A ROBUST FAST RECURSIVE LEAST SQUARES ADAPTIVE ALGORITHM

Jacob Benesty and Tomas Gänsler

Bell Labs, Lucent Technologies
700 Mountain Avenue
Murray Hill, NJ 07974, USA
email: { jbenesty, gaensler }@bell-labs.com

ABSTRACT

Very often, in the context of system identification, the error signal which is by definition the difference between the system and model filter outputs is assumed to be zero-mean, white, and Gaussian. In this case, the least squares estimator is equivalent to the maximum likelihood estimator and hence, it is asymptotically efficient. While this supposition is very convenient and extremely useful in practice, adaptive algorithms optimized on this, may be very sensitive to minor deviations from the assumptions. We propose here to model this error with a robust distribution and deduce from it a robust fast recursive least squares adaptive algorithm (least squares is a misnomer here but convenient to use). We then show how to successfully apply this new algorithm to the problem of network echo cancellation combined with a double-talk detector.

1. INTRODUCTION

Very often, in the context of system identification, the error signal (e) which is by definition the difference between the system and model filter outputs is assumed to be zero-mean, white, and Gaussian. In this case, the least squares estimator is equivalent to the maximum likelihood estimator and hence, it is asymptotically efficient. While this supposition is very convenient and extremely useful in practice, adaptive algorithms optimized on this, may be very sensitive to minor deviations from the assumptions.

One good example of system identification with the above assumptions is network echo cancellation (EC) combined with a double-talk detector (DTD). Sometimes, the DTD fails to detect the beginning or ending of a double-talk mode and as a result, a burst of speech at the output of the echo path disturbs the estimation process. The occurrence rate of these bursts depends on the efficiency of the DTD and the intensity of double-talk modes. A desirable property of an adaptive algorithm is fast tracking. A high false alarm rate of the DTD reduces the amount of information that enters the algorithm, and that reduces the tracking rate. The false alarms should therefore be minimized so that valuable data are not discarded. Fewer false alarms however, result in more detection misses and degradation of the transfer function estimate. To maintain tracking ability and high quality of the estimate, robustness against detection errors must be incorporated in the estimation algorithm itself. By *robustness* we mean insensitivity to small deviations of the the real distribution from the assumed model distribution [1].

Thus, the performance of an algorithm optimized for Gaussian noise could be very poor because of the unexpected number of large noise values that are not modeled by the Gaussian law. In our EC example, the probability density function (PDF) of the noise should be a long-tailed PDF, in order to take the bursts (due the DTD failure) into account in our model [1], [2]. Therefore, we are interested in distributional robustness since the shape of the true underlying distribution deviates slightly from the assumed model (usually the Gaussian law).

As explained in [1], a robust procedure should achieve the following:

- It should have a reasonably good efficiency at the assumed model.
- It should be robust in the sense that small deviations from the model assumptions should impair the performance only slightly.
- Somewhat, larger deviations from the model should not cause a catastrophe.

In this study, we propose to use the following PDF:

$$\begin{aligned} p(z) &= \frac{1}{2}\exp\{-\ln[\cosh(\pi z/2)]\} \\ &= \frac{1}{2\cosh(\pi z/2)}, \end{aligned} \tag{1}$$

where the mean and the variance are respectively equal to 0 and 1. It will be compared to the Gaussian density:

$$p_{\mathrm{G}}(z) = \frac{1}{\sqrt{2\pi}}\exp\left\{-z^2/2\right\}. \tag{2}$$

The PDF $p(z)$ has a heavier tail than $p_{\mathrm{G}}(z)$. If we take the derivative of $\ln[p(z)]$ and $\ln[p_{\mathrm{G}}(z)]$, it can easily be checked that the first one is bounded while the second is not; and, as it turns out, makes all the difference between a robust approach and a non-robust one. Moreover, $p(z)$ has a high kurtosis and it is well-known that PDFs with large kurtosis are good models for speech signals, which is desired in our example of EC.

In the following, we show how to derive a robust fast recursive least squares adaptive algorithm[1] from (1) and how to apply it successfully to the problem of network echo cancellation.

[1] Least squares is a misnomer for this adaptive algorithm because we do not minimize the least squares criterion but rather maximize a likelihood function. However, it is convenient to use here.

2. A ROBUST FAST RECURSIVE LEAST SQUARES ADAPTIVE ALGORITHM

In the context of system identification, the error signal at time n between the system and model filter outputs is given by

$$e(n) = y(n) - \hat{y}(n), \tag{3}$$

where

$$\hat{y}(n) = \hat{\mathbf{h}}^T \mathbf{x}(n) \tag{4}$$

is an estimate of the output signal $y(n)$,

$$\hat{\mathbf{h}} = \begin{bmatrix} \hat{h}_0 & \hat{h}_1 & \cdots & \hat{h}_{L-1} \end{bmatrix}^T$$

is the model filter, and

$$\mathbf{x}(n) = \begin{bmatrix} x(n) & x(n-1) & \cdots & x(n-L+1) \end{bmatrix}^T$$

is a vector containing the last L samples of the input signal x. Superscript ^{T} denotes transpose of a vector or a matrix.

Consider the following function:

$$J\left(\hat{\mathbf{h}}\right) = \rho\left[\frac{e(n)}{s(n)}\right], \tag{5}$$

where

$$\begin{aligned} \rho(z) &= \ln[\cosh(z)] \\ &= -\ln[2p(2z/\pi)] \end{aligned} \tag{6}$$

is a convex function and $s(n)$ is a positive scale factor more thoroughly described below. The gradient of $J\left(\hat{\mathbf{h}}\right)$ is:

$$\nabla J\left(\hat{\mathbf{h}}\right) = -\frac{\mathbf{x}(n)}{s(n)}\psi\left[\frac{e(n)}{s(n)}\right], \tag{7}$$

where

$$\begin{aligned} \psi(z) &= \frac{d\rho(z)}{dz} \\ &= \tanh(z). \end{aligned} \tag{8}$$

The second derivative of $J\left(\hat{\mathbf{h}}\right)$ is:

$$\nabla^2 J\left(\hat{\mathbf{h}}\right) = \frac{\mathbf{x}(n)\mathbf{x}^T(n)}{s^2(n)}\psi'\left[\frac{e(n)}{s(n)}\right], \tag{9}$$

where

$$\psi'(z) = \frac{1}{\cosh^2(z)} > 0,\ \forall z. \tag{10}$$

Robust Newton-type algorithms have the following form (see [3] for the Newton algorithm):

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) - \mathbf{R}_{\psi'}^{-1}\nabla J\left[\hat{\mathbf{h}}(n-1)\right], \tag{11}$$

where $\mathbf{R}_{\psi'}$ is an approximation of $E\left\{\nabla^2 J\left[\hat{\mathbf{h}}(n-1)\right]\right\}$ and $E\{\cdot\}$ denotes mathematical expectation. In this study we choose $\mathbf{R}_{\psi'} = \frac{\psi'\left[\frac{e(n)}{s(n)}\right]}{s^2(n)}\mathbf{R}$ with $\mathbf{R} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}$. This choice will allow us to derive a fast version of the algorithm. In practice, $\mathbf{R}$ is not known so we have to estimate it recursively:

$$\begin{aligned} \mathbf{R}(n) &= \sum_{i=1}^{n} \lambda^{n-i}\mathbf{x}(i)\mathbf{x}^T(i) \\ &= \lambda\mathbf{R}(n-1) + \mathbf{x}(n)\mathbf{x}^T(n), \end{aligned} \tag{12}$$

where λ $(0 < \lambda \leq 1)$ is an exponential forgetting factor.

We deduce a robust recursive least squares (RLS) adaptive algorithm:

$$e(n) = y(n) - \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n), \tag{13}$$

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \frac{s(n)}{\psi'\left[\frac{e(n)}{s(n)}\right]}\mathbf{k}(n)\psi\left[\frac{e(n)}{s(n)}\right], \tag{14}$$

where $\mathbf{k}(n) = \mathbf{R}^{-1}(n)\mathbf{x}(n)$ is the Kalman gain [4]. It can be checked that $0 < \psi'(z) \leq 1$, $\forall z$, and $\psi'(z)$ can become very small. In order to avoid divergence of the robust RLS, we do not allow $\psi'(z)$ to be lower than 0.5. So in practice, we compute $\psi'(z)$ according to (10) but we limit it to 0.5 if it is lower than 0.5. From (14) it can be understood that large errors will be limited by the function $\psi(\cdot)$. Note that if we choose $\rho(z) = z^2$, then $\psi(z) = 2z$ and $\psi'(z) = 2$, and the algorithm is exactly the non-robust RLS [4].

A robust fast RLS (FRLS) can be derived by using the *a priori* Kalman gain $\mathbf{k}'(n) = \mathbf{R}^{-1}(n-1)\mathbf{x}(n)$. This *a priori* Kalman gain can be computed recursively with $5L$ multiplications and the error as well as the adaptation parts in $2L$ multiplications. "Stabilized" versions of FRLS (with L more multiplications) exist in the literature but they are not much more stable than their non-stabilized counterparts with non-stationary signals like speech. Our approach to fix this problem is simply to re-initialize the predictor-based variables when instability is detected with the use of the maximum likelihood variable which is an inherent variable of the FRLS. This method works very well in all of the simulations that have been done. In Table 1, we give a robust FRLS algorithm with a complexity of $O(7L)$.

One other important part of the algorithm is the estimate of the scale factor s. Traditionally, the scale is used to make a robust algorithm invariant to the noise level. It should reflect the minimum mean-square error, be robust to shorter burst disturbances (double-talk in our application), and track longer changes of the residual error (echo path changes). We have chosen the scale factor estimate as

$$\begin{aligned} s(n+1) &= \lambda_{\mathrm{s}}s(n) + (1-\lambda_{\mathrm{s}})\frac{s(n)}{\psi'\left[\frac{e(n)}{s(n)}\right]}\left|\psi\left[\frac{e(n)}{s(n)}\right]\right|, \\ s(0) &= \sigma_x, \end{aligned} \tag{15}$$

which is very simple to implement. The choice of this method of estimating s is justified in [5]. With this choice, the current estimate of s is governed by the level of the error signal in the immediate past over a time interval roughly equal to $1/(1-\lambda_{\mathrm{s}})$. When the algorithm has not yet converged, s is large. Hence the limiter is in its linear portion and therefore the robust algorithm behaves roughly like the conventional RLS algorithm. When double-talk occurs, the error is determined by the limiter and by the scale of the error signal during the recent past of the error signal *before* the

Table 1 A robust FRLS algorithm.

Prediction:

$$
\begin{aligned}
e_{\mathrm{a}}(n) &= x(n) - \mathbf{a}^T(n-1)\mathbf{x}(n-1) \\
\varphi_1(n) &= \varphi(n-1) + e_{\mathrm{a}}^2(n)/E_{\mathrm{a}}(n-1) \\
\begin{bmatrix} \mathbf{t}(n) \\ m(n) \end{bmatrix} &= \begin{bmatrix} 0 \\ \mathbf{k}'(n-1) \end{bmatrix} \\
&+ \begin{bmatrix} 1 \\ -\mathbf{a}(n-1) \end{bmatrix} e_{\mathrm{a}}(n)/E_{\mathrm{a}}(n-1) \\
E_{\mathrm{a}}(n) &= \lambda[E_{\mathrm{a}}(n-1) + e_{\mathrm{a}}^2(n)/\varphi(n-1)] \\
\mathbf{a}(n) &= \mathbf{a}(n-1) + \mathbf{k}'(n-1)e_{\mathrm{a}}(n)/\varphi(n-1) \\
e_{\mathrm{b}}(n) &= E_{\mathrm{b}}(n-1)m(n) \\
\mathbf{k}'(n) &= \mathbf{t}(n) + \mathbf{b}(n-1)m(n) \\
\varphi(n) &= \varphi_1(n) - e_{\mathrm{b}}(n)m(n) \\
E_{\mathrm{b}}(n) &= \lambda[E_{\mathrm{b}}(n-1) + e_{\mathrm{b}}^2(n)/\varphi(n)] \\
\mathbf{b}(n) &= \mathbf{b}(n-1) + \mathbf{k}'(n)e_{\mathrm{b}}(n)/\varphi(n)
\end{aligned}
$$

Filtering:

$$
\begin{aligned}
e(n) &= y(n) - \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n) \\
\psi\left[\frac{e(n)}{s(n)}\right] &= \tanh\left[\frac{e(n)}{s(n)}\right] \\
\psi'\left[\frac{e(n)}{s(n)}\right] &= 1/\cosh^2\left[\frac{e(n)}{s(n)}\right] \\
\psi'_{\mathrm{f}}\left[\frac{e(n)}{s(n)}\right] &= \begin{cases} \psi'\left[\frac{e(n)}{s(n)}\right] & \text{if } \psi'\left[\frac{e(n)}{s(n)}\right] \geq 0.5 \\ 0.5 & \text{otherwise} \end{cases} \\
\hat{\mathbf{h}}(n) &= \hat{\mathbf{h}}(n-1) + \frac{s(n)}{\psi'_{\mathrm{f}}\left[\frac{e(n)}{s(n)}\right]\varphi(n)}\mathbf{k}'(n)\psi\left[\frac{e(n)}{s(n)}\right] \\
s(n+1) &= \lambda_{\mathrm{s}}s(n) + (1-\lambda_{\mathrm{s}})\frac{s(n)}{\psi'_{\mathrm{f}}\left[\frac{e(n)}{s(n)}\right]}\left|\psi\left[\frac{e(n)}{s(n)}\right]\right|
\end{aligned}
$$

double-talk occurs. Thus, divergence rate is reduced for a duration of about $1/(1-\lambda_{\mathrm{s}})$. This gives ample time for the DTD to act. If there is a system change, the algorithm will not track immediately. However, as the scale estimator tracks the larger error signal, the nonlinearity is scaled up and the convergence rate accelerates. The trade-off between robustness and tracking rate of the adaptive algorithm is thus governed by the tracking rate of the scale estimator, which is controlled by a single parameter λ_{s}.

3. APPLICATION TO NETWORK ECHO CANCELLATION AND SIMULATIONS

In telephone connections that involve connection of 4-wire and 2-wire links, an echo is generated at the hybrid. This echo has a disturbing influence on the conversation and must therefore be cancelled. Figure 1 shows the principle of a network echo canceler (EC). The far-end speech signal $x(n)$ goes through the echo path represented by a filter $\mathbf{h}$, then it is added to the near-end talker signal $v(n)$ and ambient noise $w(n)$. The composit signal is denoted $y(n)$. Most often the echo path is modeled by an adaptive FIR filter, $\hat{\mathbf{h}}(n)$, which subtracts a replica of the echo and thereby achieves cancellation. This may look like a simple straightforward system identification task for the adaptive filter; however, in most conversations there are so-called *double-talk* situations that make the identification much more problematic than what it might appear at a first glance. Double-talk occurs when the two talkers on both sides speak simultaneously, i.e. $x(n) \neq 0$ and $v(n) \neq 0$. In this situation, the near-end speech acts as a large level uncorrelated noise to the adaptive algorithm. The disturbing near-end speech may cause the adaptive filter to diverge. Hence, annoying audible echo will pass through to the far-end. A common way to alleviate this problem is to slow down or completely halt the filter adaptation when presence of near-end speech is detected. This is the very important role of the so called double-talk detector (DTD).

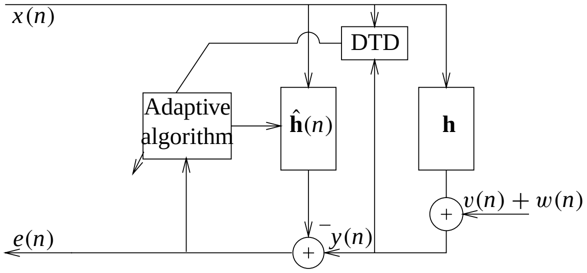

Figure 1. Block diagram of the echo canceler and double-talk detector.

In this section, we wish to compare, by way of simulation, the robust and non-robust FRLS algorithms in the context of network EC with a DTD. In these simulations, we use the Geigel DTD [6] in which the settings are chosen as commonly used in commercial hardware and assumes a 6 dB attenuation. The hybrid attenuation is 20 dB and the length of the echo path $\mathbf{h}$ is $L = 512$. The same length is used for the adaptive filter $\hat{\mathbf{h}}(n)$. The sampling rate is 8 kHz and the signal-to-noise ratio is equal to 39 dB. We have chosen $\lambda_{\mathrm{s}} = 0.992$ for the scale estimate $s(n)$; and $s(n)$ is never allowed to be lower than 0.01. For the adaptive algorithms, we used $\lambda = 1 - 1/(3L)$.

An example of the performance of the robust and non-robust FRLS algorithms during double-talk when speech is used is shown in Fig. 2. The far-end speaker is female and the near-end speaker is male, and the average far- to near-end ratio is 6 dB. The divergence rate of the algorithms does not strongly depend on the power of the near-end signal. We can see that even when the DTD is used, the non-robust FRLS diverges because the DTD does not react fast enough, while for the robust FRLS there is only a slight increase of the misalignment.

Figure 3 shows the behavior after an abrupt system change where the impulse response is shifted 200 samples at 1.5 seconds. The re-convergence rate of the robust algorithm is a little bit slower than the non-robust version but this is the price to pay for robustness against double-talk. Note that since the FRLS algorithm converges rapidly, this somewhat slower convergence is still fully acceptable.

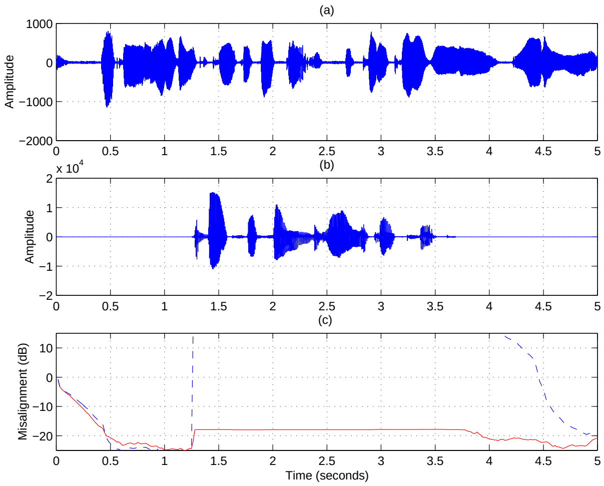

Figure 2. Double-talk situation. (a) Far-end signal. (b) Near-end signal. (c) Misalignment of the robust FRLS (—) and non-robust FRLS (– –).

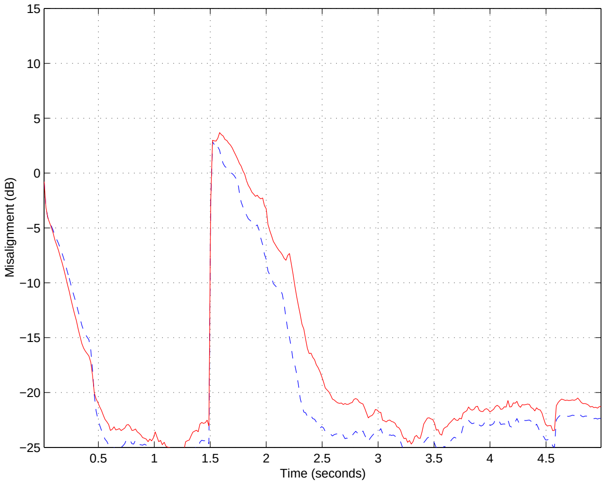

Figure 3. Reconvergence after abrupt hybrid change. Misalignment of the robust FRLS (—) and non-robust FRLS (– –).

4. CONCLUSIONS

In robust statistics, the function $\psi(\cdot)$ which can be directly derived from the model distribution of the error signal plays a key role. If we want to make a robust estimate that has good efficiency, we should choose a ψ that is bounded. In this paper, we proposed to use $\psi(z) = \tanh(z)$ but other choices are possible such as the Huber function [1]. We have shown how to derive robust Newton-type algorithms and from that we have derived a robust RLS algorithm and its fast version. We have also shown that the robust FRLS algorithm has a very nice behavior when it is used for network echo cancellation with a DTD (as simple as the Geigel algorithm) that fails to detect a double-talk situation quickly, which is almost always the case in practice.

5. REFERENCES

[1] P. J. Huber, *Robust Statistics*. Wiley, New York, 1981.

[2] S. A. Kassam and H. V. Poor, “Robust techniques for signal processing: a survey,” *Proc. of the IEEE*, vol. 73, pp. 433-481, Mar. 1985.

[3] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall PTR, Upper Saddle River, N.J., 1993.

[4] S. Haykin, *Adaptive Filter Theory*. Third Edition, Prentice Hall, Englewood Cliffs, N.J., 1996.

[5] T. Gaensler, S. L. Gay, M. M. Sondhi, and J. Benesty, “Double-talk robust fast converging algorithms for network echo cancellation,” *IEEE Trans. Speech Audio Processing*, vol. 8, Nov. 2000.

[6] D. L. Duttweiler, “A twelve-channel digital echo canceller,” *IEEE Trans. Commun.*, vol. 26 , pp. 647-653, May 1978.