ELSEVIER

Signal Processing 80 (2000) 1747–1760

SIGNAL PROCESSING

www.elsevier.nl/locate/sigpro

Adaptation of a memoryless preprocessor for nonlinear acoustic echo cancelling

Alexander Stenger*, Walter Kellermann

Telecommunications Institute I, University of Erlangen-Nuremberg, Cauerstrasse 7, 91058 Erlangen, Germany

Received 13 December 1999; received in revised form 16 February 2000

Abstract

Acoustic echo cancellers (AECs) in today's hands-free telephones rely on the assumption of a linear echo path. However, low-cost audio equipment or constraints of portable communication systems cause nonlinear distortions in the loudspeaker and its amplifier, which limit the echo reduction of linear AECs. Such an echo path can be modelled by a memoryless nonlinear function preceding the linear FIR filter. Algorithms for joint adaptation of both stages and stepsize normalizations are derived. As examples for the preprocessor a hard-clipping curve and a polynomial are considered. Fast convergence of the latter is achieved with signal orthogonalization or RLS adaptation. Adaptation stepsize control mechanisms are derived using a novel system distance measure. Experiments under adverse conditions and with real hardware demonstrate robust convergence with both models, and an echo reduction improvement by up to 10 dB at amplitude peaks. For a straightforward implementation, computational cost is increased by factor 1.5–4.5 compared to a linear AEC, but ways for complexity reduction are outlined. © 2000 Elsevier Science B.V. All rights reserved.

Zusammenfassung

Kompensatoren für akustische Echos in heutigen Freisprechanlagen gehen von einem linearen Echopfad aus. Bei preisgünstigen Audioanlagen oder bei portablen Endgeräten werden jedoch häufig nichtlineare Verzerrungen im Lautsprecher und dem zugehörigen Verstärker in Kauf genommen, so dass die Echounterdrückung linearer Kompensatoren zusätzlich begrenzt wird. Solche Echopfade können durch eine gedächtnislose Nichtlinearität vor dem linearen FIR Filter modelliert werden. Algorithmen für eine gemeinsame Adaption und Schrittweitensteuerung der beiden Einheiten werden hier abgeleitet. Als Beispiele für gedächtnislose nichtlineare Modelle werden eine Begrenzerkennlinie und Polynome betrachtet. Für Polynome wird eine schnelle Konvergenz durch Orthogonalisierungsverfahren für das Eingangssignal oder RLS-Adaption erreicht. Mithilfe eines neuen Systemabstandskriteriums werden Kontrollverfahren für die Adaptionsschrittweite abgeleitet. Experimente unter schwierigen Bedingungen und mit realer Hardware bestätigen die robuste Konvergenz bei beiden Modellen und zeigen, dass bei hoher Aussteuerung die Echounterdrückung um bis zu 10 dB verbessert werden kann. Bei direkter Implementierung erhöht sich der Rechenaufwand gegenüber dem linearen Modell um Faktoren zwischen 1.5 und 4.5, es werden jedoch zusätzlich Möglichkeiten zur Verringerung der Komplexität aufgezeigt. © 2000 Elsevier Science B.V. All rights reserved.

Résumé

Les annulateurs d'écho acoustique (AEC) des téléphones mains-libres actuels sont basés sur une hypothèse de llinéarité du chemin d'écho. Toutefois, un équipement audio de prix réduit ou des contraintes sur les systèmes de communication

* Corresponding author. Tel.: + 49-9131-852-7108; fax: + 49-9131-852-8849.
E-mail address: stenger@lnt.de (A. Stenger).

PII: S0165-1684(00)00085-2

portables causent des distortions non-linéaires au niveau du haut-parleur et de son amplificateur, ce qui limite la réduction d'écho par les AEC linéaires. Un tel chemin d'écho peut être modélisé par une fonction non-linéaire sans mémoire précédant le filtre linéaire FIR. Nous dérivons des algorithmes pour l'adaptation conjointe des deux niveaux et la normalisation des pas de mise à jour. Nous considérons comme exemples de prétraitement une courbe des saturation "dure" et un polynôme. Une convergence rapide pour ce dernier est obtenue par orthogonalisation du signal ou une adaptation RLS. Nous dérivons des mécanismes de contrôle des pas d'adaptation à l'aide d'une mesure nouvelle de la distance sur les systèmes. Des expériences réalisées dans des conditions défavorables et avec une implantation matérielle réelle montrent une convergence robuste avec les deux modèles, et une amélioration de la réduction d'écho allant jusqu'à 10 dB aux pics d'amplitude. Pour une implantation directe, le coût de calcul augmente d'un facteur allant de 1.5 à 4.5 vis-à-vis d'un AEC linéaire, mais des approches de réduction de la complexité sont suggérées. © 2000 Elsevier Science B.V. All rights reserved.

1. Introduction

Current state-of-the-art acoustic echo cancellers (AECs) [3] foresee linear adaptive filter structures to model the acoustic path of the loudspeaker–enclosure–microphone system, such as the FIR filter described by its impulse response vector $\hat{\boldsymbol{h}}[k]$ in Fig. 1. Possible adaptation strategies range from the widely used normalized least mean squares (NLMS) to sophisticated implementations of fast recursive least-squares (FRLS) algorithms. Accounting for double-talk and the estimated approximation error relative to the optimum model, a robust stepsize control is required. In many applications, however, the echo path has nonlinear components: Low-cost applications employ small loudspeakers operating beyond their range of linear transduction, and mobile communication terminals may be designed to tolerate clipping of large amplitudes in the amplifier to achieve high sound levels despite low battery voltage.

How does the hands-free communication system of Fig. 1 react to a nonlinear echo path? Let us describe the echo by $y = y_{\mathrm{NL}} + y_{\mathrm{L}}$, and let y_{L} be the portion which can be cancelled by a linear filter. Then the nonlinear echo component $y_{\mathrm{NL}}[k]$ will be transmitted to the far-end subscriber. The problem is stated analytically using the 'nonlinear echo power' $\mathscr{E}\{y_{\mathrm{NL}}^2\}$ or the 'nonlinear-to-linear echo power ratio (NLR)' $\mathscr{E}\{y_{\mathrm{NL}}^2\}/\mathscr{E}\{y_{\mathrm{L}}^2\}$, where $\mathscr{E}\{\bullet\}$ represents the statistical expectation operator.

How much nonlinear distortion can be expected in practical applications? Hands-free communication terminals will always be designed to provide satisfactory sound quality to the near-end listener. Nonlinear distortion in speech signals is generally not audible, as long as the total harmonic distortion of an audio system is less than 10% [18]. This has to be interpreted carefully, as the total harmonic distortion does not exactly equal the NLR for two reasons:

(1) As speech does not only excite a single frequency, nonlinearities will contribute wideband distortion products which are not captured by measuring harmonic distortion.

(2) If $\hat{\boldsymbol{h}}[k]$ is adapted with the goal of minimizing the mean square of $e[k]$, the uncompensatible portion y_{NL} is uncorrelated with y_{L} [8], which is not necessarily true for a fundamental frequency and all its higher-order harmonics. Therefore, y_{L} as defined above contains a nonlinear signal portion, which in contrast to harmonic distortion measures is not counted as nonlinear distortion within the NLR.

Considering both effects, we assume that the total harmonic distortion roughly approximates the NLR, so that speech signals with only NLR $= -15$ to -10 dB can still be considered to have

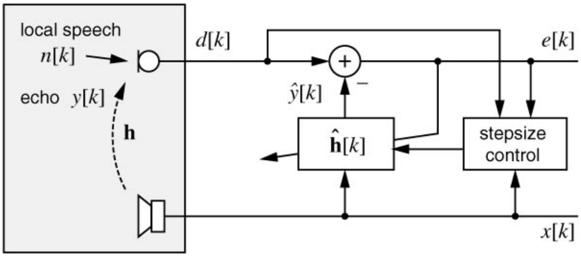

Fig. 1. AEC with linear adaptive filter.

satisfactory quality. This range has been experimentally confirmed for four different hands-free car kits, as well as for a small low-cost loudspeaker driven by a one-chip amplifier. In these experiments, the NLR was computed using an NLMS adaptive FIR filter using a very large number of taps and small stepsize. After convergence, the remaining echo power equals $\mathscr{E}\{y_{\mathrm{NL}}^2\}$ assuming that no local noise or interfering signal is present.

After discussing an appropriate model for the nonlinear echo path and presenting a structure with an adaptive nonlinear preprocessor in Section 2, appropriate adaptation algorithms are derived in Section 3 with emphasis on stepsize normalization and fast convergence of the preprocessor coefficients. Two examples for the adaptive preprocessor are discussed in detail: A polynomial model and a hard-clipping curve with a single adaptive parameter. In Section 4, system distance measures are introduced for both examples, which are used for deriving stepsize control mechanisms in Section 5. The adaptation speed of the polynomial preprocessor is experimentally investigated in Section 6.1. In Section 6.2 the appropriateness of both the polynomial and the hard-clipping model is verified using real hardware, before the new adaptation techniques including their stepsize control are tested with severe local distortion in Section 6.3. Finally, the computational complexity of the presented adaptation techniques is compared.

2. Models for the nonlinear echo path

Dominant nonlinearities are found in the loudspeaker and its amplifier (section (B) in Fig. 2) so that the nonlinear behaviour of the A/D and D/A converters can be neglected, which allows us to model them as LTI systems. Even cheap microphones are linear because of small signal amplitudes. The acoustic path is linear but highly time-variant (LTV), and the nonlinearities are weakly time-variant (wTV) as they vary mainly with temperature drift. To assume wTV nonlinearities has the additional advantage that the nonlinear model can adapt to component tolerances.

Nonlinearities in section (B) can be roughly classified as nonlinearities with and without memory. The latter manifest itself typically as saturation curves in the power amplifier driving the loudspeaker, or in the loudspeaker itself.

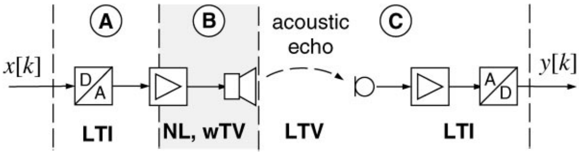

Fig. 2. Acoustic echo path with nonlinearities.

If not driven into saturation, dynamic loudspeakers cause nonlinearities with memory, since the time constants of their electro-mechanical systems are large compared to the sampling rate. Second- and third-order nonlinearities due to inhomogeneous magnetic flux and nonlinear stiffness of the suspension dominate [13], and the total harmonic distortion is only in the order of 1% except at low frequencies [5], so that a compensation is required only for high-quality applications.

An acoustic echo canceller for the compensation of nonlinearities with memory has been presented in [1]. A cascade of a neural network and an FIR filter is employed, but an extra reference microphone at the loudspeaker is required. Another approach successfully models the complete echo path with an adaptive second-order Volterra filter [16], thus avoiding the need for an extra microphone. Although the numerical expense is justified for high-quality applications, it is not well suited for saturation effects in low-cost applications.

For this case, a higher-order nonlinear model without memory is appropriate. It has been shown in [14,15] that such nonlinearities can be effectively reduced using a seventh-order polynomial model of the nonlinear section (B) and a pure delay for part (A). As the polynomial in [15] is not adaptive, and the NLMS adaptation proposed in [14] exhibits very slow convergence for seventh-order polynomials, a fast and robust adaptation scheme for the polynomial preprocessor is desirable.

In [10], parts (A) and (C) of Fig. 2 are modelled with linear adaptive FIR filters and part (B) is realized by a saturation curve with one adaptive parameter. Difficulties in predicting adaptation behaviour are blamed on local minima of the error

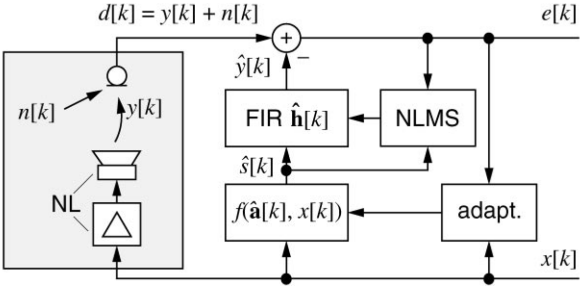

Fig. 3. Nonlinear AEC with memoryless preprocessor.

surface [10], and as no stepsize normalization is proposed, its stability depends on the input signal power. With a special initialization procedure of the adaptive coefficients, as much as 8 dB ERLE improvement over a linear adaptive filter are reported.

To avoid local minima on the error surface, we will restrict ourselves to a two-stage cascade as shown in Fig. 3. Part (B) is modelled by a memoryless nonlinear function f and part (A) is assumed to be a delay and can therefore be exchanged with the nonlinearity. It is modelled together with (C) by the linear FIR filter $\hat{\boldsymbol{h}}[k]$. The nonlinear function f has to model the saturation effects found in the loudspeaker and amplifier. Therefore, it must be monotonically increasing and should exhibit saturation behaviour for high-input amplitudes.

3. Adaptation algorithms

Using a common framework, we first derive a joint normalized LMS (NLMS) adaptation for both stages. Then, f is specialized to a saturation curve as proposed in [10] and the polynomial model of [14]. To speed up the convergence of the polynomial, an orthogonalization technique and an RLS adaptation are developed.

Fig. 3 shows an adaptive FIR filter

$$\hat{\boldsymbol{h}}[k] = [\hat{h}_0[k], \ldots, \hat{h}_{N-1}[k]]^{\mathrm{T}} \tag{1}$$

preceded by a memoryless nonlinear function $f(\hat{\boldsymbol{a}}[k], x[k])$, whose parameters are captured in the $P \times 1$ column vector

$$\hat{\boldsymbol{a}}[k] = [\hat{a}_1[k], \ldots, \hat{a}_P[k]]^{\mathrm{T}}. \tag{2}$$

As there is no reference signal available at the output of the loudspeaker, the adaptation of both stages in Fig. 3 has to rely on the joint error signal $e[k]$. In Section 3.1 a joint adaptation scheme for $\hat{\boldsymbol{h}}$ and $\hat{\boldsymbol{a}}$ will be derived for a general f. Then in Sections 3.2 and 3.3 two examples for the nonlinear function are examined. A stepsize normalization and the initialization of the adaptive parameters is discussed with both examples, and for the adaptive polynomial two methods for accelerating the convergence speed are presented.

3.1. General parametric nonlinear function

Denoting the output of the nonlinear function by

$$\hat{s}[k] = f(\hat{\boldsymbol{a}}[k], x[k]) \tag{3}$$

and combining its N latest values in the vector

$$\hat{\boldsymbol{s}}[k] = [\hat{s}[k], \hat{s}[k-1], \ldots, \hat{s}[k-N+1]]^{\mathrm{T}}, \tag{4}$$

the estimated echo reads

$$\hat{y}[k] = \hat{\boldsymbol{h}}^{\mathrm{T}}[k]\hat{\boldsymbol{s}}[k]. \tag{5}$$

If $\hat{\boldsymbol{a}}[k]$ is time-invariant, the same result is obtained by

$$\hat{y}[k] = \hat{\boldsymbol{h}}^{\mathrm{T}}[k]\boldsymbol{f}(\hat{\boldsymbol{a}}[k], \boldsymbol{x}[k]) \tag{6}$$

with the $N \times 1$ vectors

$$\boldsymbol{x}[k] = [x[k], x[k-1], \ldots, x[k-N+1]]^{\mathrm{T}} \tag{7}$$

and

$$\boldsymbol{f}(\hat{\boldsymbol{a}}[k], \boldsymbol{x}[k]) = [f(\hat{\boldsymbol{a}}[k], x[k]), \ldots, f(\hat{\boldsymbol{a}}[k], x[k-N+1])]^{\mathrm{T}}. \tag{8}$$

In [7] an LMS-type adaptation for transversal filters is derived by forming the gradient of $e^2[k]$ with respect to the adaptive coefficients. Applying this procedure to the cascaded system described by (6), we obtain

$$\nabla_h[k] = \frac{\partial e^2[k]}{\partial \hat{\boldsymbol{h}}[k]} = -2e[k]\hat{\boldsymbol{s}}[k], \tag{9}$$

$$\nabla_a[k] = \frac{\partial e^2[k]}{\partial \hat{\boldsymbol{a}}[k]} = -2e[k]\boldsymbol{f}'(\hat{\boldsymbol{a}}[k]^{\mathrm{T}}, \boldsymbol{x}[k])^{\mathrm{T}}\hat{\boldsymbol{h}}[k] \tag{10}$$

with the $N \times P$ matrix

$$\boldsymbol{f}'(\hat{\boldsymbol{a}}[k]^{\mathrm{T}},\boldsymbol{x}[k]) = \frac{\partial \boldsymbol{f}(\hat{\boldsymbol{a}}[k]^{\mathrm{T}},\boldsymbol{x}[k])}{\partial \hat{\boldsymbol{a}}[k]^{\mathrm{T}}} = \left[\frac{\partial f(\hat{\boldsymbol{a}}[k],x[k])}{\partial \hat{\boldsymbol{a}}[k]},\dots,\frac{\partial f(\hat{\boldsymbol{a}}[k],x[k-N+1])}{\partial \hat{\boldsymbol{a}}[k]}\right]^{\mathrm{T}}. \quad (11)$$

With the abbreviation

$$\boldsymbol{u}[k] = \boldsymbol{f}'(\hat{\boldsymbol{a}}[k]^{\mathrm{T}},\boldsymbol{x}[k])^{\mathrm{T}}\hat{\boldsymbol{h}}[k] \quad (12)$$

(10) becomes

$$\nabla_a = -2\boldsymbol{u}[k]e[k]. \quad (13)$$

If the coefficient vectors are updated with stepsizes μ_h and μ_a, an LMS-type adaptation algorithm results:

$$\hat{\boldsymbol{h}}[k+1] = \hat{\boldsymbol{h}}[k] - \frac{\mu_h}{2}\nabla_h = \hat{\boldsymbol{h}}[k] + \mu_h\hat{\boldsymbol{s}}[k]e[k] \quad (14)$$

and

$$\hat{\boldsymbol{a}}[k+1] = \hat{\boldsymbol{a}}[k] - \frac{\mu_a}{2}\nabla_a = \hat{\boldsymbol{a}}[k] + \mu_a\boldsymbol{u}[k]e[k]. \quad (15)$$

As for time-invariant $\hat{\boldsymbol{a}}$ the adaptation of $\hat{\boldsymbol{h}}$ in (14) is similar to the classical case (see, e.g., [7]), it is normalized according to

$$\mu_h = \frac{\alpha_h}{\|\hat{\boldsymbol{s}}[k]\|_2^2}. \quad (16)$$

Stable convergence of the average tap weight error $\mathscr{E}\{\hat{\boldsymbol{h}}[k] - \boldsymbol{h}\}$ and the average mean squared error can be proven for $0 < \alpha_h < 2$ on the basis of the "independence assumption" [7]. As the normalization of the preprocessor depends on the specific realization of f, we distinguish two different cases in the following sections. Table 1 summarizes the general form of a joint NLMS-type adaptation of both stages.

Table 1
Joint NLMS adaptation of preprocessor and FIR filter

1.	$\boldsymbol{x}[k] = [x[k], x[k-1], \dots, x[k-N+1]]^{\mathrm{T}}$
2.	$\hat{s}[k] = f(\hat{\boldsymbol{a}}[k], x[k])$
3.	$\hat{\boldsymbol{s}}[k] = [\hat{s}[k],\hat{s}[k-1],\dots,\hat{s}[k-N+1]]^{\mathrm{T}}$
4.	$e[k] = d[k] - \hat{\boldsymbol{h}}^{\mathrm{T}}[k]\boldsymbol{f}(\hat{\boldsymbol{a}}[k],\boldsymbol{x}[k])$
5.	$\hat{\boldsymbol{a}}[k+1] = \hat{\boldsymbol{a}}[k] + \mu_a(\boldsymbol{f}'(\hat{\boldsymbol{a}}[k]^{\mathrm{T}},\boldsymbol{x}[k]))^{\mathrm{T}}\hat{\boldsymbol{h}}[k]e[k]$
6.	$\hat{\boldsymbol{h}}[k+1] = \hat{\boldsymbol{h}}[k] + \frac{\alpha_h}{\|\hat{\boldsymbol{s}}[k]\|_2^2}\hat{\boldsymbol{s}}[k]e[k]$

3.2. Polynomial preprocessor

The nonlinear preprocessing can be described by a Pth-order polynomial

$$f(\hat{\boldsymbol{a}}[k], x[k]) = \sum_{p=1}^{P} \hat{a}_p[k]x^p[k]. \quad (17)$$

Introducing the abbreviations

$$\boldsymbol{x}_p[k] = [x[k], x^2[k], \dots, x^P[k]]^{\mathrm{T}} \quad (18)$$

and

$$\boldsymbol{X}_p[k] = [\boldsymbol{x}_p[k],\boldsymbol{x}_p[k-1], \dots\ \boldsymbol{x}_p[k-N+1]]^{\mathrm{T}}, \quad (19)$$

Eqs. (8), (11) and (12) become

$$\boldsymbol{f}(\hat{\boldsymbol{a}}[k],\boldsymbol{x}[k]) = \boldsymbol{X}_p[k]\hat{\boldsymbol{a}}[k], \quad (20)$$

$$\boldsymbol{f}'(\hat{\boldsymbol{a}}[k],\boldsymbol{x}[k]) = \boldsymbol{X}_p[k] \quad (21)$$

and

$$\boldsymbol{u}[k] = \boldsymbol{X}_p[k]^{\mathrm{T}}\hat{\boldsymbol{h}}[k]. \quad (22)$$

Thus, steps 2–4 of Table 1 specialize to

$$\hat{s}[k] = \boldsymbol{x}_p[k]^{\mathrm{T}}\hat{\boldsymbol{a}}[k] \quad (23)$$

and

$$e[k] = d[k] - \hat{\boldsymbol{a}}[k]^{\mathrm{T}}\boldsymbol{u}[k]. \quad (24)$$

3.2.1. Stepsize normalization

Similar to μ_h in Section 3.1, the preprocessor adaptation is normalized with respect to $\|\boldsymbol{u}[k]\|_2^2$. Stable convergence of $\mathscr{E}\{\boldsymbol{a} - \hat{\boldsymbol{a}}[k]\}$ and the average mean squared error signal can be proven in a similar way as in [7] for $\mathscr{E}\{\boldsymbol{h} - \hat{\boldsymbol{h}}[k]\}$, as $\hat{y}[k]$ linearly depends on the adaptive coefficients $\hat{\boldsymbol{a}}[k]$. This is possible, as only a linear combination and not a transversal filter structure is required [7,17]. As the independence assumptions of [7] do not hold in practice, this result must be carefully applied. In contrast to $||\hat{\boldsymbol{s}}[k]||_2^2$, the value $||\boldsymbol{u}[k]||_2^2$ becomes very small, when only few adjacent samples of the input $x[k]$ are close to zero. From (22), it follows that this effect becomes stronger the more the energy of $\hat{\boldsymbol{h}}[k]$ is concentrated in a small time region. To avoid a large adaptation step towards a random direction in this situation, a positive constant δ_u is added to the normalization term, so that step 5 of Table 1 finally reads

$$\hat{\boldsymbol{a}}[k+1] = \hat{\boldsymbol{a}}[k] + \frac{\alpha_a}{||\boldsymbol{u}[k]||_2^2 + \delta_u}\boldsymbol{u}[k]e[k]. \tag{25}$$

With an amplitude range of ± 1 for $x[k]$, $\delta_u = 0.1$–1 is a reasonable choice.

3.2.2. Initialization

Usually the adaptive FIR filter is initialized with $\hat{\boldsymbol{h}}[0] = \boldsymbol{0}$. From (22) and (25) it can be seen that then $\hat{\boldsymbol{a}}[k]$ is not adapted at time $k = 0$. Similarly, $\hat{\boldsymbol{h}}[k]$ is not adapted when $\hat{\boldsymbol{a}}[k] = \boldsymbol{0}$. Therefore, the system is trapped in the initialization state, if both $\hat{\boldsymbol{h}}[0] = \boldsymbol{0}$ and $\hat{\boldsymbol{a}}[0] = \boldsymbol{0}$. As we propose $\hat{\boldsymbol{h}}[0] = \boldsymbol{0}$, $\hat{\boldsymbol{a}}[0]$ has to be initialized other than zero. A natural and successful choice is *linear* initialization, i.e., $\hat{\boldsymbol{a}}[0] = [1,0,\dots,0]$. This ensures that even if $\hat{\boldsymbol{a}}$ is not adapted, the system performs at least as well as a linear AEC.

3.2.3. Orthogonalization of the polynomial

A polynomial model for saturation effects in nonlinear acoustic echo paths has to be of order seven or higher [15]. However, the NLMS adaptation of more than three polynomial coefficients would be too slow for practical applications because of the strong mutual correlation of $x[k]$, $x^2[k], \dots, x^P[k]$, i.e., the components of $\boldsymbol{x}_p[k]$.[1]

To circumvent this problem, the vector $\boldsymbol{x}_p[k]$ can be transformed into a vector $\boldsymbol{x}_{p,o}[k] = [x_{1,o}[k], x_{2,o}[k], \dots, x_{P,o}[k]]$ with orthogonal components

$$\mathscr{E}\{x_{i,o}[k]x_{j,o}[k]\} = 0, \quad i \neq j. \tag{26}$$

This problem is equivalent to the QR factorization problem [6], if we collect the vectors $\boldsymbol{x}_p[k] \dots \boldsymbol{x}_p[k-P+1]$ in a $(P \times P)$ matrix. Therefore, an orthogonalization can be achieved by

$$\boldsymbol{x}_{p,o}[k] = \boldsymbol{A}[k]\boldsymbol{x}_p[k] \tag{27}$$

with the lower triangular $P \times P$ matrix

$$\boldsymbol{A} = \begin{bmatrix} 1 & & & 0 \\ a_{21} & 1 & & \\ \vdots & & \ddots & \\ a_{P1} & a_{P2} & \cdots & 1 \end{bmatrix}. \tag{28}$$

The output of the preprocessor is then

$$\hat{s}[k] = \boldsymbol{x}_{p,o}[k]^{\mathrm{T}}\hat{\boldsymbol{a}}_o[k] \tag{29}$$

with a new coefficient vector $\hat{\boldsymbol{a}}_o[k]$, which results from NLMS adaptation, if $\boldsymbol{x}_p[k]$ is replaced by $\boldsymbol{x}_{p,o}[k]$. Comparing (29) and (23) yields the relation between $\hat{\boldsymbol{a}}_o[k]$ and $\hat{\boldsymbol{a}}[k]$

$$\boldsymbol{x}_p^{\mathrm{T}}\hat{\boldsymbol{a}} = (\boldsymbol{A}\boldsymbol{x}_p)^{\mathrm{T}}\hat{\boldsymbol{a}}_o = \boldsymbol{x}_p^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}\hat{\boldsymbol{a}}_o \;\Rightarrow\; \hat{\boldsymbol{a}} = \boldsymbol{A}^{\mathrm{T}}\hat{\boldsymbol{a}}_o. \tag{30}$$

For determining $\boldsymbol{A}$, we assume that $x[k]$ is Laplacian distributed, as can be done for speech signals [12]. Further assuming stationarity, $\boldsymbol{A}$ is time-invariant and can be pre-calculated from the moments of $x[k]$. We determine the elements of $\boldsymbol{A}$ recursively as follows: The $i-1$ unknown elements of the ith row can be calculated from $i-1$ orthogonality conditions arising from (26), which says that $x_{i,o}[k]$ has to be orthogonal to the signals $x[k], x^2[k], \dots, x^{i-1}[k]$. In detail, these orthogonality conditions are

$$\mathscr{E}\{x_{i,o}[k].x[k]\} = 0, \mathscr{E}\{x_{i,o}[k].x^2[k]\} = 0$$
$$\vdots \mathscr{E}\{x_{i,o}[k].x^{i-1}[k]\} = 0 \tag{31}$$

and contain only moments of $x[k]$. For Laplacian distributed $x[k]$ with zero mean and variance σ_x^2, i.e., $p_x(x) = (1/\sqrt{2}\sigma_x)\mathrm{e}^{-(\sqrt{2}/\sigma_x)|x|}$, the moments are

$$m_x^{(i)} = \begin{cases} 0 & i \text{ odd,} \\ (\frac{\sigma}{\sqrt{2}})^2 i! & i \text{ even} \end{cases} \tag{32}$$

[1] Considering the joint moment $\mathscr{E}\{x^k y^r\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} u^k v^r p_{xy}(u,v)\,\mathrm{d}u\,\mathrm{d}v$ defined in [11] we set $y^r = x^r$ and obtain for the correlation between two powers of x $\mathscr{E}\{x^{(k+r)}\} = \int_{-\infty}^{\infty} u^{(k+r)} p_x(u)\,\mathrm{d}u$ which is positive if $(k+r)$ is even, as then $u^{(k+r)} \geqslant 0$, while $p_x(u)$ is always nonnegative. However, for a symmetric density, $p_x(-u) = p_x(u)$, and $(k+r)$ being odd, we have $\mathscr{E}\{x^{(k+r)}\} = 0$, so that the powers of $x[k]$ are uncorrelated.

and $\boldsymbol{A}$ becomes

$$\boldsymbol{A} = \begin{bmatrix} 1 & & & & & \dots & 0 \\ 0 & 1 & & & & & \\ -6\sigma_x^2 & 0 & 1 & & & & \\ 0 & -15\sigma_x^2 & 0 & 1 & & & \\ 130\sigma_x^4 & 0 & -\frac{110}{3}\sigma_x^2 & 0 & 1 & & \\ \vdots & & & & & & \ddots \end{bmatrix}. \tag{33}$$

The matrix $\boldsymbol{A}$ has only few nonzero elements. Therefore, only little extra computational effort is required for the orthogonalization using a fixed matrix $\boldsymbol{A}$.

3.2.4. RLS adaptation of the polynomial

As the output of the preprocessor is a linear function of the adaptive coefficients $\hat{\boldsymbol{a}}$, least-squares (LS) techniques can be applied. With

$$\boldsymbol{U}[k] = [\boldsymbol{u}[0], \boldsymbol{u}[1], \dots, \boldsymbol{u}[k]]^{\mathrm{T}} \tag{34}$$

and

$$\boldsymbol{d}[k] = [d[0], d[1], \dots, d[k]]^{\mathrm{T}}, \tag{35}$$

the LS problem for the coefficients $\hat{\boldsymbol{a}}[k]$ reads

$$\|\boldsymbol{U}[k]\hat{\boldsymbol{a}}[k] - \boldsymbol{d}[k]\|_2^2 = \sum_{\kappa=0}^{k} e^2[\kappa] \to \min, \tag{36}$$

where $\boldsymbol{U}[k]$ is a $(k+1) \times P$ matrix, and $\hat{\boldsymbol{a}}[k]$ is given by the pseudo-inverse [9]

$$\hat{\boldsymbol{a}}[k] = \underbrace{(\boldsymbol{U}[k]^{\mathrm{T}}\boldsymbol{U}[k])^{-1}}_{\boldsymbol{P}[k]} \underbrace{\boldsymbol{U}[k]^{\mathrm{T}}\boldsymbol{d}[k]}_{\theta[k]}. \tag{37}$$

In [15] time-invariant polynomial preprocessor coefficients are determined offline using (37).

The recursive least-squares (RLS) algorithm [7] calculates $\boldsymbol{P}[k]$ and $\theta[k]$ recursively avoiding the explicit inversion of $\boldsymbol{U}[k]^{\mathrm{T}}\boldsymbol{U}[k]$. Thereby, the LS estimation is equipped with a leaky memory, i.e., $\sum_{\kappa=0}^{k} \lambda^{k-\kappa} e^2[\kappa]$ is minimized instead of (36), so that time-varying systems can be tracked as well. To avoid numerical instabilities, $\boldsymbol{P}[k]$ is periodically re-initialized with $\boldsymbol{I} \cdot \delta^{-1}$, where δ is a small positive constant. The complete algorithm reads like Table 1, if steps 4 and 5 are replaced by Table 2.[2] Note that step 4 calculates the a priori error of the RLS algorithm now, while the a posteriori error is denoted by $e[k]$.

Table 2
Upgrade: RLS adaptation of preprocessor

4.	$e_{\mathrm{pri}}[k] = d[k] - \boldsymbol{u}[k]^{\mathrm{T}}\hat{\boldsymbol{a}}[k]$
5a.	$\boldsymbol{v}[k] = \lambda^{-1}\boldsymbol{P}[k]\boldsymbol{u}[k]$
5b.	$\boldsymbol{g}[k] = \dfrac{\boldsymbol{v}[k]}{1 + \boldsymbol{u}[k]^{\mathrm{T}}\boldsymbol{v}[k]}$
5c.	$\boldsymbol{P}[k] = \lambda^{-1}\boldsymbol{P}[k] - \boldsymbol{g}[k]\boldsymbol{v}[k]^{\mathrm{T}}$
5d.	$\hat{\boldsymbol{a}}[k+1] = \hat{\boldsymbol{a}}[k] + \boldsymbol{g}[k]e_{\mathrm{pri}}[k]$
5e.	$e[k] = \boldsymbol{u}[k]^{\mathrm{T}}\hat{\boldsymbol{a}}[k+1]$

3.3. Hard-clipping preprocessor

The nonlinear function $f(\hat{\boldsymbol{a}}, x)$ of Fig. 3 is now defined as

$$f(\hat{a}, x) = \begin{cases} -\hat{a} & x \leqslant -\hat{a}, \\ x & |x| < \hat{a}, \\ \hat{a} & x \geqslant \hat{a} \end{cases} \tag{38}$$

and thus depends only on a single adaptive parameter $\hat{a}$. Differentiation with respect to $\hat{a}$ yields

$$f'(\hat{a}, x) = \begin{cases} -1 & x \leqslant -\hat{a}, \\ 0 & |x| < \hat{a}, \\ 1 & x \geqslant \hat{a}. \end{cases} \tag{39}$$

With (39) the matrix $\boldsymbol{f}_r'$ in (11) becomes an $N \times 1$ vector with elements 1, 0, or -1.

3.3.1. Stepsize normalization

As the output of the preprocessor does not linearly depend on the adaptive coefficient $\hat{a}$, stable adaptation cannot be proven for certain stepsize

[2] Relative to the nomenclature in [7], our time index k is increased by 1.

bounds similar as for the polynomial. Instead, we propose a stepsize normalization which shares one important property with the one derived in [7]: it makes the adaptation stepsize independent of the power of the input signal. The input signal to the preprocessor adaptation is $\boldsymbol{u}[k]$, which is a scalar here. Therefore, its average power cannot be reliably estimated by its L_2-norm. Also a short-time average of $u[k]$ should not be used, as it might lead to divergence at times when the power of $x[k]$ suddenly increases. In such cases, the normalization term would be too small due to delayed power estimation resulting in a large adaptation step. A robust alternative is to estimate an upper bound for the power of $u[k]$ exploiting that f' takes on only the values -1, 0, and 1

$$(\boldsymbol{f}'(\hat{a}[k],\boldsymbol{x}[k])^{\mathrm{T}}\hat{\boldsymbol{h}}[k])^2 \leqslant \left(\sum_{n=0}^{N-1} |\hat{h}_n[k]|\right)^2. \tag{40}$$

This bound is very conservative, as the function $\boldsymbol{f}'(\hat{a}[k],\boldsymbol{x}[k])$ is nonzero only if $x[k]$ lies outside the linear range. Therefore, we neglect the cross-terms in the sum on the right-hand side of (40) and obtain

$$\mu_a = \frac{\alpha_a}{\|\hat{\boldsymbol{h}}[k]\|_2^2}. \tag{41}$$

Inserting (38), (39), and (41) into Table 1 leads to the complete algorithm.

3.3.2. Initialization

As $f'(\hat{a},x)$ is always zero if $\hat{a}$ is outside the amplitude range of $x[k]$, the initial value of $\hat{a}$ must be smaller than the saturation threshold of any possible unknown system. Unfortunately, this inhibits a linear initialization and leads to additional distortion of the transmitted signal $e[k]$ as long as $\hat{a}$ is not yet adapted.

4. Distance measures

Signal-independent measures of the adaptation quality are useful for convergence analysis and parameter optimization of adaptive algorithms. For linear AEC often

$$D_h[k] = 10\log_{10}\frac{\|\boldsymbol{h}-\hat{\boldsymbol{h}}[k]\|_2^2}{\|\boldsymbol{h}\|_2^2} \tag{42}$$

is used, which equals

$$\mathrm{ERLE} = 10\log_{10}\frac{\mathscr{E}\{y[k]^2\}}{\mathscr{E}\{\varepsilon[k]^2\}} \tag{43}$$

for white $x[k]$.

To analyse the convergence of the overall system, a joint distance measure of both stages is required. Furthermore, for development of a preprocessor stepsize control, we need to measure the system distance between $\hat{\boldsymbol{a}}$ and $\boldsymbol{a}$.

4.1. Polynomial preprocessor

In Section 3.2.1 stable convergence of $\mathscr{E}\{\boldsymbol{a}-\hat{\boldsymbol{a}}[k]\}$ and $\mathscr{E}\{\boldsymbol{h}-\hat{\boldsymbol{h}}[k]\}$ was referred to. This motivates us to measure the adaptation state of the preprocessor with

$$D_a^{(\mathrm{p})}[k] = 10\log_{10}\frac{\|\boldsymbol{a}-\hat{\boldsymbol{a}}[k]\|_2^2}{\|\boldsymbol{a}\|_2^2}. \tag{44}$$

A way to construct a joint distance measure is to generalize it from both these measures, so that the joint measure $D^{(\mathrm{p})}$ equals $D_a^{(\mathrm{p})}$ for $\hat{\boldsymbol{h}}[k]=\boldsymbol{h}$ and D_h for $\hat{\boldsymbol{a}}[k]=\boldsymbol{a}$. The superscript (p) stands for ‘polynomial’. This is achieved in the following way: All product pairs of elements of $\boldsymbol{a}$ and $\boldsymbol{h}$ are combined in a $NP\times 1$ vector

$$\boldsymbol{c} = \boldsymbol{h}\otimes\boldsymbol{a}$$

$$= [h_1a_1, h_1a_2, \ldots, h_1a_P, h_2a_1, \ldots h_{N-1}a_P]^{\mathrm{T}}, \tag{45}$$

where $\otimes$ denotes the Kronecker product [4]. With the same notation for the adaptive coefficients

$$\hat{\boldsymbol{c}}[k] = \hat{\boldsymbol{h}}[k]\otimes\hat{\boldsymbol{a}}[k], \tag{46}$$

we can define a joint system distance for the polynomial preprocessor and the FIR filter

$$D^{(\mathrm{p})}[k] = 10\log_{10}\frac{\|\boldsymbol{c}-\hat{\boldsymbol{c}}[k]\|_2^2}{\|\boldsymbol{c}\|_2^2}. \tag{47}$$

Unlike $D_h[k]$, $D^{(\mathrm{p})}[k]$ can never be equal to ERLE, regardless of the statistical assumptions on $x[k]$. However, it can be shown that ERLE and $D^{(\mathrm{p})}$ attain the same order of magnitude as long as the moments of x are also in the same order of magnitude. This reflects the fact that $D^{(\mathrm{p})}$ behaves as if all moments were equal, whereas ERLE directly depends on the moments.

4.2. Hard-clipping preprocessor

A simple distance measure which does not require any a priori knowledge of the input signal or the system is

$$D^{(c)}[k] = 10\log_{10}\left(\frac{\|a\boldsymbol{h} - \hat{a}[k]\hat{\boldsymbol{h}}[k]\|_2^2}{a^2\|\boldsymbol{h}\|_2^2}\right). \tag{48}$$

It delivers the squared coefficient distance $(a-\hat{a})^2/a^2$ for the case of ideally adapted FIR filter.

Compared to ERLE, this is rather pessimistic, as at times where x is in the linear range of the system the measure $D_h < D^{(c)}$ describes the situation more exactly.

If the linear portion of the amplitude range of $x[k]$ is known, we can define a more intuitive distance measure: It weighs a linear and a nonlinear distance measure with the probability that nonlinear distortion occurs,

$$D^{(c)}[k] = 10\log_{10}\left(\frac{\|\boldsymbol{h} - \hat{\boldsymbol{h}}[k]\|_2^2}{\|\boldsymbol{h}\|_2^2}\bar{K} + \frac{\|a\boldsymbol{h} - \hat{a}[k]\hat{\boldsymbol{h}}[k]\|_2^2}{a^2\|\boldsymbol{h}\|_2^2}K\right), \tag{49}$$

where $\bar{K}$ is the probability that $x[k]$ is below the clipping margin:

$$\bar{K} = P(|x| < a) = \int_{-a}^{a} p_x(x)\,\mathrm{d}x \tag{50}$$

and $K = 1 - \bar{K}$. This measure proves to be approximately congruent with ERLE, if $x[k]$ is white and $\hat{a}[k] \approx a$.

5. Stepsize control

With linear AEC, the optimal stepsize for white Gaussian noise excitation,

$$\alpha_{\text{opt}}[k] = \frac{\mathscr{E}\{\varepsilon^2[k]\}}{\mathscr{E}\{\varepsilon^2[k]\} + \mathscr{E}\{n^2[k]\}} \tag{51}$$

minimizes the system distance $D_h[k+1]$ in the next time step [2], where $\varepsilon[k] = e[k] - n[k]$ denotes the residual echo. For the difficult task of estimating $\mathscr{E}\{\varepsilon[k]^2\}$ many techniques are known [3].

If the residual echo contains nonlinear components, this concept must be extended, as nonlinear echo components disturb the adaptation of $\hat{\boldsymbol{h}}$ and linear echo components disturb the adaptation of $\hat{\boldsymbol{a}}$.

It can be shown that $\alpha_h[k] = \alpha_{\text{opt}}[k]$ minimizes $D_h[k+1]$ in the cascaded system of Fig. 3, if the preprocessor is ideally adapted. Similarly, $\alpha_a[k] = \alpha_{\text{opt}}[k]$ minimizes the polynomial coefficient distance $D_a^{(p)}[k+1]$ if $\hat{\boldsymbol{h}} = \boldsymbol{h}$ [9]. For the saturation parameter of Section 3.3, this cannot be proven in a similar way, but as $\alpha_{\text{opt}}[k]$ serves as a double-talk detector, it is a useful basis for the control of its adaptation stepsize, too.

Therefore, only a modification of an existing stepsize control is required. The FIR filter stepsize has to be lowered at high amplitudes of $x[k]$, as only then relevant nonlinear echo components appear. As proposed in [14], we set

$$\alpha_h[k] = \alpha_{\text{opt}}[k]\frac{x_{\max}[k] - \bar{x}[k]}{x_{\max}[k]} \tag{52}$$

with the short-time peak level

$$\bar{x}[k] = \begin{cases} |x[k]|, & |x[k]| \geqslant \bar{x}[k-1], \\ \beta\bar{x}[k-1] + (1-\beta)|x[k]|, & |x[k]| < \bar{x}[k-1] \end{cases} \tag{53}$$

and the maximum value of the input signal up to time k

$$x_{\max}[k] = \max\{|x[\kappa]|\}, \quad \kappa \in [0, \dots, k]. \tag{54}$$

The averaging time is chosen to be $\beta = 1 - 1/N$ according to the duration N of the impulse response $\hat{\boldsymbol{h}}$, as nonlinear distortions at the loudspeaker travel through the room within this period, and hence are contained in the echo.

Due to the time-variance and the infinite length of the real room impulse response there is always a linear residual echo, which generally exceeds the power of the nonlinear residual echo, if the preprocessor has converged to some degree. In consequence, the pre-filter adaptation is severely disturbed even in absence of a local signal. Therefore, α_a should be chosen about 2 to 5 times smaller than $\alpha_{\text{opt}}[k]$ in case of the polynomial preprocessor. For the saturation parameter the stepsize should be even smaller, as $\alpha_{\text{opt}}[k]$ is not proven to be optimal for this case, and the adaptation of a single parameter is fast enough even with a small stepsize.

To achieve robust RLS adaptation of the polynomial, λ should be close to 1 during double-talk or if the excitation $x[k]$ is low. Therefore, we propose to control the forgetting factor depending on α_{opt} as follows:

$$\lambda[k] = 1 - \alpha_{\text{opt}}[k](1 - \lambda_{\min}), \tag{55}$$

where $\lambda_{\min}$ represents the minimum forgetting factor. As a typical value for robust operation in AEC environments we use $\lambda_{\min} = 0.995$ in our experiments. Furthermore, the RLS adaptation is halted if the short-time peak level of the excitation is below a certain threshold, which we set to $\bar{x} < x_{\max}/2$.

The sensitivity of the preprocessor to the system mismatch of the linear model makes a special startup procedure necessary: the adaptation of the preprocessor should be disabled, until $\hat{\boldsymbol{h}}$ has converged to some degree. A similar startup-strategy is reported in [10]. Some stepsize control mechanisms provide an estimate of the FIR filter alignment [3], which could be used to scale down the preprocessor stepsize not only after initialization but also after severe room changes.

6. Results

The convergence speed of the polynomial with NLMS adaptation, NLMS adaptation with orthogonalization, and RLS adaptation, respectively, is experimentally compared in Section 6.1 using ERLE and the system distance measure. To examine the appropriateness of both the polynomial and the hard-clipping model, in Section 6.2 signals recorded from low-cost hardware are used in simulations without local distortion. Then, in Section 6.3, the new adaptation techniques including their stepsize control are tested using the same recordings with additional local noise at two different levels. Finally, the computational cost of all presented adaptation techniques is given and possible ways of complexity reduction are discussed.

6.1. Adaptation with polynomial nonlinearity

To overcome the slow convergence of the NLMS-adapted polynomial, two strategies have been developed in Section 3.2, which are now compared regarding ERLE and $D^{(\text{p})}[k]$. To focus on effects of preprocessor adaptation, white Gaussian noise is used as input signal $x[k]$, so that the FIR filter converges as fast as possible. Note that the correlation properties of $x[k]$ have no influence on the preprocessor convergence speed, as it is memoryless. As in these experiments only the fundamental mechanisms are of interest, the 'unknown' system and the adaptive system have the same structure: $N = 50$ FIR taps and $P = 3$,5, or 7 polynomial coefficients. The latter are chosen such that $10\log_{10}\text{NLR} = -8$ dB holds in each of the three cases. This is illustrated by the dashed curves in Fig. 4, which show the residual error power and the system error without nonlinear preprocessing, respectively.

The other three curves in Fig. 4 are obtained with NLMS-adaptation according to (25) and $\delta_u = \sigma_x^2$ labelled 'NLMS', the same with additional orthogonalization using $\boldsymbol{A}$ as given in (33) labelled 'ortho', and RLS-adaptation with re-initialization every 1000 steps using $\delta = 0.01$.

One can see, that the NLMS adaptation is fast enough for practical applications only for a polynomial of order 3. Although the orthogonalization was performed with the matrix $\boldsymbol{A}$ pre-calculated a measured value of σ_x^2, it is not as successful as the RLS adaptation. However, the $D^{(\text{p})}$ curves for $P = 7$ indicate, that the orthogonalization still leads to a significant gain of convergence speed and could be used in applications, where the RLS adaptation seems not to be robust enough.

6.2. Verification of nonlinear model

Now we examine whether the seventh-order polynomial and the hard-clipping curve discussed above are appropriate to model the nonlinear behaviour of real audio hardware. A speech signal is passed through a one-chip amplifier and a small

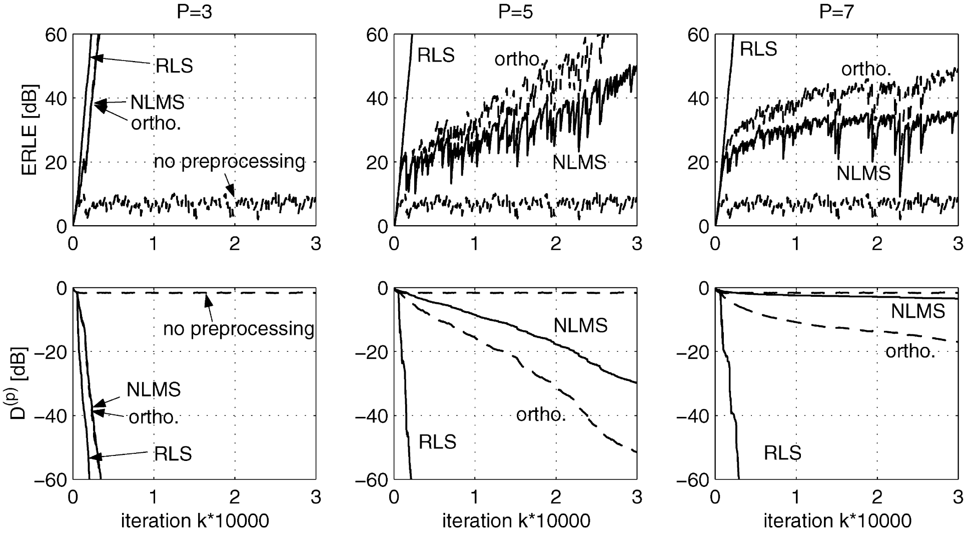

Fig. 4. Convergence with different polynomial adaptation schemes.

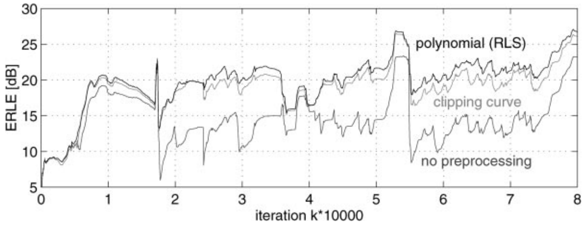

Fig. 5. Adaptation with real-world scenario.

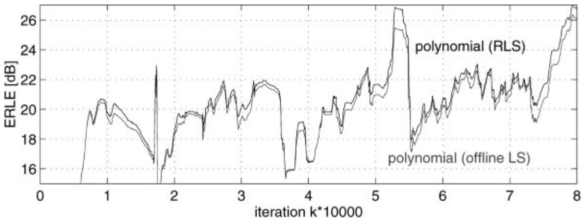

Fig. 6. RLS adaptation vs. off-line LS fitting of the polynomial.

loudspeaker, and recorded in a room with a short impulse response (reverberation time $T_{60} \approx 50$ ms) and no local noise. The signal power portion due to nonlinear distortion amounts to roughly 3% of the total power (NLR ≈ -15 dB), which explains why the ERLE curve obtained with a linear AEC (lower curve in Fig. 5) does not exceed -15 dB except at the speech pause at $k = 53\,000$. The adaptation parameters are $N = 250$, $P = 7$, $\alpha_h = 0.2$, $\alpha_a = 0.02$, and $\lambda = 0.995$. As no local noise is present, no stepsize control is applied. Fig. 5 shows that both preprocessors are able to greatly reduce the nonlinear echo power especially with loud speech portions. The adaptive polynomial can model the nonlinearity a little more accurately and reaches about 1 dB more ERLE. Fig. 6 shows a comparison of the adaptive polynomial with the non-adaptive one in [15], which is obtained off-line by LS fitting. Interestingly, the RLS adapted polynomial slightly outperforms the pre-calculated one, as it can adapt the (nonideal) polynomial model to short-time signal statistics. From these curves we can conclude that the preprocessor certainly adapts fast enough.

6.3. Experiments under adverse conditions

To test the adaptation algorithms in a more realistic scenario, the same hardware as in the last section is used, and as local signal $n[k]$ white Gaussian noise with average power σ_n^2 is added to the echo $y[k]$. A comparision with local speech signals indicates that this is a worst-case double-talk situation. The stepsize control as described in Section 5 is applied, where the expected values of $\varepsilon[k]$ and $n[k]$ in (51) are estimated through short-time averages over the given signals $n[k]$ and $e[k] - n[k]$.

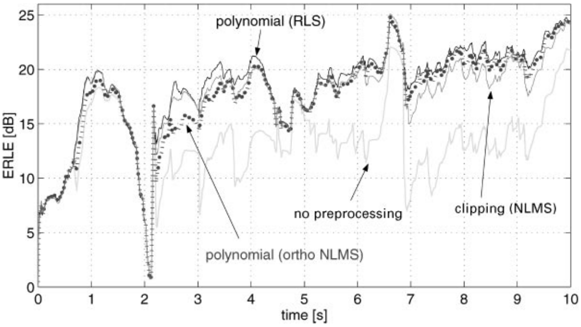

Fig. 7. Adaptation with $\sigma_n^2/\sigma_y^2 = -20$ dB.

This idealization makes the results independent of the quality of a stepsize control for α_{opt}. Note that only the idealized generation of α_{opt} has access to $n[k]$, but the adaptation algorithm has to rely on $d[k] = y[k] + n[k]$ like in real AEC situations.

The adaptive FIR filter length $N = 250$ sufficiently covers the impulse response of the measurement room, so that nonlinear echo components limit the adaptation error. As proposed from experiments in [15], the polynomial has $P = 7$ coefficients.

In the first experiment, shown in Fig. 7, a small amount of lower local noise power $\sigma_n^2 = 0.1\sigma_y^2$ is added, so that the FIR-filter reaches its steady-state after about one second. Comparison with Fig. 5, which is the same experiment without noise, shows that the stepsize control works well, as except for a somewhat slower convergence the same results are obtained. With orthogonalized NLMS adaptation, shown as dotted curve, the polynomial has converged after about 5 s and similar ERLE values as for the RLS-adapted polynomial result.

The same experiment has been conducted with 10 times higher local noise power, i.e., $\sigma_n^2 = \sigma_y^2$, so that the noise level exceeds the echo level at most times. This is illustrated in Fig. 8, where the local signal $n[k]$ and the echo signal $y[k]$ are plotted separately. In this extreme situation α_{opt} is steered near zero at almost all times, so that even after 10 s the steady state is not yet reached, and we can study the adaptation process from it. Fig. 9 shows robust behaviour of both the RLS-adapted polynomial and the NLMS-adapted hard-clipping curve. The hard-clipping curve always outperforms the linear AEC and converges faster than the polynomial. The ERLE with the RLS adapted polynomial drops below the linear AEC at some times, e.g. at 2 and 5.1 s. This problem does not occur, if the polynomial is adapted with NLMS and orthogonalization, which is plotted as dotted curve. Interestingly, the orthogonalized NLMS-adapted polynomial converges only about three times slower than the RLS-adapted one, so that after 10 s also 19 dB ERLE are reached.

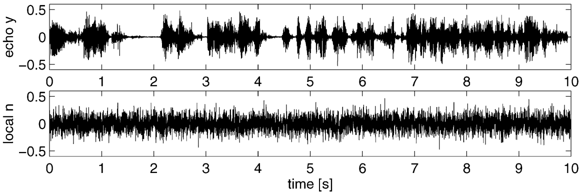

Fig. 8. Signals for simulation of severe local distortion.

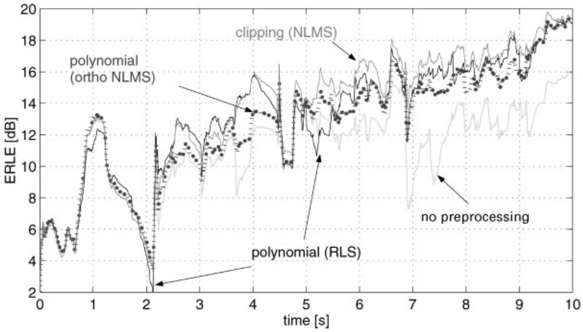

Fig. 9. Adaptation with $\sigma_n^2/\sigma_y^2 = 0$ dB.

6.4. Computational complexity

The linear AEC with NLMS adaptation requires about $2N$ multiplications per sample (MPS) and $2N$ memory words (MEM). We will not account for overhead for memory access and constant contributions to the number of MPS (i.e. computational load not depending on P or N). For the discussed algorithms, the number of additions is roughly equal to MPS.

The complexity for adaptive nonlinear preprocessing consists of two parts: the evaluation of the nonlinear function is required every sampling instant but is inexpensive due to the small number of coefficients. The main load is adaptation of the preprocessor coefficients, which generally is proportional to the product of N and the number of preprocessor coefficients P. For the polynomial, this is the evaluation of $\boldsymbol{X}_p[k]^{\mathrm{T}}\hat{\boldsymbol{h}}[k]$ in (22) which requires NP MPS. For the clipping curve, only N comparisons/sample are required instead, which we will count as N MPS.

The orthogonalization of the polynomial excitation $\boldsymbol{x}_p[k]$ requires the evaluation of $\boldsymbol{A}[k]\boldsymbol{x}_p[k]$ in (27), which costs $K = ((P-1)/2)^2$ MPS, if x has zero mean. Compared to the P MPS required for the polynomial evaluation, this is little overhead.

Adapting the polynomial with the standard RLS algorithm costs $2P^2 + 4P$ MPS instead of P MPS for the NLMS adaptation, which again is little overhead compared to NP. Fast versions of the RLS algorithm (see, e.g., [7]) explicitly exploit the transversal structure of the input vector. As this is not given here, these fast implementations cannot be used. Table 3 summarizes the MPS and memory requirements of the joint adaptation algorithms.

The high memory requirement of the polynomial adaptation can be circumvented, if instead of $\boldsymbol{X}_p$ (22) only $\boldsymbol{x}$ (7) is stored, and each row of $\boldsymbol{X}_p$ is computed from the according element of $\boldsymbol{x}$ every sampling instant. Without orthogonalization, this costs NP extra MPS, and with orthogonalization, extra NK MPS.

If the preprocessor adapts faster than practically necessary, the complexity can be reduced by sub-sampled adaptation. This is certainly the case for the RLS-adapted polynomial and the NLMS-adapted hard-clipping curve. The NLMS-adapted polynomial with orthogonalization has only limited possibilities of complexity reduction because of slow convergence. Another way of complexity reduction would be to neglect the even powers of the polynomial.

For the clipping curve, the normalization term $\|\hat{\boldsymbol{h}}\|_2^2$ contributes N MPS. As this needs not to be calculated at every sampling instant, approximately N MPS can be saved.

Table 3
Computation and memory requirement

	MPS	MEM
Polynomial (NLMS)	$2N + 3P + NP$	$2N + 2P + NP$
Polynomial (ortho + NLMS)	$2N + 2P + K + NP$	$2N + 2P + K + NP$
Polynomial (RLS)	$2N + 5P + 2P^2 + NP$	$2N + 3P + P^2 + NP$
Clipping curve	$4N$	$3N$

7. Conclusions

Nonlinearities in the acoustic echo path of low-cost hands-free communication equipment can be partially modelled by a memoryless nonlinear function preceding the typically used linear FIR filter. Special algorithms allow the joint adaption of both stages from a common error signal. For two nonlinear functions, a polynomial and a hard clipping curve, different fast adaptation algorithms and appropriate stepsize control mechanisms are proposed which provide up to 10 dB ERLE gain. They can be summarized into three working solutions with different computational cost. A clipping curve with only one NLMS-adapted parameter increases the complexity of a linear AEC by about 50%. This model is somewhat less accurate than the polynomial resulting in 1 dB less echo reduction. Robust adaptation is achieved even with severe local interference, and the linear AEC is outperformed at all times. To achieve fast adaptation of a seventh order polynomial, the coefficients can be adapted with the RLS algorithm. Using sub-sampled adaptation, similar computational cost results as with the first solution. However, at times of severe local interference, performance degradation compared to linear AEC was observed pointing to a lack of robustness. The NLMS adapted polynomial with orthogonalization is computationally expensive, as it converges too slow for sub-sampled adaptation, but it combines the accurate modelling of the polynomial with the robustness of NLMS adaptation.

References

[1] A.N. Birkett, R.A. Goubran, Acoustic echo cancellation using NLMS-neural network structures, Proceedings of the ICASSP'95, Detroit, MI, USA, pp. 2035–3038.

[2] C. Breining, Control of a hands-free telephone set, Signal Processing 61 (1997) 131–143.

[3] C. Breining, P. Dreiseitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, J. Tilp, Acoustic echo control, Signal Process. Mag. 16 (4) (July 1999) 42–69.

[4] J.L. Brewer, Kronecker products and matrix calculus in systems theory, IEEE Trans. Circuits Systems 25 (9) (1978) 772–781.

[5] F.X.Y. Gao, W.M. Snelgrove, Adaptive linearization of a loudspeaker, Proceedings of the ICASSP'91, pp. 3589–3592.

[6] G.H. Golub, C.F. van Loan, Matrix Computations, 2nd Edition, Johns Hopkins University Press, Baltimore, MD, 1989.

[7] S. Haykin, Adaptive Filter Theory, 3rd Editon, Prentice-Hall, Englewood Cliffs, NJ, 1996.

[8] F. Heinle, R. Rabenstein, A. Stenger, A measurement method for the linear and nonlinear properties of electro-acoustic transmission systems, Signal Processing 64 (1) (1998) 49–60.

[9] C.L. Lawson, R.J. Hanson, Solving Least Squares Problems, Prentice-Hall, Englewood Cliffs, NJ, 1974.

[10] B.S. Nollet, D.L. Jones, Nonlinear echo cancellation for hands-free speakerphones, Proceedings of the NSIP'97, September 8–10, 1997, Michigan, USA.

[11] A. Papoulis, Probabiliy, Random Variables, and Stochastic Processes, 2nd Edition, McGraw-Hill, Singapore, 1984.

[12] L.R. Rabiner, R.W. Schafer, Digital Processing of Speech Signals, Prentice-Hall, Englewood Cliffs, NJ, 1978.

[13] H. Schurer, Linearization of Electroacoustic Transducers Print Partners Ipskamp, Enschede, Netherlands, 1997.

[14] A. Stenger, W. Kellermann, R. Rabenstein, Adaptation of acoustic echo cancellers incorporating a memoryless nonlinearity, Proceedings of the IEEE Workshop on Acoustic Echo and Noise Control (IWAENC'99), Pocono Manor, PA, USA, September, 1999, pp. 168–171.

[15] A. Stenger, R. Rabenstein, An acoustic echo canceller with compensation of nonlinearities, Proceedings of the EUSIPCO'98, Isle of Rhodes, Greece, September 1998, pp. 969–972.

[16] A. Stenger, L. Trautmann, R. Rabenstein, Nonlinear acoustic echo cancellation with 2nd order adaptive volterra filters, Proceedings of the ICASSP'99, IEEE, Phoenix, USA, 1999 pp. 877–880.

[17] S.D. Widrow, B. Stearns, Adaptive Signal Processing, Prentice-Hall, Englewood Cliffs, NJ, 1985.

[18] M. Zollner, E. Zwicker, Elektroakustik, 3rd Edition, Springer, Berlin, 1993 (in German).