Robust Fast Affine Projection Algorithm for Nonlinear Acoustic Echo Cancellation

R. Niemistö[1], *T. Mäkelä*[1] *and V. Myllylä*[2]
[1]Institute of Signal Processing, Tampere University of Technology, Finland.
[2]Nokia Research Center, Finland.
e-mail: {riitta.niemisto, tuomo.makela}@tut.fi, ville.myllyla@nokia.com

ABSTRACT

This paper introduces a robust affine projection algorithm for nonlinear acoustic echo cancellation using simplified Volterra filters. Numerically unstable step of affine projection algorithm, matrix inversion, is improved by approximating the symmetrical matrix with a Toeplitz matrix and applying Levinson-Durbin recursion. Furthermore, the adaptive filter weight update is simplified such that the complexity of the algorithm is comparable to the number of the filter parameters. The simulation results show a slightly better performance than that of the well known normalized LMS algorithm.

1 Introduction

Acoustic echo cancellation has received growing interest in recent years since many governments have forbidden or will forbid the use of cellular phone while driving a car unless a hands-free set is used. In hands-free sets amplifiers, loudspeakers and microphones must be small and they cannot be very expensive. Therefore, they introduce considerable nonlinearities (acoustic distortion) when volume level is high.

General configuration of an echo canceler is shown in Figure 1. We consider echo cancellation during single talk and assume that the level of the near-end signal $e(n)$ is low compared to the level of echo $y(n)$. For hands-free telephones echo attenuation has to be 40 dB during single talk, according to ITU-T Recommendation G.167, while when an adaptive linear filter is used, the echo return loss enhancement

$$\mathrm{ERLE} = 10 \log_{10} \frac{\mathrm{E}(d^2(n))}{\mathrm{E}(\hat{e}^2(n))}$$

achieves its best at about 20–30 dB ($d(n)$ – echo, $\hat{e}(n)$ – residual echo) in realistic environments [3].

Acoustic linear echo path has been successfully modeled using an adaptive filter, usually normalized LMS algorithm, but also affine projection algorithm or RLS algorithm [4]. When model of the echo path is linear in parameters, the parameters can be updated directly using a linear adaptive filtering algorithm. Parameter update is proportional to the echo canceler output $\hat{e}(n)$

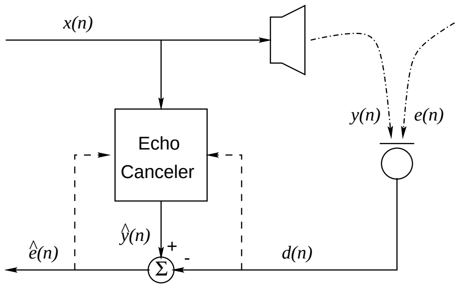

Figure 1: General configuration of an acoustic echo canceler. Signals: $x(n)$ – far-end, $y(n)$ – echo, $e(n)$ – near-end, $d(n)$ – microphone, $\hat{y}(n)$ – replica of the echo and $\hat{e}(n)$ – echo canceler output.

unless a numerically stable variant of RLS, QR–RLS - algorithm, is used. Then the parameters are found by solving a triangular system of equations that is updated using far-end signal $x(n)$ and microphone signal $d(n)$ directly.

There is always some considerable misadjustement present when normalized LMS algorithm is used while RLS algorithm computes an optimal linear estimate of the underlying echo path in least squares sense. RLS algorithm (and its numerically stable variant QR–RLS) is computationally demanding and its fast variants are very sensitive to numerical error. This has lead to the development of affine projection algorithm [2].

The computational complexity of the ordinary affine projection algorithm is $\mathcal{O}(MP + P^2)$, where M is the number of parameters of the underlying adaptive filter and P is the projection order, while the computational complexity of normalized LMS is $\mathcal{O}(M)$ and of RLS is $\mathcal{O}(M^2)$. The projection order P is usually much smaller than the number of parameters M, but when M is large or the projection order P moderate there is still a need to further simplify the algorithm.

In this paper we generalize the robust fast affine projection algorithm presented in [5] for simplified Volterra filters [1] and analyze its performance using real data measured in realistic environment. We compare its performance to that of linear filter and to the performance of adaptive simplified Volterra filter where the normalized LMS or QR–RLS algorithm is used for adaptation.

The organization of this paper is as follows. In the next section we review the robust affine projection algorithm and generalize it for simplified Volterra filters. In section 3 we simulate and compare performance of the filters in acoustic echo cancellation in hands-free sets. Some concluding remarks are found in the last section.

2 The algorithm

We will begin from the ordinary affine projection that is a generalization of the well known normalized LMS algorithm. Let $x(n)$ be the input (far-end signal), $d(n)$ the desired output (microphone signal) and let

$$\vec{u}(n) = (x(n), x(n-1), \ldots, x(n-M+1))^T$$

be the time varying input vector,

$$U(n) = (\vec{u}(n), \vec{u}(n-1), \ldots, \vec{u}(n-P+1))$$

the corresponding input matrix, and

$$\vec{d}(n) = (d(n), d(n-1), \ldots, d(n-P+1))^T$$

the desired response vector. Then the filter weights $\vec{w}(n)$ can be updated as follows

$$\begin{aligned} \vec{y}(n) &= U^T(n)\vec{w}(n-1) \\ \vec{e}(n) &= \vec{d}(n) - \vec{y}(n) \\ R(n) &= U^T(n)U(n) + \delta I \\ \vec{w}(n) &= \vec{w}(n-1) + \mu U(n) R^{-1}(n)\vec{e}(n), \end{aligned}$$

where μ is the step-size and δ the regularization parameter (I being an identity matrix). When the projection order P equals one the algorithm reduces to normalized LMS algorithm.

When the affine projection algorithm is applied for adaptive nonlinear filters the input vector $\vec{u}(n)$ is simply replaced by an extended input vector. When the robust fast affine projection algorithm is generalized we must consider certain simplifications.

The first simplification follows from that if the step size μ equals one and the regularization parameter δ equals zero, then $\vec{e}(n) = (e(n), 0, \ldots, 0)^T$ where $e(n) = d(n) - y(n)$ (the echo canceler output), $y(n) = \vec{u}(n)^T\vec{w}(n-1)$ being the adaptive filter output (replica of echo). Then only the first column of the inverse of $R(n)$ is needed. When the step-size differs from one or the regularization parameter differs from zero we may still approximate the vector $\vec{e}(n)$ with $\hat{\vec{e}}(n) = (e(n), 0, \ldots, 0)^T$ pretty accurately.

The second simplification concerns the symmetrical matrix $R(n)$. The matrix $R(n)$ is of the form

$$R(n) = \begin{pmatrix} r_0(n) & r_1(n) & \cdots & r_{P-1}(n) \\ r_1(n) & r_0(n-1) & \cdots & r_{P-2}(n-1) \\ \vdots & \vdots & \ddots & \vdots \\ r_{P-1}(n) & r_{P-2}(n-1) & \cdots & r_0(n-P+1) \end{pmatrix},$$

where

$$r_p(n) = \vec{u}(n)^T\vec{u}(n-p) = \sum_{m=0}^{M-1} x(n-m)x(n-m-p).$$

Since usually $M >> P$ the elements

$$r_p(n) \approx r_p(n-1) \approx \cdots \approx r_p(n-P+1)$$

and we may approximate $R(n)$ with a Toeplitz matrix

$$\hat{R}(n) = \begin{pmatrix} r_0(n) & r_1(n) & \cdots & r_{P-1}(n) \\ r_1(n) & r_0(n) & \cdots & r_{P-2}(n) \\ \vdots & \vdots & \ddots & \vdots \\ r_{P-1}(n) & r_{P-2}(n) & \cdots & r_0(n) \end{pmatrix}.$$

Using both the simplifications we may compute an estimate $\vec{g}(n)$ of the product $R^{-1}(n)\vec{e}(n)$ efficiently in $\mathcal{O}(P)$ operations using Levinson–Durbin -algorithm. Furthermore, the numerical stability of the inversion can be monitored.

The third simplification concerns the computation of $U(n)\vec{g}(n)$ that as such requires $\mathcal{O}(MP)$ operations. Firstly, the product is approximated by

$$\mu U(n)\vec{g}(n) \approx \vec{u}(n-P+1)s_{P-1}(n),$$

where

$$s_{P-1}(n) = \mu(g_0(n-P+1) + g_1(n-P+2) + \ldots + g_{P-1}(n))$$

and it is computed iteratively. However, this approximation is not usually very accurate and it is compensated when the adaptive filter output is computed by

$$y(n) = \sum_{m=0}^{M-1} x(n-m)w_m(n-1) + \sum_{p=1}^{P-1} r_p(n)s_{p-1}(n-1).$$

The correlations $r_p(n)$ ($p = 0, 1, \ldots, P-1$) result from computation of $\vec{u}^T(n)U(n-1)$ with the first element and vector of $\vec{u}(n)$ and of $U(n-1)$ removed, respectively.

Now we have all to present the robust affine projection algorithm for simplified Volterra filters. Input-output relationship of a Volterra filter can be given as

$$\begin{aligned} \hat{y}(n) = h_0 &+ \sum_{i=0}^{M-1} h_1(i)x(n-i) + \\ &\sum_{i_1=0}^{M-1}\sum_{i_2=i_1}^{M-1} h_2(i_1, i_2)x(n-i_1)x(n-i_2) + \\ &\sum_{i_1=0}^{M-1}\cdots\sum_{i_o=i_{o-1}}^{M-1} h_o(i_1, \ldots, i_o)x(n-i_1)\ldots x(n-i_o), \end{aligned}$$

and there have been several approaches to simplify the relationship [1]. We consider memoryless nonlinearities [6] that have been found as a reliable model for the

Table 1: Robust fast affine projection algorithm for simplified Volterra filtering

Echo canceler output ($p = 0, \ldots, P$):

$u_{(o-1)M+m}(n) = x(n-m)^o, \; m = 0, \ldots, M-1, \; o = 1, \ldots, O$

$$
\begin{aligned}
r_p(n) &= r_p(n-1) + \sum_{o=1}^{O} x(n)^o x(n-p)^o - \\
&\quad \sum_{o=1}^{O} x(n-M+1)^o x(n-p-M+1)^o \\
y(n) &= \sum_{m=0}^{MO-1} w_m(n-1) u_m(n) + \\
&\quad \sum_{p=1}^{P-1} r_p(n) s_{p-1}(n-1) \\
e(n) &= d(n) - y(n)
\end{aligned}
$$

Levinson-Durbin($p = 1, \ldots, P-1$):

$$
\begin{aligned}
E_0(n) &= r_0(n), \; C_0(n) = r_1(n) \\
K_p(n) &= -C_{p-1}(n)/E_{p-1}(n) \\
a_p(n) &= K_p(n), \; a_0(n) = 1 \\
a_i(n) &= a_i(n) + K_p(n) a_{p-i}(n), \; i = 1, \ldots, p-1 \\
E_p(n) &= E_{p-1}(n) + K_p(n) C_{p-1}(n) \\
C_p(n) &= \sum_{i=0}^{p} r_{i+1}(n) a_{p-i}(n)
\end{aligned}
$$

Weight update ($p = 1, \ldots, P, \; m = 0, \ldots, MO-1$):

$$
\begin{aligned}
g_p(n) &= e(n)/E_{P-1}(n) a_p(n) \\
s_p(n) &= s_{p-1}(n-1) + \mu g_p(n) \quad (s_0(n) = 0) \\
w_m(n) &= w_m(n-1) + u_m(n-P+1) s_P(n)
\end{aligned}
$$

nonlinearities that occur in amplifiers and loudspeakers. Then the extended input vector is of the form

$$
\begin{aligned}
\vec{u}(n) = \big(& x(n), \ldots, x(n-M+1), x^2(n), \ldots, \\
& x^2(n-M+1), \ldots, x^O(n), \ldots, x^O(n-M+1) \big),
\end{aligned}
$$

where O is the order of the simplified Volterra filter. When the robust fast affine projection algorithm is generalized for the simplified Volterra case there are only two things that are affected, namely, the computation of the input vector and the computation of the generalized correlations $r_p(n)$. The resulting algorithm is summarized in Table 1.

Numerical stability of the algorithm can be monitored from the reflection coefficient $K_p(n)$. If the underlying Toeplitz matrix is positive definite (i.e. invertible) then $|K_p(n)| < 1$.

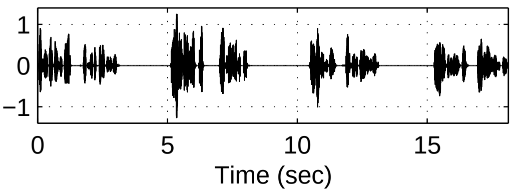

Figure 2: Far-end signal

3 Experimental results

We simulated the performance of the robust affine projection algorithm using test signals that have been measured in a car cabin using realistic test sequences and a hands-free set. The measurements had been carried out on normal and high volume level. Far-end signal is shown in Figure 2. Performance of the algorithm was measured in terms of ERLE computed in windows of length 100 ms. The nonlinear parameters were adapted only during voiced sections, since in the preliminary simulations the simplified Volterra filters were very sensitive to voice activity detection.

We compared the performance of robust affine projection algorithm to the performance of normalized LMS and QR–RLS -algorithm such that we compared the performance of the simplified 5th order Volterra filters to that of linear filters. As shown in Figure 3, we could achieve some improvement (6–8 dB) in terms of ERLE, when QR–RLS algorithm was used, especially in high volume level. Actually, when we used the nonlinear filter on high volume level we could cover almost half of the reduction in performance of linear filters that occurred when volume level was switched from normal to high. When normalized LMS algorithm was used we could not find such improvement, the difference was only 0-3 dB as shown in Figure 4.

Now, ideally with the robust affine projection algorithm we should get better results than with the normalized LMS algorithm the performance of QR–RLS being an upper bound for what we can achieve. However, the resulting ERLE was only slightly higher than that of normalized LMS, as shown in Figure 5 and the robust affine projection algorithm seemed not to be as robust as expected. It tended to become unstable with a high projection order P unless the regularization parameter δ was rather large that resukted as slow convergence.

When QR–RLS algorithm was used with the simplified Volterra filter, the simulation took hours or even days depending on the order of nonlinearity O (we used for experiments a PC with Pentium III at 700MHz), while when normalized LMS algorithm or robust fast affine projection algorith was used adaptation of the simplified Volterra filter could be carried out in real time.

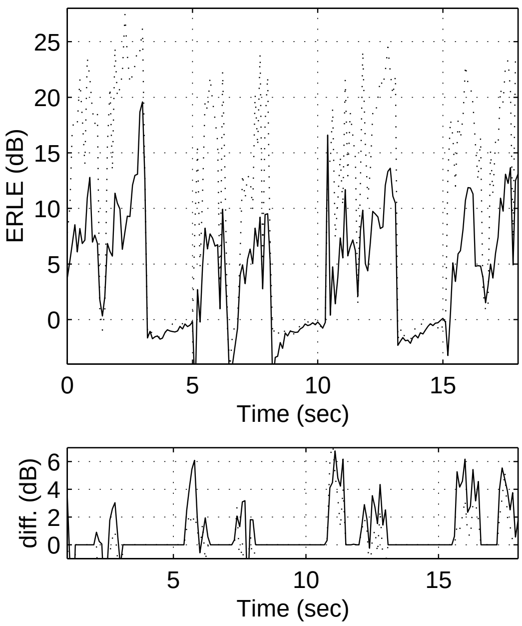

Figure 3: Echo return loss enhancement (ERLE) of a linear acoustic echo canceler when QR–RLS algorithm is used on high (solid) and normal (dotted) volume level and the improvement that can be achieved using simplified 5th order Volterra filters

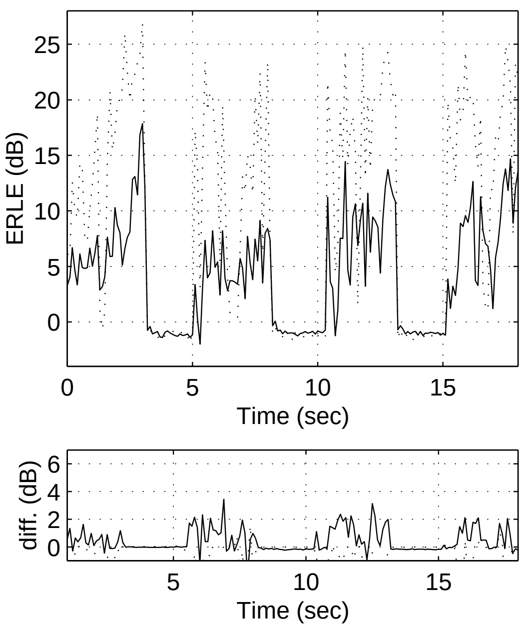

Figure 4: Echo return loss enhancement (ERLE) of a linear acoustic echo canceler when normalized LMS algorithm is used on high (solid) and normal (dotted) volume level and the improvement that can be achieved using simplified 5th order Volterra filters

4 Conclusions

We have generalized the robust fast affine projection algorithm presented in [5] for simplified Volterra filters

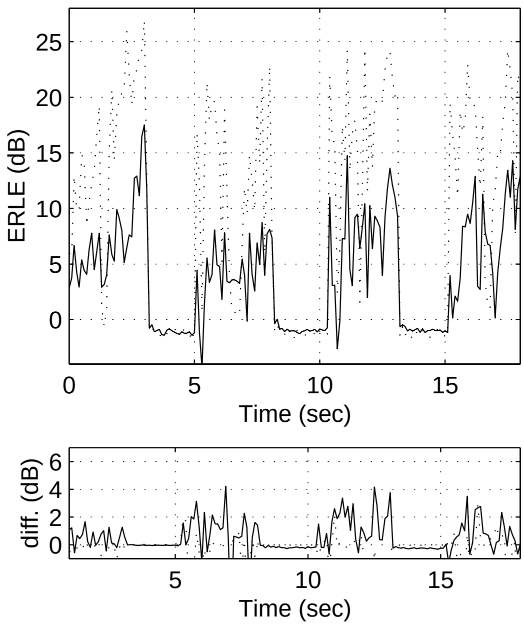

Figure 5: Echo return loss enhancement (ERLE) of a linear acoustic echo canceler when robust affine projection algorithm is used on high (solid) and normal (dotted) volume level and the improvement that can be achieved using simplified 5th order Volterra filters, projection order $P = 2$

and compared its performance to other adaptive algorithms generalized for simplified Volterra filters.

When projection order $P = 2$ we got slightly better performance than that of normalized LMS, but when the projection order was increased the algorithm tended to become unstable The algorithm was outperformed by the QR–RLS -algorithm in the cost of added complexity.

References

[1] A. Fermo, A. Carini, and G. L. Sicuranza. Simplified Volterra filters for acoustic echo cancellation in GSM receivers. In *Proceedings of X European Signal Processing Conference*, volume 4, pages 2413–2416, September 2000.

[2] S. L. Gay and J. Benesty (editors). *Acoustic Signal Processing for Telecommunications*. Kluwer Academic Publisher, Boston/Dordrecht/London, 2000.

[3] E. Hänsler. The hands-free telephone problem: a second annotated bibliography update. In *Proceedings of The International Workshop on Acoustic Echo and Noise Control*, pages 105–114, June 1995.

[4] E. Hänsler. Acoustic echo and noise control: Where do we come from – where do we go? In *Proceedings of Seventh International Workshop on Acoustic Echo and Noise Control*, pages 1–4, September 2001.

[5] V. Myllylä. Robust fast affine projection algorithm for acoustic echo cancelation. In *Proceedings of Seventh International Workshop on Acoustic Echo and Noise Control*, pages 143–146, September 2001.

[6] A. Stenger and W. Kellerman. Adaptation of a memoryless preprocessor for nonlinear acoustic echo cancelling. *Signal Processing*, 80:1747–1760, 2000.