Provided for non-commercial research and education use.
Not for reproduction, distribution or commercial use.

(This is a sample cover image for this issue. The actual cover is not yet available at this time.)

This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier's archiving and manuscript policies are encouraged to visit:

http://www.elsevier.com/copyright

Author's personal copy

Signal Processing 93 (2013) 1210–1220

Contents lists available at SciVerse ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Efficient adaptive identification of linear-in-the-parameters nonlinear filters using periodic input sequences

Alberto Carini [a,*], Giovanni L. Sicuranza [b], V. John Mathews [c]

[a] DiSBeF, University of Urbino "Carlo Bo", 61029 Urbino, Italy
[b] DIA, University of Trieste, 34127 Trieste, Italy
[c] Department of Electrical and Computer Engineering, University of Utah, Salt Lake City, UT 84112-9206, USA

ARTICLE INFO

Article history:
Received 13 September 2012
Received in revised form
16 December 2012
Accepted 17 December 2012
Available online 26 December 2012

Keywords:
Adaptive filters
Adaptive signal processing
Nonlinear system identification
Nonlinear systems

ABSTRACT

This paper introduces computationally efficient NLMS and RLS adaptive algorithms for identifying non-recursive, linear-in-the-parameters (LIP) nonlinear systems using periodic input sequences. The algorithms presented in the paper are exact and require a real-time computational effort of a single multiplication, an addition and a subtraction per input sample. The transient, steady state, and tracking behavior of the algorithms as well as the effect of model mismatch is studied in the paper. The low computational complexity, good performance and broad applicability make the approach of this paper a valuable alternative to the current techniques for nonlinear system identification.
© 2012 Elsevier B.V. All rights reserved.

1. Introduction

This paper considers the problem of identifying or tracking non-recursive linear-in-the-parameters (LIP) nonlinear systems. This class of nonlinear systems and filters is composed of all systems whose input–output relationships depend linearly on the system parameters [1]. The class includes, in addition to linear filters, truncated Volterra filters [2,3], extended Volterra filters [4], FLANN filters [5] based on trigonometric, polynomial, and piece-wise linear expansions [6–9], radial basis function networks [10,11], interpolated Volterra filters [12–14], generalized memory polynomial filters [15], and many other nonlinear structures [1]. Different approaches have been proposed in the literature to identify these systems. A fundamental difficulty with the identification of these nonlinear systems is the complexity of the system model and the correspondingly large computational complexity of the identification algorithm. For example, a generic pth order truncated Volterra system model with N sample memory has $O(N^p)$ coefficients. The lowest computational complexity for adaptive identification and tracking available today is $O(N^p)$ arithmetical operations per input signal sample. Another fundamental problem is the relatively slow learning speed of the identification algorithm. Even when the input signal is white Gaussian, the autocorrelation matrix of the input data is often non-diagonal and ill-conditioned [3].

Recently, the first author of this paper presented an algorithm for the identification and tracking of linear FIR systems using periodic input sequences [16]. The algorithm was derived on the basis of the early work of Antweiler on the NLMS algorithm with perfect periodic inputs[1] [17–19]. In [16], efficient NLMS and RLS algorithms that have a real-time computational effort of a single multiplication, an addition and a subtraction per input sample were discussed. These algorithms do not evaluate the coefficients of the underlying system directly. Instead, they determine the coefficients of an equivalent representation, from which the impulse response can be easily computed. The paper also showed

* Corresponding author. Tel.: +39 0722 304412; fax: +39 0722 4475.
E-mail address: alberto.carini@uniurb.it (A. Carini).

[1] A perfect periodic sequence of period M has autocorrelation given by the repetition with period M of a unit pulse.

0165-1684/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.sigpro.2012.12.012

Author's personal copy

that the algorithms have convergence and tracking properties that can be better than or comparable to the NLMS algorithm for white noise input.

In this paper, we extend the approach of [16] to the identification and tracking of nonrecursive LIP nonlinear systems (some preliminary results were presented in [20]). The resulting systems preserve the low computational complexity as well as the good convergence and tracking properties exhibited by their linear counterparts. The derivation of the algorithms as well as the analysis techniques shares some similarity with those for linear systems. However, there are substantial differences in the influence of the input signal as well as its design for the nonlinear case. Similarly, there are differences in the transient, steady-state and tracking analyses as well as model mismatch analyses between the linear and nonlinear cases. We will refer to the derivations in [16] in cases they are similar to the linear case and concentrate our attention on aspects that are novel with respect to the linear case. Of particular interest is the discussion of the characteristics and design of an ideal periodic sequence suitable for nonlinear system identification.

The properties of pseudorandom multilevel sequences, i.e., of periodic sequences, used to identify Volterra and extended Volterra filters were studied in [4]. It was shown that a pseudorandom multilevel sequence of degree D [4] can persistently excite an extended Volterra filter of order P and memory length N if and only if it has at least $P+1$ distinct levels and $D \geq N$. An efficient algorithm for the least-square parameter estimation was also proposed in [4]. In [21], a Wiener model was estimated using a multilevel sequence. The approach of this paper differs from those of [4,21] in two ways: (1) our derivations are applicable to the broader class of LIP nonlinear filters and (2) this paper deals with the adaptive NLMS and RLS algorithms.

The paper is organized as follows. In Section 2 the class of LIP nonlinear filters is reviewed and a representation using periodic inputs is introduced. In Section 3 the efficient NLMS and RLS algorithms are derived. The transient, steady-state and tracking behaviors of the algorithms are analyzed in Section 4. The effect of a model mismatch between the unknown system and the adaptive filter is discussed in Section 5. Simulation results are presented in Section 6. Concluding remarks are given in Section 7.

Throughout the paper, lowercase boldface letters denote vectors, uppercase boldface letters denote matrices, the symbol $\otimes$ indicates the Kronecker product, $E[\cdot]$ denotes the statistical expectation, $\|\cdot\|$ denotes the Euclidean norm, $\|\cdot\|_F$ represents the Frobenius norm, $\lfloor\cdot\rfloor$ is the largest integer smaller than or equal to the argument, a mod b is the remainder of the division of a by b, $\mathrm{Cond}_2(\mathbf{X})$ is the condition number in the 2-norm of matrix $\mathbf{X}$, $\mathrm{Cond}_F(\mathbf{X})$ is the condition number in the Frobenius norm of $\mathbf{X}$, $\mathbf{X}^{-T}$ is the transposed inverse of $\mathbf{X}$, and $\mathbf{I}$ is an identity matrix.

2. LIP nonlinear filters with periodic input signals

2.1. A review of LIP nonlinear filters

The algorithms described in the next section apply to non-recursive LIP models with a finite memory of N samples. The input–output relationship of such models can be expressed in the vector form as

$$y(n) = \mathbf{h}^T(n)\mathbf{x}_{\mathcal{F}}(n), \tag{1}$$

where $\mathbf{h}(n)$ is a length M coefficient vector and $\mathbf{x}_{\mathcal{F}}(n)$ is an input data vector of the same length. Let us assume that the vector

$$\mathbf{x}_{\mathcal{F}}(n) = [x_{\mathcal{F}1}(n), x_{\mathcal{F}2}(n), \ldots, x_{\mathcal{F}M}(n)] \tag{2}$$

is composed of M terms formed with any linear or nonlinear combination and/or nonlinear expansion of the N most recent samples of the input signal

$$\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-N+1)]. \tag{3}$$

More specifically, each term $x_{\mathcal{F}r}(n)$ in (2), with $r = 1, \ldots, M$, is a nonlinear function $f_r(\mathbf{x}_r(n))$ of vector $\mathbf{x}_r(n)$ and $\mathbf{x}_r(n)$ a vector formed by a subset of the elements of (3). The class of filters in (1) is broad and includes many common nonlinear models [1]. In Section 6, simulation results are provided for a second order FLANN filter based on trigonometric expansions [7], where

$$\begin{aligned}\mathbf{x}_{\mathcal{F}}(n) = [&x(n), \ldots, x(n-N+1), \sin(\pi x(n)), \ldots, \sin(\pi x(n-N+1)),\\ &\cos(\pi x(n)), \ldots, \cos(\pi x(n-N+1)), \sin(2\pi x(n)), \ldots,\\ &\sin(2\pi x(n-N+1)), \cos(2\pi x(n)), \ldots, \cos(2\pi x(n-N+1))]^T,\end{aligned} \tag{4}$$

the truncated second-order Volterra filters [3], where

$$\begin{aligned}\mathbf{x}_{\mathcal{F}}(n) = [&x(n), \ldots, x(n-N+1), x^2(n), \ldots, x^2(n-N+1),\\ &x(n)x(n-1), \ldots, x(n-N+2)x(n-N+1), \ldots, x(n)x(n-N+1)]^T,\end{aligned} \tag{5}$$

and the extended second-order Volterra filter [4], where

$$\mathbf{x}_{\mathcal{F}}(n) = \begin{bmatrix}1\\ x(n)\\ x^2(n)\end{bmatrix} \otimes \begin{bmatrix}1\\ x(n-1)\\ x^2(n-1)\end{bmatrix} \otimes \cdots \otimes \begin{bmatrix}1\\ x(n-N+1)\\ x^2(n-N+1)\end{bmatrix}. \tag{6}$$

2.2. LIP nonlinear filters with periodic inputs

Assume that the input sequence $x(n)$ is periodic with period M. Then, the data vector $\mathbf{x}_{\mathcal{F}}(n)$ can take only one of M different values, say $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{M-1}$, i.e.

$$\mathbf{x}_{\mathcal{F}}(n) = \mathbf{x}_i \quad \text{for } i = n \bmod M. \tag{7}$$

If the $M \times M$ matrix

$$\mathbf{X}_{\mathcal{F}} = [\mathbf{x}_0, \ldots, \mathbf{x}_{M-1}] \tag{8}$$

is invertible, we can define an $M \times M$ matrix $\mathbf{W}$ such that

$$\mathbf{W}\mathbf{X}_{\mathcal{F}}^T = \mathbf{I}. \tag{9}$$

Since $\mathbf{W}$ is invertible, we can find a vector $\mathbf{c}(n) = [c_0(n), \ldots, c_{M-1}(n)]^T$ such that

$$\mathbf{h}(n) = \mathbf{W}\mathbf{c}(n). \tag{10}$$

Let $\mathbf{w}_0, \ldots, \mathbf{w}_{M-1}$ represent the M columns of $\mathbf{W}$. These vectors are in general not orthogonal (even when the input sequence is a perfect periodic sequence [19]) but they are linearly independent (for the invertibility of $\mathbf{X}_{\mathcal{F}}$). Therefore, the expression in (1) can be equivalently

Author's personal copy

rewritten as

$$y(n) = \mathbf{c}^T(n)\mathbf{W}^T\mathbf{x}_{\mathcal{F}}(n) = \sum_{i=0}^{M-1} c_i(n)\mathbf{w}_i^T\mathbf{x}_{\mathcal{F}}(n), \quad (11)$$

where the vectors $\mathbf{w}_i$ are known once the periodic sequence is chosen.

According to (11) the nonlinear filter can be decomposed in a parallel structure formed by M nonlinear filters with fixed coefficients $\mathbf{w}_i$. The output signal $y(n)$ is the linear combination with coefficients $c_i(n)$ of the outputs of these nonlinear filters, as described in Fig. 1.

The coefficient vector $\mathbf{c}(n)$ characterizes the nonlinear filter in (1) as well as the coefficient vector $\mathbf{h}(n)$, and $\mathbf{h}(n)$ can be easily estimated from the knowledge of $\mathbf{c}(n)$ as in (10). Taking into account (7) and since according to (9)

$$\mathbf{w}_i^T\mathbf{x}_j = \begin{cases} 1 & \text{when } i = j, \\ 0 & \text{when } i \neq j, \end{cases} \quad (12)$$

we have

$$y(n) = c_i(n) \quad \text{with } i = n \bmod M. \quad (13)$$

Thus, the periodic output signal for the nonlinear system is identical to one of the coefficients $c_i(n)$ of the equivalent filter $\mathbf{c}(n)$ in (10). This is the same property exploited in [16] to derive the efficient NLMS and RLS algorithms for identifying linear FIR systems.

The nonlinear filter representation in (11) is meaningful (and the adaptive filters of the next section can identify the nonlinear filter in (1)) only when the matrix $\mathbf{X}_{\mathcal{F}}$ is invertible. Design of periodic sequences for which $\mathbf{X}_{\mathcal{F}}$ is invertible and has good condition number is discussed in Section 4.3.

3. Efficient NLMS and RLS algorithms

To derive the NLMS algorithm, we apply the standard procedure used to develop the LMS algorithm and we show that it results in a normalized LMS algorithm for the filter structure of this paper.

We want to find the coefficients $c_i(n)$ that minimize the following minimum-mean-square cost function:

$$J(n) = E[(d(n) - y(n))^2], \quad (14)$$

where $d(n)$ is the desired signal and $y(n)$ is as given in (11).

The coefficients are adapted with the gradient method

$$c_i(n+1) = c_i(n) - \frac{\mu}{2}\frac{\partial J(n)}{\partial c_i(n)}. \quad (15)$$

By approximating $J(n)$ with $(d(n)-y(n))^2$ and taking into account (13), it can be verified that

$$\frac{\partial J(n)}{\partial c_i(n)} \simeq \begin{cases} -2[d(n) - c_i(n)] & \text{when } i = n \bmod M, \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

Thus

$$c_i(n+1) = \begin{cases} c_i(n) + \mu[d(n) - c_i(n)] & \text{when } i = n \bmod M, \\ c_i(n) & \text{otherwise.} \end{cases} \quad (17)$$

This adaptation equation can also be written in the vector form as follows:

$$\mathbf{c}(n+1) = \mathbf{c}(n) + \mu[d(n) - \mathbf{c}^T(n)\mathbf{e}_i]\mathbf{e}_i, \quad (18)$$

where $i = n \bmod M$ and $\mathbf{e}_i$ is the $i+1$-th column of the $M \times M$ element identity matrix. For $\mu \neq 1$, this adaptive filter requires only a multiplication, an addition and a subtraction per input sample. When μ is an integer-power of two, this is a multiplication-free adaptive filter.

The vector expression in (18) proves that the adaptive algorithm is an NLMS algorithm because $\mathbf{e}_i$ has unit norm. The algorithm is also an affine projection algorithm of order M [22]. It is shown in Section 4 that, for $\mu = 1$, $[d(n-k) - c_i(n+1)] = 0$ with $i = (n-k) \bmod M$ and $0 \le k < M$. Thus, the algorithm provides the minimum coefficient variation that set to zero the last M *a posteriori* estimation errors.

Similarly, in the exponentially weighted RLS algorithm we want to find the coefficients $c_i(n)$ that minimize the cost function

$$J(n) = \sum_{j=0}^{n} \lambda^{n-j}[d(j) - y(j)]^2, \quad (19)$$

where $y(n)$ is as given in (11) and λ is a forgetting factor, $0 \ll \lambda \le 1$. By following arguments similar to those in [16], it can be proved that

$$c_i(n+1) = \begin{cases} c_i(n) + \rho\left(\left\lfloor \frac{n}{M} \right\rfloor\right)[d(n) - c_i(n)] & \text{for } i = n \bmod M, \\ c_i(n) & \text{otherwise,} \end{cases} \quad (20)$$

or in vector form (with $i = n \bmod M$)

$$\mathbf{c}(n+1) = \mathbf{c}(n) + \rho\left(\left\lfloor \frac{n}{M} \right\rfloor\right)[d(n) - \mathbf{c}^T(n)\mathbf{e}_i]\mathbf{e}_i. \quad (21)$$

For $\lambda < 1$, $\rho(m) = (1-\lambda^M)/(1-\lambda^{M(m+1)})$, and for $\lambda = 1$, $\rho(m) = 1/(m+1)$. Since $\rho(n)$ is computed only once every M samples, Eq. (20) requires only a multiplication, an addition and a subtraction per input sample.

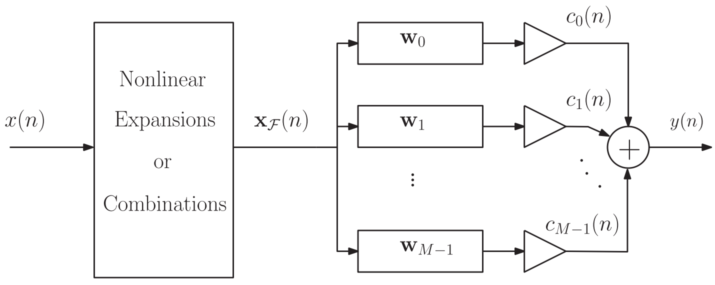

Fig. 1. Block diagram of the nonlinear filter structure of Eq. (11).

Author's personal copy

By comparing the adaptation rules of (17) and (20), we see that the RLS algorithms with a periodic sequence excitation can be interpreted as variable step-size NLMS algorithms. In particular, for $\lambda < 1$

$$\lim_{n \to +\infty} \rho(n) = 1 - \lambda^M. \quad (22)$$

Thus, as was the case for linear FIR systems in [16], for $n \to +\infty$, the RLS algorithm in (20) with $\lambda < 1$ exhibits similar tracking and steady-state performances as the NLMS algorithm of (17) with $\mu = 1 - \lambda^M$.

A substantial advantage of the algorithms in (17) and (20) is their reduced computational complexity. Furthermore, from (18) and (21) we have that the input data autocorrelation matrix for both algorithms is given by

$$\mathbf{R} = \frac{1}{M} \sum_{i=0}^{M} \mathbf{e}_i \mathbf{e}_i^T = \frac{1}{M} \mathbf{I}, \quad (23)$$

independently from the choice of the periodic input sequence. Since $\mathbf{R}$ is a diagonal matrix, the algorithms should exhibit good transient, steady-state and tracking properties. These issues are explored next.

4. Transient, steady-state, and tracking analyses

In this section, we first study the transient and steady-state behavior of the algorithms in the identification of time-invariant LIP systems. Later we study the algorithms' tracking properties in the case of time-varying LIP systems. Finally, we discuss the design of periodic sequences suitable for identification and tracking of LIP nonlinear systems. We consider algorithms of the form (20), noting that replacing $\rho(m)$ by μ in (20) leads to the NLMS algorithm, and when $\rho(m)$ varies as defined after (21), we get the RLS algorithm.

4.1. Transient and steady-state analysis

Assume that we wish to identify a time-invariant nonlinear system with memory length N and input–output relationship

$$\begin{aligned} d(n) &= \tilde{\mathbf{h}}^T \mathbf{x}_{\mathcal{F}}(n) + \nu(n) \\ &= \sum_{k=0}^{M-1} \tilde{c}_k \mathbf{w}_k^T \mathbf{x}_{\mathcal{F}}(n) + \nu(n) \\ &= \tilde{c}_i + \nu(n) \quad \text{for } i = n \bmod M, \end{aligned} \quad (24)$$

where $\tilde{\mathbf{h}}$ and $\tilde{\mathbf{c}} = [\tilde{c}_0, \tilde{c}_1, \ldots, \tilde{c}_{M-1}]^T$ are the fixed coefficient vectors of the nonlinear system to be identified related by (10), and $\nu(n)$ is a zero-mean stationary additive measurement noise uncorrelated with $x(n)$ and with power σ_ν^2. To simplify the analysis and avoid any transient effect on $d(n)$, we assume that the unknown system has operated on the periodic input signal for at least N samples before the algorithm was activated.

Let us define the system errors $\overline{c}_i(n) = c_i(n) - \tilde{c}_i$. By subtracting $\tilde{c}_i$ from both sides of (20) and taking into account (24), we obtain

$$\overline{c}_i(n+1) = \begin{cases} \overline{c}_i(n) + \rho\left(\left\lfloor \frac{n}{M} \right\rfloor\right)(\nu(n) - \overline{c}_i(n)) & \text{for } i = n \bmod M, \\ \overline{c}_i(n) & \text{otherwise.} \end{cases} \quad (25)$$

Recall that $\rho(\lfloor n/M \rfloor) = 1$ for $0 \le n < M$ in the RLS algorithm. It immediately follows that, for the noiseless case ($\nu(n) = 0$), $\overline{c}_i(n) = 0$ for $n \ge M$ for the RLS algorithm and the NLMS algorithm with $\mu = 1$. Thus, both systems converge within M samples in the noiseless case.

We now study the mean-square error (MSE)

$$\text{MSE}(n) = E[\| d(n) - y(n) \|^2], \quad (26)$$

the mean-square deviation (MSD) of $\mathbf{c}(n)$

$$\text{MSD}_{\mathbf{c}}(n) = E[\| \mathbf{c}(n) - \tilde{\mathbf{c}} \|^2] = E[\| \overline{\mathbf{c}}(n) \|^2], \quad (27)$$

with $\overline{\mathbf{c}}(n) = [\overline{c}_0(n), \ldots, \overline{c}_{M-1}(n)]^T$, and the MSD of $\mathbf{h}(n) = \sum_{i=0}^{M-1} c_i(n) \mathbf{w}_i$

$$\text{MSD}_{\mathbf{h}}(n) = E[\| \mathbf{h}(n) - \tilde{\mathbf{h}} \|^2] = E\left[\left\| \sum_{i=0}^{M-1} \overline{c}_i(n) \mathbf{w}_i \right\|^2 \right]. \quad (28)$$

The derivations and the expressions of MSE(n) and $\text{MSD}_{\mathbf{c}}(n)$ are identical to the linear case studied in [16], and we provide here only the final expressions. The result for $\text{MSD}_{\mathbf{h}}(n)$ is different from that of the linear case, and thus we discuss its derivation in more detail.

For $n = mM + j$, with $0 \le j < M$, we can prove that for all j

$$\text{MSE}[(m+1)M + j] = [1 - \rho(m)]^2 \text{MSE}(mM + j) + 2\rho(m)\sigma_\nu^2. \quad (29)$$

Similarly, when $\rho(m) = \mu$, the following holds:

$$\text{MSD}_{\mathbf{c}}((m+1)M + j) = [1 - \rho(m)]^2 \text{MSD}_{\mathbf{c}}(mM + j) + \rho^2(m) M \sigma_\nu^2. \quad (30)$$

This equation approximately holds also for a variable $\rho(m)$ provided $\rho(m+1) \simeq \rho(m)$, which is true in RLS algorithms for $n \to +\infty$.

For $\text{MSD}_{\mathbf{h}}(n)$, we work under the hypothesis that $\nu(n)$ and $\overline{c}_i(n)$ are independent and that the $\overline{c}_i(n)$'s are uncorrelated. With the latter hypothesis, we have from (28)

$$\text{MSD}_{\mathbf{h}}(n) = \sum_{i=0}^{M-1} E[\overline{c}_i^2(n)] \| \mathbf{w}_i \|^2. \quad (31)$$

Since only $\overline{c}_i(n)$ with $i = n \bmod M$ varies at each time n

$$\text{MSD}_{\mathbf{h}}(n+1) = \text{MSD}_{\mathbf{h}}(n) + E[\overline{c}_i^2(n+1) - \overline{c}_i^2(n)] \| \mathbf{w}_i \|^2. \quad (32)$$

From (25) we have

$$E[\overline{c}_i^2(n+1)] = E[\overline{c}_i^2(n)] \left(1 - \rho\left(\left\lfloor \frac{n}{M} \right\rfloor\right)\right)^2 + \rho^2\left(\left\lfloor \frac{n}{M} \right\rfloor\right) \sigma_n u^2, \quad (33)$$

with $i = n \bmod M$. Substituting (33) into (32) gives

$$\begin{aligned} \text{MSD}_{\mathbf{h}}(n+1) = \text{MSD}_{\mathbf{h}}(n) - \Big[2\rho\left(\left\lfloor \frac{n}{M} \right\rfloor\right) \\ - \rho^2\left(\left\lfloor \frac{n}{M} \right\rfloor\right) \Big] E[\overline{c}_i^2(n)] \| \mathbf{w}_i \|^2 + \rho^2\left(\left\lfloor \frac{n}{M} \right\rfloor\right) \sigma_\nu^2 \| \mathbf{w}_i \|^2. \end{aligned} \quad (34)$$

Author's personal copy

Table 1
$\mathrm{MSE}(+\infty)$, $\mathrm{MSD}_{\mathbf{c}}(+\infty)$, $\mathrm{MSD}_{\mathbf{h}}(+\infty)$ of the efficient NLMS and RLS algorithms.

	$\rho(m)=\mu$	$\rho(m)=\dfrac{1-\lambda^M}{1-\lambda^{M(m+1)}}$	$\rho(m)=\dfrac{1}{m+1}$
$\mathrm{MSE}(+\infty)$	$\dfrac{2}{2-\mu}\sigma_v^2$	$\dfrac{2}{1+\lambda^M}\sigma_v^2$	σ_v^2
$\mathrm{MSD}_{\mathbf{c}}(+\infty)$	$\dfrac{\mu}{2-\mu}M\sigma_v^2$	$\dfrac{1-\lambda^M}{1+\lambda^M}M\sigma_v^2$	0
$\mathrm{MSD}_{\mathbf{h}}(+\infty)$	$\dfrac{\mu}{2-\mu}\sigma_v^2\sum_{i=0}^{M-1}\lVert\mathbf{w}_i\rVert^2$	$\dfrac{1-\lambda^M}{1+\lambda^M}\sigma_v^2\sum_{i=0}^{M-1}\lVert\mathbf{w}_i\rVert^2$	0

Since, under our assumptions

$$\mathrm{MSD}_{\mathbf{h}}(n)=\sum_{i=0}^{M-1}E[\bar{c}_i^2(n)]\lVert\mathbf{w}_i\rVert^2, \tag{35}$$

we can iterate (34) from $n=mM$ to $n=(m+1)M-1$ to get

$$\begin{aligned}\mathrm{MSD}_{\mathbf{h}}[(m+1)M]&=\mathrm{MSD}_{\mathbf{h}}(mM)-[2\rho(m)-\rho^2(m)]\mathrm{MSD}_{\mathbf{h}}(mM)\\&\quad+\rho^2(m)\sigma_v^2\sum_{i=0}^{M-1}\lVert\mathbf{w}_i\rVert^2\\&=[1-\rho(m)]^2\mathrm{MSD}_{\mathbf{h}}(mM)\\&\quad+\rho^2(m)\sigma_v^2\sum_{i=0}^{M-1}\lVert\mathbf{w}_i\rVert^2.\end{aligned} \tag{36}$$

In the case of the NLMS algorithm with constant step-size μ, we can prove in a similar way that for all j

$$\mathrm{MSD}_{\mathbf{h}}[(m+1)M+j]=[1-\rho(m)]^2\mathrm{MSD}_{\mathbf{h}}(mM+j)+\rho^2(m)\sigma_v^2\sum_{i=0}^{M-1}\lVert\mathbf{w}_i\rVert^2, \tag{37}$$

where $\rho(m)$ is assumed equal to μ. This result holds approximately for the RLS algorithm also for $n\to+\infty$.

The expression of (37) differs from the linear case since the Euclidean norms of the vectors $\mathbf{w}_i$ are in general different from each other. The transient and steady-state properties of the algorithms depend only on the step-size $\rho(m)$ and the noise power σ_v^2. In particular, $\mathrm{MSE}(n)$ and $\mathrm{MSD}_{\mathbf{c}}(n)$ do not depend on the input periodic sequence. Only $\mathrm{MSD}_{\mathbf{h}}(n)$ depends on the choice of the periodic sequence through $\sum_{i=0}^{M-1}\lVert\mathbf{w}_i\rVert^2$. The algorithms are numerically stable and exponentially converge to the unknown system coefficients for $0<\epsilon<\rho(m)<2-\epsilon<2\ \forall m$ and for some small positive constant ϵ.

The steady-state values of $\mathrm{MSE}(n)$, $\mathrm{MSD}_{\mathbf{c}}(n)$, and $\mathrm{MSD}_{\mathbf{h}}(n)$ for $n\to+\infty$ are tabulated in Table 1 for the three cases of $\rho(m)$ with $m=\lfloor n/M\rfloor$.

4.2. Tracking analysis

Assume that we want to track an LIP nonlinear system of memory length N and M parameters, whose vector $\tilde{\mathbf{c}}(n)=[\tilde{c}_0(n),\ldots,\tilde{c}_{M-1}(n)]^T$ varies according to the random walk model [23],[2]

Table 2
$\mathrm{MSE}(+\infty)$, $\mathrm{MSD}_{\mathbf{c}}(+\infty)$, and $\mathrm{MSD}_{\mathbf{h}}(+\infty)$ of the NLMS and RLS algorithms for the random walk model.

$\rho(m)=\mu$	
$\mathrm{MSE}(+\infty)$	$\dfrac{1}{2\mu-\mu^2}M\sigma_q^2+\dfrac{2}{2-\mu}\sigma_v^2$
$\mathrm{MSD}_{\mathbf{c}}(+\infty)$	$\dfrac{1}{2\mu-\mu^2}\dfrac{M(M+1)}{2}\sigma_q^2+\dfrac{(1-\mu)^2}{2\mu-\mu^2}\dfrac{M(M-1)}{2}\sigma_q^2+\dfrac{\mu}{2-\mu}M\sigma_v^2$
$\mathrm{MSD}_{\mathbf{h}}(+\infty)$	$\mathrm{MSD}_{\mathbf{c}}(+\infty)\dfrac{1}{M}\sum_{i=0}^{M-1}\lVert\mathbf{w}_i\rVert^2$
$\rho(m)=\dfrac{1-\lambda^M}{1-\lambda^{M(m+1)}}$	
$\mathrm{MSE}(+\infty)$	$\dfrac{1}{1-\lambda^{2M}}M\sigma_q^2+\dfrac{2}{1+\lambda^M}\sigma_v^2$
$\mathrm{MSD}_{\mathbf{c}}(+\infty)$	$\dfrac{1}{1-\lambda^{2M}}\dfrac{M(M+1)}{2}\sigma_q^2+\dfrac{\lambda^{2M}}{1-\lambda^{2M}}\dfrac{M(M-1)}{2}\sigma_q^2+\dfrac{1-\lambda^M}{1+\lambda^M}M\sigma_v^2$
$\mathrm{MSD}_{\mathbf{h}}(+\infty)$	$\mathrm{MSD}_{\mathbf{c}}(+\infty)\dfrac{1}{M}\sum_{i=0}^{M-1}\lVert\mathbf{w}_i\rVert^2$

$$\tilde{c}_i(n+1)=\tilde{c}_i(n)+q_i(n), \tag{38}$$

for $0\le i<M$, where $q_i(n)$ are zero mean, independent, identically distributed sequences with $E[q_i^2(n)]=\sigma_q^2$, and independent from the noise $v(n)$.

By subtracting (38) from both sides of (20), we have

$$\bar{c}_i(n+1)=\begin{cases}\bar{c}_i(n)-q_i(n)+\rho\left(\left\lfloor\frac{n}{M}\right\rfloor\right)[v(n)-\bar{c}_i(n)] & \text{when } i=n \bmod M,\\ \bar{c}_i(n)-q_i(n) & \text{otherwise.}\end{cases} \tag{39}$$

By iteratively applying this equation from $n=mM$ to $n=(m+1)M-1$, it can be proved similarly as in [16] that

$$\begin{aligned}\mathrm{MSD}_{\mathbf{c}}[(m+1)M]&=[1-\rho(m)]^2\mathrm{MSD}_{\mathbf{c}}(mM)\\&\quad+\frac{M(M+1)}{2}\sigma_q^2+\frac{M(M-1)}{2}[1-\rho(m)]^2\sigma_q^2\\&\quad+\rho^2(m)M\sigma_v^2.\end{aligned} \tag{40}$$

The $\mathrm{MSE}(n)$ and $\mathrm{MSD}_{\mathbf{h}}(n)$ can be computed similarly.

Table 2 provides the steady-state values of $\mathrm{MSE}(n)$, $\mathrm{MSD}_{\mathbf{c}}(n)$, and $\mathrm{MSD}_{\mathbf{h}}(n)$ for the NLMS algorithm and the

[2] While many authors consider $\tilde{h}$ changing according to the random walk model, we prefer to apply the model to $\tilde{c}$ for the sake of simplicity. Note that in our model $\tilde{h}$ also changes according to a random walk model, but with correlated increments.

Author's personal copy

RLS algorithm with $\lambda \neq 1$. (The algorithm for $\lambda = 1$ is not suitable for tracking.)

4.3. *Design of periodic sequences for nonlinear system identification*

It is clear from the above analyses that the learning speed, the tracking speed and the asymptotic values of $\mathrm{MSE}(n)$ and of $\mathrm{MSD}_{\mathbf{c}}(n)$ are independent of the properties of the input sequence. Contrary to this, the asymptotic value of $\mathrm{MSD}_{\mathbf{h}}(n)$ is proportional to $\sum_{i=0}^{M-1} \|\mathbf{w}_i\|^2 = \|\mathbf{W}\|_F^2$, i.e. the Frobenius norm of matrix $\mathbf{W}$. When perfect periodic sequences are used for linear system identification, the Frobenius norm of the matrix $\mathbf{W}$ is

$$\|\mathbf{W}\|_F = M / \|\mathbf{X}_{\mathcal{F}}\|_F, \tag{41}$$

and thus it depends only on the power of the periodic sequence. In the nonlinear case, even for perfect periodic sequences with fixed power, $\|\mathbf{W}\|_F$ can vary considerably. Consequently it is important to design periodic sequences for which $\|\mathbf{W}\|_F$ is as low as possible.

Table 3 provides details of an *ad hoc* iterative dichotomic search algorithm that has been used to find periodic sequences with low $\|\mathbf{W}\|_F$. Starting from an initial random sequence with specified power, the algorithm chooses a random direction and moves in that direction until a segment (SeqL, SeqR) where there is a minimum of $\|\mathbf{W}\|_F$ is found. It then improves the minimum estimation by iteratively halving the segment (SeqL, SeqR). The procedure of choosing a random direction and optimizing the segment is iteratively repeated to optimize the sequence. While this algorithm provides only a suboptimal solution to the problem of finding the periodic sequence with the smallest $\|\mathbf{W}\|_F$ and could end up in local minima, the values of $\|\mathbf{W}\|_F$ obtained are much lower than those that can be found with a random choice of the periodic sequence or with perfect periodic sequences.

Table 3
Algorithm for $\|\mathbf{W}\|_F$ sequence optimization.

```
Normalize(): function that normalizes the sequence power
Start from a random sequence with unit power: SeqC
Compute ||W||_F(SeqC)
For jj=1: Number of Iterations
  Step=Step Initial
  Choose a random direction Dir
  Compute Seq=Normalize(SeqC ± Step * Dir)
    until three consecutive sequences: SeqL, SeqC, SeqR with
    ||W||_F(SeqL) ≤ ||W||_F(SeqC) ≤ ||W||_F(SeqR) are formed
  For ii=1: Number of Refinements
      Step=Step/2
      SeqCl=Normalize(SeqC−Step*Dir)
      SeqCr=Normalize(SeqC+Step*Dir)
      SeqCo=SeqC
      SeqC=min_{||W||_F} {SeqCL, SeqCo, SeqCR}
      If (SeqC==SeqCo)
        SeqL=SeqCl
        SeqR=SeqCR
      Elseif (SeqC==SeqCl)
        SeqR=SeqCo
      Else (SeqC==SeqCR)
        SeqL=SeqCo
      End If
  End For ii
End For jj
Output SeqC
```

5. Model mismatch analysis

Here we analyze the behavior of the algorithms when there is a model mismatch between the adaptive filter and the unknown nonlinear system. This analysis is performed under the simplifying assumption that there is no measurement noise, i.e., $\nu(n) = 0$. Under this condition, the NLMS algorithm with $\mu = 1$ and the RLS algorithm converge in just M samples. Since

$$\mathbf{c}(M) = [d(0), d(1), \ldots, d(M-1)]^T = \mathbf{d}, \tag{42}$$

we have from (10) that the adaptive filter converges to

$$\mathbf{h}(M) = \mathbf{W}\mathbf{d}. \tag{43}$$

We proceed by considering four different cases.

Case 1. *The system is perfectly matched*: In this case, it is straightforward to show that the algorithm is able to identify the unknown system.

Case 2. *The system is overdetermined*: In this case, the adaptive filter has more coefficients than needed. If $\tilde{h}_0, \ldots, \tilde{h}_{L-1}$ are the coefficients of the unknown model, with $L < M$, we have

$$d(i) = [\tilde{h}_0, \ldots, \tilde{h}_{L-1}, 0, \ldots, 0]\mathbf{x}_i, \tag{44}$$

$$\mathbf{d} = \mathbf{X}_{\mathcal{F}}^T [\tilde{h}_0, \ldots, \tilde{h}_{L-1}, 0, \ldots, 0]^T, \tag{45}$$

$$\mathbf{h}(M) = \mathbf{W}\mathbf{X}_{\mathcal{F}}^T [\tilde{h}_0, \ldots, \tilde{h}_L, 0, \ldots, 0]^T = [\tilde{h}_0, \ldots, \tilde{h}_L, 0, \ldots, 0]^T. \tag{46}$$

Thus, in the adaptive filter model some coefficients are redundant and converge to zero, but the algorithm is still able to identify the unknown system.

Case 3. *The system is underdetermined by memory*, i.e., some terms are missing in $\mathbf{x}_{\mathcal{F}}(n)$ and the missing terms are delayed version of those present in $\mathbf{x}_{\mathcal{F}}(n)$. For simplicity, let us assume without loss of generality that

$$d(n) = \tilde{\mathbf{h}}_1^T \mathbf{x}_{\mathcal{F}}(n) + \tilde{\mathbf{h}}_2^T \mathbf{x}_{\mathcal{F}}(n-1), \tag{47}$$

where some of the coefficients of $\tilde{\mathbf{h}}_2$ could be zero. Then

$$d(i) = \tilde{\mathbf{h}}_1^T \mathbf{x}_i(n) + \tilde{\mathbf{h}}_2^T \mathbf{x}_{i-1}(n), \tag{48}$$

$$\mathbf{d} = \mathbf{X}_{\mathcal{F}}^T \tilde{\mathbf{h}}_1 + \mathbf{X}_{\mathcal{F}}^T \mathbf{U} \tilde{\mathbf{h}}_2, \tag{49}$$

where $\mathbf{U}$ is a permutation matrix that rotate upward by one position the rows of $\mathbf{X}_{\mathcal{F}}^T(n)$. In these conditions, the adaptive filter converges to

$$\mathbf{h}(M) = \mathbf{W}\mathbf{X}_{\mathcal{F}}^T(n)\tilde{\mathbf{h}}_1 + \mathbf{W}\mathbf{X}_{\mathcal{F}}^T(n)\mathbf{U}\tilde{\mathbf{h}}_2 = \tilde{\mathbf{h}}_1 + \mathbf{U}\tilde{\mathbf{h}}_2. \tag{50}$$

Thus, the unknown system estimation is affected by an aliasing error that depends on $\tilde{\mathbf{h}}_2$.

Case 4. *The system is underdetermined by order*, i.e., in the input data vector $\mathbf{x}_{\mathcal{F}}(n)$ some terms are missing and the missing terms are not delayed versions of those present in

Author's personal copy

$\mathbf{x}_{\mathcal{F}}(n)$. Let us assume that

$$d(n) = \tilde{\mathbf{h}}_1^T \mathbf{x}_{\mathcal{F}}(n) + \tilde{\mathbf{h}}_2^T \mathbf{x}_{\mathcal{E}}(n), \quad (51)$$

with $\mathbf{x}_{\mathcal{E}}(n)$ being a nonlinear vector function of the input samples $x(n), \ldots, x(n-N+1)$. Since $\mathbf{X}_{\mathcal{F}}$ is invertible, by introducing the matrix $\mathbf{V}$ such that

$$\mathbf{X}_{\mathcal{F}}^T \mathbf{V} = [\mathbf{x}_{\mathcal{E}}(0), \ldots, \mathbf{x}_{\mathcal{E}}(M-1)]^T, \quad (52)$$

we have

$$\mathbf{d} = \mathbf{X}_{\mathcal{F}}^T \tilde{\mathbf{h}}_1 + \mathbf{X}_{\mathcal{F}}^T \mathbf{V} \tilde{\mathbf{h}}_2. \quad (53)$$

Thus, the adaptive filters converge to

$$\mathbf{h}(M) = \mathbf{W}\mathbf{X}_{\mathcal{F}}^T(n)\tilde{\mathbf{h}}_1 + \mathbf{W}\mathbf{X}_{\mathcal{F}}^T(n)\mathbf{V}\tilde{\mathbf{h}}_2 \quad (54)$$

$$\mathbf{h}(M) = \tilde{\mathbf{h}}_1 + \mathbf{V}\tilde{\mathbf{h}}_2. \quad (55)$$

In this case also we have an aliasing error that depends on $\tilde{\mathbf{h}}_2$. We refer to it as an aliasing error even though the matrix $\mathbf{V}$ spreads the coefficients of $\tilde{\mathbf{h}}_2$ over all the coefficients of the adaptive filter.

The results presented in this section are interesting not only from a theoretical point of view, but also useful because they determine which adaptive filter coefficients are affected by an underestimated model.

6. Simulation results

6.1. *Identification of nonlinear systems*

In this subsection we provide some simulation results for the identification of a second order FLANN filter of memory length 50 samples, a second order truncated Volterra filter of memory length 20 samples, and a second order extended Volterra filter of memory length 5 samples without the constant term.[3] All filters have approximately the same number of coefficients, i.e. 250 for the FLANN filter, 230 for the truncated Volterra filter, and 242 for the extended Volterra filter. The input signal was designed using the algorithm of Table 3, with period equal to the number of coefficients of the unknown system. Table 4 provides the condition number in the 2-norm and in the Frobenius norm of $\mathbf{X}_{\mathcal{F}}$, and the squared Frobenius norm of $\mathbf{W}$ of the three periodic sequences. The coefficients of the system to be identified were randomly generated using the additional constraint that the output power of the nonlinear part of the system was 10 dB below the output power of the linear part. The adaptive filter was perfectly matched to the unknown nonlinear system.

To compare the capabilities of the algorithms of this paper with the traditional NLMS algorithm, we first computed the ensemble averages over 200 simulations, shown in Figs. 2–4, of the MSE(n), and the MSD$_{\mathbf{h}}(n)$ for different output signal SNRs of the FLANN filter, the truncated Volterra filter, and the extended Volterra filter, respectively, for the standard NLMS algorithm with a unit power white noise input and with step-size $\mu = 0.2$. In this algorithm the filter was adapted as follows:

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu \frac{d(n) - y(n)}{\mathbf{x}_{\mathcal{F}}^T(n)\mathbf{x}_{\mathcal{F}}(n) + \delta} \mathbf{x}_{\mathcal{F}}(n), \quad (56)$$

with δ being a small positive constant used to avoid instabilities. Then, using the equations of MSD$_{\mathbf{h}}(\infty)$ of Table 1, we estimated parameters of the algorithms in (17) and (20), that provide the same MSD$_{\mathbf{h}}(\infty)$ as the NLMS algorithm in (56) for $\mu = 0.2$, obtaining $\mu = 1 - \lambda^M = 0.1116$ for the FLANN filter, 0.0675 for the truncated Volterra filter, and 0.0445 for the extended Volterra filter, respectively. These parameters were used in the simulations employing periodic signals.

Figs. 5–7 display the ensemble averages over 200 simulations of MSE(n) (plots (a) and (d)), MSD$_{\mathbf{c}}(n)$ (plots (b) and (e)), and MSD$_{\mathbf{h}}(n)$ (plots (c) and (f)) for different output signal SNRs of the FLANN filter, the truncated Volterra filter, the extended Volterra filter, respectively. Plots (a)–(c) show the results when the filter was adapted with the NLMS algorithm in (17) and plots (c)–(e) display the corresponding results for the filter adapted with the RLS algorithm in (20). In the figures, the dashed lines represent the theoretical values of MSE$(+\infty)$, MSD$_{\mathbf{c}}(+\infty)$, or MSD$_{\mathbf{h}}(+\infty)$ as appropriate, and as given in Table 1. In all simulations, the theoretical values of MSE$(+\infty)$, MSD$_{\mathbf{c}}(+\infty)$, and MSD$_{\mathbf{h}}(+\infty)$ of Table 1 match the results of the simulations accurately. As predicted in Section 4, the RLS algorithm in (20) with $\lambda < 1$ provides the same asymptotic values as the NLMS algorithm of (17) and with $\mu = 1 - \lambda^M$. Given the faster convergence speed, the RLS algorithm should be preferred for system identification.

Comparing Figs. 2–7, we can see that differently from the linear case, for the same value of MSD$_{\mathbf{h}}(\infty)$ the NLMS algorithm in (17) can have a learning speed that is lower than the NLMS algorithm in (56) with a white noise input. This is the case for the FLANN filter and the truncated Volterra filter. The learning speed could be improved by finding periodic sequences with lower $\|\mathbf{W}\|_F^2$. On the other hand, for the extended Volterra filter, the NLMS algorithm in (56) has a lower convergence speed and a larger excess MSE than the NLMS algorithm in (17), so that it is almost impossible to distinguish the different learning curves of Fig. 4(a). To understand the good performance for the filter of this paper in this example, it is worth noting that the parameters in Table 4 depend on the filter complexity, and thus their values are specific of each system structure. In spite of the fact that the value of the squared Frobenius norm of $\mathbf{W}$ is greater than for the other two filters, the designed periodic sequence is already sufficiently good in this case.

Table 4
Condition number in the 2-norm and in the Frobenius norm of $\mathbf{X}_{\mathcal{F}}$, and squared Frobenius norm of $\mathbf{W}$ for the input sequences.

Filter	Cond$_2(\mathbf{X}_{\mathcal{F}})$	Cond$_F(\mathbf{X}_{\mathcal{F}})$	$\|\mathbf{W}\|_F^2$
FLANN	7.03	3.44×10^2	3.15
Truncated Volterra	28.9	4.72×10^2	3.97
Extended Volterra	1.76×10^2	2.06×10^3	56.7

[3] This extended Volterra filter is also a particular case of truncated Volterra filter of the 10th order.

Author's personal copy

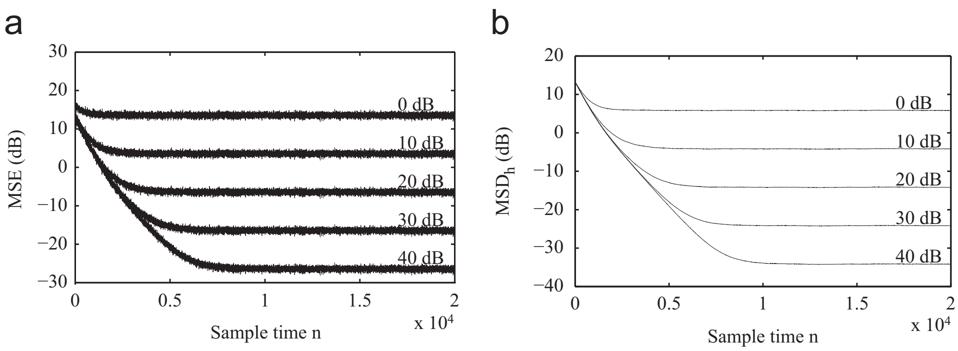

Fig. 2. Second order FLANN filter: evolution of (a) MSE(n), (b) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the NLMS algorithm in (56) for $\mu = 0.2$, for different output SNRs.

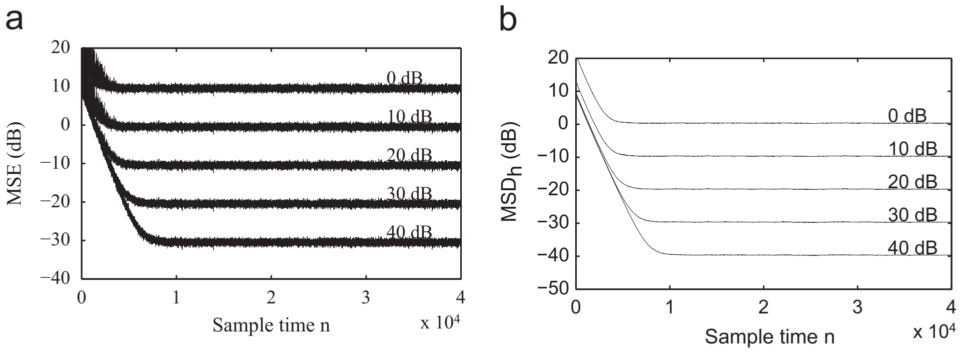

Fig. 3. Second order truncated Volterra filter: evolution of (a) MSE(n), (b) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the NLMS algorithm in (56) for $\mu = 0.2$, for different output SNRs.

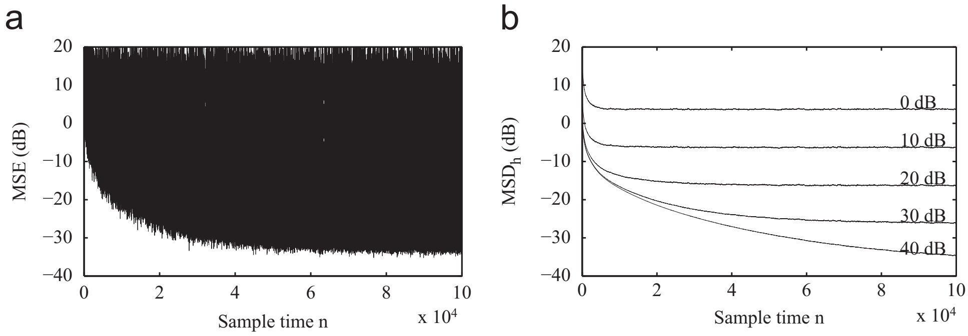

Fig. 4. Second order extended Volterra filter: evolution of (a) MSE(n), (b) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the NLMS algorithm in (56) for $\mu = 0.2$, for different output SNRs.

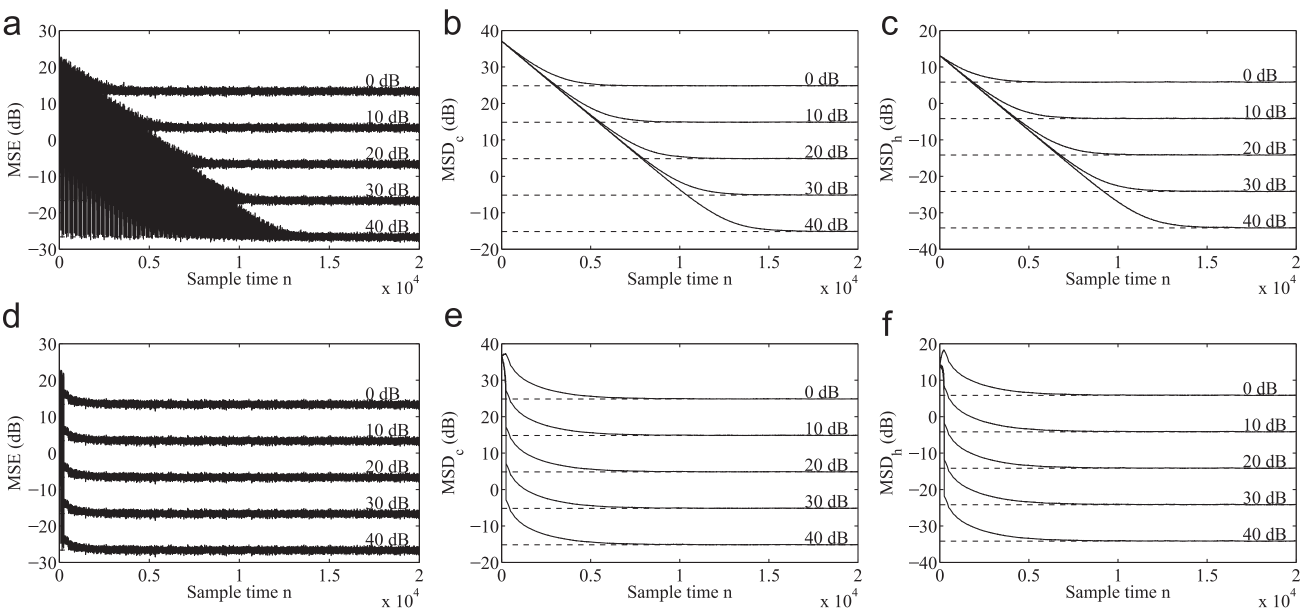

Fig. 5. Second order FLANN filter: evolution of (a) MSE(n), (b) $\mathrm{MSD}_{\mathbf{c}}(n)$, (c) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the NLMS algorithm in (17) for $\mu = 0.1116$, (d) MSE(n), (e) $\mathrm{MSD}_{\mathbf{c}}(n)$, (f) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the RLS algorithm in (20) for $\lambda^M = 1 - 0.1116$, for different output SNRs.

Author's personal copy

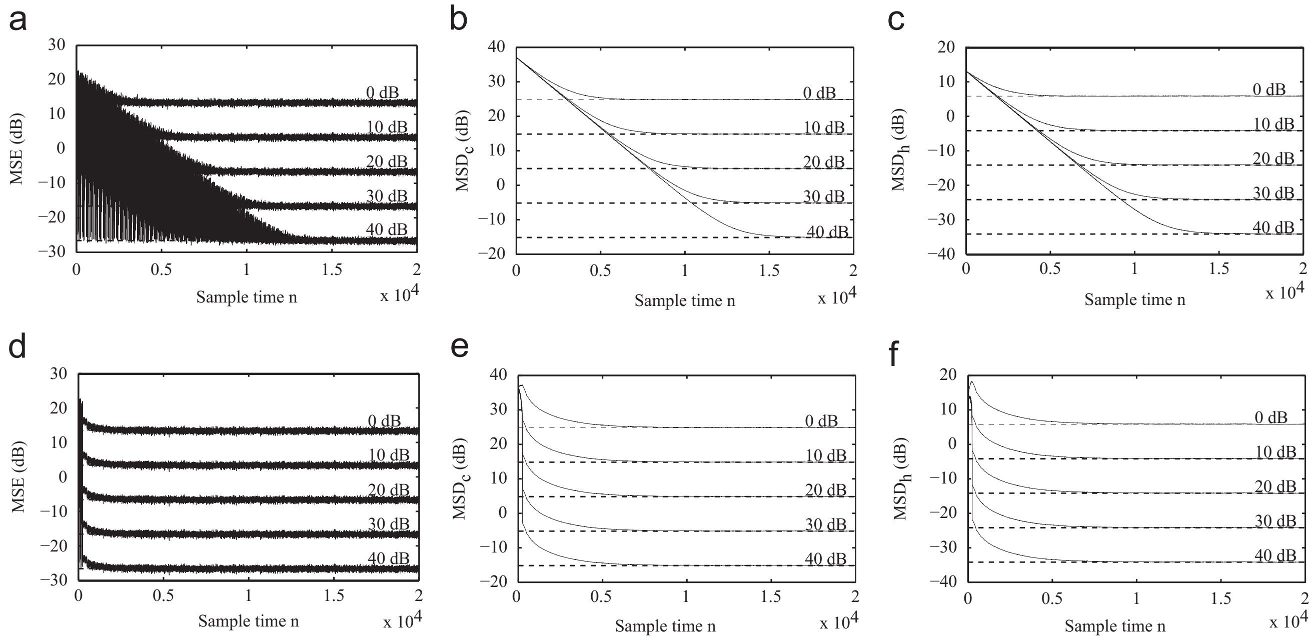

Fig. 6. Second order truncated Volterra filter: evolution of (a) MSE(n), (b) $\mathrm{MSD}_{\mathbf{c}}(n)$, (c) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the NLMS algorithm in (17) for $\mu = 0.0675$, (d) MSE(n), (e) $\mathrm{MSD}_{\mathbf{c}}(n)$, (f) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the RLS algorithm in (20) for $\lambda^M = 1-0.0675$, for different output SNRs.

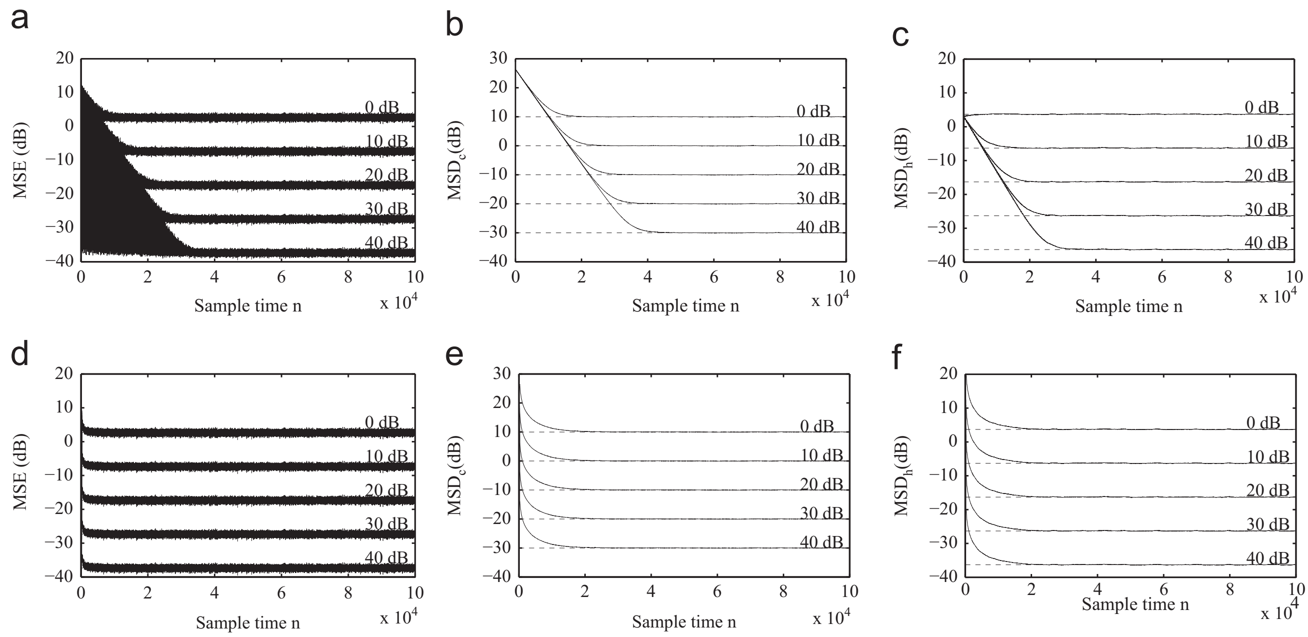

Fig. 7. Second order extended Volterra filter: evolution of (a) MSE(n), (b) $\mathrm{MSD}_{\mathbf{c}}(n)$, (c) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the NLMS algorithm in (17) for $\mu = 0.0445$, (d) MSE(n), (e) $\mathrm{MSD}_{\mathbf{c}}(n)$, (f) $\mathrm{MSD}_{\mathbf{h}}(n)$, of the RLS algorithm in (20) for $\lambda^M = 1-0.0445$, for different output SNRs.

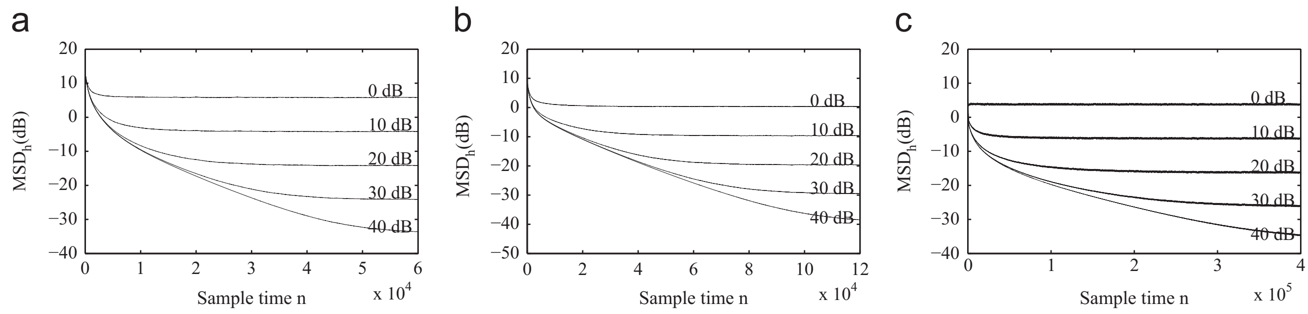

Fig. 8. Learning curves of $\mathrm{MSD}_{\mathbf{h}}(n)$ of the NLMS algorithm in (56) for (a) the FLANN filter with $\mu = 0.2$, (b) the second order Volterra filter with $\mu = 0.25$ (c) the extended Volterra filter with $\mu = 0.6$, with a periodic input sequence for different output SNRs.

Author's personal copy

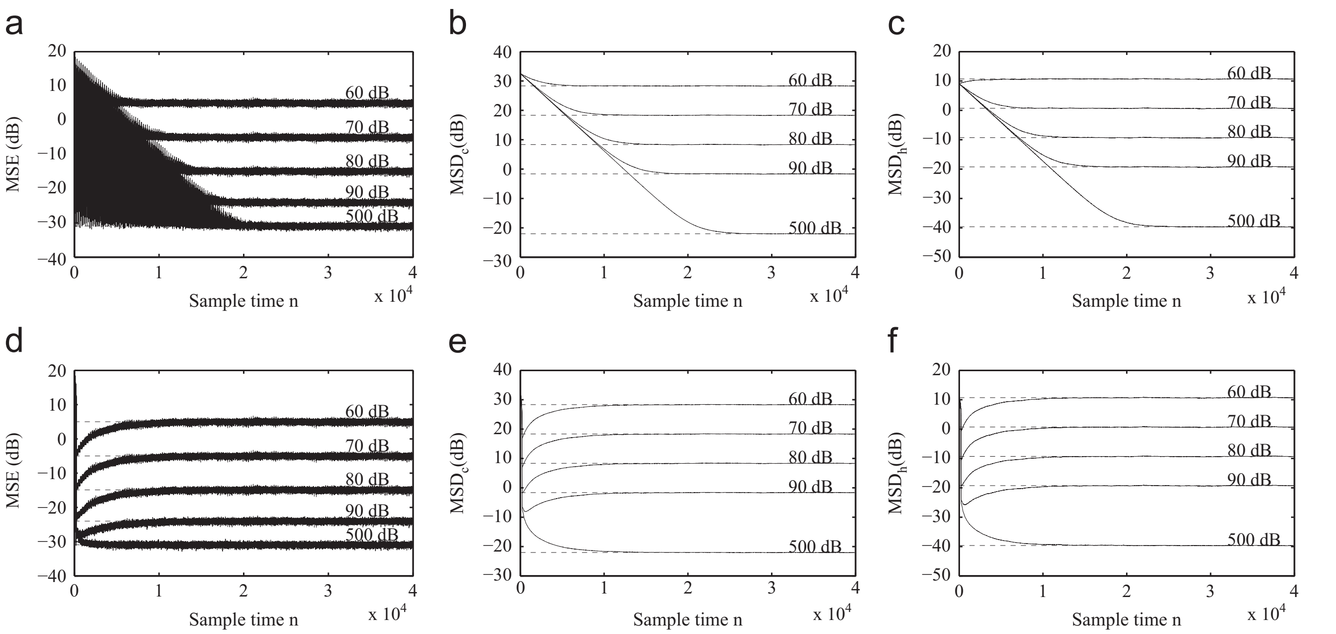

Fig. 9. Second order truncated Volterra filter: evolution of (a) MSE(n), (b) MSD$_\mathbf{c}$(n), (c) MSD$_\mathbf{h}$(n), of the NLMS algorithm in (17), (d) MSE(n), (e) MSD$_\mathbf{c}$(n), (f) MSD$_\mathbf{h}$(n), of the RLS algorithm in (20), for different ratios $\sigma_q^2/\|\tilde{c}(0)\|^2$.

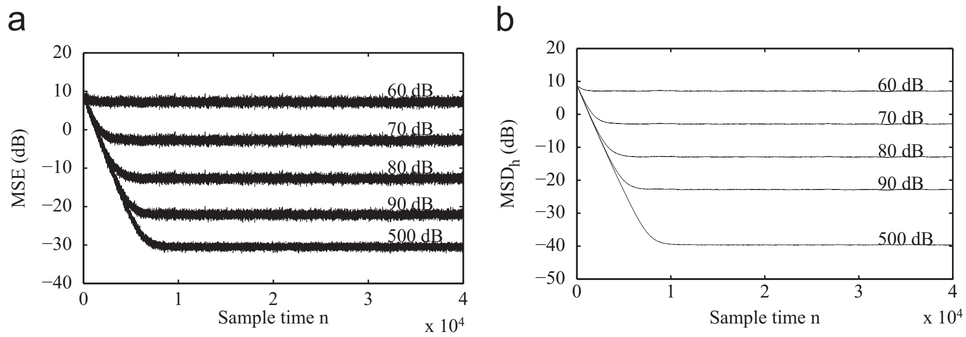

Fig. 10. Second order truncated Volterra filter: evolution of (a) MSE(n), (b) MSD$_\mathbf{h}$(n), of the NLMS algorithm in (56), for a 40 dB output SNR, for different ratios $\sigma_q^2/\|\tilde{c}(0)\|^2$.

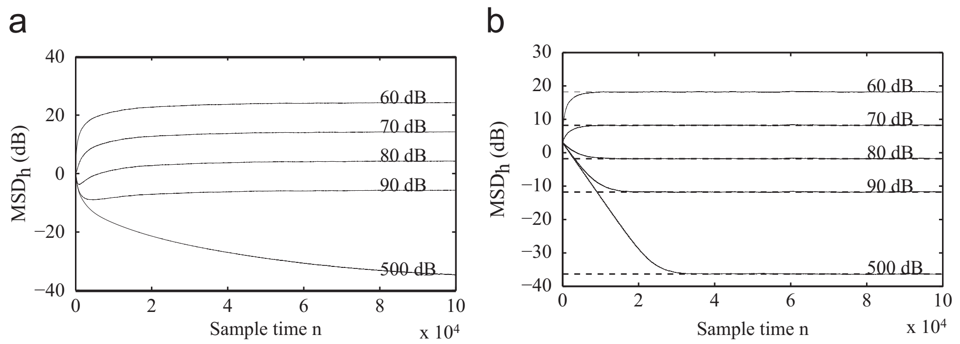

Fig. 11. Extended Volterra filter: learning curve of MSD$_\mathbf{h}$(n) of (a) the NLMS algorithm in (56) and (b) the NLMS algorithm in (17) for a 40 dB output SNR, for different ratios $\sigma_q^2/\|\tilde{c}(0)\|^2$.

Finally, Fig. 8 shows the learning curves of the MSD$_\mathbf{h}$(n) of the NLMS algorithm in (56) for the three nonlinear filters and for different output SNRs with the same periodic input sequence of Figs. 5–7. In Fig. 8, the step-sizes have been chosen by trial and error to give the same MSD$_\mathbf{h}$(∞) of Figs. 2–7. From Fig. 8 we can notice that for the same input periodic sequence the proposed algorithms provide a much faster convergence speed than the NLMS algorithm in (56).

6.2. Tracking nonlinear systems

In the second set of simulations we deal with tracking a nonlinear time-varying system that varies according to

Author's personal copy

the random walk of (38), starting from the same model coefficients of the first set of simulations. We consider a 40 dB output SNR and we repeat the simulations for the second order truncated Volterra filter using the same parameters as in Section 6.1.

Fig. 9 displays the ensemble averages over 200 simulations of MSE(n) (plots (a) and (d)), $\mathrm{MSD}_{\mathbf{c}}(n)$ (plots (b) and (e)), and $\mathrm{MSD}_{\mathbf{h}}(n)$ (plots (c) and (f)) for different choices of $\sigma_q^2/\|\tilde{\mathbf{c}}(0)\|^2$. Plots (a)–(c) show the results when the filter is adapted with the NLMS algorithm in (17). Plots (d)–(f) display the corresponding results for the RLS algorithm in (20) with $1-\lambda^N$ equal to the step-size of the NLMS algorithm. In the figures, the dashed lines represent the theoretical values of $\mathrm{MSE}(+\infty)$, $\mathrm{MSD}_{\mathbf{c}}(+\infty)$, or $\mathrm{MSD}_{\mathbf{h}}(+\infty)$ as given in Table 2. In all simulations, the theoretical values of $\mathrm{MSE}(+\infty)$, $\mathrm{MSD}_{\mathbf{c}}(+\infty)$, and $\mathrm{MSD}_{\mathbf{h}}(+\infty)$ of Table 2 match the results of the simulations accurately. As predicted, the NLMS algorithm in (17) and the RLS algorithm in (20) with $1-\lambda^N=\mu$ have the same asymptotic tracking properties. Similar results were obtained with the FLANN filter and the extended Volterra filter of the first set of simulations.

For comparison purposes Fig. 10 displays the ensemble averages over 200 simulations of MSE(n) (plot (a)), and $\mathrm{MSD}_{\mathbf{h}}(n)$ (plot (b)) for different ratios $\sigma_q^2/\|\tilde{\mathbf{c}}(0)\|^2$ for the standard NLMS algorithm of (56) with step-size $\mu=0.2$ and a unit power white noise input. Comparing Figs. 9 and 10 we see that the tracking behavior of the NLMS algorithm in (17) is slightly poorer than the NLMS algorithm in (56) with a white noise input. The same behavior was observed in our simulations with the FLANN filter. Even in this case, a much better tracking behavior was observed for the extended Volterra filter, as shown in Fig. 11, which compares the $\mathrm{MSD}_{\mathbf{h}}(n)$ of the standard NLMS algorithm in (56) and of the NLMS algorithm in (17) for different ratios $\sigma_q^2/\|\tilde{\mathbf{c}}(0)\|^2$ and a unit power white noise input.

7. Concluding remarks

This paper discussed computationally efficient, exact NLMS and RLS algorithms for identifying and tracking a class of nonlinear systems that are linear in their parameters. The algorithms require a computational effort of just a multiplication, an addition and a subtraction per input sample. We analyzed the transient, steady-state, and tracking behavior of the algorithms and derived exact expressions for the steady-state values of the MSE(n), $\mathrm{MSD}_{\mathbf{c}}(n)$, and $\mathrm{MSD}_{\mathbf{h}}(n)$. We also studied the behavior of the algorithms in the case of a model mismatch. The simulation results presented in the paper demonstrate the validity of the algorithms as well as the accuracy of the theoretical results derived in the paper. The low computational complexity, the good performance, and the generality of the algorithms make the approach of this paper a valid alternative to current techniques for identification of a large class of nonlinear systems.

References

[1] G.L. Sicuranza, A. Carini, On a class of nonlinear filters, in: M.G.I. Tabus, K. Egiazarian (Eds.), Festschrift in Honor of Jaakko Astola on the Occasion of his 60th Birthday, Vol. TICSP Report #47, 2009, pp. 115–144.

[2] M. Schetzen, The Volterra and Wiener Theories of Nonlinear Systems, Krieger Publishing Company, Malabar, FL, 2006.

[3] V.J. Mathews, G.L. Sicuranza, Polynomial Signal Processing, Wiley, New York, 2000.

[4] P. Nowak, B.V. Veen, Random and pseudorandom inputs for Volterra filter identification, IEEE Transactions on Signal Processing 32 (1994) 2124–2135.

[5] Y.H. Pao, Adaptive Pattern Recognition and Neural Networks, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.

[6] J.C. Patra, R.N. Pal, A functional link artificial neural network for adaptive channel equalization, Signal Processing 43 (1995) 181–195.

[7] D.P. Das, G. Panda, Active mitigation of nonlinear noise processes using a novel filtered-s LMS algorithm, IEEE Transactions on Speech and Audio Processing 12 (3) (2004) 313–322.

[8] G.L. Sicuranza, A. Carini, On the accuracy of generalized Hammerstein models for nonlinear active noise control, in: Proceedings of the 2006 IEEE Instrumentation and Measurement Technology Conference, Sorrento, Italy, 2006, pp. 1411–1416.

[9] G.L. Sicuranza, A. Carini, Piecewise-linear expansions for nonlinear active noise control, in: Proceedings of the ICASSP 2006, International Conference on Acoustic, Speech, Signal Processing, Toulouse, France, 2006, pp. 209–212.

[10] M.J.D. Powell, Radial basis functions for multi-variable interpolation: a review, in: Proceedings of the IMA Conference on Algorithms for the Approximation of Functions and Data, RMCS Shrivenham, UK, 1985, pp. 143–167.

[11] T. Poggio, F. Girosi, Regularization algorithms for learning that are equivalent to multilayer networks, Science, New Series 247 (4945) (1990) 978–982.

[12] E.L.O. Batista, O.J. Tobias, R. Seara, A fully LMS adaptive interpolated Volterra structure, in: Proceedings of the ICASSP 2008, International Conference on Acoustic, Speech, Signal Processing, Las Vegas, NV, 2008, pp. 3613–3616.

[13] E.L.O. Batista, O.J. Tobias, R. Seara, A sparse-interpolated scheme for implementing adaptive Volterra filters, IEEE Transactions on Signal Processing 58 (4) (2010) 2022–2035.

[14] E.L.O. Batista, R. Seara, A fully LMS/NLMS adaptive scheme applied to sparse-interpolated Volterra filters with removed boundary effect, Signal Processing 92 (10) (2012) 2381–2393.

[15] D.R. Morgan, Z. Ma, J. Kim, M.G. Zierdt, J. Pastalan, A generalized memory polynomial model for digital predistortion of RF power amplifiers, IEEE Transactions on Signal Processing 54 (2006) 3852–3860.

[16] A. Carini, Efficient NLMS and RLS algorithms for perfect and imperfect periodic sequences, IEEE Transactions on Signal Processing 58 (4) (2010) 2048–2059.

[17] C. Antweiler, M. Dörbecker, Perfect sequence excitation of the NLMS algorithm and its application to acoustic echo control, Annales des Telecommunications 49 (7–8) (1994) 386–397.

[18] C. Antweiler, M. Antweiler, System identification with perfect sequences based on the NLMS algorithm, International Journal of Electronics and Communications (AEU) 49 (3) (1995) 129–134.

[19] C. Antweiler, Multi-channel system identification with perfect sequences, in: U.H.R. Martin, C. Antweiler (Eds.), Advances in Digital Speech Transmission, SPIE, John Wiley & Sons, 2008, pp. 171–198.

[20] A. Carini, V.J. Mathews, G.L. Sicuranza, Efficient NLMS and RLS algorithms for a class of nonlinear filters using periodic input sequences, in: Proceedings of the ICASSP 2011, International Conference on Acoustic, Speech, Signal Processing, Prague, Czech Republic, 2011.

[21] M.J. Reed, M.O.J. Hawksford, Identification of discrete Volterra series using maximum length sequences, IEE Proceedings Circuits, Devices & Systems 143 (5) (1996) 241–248.

[22] K. Ozeki, T. Umeda, An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties, Electronics and Communications in Japan 67-A (1984) 126–132.

[23] A.H. Sayed, Fundamentals of Adaptive Filtering, Wiley, New York, 2003.