ELSEVIER

Neurocomputing 48 (2002) 763–778

NEUROCOMPUTING

www.elsevier.com/locate/neucom

Blind deconvolution by simple adaptive activation function neuron

Simone Fiori*

Neural Networks and Adaptive Systems Research Group - Dept. of Industrial Engineering, University of Perugia, Perugia, Italy

Received 16 September 2000; accepted 13 August 2001

Abstract

The 'Bussgang' algorithm is one among the most known blind deconvolution techniques in the adaptive signal processing literature. It relies on a Bayesian estimator of the source signal that requires the prior knowledge of the source statistics as well as the deconvolution noise characteristics. In this paper, we propose to implement the estimator with a simple adaptive activation function neuron, whose activation function is endowed with one learnable parameter; in this way, the algorithm does not require to hypothesize deconvolution noise level. Neuron's weights adapt through an unsupervised learning rule that closely recalls non-linear minor component analysis. In order to assess the effectiveness of the proposed method, computer simulations are presented and discussed. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Blind digital deconvolution; Adaptive activation function neurons; Unsupervised learning; Neurons with input tapped-delay line; Filtering synapses; Minor component analysis

1. Introduction

Many physical phenomena may be described by a discrete convolutional model, as for instance in electrical communications by transmission channels [5,6,22], geophysical measurements and prospecting [21,37,43], and image distortion description [27,30]. The inverse problem associated to these forward models consists in

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

PII: S0925-2312(01)00672-5

recovering the signal distorted from the physical mean that it propagates within. This may be achieved by estimating the system's impulse response and successively computing its inverse; such method involves the identification of the system, which may be problematic if it is non-stationary (as in telecommunication channels) and because of the disturbances.

Blind digital deconvolution [5–7,18,22,25,41] focuses on the problem of recovering a source signal distorted by a linear system from observations of the system's response only, with little knowledge about its impulse response and about the statistics and temporal characteristics of the source. In vector notation, the linear system's input/output model writes:

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

where $\mathbf{s}(t)$ is a vector containing the input samples:

$$s(t),\ s(t-1),\ s(t-2),\ldots,s(t-\ell+1),$$

with ℓ being the number of entries in $\mathbf{h}$, and $N(t)$ is a zero-mean additive noise that originates from many simultaneous effects, as additive external disturbance, measurement errors, sampling and roundoff errors. The following minimal hypotheses are usually made about the channel and data stream: Channel's impulse response satisfies $\mathbf{h}^{\mathrm{T}}\mathbf{h} = 1$ and its inverse has finite energy, $E_s[s(t)] = 0$, $E_s[s^2(t)] = 1$; also, if $p_s(s)$ denotes the probability density function (PDF) of the input signal, it is supposed that $p_s(s) = p_s(-s)$ and that it is non-Gaussian.

A filter described by its impulse response $\mathbf{w}$ is the inverse of system (1) if $\mathbf{w}$ cancels the effects of $\mathbf{h}$ on the source signal. Denoting by $\mathbf{x}(t)$ the vector containing the samples:

$$x(t),\ x(t-1),\ x(t-2),\ldots,x(t-m+1),$$

where m is the number of tap-weights in $\mathbf{w}$, the output of the filter writes:

$$z(t) = \mathbf{w}^{\mathrm{T}}(t)\mathbf{x}(t). \tag{2}$$

Since $\mathbf{h}$ and $s(t)$ are unknown, the filter $\mathbf{w}_\star$ such that $z(t) \sim s(t)$ has to be *blindly* identified possibly by means of an iterative algorithm [5,24]. When $\mathbf{h}$ represents a non-minimum phase system; however, its inversion cannot be exactly performed by means of the FIR filter (2), therefore every time an FIR structure is used an approximation error must be taken into account [6,24]. In formulas, we get

$$z(t) = As(t-\delta) + n(t), \tag{3}$$

where $n(t)$ is the so-called *deconvolution noise*, A is an amplitude factor and δ is a finite delay. A suitable representation of $n(t)$ is a zero-mean Gaussian random process [24] with variance denoted here with σ^2, that is, the PDF of the convolutional noise reads $p_n(n) = (1/\sqrt{2\pi}\sigma)\mathrm{e}^{-n^2/2\sigma^2}$.

The convolutional noise $n(t)$ takes into account four concurrent phenomena:

(1) During filter adaptation $\mathbf{w} \neq \mathbf{w}_\star$: The more the filter's impulse response differs from the optimal one, the larger is the deconvolution noise variance.
(2) The filter (2) might be unsuitable because the length m does not suffice to represent system's inverse.
(3) The model (2) might be unsuitable because of the disturbance $N(t)$.
(4) The system's impulse response might be slowly time-varying, i.e. $\mathbf{h} = \mathbf{h}(t)$.

Following the pioneering work of Sato [38], several blind deconvolution algorithms have been proposed through years. One of the most known algorithms is the 'Bussgang' one by Bellini [5,24], based on a memoryless Bayesian estimation of source data by the observation of inverse system's output and a LMS-style adjustment of inverse system's impulse response under the hypothesis of independent identically distributed (IID) source sequences. In the original Bellini theory, optimal estimator depends on the statistics of the deconvolution noise and of the source sequence, thus models of them are required [5]. He considered different cases [25], and recently Destro-Filho et al. developed a special algorithm suited for binary sources [11]. Under the hypothesis that $n(t)$ is Gaussian and the source sequence has a uniform distribution [5,25], the estimator depends on σ^2, considered as a constant value to be decided during the design phase.

In the recent years, neural networks based deconvolution methods have been developed, see for instance [1,4,23,28,33,35] and references therein. In particular, in [20] we recently presented a detailed study about a neuromorphic digital filtering theory based on the fusion of digital filters and unsupervised neural networks, where the adaptivity of the latter proves to grow the learning ability and computational power of the simpler digital structure.

In the present paper, we propose a self-tuning procedure that allows to automatically determine the optimal parameter of a flexible approximated estimator for 'Bussgang' technique. Such self-tuning behavior allows to overcome the problem of manually fixing a suitable value of σ^2. The estimator is structured according to an adaptive activation function neural network theory recently developed in [14]: It is embodied in the non-linear activation function of a simple neuron endowed with an input buffer for time-series processing.

2. Overview of 'Bussgang' algorithm

From filter output signal model (3), a way can be envisaged to get an estimate of the source sequence $s(t)$, knowing $z(t)$, by means of a statistical estimator: In fact, the model (3) reveals that the relationship between $z(t)$ and $s(t)$ is deterministic but for the convolutional noise. In symbols, we write the estimate $g(z(t))$, with $g(\cdot)$ being the stochastic estimator. Let us suppose that $\mathbf{w}$ is randomly initialized, thus initially the inverse filter performs poorly and the convolutional noise is strong. Consequently, the source signal estimation would not be reliable; however, this is the only information available to the algorithm and can be used to refine the filter's parameters and to lessen deconvolution noise; this, in turn, allows a better source estimation and so forth. Such iteration terminates with the optimal inverse filter and the lowest convolutional noise. It is important to remark that $g(\cdot)$ should be an injective function in order to give unambiguous estimates, and should look as an odd function to match source PDF's symmetry; also, usually $b(0)=0$ because deconvolution noise is zero-mean. (A general study about the minimal properties of a generic estimator for 'Bussgang' filtering has been recently proposed in [19].)

As a method to find iteratively the optimal filter given observations of the channel output $z(t)$, an LMS-style algorithm was used [5,24]: By interpreting the difference $g(z)-z$ as an 'error', the structure of the algorithm proposed by Bellini is

$$\Delta \mathbf{w} = \mu[g(z)-z]\mathbf{x}, \tag{4}$$

where μ is a positive learning stepsize.

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

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

$$\mathrm{MSE}[\hat{v}_x] \stackrel{\text{def}}{=} \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} [v_x - \hat{v}_x(v_y)]^2 p_{v_x,v_y}(v_x,v_y)\,\mathrm{d}v_x\,\mathrm{d}v_y.$$

It is important to note that $\hat{v}_x(\cdot)$ is not a single value, but a *function*. In order to solve the above variational problem, let us introduce the conditional PDF:

$$p_{v_x|v_y}(v_x|v_y) \stackrel{\text{def}}{=} \frac{p_{v_x,v_y}(v_x,v_y)}{p_{v_y}(v_y)},$$

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

$$\hat{v}_x(v_y) = \int_{-\infty}^{+\infty} v_x\, p_{v_x|v_y}(v_x|v_y)\,\mathrm{d}v_x.$$

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

$$\hat{s} = g(z) \stackrel{\text{def}}{=} E_s[s|z] = \int_{-\infty}^{+\infty} s\, p_{s|z}(s|z)\,\mathrm{d}s, \tag{5}$$

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

$$p_{s|z}(s|z) = \frac{p_{z|s}(z|s)\,p_s(s)}{p_z(z)} = \frac{p_{z|s}(z|s)\,p_s(s)}{\int_{-\infty}^{+\infty} p_{z|s}(z|s)\,p_s(s)\,\mathrm{d}s},$$

where $p_{z|s}(z|s)$ statistically describes the formation of system output values from the knowledge of input ones. As we already observed, the system model is deterministic but for the convolutional noise, thus the latter probability density function writes $p_{z|s}(z|s) = p_n(n) = p_n(z - As)$; then, the estimator is given by

$$g(z) = \frac{\int_{-\infty}^{+\infty} s\,p_n(z - As)\,p_s(s)\,\mathrm{d}s}{\int_{-\infty}^{+\infty} p_n(z - As)\,p_s(s)\,\mathrm{d}s}.$$

Owing to the form of $p_n(n)$, the integrals above might be tractable and, as a meaningful result, it is possible to write $g(z)$ as a function of z and $p_z(z)$ only. In fact, note that the following identity holds:

$$\frac{\mathrm{d}}{\mathrm{d}z}\int_{-\infty}^{+\infty} p_n(z - As)\,p_s(s)\,\mathrm{d}s = -\frac{z}{\sigma^2}\int_{-\infty}^{+\infty} p_n(z - As)\,p_s(s)\,\mathrm{d}s + \frac{A}{\sigma^2}\int_{-\infty}^{+\infty} s p_n(z - As)\,p_s(s)\,\mathrm{d}s.$$

As mentioned, the Bellini's expression for Bayesian estimator $g(z)$ is dependent upon the deconvolution noise power σ^2 [5,24]; in fact, in its general form, it writes:

$$g_{\mathrm{B}}(z) = \frac{\sigma^2}{A}\left[\frac{p_z'(z)}{p_z(z)} + \frac{z}{\sigma^2}\right], \quad p_z(z) = \frac{1}{A}p_s\left(\frac{z}{A}\right) \star p_n(z),$$

where symbol '★' denotes continuous convolution. The choice of a suitable estimate for σ^2 is quite difficult; moreover, an optimal value for σ^2 does not exist since it should be changed through time according to the adaptation progress, as already outlined in [5,24,25]. Despite this, for a wide range of the noise power a suitable approximation of $g_{\mathrm{B}}(z)$ seems to be [24] the bilateral sigmoid:

$$g_{\mathrm{H}}(z) \stackrel{\mathrm{def}}{=} a\tanh(bz), \tag{6}$$

with a and b being properly fixed parameters. In [24], a pair of values for a and b is obtained by fitting the expression (6) with the actual Bellini's function for a given convolutional noise level. Anyway, it is clear that as an optimal constant value for σ^2 cannot be found, a suitable pair of *constant* parameters a and b cannot be fixed.

3. Modified 'Bussgang' algorithm

In the present section we derive a modified algorithm stemming from the definition of a proper cost function which closely resembles a robust minor component analysis (MCA) criterion.

3.1. Algorithm derivation

The main idea behind 'Bussgang' adaptive deconvolution is that on the basis of source signal estimation provided by function $g(z)$ the performance of deconvolution system can be measured by the error-quantity:

$$r(z(t)) \stackrel{\text{def}}{=} g(z(t)) - z(t); \tag{7}$$

by definition, $r(z)$ is a zero-mean random process whose variance measures the average performance of deconvolution system over the training data-set. In a neural perspective, we can imagine a neuron with activation level $z(t)$ and response level $r(t)$ that learns how to minimize its output variance. Clearly, as the function r is monotonically increasing with respect to $\mathbf{w}$ and vanishes for $z=0$, some constraint on the values of $\mathbf{w}$ must be established in order to prevent the trivial solution $\|\mathbf{w}\|=0$, i.e. to prevent the weight vector from vanishing; this is quite natural in the context of e.g. blind channel equalization [25] where the constraint on the inverse filter impulse response is known as automatic gain control (AGC) and is used to keep at a fixed value the power of filter's output signal. A way to ensure AGC is for instance to add the constraint $\mathbf{w}^{\mathrm{T}}\mathbf{w}=\kappa^2$, where $\kappa > 0$ is an arbitrarily chosen constant. Thus, the proposed unsupervised optimization principle that drives neuron's learning reads:

$$\min_{\mathbf{w}} U(\mathbf{w}) \quad \text{under } \mathbf{w}^{\mathrm{T}}\mathbf{w}=\kappa^2, \quad U(\mathbf{w}) \stackrel{\text{def}}{=} E_s[r^2(\mathbf{w}^{\mathrm{T}}\mathbf{y})]. \tag{8}$$

This closely resembles the learning criterion used in neural non-linear minor component analysis (MCA), which is the counterpart of principal component analysis (PCA), in its non-linear (or robust) extension [9,10,16,17,26,31,32,34,44,45]. In this case, the inputs to the neuron are provided by a tapped delay-line that operates as a buffer for the incoming signal $y(t)$ [42]. A buffered neuron with the activation function is depicted in Fig. 1.

The minimization of the cost function U can be performed by means of a stochastic gradient steepest descent (SGSD) algorithm described by $\Delta\mathbf{w} \propto -\partial U/\partial\mathbf{w}$. In the present context, this rule assumes the following expression:

$$\Delta\mathbf{w} = -\eta(g'(z)-1)(g(z)-z)\mathbf{x}, \tag{9}$$

where η is a positive stepsize and $g'(z)$ denotes the derivative of the function $g(z)$ with respect to z.

To define a suitable estimator, we start from expression (6): First, we proposed to adapt through time neuron's parameters a and b by means of a SGSD algorithm

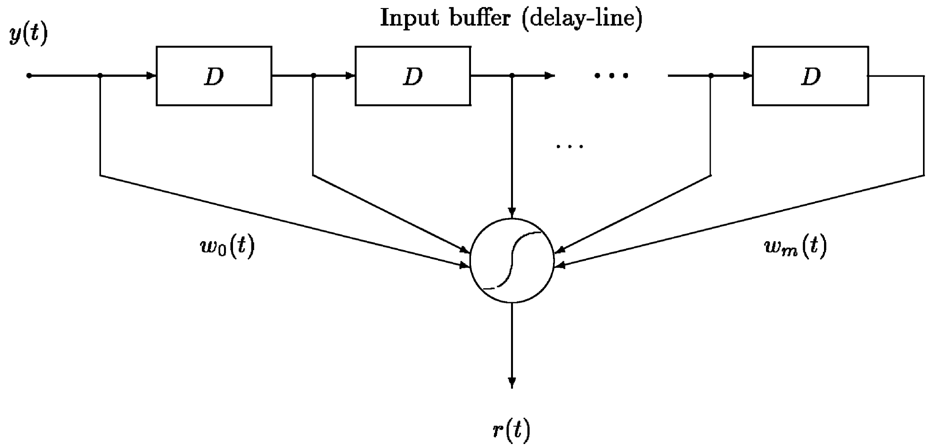

Fig. 1. Adaptive activation function neuron with input buffer. (Blocks marked with D represent unitary delays, i.e. $Dy(t)\stackrel{\text{def}}{=}y(t-1)$.)

applied to U (thought of as a function of a, b and z) [13,20]. However, a close examination of the truncated MacLaurin expansion of cost function U with the activation (6) reveals that the learning algorithm tends to force the product ab to approach 1; this conclusion, also supported by computer simulations, lead us to propose instead the following neural estimator:

$$g(z)\stackrel{\text{def}}{=}\frac{\Lambda}{\lambda}\tanh(\lambda z), \tag{10}$$

with $\Lambda > 1$ being a properly chosen constant close to 1; the learnable parameter $\lambda(t)$ adapts by

$$\Delta\lambda = -\gamma\frac{\partial U}{\partial\lambda} \approx -\gamma r(z)\left[-\frac{1}{\lambda}g(z) + (1-\lambda^2 g^2(z))\frac{z}{\lambda}\right]\Lambda, \tag{11}$$

where γ is a constant positive learning stepsize and the stochastic gradient approximation was used.

The AGC condition on $\mathbf{w}$ could be taken into account via the Lagrange multipliers method, but it is rather computationally demanding; here we prefer instead to use a simpler iterative re-normalization of the vector $\mathbf{w}$. Ultimately, the learning rule for $\mathbf{w}$ reads:

$$\Delta\tilde{\mathbf{w}} = -\eta r(z)(\Lambda - 1 - \lambda^2 g^2(z)\Lambda)\mathbf{x}, \quad \mathbf{w} = \frac{\kappa\tilde{\mathbf{w}}}{\|\tilde{\mathbf{w}}\|}. \tag{12}$$

Eqs. (12) and (11) give a new gradient-based blind deconvolution method with the adaptive neuron estimator (10). Remarkably, this learning theory shows a close relationship with the Sudjianto–Hassoun interpretation of Hebbian learning (for more details, see [15,40]). In fact, the model (3) may be thought of as the superposition of the useful signal and a disturbance, and the neuron, through the non-linear adaptive activation function, is matched to the cumulative distribution function (CDF) of the unwanted component and is thus able to filter out this component. [1]

The learning rules (12) and (11) involve two learning stepsizes (η, γ). Here we always consider constant stepsizes, i.e. no cooling schemes $\eta = \eta(t)$ and $\gamma = \gamma(t)$ are required in order for the algorithm to converge. Intuitively, and as will be proven by numerical simulations, this is very useful when non-stationary linear systems are to be deconvolved.

3.2. Analytical considerations on convergence

The name of the considered class of blind deconvolution algorithms derives from the fact that, at convergence, the inverse filter output signal is Bussgang [24], i.e. it satisfies the relationship $E_s[z(t-k)z(t)] = E_s[z(t-k)g(z(t))]\ \forall k$.

However, very little can be decided about the convergence of FIR adaptive filters [25,29], thus in order to get some results we should hypothesize that both the system's impulse response and inverse filter impulse response are IIR, i.e. they have infinitely many elements (however, they have finite norm and should thus belong to a Hilbert space [6]). Also, we are unable to prove the convergence of the algorithm with the variable estimator, thus we should further hypothesize that the value of $\lambda(t)$ either quickly stabilizes to its optimal value, so that it can be considered as a constant for t sufficiently large, or it varies very slowly, so that it is almost constant.

With these premises, we recall that the 'Bussgang' algorithm belongs to the class of Benveniste–Goursat–Ruget optimization rules, thus one of their fundamental results [6] might be used to get a *sufficient* condition for the asymptotic convergence of the modified 'Bussgang' algorithm with estimator $g(\cdot)$. It is worth noting that Benveniste, Goursat, and Ruget used very simple non-linear functions and were able to arrive at closed-form results; in the present case, unfortunately, these analytical results are not profitable because the involved quantities are rather difficult to be treated analytically.

4. Computer simulation results

In support of the new linear system deconvolution theory, as an experimental case, we present simulations performed under the following conditions:

- As vector $\mathbf{h}$, we take the sampled impulse response of a typical non-minimum phase telephonic channel with $\ell = 14$ used in [6]; the bar-graph of $\mathbf{h}$ is shown in Fig. 2 along with its (truncated) inverse obtained by the use of discrete-time Fourier transform. This channel causes both phase and amplitude distortions.

[1] Note that the $\tanh(\cdot)$ function is a good approximation of the $\mathrm{erf}(\cdot)$ function, which is the true CDF of a Gaussian random signal.

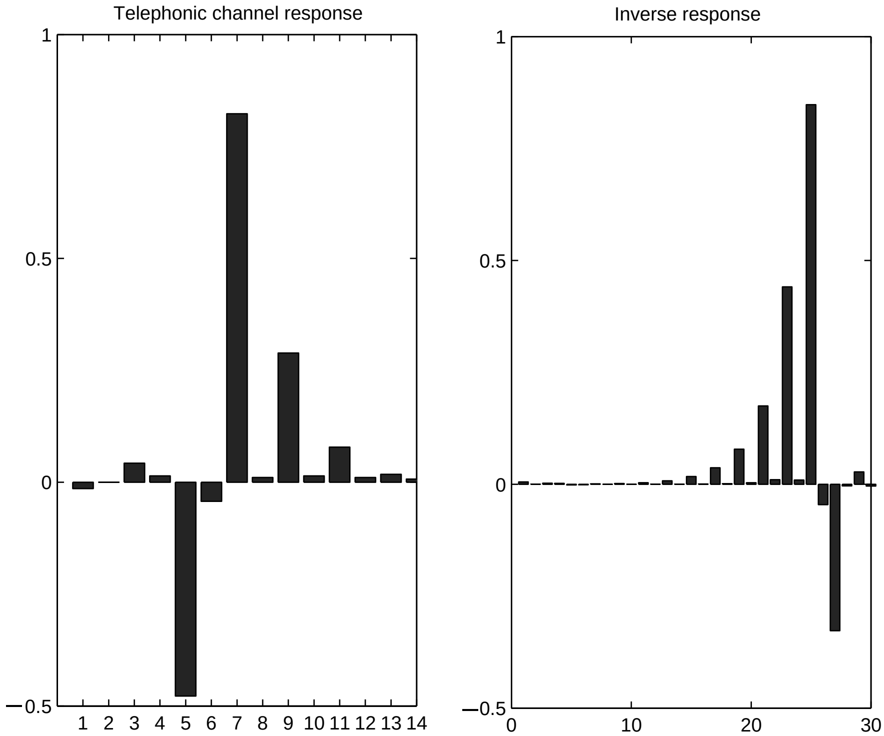

Fig. 2. Sampled telephonic channel response **h** and its approximated inverse.

- As source signal, a sub-Gaussian random process uniformly distributed within $[-\sqrt{3}, \sqrt{3}]$ has been taken, like the one described in [5]. A uniform distribution represents well an M-ary alphabet of equally probable symbols for large M; uniformly distributed alphabets are often used in telecommunications because they ensure the highest quantity of carried information.
- As deconvolving structure, a buffered neuron with $m = 20$ input-delays as recommended in [6] is used; the suitability of this neuron's buffer size is empirically confirmed by Fig. 2.
- The algorithm starts with null tap-weights except for the central one equal to 1, and $\lambda(0) = 1$; as neuron output amplitude gain we chose $\kappa = 0.5$; also we took $\Lambda = 1.05$.

In the simulations, each learning epoch counts 500 samples. The actual deconvolution accuracy degree is measured by means of the residual inter-symbol interference (ISI) defined as in [39]:

$$\mathrm{ISI} = \frac{\|\mathbf{c}\|^2 - c_{\max}^2}{c_{\max}^2}, \tag{13}$$

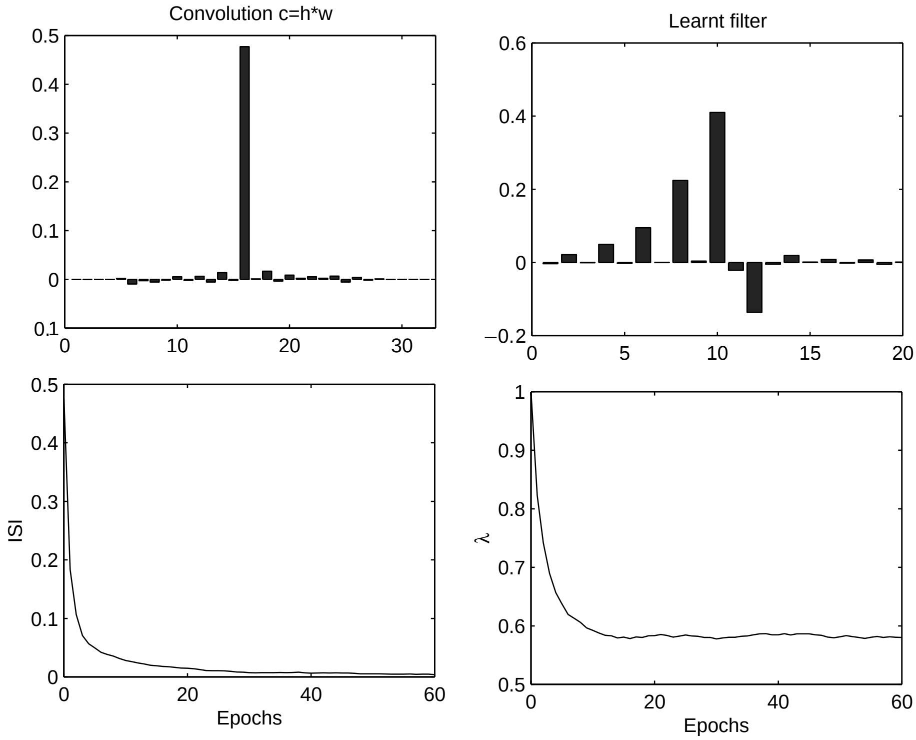

Fig. 3. Results in the noiseless case.

where **c** denotes the discrete convolution between **w** and **h**, and c_{max} is the component of **c** having the maximal absolute value. Perfect equalization would imply **c** has only a central entry different from zero, but in a real-world context some residual interference should be tolerated.

Fig. 3 shows the inverse neuron-filter response learnt after 60 epochs and the corresponding convolution **c**; it also shows the ISI computed at the end of any epoch as well as the course of parameter λ during neuron's learning. These results refer to a noiseless channel (i.e. $N(t)=0$) and have been obtained with $\gamma=\eta=0.05$. It is very interesting to note that the algorithm does not need a pre-whitening stage, that is, it exhibits self-whitening abilities [12].

Fig. 4 refers instead to a noisy channel, where $N(t)$ is a zero-mean additive white Gaussian noise (AWGN) and three signal-to-noise ratios (SNRs) have been considered. The obtained ISI in the three cases is shown as the average over 20 independent trials, in order to mitigate the weak fluctuations in the results owing to different source signal realizations. Again, we set $\gamma=\eta=0.05$ in the simulations.

In both cases, the algorithm performs well, and the second simulation shows that it is insensitive with respect to Gaussian additive noise as long as the SNR level keeps higher than 10 dB.

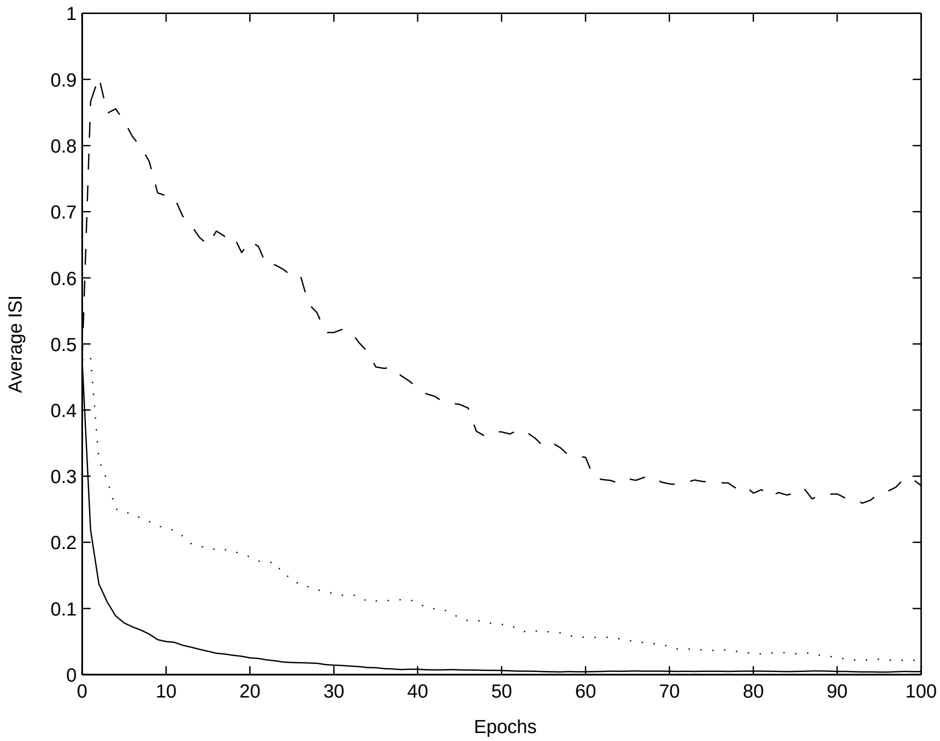

Fig. 4. Results in the noisy case (solid line: SNR = 20 dB; dotted line: SNR = 10 dB; dashed line: SNR = 1 dB).

It could be interesting to simulate the behavior of the algorithm in the deconvolution of non-stationary systems, as for instance of a switching channel, i.e. a channel whose impulse response $\mathbf{h}(t)$ suddenly switches from $\mathbf{h}_1$ to $\mathbf{h}_2$ (see e.g. [46]). To this aim, we recall the non-minimum phase impulse response introduced by Shalvi and Weinstein [39] (14 taps long) and consider a system that switches, any 100 epochs, from the telephonic channel to the Shalvi–Weinstein channel and back. Fig. 5 refers again to a noisy case, where $N(t)$ is a zero-mean AWGN and three SNRs have been considered; the obtained ISI is shown averaged over 20 independent trials for $\gamma = \eta = 0.05$. The results on the simulated operation over a non-stationary channel fully agree with the previous results. Interestingly, on Shalvi–Weinstein channel, the algorithm is more robust against AWGN, because of the fact that it is an all-pass channel, i.e. it only causes phase distortions [13].

To end with, we present a comparison among the original 'Bussgang' algorithm, the one proposed here based on a neural Bayesian approximated estimator, and the modification of 'Bussgang' learning algorithm introduced by Amari et al. [3] based on the concept of natural gradient in the systems' space [2]. The algorithms have been tested again with neuron buffer size of 20, over 60 learning epochs.

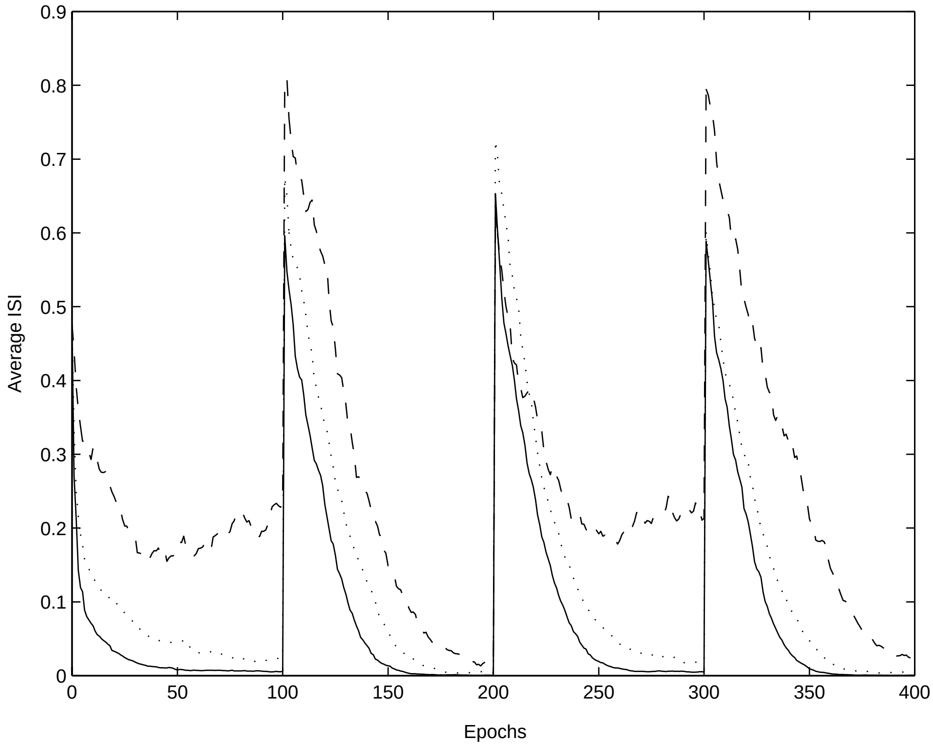

Fig. 5. Simulations of operation over a noisy/non-stationary channel (solid line: SNR = 20 dB; dotted line: SNR = 10 dB; dashed line: SNR = 1 dB. Epochs 0–99, 200–299: Telephonic channel; Epochs 100–199, 300–399: Shalvi–Weinstein channel).

Any learning algorithm requires some parameters to set: In order to make the comparison fair, we chose these values by a trial-and-error procedure, as a trade-off in order to prevent the numerical instability and to obtain the best performances. Note that this was a quite critical and difficult task, since for instance the magnitude order of the learning stepsize largely varies from algorithm to algorithm; also, the 'Bussgang' one requires the user to make some assumption not supported from any information, that is consequently rather subjective. For each algorithm, we also 'measured' two important characteristics: The average number of floating-point operations (flops) per iteration required by the algorithms to run and the elapsed time for the whole learning phase; the elapsed time is important because it takes into account not only the computation time but also the time spent to store and retrieve the data required, e.g., when updating the neuron buffers.

The ISI averaged over 20 independent realizations of the input sequence is shown in the Fig. 6, which refers to a noiseless telephonic channel. Also, Table 1 reports the flops and elapsed times for the implemented algorithms; they are referred to a 500 MHz machine with 64 MB memory. The numbers of necessary flops do not

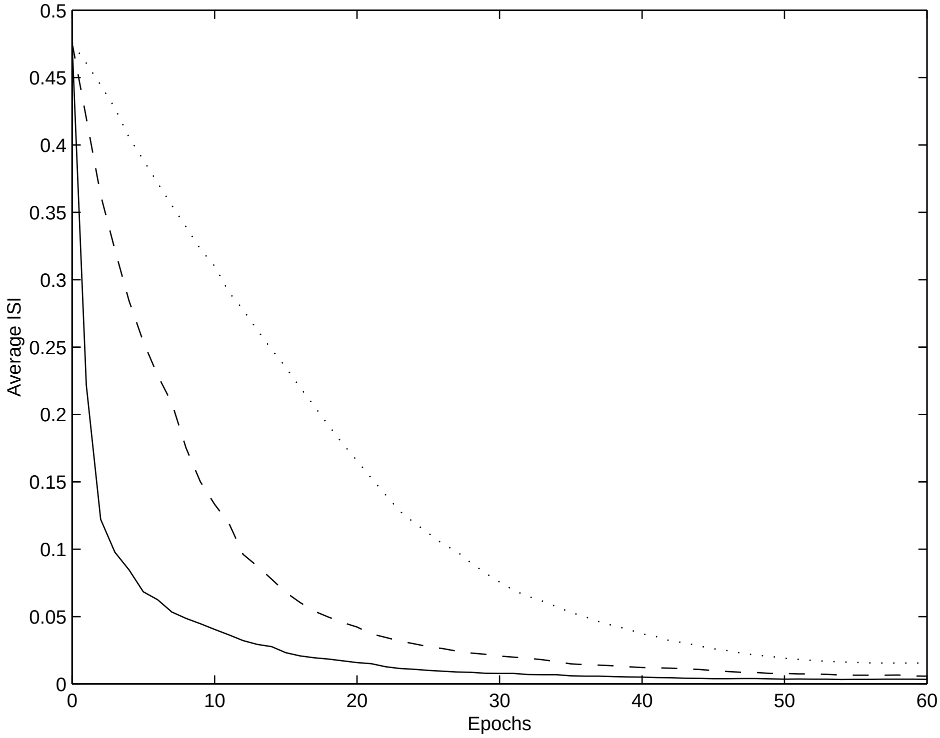

Fig. 6. Comparison of 'Bussgang' algorithm (dashed line), natural-gradient-'Bussgang' (dotted line) and the neural 'Bussgang' (solid line).

Table 1
Estimated computational complexity (flops per iteration) and computation time (averaged over 20 independent trials)

Algorithm	Flops per iteration	Elapsed time
Neural 'Bussgang'	273.3210	28.94 s
'Bussgang'	328.0032	83.16 s
'Bussgang' w. nat. grad.	370.6079	92.88 s

differ much, while the elapsed times do: This is because the Bayesian estimator is a complex function to evaluate and, in addition, the natural gradient (optimized as recommended in [3]) has two additional buffers compared to the others algorithms.

5. Conclusion and further work

The aim of this paper was to propose an improvement to the Bellini 'Bussgang' algorithm that relies on using a neural adaptive approximation of the Bayesian estimator required in the original theory. Computer simulation results show the effectiveness of the proposed approach both with noiseless and noisy systems, also in the non-stationary (switching) case.

We believe the approximation capabilities of the non-linear function $g(\cdot)$ as well as the performances of the algorithm can be enhanced by the use of more flexible functions provided by the theory of adaptive activation function neurons, already proven to be effective in independent component analysis [14]. Furthermore, extension to blind deconvolution of linear complex channels and of non-linear channels [8,23,36] are currently under investigation.

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 Comput. 10 (1998) 251–276.
[3] S. 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] A.J. Bell, T.J. Sejnowski, An information maximisation approach to blind separation and blind deconvolution, Neural Comput. 7 (6) (1995) 1129–1159.
[5] S. Bellini, Blind Equalization, Alta Frequenza, AEI (Italian Electrotechnical Association), Vol. 57, 1988, pp. 445–450.
[6] A. Benveniste, M. Goursat, G. Ruget, Robust identification of a nonminimum phase system: blind adjustment of a linear equalizer in data communication, IEEE Trans. Automat. Control AC-25 (3) (1980) 385–399.
[7] J.A. Cadzow, Blind deconvolution via cumulant extrema, IEEE Signal Processing Magazine, May 1996, pp. 24–42.
[8] S. Chen, S. McLaughlin, B. Mulgrew, Complex-valued radial basis function network, Part II: Applications to digital communications channel equalisation, Signal Process. 36 (1994) 175–188.
[9] T.-P. Chen, S.-I. Amari, N. Murata, Sequential extraction of minor components, Neural Process. Lett. 13 (3) (2001) 195–201.
[10] S. Costa, S. Fiori, Image compression using principal component neural networks, Image Vision Comput. J. (special issue on "Artificial Neural Network for Image Analysis and Computer Vision") 19 (9–10) (2001) 649–668.
[11] J.B. Destro Filho, G. Favier, J.M. Travassos Romano, New Bussgang methods for blind equalization, Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP), 1997, pp. 2269–2272.
[12] S.C. Douglas, A. Cichocki, S.-I. Amari, Self-whitening algorithms for adaptive equalization and deconvolution, IEEE Trans. Signal Process. 47 (4) (1999) 1161–1165.
[13] S. Fiori, Neural artificial unsupervised learning models for blind signal processing, M. Eng. Thesis, Dept. of Electronics and Automatics, University of Ancona, Ancona (Italy), July 1996.
[14] S. Fiori, Blind signal processing by the adaptive activation function neurons, Neural Networks 13 (6) (2000) 597–611.
[15] S. Fiori, Blind separation of circularly distributed source signals by the neural extended APEX algorithm, Neurocomputing 34 (1–4) (2000) 239–252.
[16] S. Fiori, F. Piazza, A general class of ψ-APEX PCA neural algorithms, IEEE Trans. Circuits Systems — Part I 47 (2000) 1394–1398.
[17] S. Fiori, F. Piazza, Neural MCA for robust beamforming, Proceedings of the International Symposium on Circuits and Systems (ISCAS'2000), Vol. III, May 2000, pp. 614–617.
[18] S. Fiori, G. Maiolini, Weighted least-squares blind deconvolution of non-minimum phase systems, IEE Proc. Vision Image Signal Process. 147 (6) (2000) 557–563.

[19] S. Fiori, On blind separation of complex-valued sources by extended Hebbian learning, IEEE Signal Process. Lett. 8 (8) (2001) 217–220.
[20] S. Fiori, A contribution to (neuromorphic) blind deconvolution by flexible approximated Bayesian estimation, Signal Process. 81 (10) (2001) 2131–2153.
[21] R. Godfrey, F. Rocca, Zero Memory Nonlinear Deconvolution, Geophy. Prospect. 29 (1981) 189–228.
[22] D.N. Godard, Self recovering data communication system, IEEE Trans. Comm. COM-28 (11) (1980) 1867–1875.
[23] S. He, Z. He, Blind equalization of nonlinear communication channels using recurrent wavelet neural networks, Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vol. 4, 1997, pp. 3305–3309.
[24] S. Haykin, Blind deconvolution, in: Adaptive Filtering, Prentice-Hall, Englewood Cliffs, NJ, 1991 (Chapter 20).
[25] S. Haykin (Ed.), Blind Deconvolution, Prentice-Hall, Englewood Cliffs, NJ, 1994.
[26] J. Karhunen, J. Joutsensalo, Generalizations of PCA, optimization problems, and neural networks, Neural Networks 8 (4) (1995) 549–562.
[27] D. Kundur, D. Hatzinakos, Blind image deconvolution, IEEE Signal Process. Mag. 13 (3) (1996) 43–64.
[28] M. Ibnkahla, J. Sombrin, F. Castanie, N.J. Bershad, Neural networks for modelling monlinear memoryless channels, IEEE Trans. Comm. 45 (7) (1997) 768–771.
[29] Y. Li, Z. Ding, Convergence analysis of finite length blind adaptive equalizers, IEEE Trans. Signal Process. (1995) 2120–2129.
[30] 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.
[31] E. Oja, Principal components, minor components, and linear neural networks, Neural Networks 5 (1992) 927–935.
[32] E. Oja, Nonlinear PCA criterion and maximum likelihood in independent component analysis, Proceedings of ICA'98, 1998, pp. 143–148.
[33] 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.
[34] F. Palmieri, J. Zhu, Linear neural networks which minimize the output variance, Proceedings of the International Joint Conference on Neural Networks (IJCNN), Vol. I, 1991, pp. 791–797.
[35] 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) (1997) 2731–2739.
[36] J.P. Saini, M.C. Srivastava, A new polynomial structure for channel equalization and ACI suppressioni in 64-QAM reception, Proceedings of the International Symposium on Circuits and Systems (ISCAS), Vol. IV, 1998, pp. 457–460.
[37] E.H. Satorius, J.J. Mulligan, Minimum entropy deconvolution and blind equalisation, Electron. Lett. 28 (16) (1992) 1534–1535.
[38] Y. Sato, A method for self-recovering equalization for multilevel amplitude-modulation system, IEEE Trans. Comm. COM-23 (1975) 679–682.
[39] O. Shalvi, E. Weinstein, New criteria for blind deconvolution of nonmimimum phase systems (channels), IEEE Trans. Inform. Theory 36 (2) (1990) 312–321.
[40] A. Sudjianto, M.H. Hassoun, Statistical basis of nonlinear Hebbian learning and application to clustering, Neural Networks 8 (1995) 707–715.
[41] O. Tanrikulu, B. Buyurman, A.G. Constantinides, J.A. Chambers, Soft constrained satisfaction blind channel equalization algorithms, Internat. J. Adaptive Control Signal Process. 12 (1998) 117–134.
[42] A.C. Tsoi, A.D. Back, Discrete time recurrent neural network architectures: a unifying review, Neurocomputing 15 (1997) 183–223.
[43] R.A. Wiggins, Minimum entropy deconvolution, Geoexploration 16 (1978) 21–35.

[44] L. Wiscott, Learning invariance manifolds, Proceedings of the International Conference on Artificial Neural Networks (ICANN), 1998, pp. 555–560.

[45] L. Xu, E. Oja, C.Y. Suen, Modified Hebbian learning for curve and surface fitting, Neural Networks 5 (1992) 441–457.

[46] H.H. Yang, S.-I. Amari, Blind equalization of switching channels by ICA and learning of learning rate, Proceedings of the International Conference on Acoustics, Speech and Signal Processings (ICASSP), Vol. III, 1997, pp. 1849–1852.

Dr. Simone Fiori was born in Rimini (Italy) in 1971. He received the Italian Laurea (Dr.Ing.) cum laude in electronics engineering in July 1996 from the University of Ancona (Italy), and the Ph.D. degree in electrical engineering (circuit theory) in March 2000 from the University of Bologna (Italy). In 1998 he joined the Neural Networks and Adaptive Systems Research Group at the Department of Industrial Engineering of the Perugia University as two-year visiting researcher. In November 2000 he has been appointed Assistant Professor at the Department of Industrial Engineering of the Perugia University. His research interests include unsupervised learning theory for artificial neural networks (ANNs), linear and non-linear adaptive discrete-time filtering, vision and image processing by neural networks, continuous-time and discrete-time circuits for stochastic information processing. He is author of more than 50 refereed journal and conference papers on these topics and is serving as reviewer for the major journals in these areas as the IEEE Transactions on Neural Networks, the IEEE Transactions on Signal Processing, Neurocomputing, and the IEEE Transactions on Antennas and Propagation.