

Fig. 2. Pre-processor module: the clipping function represents the amplifier effects while the power filter models the loudspeaker.

2.1. Pre-processor and clipping model

The pre-processor elements (block 1 in Figure 1, expanded in Figure 2) involve models of the downlink path which includes the amplifier and loudspeaker. In a typical system these components are well-known to have the greatest contribution to non-linearities [5, 8, 9]. They stem from the use of smaller loudspeakers, high signal levels in hands-free mode and from limited amplifier power which may introduce clipping distortion. As illustrated in Figure 2 clipping distortion is modelled with a hard clipping function [3, 4, 10] which has a single parameter c operating on the input $x(n)$:

$$z(n) = f_c(x(n)) = \begin{cases} \text{sign}(x(n))c & \text{if } |x(n)| \geq c \\ x(n) & \text{if } |x(n)| < c \end{cases} \quad (1)$$

where $c \geq 0$ represents the absolute clipping value.

The loudspeaker is modelled with a power filter [6, 9]. Short impulse responses and slow variability (relative to the acoustic channel changes) are generally assumed. As illustrated in Figure 2 the output of the power filter can be written as:

$$\hat{y}_P(n) = \sum_{p=1}^P \underbrace{\mathbf{h}_p^T(n) \mathbf{z}_p(n)}_{=\hat{y}_p(n)} \quad (2)$$

where $\mathbf{z}_p(n) = [z^p(n), z^p(n-1), \dots, z^p(n-N_p-1)]^T$ is the input signal to the p^{th} sub-filter $\mathbf{h}_p(n)$ which has N_p taps and output $\hat{y}_p(n)$.

2.2. Decorrelation filtering and linear AEC

The adaptive decorrelation filter (block 2 in Figure 1) is represented by the adaptive filter $\mathbf{w}(n)$ and is applied to the pre-processor output. Duplicate filtering is applied to the echo signal $y(n)$ so that the echo path estimate will still converge to $\mathbf{h}(n)$ [1, 11]. As in [7, 12] the output is given by:

$$\hat{y}_P^w(n) = \hat{y}_P(n) - \mathbf{w}^T(n) \hat{\mathbf{y}}_P(n-1), \quad (3)$$

which, according to classical LP analysis, should be minimized so that $\hat{y}_P^w(n)$ is decorrelated.

The linear AEC module (block 3 in Figure 1) represents the concatenation of the acoustic channel and the up-link path. The acoustic channel is assumed to be linear and has a significantly longer impulse response and also a higher degree of time variability relative to down-link components (e.g. in the case of a moving, near-end user). Estimation of the echo path is thus generally linear and adaptive in nature [1]. The up-link path includes a microphone and amplifier which generally introduce only small distortion due to low signal levels. It is also generally assumed to be linear [4, 5, 9].

On account of the decorrelation filter the linear AEC operates on

$\hat{y}_P^w(n)$. Thus $\mathbf{h}(n)$ is duplicated and applied to $\hat{y}_P(n)$ as illustrated in Figure 1. The output of the linear AEC module is given by:

$$\hat{y}^w(n) = \mathbf{h}^T(n) \hat{\mathbf{y}}_P^w(n)$$

where $\hat{\mathbf{y}}_P^w(n) = [\hat{y}_P^w(n), \hat{y}_P^w(n-1), \dots, \hat{y}_P^w(n-N-1)]^T$. The real echo estimate $\hat{y}(n)$ is then obtained an identical filter $\hat{\mathbf{h}}(n)$ which is applied to $\hat{y}_P(n)$. The output $\hat{y}(n)$ is therefore exactly the same as in an otherwise entirely conventional AEC implementation and is subtracted from $y(n)$ in the usual way.

3. PARAMETER ESTIMATION

Though conceptually straight forward in principal, the integration of the new clipping compensator and adaptive decorrelation filter into our existing system [6] requires significant development. It is presented here starting with a description of our baseline system.

3.1. Baseline system

The cascaded power filter and linear AEC system are presented in detail in [6] and thus we give here the essential baseline estimation procedures with minimal detail only. Ignoring the clipping compensator in Figure 2, i.e. by assuming that $x(n) = z(n)$, the pre-processor estimate is obtained according to:

$$\hat{\mathbf{h}}_p(n+1) = \hat{\mathbf{h}}_p(n) + \mu_p(n) \underbrace{[\hat{\mathbf{h}}^T(n) \mathbf{Z}_p(n)]^T e(n)}_{=\Delta \mathbf{h}_p(n)} \quad (4)$$

where $\mathbf{Z}_p(n) = [\mathbf{z}_p(n), \mathbf{z}_p(n-1), \dots, \mathbf{z}_p(n-N-1)]^T$ and where $\mathbf{z}_p(n)$ is an input vector with length N_p where $\mu_p(n) = \frac{0.01}{\|\hat{\mathbf{h}}^T(n) \mathbf{Z}_p(n)\|^2 + \epsilon}$ and where ϵ is a regularization factor to avoid division by zero. The estimate of linear filter $\mathbf{h}(n)$ is given by:

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) + \mu_l(n) \hat{\mathbf{y}}_P(n) e(n), \quad (5)$$

where $\mu_l(n) = \frac{0.75}{\|\hat{\mathbf{y}}_P(n)\|^2 + \epsilon}$.

3.2. Clipping compensation (CC)

The proposed clipping estimator is based on the system presented in [10]. In order to derive an LMS-based estimate of the clipping level we need to incorporate the clipping function (1) into an expression for the feedback error leading to:

$$e(n) = y(n) - \mathbf{h}^T(n) \sum_{p=1}^P \mathbf{h}_p^T(n) \underbrace{[f_c(\mathbf{X}(n))]_p}_{=\mathbf{Z}_p^T(n)}$$

where $[f_c(\mathbf{X}(n))]_p$ indicates that the function $f_c(x(n))$ is applied to each element of the matrix $\mathbf{X}(n) = [\mathbf{x}(n), \mathbf{x}(n-1), \dots, \mathbf{x}(n-N-1)]^T$ where $\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-N_p-1)]^T$.

The clipping level is estimated recursively by setting the derivative of the square error with respect to c equal to zero. This leads to:

$$\hat{c}(n+1) = \hat{c}(n) + \mu_c(n) \mathbf{h}^T(n) \sum_{p=1}^P \mathbf{h}_p^T(n) \underbrace{\left[\frac{\partial f_c}{\partial c}(\mathbf{X}(n)) \right]_p}_{=\dot{\mathbf{Z}}_p^T(n)} e(n) \quad (6)$$

where $\mu_c(n) = \frac{0.01}{1+G_c(n) \cdot 0.01}$ is an adaptive step size derived from the least perturbation approach as given in [13] and where $G_c(n) = G_c(n-1) + e^2(n)$. The derivative $\frac{\partial f_c}{\partial c}(x(n))$ is equal to:

$$\frac{\partial f_c}{\partial c}(x(n)) = \begin{cases} \text{sign}(x(n)) & \text{if } |x(n)| \leq c \\ 0 & \text{elsewhere} \end{cases} \quad (7)$$

According to [10] we can furthermore simplify (6) by constraining $\hat{h}_1(n)$ to $\delta(n)$, where $\delta(n)$ is the Dirac function. Equation (6) then becomes:

$$\hat{c}(n+1) = \hat{c}(n) + \mu_c(n) \hat{\mathbf{h}}^T(n) \frac{\partial f_c}{\partial c}(\mathbf{x}_1(n)) e(n) \quad (8)$$

As explained in [5, 10] the constraining of $\hat{h}_1(n)$ to $\delta(n)$ also modifies sub-filter estimates for $p \geq 2$ and the linear AEC estimate $\hat{h}(n)$. In this case the sub-filters $\hat{h}_p(n)$ will converge to $h_1^{-1}(n) * h_p(n)$ and $\hat{h}(n)$ will converge to $h_1(n) * h(n)$.

We also propose here an approach to reduce sub-filter estimation complexity which aims to offset the extra computation introduced through clipping compensation. Computation of the gradient $\Delta \mathbf{h}_p(n)$ in (4) is rather complex as the calculation of $\mathbf{Z}_p(n) \hat{\mathbf{h}}^T(n)$ requires $N_p \times N$ multiplications. A more efficient approximation can be obtained if, for all but the first coefficient of the gradient $\Delta \mathbf{h}_p(n)$, $\hat{\mathbf{h}}(n)$ is replaced by previously calculated echo path estimates. Thus, instead of:

$$\hat{\mathbf{h}}^T(n) \mathbf{Z}_p(n) = [\hat{\mathbf{h}}^T(n) \mathbf{z}_p(n), \dots, \underbrace{\hat{\mathbf{h}}^T(n) \mathbf{z}_p(l)}_{=\tilde{z}_p(l)}, \dots, \hat{\mathbf{h}}^T(n) \mathbf{z}_p(n - N_p - 1)]$$

where $\tilde{z}_p(l) = \hat{\mathbf{h}}^T(n) \mathbf{z}_p(l)$ depends on the current estimate $\hat{\mathbf{h}}(n)$ we use $\tilde{z}_p(l) = \hat{\mathbf{h}}^T(l) \mathbf{z}_p(l)$ which depends on $\hat{\mathbf{h}}(l)$ calculated in previous iterations. This approximation does not require any computation for $l < n$ and leads to:

$$\hat{\mathbf{h}}^T(n) \mathbf{Z}_p(n) = [\hat{\mathbf{h}}^T(n) \mathbf{z}_p(n), \dots, \underbrace{\hat{\mathbf{h}}^T(l) \mathbf{z}_p(l)}_{=\tilde{z}_p(l)}, \dots, \hat{\mathbf{h}}^T(n - N_p - 1) \mathbf{z}_p(n - N_p - 1)]$$

Complexity is thus reduced by a factor of N_p per sub-filter with the added advantage of reacting faster to changes in the echo path. The only drawback is that initial convergence is somewhat slower. Note that a similar simplification can be applied to other cascaded approaches, for example those in [3–5].

3.3. Decorrelation filtering (DF)

Conventional, fixed approaches to decorrelation are not appropriate here due to the use of pre-processing to which the decorrelation filter must adapt. Adaptive decorrelation is thus necessary but is inevitably more complex. Using the LMS criteria to minimise the decorrelation filter output $\hat{y}_P^w(n)$ we obtain an adaptive estimate of $\mathbf{w}(n)$ according to:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu_w(n) \hat{\mathbf{y}}_P(n-1) \hat{y}_P^w(n)$$

where $\mu_w(n) = \frac{\mu}{\|\hat{\mathbf{y}}_P(n-1)\|^2 + \epsilon}$ and where $\mu \leq 0.01$. In practice a larger value of μ is used initially to encourage rapid convergence

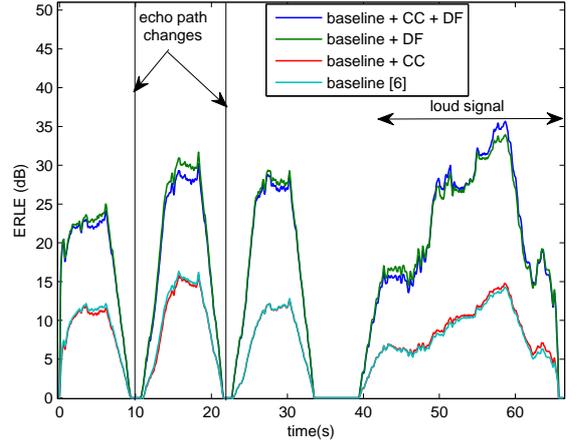


Fig. 3. ERLE against time in an environment where the acoustic channel varies and with an interval of loud speech.

whereas a smaller value is used subsequently for improved stability.

We now consider the effect of decorrelation filtering on other system elements. First $\hat{\mathbf{y}}_P(n)$ in (5) is replaced by $\hat{\mathbf{y}}_P^w(n)$ and similarly $e(n)$ is replaced by $e^w(n)$ as in Figure 1. The input to the linear AEC module is thus decorrelated and so convergence is improved. Second, on account of adaptive pre-processing, the signal $\hat{y}_P(n)$ in (2) will be highly non-stationary. It is then necessary to apply lower step-sizes to sub-filter estimation in order to reduce the non-stationarity in $\hat{y}_P(n)$ and thus to improve decorrelation filtering. For this reason the decorrelation filter is of short order so that it can reliably follow variations in pre-processing. The decorrelation filter also has secondary benefits. In (4) we see that sub-filter estimation uses the linear AEC estimate $\hat{h}(n)$ and will now be more accurate (faster convergence). The pre-processor estimate is then itself more accurate and will converge faster, resulting in more stable sub-filter estimation $\hat{\mathbf{h}}_p(n)$. Note also that, due to the presence of clipping compensation, decorrelation filter estimation should be paused during intervals in which clipping compensation is applied, i.e. when $z(n) = \hat{c}(n)$, since in these intervals a constant-level pre-processor output may disturb estimation. A solution involves changing the decorrelation filter step size to $\bar{\mu}_w(n) = (\neg \frac{\partial f_c}{\partial c}(x(n))) \cdot \mu_w(n)$ where \neg is the logic ‘NOT’ and where $\frac{\partial f_c}{\partial c}(x(n))$ is as given in (7). Hence $\neg \frac{\partial f_c}{\partial c}(x(n))$ is equal to 0 when $z(n) = \hat{c}(n)$ and equal to $\delta(n)$ otherwise.

4. EXPERIMENTAL WORK

All algorithms are assessed using real speech signals recorded from mobile phones in office environments. Two different experiments are reported. The first aims to assess tracking performance whereas the second aims to assess clipping compensation performance. Four algorithms are assessed in both cases: the baseline system [6] with and without clipping compensation (CC) and the baseline system with decorrelation filtering (DF), both with and without clipping compensation (CC). In all cases the linear AEC has $N = 200$ taps, $N_1 = 1$ tap and $N_{p=2,3} = 5$ taps. The decorrelation filter $\mathbf{w}(n)$ has 3 taps. Echo cancellation is assessed in terms of echo return loss enhancement (ERLE) given as $ERLE(m) = 10 \cdot \log_{10} \left(\frac{\sum_{n=m}^{m+M} y^2(n)}{\sum_{n=m}^{m+M} e^2(n)} \right)$ where M is the frame length equal to 512 samples.

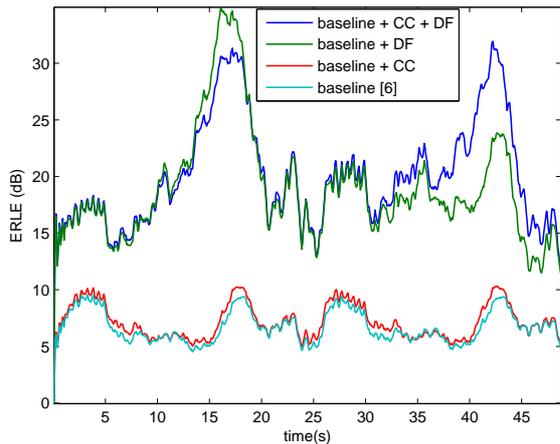


Fig. 4. ERLE against time in environment with high amplifier saturation.

4.1. Tracking performance

In the first experiment tracking performance is assessed using recordings of real mobile devices in hands-free mode with abrupt echo path changes and an interval containing high-level signals in order to induce clipping. Results are illustrated in Figure 3. Clipping compensation without decorrelation filtering (baseline + CC) results in poorer performance than the baseline until after 20s, when the clipping compensation estimate converges, after which there is little difference in performance as the signal level is moderate. After 45s, however, (baseline + CC) performance subsequently improves slightly where compensation is then effective in the case of clipping induced by high-level signals. With decorrelation filtering (baseline + DF) performance improves significantly and we observe rapid convergence during echo path changes or pause to speech transitions around 11, 22 and 39s. When decorrelation filtering is combined with clipping compensation (baseline + CC + DF) performance improves still further where the signal level is high. These results show that decorrelation filtering offsets the effects of pre-processing which introduces greater levels of non-linearity and more correlation through sub-filtering, both of which are harmful to LMS-based adaptive filtering. The fact that (baseline + DF) is better than (baseline + CC) highlights the importance of rapid convergence and accurate acoustic path estimates in cascaded approaches to echo cancellation. We note that clipping compensation is effective only when the acoustic path estimate is accurate (i.e. baseline + CC + DF). With clipping compensation only (i.e. baseline + CC) improvements are negligible.

4.2. Clipping compensation

Clipping compensation performance is assessed by forcing the analog amplifier gain of the mobile terminal to its maximum in order to induce saturation. Results are illustrated in Figure 4 which again shows improved results, particularly with decorrelation filtering. Here, however, as clipping is continuously active, we observe that performance is improved even without decorrelation filtering (baseline + CC). This is expected since power filtering cannot accurately approximate clipping even with Gram-Schmidt orthogonalization [14]. With decorrelation filtering, however, performance is consistently better with clipping (baseline + CC + DF) compensa-

tion than without. Decorrelation filtering improves convergence and ensures accurate estimation of the acoustic path and hence, therefore, also of the clipping level which is dependent upon it (8). We remark that, between 10 and 20s the baseline + DF gives better performance than the baseline + CC + DF. This is explained by the time required for the clipping compensator to converge.

All these observations show that acoustic path tracking is as important as non-linear modelling and that, even with a good model of non-linearities, better performance is difficult to attain when the linear acoustic path estimate is inaccurate.

5. CONCLUSIONS

This paper proposes a simplified approach to model loudspeaker and amplifier non-linearity through independent power filtering and clipping compensation. Decorrelation filtering is shown to bring large improvements in global system performance. The interaction between different system elements is described and estimation procedures for efficient integration are presented. Experimental results show that tracking of the acoustic path is as important as reliable modelling of non-linearities; robust tracking improves pre-processor estimation accuracy, particularly of the clipping compensator. The baseline NLMS-based non-linear AEC algorithm is shown to be largely inefficient for non-linear AEC due to slow convergence; even with a good model of the LEM system improvements are marginal. Thus with cascaded approaches to non-linear AEC involving clipping compensation and decorrelation filtering good tracking performance is essential to global system performance.

6. REFERENCES

- [1] C. Breining, P. Dreiseitel, E. Hänslér, K. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, and J. Tilp, "Acoustic Echo Control, An Application of Very-High-Order Adaptive Filter," *IEEE SP magazine*, pp. 42–69, July 1999.
- [2] A. Fermo, A. Carini, and G. Sicuranza, "Analysis of Different Low Complexity Nonlinear Filters for Acoustic Echo Cancellation," *IWISPA*, pp. 261–266, June 2000.
- [3] B. S. Nollett and D. L. Jones, "Nonlinear Echo Cancellation for Hands-Free Speakerphones," *NSIP*, Sept 1997.
- [4] A. Stenger and W. Kellermann, "Adaptation of a Memoryless Preprocessor for Nonlinear Acoustic Echo Cancellation," *Signal Processing*, vol. 80, pp. 1747–1760, Feb 2000.
- [5] A. Guerin, G. Faucon, and R. L. Bouquin-Jeannes, "Nonlinear Acoustic Echo Cancellation Based on Volterra Filters," *IEEE Trans. on Speech and Audio Proc.*, vol. 11, pp. 672 – 683, Nov 2003.
- [6] M. I. Mossi, C. Yemdji, N. Evans, C. Beaugeant, and P. Degry, "Robust and Low-Cost Non-linear Acoustic Echo Cancellation," *ICASSP*, Mar 2010.
- [7] S. Haykin, *Adaptive Filter Theory 4th Ed.* Prentice Hall, 2001.
- [8] M. I. Mossi, C. Yemdji, N. Evans, C. Herglotz, C. Beaugeant, and P. Degry, "New Models for Characterizing Non-linear Distortions in Mobile Terminal Loudspeakers," *IWAENC*, Sept 2010.
- [9] F. Kuech, A. Mitnacht, and W. Kellermann, "Nonlinear Acoustic Echo Cancellation Using Adaptive Orthogonalized Power Filters," *ICASSP*, vol. III, pp. 105–108, Mar 2005.
- [10] M. I. Mossi, C. Yemdji, N. Evans, and C. Beaugeant, "A Cascaded Non-linear Acoustic Echo Canceller Combining Power Filtering and Clipping Compensation," *EURECOM Research Report*, no. RR-11-258, Aug 2011.
- [11] E. Hänslér and G. Schmidt, *Acoustic Echo and Noise Control, A Practical Approach*. John Wiley & Sons, 2004.
- [12] B. Widrow, *Aspect of Network and System Theory*. Holt, Rinehart and Winston, Inc, 1971, ch. Adaptive Filters.
- [13] Z. Ramadan and A. Poularikas, "New LMS Algorithms Based on Data and Error Nonlinear Functions," *WSEAS*, vol. 3, pp. 2249–2253, July 2004.
- [14] S. Malik and G. Enzner, "Fourier Expansion of Hammerstein Models for Nonlinear Acoustic System Identification," *ICASSP*, March 2011.