ELSEVIER

Signal Processing 81 (2001) 2131–2153

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

A contribution to (neuromorphic) blind deconvolution by flexible approximated Bayesian estimation

Simone Fiori *

Neural Networks and Adaptive Systems Research Group, DIE—University of Perugia, Perugia, Italy

Received 24 August 2000; received in revised form 19 May 2001

Abstract

'Bussgang' deconvolution techniques for blind digital channels equalization rely on a Bayesian estimator of the source sequence defined on the basis of channel/equalizer cascade model which involves the definition of deconvolution noise. In this paper we consider four 'Bussgang' blind deconvolution algorithms for uniformly distributed source signals and investigate their numerical performances as well as some of their analytical features. Particularly, we show that the algorithm, introduced by the present author, provided by a flexible (neuromorphic) estimator is effective as it does not require to make any hypothesis about convolutional noise level and exhibits satisfactory numerical performances.

Keywords: 'Bussgang' blind deconvolution/equalization; Bayesian estimation; Non-linear adaptive (neuromorphic) filtering; Adaptive activation function neuron; Pseudo-LMS filtering

1. Introduction

Digital blind deconvolution of a one-dimensional channel consists in retrieving the input or source data, and sometimes the relevant features of the channel, given only the channel output and some information on the statistics of the source. The transmission system as well as the statistics of the source can be time-varying, but usually it is assumed that these variations are slow enough to be treated as time-invariant. Also, a common assumption is that the transmission system is linear, so that deconvolution may be performed with a linear inverse filter.

Conventional deconvolution in communications, referred to as channel equalization, relies on the transmission of a training signal which is known to the receiver, in order to identify the parameters that describe the communication channel. This transmission must be repeated at regular intervals according to the expected time-variability of the channel, making the efficiency of bandwidth utilization be constrained by the proportion of transmission time devoted to equalization. With the growth in the telecommunications industry, bandwidth appears a fundamental limitation for the expansion of new systems and services [53]. For this reason a number of researchers have focused their interests upon adaptive channel equalization algorithms which do not need a training

* Fax: +39-0744-492-925.
E-mail address: sfr@unipg.it (S. Fiori).

0165-1684/01/$ - see front matter
PII: S0165-1684(01)00108-6

sequence (a partial list of references is given by [1,5,6,10,12,19,26,30,32,37,39,54,56,57,59] and citations therein).

In the present paper, we employ the vector notation to represent discrete-time series, so that digital convolution may be computed as a scalar product and the superposition of two time-series is represented by vector addition; also, it is easy to define a zero time-series $0(t)$ and a one time-series $1(t)$ (which coincide to the unitary impulse function). Then, it is readily recognized that a set of time series in a Hilbert space endowed with zero element, one element, additive operation and multiplicative operation is a commutative ring. A division operation (corresponding to deconvolution or convolution with an inverse filter) may be defined too [52], and this would make the space be a field.

In the non-causal linear model, the input–output transmission system description writes

$$y(t) = \mathbf{h}^{\mathrm{T}}\mathbf{s}(t) + \mathcal{N}(t), \tag{1}$$

where $\mathbf{s}(t)$ is a vector containing zero-mean time-shifted input samples and $\mathbf{h}$ is the system's impulse response vector, respectively defined by

$$\mathbf{s}(t) \stackrel{\text{def}}{=} \begin{bmatrix} \cdots \\ s(t+2) \\ s(t+1) \\ s(t) \\ s(t-1) \\ s(t-2) \\ \cdots \end{bmatrix}, \qquad \mathbf{h} \stackrel{\text{def}}{=} \begin{bmatrix} \cdots \\ h_{-2} \\ h_{-1} \\ h_0 \\ h_{+1} \\ h_{+2} \\ \cdots \end{bmatrix}. \tag{2}$$

Also, t indicates discrete time, and $\mathcal{N}(t)$ denotes zero-mean additive channel noise that may originate from many simultaneous effects [50], as crosstalk and sampling errors. Since both source signal and channel noise are unobservable, they cannot be distinguished, thus usually the latter is ignored in the theoretical developments [30]; also, usually the need for blind deconvolution arises from severe inter-symbol interference (ISI) and not from additive noise, thus the effect of noise on the algorithms and their performances is usually small [31].

A linear equalizer described by its impulse response $\mathbf{w}$ deconvolves $s(t)$ if it cancels the effects produced by $\mathbf{h}$ on the source signal. Denoting by $\mathbf{y}(t)$ the vector containing time-shifted observed samples $y(t)$, the equation describing the inverse discrete-time filter output $x(t)$ reads

$$x(t) = \mathbf{w}^{\mathrm{T}}(t)\mathbf{y}(t). \tag{3}$$

Vectors $\mathbf{y}(t)$ and $\mathbf{w}(t)$ are thus defined as

$$\mathbf{y}(t) \stackrel{\text{def}}{=} \begin{bmatrix} \cdots \\ y(t+2) \\ y(t+1) \\ y(t) \\ y(t-1) \\ y(t-2) \\ \cdots \end{bmatrix}, \qquad \mathbf{w}(t) \stackrel{\text{def}}{=} \begin{bmatrix} \cdots \\ w_{-2}(t) \\ w_{-1}(t) \\ w_0(t) \\ w_{+1}(t) \\ w_{+2}(t) \\ \cdots \end{bmatrix}. \tag{4}$$

In general, $\mathbf{h}$ represents a non-minimum phase system. Denoting by $H_{\mathrm{c}}(z)$ the channel transfer function, from a known theorem of digital filter theory it is possible to write:

$$H_{\mathrm{c}}(z) = H_{\mathrm{mp}}(z)H_{\mathrm{ap}}(z),$$

where $H_{\mathrm{mp}}(z)$ denotes the minimum phase part of $H_{\mathrm{c}}(z)$, and $H_{\mathrm{ap}}(z)$ represents the remaining all-pass part. By using classical *second-order* methods involving $|H_{\mathrm{c}}(z)|^2$ only, it is impossible to invert the all-pass part of the system, thus its deconvolution may not be attained, hence higher-order methods are needed [10]. In addition, when $\mathbf{h}$ is non-minimum phase the inverse $H_{\mathrm{c}}^{-1}(z)$ is unstable or non-causal, thus real-time deconvolution is not permitted [30], and the inverse has to be approximated by an all-zeros function, that is an FIR inverse filter allowing for on-line deconvolution with a usually negligible delay. On the other hand, every time an FIR deconvolving filter is used, an approximation error occurs [30]. Moreover, let us denote as $\mathbf{w}_\star$ the true equalizing filter, for which $cs(t-\delta) = \mathbf{w}_\star^{\mathrm{T}}\mathbf{y}(t)$, and as $\mathbf{H}$ the matrix such that $\mathbf{y}(t) = \mathbf{H}\mathbf{s}(t)$, where c is a possible amplitude factor that takes into account the indeterminacy about the true source power and δ is the total inversion delay.[1] At each adaptation step, the difference between the filter output and the scaled-delayed

[1] As long as $\mathbf{h}$ is IIR, the matrix $\mathbf{H}$ has infinitely many entries; when the channel impulse response is FIR, $\mathbf{H}$ is a matrix that can be easily computed on the basis of the channel's impulse response. In any case, the global impulse response $\mathbf{T} \stackrel{\text{def}}{=} \mathbf{H}^{\mathrm{T}}\mathbf{w}_\star$ would result in an (approximately) all-zero vector except for the δth entry equal to c.

source signal can be written as

$$x(t) - cs(t-\delta) = [\mathbf{w}(t) - \mathbf{w}_\star]^{\mathrm{T}}\mathbf{y}(t) = [\mathbf{w}(t) - \mathbf{w}_\star]^{\mathrm{T}}\mathbf{H}\mathbf{s}(t). \quad (5)$$

Thus $x(t) - cs(t-\delta)$ may be thought of as an infinitely long linear combination of independent source random variables and, in virtue of central limit theorem of statistics, it may be well-represented as a Gaussian random process $n(t)$ termed *deconvolution noise* [6,30]. Formally

$$x(t) = cs(t-\delta) + n(t). \quad (6)$$

The noise $n(t)$ is zero-mean and completely characterized by its variance denoted here with σ^2 (termed 'deconvolution noise power').

When $\mathbf{h}$ and $s(t)$ are unknown and unobservable, the response $\mathbf{w}_*$ such that $y(t)$ coincides to $s(t)$, except for a finite delay and a scale factor, has to be blindly identified [6,10,30,38,57]. It is only required that $\mathbf{h}$ has *finite energy* (as well as its inverse) and supposed that $s(t)$ is an *independent identically distributed* (IID) sequence with zero-mean and unitary variance, namely

$$0 < \mathbf{h}^{\mathrm{T}}\mathbf{h} = \sum_{i=-\infty}^{+\infty} h_{-i}^2 < +\infty,$$

$$E_s[s] = 0, \quad E_s[s^2] = 1. \quad (7)$$

Since the pioneering work of Sato (see for instance [10,48,61] and references therein), several blind deconvolution methods have been developed, requiring different levels of knowledge of source statistics; they have been applied to e.g. the equalization of communication channels [6,9,46], seismic traces deconvolution for geophysical measurements [26,49,62], vocal tract identification [30], blind image restoration [29,33,41], optical memory-support storage and retrieval performance enhancement [16,43], and echographic data focusing [13,36]. Much research has been carried out in order to develop both suitable filtering structures [1,9,15,30,34,44–46] and cost functions and statistical estimation methods [6,7,12,20,23,25,27,28,40,42,49–51,59]. In this paper a transversal filter as depicted in Fig. 1 is used, where the $w_i(t)$'s denote the adjustable tap-weights.

One of the most known and studied algorithms is the 'Bussgang' one developed by Bellini [6,8,23,26,30], based on a memoryless Bayesian estimator and a pseudo-LMS adjustment of inverse filter response. Since the estimator depends on σ^2, a problem related to that algorithm is the updated estimation of the deconvolution noise power.

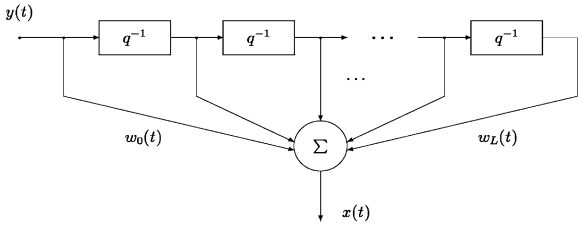

Fig. 1. An FIR adaptive filter with $L+1$ tap-weights. (Blocks marked with q^{-1} represent unitary delays.)

In this paper we revise four contributions about 'Bussgang' inverse filtering with the aim to gain a deep insight into original algorithm and recent theoretical developments. Namely, we discuss on

- The 'Bussgang' algorithm, devised on the basis of Bayesian estimation technique which requires the knowledge of the input data distribution;
- The modification proposed in [30], which consists in approximating the exact Bayesian estimator with a (neuromorphic) non-adaptive function, in case of uniformly distributed sources threads;
- The modification proposed by the present author in [21], consisting in a self-tuning procedure that allows to automatically determine optimal parameters of an adaptive activation function neuron as flexible approximated estimator;
- The improved learning procedure devised by Amari [2] relying on *natural gradient* optimization, based on the concept of steepest descent search on a curved manifold which exploits the local properties of parameters space.

The self-tuning behavior proposed in [21] allows to overcome the problem of finding a suitable value of σ^2. Moreover, since the parameters are continuously refined through time, suitable values are used during any learning phase: this alleviates also the common problem of finding appropriate learning stepsizes. The adapting theory early proposed in [21] has its basis in a learning theory for neural networks developed for probability density estimation by adaptive activation function neurons [24], which

has recently been successfully applied to eterokurtic blind source separation [22].

Also, the present paper is devoted to the theoretical analysis of asymptotic behavior of the adapting algorithms, which allows us to discuss the relationships with other deconvolution theories known from the scientific literature.

The paper is organized as follows. In Section 2 a gradient-based version of the 'Bussgang' algorithm is shown and a particularization of it with a flexible approximated Bayesian estimator, suited for uniformly distributed source sequences, is presented. The case of uniformly distributed random signal is a very studied and useful one [6,8,30], also because it is almost always possible to make uniform an arbitrarily distributed IID random process, by using several methods (see, for instance, [24]). Section 3 is devoted to analytical studies on the asymptotic properties of considered algorithms through the mean-field steady-state analysis. In Section 4 we report the results of some exploratory experiments aiming at giving some examples to numerically asses the theoretical developments, while in Section 5 we show through simulations that the gradient-based self-tuning algorithm is effective, fast and accurate. Section 6 concludes the paper.

2. 'Bussgang' blind deconvolution and its neuromorphic extensions

2.1. Bayesian estimation for blind deconvolution

From filter output signal model (6) we can imagine a way to get an estimation of the source sequence $s(t)$ from $x(t)$ by means of a rough estimator. In fact, (6) is a deterministic model but for the deconvolution noise, thus the above question gives rise to a classical estimation problem. Motivated from simplicity and robustness, a memoryless estimator was suggested in [6,8]. Likely, the estimator will depend upon the inverse filter response through the probability density function (pdf) of the source and on the level of convolutional noise. In symbols we write

$$\hat{s}(t-\delta) = b(x(t)), \tag{8}$$

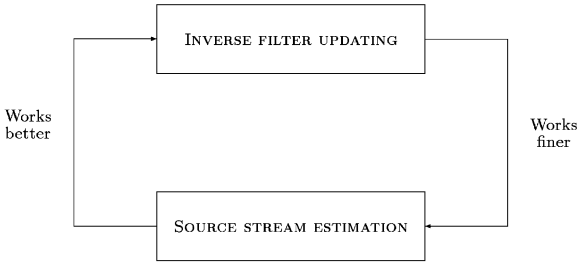

Fig. 2. Inverse filter updating and source signal estimation cycle.

where any dependence is understood. At first the estimation will not be reliable, but we have no better guess of the source thread, thus we may use $\hat{s}(t)$ to get a better estimation of the inverse filter; a better deconvolving filter results in lesser deconvolution noise, which, in turn, allows for a finer estimation of the source stream, in a cycle that ultimately converges to optimal deconvolving filter and good estimation of source data, as shown in Fig. 2. In this cycle, the inverse filter may be updated by trying to iteratively minimize the estimated error $x(t) - c\hat{s}(t-\delta)$; this aspect will be detailedly discussed later.

The main question is now how to select an appropriate estimator. The answer comes from Bayesian estimation theory.

Let r_x be a random variable and r_y be another random variable having a functional/stochastic dependence on r_x. If r_y is observable while r_x is unobservable, we wonder if is it possible to get an estimate of the values of r_x given samples of the variable r_y, that is an inverse problem. Let us denote with $\hat{r}_x(r_y)$ such an estimate. Provided that the joint probability density function $p_{r_x,r_y}(r_x, r_y)$ of the two variables be known, a method to get the required estimation is the minimization of the mean squared error (MSE), defined as the functional:

$$\mathrm{MSE}[\hat{r}_x] \stackrel{\mathrm{def}}{=} \int\int_{\mathscr{R}^2} [r_x - \hat{r}_x(r_y)]^2 \, p_{r_x,r_y}(r_x, r_y)\, \mathrm{d}r_x\, \mathrm{d}r_y.$$

It is important to note that $\hat{r}_x(\cdot)$ is not a single value, but a *function*. Let us introduce the conditional pdf:

$$p_{r_x|r_y}(r_x|r_y) \stackrel{\mathrm{def}}{=} \frac{p_{r_x,r_y}(r_x, r_y)}{p_{r_y}(r_y)}$$

with $p_{r_y}(r_y)$ being the marginal pdf of the random variable r_y; it describes the inverse functional/stochastic relationship between r_x and r_y. By solving a variational minimization problem, it is straightforward to find that the only minimum of the MSE functional is

$$\hat{r}_x(r_y) = \int_{\mathscr{R}} r_x\, p_{r_x|r_y}(r_x|r_y)\,\mathrm{d}r_x.$$

In our notation, the non-linear Bayesian estimator of the source signal writes:

$$\hat{s} = b(x) \stackrel{\text{def}}{=} E_s[s|x] = \int_{\mathscr{R}} s p_{s|x}(s|x)\,\mathrm{d}s, \tag{9}$$

where $p_{s|x}(s|x)$ is the pdf of s conditioned to the knowledge of x. From Bayes theorem we have

$$p_{s|x}(s|x) = \frac{p_{x|s}(x|s)\,p_s(s)}{p_x(x)} = \frac{p_{x|s}(x|s)\,p_s(s)}{\int_{\mathscr{R}} p_{x|s}(x|s)\,p_s(s)\,\mathrm{d}s},$$

where $p_s(s)$ denotes the statistical distribution of source data and $p_{x|s}(x|s)$ statistically describes the formation of channel output values from the knowledge of input ones. As we already observed, the channel is deterministic but for the convolutional noise, thus the latter probability density function writes $p_{x|s}(x|s) = p_n(n) = p_n(x - cs)$, with $p_n(n)$ being the Gaussian distribution of deconvolution noise; then, the estimator is given by

$$b(x) = \frac{\int_{\mathscr{R}} s\, p_n(x - cs)\, p_s(s)\,\mathrm{d}s}{\int_{\mathscr{R}} p_n(x - cs)\, p_s(s)\,\mathrm{d}s},$$

$$p_n(n) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{n^2}{2\sigma^2}\right].$$

Because of the form of $p_n(n)$, the integrals above might be tractable and, as a meaningful result, it is possible to write $b(x)$ as a function of x and $p_x(x)$ only. In fact, by observing that the following identity holds:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}x}\int_{\mathscr{R}} p_n(x - cs)\, p_s(s)\,\mathrm{d}s &= -\frac{x}{\sigma^2}\int_{\mathscr{R}} p_n(x - cs)\, p_s(s)\,\mathrm{d}s \\ &\quad + \frac{c}{\sigma^2}\int_{\mathscr{R}} s\, p_n(x - cs)\, p_s(s)\,\mathrm{d}s,\end{aligned}$$

we get

$$b(x) = \frac{\sigma^2}{c}\left[\frac{p_x'(x)}{p_x(x)} + \frac{x}{\sigma^2}\right]. \tag{10}$$

Provided that the source statistics be known, channel output statistics might be evaluated and the required estimator be computed, as well. As expected, the chosen estimator depends on deconvolution noise power and on the inverse filter response through $p_x(x)$; because of its origin, the above function may be thought of as the maximum a-posteriori (MAP) estimator of the data plus convolutional noise [35].

2.2. 'Bussgang' adaptive blind deconvolution

On the basis of the available estimator, in [6,8] an error criterion like

$$\tilde{U}(\mathbf{w}) = \frac{1}{2}E_x[(cb(x) - x)^2] \tag{11}$$

was proposed. The function $b(x)$ provides an estimation of the source signal, so the optimal deconvolving filter $\mathbf{w}_\star$ minimizes $\tilde{U}$ because it assumes its lowest values when $x \approx cs \approx cb$ or, equivalently, when the deconvolution noise $\sigma^2 \approx 2\tilde{U}$ minimizes.

As a method to find iteratively the optimal filter given observations of the channel output $x(t)$, a pseudo-LMS approach was used [6,30]. By interpreting the difference $cb(x) - x$ in (11) as an 'error', the structure of the algorithm proposed by Bellini is

$$\Delta\mathbf{w} = \rho[cb(x) - x]\mathbf{y} \quad \text{with } x = \mathbf{w}^{\mathrm{T}}\mathbf{y}, \tag{12}$$

where ρ is a positive learning stepsize. It belongs to the class of Robbins–Monro stochastic algorithms for finding the zeros of non-linear functions [11,47] that is, in this case, of the deconvolution error.

As pointed out in [17,21], there are no theoretical reasons to use the LMS-type procedure rather than others commonly available in the literature on iterative optimization. So, the minimization of the same cost function $\tilde{U}$ may be achieved also by means of a stochastic gradient steepest descent (SGSD) algorithm described by

$$\Delta\mathbf{w} \propto -\frac{\partial U}{\partial \mathbf{w}},$$

where $U(x)$ is the *instantaneous stochastic approximation* of $\tilde{U}$, i.e. $U = \frac{1}{2}(cb - x)^2$. In the present

context this rule assumes the following expression:

$$\Delta\mathbf{w} = -\eta[cb'(x) - 1][cb(x) - x]\mathbf{y}, \tag{13}$$

where η is a positive learning rate and $b'(x)$ denotes the derivative of the function $b(x)$ with respect to x.

By comparing Eqs. (12) and (13), one gathers that they coincide if in (12) the variable stepsize

$$\rho(x) = -\eta[cb'(x) - 1] = -\eta\frac{U'(x)}{\sqrt{2U(x)}} \tag{14}$$

is used. Usually, the meaning of a variable stepsize is that of a *self-controlled adaptation rate* which assumes large values at the beginning of learning, and takes more and more small values with learning progress; this is evidenced by the proportionality of ρ to U'. Nevertheless, it can take again large values when contour conditions change, e.g. when a channel commutation happens [27,60]. The usefulness of the self-controlled stepsize will be decided on the basis of its predicted asymptotic behavior and of numerical experimental results.

Whatever adapting algorithm is utilized, the overall channel-filter-estimator system may be represented as in Fig. 3.

The theory was initially developed by assuming $s(t)$ as a random process uniformly distributed within $[-\sqrt{3}, +\sqrt{3}]$ [6,8,30]. Straightforward calculations show that, in this case

$$b(x) = \frac{x}{c} + \frac{\sigma}{c}\frac{\Phi((x + c\sqrt{3})/\sigma) - \Phi((x - c\sqrt{3})/\sigma)}{\Psi((x - c\sqrt{3})/\sigma) - \Psi((x + c\sqrt{3})/\sigma)}, \tag{15}$$

where, by definition

$$\Phi(\xi) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\xi^2}{2}\right),$$

$$\Psi(\xi) = \int_{\xi}^{+\infty}\Phi(\zeta)\,\mathrm{d}\zeta = \frac{1}{2}\operatorname{erfc}\left(\frac{\xi}{\sqrt{2}}\right).$$

In absence of any other constraint, the quantities c and σ^2 are independent. If the power of filter output is kept constant to some value by an automatic gain control (AGC), e.g. $E_s[x^2] = 1$, from model (6) we instead have $c^2 + \sigma^2 = 1$. In this case, the distortion power, where the convolutional distortion [31] is defined as $\varepsilon(t) \stackrel{\text{def}}{=} s(t) - cs(t - \delta) - n(t)$, is $E_s[\varepsilon^2] = (1 - c)^2 + \sigma^2 = 2(1 - c)$.

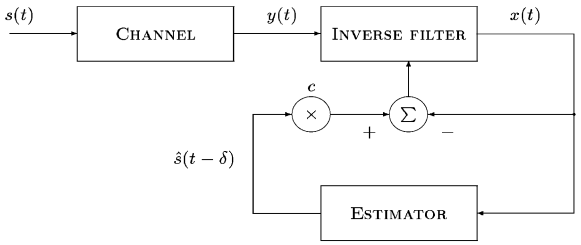

Fig. 3. Schematic of the overall channel-filter-estimator system.

2.3. Neuromorphic deconvolution with flexible estimator

As expected, the function $b(x)$ is dependent upon the deconvolution noise power σ^2 [6,8,30]. A suitable estimation for this parameter is quite difficult; moreover, an optimal value for σ^2 likely does not exist since it should be changed through time according to the adaptation progress. It is in fact easy to show, from Eq. (5) and from the statistical properties of source data, that:

$$\sigma^2(t) = E_s[n^2(t)] = [\mathbf{w}(t) - \mathbf{w}_\star]^{\mathrm{T}}\mathbf{H}\mathbf{H}^{\mathrm{T}}[\mathbf{w}(t) - \mathbf{w}_\star],$$

whereby the dependence of the deconvolution noise power from inverse filter misadjustment and, hence, from the time.

Despite this, for a wide noise power range a suitable approximation of the actual Bayesian estimator $b(x)$ for uniformly distributed source sequences (15) is [30] the bilateral 'sigmoidal' function

$$\hat{b}(x) = \frac{a}{c}\tanh(\lambda x) \tag{16}$$

with a and λ being properly chosen parameters. About the suitability of the above approximation, it is also worth noting, for example, that when the source signal is a 2-PAM with symbols in $\{-1, +1\}$, the *exact* Bayesian estimator, computed in [17,31], easily recasts into

$$b_{2-\mathrm{PAM}}(x) = \frac{1}{c}\tanh\left(\frac{cx}{\sigma^2}\right).$$

The linear filter of Fig. 1 may be now equipped with a simple non-linear function that represents the Bayesian estimator and looks as in Fig. 4. Generally speaking, this is a simplified model of artificial neurons endowed with filtering synapses, which give rise to IIR-MLP neural networks (see e.g. [14] and references therein).

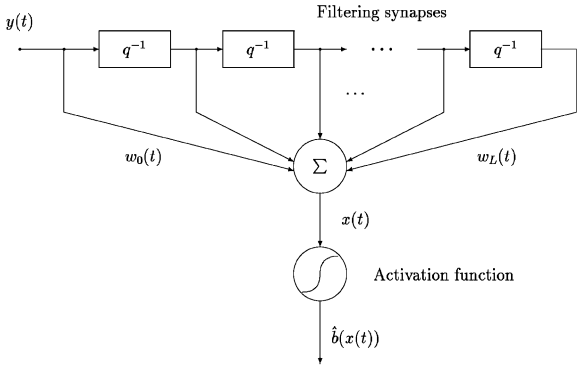

Fig. 4. Linear FIR filter equipped with the approximated Bayesian estimator as a neuron with filtering synapses.

If the above approximating expression is used, the gradient of U becomes

$$\frac{\partial U}{\partial \mathbf{w}} = \left[a\lambda - \frac{\lambda}{a}c^2\hat{b}^2(x) - 1\right][c\hat{b}(x) - x]\mathbf{y}. \tag{17}$$

In [30] a pair of values for a and λ is obtained by fitting the expression (16) with the actual estimator for a given convolutional distortion level. Anyway, this approximation suffers of two problems:

1. it is clear that as an optimal constant value for σ^2 cannot be found, a suitable pair of *constant* parameters a and λ cannot be determined, too;
2. in practice, the discussed approximation holds true only for uniformly distributed source data.

In the present paper we deal with uniformly distributed real-valued source signals, and focus on the best choice of parameters. However, as we shall briefly illustrate later, the two questions are actually closely related, and a more general neural structure can be envisaged to answer both.

To overcome this problem we propose to adapt a and λ through time by means of a SGSD algorithm applied to U (thought of as a function of a, λ and x). In formulas we get

$$\Delta a = -\eta_a \frac{\partial U}{\partial a} = -\eta_a[c\hat{b}(x) - x]\frac{c\hat{b}(x)}{a}, \tag{18}$$

$$\begin{aligned}\Delta\lambda &= -\eta_\lambda \frac{\partial U}{\partial \lambda} \\ &= -\eta_\lambda[c\hat{b}(x) - x][a^2 - c^2\hat{b}^2(x)]\frac{x}{a},\end{aligned} \tag{19}$$

where η_a and η_λ are constant positive learning step-sizes.

In our algorithm all parameters a, λ and $\mathbf{w}$ adapt at the same time by means of Eqs. (13), (18) and (19). Because of the structure of U (16), now the problem of minimizing the cost function is ill-posed, because a simple way to minimize U is to vanish $||\mathbf{w}||$ to zero.

To prevent such behavior, it is possible to add a simple constraint on the norm of $\mathbf{w}$, that is $\mathbf{w}^{\mathrm{T}}\mathbf{w} - \kappa^2 = 0$, where κ^2 is an arbitrarily chosen non-null constant that provides an amplification of the filter output signal with a factor $|\kappa|$. This condition, corresponding to automatic gain control, can be taken into account by defining a new criterion $\tilde{J}$ as

$$\tilde{J}(\mathbf{w}) = \tilde{U}(\mathbf{w}) + \ell(\mathbf{w}^{\mathrm{T}}\mathbf{w} - \kappa^2), \tag{20}$$

where ℓ is a *Lagrange multiplier*. The optimum $\mathbf{w}$ satisfies now

$$\frac{\partial \tilde{J}}{\partial \mathbf{w}} = \frac{\partial \tilde{U}}{\partial \mathbf{w}} + 2\ell\mathbf{w} = \mathbf{0},$$

with $\mathbf{w}^{\mathrm{T}}\mathbf{w} = \kappa^2$, therefore

$$\mathbf{w}^{\mathrm{T}}\left(\frac{\partial \tilde{U}}{\partial \mathbf{w}}\right) + 2\ell\mathbf{w}^{\mathrm{T}}\mathbf{w} = \mathbf{w}^{\mathrm{T}}\left(\frac{\partial \tilde{U}}{\partial \mathbf{w}}\right) + 2\ell\kappa^2 = 0.$$

Hence the corresponding stochastic optimum ℓ is

$$\begin{aligned}\ell_\star &= -\frac{1}{2\kappa^2}\mathbf{w}^{\mathrm{T}}\left(\frac{\partial U}{\partial \mathbf{w}}\right) \\ &= -\left[a\lambda - \frac{\lambda}{a}c^2\hat{g}^2(x) - 1\right][c\hat{b}(x) - x]\frac{x}{2\kappa^2},\end{aligned} \tag{21}$$

where Eq. (17) has been used.

If the SGSD algorithm

$$\Delta\mathbf{w} = -\eta\frac{\partial J}{\partial \mathbf{w}}$$

is still used to search for the minimum of J, the unconstrained rule (13) may be replaced by

$$\begin{aligned}\Delta\mathbf{w} = -\eta &\left[a\lambda - \frac{\lambda}{a}c^2\hat{b}^2(x) - 1\right] \\ &\times[c\hat{b}(x) - x]\left(\mathbf{y} - \frac{x}{\kappa^2}\mathbf{w}\right).\end{aligned} \tag{22}$$

Eqs. (22), (18) and (19) give a new gradient-based blind equalization method with the flexible estimator (16), as depicted in Fig. 5. It relies on the self-tuning capability of neuromorphic filter of

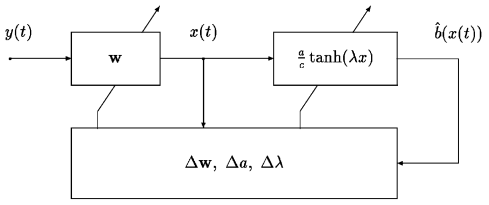

Fig. 5. Block diagram of gradient-based blind equalizer with flexible estimator.

Fig. 4; it is worth noting that the dependence of the above learning quantities is on the product $c\hat{b}$, thus from the definition of flexible approximated estimator, *the equations actually do not depend on the scale factor c*.

Another way to prevent $||\mathbf{w}||$ from vanishing is to constrain the power of filter output to be unitary, which corresponds to AGC. In this case, the new criterion writes

$$\tilde{J}_{\mathrm{AGC}}(\mathbf{w}) \stackrel{\mathrm{def}}{=} \tilde{U}(\mathbf{w}) + \ell_{\mathrm{AGC}}(E_s[x^2] - 1).$$

The optimal multiplier would find by

$$\mathbf{w}^{\mathrm{T}}\left(\frac{\partial \tilde{U}}{\partial \mathbf{w}}\right) + 2\ell_{\mathrm{AGC}}\mathbf{w}^{\mathrm{T}}E_s[x\mathbf{y}] = 0,$$

where $c^2 = \mathbf{w}^{\mathrm{T}}E_s[x\mathbf{y}] = 1$, thus $\mathbf{w}^{\mathrm{T}}\mathbf{w} = (\mathbf{h}^2\mathbf{h})^{-1}$.

A very simple and useful way to constrain the norm of $\mathbf{w}$ to be constant at $|\kappa|$ is to re-normalize the weight-vector at each iteration (or less frequently) by

$$\mathbf{w} \leftarrow \frac{|\kappa|\mathbf{w}}{||\mathbf{w}||}. \tag{23}$$

This provides exact meeting of the AGC constraint at any time.

It is worth mentioning that the normalization rule has important implications on the local minima avoidance problem: in [54] it has been shown that including normalization (or modifying the cost function, in fact) may help escaping the local solutions.

2.4. Natural-gradient based stochastic optimization

The adapting theory developed in the previous sections rely on the basic stochastic gradient-based optimization of the criterion $\tilde{U}(\mathbf{w})$; actually, the used gradient $\mathrm{d}\tilde{U}(\mathbf{w})/\mathrm{d}\mathbf{w}$ is the Jacobian or ordinary gradient of the function which, in general, *is not ensured to describe the steepest search direction* when the manifold of the parameters $\mathbf{w}$ is curved. This problem has been pointed out by Amari in the context of neural networks learning, in a series of works (see [2] and references therein), as an important cause of slow convergence in ordinary-gradient based learning algorithms. A possible solution is given by the theory of *natural gradient*, whereby the standard gradient search direction is altered according to the local Riemannian structure of the parameter space [2]. In the main paper [2], Amari devised the natural gradient optimization technique for vector spaces, matrix spaces in relation with information geometry, and systems spaces (i.e. for linear filters), which will be recalled in the following.

By denoting as $W(q^{-1},t)$ the quantity

$$W(q^{-1},t) \stackrel{\mathrm{def}}{=} \sum_{k=-\infty}^{k=+\infty} w_k(t)q^{-k},$$

where q^{-1} denotes the unit-delay operator, i.e. $q^{-1}x(t) \stackrel{\mathrm{def}}{=} x(t-1)$, the natural gradient of function $\tilde{U}(\mathbf{w})$ writes

$$\nabla_{\mathrm{NAT}}\tilde{U}(\mathbf{w}) \stackrel{\mathrm{def}}{=} \frac{\mathrm{d}\tilde{U}(\mathbf{w})}{\mathrm{d}\mathbf{w}}W(q^{-1},t)W(q,t), \tag{24}$$

where the filter $W(q^{-1},t)W(q,t)$ accounts for the local geometrical properties of Riemannian manifold of parameters at $\mathbf{w}$.

Let us examine now how learning algorithms (12) and (13) modify on the basis of new gradient definition (24). A unified expression for them is

$$\Delta\mathbf{w}(t) = \eta\gamma(x(t))\mathbf{y}(t) \tag{25}$$

and operator $W(q^{-1},t)W(q,t)$ applies to the regression vector $\mathbf{y}$ only [2,3]. To employ our short notation as before, we need to define the following two vector-streams:

$$\mathbf{x}(t) \stackrel{\mathrm{def}}{=} \begin{bmatrix} \cdots \\ x(t+2) \\ x(t+1) \\ x(t) \\ x(t-1) \\ x(t-2) \\ \cdots \end{bmatrix}, \quad \mathbf{u}(t) \stackrel{\mathrm{def}}{=} \begin{bmatrix} \cdots \\ u(t+2) \\ u(t+1) \\ u(t) \\ u(t-1) \\ u(t-2) \\ \cdots \end{bmatrix}, \tag{26}$$

where again we have $x(t) = \mathbf{w}^{\mathrm{T}}(t)\mathbf{y}(t)$, and $u(t) \stackrel{\mathrm{def}}{=} \mathbf{w}^{\mathrm{T}}(t)\mathrm{rev}[\mathbf{x}(t)]$: the operator rev[·] simply reverses the order of the entries of the operated vector or, equivalently, reverses the time-index.

It is now easy to show that $y(t-i)W(q^{-1},t)$ $W(q,t) = u(t-i)$, thus the natural-gradient version of learning equation (25) writes

$$\Delta\mathbf{w}(t) = \eta\gamma(x(t))\mathbf{u}(t),$$
$$u(t) = \mathbf{w}^{\mathrm{T}}(t)\,\mathrm{rev}[\mathbf{x}(t)]. \tag{27}$$

Such algorithm is closely related to the multiple-constraint algorithm discussed in [54], which performs the projection of current input vector onto previous input vectors to whiten it and to improve convergence speed. This approach finds its theoretical roots for adaptive filtering in [4].

For a real-world FIR adaptive filter with impulse response of length $L+1$, the above expression for $\Delta\mathbf{w}(t)$ suffers from the problem of non-causality, thus the following modification to the above equations can be considered [3]:

$$w_i(t+1) = w_i(t) + \eta\gamma(x(t-L))u(t-i),$$
$$u(t) = \sum_{k=0}^{L} w_{L-k}x(t-k). \tag{28}$$

Such algorithm proves to be useful as it is causal. However, the application of the filter $W(q^{-1},t)W(q,t)$ results in an additional computational effort; also, because of accelerated convergence, it may happen that the minimum it converges first is not a stable one, which also makes the algorithm much sensitive to learning stepsize value.

3. Theoretical analysis and developments

In the following, theoretical aspects of the considered blind channel equalization approaches are analyzed, both for the unconstrained and constrained adaptation schemes. Equilibrium conditions presented here establish transcendental relationships among variables a, λ and constant c at the equilibrium. Finding these relationships is a quite difficult task due to the non-linearity of the equations, thus only approximated results can be issued.

3.1. Unconstrained algorithm steady-state analysis

The steady-state analysis of unconstrained adaptation rule (13) may be performed by studying the mean-field steady-state equation $E_s[\gamma(x)\mathbf{y}] = \mathbf{0}$, where, by definition $\gamma(x) \stackrel{\mathrm{def}}{=} [cb'(x)-1][cb(x)-x]$. The same vector equation is equivalent to the set of scalar conditions

$$E_s[\gamma(x)y(t-i)] = 0, \quad \forall i. \tag{29}$$

Let us suppose the function $\gamma(x)$ be analytical, thus it can be expanded as $\gamma(x) = \sum_{m=0}^{+\infty}\gamma_m x^m$. Besides, each component of $\mathbf{y}(t)$ may be expressed as $y(t) = \sum_{k=-\infty}^{+\infty} h_k s(t-k)$. Since at the equilibrium, corresponding to perfect equalization, it should hold $x(t) = cs(t-\delta)$, conditions (29) rewrite

$$E_s\left\{\sum_{m=0}^{+\infty}\gamma_m[cs(t-\delta)]^m \sum_{k=-\infty}^{+\infty} h_k s(t-i-k)\right\} = 0, \quad \forall i$$

or equivalently

$$\sum_{m=0}^{+\infty}\sum_{k=-\infty}^{+\infty}\gamma_m c^m h_k E_s[s^m(t-\delta)s(t-i-k)] = 0, \quad \forall i. \tag{30}$$

By hypothesis $s(t)$ is a zero-mean IID sequence, that means $E_s[s^m(t-\delta)s(t-i-k)] = E_s[s^{m+1}]$ for $k = \delta - i$ and zero otherwise, thus conditions (30) become

$$\frac{1}{c}\sum_{m=0}^{+\infty}\gamma_m c^{m+1}h_{\delta-i}E_s[s^{m+1}] = 0, \quad \forall i.$$

In virtue of the first of properties (7), surely at least one among the coefficients of $\mathbf{h}$ differs from zero, hence conditions above imply $E_s[x\gamma(x)] = 0$, that gives

$$E_s[x^2b'(x)] + E_s[xb(x)] - cE_s[xb'(x)g(x)] = c. \tag{31}$$

The equilibrium condition of algorithm (12) differs from the one of algorithm (13) only for the

structure of $\gamma(x)$; the original 'Bussgang' algorithm has in fact $\gamma_{\mathrm{B}}(x)=cb(x)-x$, thus its equilibrium condition $E_s[\gamma_{\mathrm{B}}(x)\mathbf{y}]=\mathbf{0}$ reads $E_s[xb(x)]=c$.

The asymptotic stability of the algorithm (13) can be studied by means of the Benveniste–Goursat–Ruget theory presented in [10]. Here stability properties of the algorithm depend on function $\gamma(x)$.

3.2. Constrained algorithm steady-state analysis

The mean-field steady-state analysis theory applied to adaptation equations (22), (18) and (19) give conditions

$$E_s\left[\gamma(x)\left(\mathbf{y}-\frac{x}{\kappa^2}\mathbf{w}\right)\right]=\mathbf{0}, \tag{32}$$

$$E_s[c^2\hat{b}^2(x)-cx\hat{b}(x)]=0, \tag{33}$$

$$E_s[(c\hat{b}(x)-x)(a^2-c^2\hat{b}^2(x))x]=0. \tag{34}$$

From Eq. (32) that may be rewritten as $E_s[\gamma(x)\mathbf{y}]-(1/\kappa^2)E_s[\gamma(x)x]\mathbf{w}=\mathbf{0}$, it is easily seen that optimal equalizing filter has structure

$$\mathbf{w}_\star=\kappa^2\frac{E_s[\gamma(x)\mathbf{y}]}{E_s[\gamma(x)x]}. \tag{35}$$

Since at the equilibrium condition $\mathbf{w}^{\mathrm{T}}\mathbf{w}=\kappa^2$ has to be fulfilled, by using Eq. (35) it is found that

$$\kappa^2E_s[\gamma(x)\mathbf{y}^{\mathrm{T}}]E_s[\gamma(x)\mathbf{y}]=E_s^2[\gamma(x)x]$$

or equivalently:

$$\kappa^2\sum_{i=-\infty}^{+\infty}E_s^2[\gamma(x)y(t-i)]=E_s^2[\gamma(x)x]. \tag{36}$$

By using again expansions as above, it is found that, at the equilibrium

$$E_s[\gamma(x)y(t-i)]=\frac{1}{c}\sum_{m=0}^{+\infty}\gamma_m h_{\delta-i}c^{m+1}E[s^{m+1}],$$

$$E_s[\gamma(x)x]=\sum_{m=0}^{+\infty}\gamma_m c^{m+1}E_s[s^{m+1}]$$

from which it further follows that

$$E_s^2[\gamma(x)y(t-i)]=\frac{1}{c^2}h_{\delta-i}^2E_s^2[\gamma(x)x].$$

Summing both hands of the above equation over index i yields

$$\sum_{i=-\infty}^{+\infty}E_s^2[\gamma(x)y(t-i)]=\frac{1}{c^2}E_s^2[\gamma(x)x]\left(\sum_{i=-\infty}^{+\infty}h_{\delta-i}^2\right).$$

Hence, by using Eq. (36) on the latter condition allows finding the optimal value of the amplitude factor c, that satisfies:

$$c^2=\kappa^2\left(\sum_{i=-\infty}^{+\infty}h_{-i}^2\right). \tag{37}$$

Condition (33) gives a useful and simple relationship among a and λ at the equilibrium. To see what, let us linearize function $\hat{b}(x)$ around the origin, that means taking $c\hat{b}(x)=a\lambda x+\text{h.o.t.}$ This way we obtain

$$c^2\hat{b}^2(x)=a^2\lambda^2x^2+\text{h.o.t.},$$

$$c\hat{b}(x)x=a\lambda x^2+\text{h.o.t.}$$

Then, condition $c^2E_s[\hat{b}^2(x)]=cE_s[\hat{b}(x)x]$ gives, approximately, $a_\star^2\lambda_\star^2E_s[x^2]\cong a_\star\lambda_\star E_s[x^2]$, where $a_\star$ and $\lambda_\star$ represent the values of the parameters at the equilibrium, leading to the restriction

$$a_\star\lambda_\star\cong 1. \tag{38}$$

Furthermore, condition (34) establishes a relationship between the value of λ at equilibrium and the constant c. First, Eq. (34) may be rewritten by using condition (33) as

$$a^2E_s[\hat{b}^2(x)]-a^2E_s[x^2]=E_s[\hat{b}^3(x)x]-E_s[x^2\hat{b}^2(x)]. \tag{39}$$

Expanding function $\hat{b}(x)$ up to the third order, [2] and recalling that near the equilibrium it holds $a\lambda\cong 1$, gives

$$c\hat{b}(x)=x-\frac{1}{3}\lambda^2x^3+\text{h.o.t.},$$

$$c^2\hat{b}^2(x)=x^2-\frac{2}{3}\lambda^2x^4+\frac{1}{9}\lambda^4x^6+\text{h.o.t.},$$

$$c^3\hat{b}^3(x)=x^3-\lambda^2x^5+\frac{1}{3}\lambda^4x^7-\frac{1}{27}\lambda^6x^9+\text{h.o.t.}$$

[2] The third-order approximation of $\tanh(x)$ is usually considered 'good enough' as long as $x^2\leqslant\pi^2/4$; this may be achieved by a proper AGC.

Plugging these approximations in Eq. (39) yields the equivalent condition

$$\lambda^6 E_s[x^{10}] - 6\lambda^4 E_s[x^8] + 12\lambda^2 E_s[x^6] - 18E_s[x^4] \cong 0.$$

By defining source moments μ_m as

$$\mu_m \stackrel{\text{def}}{=} E_s[s^m] = \int_{-\infty}^{+\infty} s^m p_s(s)\,\mathrm{d}s \tag{40}$$

and a new unknown $\xi \stackrel{\text{def}}{=} \lambda c$, the latter condition may be approximated by the 6th order algebraic equation

$$\xi^6\mu_{10} - 6\xi^4\mu_8 + 12\xi^2\mu_6 - 18\mu_4 = 0. \tag{41}$$

If source moments are known up to sixth order, optimal product $\lambda_\star c$ can be evaluated, thus optimal value $\lambda_\star$ may be found as well as $a_\star$. Refined results may be obtained by using a higher-order approximation for $\hat{b}(x)$.

As a numerical example, let us consider the case of zero-mean uniformly distributed source-data with unitary variance. The even moments (40) of source signal uniformly distributed within $[-\sqrt{3}, \sqrt{3}]$ are found to be

$$\mu_{2m} = \frac{3^m}{2m+1},$$

thus, in this case

$$\mu_4 = \frac{9}{5}, \quad \mu_6 = \frac{27}{7}, \quad \mu_8 = 9, \quad \mu_{10} = \frac{243}{11}.$$

The algebraic equation (41) was solved numerically: its unique real roots are $\lambda_\star c \cong \pm 1.3120$. Note that the sign of $\lambda_\star$ does not care since condition $a_\star\lambda_\star = 1$ ensures $\hat{b}(x)$ to be always increasing. Furthermore, under the hypothesis that $\sum_i h_{-i}^2 = 1$ and $\kappa^2 = 4$, formula (37) gives $c = \pm 2$.

3.3. Analysis of steady state of natural-gradient-based learning algorithms

The steady-state analysis of the natural-gradient based algorithms may be performed by considering the equilibrium equation

$$E_s\left[\gamma(x)\sum_{k=-\infty}^{k=+\infty} w_{-k}x(t+k-i)\right] = 0, \quad \forall i. \tag{42}$$

By the usual expansion of function $\gamma(x)$ we get

$$E_s\left\{\sum_{m=0}^{+\infty}\gamma_m[cs(t-\delta)]^m \sum_{k=-\infty}^{k=+\infty} w_{-k}cs(t+k-i-\delta)\right\} = 0, \quad \forall i.$$

This expression leads to $\sum_{m=0}^{+\infty} E_s\gamma_m c^{m+1}s^{m+1}(t-\delta)w_i = 0$ for all the values of i. By invoking the AGC, by virtue of which at least one among the entries of $\mathbf{w}$ must differ from zero, once again we arrive at the steady-state condition $E_s[\gamma(x)x] = 0$, thus the natural-gradient theory does not alter the optimal inverse filter, but only affects the way the learning algorithm looks for it.

3.4. On the cost function and learning stepsize

It is worth examining the structure of the cost function $\tilde{U}$ as given by (11) when the flexible estimator $\hat{b}$ takes the place of the actual one. Expanding function $\hat{b}(x)$ up to the third order gives the following approximation:

$$\tilde{U}(\mathbf{w}) = \frac{1}{2}(a\lambda-1)^2E_s[x^2] - \frac{1}{3}(a\lambda-1)\lambda^3E_s[x^4] + \frac{1}{18}a^2\lambda^6E_s[x^6] + \text{h.o.t.} \tag{43}$$

Far from the equilibrium, the algorithm tries to maximize the fourth moment of the filter output signal under the energy-conserving restriction $\mathbf{w}^{\mathrm{T}}\mathbf{w} = \kappa^2$ as usual [50,57], like that approximately does equalizer (12) with estimator (16), that can be expressed as

$$\Delta\mathbf{w} = \rho\frac{\partial}{\partial\mathbf{w}}E_s\left[\int_0^{\mathbf{w}^{\mathrm{T}}\mathbf{y}}(cb(\xi)-\xi)\,\mathrm{d}\xi\right]. \tag{44}$$

Since near the equilibrium $a\lambda \approx 1$, it asymptotically holds:

$$\tilde{U}(\mathbf{w}) \propto \lambda^4 E_s[(\mathbf{w}^{\mathrm{T}}\mathbf{H}\mathbf{s})^6] \tag{45}$$

that means near the equilibrium, the algorithm tries to minimize the sixth moment of x under the AGC constraint $\mathbf{w}^{\mathrm{T}}\mathbf{w} = \kappa^2$. This resembles the Gray's *variable norm* deconvolution method [28], where the cost function $V_\beta^\alpha = E[x^\alpha]/E^{\alpha/\beta}[x^\beta]$, is used as deconvolution measure. In this case, by assuming $\alpha = 6$ and $\beta = 2$ we may rewrite the expression of the asymptotic cost function (45) exactly as

$\tilde{U} \propto V_2^6/\lambda^2$. Following Satorius and Mulligan [49], it has to be minimized with respect to the deconvolving filter $\mathbf{w}$ as long as $\alpha > \beta$. A more detailed discussion on the considered cost function may be found in the dedicated contribution [25].

Also, on the basis of the same approximated reasoning explained above, the learning step-size $\rho(x)$ given by formula (14) reads

$$\rho(x) = -\eta(a\lambda - a\lambda^3 x^2 - 1 + \text{h.o.t.}),$$

thus, in the mean, we can predict the asymptotic magnitude order of the stepsize as:

$$E_s[\rho] \approx \eta\lambda^2 E_s[x^2] \approx \eta\lambda^2 c^2. \tag{46}$$

4. Exploratory experimental results

The aim of this section is to present computer simulations performed with the above described algorithms in order to gain an insight into their numerical behavior and to compare their performances.

The algorithms are tested to adaptively learn an inverse filter for the sampled telephonic channel described in [10]. Its impulse response $\mathbf{h}$ is depicted in Fig. 6 along with its amplitude and (unwrapped) phase spectra; also, its approximate inverse is shown (along with its spectra); the inverse impulse response has approximately 20 non-negligible samples, thus we assume, as inverse system, a 20-tap FIR filter. The central peak in the shape of $\mathbf{h}$ suggests that $\mathbf{h}$ is non-minimum-phase, as confirmed by Fig. 7 which illustrates the zero-loci of the channel. The channel impulse response was normalized so that $\mathbf{h}^T\mathbf{h} = 1$.

The actual accuracy degree of the deconvolution can be quantitatively measured by means of the residual ISI (inter-symbol interference) defined as in reference paper [50]

$$\text{ISI} \stackrel{\text{def}}{=} \frac{\mathbf{T}^T\mathbf{T} - T_{\max}^2}{T_{\max}^2}, \tag{47}$$

where $\mathbf{T}$ denotes again the convolution between the channel impulse response and the inverse filter impulse response; $T_{\max}$ is the component of $\mathbf{T}$ having the maximal absolute value.

A constant of all the following simulations is that the initial impulse response of the filters $\mathbf{w}(0)$ is assumed as a null sequence, except for the central bar which equals one.

4.1. Simulations with the 'Bussgang' algorithm

It is interesting to see a representation of the Bayesian estimator for uniformly-distributed source data at different levels of distortion powers $E_s[\varepsilon^2]$. The Fig. 8 illustrates estimator (15) for three levels of distortion.

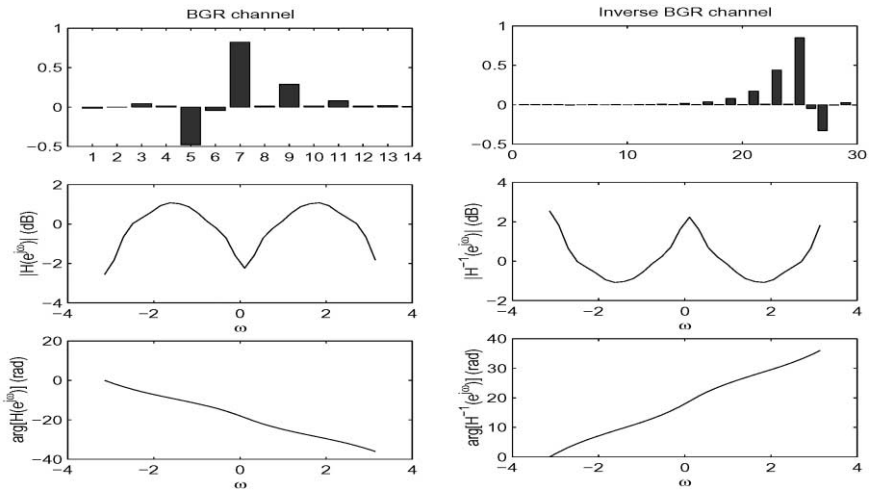

Fig. 6. Sampled telephonic channel response $\mathbf{h}$ and its inverse, along with their amplitude/phase spectra.

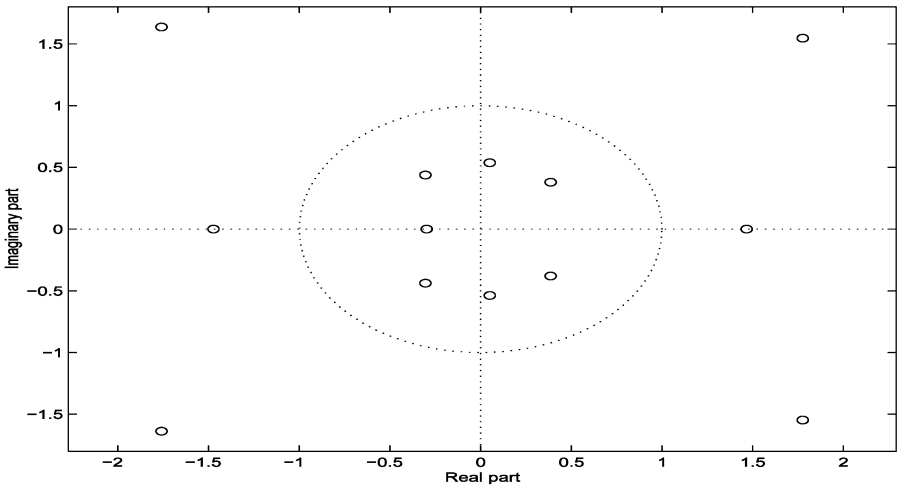

Fig. 7. Zero-plot of the telephonic channel response.

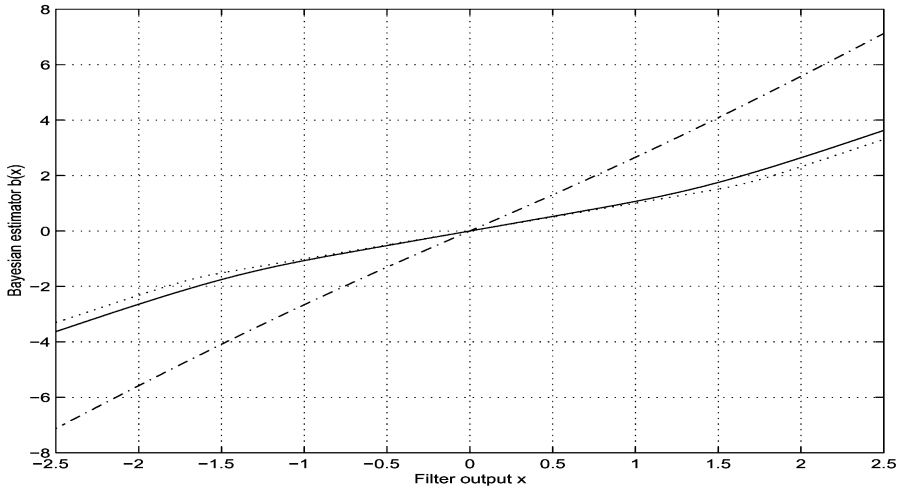

Fig. 8. Bayesian estimator for uniformly-distributed source data (dotted line: $E_s[\varepsilon^2]=0.01$, solid line: $E_s[\varepsilon^2]=0.1$, dot-dashed line: $E_s[\varepsilon^2]=0.8$).

The 'Bussgang' algorithm (12) was implemented with the exact Bayesian estimator (15) and the channel was excited with a white data sequence uniformly distributed within $[-\sqrt{3},+\sqrt{3}]$. As the channel impulse response has unitary energy and the source stream has unitary power, the AGC was simply achieved by iteratively normalizing the inverse filter response to keep $\mathbf{w}^{\mathrm{T}}\mathbf{w}=1$.

In the experiments on 'Bussgang' we monitored both the inverse filter, the overall convolution $\mathbf{T}$, the ISI during inverse filter adaptation and estimated the convolutional noise power σ^2 by freezing the filter at the end of each learning epoch and averaging the square convolutional noise over the whole training set. Figs. 9 and 10 show the numerical results obtained for two different values of distortion: decreasing distortion levels result in better deconvolution results. The numerical results are satisfactory and this simulation is interesting because it allows us to verify that the convolutional

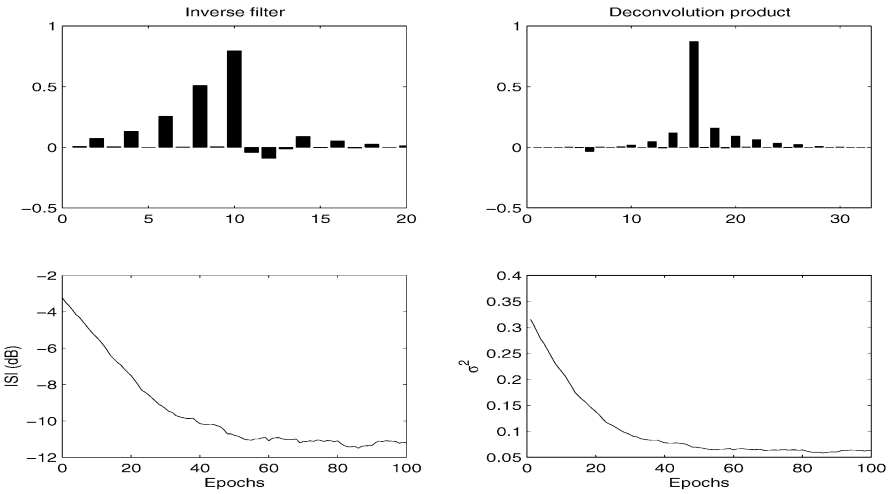

Fig. 9. Simulation results with 'Bussgang' for $E_s[\varepsilon^2] = 0.1$ and $\mu = 0.001$.

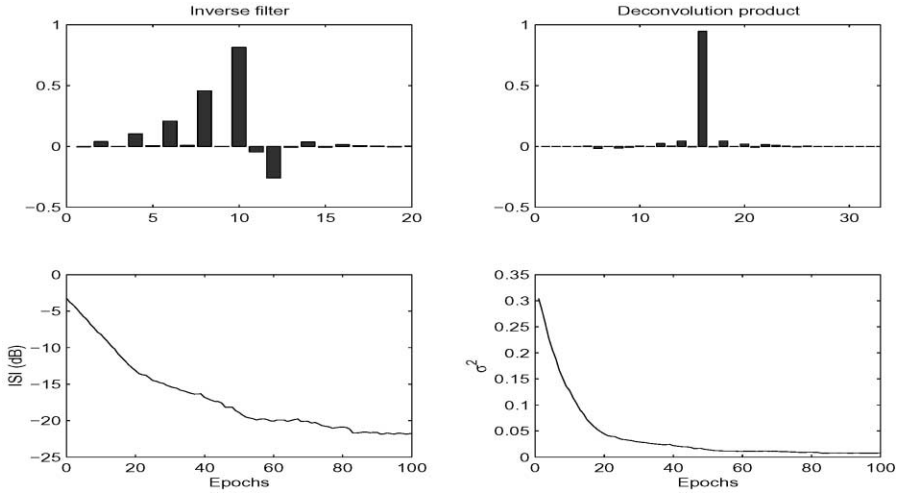

Fig. 10. Simulation results with 'Bussgang' for $E_s[\varepsilon^2] = 0.02$ and $\mu = 0.004$.

noise power varies (of course, decreases) as learning goes on.

4.2. Simulations with the constrained flexible-approximated-estimator based algorithm and comments

Running the algorithm (22) + (18) + (19) needs three 'nominal' learning stepsizes η, η_a and η_λ. We chose three values that experimentally prevent instability and provide fast convergence, namely $\eta = 0.003$, $\eta_a = 0.003$ and $\eta_\lambda = 0.01$. As a filter output amplitude gain we chose $\kappa = 2$ and $a(0) = 1$, $\lambda(0) = 2$.

During the learning phase the behavior of the algorithm has been monitored by computing an error measure $\tilde{U}$ obtained by averaging function U over batches of 200 samples, as well as ISI. Fig. 11

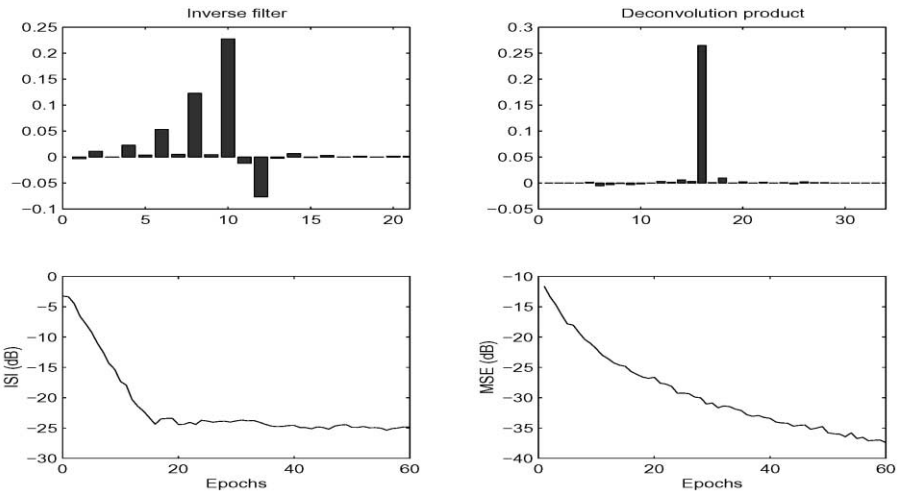

Fig. 11. Simulation results with modified 'Bussgang' (constrained by Lagrange multiplier). The MSE stands for the value of $\tilde{U}$.

shows the learnt filter **w**, the convolutional product **T**, and the courses of ISI and $\tilde{U}$. Clearly the major part of learning happens within the first 15 epochs, then the refinement continues slowly. The learnt inverse filter can be directly compared with the 'exact' inverse filter of Fig. 6. At the end of learning we have ISI $= -25$ dB. Simulation results show that the adaptive-activation function neuron performs better than the static estimator; also, it is easier to implement owing to the very simple expression of the estimator.

It is interesting to verify the theoretical result found about product $a\lambda$ that should tend to one. To this aim, we need to make us sure that the AGC constraint be met with good faith; thus, let us consider the iterative re-normalization (23) instead of the Lagrange-multiplier based one. The course of the product of the two variables during neuron learning is shown in Fig. 12, as well as the value of the adaptive learning stepsize $\rho(x)$ defined as in (14); they have been obtained with $\eta = 0.005$, $\eta_a = 0.01$, $\eta_\lambda = 0.01$, $\kappa = 1$. The initial values of the estimator parameters were $a(0) = 1$ and $\lambda(0) = 1$. The asymptotic values of the parameters are about $a \approx 2.6416$, $\lambda \approx 0.4103$; also, the estimated neuron's output power is $c^2 \approx 0.8360$. The average value of learning stepsize ρ, estimated during neuron's learning, is $E_s[\rho] \approx 3.8580 \times 10^{-4}$. On the basis of the experimental results, formula (46) gives instead the prediction $E_s[\rho] \approx 3.4321 \times 10^{-4}$, which is in excellent accordance with the measured value.

4.3. Simulations with the natural-gradient-based algorithms

The same blind deconvolution problem as above was tackled by natural-gradient based learning algorithms, too. The numerical results we found about natural-gradient modified 'Bussgang' are not completely satisfactory: the main problem is that in the first learning phase the algorithm does converge to the right solution, but after approaching the optimum inverse filter it diverges. An example of this behavior is shown in the Fig. 13: it can be readily seen that the ISI residue (as well as the estimated convolutional noise power) reaches very low values, but suddenly it starts growing. This seems to happen regardless of the learning stepsize magnitude.

It should be mentioned that in [3] a variable stepsize is employed, which grants good convergence in a reasonably number of epochs; however, in this paper we chose not to use ad-hoc learning stepsize changing laws for all the algorithms.

The natural-gradient version of 'Bussgang' with flexible estimator gives no better results, unfortunately. An example of the behavior of the algorithm

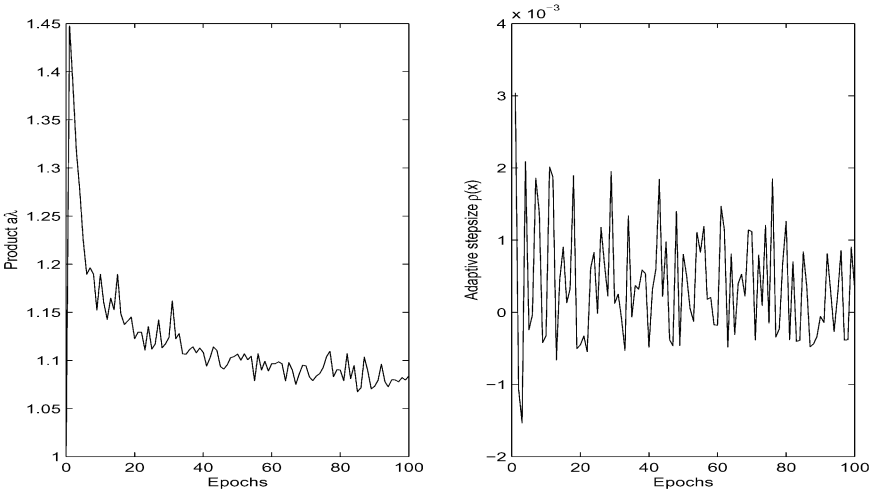

Fig. 12. Product $a\lambda$ and adaptive stepsize $\rho(x)$ for modified 'Bussgang' (constrained by iterative re-normalization).

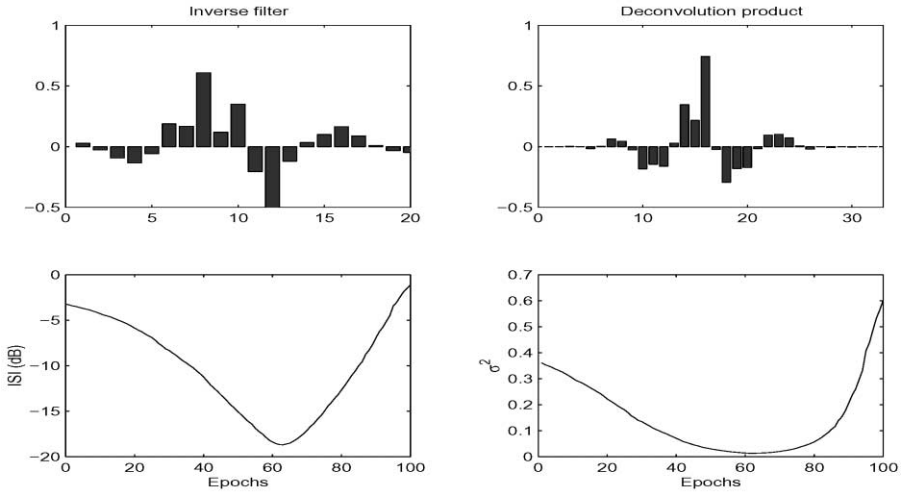

Fig. 13. 'Bussgang' modified by the natural gradient; $E_s[\varepsilon^2] = 0.05$, $\eta = 0.0005$.

is illustrated by Fig. 14; it refers to $\eta_a = \eta_\lambda = 0.001$ and $\eta = 0.0003$; also in this case we took $\kappa = 2$.

4.4. Summary of the results of exploratory experiments and discussion

To summarize the numerical behavior of the discussed algorithms, we reported in the Fig. 15 the ISI curves pertaining to 'Bussgang', neuromorphic 'Bussgang', natural-gradient 'Bussgang', and natural-gradient neuromorphic 'Bussgang' algorithms. The learning parameters have been chosen for the algorithms to better illustrate their normal performances. As can be readily seen, while the natural-gradient based versions do not provide reliable results, the flexible-estimator based deconvolution algorithm performs rather similarly to Bellini's one; indeed, it would be expected that the latter performs slightly better because it uses the *exact* non-linearity as Bayesian estimator.

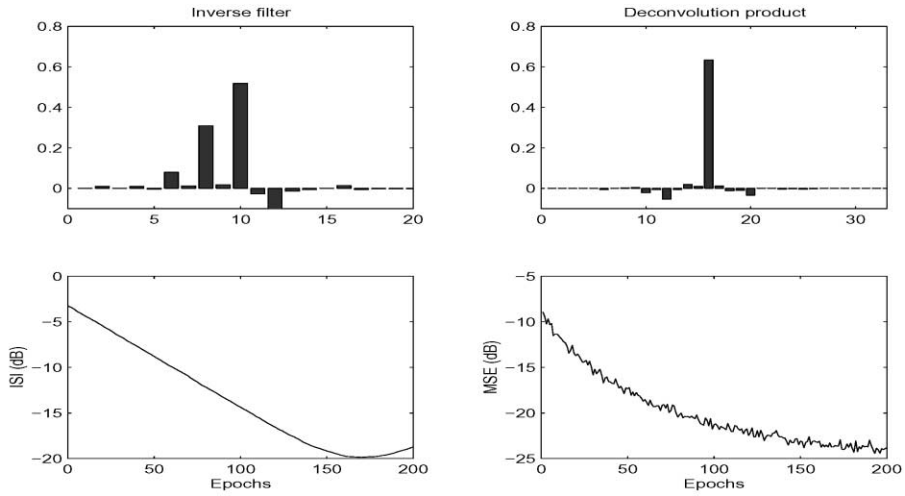

Fig. 14. Natural-gradient version of 'Bussgang' with flexible estimator (constrained by Lagrange multiplier). The MSE stands for the value of $\tilde{U}$.

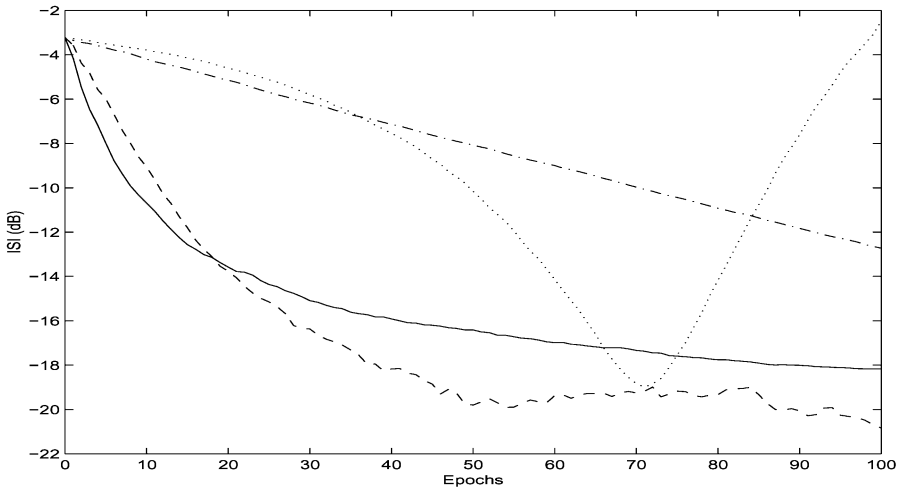

Fig. 15. Comparison of four algorithms. dashed-line: 'Bussgang' ($\eta = 0.004, E_s[\varepsilon^2] = 0.02$); solid-line: neuromorphic 'Bussgang' ($\eta = 0.005,\ \eta_a = 0.003,\ \eta_\lambda = 0.01,\ \kappa = 2,\ a(0) = 1,\ b(0) = 2$); dotted-line: natural-gradient 'Bussgang' ($\eta = 0.0002, E_s[\varepsilon^2] = 0.2$); dot-dashed line: natural-gradient neuromorphic 'Bussgang' ($\eta = 0.0003,\ \eta_a = 0.001,\ \eta_\lambda = 0.001,\ \kappa = 2,\ a(0) = 1,\ b(0) = 2$).

From the previous experiments we can gather some comforting confirmations about the suitability of the theoretical developments and some important information about the numerical performances of the considered algorithms. Some observations may be summarized as follows:

- The flexible-estimator based 'Bussgang' shows a better suitability than Bayesian-estimator based one because it does not require to hypothesize the convolutional noise level.
- The variable learning stepsize stemming from gradient-based optimization of the basic cost function does not seem to produce any particular benefit to learning process as quantity $\rho(x(t))/\eta$ rapidly stabilizes around 10^{-1} thus providing an additional attenuation to the updation process.

Table 1
Estimated computational complexity (flops per iteration, averaged over 100 epochs of 500 samples each) and total computation time (500×100 iterations)

Algorithm	Flops	Total time (s)
'Bussgang'	327.62	142.31
Neuromorphic 'Bussgang'	213.17	46.30
Nat.grad. 'Bussgang'	372.80	150.44
Nat.grad. neuromorphic 'Bussgang'	250.46	59.05

- In the present context, and in absence of any ad-hoc learning stepsize control, the natural-gradient approach does not provide any appreciable improvement over ordinary-gradient based algorithms.

Also, two important elements in the comparison are considered: the number of flops, and the computation time, which also accounts for the computational efforts necessary e.g. to store/retrieve buffer parameters. They refer to a MATLAB code implemented on a 500 MHz machine with 64 MB memory and are reported in Table 1. The numbers of flops do not differ much, while the computation times do: this is because the Bayesian estimator is a complicated function to evaluate and, in addition, the natural gradient approach (albeit optimized as recommended in [3]) requires two additional buffers than the other algorithms.

On the basis of these observations, in the following section we focus the experiments on flexible (neuromorphic) estimators based algorithm with non-adaptive learning stepsize (i.e. pseudo-LMS) on some interesting practical cases.

5. Some experiments on 'Bussgang' with adaptive estimator

To start with, we consider the simpler pseudo-LMS expression of the 'Bussgang' algorithm with adaptive estimator, given by

$$\Delta\mathbf{w} = -\eta[c\hat{b}(x) - x]\mathbf{y}, \quad \mathbf{w} \leftarrow \frac{|\kappa|\mathbf{w}}{||\mathbf{w}||} \tag{48}$$

plus Eqs. (18) and (19) to adapt the coefficients of the estimator (16). An example of the results obtainable with the simplified 'Bussgang'endowed with approximated flexible estimator is illustrated by Fig. 16; they have been obtained with $\eta = 0.005$, $\eta_a = 0.005$, $\eta_\lambda = 0.005$, $a(0) = 1$, $\lambda(0) = 1$, $\kappa = 1$.

An important parameter in the design of adaptive filter is the length $L+1$ of neuron's buffer; through computer simulations we found that when the buffer size exceeds 20 the performances in term of ISI of the neuron does not ameliorate anymore.

5.1. Behavior on noisy/non-stationary channels

We wish to investigate on the behavior of 'Bussgang' algorithm with adaptive flexible estimator when operated on noisy or non-stationary channels.

In the previous simulations, we supposed the channel noise $\mathcal{N}(t)$ to be null. If the additive noise is non-null but is uncorrelated from the observed signal, it may be partially removed by a correlation filter before deconvolution. This is a basic issue discussed in many papers and book [30,31]; we wish to cite here, as an interesting reference, the very clever and pervasive application of uncorrelated noise removal not strictly related to blind deconvolution by Vaz and Thakor [58], developed for the very low signal-to-noise ratio measures treated in evoked potentials estimation and analysis.

To show the effects of residual noise, here we consider a white Gaussian additive noise that affects the channel as in [50]. Three different situations have been considered, so that the signal-to-noise ratio (SNR) equals 20, 10 and 5 dB, respectively. The simulations have been carried out with $\eta = \eta_a = \eta_\lambda = 0.0005$, and $\kappa = 1$; any epoch counts 500 learning samples. Results are shown in the Fig. 17. The algorithm is quite robust with respect to considered noise, but it usually diverges for stronger additive noise.

Also, let us consider operation over a non-stationary channel. To this aim, we introduce another channel impulse response, used by Shalvi–Weinstein in [50], referred to as SW, which represents an all-pass filter; the Benveniste–Goursat–Ruget telephonic channel will be in the following referred to as BGR channel for easier notation.

In order to simulate a non-stationary (switching) channel, we supposed that a BGR channel switches, after 50 epochs, to a SW channel; it suddenly switches back to BGR channel after 50 epochs, and

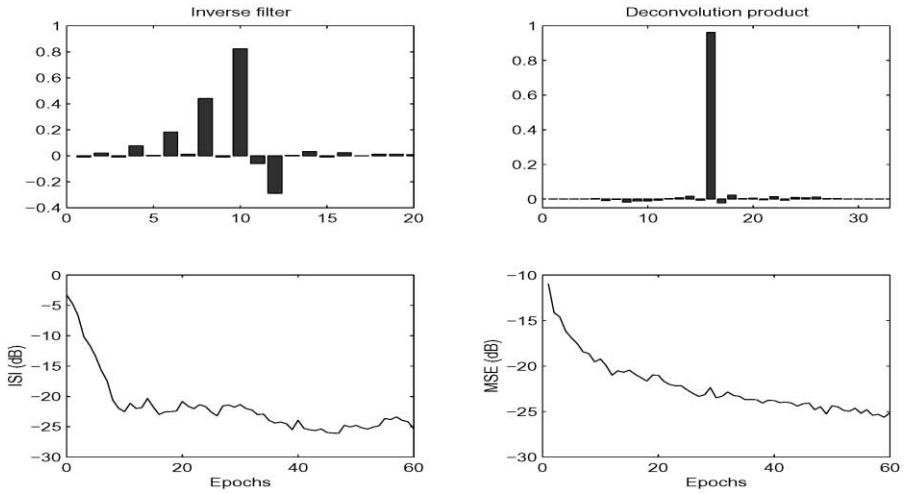

Fig. 16. An example of the results obtainable with the simplified 'Bussgang' algorithm with approximated flexible estimator. The MSE stands for the value of $\tilde{U}$.

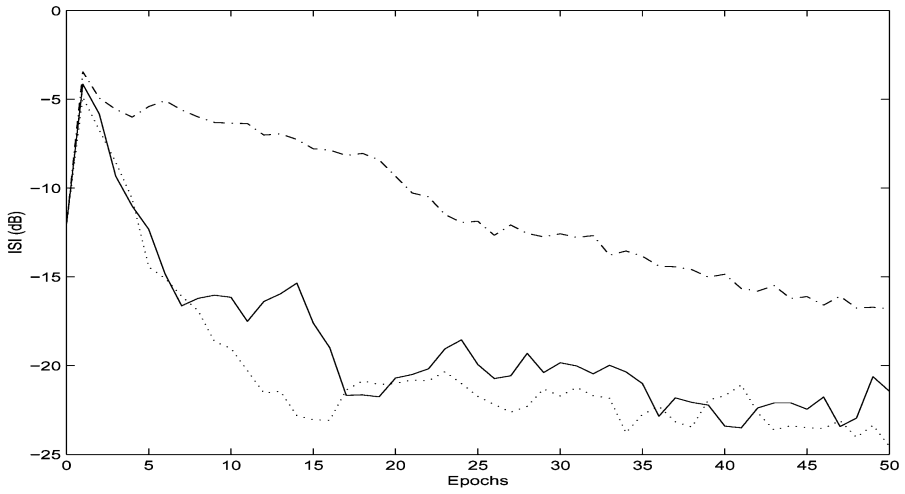

Fig. 17. Simplified 'Bussgang' with adaptive Bayesian estimator (dot-dashed line: SNR = 5 dB; solid line: SNR = 10 dB; dotted line: SNR = 20 dB).

so forth. The obtained results are shown in Fig. 18; they refer to $\eta = 0.002$, $\eta_a = 0.001$, and $\eta_\lambda = 0.001$.

5.2. Behavior on non-uniformly distributed source data

We wish to investigate on the behavior of 'Bussgang' algorithm with adaptive flexible estimator when non-uniformly distributed source data are dealt with.

Results of Fig. 19 refer to a 2-PAM source stream for $\eta = 0.01$, $\eta_a = 0.01$, $\eta_\lambda = 0.01$, $a(0) = 1$, $\lambda(0) = 1$ (here an epoch counts 200 samples). It is worth noting that the algorithm quickly converges to the expected inverse filter for the noisy channels as long as SNR > 5 dB.

In this special case, we can perform a comparison with one of the most efficient and well-known method for SISO and SIMO (multi-channel) blind equalization, that is the Godard's constant-modulus

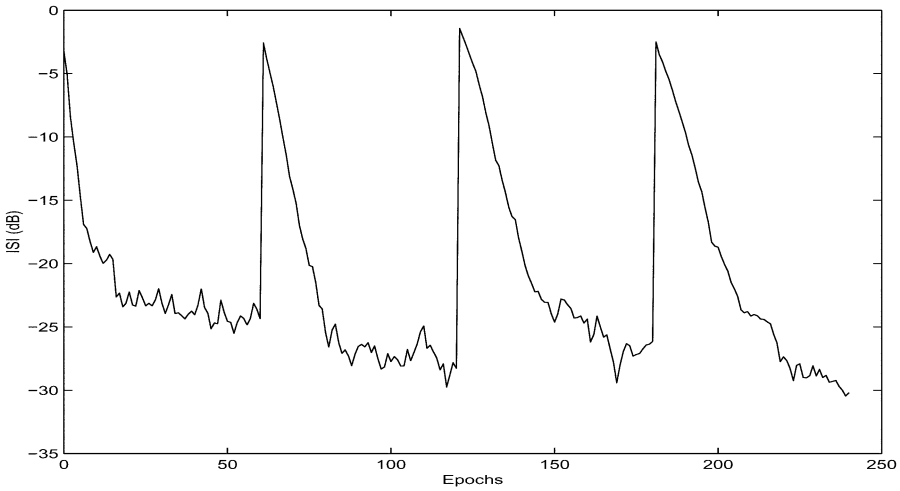

Fig. 18. Simplified 'Bussgang' with adaptive Bayesian estimator operating on a switching channel.

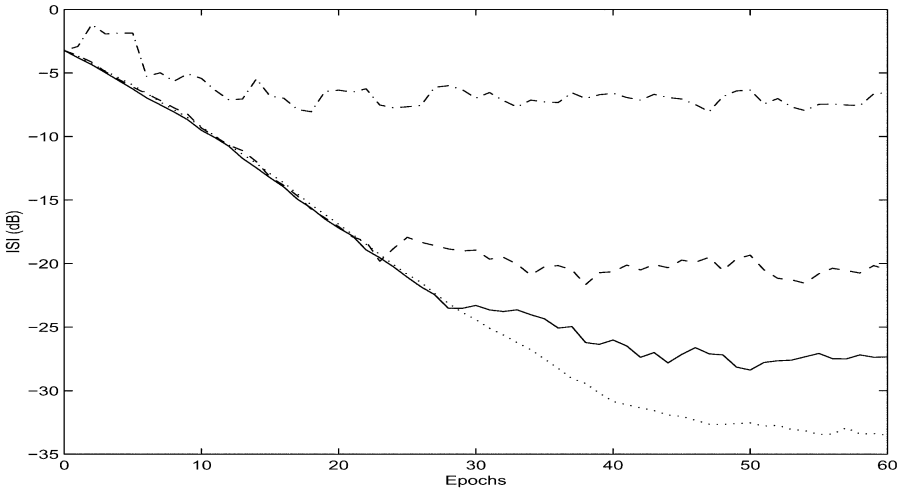

Fig. 19. Simplified 'Bussgang' with flexible estimator on 2–PAM source data and noisy channel (dot-dashed line: SNR= 1 dB; dashed line: SNR= 5 dB; solid line: SNR= 10 dB, dotted line: SNR= 20 dB).

algorithm (CMA) (see e.g.[27,55] and references therein). It is interesting to notice here that CMA belongs to the class of 'Bussgang' algorithms (25) in that it writes, with the usual notation

$$\Delta \mathbf{w} = \eta \gamma(x)\mathbf{y}.$$

Function $\gamma(\cdot)$ denotes the non-linearity which determines the capability of the algorithm; in the case of CMA for real-valued signals it reads $\gamma(x) = (A - x^2)x$, with A being the constant modulus [3].

Comparative results are reported in the Fig. 20, where the learning parameters have been chosen in order for the algorithms to exhibit good trade-off between convergence speed and steady-state error value. The flexible-estimator based 'Bussgang'

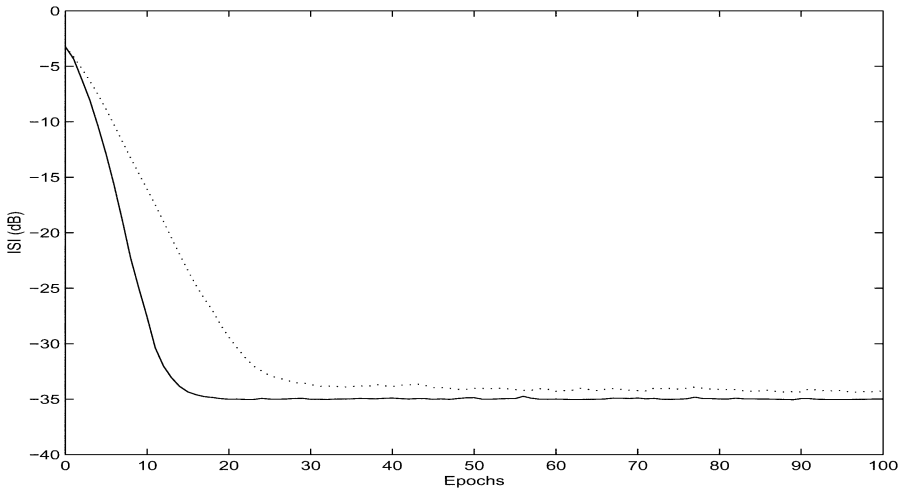

Fig. 20. Comparison of two algorithms on 2–PAM source data. Solid line: simplified flexible-estimator based 'Bussgang' ($\eta = 0.004$, $\kappa = 1$, $a(0) = 1$, $b(0) = 1$, $\eta_a = 0.001$, $\eta_\lambda = 0.01$); dotted line: CMA ($\eta = 0.0005$, $A = 1$).

algorithm show similar performances to the CMA one.

6. Conclusion and further work

In this paper a new adaptive blind channel equalization method for uniformly distributed source sequence, based on the Bellini's Bayesian estimation technique, was presented. We proposed to overcome the problem of the correct deconvolution noise power assumption, by introducing self-tuned parameters in a suitable approximation of the original estimator. Finally, we showed through simulations that the pseudo-LMS optimization algorithm together with a self-tuning mechanism give a fast and accurate blind equalization algorithm, whose performance may be well explained using the proposed mean-field steady-state analysis. However, the theoretical analysis given here may yield only approximately matching results since it involved approximations of true functions and an IIR equalizing filter model.

The heaviest restriction of the theory developed here is that it is suited only for uniformly distributed source signals, thus it cannot cope with any kind of source sequences: estimator (16) is not a 'universal' approximating one, and suggests that a better flexible function, endowed with more parameters to adapt, would be more effective and useful. This enhancement, suggested and already proven useful in blind density shaping and in blind source separation of non-convolutional (instantaneous) mixtures, is currently under investigation.

Also, the extension of the presented algorithms to complex-valued channel equalization and system inversion are of interest [18], and this would open room for a comparison with well-established theoretical and practical results, such as the Godard's CMA algorithm (eventually combined with decision-direct feedback equalizer).

References

[1] A. Alkulaibi, J.J. Soraghan, Hybrid higher-order cepstrum and functional link network-based blind equaliser, Signal Processing 62 (1997) 101–109.
[2] S.-I. Amari, Natural gradient works efficiently in learning, Neural Computation 10 (1998) 251–276.
[3] S.-I. Amari, S. Douglas, A. Cichocki, H.H. Yang, Novel on-line algorithms for blind deconvolution using natural gradient approach, in: Proceedings of the 11th IFAC Symposium on System Identification, SYSID-97, Kitakyushu, Japan, 8–11 July 1997, pp. 1057–1062.
[4] B. Baykal, A.G. Constantinides, Undetermined-order recursive least-squares adaptive filtering, the concept and algorithm, IEEE Trans. Signal Process. 45 (2) (1997) 346–362.

[5] B. Baykal, O. Tanrikulu, J.A. Chambers, A.G. Constantinides, A new family of blind adaptive equalization algorithms, IEEE Comm. Lett. 3 (4) (1999) 109–110.
[6] S. Bellini, Blind Equalization, Alta Frequenza 57 (1988) 445–450.
[7] A.J. Bell, T.J. Sejnowski, An information maximisation approach to blind separation and blind deconvolution, Neural Computation 7 (6) (1995) 1129–1159.
[8] S. Bellini, Bussgang techniques for blind equalization, in: IEEE Global Telecommunication Conference Records, December 1986, pp. 1634–1640.
[9] A. Benveniste, M. Goursat, Blind Equalizers, IEEE Trans. Commun. COM-32 (8) (August 1984) 871–883.
[10] A. Benveniste, M. Goursat, G. Ruget, Robust Identification of a nonminimum phase system: blind adjustment of a linear equalizer in data communication, IEEE Trans. Automatic Control AC-25 (3) (June 1980) 385–399.
[11] R. Blumm, Multidimensional stochastic approximation methods, Ann. Math. Statist. 25 (1954) 737–744.
[12] J.A. Cadzow, Blind deconvolution via cumulant extrema, IEEE Signal Process. Mag 13, No. 3 (May 1996) 24–42.
[13] C. Cafforio, C. Prati, F. Rocca, Full resolution focusing of Seasat SAR images in the frequency-wavenumber domain, Internat. J. Remote Sensing 12 (1991) 491–510.
[14] P. Campolucci, S. Fiori, A. Uncini, F. Piazza, A new IIR-MLP learning algorithm for on-line signal processing, Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 1997, pp. 3293–3356.
[15] J.-W. Cho, S.-Y. Lee, Analog neuro-chip with on-chip learning capability for adaptive non-linear equalizers, Proceedings of the International Joint Conference on Neural Networks (IJCNN), 1998, pp. 581–586.
[16] S. Choi, S. Ong, J. Cho, C. You, D. Hong, Performances of neural equalizers on partial erasure model, IEEE Trans. Magnetics 33 (5) (September 1997) 2788–2790.
[17] J.B. Destro Filho, G. Favier, J.M. Travassos Romano, New Bussgang methods for blind equalization, Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 1997, pp. 2269–2272.
[18] Z. Ding, Y. Li, Blind Equalization and Identification, Marcel Dekker, New York, 2001.
[19] S.C. Douglas, A. Cichocki, S.-I. Amari, Self-whitening algorithms for adaptive equalization and deconvolution, IEEE Trans. Signal Process. 47 (4) (April 1999) 1161–1165.
[20] S. Fiori, Blind deconvolution by spectral weighted least-squares technique, Electron. Lett. 35 (10) (May 1999) 776–777.
[21] S. Fiori, Neural networks for blind signal processing, Ph.D. Thesis. Department of Electrical Engineering, University of Bologna, Italy, March 2000, (in English).
[22] S. Fiori, Blind signal processing by the adaptive activation function neurons, Neural Networks 13 (6) (August 2000) 597–611.
[23] S. Fiori, Notes on cost functions and estimators for 'Bussgang' adaptive blind equalization, Eur. Trans. Telecomm. (ETT), (2002), forthcoming.
[24] S. Fiori, P. Bucciarelli, Probability density estimation using adaptive activation function neurons, Neural Process. Lett. 13 (1) (February 2001) 31–42.
[25] S. Fiori, G. Maiolini, Weighted least-squares blind deconvolution of non-minimum phase systems, IEE Proc.—Vision, Image Signal Process. 147 (6) (December 2000) 557–563.
[26] R. Godfrey, F. Rocca, Zero memory nonlinear deconvolution, Geophys. Prospecting 29 (April 1981) 189–228.
[27] D.N. Godard, Self recovering equalization and carrier tracking in two-dimensional data communication system, IEEE Trans. Comm. COM-28 (11) (November 1980) 1867–1875.
[28] W. Gray, Variable norm deconvolution, Ph.D. Thesis, Department of Geophysics, Stanford University, 1979.
[29] P.R. Hiesinger, M. Scholz, I.A. Meinertzhagen, K.-F. Fischbach, K. Obermayer, Visualization of synaptic markers in the optic neuropils of Drosophila using a new constrained deconvolution method, J. Comp. Neurol. 429 (2001) 277–288.
[30] S. Haykin, Blind deconvolution, in: S. Haykin (Ed.), Adaptive Filter Theory, Prentice-Hall, Englewood cliffs, NJ, 1991 (Chapter 20).
[31] S. Haykin, Bussgang Techniques for Blind Deconvolution and Equalization, in: S. Haykin (Ed.), Blind Deconvolutiom, Prentice-Hall, 1994 (Chapter 2).
[32] T. Kailath, From matched filters to martingales, IEEE Inform. Theory Newslett. 48 (4) (December 1998).
[33] D. Kundur, D. Hatzinakos, Blind image deconvolution, IEEE Signal Process. Mag. 13 (3) (May 1996) 43–64.
[34] M. Ibnkahla, J. Sombrin, F. Castanie, N.J. Bershad, Neural networks for modelling nonlinear memoryless channels, IEEE Trans. Comm. 45 (7) (July 1997) 768–771.
[35] R.H. Lambert, C.L. Nikias, Blind equalization cost functions and maximum a-posteriori, Proceedings of the IEEE MILCOM Conference on Military Communications, Ft. Monmouth, NJ, USA, 1994, pp. 291–295.
[36] F.K. Li, D. Held, J. Curlander, C. Wu, Doppler parameter estimation for spaceborne synthetic aperture radar, IEEE Trans. Geosci. Remote Sensing GE-23 (1985) 47–51.
[37] Y. Li, Z. Ding, Convergence analysis of finite length blind adaptive equalizers, IEEE Trans. Signal Process. 43 (September 1995) 2120–2129.
[38] O. Macchi, E. Eweda, Convergence analysis of self-adaptive equalizers, IEEE Trans. Inform. Theory IT-30 (2) (March 1984) 161–176.
[39] R. Makowski, A blind deconvolution method, Proceedings of Signal Processing V: Theories and Applications (EUSIPCO), Vol. III, 1990, pp. 1959–1962.
[40] F. Mazzenga, On the identifiability of a channel transfer function from cyclic and conjugate cyclic statistics, European Trans. Telecomm. (ETT) 11 (3) (May/June 2000) 293–296.

[41] R. Molina, A. Katsaggelos, J. Abad, J. Mateos, A Bayesian approach to blind deconvolution based on Dirichlet distributions, Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vol. 4, 1997, pp. 2809–2812.
[42] E. Oja, Nonlinear PCA criterion and maximum likelihood in independent component analysis, Proceedings of the Independent Component Analysis (ICA'98), 1998, pp. 143–148.
[43] S. Ong, S. Choi, C. You, D. Hong, A decision feedback recurrent neural equalizer for digital communication, IEEE Trans. Magnetics 33 (5) (September 1997) 2767–2769.
[44] W. Paquier, M. Ibnkahla, Self-organizing maps for rapidly fading nonlinear channel equalization, Proceedings of the International Joint Conference on Neural Networks (IJCNN'98), 1998, pp. 865–869.
[45] R. Parisi, E.D. Di Claudio, G. Orlandi, B.D. Rao, Fast adaptive digital equalization by recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (November 1997) 2731–2739.
[46] G. Picchi, G. Prati, Blind equalization and carrier recovery using a "Stop-and-Go" decision-direct algorithm, IEEE Trans. Comm. COM-35 (9) (September 1987) 877–887.
[47] H. Robbins, S. Monro, A stochastic approximation method, Ann. Math. Statist. 22 (1951) 400–407.
[48] Y. Sato, A method for self-recovering equalization for multilevel amplitude-modulation system, IEEE Trans. Comm. COM-23 (June 1975) 679–682.
[49] E.H. Satorius, J.J. Mulligan, Minimum entropy deconvolution and blind equalisation, Electron. Lett. 28 (16) (July 1992) 1534–1535.
[50] O. Shalvi, E. Weinstein, New criteria for blind deconvolution of nonminimum phase systems (channels), IEEE Trans. Inform. Theory IT-36 (2) (March 1990) 312–321.
[51] V. Shtrom, H. Fan, A refined class of cost functions in blind equalization, International Conference on Acoustics, Speech and Signal Processing (ICASSP) 3 (1997) 2273–2276.
[52] I. Sabala, A. Cichocki, S.-I. Amari, Relationships between instantaneous blind source separation and multichannel blind deconvolution, Proceedings of International Joint Conference on Neural Networks (IJCNN), (1998) pp. 39–44.
[53] R. Steele, Mobile radio communications, IEEE Pentec Press, Piscataway (NJ, USA) 1992.
[54] O. Tanrikulu, B. Baykal, A.G. Constantinides, J.A. Chambers, Soft constrained satisfaction (SCS) blind channel equalization algorithms, Internat. J. Adaptive Control Signal Process. 12 (1998) 117–134.
[55] J.R. Treichler, B.G. Agee, A new approach to multipath correction of constant modulus signals, IEEE Trans. Acoust. Speech, Signal Process. ASSP-31 (2) (April 1983) 459–472.
[56] K. Torkkola, Blind deconvolution, Information maximization and recursive filters, Proceedings of International Conference on Acoustics, Speech and Signal Processing, 1997, pp. 3301–3304.
[57] J.K. Tugnait, Identification of linear stochastic systems via second- and fourth-order cumulant matching, IEEE Trans. Inform. Theory IT-33 (3) (May 1987) 393–407.
[58] C.A. Vaz, N.V. Thakor, Adaptive Fourier estimation of time-varying evoked potentials, IEEE Trans. Biomed. Eng. 36 (4) (April 1989) 448–455.
[59] S. Vembu, S. Verdù, R.A. Kennedy, W. Sethares, Convex cost functions in blind equalization, IEEE Trans. Signal Process. 42 (8) (August 1994) 1952–1959.
[60] H.H. Yang, S.-I. Amari, Blind equalization of switching channels by ICA and learning of learning rate, Proceedings of International Conference on Acoustics, Speech and Signal Processing, Vol. III, 1997, pp. 1849–1852.
[61] V. Weerackody, S.A. Kassam, K.R. Laker, Convergence analysis of an algorithm for blind equalization, IEEE Trans. Comm. COM-39 (6) (June 1991) 856–865.
[62] R.A. Wiggins, Minimum entropy deconvolution, Geoexploration 16 (1978) 21–35.