A dual filtering scheme for nonlinear active noise control

D. Delvecchio and L. Piroddi*,†

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Via Ponzio 34/5, 20133, Milano, Italy

Received 6 November 2012; Revised 27 February 2013; Accepted 28 October 2013

1. INTRODUCTION

Active noise control (ANC) (e.g., [1–4]) is concerned with the reduction of undesired acoustic noise through the application of secondary acoustic sources designed to cancel the noise by destructive interference. Both adaptive feedforward and feedback techniques have been proposed for this purpose, the former solution being often preferred due to the inherently time-varying characteristics and the delays involved in the sound transmission dynamics [1].

In the adaptive feedforward setting a reference signal, well correlated with the noise source, is suitably filtered by the controller to provide the driving signal for the secondary acoustic source, in order to maximize the noise reduction at a given measurement location. The system dynamics seen by the controller is generally denoted secondary path (SP), as opposed to the primary acoustic path (PP) that carries the offending noise. The controller is generally implemented as an adaptive filter, whose parameters (weights) are tuned by means of an adaptive algorithm of the least mean squares (LMS) family. Two widely used adaptive algorithms for this purpose are the filtered-x LMS (FXLMS) and the filtered-u LMS (FULMS) algorithms [1]. These schemes are particularly appealing in their simplicity, because the gradient of the error with respect to the filter parameters can be simply calculated by a filtering operation.

Several sources of nonlinearity may affect the described system, ranging from the noise characteristics [5] to the dynamics of the involved acoustic paths [6]. Distortion and saturation of microphones, amplifiers, loudspeakers and converters are commonly experienced in practice [7].

*Correspondence to: L. Piroddi, Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Via Ponzio 34/5, 20133, Milano, Italy.

†E-mail: luigi.piroddi@polimi.it

Specific models and algorithms must be employed to properly account for these nonlinearities. In this respect, several nonlinear ANC (NANC) approaches have been recently introduced in the literature that use different types of nonlinear control filters [7–23]. Most of these methods rely on the linear SP assumption, which allows a direct extension of the basic FXLMS and FULMS schemes to the nonlinear case [10, 14, 15, 17–19, 24]. The more general case where the assumption of linearity of the SP is relaxed poses a much harder challenge [20, 22, 23] for two main reasons:

1. The minimization of the squared instantaneous error as in the LMS-type methods requires the calculation of the error gradient with respect to the filter weights. The latter depends in a complex way on the input/output gradient of the SP, and its calculation requires the application of nonlinear recursive filtering operations, whose stability cannot be guaranteed, let alone the computational complexity [23]. Crude simplifications (Feintuch's assumption, [25]) are generally adopted to solve this issue, but they affect the accuracy and performance of the approach.
2. The introduction of a model structure selection procedure, as commonly carried out in nonlinear model identification, is crucial in NANC applications, because the number of parameters of a nonlinear model rapidly increases with its flexibility (curse of dimensionality), posing unrealistic demands on the ANC system in terms of computational load and memory requirements. Besides, it is well known that model overparametrization is responsible for a host of undesired effects, such as overfitting, parameter fluctuation, poor model generalization capabilities, local minima and even model instability (e.g., [26–28]). The potential benefits of model structure selection for NANC applications have been emphasized in [23]. Unfortunately, the indirect model identification setting of the NANC problem, due to the presence of the SP between the filter to be estimated and the available error signal, prevents the inclusion in the adaptation scheme of online model selection algorithms already available in the literature, which are based on a direct model identification setting.

To address both these issues, an alternative NANC scheme is proposed in this work that operates in the general setting where the SP is assumed to be nonlinear, but configures the control filter adaptation in a direct identification mode. The proposed approach is inspired by the modified FXLMS scheme described in [29], where the control filter is adapted *as if* it were commuted with the SP, and a compensating scheme is included to account for the commutation error (see also [30, 31]). The latter is implemented as a secondary adaptation scheme, which identifies a modified SP to be used to filter the reference signal in the primary adaptation loop. Accordingly, the novel NANC scheme is denoted dual filtering LMS (DFLMS).

Thanks to the reformulation of the control filter adaptation problem as a direct identification one, the DFLMS approach seamlessly allows the application of model selection algorithms to adjust the structure of the filter model online. One online model selection algorithm recently introduced in [32] has been tested for this purpose. Some simulation examples are provided to show the effectiveness of the proposed NANC scheme.

The paper is organized as follows. Section 2 gives a model identification-oriented interpretation of the ANC problem and clarifies the role of the SP. Section 3 briefly reviews NANC solutions found in the literature. Section 4 discusses the DFLMS, providing also a comparison with the NFGLMS in terms of computational complexity. Section 5 is devoted to the extension of DFLMS with online model selection algorithms. The proposed algorithms are then tested and compared with alternative techniques in Section 6. Some conclusions are drawn in Section 7.

2. INTERPRETATION OF ANC AS A MODEL IDENTIFICATION PROBLEM

The generic feedforward ANC setting is represented in Figure 1, where the noise signal is modeled as a filtered version of the (measurable) reference signal $x(k)$ through the (unknown) PP P, essentially representing the acoustic path from the reference measurement to the error measurement. A secondary acoustic source is suitably driven by a controller C to cancel the noise at the error measurement location. Block S accounts for the SP, that is, the measurement and control chain, as well

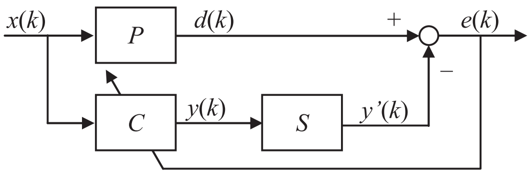

Figure 1. Block diagram of an adaptive feedforward active noise control system, where P, S and C denote the primary acoustic path, secondary path and control filter, respectively, and x, d and e are the reference, noise and error signals, respectively.

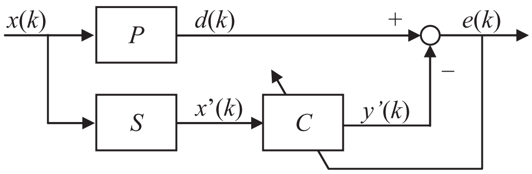

Figure 2. Active noise control scheme with S and C commuted.

as the acoustic path from the secondary source to the error measurement. The error is defined as

$$e(k) = d(k) - y'(k),$$

where $y'(k)$ denotes the filtered version of the control variable $y(k)$ through the SP. The controller is implemented as a parametric filter, whose coefficients (or weights) must be tuned to minimize the error.

The general ANC scheme of Figure 1 can be essentially interpreted as a model identification problem, where the objective is to find a model for the control filter C such that the series between C and S behaves similar to P to the extent possible given the excitation characteristics of the reference signal $x(k)$. The identification problem is *indirect*, in that the target output for model C is not accessible, and the only information available is not directly related to the output of C, but to a filtered version of it, through S.

In a purely linear framework, and assuming that the adaptation process is sufficiently slow, the scheme of Figure 1 is equivalent to that represented in Figure 2, where blocks C and S have been exchanged. This suggests that C can actually be identified in a direct fashion *as if* it had the filtered reference signal $x'(k)$ as input and $y'(k)$ as output.

This is the rationale behind the FXLMS and FULMS algorithms, where the block commutation results in the filtering of the reference signal (and the controller output in the FULMS case), [1]. More precisely, assume a finite impulse response (FIR) control filter structure:

$$y(k) = \sum_{m=0}^{L} w_m(k)x(k-m), \tag{1}$$

which can synthetically be represented in the general linear regression form:

$$y(k) = \boldsymbol{x}(k)^T \boldsymbol{w}(k)^{\ddagger}, \tag{2}$$

where $\boldsymbol{x}(k) = [x(k)\ \ x(k-1)\ \ \ldots\ \ x(k-L)]^T$ is the regressor vector, and $\boldsymbol{w}(k) = [w_0(k)\ \ w_1(k)\ldots w_L(k)]^T$ is the parameter vector at the kth iteration.

By using the stochastic gradient approach [1], the parameter update equation is derived as follows:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \frac{\mu}{2}\frac{\partial e(k)^2}{\partial \boldsymbol{w}(k)}^T = \boldsymbol{w}(k) + \mu \frac{\partial y'(k)}{\partial \boldsymbol{w}(k)}^T e(k). \tag{3}$$

‡Vector quantities are indicated with boldface lower case letters.

Given that $y'(k) = s(k) * (\boldsymbol{w}(k)^T \boldsymbol{x}(k))$, where $s(k)$ is the SP impulse response and '$*$' denotes the convolution operator, the gradient appearing in expression (3) is approximated as

$$\frac{\partial y'(k)}{\partial \boldsymbol{w}(k)} \cong s(k) * \boldsymbol{x}(k) = \boldsymbol{x}'(k),$$

although this equation rigorously holds only for constant $\boldsymbol{w}$ (slow adaptation hypothesis). Overall, the FXLMS weight update equation becomes

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \mu \boldsymbol{x}'(k) e(k). \tag{4}$$

The same rationale is applied in the case of a control filter with infinite impulse response structure, that is,

$$y(k) = \sum_{i=0}^{L} a_i(k) x(k-i) + \sum_{i=0}^{L} b_j(k) y(k-j), \tag{5}$$

and the corresponding FULMS weight update equation is formally equal to (4), with $\boldsymbol{w}(k) = [a_0(k) \ldots a_L(k) \quad b_1(k) \ldots b_L(k)]^T$ and $\boldsymbol{x}(k) = [x(k) \ \ldots \ x(k-L) \ y(k-1) \ \ldots \ y(k-L)]^T$. A simplification is applied to obtain the FULMS weight update rule, that is, the elimination of the recursive terms in the computation of the error gradient, based on the so-called Feintuch's assumption [25].

These algorithms have been shown to converge surprisingly well in practice even if the slow adaptation assumption does not hold. Some researchers have also examined in more detail the role of the commutation error and have proposed variations of the standard FXLMS scheme that take it explicitly into account (e.g., [30, 31]).

The commutability of blocks C or S is not possible in the general nonlinear case, and the previous algorithms cannot be extended as such unless the mentioned blocks possess specific structures, as discussed in the next section.

3. NONLINEAR ACTIVE NOISE CONTROL METHODS

Nonlinear ANC methods are characterized by the use of a nonlinear control filter structure in the ANC scheme.

3.1. Models and methods for NANC in the linear SP case

Truncated quadratic Volterra series [9–12], functional link artificial neural networks (FLANN) with trigonometric [11, 14–17], or piecewise linear functional expansions [18], general function expansion nonlinear filters [20], are examples of non-recursive nonlinear models that have been used in the NANC context. As the FIR filter they are structurally BIBO stable. They all have a linear-in-the-parameters structure and can be recast in the general linear regression form (2), where the elements of $\boldsymbol{x}(k)$ are nonlinear functions of the recent samples of the reference signal.

The dynamics of the model can be essentially separated in a nonlinear input block followed by a linear block, and the latter can be commuted with the (linear) SP, whereas the former takes care of the signal filtering. As a result, the adaptation methods that have been proposed for these models, that is, the Volterra FXLMS (VFXLMS) [9, 10], the filtered-s LMS for FLANN filters [14, 15, 24], the filtered-x affine projection algorithm for both the adaptive Volterra filter and the FLANN structures [18], the filtered error LMS (FELMS) algorithm for FLANN filters [17], are all generalizations of the FXLMS.

More compact models are generally derived with recursive filters, such as bilinear models [19], recursive second order Volterra filters [21], recursive FLANN filters [22] or nonlinear autoregressive models with exogenous variables (NARX) [33]. As noted, for example, in [19], recursive models (even in the linear case) may be unstable unless the ANC algorithm is carefully designed, by suitably setting the step size or adopting specific algorithmic modifications, such as the introduction of a leakage factor. BIBO stability conditions have been derived for recursive Volterra filters [34], bilinear models [35], second order polynomial models [36, 37], recursive FLANN filters [22], essentially relying on the asymptotic stability of the linear submodel and ensuring the boundedness of the nonlinear terms. Interestingly enough, only the former assumption is necessary for recursive FLANN filters, given the specific bounded structure of the nonlinear terms [22].

Also, the error function depends nonlinearly on the coefficients and is therefore multimodal, so that the (gradient-based) ANC algorithm may become stuck in a local minimum. This is not considered an important limitation, as long as the noise reduction is still effective [4, 19].

In principle, due to the recursive nature of the model, the weight update mechanism should include the gradient of the SP, which implies computationally intensive recursive calculations. However, these recursions are commonly simplified according to the already mentioned Feintuch's assumption [25], resulting in various generalizations of the FULMS algorithm, such as the bilinear FXLMS (BFXLMS) [19] or the FELMS algorithm for recursive second order Volterra filters [17].

Besides the linear SP assumption, the main drawback of these approaches is that, in order to achieve sufficient model accuracy, the number of parameters must be large, giving rise to models of excessive size for parameter estimation purposes.

3.2. NARX models and model selection

In this work, we are mainly concerned with NARX models [33], which are a general nonlinear model class that encompasses all the previously mentioned models, and for which several techniques have been developed precisely to select the most appropriate and compact structure. The usage of NARX models in ANC schemes was first suggested in [23].

The deterministic NARX model is an input–output recursive model where the current output is given by a nonlinear functional expansion of lagged inputs and outputs:

$$y(k) = f(y(k-1), \dots, y(k-L), x(k), \dots, x(k-L)), \tag{6}$$

where $f(\cdot)$ is a generic nonlinear function. For identification purposes function $f(\cdot)$ is parameterized through a suitable functional expansion, the truncated polynomial being the most common choice. In that case, expression (6) can be rewritten as a linear regression of type (2) as well, the regressors being monomials in the arguments $y(k-1), \dots, y(k-L), x(k), \dots, x(k-L)$.

The full polynomial expansion (up to a given degree l) is generally avoided for the mentioned overparametrization issues, and efficient model selection methods have been developed to contain the model size, such as the forward-regression orthogonal estimator (FROE) [38], the simulation error minimization with pruning (SEMP) approach [28], and the fast recursive algorithm (FRA) [39]. As noted in [23], the combined use of a flexible model structure such as (6) with a model selection technique can yield excellent results in NANC applications, and efficiently reduces the number of weights to be updated on-line. On the downside, the application of standard model selection techniques in the ANC setting, which includes the SP, is non-trivial, due to the indirect identification setting of the problem.

Regarding the weight update mechanism, FULMS-type rules can be derived for NARX models as well, in the linear SP assumption.

3.3. NANC methods for nonlinear secondary paths

The nonlinearity of the SP makes the NANC problem inherently more complex and few solutions have been proposed in the literature so far for this more general problem. In [7, 13], multilayer artificial neural networks are used for this purpose. More recently, an adaptive control algorithm was developed in [20] for general function expansion nonlinear filters, based on the so-called virtual SP filter, which is related to the gradient of the SP. A FULMS algorithm for recursive FLANN filters with the SP modeled as a Hammerstein filter is proposed in [22].

In the following, we will briefly review the nonlinear filtered-gradient LMS (NFGLMS) [23] algorithm, which is derived from the cited algorithm of Zhou and DeBrunner [20], but is designed for recursive-type models of the polynomial NARX class. Accordingly, let the adaptive control filter be described as

$$y(k) = \boldsymbol{\phi}(z(k))^T \, \boldsymbol{w}(k), \tag{7}$$

where $z(k) = [y(k-1) \ldots y(k-L) \;\; x(k) \ldots x(k-L)]$, $\boldsymbol{w}(k)$ is the vector of (unknown) N_C parameters, and the terms $\phi_j(z(k))$ are monomials in the arguments $z(k)$, that is, $\phi_j(z(k)) = \prod_{i=1}^{2L+1} z_i(k)^{l_{ji}}$, l_{ji} being nonnegative integers such that $0 \leqslant \sum_{i=1}^{2L+1} l_{ji} \leqslant l$, $j = 1, \ldots, N_C$.

We will also assume that the SP can be described as a deterministic polynomial NARX model as well, that is,

$$y'(k) = \boldsymbol{\psi}(z'(k))^T \, \boldsymbol{v}(k), \tag{8}$$

where $z'(k) = [y'(k-1) \ldots y'(k-M) \;\; y(k) \ldots y(k-M)]$, $\boldsymbol{v}(k)$ is a vector of (known) N_S coefficients, and the $\psi_j(z'(k))$ terms are monomials in the arguments $z'(k)$, that is, $\psi_j(z'(k)) = \prod_{i=1}^{2M+1} z'_i(k)^{m_{ji}}$, m_{ji} being nonnegative integers such that $0 \leqslant \sum_{i=1}^{2M+1} m_{ji} \leqslant m$, $j = 1, \ldots, N_S$.

Recalling the parameter update equation (3), one has to derive the gradient term $y'_{\boldsymbol{w}}(k) = \partial y'(k)/\partial \boldsymbol{w}(k)$. The latter can be iteratively computed by means of the recursive nonlinear filter:

$$y'_{\boldsymbol{w}}(k) = \frac{\partial y'(k)}{\partial z'(k)} z'_{\boldsymbol{w}}(k), \tag{9}$$

where $z'_{\boldsymbol{w}}(k) = [y'_{\boldsymbol{w}}(k-1)^T \ldots y'_{\boldsymbol{w}}(k-M)^T \, y_{\boldsymbol{w}}(k)^T \ldots y_{\boldsymbol{w}}(k-M)^T]^T$, with $y_{\boldsymbol{w}}(k) = \partial y(k)/\partial \boldsymbol{w}(k)$, and the approximations

$$\frac{\partial y'(k-m)}{\partial \boldsymbol{w}(k)} \approx \frac{\partial y'(k-m)}{\partial \boldsymbol{w}(k-m)} = y'_{\boldsymbol{w}}(k-m), \tag{10}$$

$$\frac{\partial y(k-m)}{\partial \boldsymbol{w}(k)} \approx \frac{\partial y(k-m)}{\partial \boldsymbol{w}(k-m)} = y_{\boldsymbol{w}}(k-m), \tag{11}$$

have been used, in the hypothesis that the step size is sufficiently small for slow convergence [1]. The derivative of $y'(k)$ with respect to terms $z'(k)$, appearing in expression (9) can be easily computed as follows (model (8) is assumed given or estimated):

$$\frac{\partial y'(k)}{\partial z'(k)} = \frac{\partial \boldsymbol{\psi}(z'(k))}{\partial z'(k)}^T \boldsymbol{v}(k),$$

where $\frac{\partial \psi_j(z'(k))}{\partial z_i{}'(k)} = l_{ji} \frac{\psi_j(z'(k))}{z_i{}'(k)}, i = 1, \ldots, 2M+1, j-1, \ldots, N_S$.

The derivative $y_{\boldsymbol{w}}(k)$ can be computed recursively by using the following nonlinear dynamic filter:

$$y_{\boldsymbol{w}}(k) = \frac{\partial y(k)}{\partial \boldsymbol{w}(k)} = \boldsymbol{\phi}(z(k))^T + \frac{\partial \boldsymbol{\phi}(z(k))}{\partial \boldsymbol{w}(k)}^T \boldsymbol{w}(k) \tag{12}$$

where

$$\frac{\partial \phi_j(z(k))}{\partial \boldsymbol{w}(k)} = \sum_{i=1}^{M} l_{ji} \frac{\phi_j(z(k))}{y(k-i)}, y_{\boldsymbol{w}}(k-i), j = 1, \ldots, N_C.$$

employing once again the approximation (11). Notice that the last sum operator is extended only to the $y(\cdot)$ terms of $z(k)$. Finally, applying Feintuch's assumption (e.g., [1, 25]), one obtains the simplified expressions:

$$y'_{\boldsymbol{w}}(k) = \sum_{i=1}^{M} \frac{\partial y'(k)}{\partial y(k-i)} y_{\boldsymbol{w}}(k-i), \tag{13}$$

$$y_{\boldsymbol{w}}(k) = \boldsymbol{\phi}(z(k))^T, \tag{14}$$

respectively. Equations (3), (13) and (14) constitute the NFGLMS, the name following from the fact that the developed adaptation equation involves the filtering of the gradient of the controller output with respect to its weights, instead of the reference signal. The NFGLMS has been successfully used to deal with a NANC problem in which the nonlinearities are due to the saturation effects of the reference and error microphones. It was also shown in [23] that an *a priori* model selection phase that tailors the model structure and size to the given problem can be extremely beneficial.

Most of the computational effort of the NFGLMS is due to the nonlinear filtering tasks resulting from the indirect identification scheme. The adoption of Feintuch's assumption eliminates the stability problems related to the involved nonlinear filters (by eliminating the recursive terms), but only partially mitigates the computational issues. It also produces significant approximation in the gradient calculation (especially in the nonlinear case), resulting in reduced performance. This motivates the quest for alternative methods that can be cast in a direct identification setting, for further reductions of the computational cost. Moreover, given the intrinsic time variability of the NANC setting, it is also of interest to complement the adaptation method with an online structure selection procedure, to keep the model tuned to the current system dynamics. Efficient methods are available in the literature that could be employed for this purpose, provided that the problem be recast in a direct identification setting.

4. THE DFLMS ALGORITHM

4.1. The proposed NANC scheme

For ease of notation let the expressions

$$y(k) = C x(k), \tag{15}$$

$$y'(k) = S y(k), \tag{16}$$

succinctly represent the control filter and the SP, C and S denoting the nonlinear dynamical operators corresponding to the NARX models of Eqs (7) and (8), respectively. Let also AB denote the composition of two nonlinear dynamical operators, corresponding to the series connection of A and B. Accordingly, the secondary signal $y'(k)$ the standard ANC scheme can be also expressed as

$$y'(k) = S\ y(k) = S(Cx)(k) = SC\, x(k). \tag{17}$$

Clearly, in general

$$y'(k) = SC\, x(k) \neq CS\, x(k), \tag{18}$$

even if both nonlinear systems are assumed time-invariant.

However, there may exist an auxiliary nonlinear system $\tilde{S}$ such that

$$y'(k) = C\tilde{S}\ x(k).$$

In that case, the NANC scheme would be equivalent to that of Figure 2, with $\tilde{S}$ in the place of the true SP S, and a direct identification scheme could be applied for the identification of C, using the filtered reference signal

$$\tilde{x}(k) = \tilde{S}\, x(k)$$

as input to C and the measured signal $e(k)$ as modeling error. More precisely, from Eq. (3), the controller weight update equation is obtained as

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \mu \frac{\partial y'(k)}{\partial \boldsymbol{w}(k)}^{T} e(k), \tag{19}$$

where

$$y'(k) = SC\, x(k) \;\cong\; C\tilde{S}x(k) = C\tilde{x}(k) = \boldsymbol{w}(k)^T\, \tilde{\boldsymbol{x}}(k)$$

with $\tilde{\boldsymbol{x}}(k) = \boldsymbol{\phi}(\tilde{z}(k))$, $\phi_j(\tilde{z}(k))$ being a monomial in the arguments $\tilde{z}(k) = [y(k-1) \ldots y(k-L)\; \tilde{x}(k) \ldots \tilde{x}(k-L)]$. The resulting weight update law is then simply:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \mu \tilde{\boldsymbol{x}}(k) e(k), \tag{20}$$

which is an extension of the FXLMS algorithm if C has a non-recursive structure or of the FULMS otherwise.

Finding a general analytic expression for $\tilde{S}$ such that the swapping condition (18) is satisfied is a difficult task because of the nonlinearities introduced by the NARX controller and the SP. Actually, there is no guarantee that an auxiliary nonlinear system $\tilde{S}$ such that $C\tilde{S}\, x(k) = SC\, x(k)$ exists, but it is arguable that it is actually necessary. In the linear framework the estimated SP used to prefilter the reference signal in the FXLMS is only required to be within a $\pm 90^\circ$ phase range from S at every frequency, for algorithm stability. Although a similar condition has not yet been proven in the nonlinear domain, the simulation examples shown in the sequel show that excellent cancellation performance can be achieved even with an approximate auxiliary system. Building on this, a dual adaptation scheme is here proposed for the online estimation of $\tilde{S}$, based on the minimization of the commutation error

$$\tilde{e}(k) = y'(k) - \tilde{y}'(k) = y'(k) - C\tilde{S}\, x(k). \tag{21}$$

The estimation of $\tilde{S}$ configures an indirect identification problem where the (nonlinear) control filter C has the role of the SP, but, given the much less stringent accuracy requirements on $\tilde{S}$ compared with C, a fixed structure can be assumed for $\tilde{S}$, and an indirect algorithm such as the NFGLMS can be employed for its estimation, given the current controller C. More precisely, representing $\tilde{S}$ also as a NARX model:

$$\tilde{x}(k) = \boldsymbol{\eta}(\boldsymbol{z}(k))^T \boldsymbol{u}(k) \tag{22}$$

where $\boldsymbol{z}(k) = [\tilde{x}(k-1) \ldots \tilde{x}(k-M) \;\; x(k) \ldots x(k-M)], \boldsymbol{u}(k)$ is a vector of coefficients, and $\eta_j(\boldsymbol{z}(k))$ is a monomial in the arguments $\boldsymbol{z}(k)$, $j = 1, \ldots, N_{\tilde{S}}$; the NFGLMS weight update rule is given by:

$$\boldsymbol{u}(k+1) = \boldsymbol{u}(k) + \gamma \frac{\partial \tilde{y}(k)}{\partial \boldsymbol{u}(k)}^{T} \tilde{e}(k), \tag{23}$$

where γ is the step size of the auxiliary loop. The derivative in Eq. (23) is computed using the chain rule as

$$\frac{\partial \tilde{y}(k)}{\partial \boldsymbol{u}(k)} = \boldsymbol{w}(k)^T \frac{\partial \boldsymbol{\phi}(\tilde{z}'(k))}{\partial \tilde{z}'(k)} \frac{\partial \tilde{z}'(k)}{\partial \boldsymbol{u}(k)}, \tag{24}$$

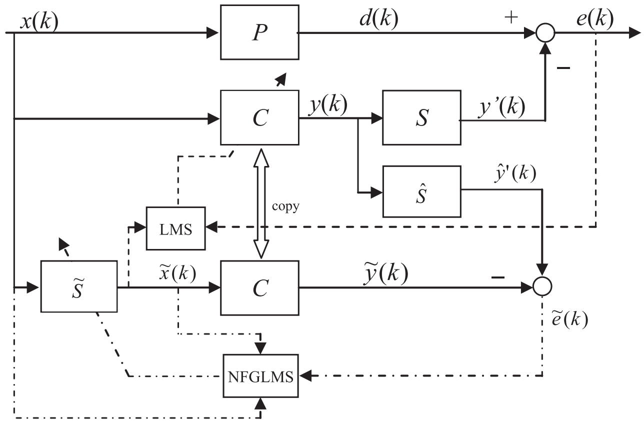

Figure 3. DFLMS scheme for NANC using NARX filters: dual adaptation of $\tilde{S}$ (dash–dot lines) and C (dashed lines).

using the gradient of C, where $\tilde{z}'(k) = [\tilde{y}(k-1)\ldots\tilde{y}(k-L)\ \tilde{x}(k)\ldots\tilde{x}(k-L)]$ (see Section 3.3 for an efficient rule for computing the recursive expression (24)). Notice that both weight update Eqs (20) and (23) are computed on the basis of prefiltered signals. In (20) the filtering function is $\tilde{S}$, whereas the filtered gradient of C is used in (23). Accordingly, the overall algorithm is named DFLMS.

The proposed strategy is schematically depicted in Figure 3.

Notice that a model of the SP is required for the computation of the auxiliary error $\tilde{e}(k)$, whereas a copy of the current control filter is used in the auxiliary loop. In the implemented scheme, at each step, the auxiliary model $\tilde{S}$ is updated first based on the current controller output, and the control filter C next using the filtered reference noise obtained with the updated $\tilde{S}$.

4.2. *Computational complexity analysis*

It is interesting to compare the computational complexity of the proposed DFLMS with the standard linear algorithm FXLMS, as well as the NFGLMS, which is also designed for general NANC problems as the DFLMS. The three algorithms can be decomposed into three main steps that can be analyzed separately [19]:

1. computation of the control filter output $y(k)$ (and auxiliary system output $\tilde{x}(k)$ for the DFLMS);
2. computation of the filtered signals ($x'(k)$, $y'(k)$, $\tilde{y}(k)$, or the filtered output gradient, as appropriate); and
3. updating of the filters weights.

The obtained results for the analyzed approaches are summarized in Table I, where N_C, N_S and $N_{\tilde{S}}$ are the sizes of the control filter C, the SP S and the auxiliary block $\tilde{S}$, respectively, in terms of number of parameters (the cost of computing nonlinear regressors from the elementary signals has been overlooked in this analysis), and L and M are the maximum lags of the control filter and the SP, respectively ($L << N_C$, $M << N_S$). Regarding the NFGLMS, notice that explicit formulas for the calculation of the derivatives $\partial y'(k)/\partial y(k-i)$ in expression (13) can be precalculated if the SP is assumed fixed and given (or estimated). Even neglecting this cost, Eq. (13) requires M products and $M-1$ sums for each of the N_C regressors of C. Notice that, in terms of the respective number of parameters, the FXLMS and the NFGLMS have equal complexity for steps 1 and 3 (step 2 costs more for the NFGLMS in view of Eq. (13)). The DFLMS (which combines the use of both algorithms) has a generally higher cost. Furthermore, the NFGLMS and the DFLMS use a NARX control filter model as opposed to a simple FIR as in the FXLMS. In the latter case $N_C = L+1$, whereas a full quadratic NARX with the same maximum lag amounts to $N_C = 2L^2+5L+2$ terms. Thus, in practice, the NFGLMS and DLMS schemes are much costlier than the FXLMS, unless the

Table I. Computational complexity comparison of the active noise control algorithms.

Method	# of op.s	Step 1	Step 2	Step 3	Total
FXLMS	mult.s	N_C	N_S	$N_C + 1$	$2N_C + N_S + 1$
	add.s	$N_C - 1$	$N_S - 1$	N_C	$2N_C + N_S - 2$
NFGLMS	mult.s	N_C	$N_C M$	$N_C + 1$	$N_C(M+2) + 1$
	add.s	$N_C - 1$	$N_C(M-1)$	N_C	$N_C(M+2) - 1$
DFLMS	mult.s	$N_C + N_{\tilde{S}}$	$N_S + N_{\tilde{S}} L$	$N_C + N_{\tilde{S}} + 2$	$2N_C + N_S + N_{\tilde{S}}(L+2) + 2$
	add.s	$N_C + N_{\tilde{S}} - 2$	$N_S - 1 + N_S(L-1)$	$N_C + N_{\tilde{S}}$	$2N_C + N_S + N_{\tilde{S}}(L+1) - 3$

NARX model structure is suitably optimized. In fact, to achieve a sufficient accuracy with an FIR, it is generally necessary to employ a large L, so that the FIR size is generally much larger than that of an optimized NARX model.

The main advantage of the DFLMS is precisely that it allows to apply powerful online model selection tools to reduce the size of the NARX control filter, as described in the next section. In addition, the adaptation of the auxiliary block $\tilde{S}$, which is responsible for the quadratic term in the complexity of the DFLMS, but is less critical, can be suitably downsampled to distribute the load over several steps. In the latter case, the complexity of the DFLMS can be made much lower than the NFGLMS, in practice.

5. NANC WITH ONLINE MODEL SELECTION

5.1. Online model selection techniques for NARX models

Several model selection techniques have been developed for NARX models, such as the already mentioned FROE [38], SEMP [28], and FRA [39] algorithms. All of these techniques are based on batch algorithms and can consequently perform the selection task only off-line, using a sufficiently large collection of data. Recently, some efficient online model selection methods have also been proposed [32].

The recursive forward regression with pruning (RFRP) algorithm is based on a recursive orthogonal least squares (ROLS) procedure that efficiently integrates model augmentation and pruning to reduce processing time whenever new data are available. The selection is operated over a user-defined candidate regressor set and a sliding window of data is used for model updating. When a new datum arrives the current model is re-estimated and evaluated. If the accuracy has fallen below a given threshold, a model structure modification is triggered. This consists in one or more model addition/pruning iterations until the desired accuracy level is recovered. More in detail, at each of these iterations the RFRP evaluates for possible inclusion all the regressors left out of the current model, based on the error reduction ratio (ERR) criterion [38], which basically measures the marginal model improvement obtained by adding each regressor. The regressor with the highest ERR is then added to the model. Subsequently, it prunes redundant terms, as long as the combined addition/elimination of terms improves the model accuracy. The main tuning knobs of the RFRP are the size W of the data window and the accuracy thresholds for the overall model (J_{thres}) and the model increment (ERR_{thres}), respectively. An efficient implementation of the described algorithm is described in [32]. It is also shown in the same reference that the RFRP provides excellent model tracking, albeit at the cost of a significant computational load.

The ROLS-LASSO algorithm [32] also exploits the ROLS approach, combining it with the least absolute shrinkage and selection operator (LASSO) statistical regularization method [40]. Although slightly less accurate than the RFRP, the ROLS-LASSO is a much lighter method, suitable for time- or computation-critical applications, and thus provides a viable low-cost alternative to the RFRP.

5.2. Integration of the RFRP online selection method in the DFLMS scheme

The main achievement of the DFLMS scheme consists in having reformulated the adaptation problem of the control filter C as a direct identification problem, using the approximate block

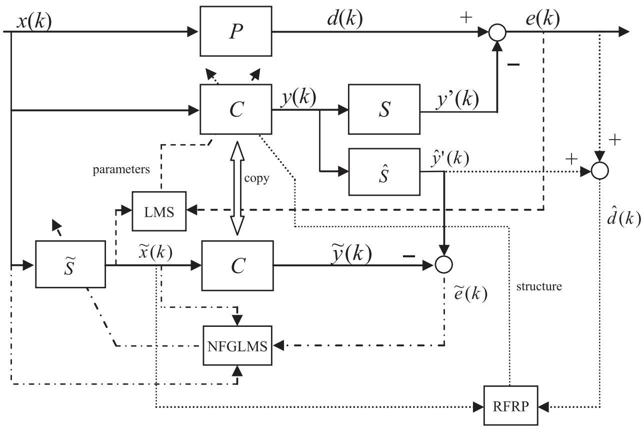

Figure 4. RFRP-DFLMS scheme for NANC using NARX filters: adaptation of $\tilde{S}$ (dash–dot lines), C (dashed lines) and model structure (dotted lines).

commutation scheme by means of the auxiliary system $\tilde{S}$. More precisely the identification of C amounts to a linear regression problem (the NARX model is linear in the parameters), where the input to the model is $\tilde{x}(k)$, and the model error is directly represented by the measured signal $e(k)$. This allows the direct implementation of an online model structure selection algorithm, as for example, the mentioned RFRP (Figure 4). The resulting algorithm is denoted RFRP-DFLMS.

The overall computational complexity of the RFRP-DFLMS is the sum of those of the two algorithms. In particular, the computational load of the RFRP is discussed in detail in [32]. Briefly, defining N_{tot} as the total number of regressors and N_{sel} as the number of selected regressors (typically $N_{sel} << N_{tot}$), the computational cost of the RFRP is of the order $O(N_{tot} N_{sel}^3)$, assuming that on average the algorithm performs one full forward selection and pruning iteration at each step. In practice, the RFRP should be triggered only occasionally, when the assigned minimal accuracy level is not met. The cost of RFRP iterations is generally well repaid by the reduction of the model size, which implies a reduction of the core weight update computations, as opposed to using the full NARX model with all N_{tot} regressors. In addition, the online model selection algorithm can be efficiently parallelized with the parameter adaptation mechanism and even suitably downsampled if necessary.

6. SIMULATION EXPERIMENTS

6.1. Example 1

The first example aims at a performance evaluation of the DFLMS scheme (compared with the NFGLMS and the FXLMS) in terms of disturbance rejection and speed of convergence. A simple time-invariant scenario [10] has been used for this purpose, where the PP is modeled as a polynomial model with maximum lag 2:

$$d(k) = 0.5d(k-1) - 0.3x(k-2) + 0.25x(k-1)^2 x(k-2), \tag{25}$$

and the SP is modeled as a non-minimum phase 4^{th} order FIR system:

$$y'(k) = y(k-2) + 1.5y(k-3) - y(k-4), \tag{26}$$

As for the reference noise $x(k)$, the same signal used in [10] has been considered, that is, a sinusoidal wave of 500 Hz sampled at an 8 kHz rate:

$$x(k) = \sqrt{(2)} \sin\left(\frac{2 \quad \cdot 500 \cdot k}{8000}\right) + v(k), \tag{27}$$

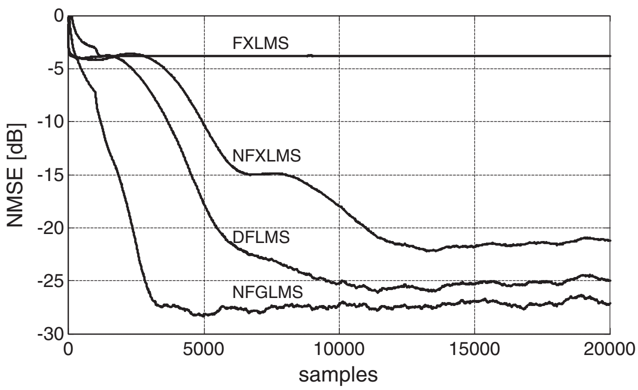

Figure 5. Comparison of different nonlinear active noise control strategies for setting 1: NFGLMS, FXLMS, NFXLMS and DFLMS.

where an additive white noise process $v(k)$ with Gaussian distribution has also been assumed. The variance of $v(k)$ is chosen such that the power signal-to-noise ratio is equal to 40 dB.

A fixed NARX structure has been assumed for the control filter, that is, a full 3$^{\text{rd}}$ degree polynomial expansion with a maximum lag of 3, for a total number of 119 parameters. Two different variants of the DFLMS schemes have been analyzed. In the first, the adaptation of $\tilde{S}$ is switched off ($\gamma = 0$), so that DFLMS degenerates to a nonlinear version of the FXLMS, here denoted as NFXLMS ($\mu = 5 \cdot 10^{-5}$). The second variant ($\mu = 10^{-4}$ and $\gamma = 0.01$) considers an eight-tap FIR filter model for $\tilde{S}$. In both cases, the auxiliary model $\tilde{S}$ is initialized to be equal to S, for simplicity. The NFGLMS ($\mu = 5 \cdot 10^{-4}$) was tested as well, for comparison purposes. The NANC schemes are also compared with the standard linear FXLMS ($\mu = 10^{-3}$), where the order of the FIR controller has been chosen equal to the number of parameters of the NARX controller, for fairness of comparison. The adaptation gains have been optimized for all the considered schemes, so as to guarantee the best convergence speed without compromising stability. The control filter weights are initially set to zero.

Figure 5 displays the attenuation performance of the mentioned algorithms in terms of the normalized mean square error (NMSE) [10], expressed as

$$NMSE = 10 \log_{10} \left(\frac{E[e(k)^2]}{\sigma_{d^2}} \right), \tag{28}$$

versus the iteration steps. It is clearly noticeable that the linear FXLMS, albeit with the same number of parameters of the NANC schemes, provides the worst performance in term of disturbance cancellation (only −3.8 dB), whereas, as expected, the NFGLMS turns out to be the best algorithm (−27.2 dB). The DFLMS provides an intermediate performance level that approaches the steady-state attenuation of the NFGLMS when the online adaptation of $\tilde{S}$ is active.

As shown in Figure 6, the nonlinear filtered-x LMS suffers from a high commutation error, which slows down convergence and reduces the rejection capability at steady state (−21.2 dB) compared with the NFGLMS. The online adaptation of $\tilde{S}$ allows the DFLMS algorithms to reduce the commutation error, as demonstrated in Figure 6, significantly improving both the speed of convergence and the canceling performance (−25 dB).

The slower convergence of the DFLMS, as opposed to the NFGLMS, is mainly because of, to prevent instability of the adaptation loops, the step sizes of the DFLMS cannot be increased too much.

A further simulation was carried out using a quadratic nonlinear FIR filter with maximum lag 8 to model $\tilde{S}$, but no significant improvement was achieved with respect to the FIR case. Therefore, these results are omitted.

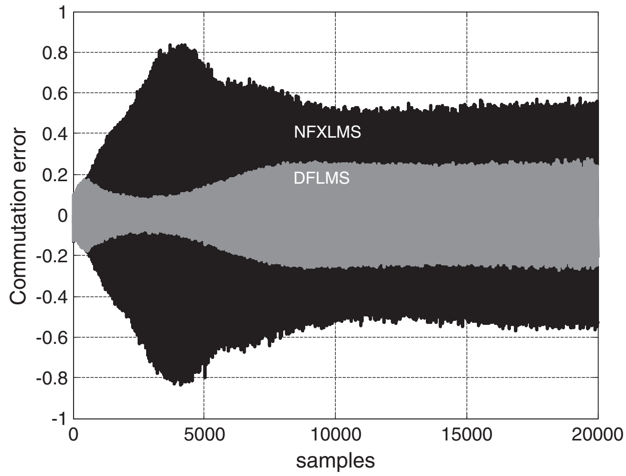

Figure 6. Time history of the commutation error in setting 1: NFXLMS (black) and DFLMS (gray).

6.2. Example 2

The second setting aims at showing the performance achievable by means of the inclusion of an online model selection approach. In this case, the primary noise at the canceling point is generated by the following 3^{rd} degree polynomial model [10]:

$$d(k) = w(k-2) + 0.02w(k-2)^2 - 0.04w(k-2)^3, \tag{29}$$

where

$$w(k) = x(k-3) - 0.3x(k-4) + 0.2x(k-5). \tag{30}$$

The reference noise $x(k)$ considered in this case is a band-limited white noise in the (normalized) frequency interval [0.01, 0.1] and with unitary variance.

To emphasize the advantage that can be gained using the model selection approach, a time-varying SP is assumed. Specifically, at the commutation instant $\bar{k} = 10000$, the SP changes from Eq. (26) to

$$y'(k) = 0.7756y(k) + 0.5171y(k-1) - 0.362y(k-2), \tag{31}$$

that is, the non-minimum phase FIR SP used in [23].

The RFRP-DFLMS has been compared with the NFGLMS ($\mu = 5 \cdot 10^{-4}$) and the plain DFLMS ($\mu = 10^{-4}$, $\gamma = 0.01$) with a quadratic polynomial NARX control filter. For the RFRP-DFLMS, the model structure is modified according to the selection algorithm and starting from a plain FIR filter; whereas in the other cases, the controller structure is fixed to a full quadratic expansion, with a maximum lag of 10. The parameters of the RFRP-DFLMS are set to $\mu = 10^{-3}$, $\gamma = 0.1$, $W = 600$, $J_{thres} = 10^{-6}$, $ERR_{thres} = 5{\cdot}10^{-5}$. The initial controller parameters are set to zero, and the step sizes are selected to guarantee the best trade-off between performance and stability, for all the schemes. The auxiliary block is an eight-tap FIR filter, for simplicity. The NMSE results are plotted in Figure 7.

At the onset of the simulation, it takes the DFLMS methods a longer time to achieve the same rejection performance as the NFGLMS, but the RFRP-DFLMS is not particularly penalized by the model structure adaptation process. On the other hand, after the structural modification of the SP, the RFRP-DFLMS reacts much more rapidly and efficiently than both other algorithms, achieving in only 1000 samples the level of rejection performance reached by the NFGLMS after 20 000 iterations (the plain DFLMS is even slower).

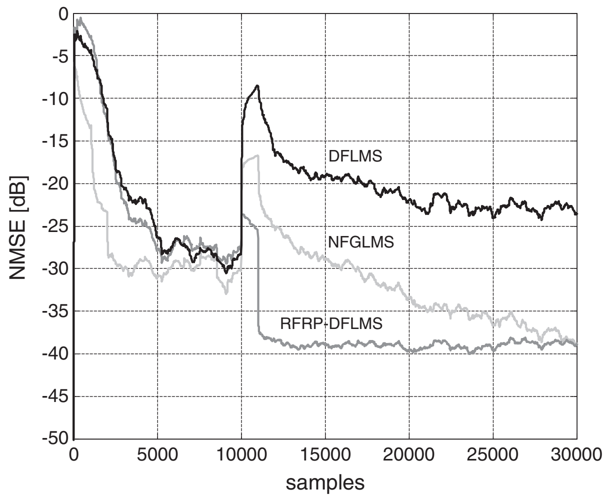

Figure 7. NMSE results for setting 2: NFGLMS (light gray), DFLMS (black) and RFRP-DFLMS (dark gray).

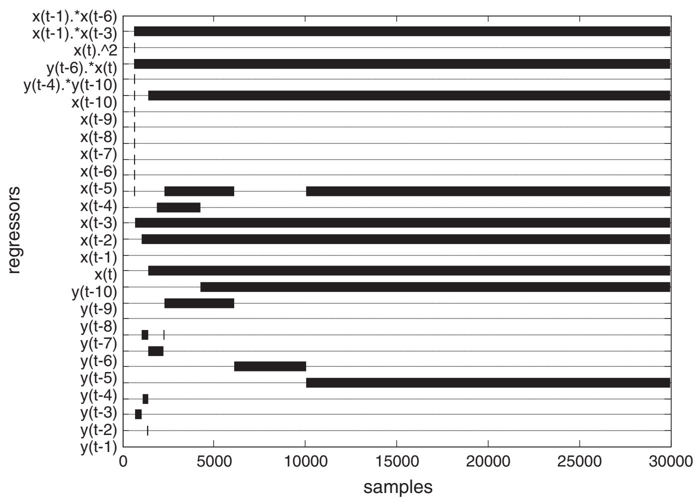

Figure 8. Regressors selected by the RFRP algorithm in experiment 2.

It is interesting to remark that the results provided by the RFRP-DFLMS are obtained with a minimal control filter structure, as pointed out in Figure 8, which displays the selected regressors versus the number of iterations (for brevity, only the regressors that have been selected at least once by the RFRP are reported). The average number of selected regressors is 9, against a total of 252 terms considered by the NFGLMS and the plain DFLMS.

6.3. *Example 3*

We next consider an example where both acoustic paths are nonlinear, taken from [20]. Precisely, both the PP and the SP are given by Volterra series, respectively:

$$\begin{aligned} d(k) = {} & x(k) + 0.8x(k-1) + 0.3x(k-2) + 0.4x(k-3) - 0.8x(k)x(k-1) \\ & + 0.9x(k)x(k-2) + 0.7x(k)x(k-3), \end{aligned} \tag{32}$$

$$y'(k) = y(k) + 0.35y(k-1) + 0.09y(k-2) - 0.5y(k)y(k-1) + 0.4y(k)y(k-2), \tag{33}$$

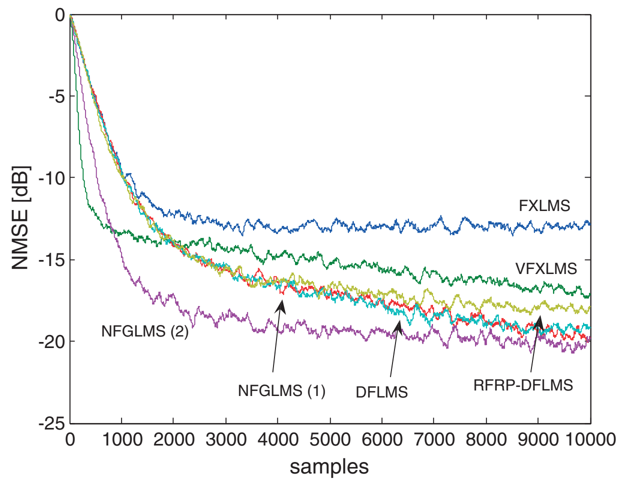

Figure 9. Example 3: average learning curves (NMSE) for various models and algorithms (FXLMS: FIR, VFXLMS, NFGLMS $^{(1)}$ and DFLMS: Volterra model; NFGLMS $^{(2)}$ and RFRP-DFLMS: NARX model).

and the reference signal is a Gaussian white noise. Several model structures and methods have been tried on this example. A linear 10-tap FIR controller was first obtained with the standard FXLMS ($\mu = 0.01$). Then, the VFXLMS ($\mu_1 = 0.05$, $\mu_2 = 0.005$), the NFGLMS ($\mu = 0.01$) and the proposed DFLMS ($\mu = 0.01$, $\gamma = 0.0025$) have been tested on a full quadratic Volterra model with $L = 10$ (77 terms). Finally, the NFGLMS ($\mu = 0.02$) and the RFRP-DFLMS ($\mu = 0.001$, $\gamma = 0.005$) have been tested on a full quadratic NARX model with $L = 4$ (54 terms). In the DFLMS and RFRP-DFLMS simulations, the auxiliary system $\tilde{S}$ has been initialized to S. Algorithm gains have been optimized individually.

Figure 9 reports the average learning curves for 50 runs of each of the algorithms, similarly to [20] (a further moving average smoothing is applied to improve readability). As expected, the FXLMS is the least effective algorithm regarding noise attenuation performance. All other methods tend toward a one order of magnitude noise reduction. In particular, regarding the Volterra model, the most successful method appears to be the NFGLMS, which achieves a 20 dB reduction on average, whereas the VFXLMS obtains 17 dB. The VFXLMS displays a quite different transient compared with the other algorithms, essentially due to the different step size in the weight update related to linear and quadratic terms. Interestingly enough, the DFLMS almost perfectly matches the performance of the NFGLMS. Regarding the auxiliary loop in the DFLMS, the 1-norm of the commutation error settles at 0.05.

Despite the shorter maximum lag and the smaller size, the considered quadratic NARX structure is flexible enough to guarantee a significative improvement in the performance if adapted with the NFGLMS. The RFRP-DFLMS has been applied to the same model, yielding a slightly worse performance on average. However, it is capable of rapidly reducing the effective model size to an average of 27.6 terms (over a total of 54). This guarantees a further halving of the computational cost once the RFRP is turned off. Overall, the RFRP-DFLMS provides a convenient trade-off between attenuation performance and computational cost. In this respect, notice that this trade-off can be modulated by the user by setting the threshold parameters of the selection algorithm.

6.4. *Example 4*

Another stimulating NANC problem is discussed in [19] and [23], concerning a system affected by microphone saturation both at the reference and the error microphone, as represented in Figure 10.

In the ANC scheme, the acoustic paths are assumed linear, the nonlinearity being introduced by the saturations on the microphone signals. Following [23], the PP and SP are modeled by two FIR filters, taken from [41]:

$$d(k) = 0.0179x(k) + 0.1005x(k-1) + 0.279x(k-2) + 0.489x(k-3) + 0.586x(k-4) + 0.489x(k-5) + 0.279x(k-6) + 0.1005x(k-7) + 0.0179x(k-8), \quad (34)$$

$$y'(k) = 0.7756y(k) + 0.5171y(k-1) + 0.362y(k-2), \quad (35)$$

The reference signal is the sum of three sine waves at the normalized frequencies of 0.02, 0.04 and 0.08. Three different settings of increasing complexity are considered:

- setting WU) reference signal weakly saturated,

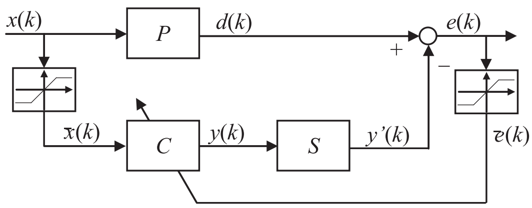

Figure 10. Block diagram of a nonlinear active noise control system in the presence of saturated reference and error signals.

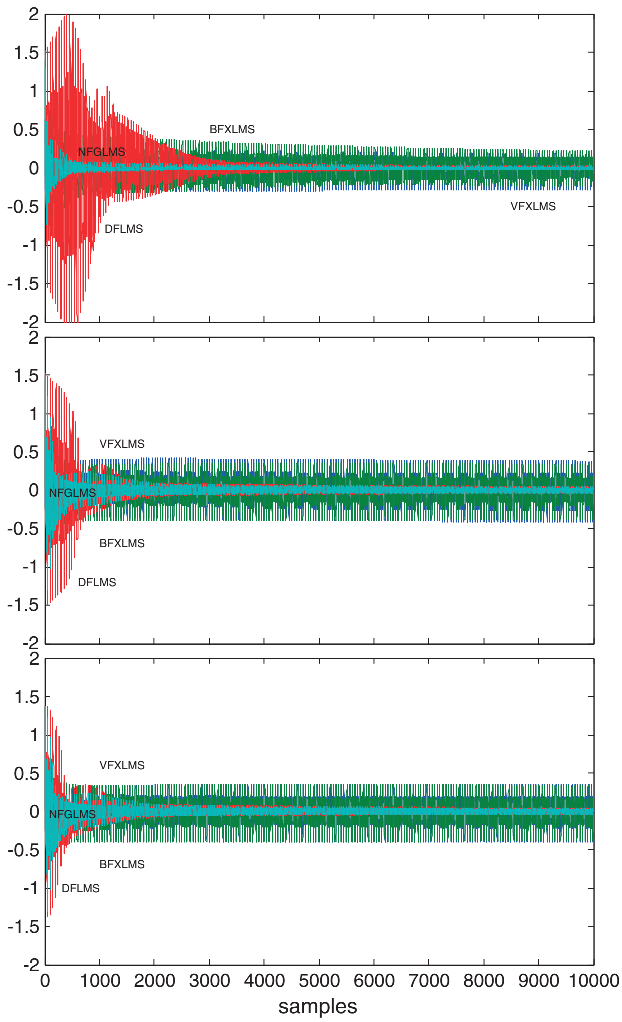

Figure 11. Residual noise for settings WU (top), SU (middle), and SW (bottom).

- setting SU) reference signal strongly saturated, and
- setting SW) reference signal strongly saturated, error signal weakly saturated,

where weak and strong saturation refer respectively to a 90% and a 50% clipping of the signal with respect to its maximum value [6, 19].

Figure 11 reports the attenuation performance of the VFXLMS, BFXLMS, NFGLMS and DFLMS algorithms in the three settings. A full quadratic NARX model has been used in the last two cases. All models have the same L for a given setting ($L = 4$ for setting WU, $L = 8$ for settings SU and SW). A simple 10-tap FIR model (with random initial parameters) has been used as auxiliary model for the DFLMS. All algorithm gains have been optimized individually.

In all three settings, both the VFXLMS and BFXLMS achieve a noise reduction of only a factor 3, whereas the NMSE (after convergence) equals -38 dB for setting WU and -33 dB for settings SU and SW for both NFGLMS and DFLMS. Indeed, after a brief transient required for the estimation of $\tilde{S}$, the DFLMS provides almost equal performance to the NFGLMS. This is all the more remarkable, considering that the auxiliary system is initialized with random parameters.

7. CONCLUSIONS

A novel NANC scheme exploiting the flexibility of NARX models has been proposed. Its main feature is the commutation of the control filter block and the SP dynamics, as commonly assumed in the linear case, compensating the commutation error by means of an auxiliary filter also adapted online. Both the main and the auxiliary adaptation loops operate with weight update rules based on the use of suitable prefiltered signals. The control filter adaptation essentially requires the same computational effort as the standard FXLMS/FULMS algorithms, whereas the adaptation of the auxiliary filter, which is not time critical, can be slowed down and downsampled. In view of the dual adaptation loops, the algorithm is called DFLMS.

The proposed scheme configures the control filter adaptation task as a direct identification problem, which allows the use of an online model selection algorithm. The latter can be used to trim the model size and to adapt the model structure to account for the variability of the ANC setting. An extended version of the DFLMS has thus been implemented using the RFRP algorithm as model selection strategy.

The proposed methods have been tested on several scenarios of different complexity. The DFLMS achieves almost the same level of steady state disturbance noise cancellation as the NFGLMS, although with a slightly slower convergence rate. The RFRP-DFLMS, on the other hand, is very efficient in reducing the model size, thereby cutting to a minimum the computational load, and reacts much more rapidly than the NFGLMS to sudden variations of the system.

REFERENCES

1. Kuo SM, Morgan DR. *Active Noise Control Systems – Algorithms and DSP Implementations*. Wiley: New York, 1996.
2. Elliott SJ. Down with noise. *IEEE Spectrum* 1999; **36**:54–61.
3. Kuo SM, Morgan DR. Active noise control: a tutorial review. *Proceedings of the IEEE* 1999; **87**(6):943–973.
4. Elliott SJ. *Signal Processing for Active Control*. Academic Press: San Diego, CA, 2001.
5. Matsuura T, Hiei T, Itoh H, Torikoshi K. Active noise control by using prediction of time series data with a neural network. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics,* vol. 3, Vancouver, BC, Canada, 1995; 2070–2075.
6. Kuo SM, Wu H-T, Chen F-K, Gunnala MR. Saturation effects in active noise control systems. *IEEE Transactions on Circuits and Systems I: Regular Papers* 2004; **51**(6):1163–1171.
7. Snyder SD, Tanaka N. Active control of vibration using a neural network. *IEEE Transactions on Neural Networks* 1995; **6**(4):819–828.
8. George NV, Panda G. Advances in active noise control: a survey, with emphasis on recent nonlinear techniques. *Signal Processing* 2013; **93**:363–377.
9. Strauch P, Mulgrew B. Active control of nonlinear noise processes in a linear duct. *IEEE Transactions on Signal Processing* 1998; **46**(9):2404–2412.
10. Tan L, Jiang J. Adaptive Volterra filters for active control of nonlinear noise processes. *IEEE Transactions on Signal Processing* 2001; **49**(8):1667–1676.

11. Reddy EP, Das DP, Prabhu KMM. Fast adaptive algorithms for active control of nonlinear noise processes. *IEEE Transactions on Signal Processing* 2008; **56**(9):4530–4535.
12. Zhao H, Zeng X, Zhang X, He Z, Li T, Jin W. Adaptive extended pipelined second-order Volterra filter for nonlinear active noise controller. *IEEE Transactions on Audio, Speech, and Language Processing* 2012; **20**(4):1394–1399.
13. Bouchard M, Paillard B, Dinh CTL. Improved training of neural networks for the nonlinear active control of sound and vibration. *IEEE Transactions on Neural Networks* 1999; **10**(2):391–401.
14. Das DP, Panda G. Active mitigation of nonlinear noise processes using a novel filtered-s LMS algorithm. *IEEE Transactions on Speech and Audio Processing* 2004; **12**(3):313–322.
15. Das DP, Mohapatra SR, Routray A, Basu TK. Filtered-s LMS algorithm for multichannel active control of nonlinear noise processes. *IEEE Transactions on Audio, Speech and Language Processing* 2006; **14**(5):1875–1880.
16. Sicuranza GL, Carini A. A generalized FLANN filter for nonlinear active noise control. *IEEE Transactions on Audio, Speech and Language Processing* 2011; **19**(8):2412–2417.
17. Zhao H, Zeng X, Zhang J. Adaptive reduced feedback FLNN nonlinear filter for active control of nonlinear noise processes. *Signal Processing* 2010; **90**(3):834–847.
18. Sicuranza GL, Carini A. Piecewise–linear expansions for nonlinear active noise control. *Proceeding of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France, 2006; 209–212.
19. Kuo SM, Wu H-T. Nonlinear adaptive bilinear filters for active noise control systems. *IEEE Transactions on Circuits and Systems I: Regular Papers* 2005; **52**(3):1–8.
20. Zhou D, DeBrunner V. Efficient adaptive nonlinear filters for nonlinear active noise control. *IEEE Transactions on Circuits and Systems I: Regular Papers* 2007; **54**(3):669–681.
21. Zhao H, Zeng X, He Z, Li T. Adaptive RSOV filter using the FELMS algorithm for nonlinear active noise control systems. *Mechanical Systems and Signal Processing* 2013; **34**(1–2):378–392.
22. Sicuranza GL, Carini A. On the BIBO stability condition of adaptive recursive FLANN filters With application to nonlinear active noise control. *IEEE Transactions on Audio, Speech and Language Processing* 2012; **20**(1): 234–245.
23. Napoli R, Piroddi L. Nonlinear active noise control with NARX models. *IEEE Transactions on Audio, Speech and Language Processing* 2010; **18**(2):286–295.
24. Das DP, Panda G, Nayak DK. Development of frequency domain block filtered–s LMS (FBFSLMS) algorithm for active noise control system. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France, 2006; 289–292.
25. Feintuch PL. An adaptive recursive LMS filter. *Proceedings of the IEEE* 1976; **64**(11):1622–1624.
26. Söderström T, Stoica P. *System Identification*. Prentice-Hall: London, 1989.
27. Aguirre LA, Billings SA. Dynamical effects of overparametrization in nonlinear models. *Physica D: Nonlinear Phenomena* 1995; **80**:26–40.
28. Piroddi L, Spinelli W. An identification algorithm for polynomial NARX models based on simulation error minimization. *International Journal of Control* 2003; **76**(17):1767–1781.
29. Bjarnason E. Active noise cancellation using a modified form of the filtered-x. *Proceedings of the EURASIP European Signal Processing Conference (EUSIPCO),* vol. 2, Amsterdam: Elsevier, 1992; 1053–1056.
30. Liao C, Lin J. New FIR filter-based adaptive algorithms incorporating with commutation error to improve active noise control performance. *Automatica* 2007; **42**(2):325–331.
31. Lin J, Liao C. New IIR filter-based adaptive algorithm in active noise control applications: commutation error-introduced LMS algorithm and associated convergence assessment by a deterministic approach. *Automatica* 2008; **44**(11):2916–2922.
32. Cantelmo C, Piroddi L. Adaptive model selection for polynomial NARX models. *IET Control Theory and Applications* 2010; **4**(12):2693–2706.
33. Leontaritis IJ, Billings SA. Input–output parametric models for non-linear systems – Part II: stochastic non-linear systems. *International Journal of Control* 1985; **41**:329–344.
34. Mumolo E, Carini A. On the stability of discrete time recursive Volterra filters. *IEEE Signal Processing Letters* 1999; **6**:230–232.
35. Mathews VJ, Sicuranza GL. *Polynomial Signal Processing*. Wiley: New York, 2000.
36. Mumolo E, Carini A. A stability condition for adaptive recursive second-order polynomial filters. *Signal Processing* 1996; **54**(1):85–90.
37. Roy E, Stewart RW, Durrani TS. Stability condition for certain recursive second-order polynomial filters. *Electronics Letters* 1996; **32**(16):1522–1523.
38. Billings SA, Chen S, Korenberg MJ. Identification of MIMO non-linear systems using a forward-regression orthogonal estimator. *International Journal of Control* 1989; **49**:2157–2189.
39. Li K, Peng J-X, Irwin GW. A fast nonlinear model identification method. *IEEE Transactions on Automatic Control* 2005; **50**(8):1211–1216.
40. Tibshirani R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B* 1996; **58**(1):267–288.
41. Tobias OJ, Seara R. Performance comparison of the FXLMS, nonlinear FXLMS and leaky FXLMS algorithms in nonlinear active noise applications. *Proceedings of 11th European Signal Processing Conference,* vol. 1, Toulouse, France, 2002; 155–158.