BLIND DECONVOLUTION, INFORMATION MAXIMIZATION AND RECURSIVE FILTERS

Kari Torkkola
Motorola, Phoenix Corporate Research Laboratories
2100 East Elliot Rd, MD EL508, Tempe, AZ 85284, USA
tel: (602)413-4129, fax: (602)413-7281, email: a540aa@email.mot.com

ABSTRACT

Starting from maximizing information flow through a nonlinear neuron Bell and Sejnowski derived adaptation equations for blind deconvolution using an FIR filter [1]. In this paper we will derive a simpler form of the adaptation and we will apply it to more complex filter structures, such as recursive filters. As an application, we study blind echo cancellation for speech signals. We will also present a method that avoids whitening the signals in the procedure.

1. BLIND DECONVOLUTION

Assume an unknown signal s convolved with an unknown filter with impulse response a (which can be any kind of a filter, for example, a causal FIR filter $a_k, k = 0, ..., L_a$). The resulting corrupted signal x is a convolution $x = a * s$. The task is to recover s by learning a filter w which reverses the effect of filter a so that $u = w * x$ would be equal to the original signal s upto a delay and a constant.

The corrupting filter spreads information from one sample s_t to all the samples $x_t, ..., x_{t+L_a}$. The task of blind deconvolution is now to remove these redundancies assuming that the samples of the original signal s_t are statistically independent. Some practical applications include blind acoustic echo cancellation, (where only the echo-corrupted signal is available) and suppression of intersymbol interference in communications (blind equalization) [3].

Several methods for blind deconvolution are based on the fact that if a source signal having a non-Gaussian PDF (probability density function) is convolved with a filter, the PDF of the resulting signal is closer to a Gaussian PDF due to the central limit theorem. Deconvolution can then be achieved by finding a filter which drives the output PDF away from a Gaussian. Functions of higher-order statistics, for example, kurtosis, can be used as a cost function to minimize/maximize [6, 5, 2, 3, 4].

Bell and Sejnowski formulated blind deconvolution as redundancy reduction between samples of data [1]. We will first review their information maximization approach. By viewing their approach rather as shaping of the output PDF, we will show that the same learning rule can be achieved via a slightly simpler path. We will show how this facilitates learning more complicated filter structures for blind deconvolution. Finally, some experiments with blind acoustic echo cancellation will be presented.

2. INFORMATION MAXIMIZATION

Bell and Sejnowski proposed to learn the restoring filter w by using an information theoretic measure [1]. In their configuration, w is a causal FIR filter[1].

$$u_t = \sum_{k=0}^{L} w_k x_{t-k} \quad (1)$$

The output of the filter is passed through a nonlinear squashing function, for example, $y_t = g(u_t) = tanh(u_t)$. By maximizing the information transferred through this system (or, entropy of the output) a filter is learned that removes the redundancies.

The approach in [1] was to chop the signal x into blocks of length M, represented as vectors $X = [x_{t-(M-1)}, ..., x_t]^T$. The filtering is formulated as a multiplication of a block by a lower triangular matrix with coefficients of w, followed by the nonlinear function g.

$$Y = g(U) = g(WX),$$

$$W = \begin{bmatrix} w_0 & 0 & \dots & 0 & 0 \\ w_1 & w_0 & 0 & \dots & 0 \\ \vdots & & & & \vdots \\ w_L & \dots & w_0 & 0 & \\ \vdots & & & & \vdots \\ 0 & \dots & w_L & \dots & w_0 \end{bmatrix}$$

When the information at transformed output block Y is maximized, redundancies caused by a, the distorting filter, are removed *within the block*. Bell and Sejnowski showed that information maximization is equal to maximizing the entropy at the output, which can be written as the expectation of the log probability density function of the output. Since $f_Y(Y) = f_X(X)/|J|$, where J is the Jacobian of the whole system, we get

$$\begin{aligned} H(Y) &= -E[ln(f_Y(Y))] \\ &= -E[ln(f_X(X)/|J|)] \\ &= E[ln|J|] - E[ln(f_X(X))]. \end{aligned} \quad (2)$$

Maximizing $H(Y)$ equals now maximizing $E[ln|J|]$, since $f_X(X)$ does not depend on W. The Jacobian J tells how the input affects the output and is written as the following matrix of partial derivatives of each component of the output vector with respect to each component of the input vector, that is, $J = [\partial y_i / \partial x_j]_{ij}$.

We need to compute its determinant, which can be decomposed into the determinant of the weight matrix and the product of the slopes of the nonlinear function g. Since

[1] Subscripts refer to time, or, with filter coefficients to the delay from the current sample. t refers to the present time. The filter coefficient with zero delay from current sample x_t is denoted by w_0 whereas in [1] w_L was used.

W is a lower diagonal matrix its determinant is simply the product of its diagonal values:

$$detJ = |J| = (detW) \prod_{k=0}^{M-1} \hat{y}_{t-k} \quad and$$

$$ln|J| = ln(w_0^M) + \sum_{k=0}^{M-1} ln(\hat{y}_{t-k})$$

where for the $tanh$ function $\hat{y}_t = \partial y_t / \partial u_t = 1 - y_t^2$.

The quantity to maximize is now $E[ln|J|]$. By computing the gradient of $ln|J|$ with respect to each weight w_j, Bell and Sejnowski derived a stochastic gradient ascent rule to update the weights. For the zero delay weight:

$$\begin{aligned} \Delta w_0 &\propto \frac{\partial(ln|J|)}{\partial w_0} \\ &= M\frac{1}{w_0} + \sum_{k=0}^{M-1} \frac{1}{\hat{y}_{t-k}} \frac{\partial \hat{y}_{t-k}}{\partial y_{t-k}} \frac{\partial y_{t-k}}{\partial u_{t-k}} \frac{\partial u_{t-k}}{\partial w_0} \\ &= \sum_{k=0}^{M-1} (\frac{1}{w_0} - 2y_{t-k}x_{t-k}) \end{aligned} \tag{3}$$

In a similar fashion the update rule for all the other weights can be derived:

$$\begin{aligned} \Delta w_j &\propto \frac{\partial(ln|J|)}{\partial w_j} \\ &= \sum_{k=0}^{M-1-j} (-2y_{t-k}x_{t-k-j}) \end{aligned} \tag{4}$$

3. SIMPLER DERIVATION

However, it is possible to arrive almost to the same rule via a simpler route. This approach also allows simple derivation of the learning rules for other types of filters, for example, for recursive filters. Instead of looking at a block of output samples, let us look at the output a single sample at the time:

$$u_t = \sum_{k=0}^{L} w_k x_{t-k}; \quad y_t = g(u_t). \tag{5}$$

Since entropy of y, $H(y) = -E[ln(f_y(y))]$, is an expectation, the whole signal y is already taken into consideration. Nothing is gained by maximizing an expectation over *blocks* of y compared to maximizing an expectation over *single samples.*

An intuitive rationale behind the approach is roughly as follows. $g(u)$ is chosen to be close to the true cumulative density function (CDF) of the data[2]. Thus, the derivative of $g(u)$ is close to the probability density function (PDF) of the data. On the other hand, the PDF of convolved data approximates a Gaussian PDF due to the central limit theorem. Now, when data is passed through a function that approximates its CDF, the density of the output is close to uniform density, which is the PDF that has the largest entropy of all PDFs. The deconvolving filter w can be learned by passing the deconvolved signal u through g, and by finding the w which produces the true density of the data, which in turn will be observed as uniform density at the output of g. This is equal to maximizing the entropy of the output.

In this single sample case the Jacobian of (5) is a scalar

$$J = y_t' = \frac{\partial y_t}{\partial x_t} = \frac{\partial y_t}{\partial u_t}\frac{\partial u_t}{\partial x_t} = \hat{y}_t w_0 = (1 - y_t^2) w_0 \tag{6}$$

As in the derivation of Bell and Sejnowski, we can arrive at a stochastic gradient ascent rule by taking the gradient of $ln(J)$ with respect to the weights. Let us first compute

$$\frac{\partial y_t'}{\partial w_0} = \hat{y}_t + w_0 \frac{\partial \hat{y}_t}{\partial w_0} = \hat{y}_t - 2w_0 y_t \hat{y}_t x_t$$

The adaptation rule for w_0 is now readily obtained:

$$\Delta w_0 \propto \frac{\partial ln(y_t')}{\partial w_0} = \frac{1}{y_t'}\frac{\partial y_t'}{\partial w_0} = \frac{1}{w_0} - 2y_t x_t \tag{7}$$

By first computing

$$\frac{\partial y_t'}{\partial w_j} = -2w_0 y_t \hat{y}_t x_{t-j}, \tag{8}$$

we can derive the following rule for the other weights:

$$\Delta w_j \propto \frac{1}{y_t'}\frac{\partial y_t'}{\partial w_j} = -2y_t x_{t-j} \tag{9}$$

What is the difference between the adaptation rules of Bell and Sejnowski, (3) and (4), and the rules (7) and (9)? In practice there is not much difference. (3) and (4) accumulate the weight changes in a block of M samples before doing the adjustment. Our rule is a true stochastic gradient ascent rule for each sample separately. In practice, with this kind of adaptation rules it is good to accumulate the weight changes from a number of training samples before making the change to the actual weights. How many samples to use can be determined by experimentation.

In addition, (4) has an adverse border effect if M is not much larger than L. Fewer samples of data (only M-L samples) contribute to weights at the end of filter w compared to weights in the beginning of the filter (M samples). Thus looking at the data one sample at the time results in a more accurate adaptation rule. However, the biggest advantage is that it allows simple derivation of the adaptation for more complex filter structures. We will look at recursive filters in the next section.

4. RECURSIVE FILTERS

We will now look at a recursive filter (IIR) in the direct form and derive the adaptation equations in the similar fashion as above. The filter output before the nonlinearity is

$$u_t = w_0 x_t + \sum_{k=1}^{L} w_k u_{t-k} \tag{10}$$

The quantity to maximize remains the same, $E[ln(J)]$. The Jacobian of the filter is now exactly the same as in equation (6). Also $\partial y_t'/\partial w_0$ and the adaptation rule for w_0 turn out to be the same as for an FIR-filter, which should be no surprise since the filters are equal as far as w_0 is concerned. To derive the adaptation for other weights w_j, we will first write

$$\frac{\partial y_t'}{\partial w_j} = \frac{\partial(1 - y_t^2) w_0}{\partial w_j} = w_0(-2y_t)\hat{y}_t \frac{\partial u_t}{\partial w_j}. \tag{11}$$

[2] $tanh$ is a reasonable approximation of the cdf for positively kurtotic signals (super-Gaussian), like speech.

A difficulty is caused by $\partial u_t/\partial w_j$ which is a recursive quantity. Taking the derivative of (10) with respect to w_j gives:

$$\begin{aligned}\frac{\partial u_t}{\partial w_j} &= u_{t-j} + w_j \frac{\partial u_{t-j}}{\partial w_j} \\ &= u_{t-j} + w_j(u_{t-2j} + w_j \frac{\partial u_{t-2j}}{\partial w_j}) \\ &= \sum_{k=1}^{t/j} (w_j)^{k-1} u_{t-k\cdot j}. \end{aligned} \quad (12)$$

We will first define the following recursive quantity in a fashion similar to deriving LMS algorithm for adaptive recursive filters [7]:

$$\alpha_j^t \equiv \frac{\partial u_t}{\partial w_j} = u_{t-j} + w_j \alpha_j^{t-j} \quad (13)$$

Now we can readily obtain the rule for w_j

$$\Delta w_j \propto \frac{1}{y_t'} \frac{\partial y_t'}{\partial w_j} = -2 y_t \alpha_j^t \quad (14)$$

However, it will be necessary to keep track of α_j^t for each filter coefficient w_j.

We will now show that an approximation of this rule leads to the same convergence condition (see [1] for an interpretation of the convergence condition as an independence test). Convergence of the adaptation rule (14) is achieved when the weight change becomes zero, that is when

$$E[\Delta w_j] = E[-2 y_t \alpha_j^t] = E[y_t \sum_{k=1}^{t/j} (w_j)^{k-1} u_{t-k\cdot j}] = 0$$

$$\Leftrightarrow \quad \sum_{k=1}^{t/j} (w_j)^{k-1} E[y_t u_{t-k\cdot j}] = 0$$

holds for all j. This is true if $E[y_t u_{t-j}] = 0$ for all j, a more restrictive condition, which is the convergence condition of the adaptation rule obtained from (14) by replacing α_j^t by u_{t-j} yielding

$$\Delta w_j \propto -2 y_t u_{t-j} \quad (15)$$

This is our final adaptation rule for the coefficients of the recursive filter. Comparing (15) to (13) shows that in effect we have dropped the second term from the right side of (13). We will show experimentally in Sec. 5. that there is no difference between learning rules (14) and (15).

For an effective implementation it is necessary to use the training data sequentially, because the previous values of u_{t-j} must be stored in a buffer. In contrast, the FIR filter can be trained by picking the training points randomly in the signal.

The same approach can be applied to filters of any form, for example, to a filter that is a cascade of second order sections, to a lattice filter, to a nonlinear filter, ect.

5. ECHO CANCELLATION EXPERIMENTS

We will now present some examples of blind echo cancellation using speech signals and artificial echoes. In all experiments we used the same recording of 7 seconds of speech as the training material. The gradient was accumulated from 100 speech samples before updating the weights, and 10000 - 40000 gradient updates were performed.

Short-time prewhitening. Note that speech signals violate the assumption of samples being independent. The speech signal contains *other dependencies* besides the possible echos. Consecutive samples of a speech signal are *very* dependent of each other, and the strongest of these dependencies have a scope of about 2 milliseconds, corresponding to 16 samples at the sampling frequency of 8 kHz. Applying blind deconvolution to a speech signal results in a filter that produces whitened output, i.e., all time-dependencies will be removed. This is not a desirable side effect in speech signal processing. However, this effect can be avoided using the following scheme: The short-time dependencies in the speech signals will be first removed by a whitening filter with a short time span (for example, 20-100 samples at 8kHz sampling frequency). Figure 1 depicts such a whitener that has 60 taps, also learned using blind deconvolution.

This whitener only removes the inherent dependencies in the speech signal (on the average) leaving echoes with longer delays intact. Now, blind deconvolution can be applied to learn the echo removal filter from the whitened signal. Finally, the learned filter will be applied to the original speech signal, which contains both the inherent short-time dependencies, and the echo-related dependencies with longer delays. The effect is to remove only the echoes leaving the speech signal otherwise intact. Note that this only works if the unwanted dependencies have longer delays than the desired ones. This scheme was used in all following experiments.

Figure 1. Coefficients of a whitener of 60 taps.

Single echo. For this experiment, a single echo with amplitude 0.5 was added to a speech signal at a delay of 500 samples, corresponding to a delay of 1/16 seconds at 8kHz sampling frequency. The length of the prewhitener was 100 taps in this and in the following experiments. First, an FIR blind echo removal filter of 2002 taps was trained using (7) and (9). The coefficients of the resulting filter are depicted in Fig. 2 (The zeroth coefficient will always be equal to one, but it will be cut out of this and the following figures due to space limitations).

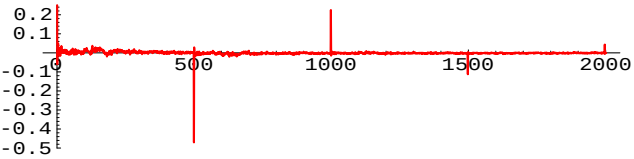

Figure 2. Coefficients of a blind single echo cancelling FIR filter.

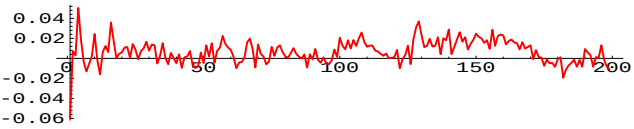

Figure 3. Coefficients 1-200 of a blind single echo cancelling FIR filter.

This filter should have a negative peak at the delay of 500 samples, with amplitude corresponding to the echo amplitude. As the ideal FIR has an infinite length for this task, there should also be exponentially decaying peaks at integer multiplies of these delay values. This seems to be the case.

The audible quality of the deconvolved signal was good, the echo was removed with no other effects. To give an idea of how accurate the filter coefficients are, coefficients of taps 1-200 are depicted in Fig. 3 with more resolution. Ideally, all these should be zeroes. As a messure of goodness we computed the following:

$$P_{diff} = \frac{power(IR_{ideal} - IR_{learned})}{power(IR_{ideal})}. \qquad (16)$$

For this FIR filter P_{diff} equals -12.4dB. This is caused by the noise due to about 2000 nonzero coefficients that are supposed to be zero ideally.

Next, we trained a recursive filter of 502 taps for the same task using (7) and (15). Since the filter is in the direct form, only one nonzero coefficient at the delay of the echo is sufficient for the task. The resulting filter coefficients are shown in Fig. 4, and they appear to be in order.

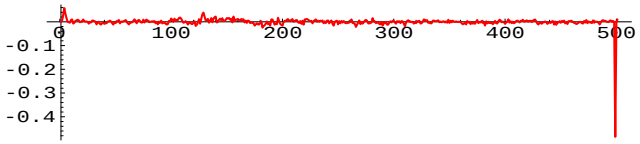

Figure 4. Coefficients of a blind single echo cancelling IIR filter.

The impulse response of the recursive filter (Fig. 5) is almost identical to the impulse response of the FIR filter which had four times the number of taps and thus also four times the computational complexity. The audible quality of the result processed with the IIR deconvolver was similar to the FIR deconvolver. For this IIR filter P_{diff} amounts to -10.2dB, which is visible comparing Figures 2 and 5.

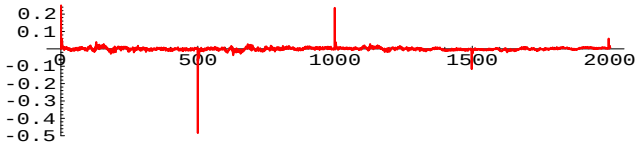

Figure 5. Impulse response (2002 samples) of a blind single echo cancelling IIR filter.

Double echo. Now, two artificial echoes were added to a speech signal, at delays of 200 and 500 samples, both with amplitude 0.5. Filters similar to previous experiment are depicted in Figs 6, 7, and 8 for this task.

Full adaptation. Finally, we applied the full recursive adaptation of (13) and (14) to IIR filters in the single echo case. Resulting filter coefficients and the impulse response are depicted in Figs. 9 and 10. Comparing these to the filter trained with the approximative adaptation (Figs. 4 and 5) reveals that there are no differences.

6. CONCLUSION

We have shown that the information maximization principle for blind deconvolution can be extended to more complex filter structures than FIR filters. As an example, we derived the adaptation equations for a recursive filter (IIR filter) in direct form. An advantage in using recursive filters is that they are able to model complicated and long impulse responses with a small number of coefficients, and with a small computational complexity. A limitation with recursive filters is that if the inverse of the convolving filter a is unstable, the deconvolving w will be unstable and cannot be learned using this procedure.

To illustrate the adaptation of the filters, we presented speech signal echo cancellation examples together with a

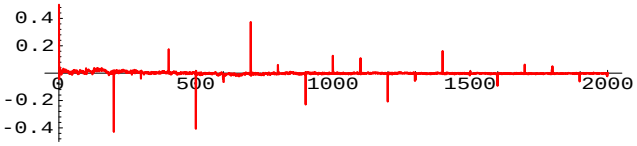

Figure 6. Coefficients of a blind double echo cancelling FIR filter.

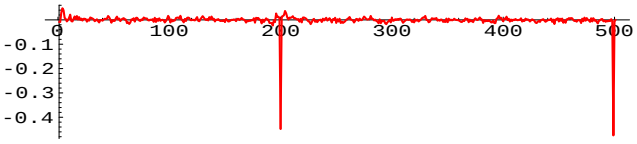

Figure 7. Coefficients of a blind double echo cancelling IIR filter.

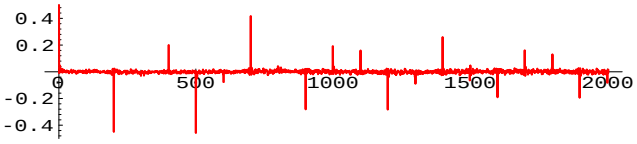

Figure 8. Impulse response (2002 samples) of a blind double echo cancelling IIR filter.

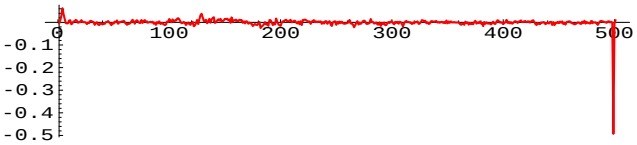

Figure 9. Coefficients of a blind single echo cancelling IIR filter trained using full recursive adaptation.

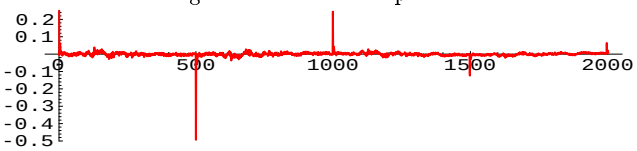

Figure 10. Impulse response (2002 samples) of a blind single echo cancelling IIR filter trained using full recursive adaptation.

method that avoids whitening the signals, which otherwise would be an undesirable side effect of blind deconvolution.

Future work includes analysis of the convergence of the adaptation, adding filter stability conditions directly to the adaptation, and analysis of the misadjustment in the adaptation, i.e., how close the solution is to the ideal solution. This will definitely turn out to be an issue with long filters as the sum of small misadjustments through a long filter amounts to a significant proportion of noise in the result.

REFERENCES

[1] A. J. Bell and T. J. Sejnowski. An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1004–1034, 1995.

[2] J. A. Cadzow. Blind deconvolution via cumulant extrema. *IEEE Signal Processing Magazine*, 13(3):24–42, May 1993.

[3] S. Haykin, editor. *Blind Deconvolution*. Prentice-Hall, 1994.

[4] R. H. Lambert. *Multichannel blind deconvolution: FIR matrix algebra and separation of multipath mixtures*. PhD thesis, University of Southern California, May 1996.

[5] E. H. Satorius and J. J. Mulligan. Minimum entropy deconvolution and blind equalization. *Electronics Letters*, 28(16):1534–1535, 1992.

[6] O. Shalvi and E. Weinstein. New criteria for blind deconvolution of nonminimum phase systems (channels). *IEEE Trans. on Information Theory*, 36(2):312–321, 1990.

[7] B. Widrow and S. Stearns. *Adaptive Signal Processing*. Prentice-Hall, 1985.