. Pirani, C. Turchetti, and M. Conti, "Practical notes on two Volterra filter identification direct methods," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS '02)*, vol. 3, pp. 587–590, Phoenix-Scottsdale, Ariz, USA, May 2002.

[7] V. Z. Marmarelis, "Random versus pseudorandom test signals in nonlinear-system identification," *Proc. IEE*, vol. 125, no. 5, pp. 425–428, 1978.

[8] D. T. Westwick, B. Suki, and K. R. Lutchen, "Sensitivity analysis of kernel estimates: implications in nonlinear physiological system identification," *Annals of Biomedical Engineering*, vol. 26, no. 3, pp. 488–501, 1998.

[9] Y. Goussard, W. Krenz, and L. Stark, "An improvement of the Lee and Schetzen cross-correlation method," *IEEE Trans. Automatic Control*, vol. 30, no. 9, pp. 895–898, 1985.

[10] P. Koukoulas and N. Kalouptsidis, "Nonlinear system identification using Gaussian inputs," *IEEE Trans. Signal Processing*, vol. 43, no. 8, pp. 1831–1841, 1995.

[11] V. P. Leonov and A. N. Shiryaev, "On a method of calculation of semi-invariants," *Theory of Probability and Its Applications*, vol. 4, pp. 319–328, 1959.

[12] N. Rozario and A. Papoulis, "Some results in the application of polyspectra to certain nonlinear communication systems," in *Proc. IEEE Signal Processing Workshop on Higher-Order Statistics*, pp. 37–41, South Lake Tahoe, Calif, USA, June 1993.

[13] V. Volterra, *Theory of Functionals and of Integral and Integro-Differential Equations*, Dover Publications, New York, NY, USA, 1959.

[14] G. Palm, "On representation and approximation of nonlinear systems," *Biological Cybernetics*, vol. 31, no. 2, pp. 119–124, 1978.

[15] I. W. Sandberg and L. Xu, "Approximation of myopic systems whose inputs need not be continuous," *Multidimensional Systems and Signal Processing*, vol. 9, no. 2, pp. 207–225, 1998.

[16] I. W. Sandberg, "Bounds for discrete-time Volterra series representations," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 46, no. 1, pp. 135–139, 1999.

[17] I. W. Sandberg, "Uniform approximation with doubly-finite Volterra series," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS '91)*, vol. 2, pp. 754–757, Singapore, June 1991.

[18] S. Boyd and L. Chua, "Fading memory and the problem of approximating nonlinear operators with Volterra series," *IEEE Trans. Circuits and Systems*, vol. 32, no. 11, pp. 1150–1161, 1985.

Massimiliano Pirani received the Laurea and Ph.D. degrees in electronics engineering from the Università degli Studi di Ancona, Ancona, Italy, in 2000 and 2004, respectively. His research interests include identification of nonlinear systems, higher-order statistics, nonlinear signal processing, audio electronics, speech processing, and loudspeaker systems enhancement.

Simone Orcioni received the Laurea (M.S.) degree in electronics engineering from the Università degli Studi di Ancona, Italy, in 1992, and the Ph.D. degree in 1995. From 1997 to 1999, he was postdoctoral fellow at the same university, where, in 2000, he became an Assistant Professor. Recently, he has joined the Department of Electronics, Artificial Intelligence, and Telecommunications of the Università Politecnica delle Marche, Ancona. Since 1992, he has been working in statistical device modeling and simulation, parametric yield optimization, neural networks, fuzzy systems, and analog circuit design. His research interests also include RF, system level circuit design, and Wiener-Volterra series.

Claudio Turchetti received the Laurea (M.S.) degree in electronics engineering from the Università degli Studi di Ancona, Italy, in 1979. He joined the Department of Electronics, Università degli Studi di Ancona in 1980. He is currently a full Professor of Applied Electronics and Integrated Circuits Design and Head of the Department of Electronics, Artificial Intelligence, and Telecommunications at the Università Politecnica delle Marche, Ancona. He has been active in the areas of device modeling, circuits simulation at the device level, and design of integrated circuits. His current research interests are also in analog neural networks and statistical analysis of integrated circuits for parametric yield optimization.