Abstract—Adaptive filters for echo cancellation generally need update control schemes to avoid divergence in case of significant disturbances. The two-path algorithm avoids the problem of unnecessary halting of the adaptive filter when the control scheme gives an erroneous output. Versions of this algorithm have previously been presented for echo cancellation. This paper presents a transfer logic which improves the convergence speed of the two-path algorithm for acoustic echo cancellation, while retaining the robustness. Results from simulations show an improved performance, and a fixed-point DSP implementation verifies the performance in real-time.

Index Terms—Acoustic echo cancellation (AEC), two-path.

I. INTRODUCTION

In conventional acoustic echo cancellation (AEC), the echo path, i.e., the loudspeaker-enclosure-microphone (LEM) system, is commonly modeled by a single adaptive finite-impulse response (FIR) filter [1]–[4]. In such a scheme, it is of outmost importance that the filter is not adapted when doubletalk is present, i.e., when both the far-end and the near-end talker are active simultaneously. Updating the filter during doubletalk might lead to filter divergence and a poor AEC performance. Several doubletalk detectors (DTDs)/step-gain controllers, which halt the adaptation during doubletalk, have been proposed [5]–[10]. However, a badly tuned DTD induces the risk of halting the adaptive filter when it should not be halted, e.g., in an echo path change situation. One way to guarantee that the adaptive filter is not unnecessarily halted is to use a secondary FIR filter as in the two-path algorithm [10], [11]. The first (background) filter is continuously adapted, i.e., it is never halted, and the second (foreground) filter is mostly kept in a fixed state. The fixed second filter produces the output. When the first filter is considered to perform better than the second, the filter coefficients of the first filter are copied to the second filter. Several versions of this structure have been proposed for echo cancellation [11]–[16].

In the two-path algorithm, transfer logic controls copying of coefficients from the first to the second filter. Previously, this transfer logic has essentially been based on filter output error comparison [11], [12], [16]. This paper presents an improvement of this transfer logic by the use of a filter deviation estimation method [2], [6], [17].

II. TWO-PATH ALGORITHM

In this section, the two-path algorithm, depicted in Fig. 1, is presented. The far-end speech (loudspeaker) signal \( x(k) \) produces an echo (or “desired” in system identification terminology) signal \( d(k) \) as it passes through the LEM system (\( k \) denotes the sample index). This echo adds to the background noise \( n(k) \) and possible near-end speech \( s(k) \) to form the microphone signal \( y(k) \), i.e., \( y(k) = d(k) + n(k) + s(k) \). The foreground filter, \( \mathbf{w}_f(k) = [w_{f,0}(k), \ldots, w_{f,N-1}(k)]^T \), where \( N \) is the filter length, produces an estimate of the acoustic echo \( \hat{d}_f(k) \). A corresponding echo cancelled (or “error”) signal \( e_f(k) \) is obtained by subtracting this estimate from the microphone signal

\[
e_f(k) = y(k) - \hat{d}_f(k) = y(k) - \mathbf{w}_f(k)^T \mathbf{x}(k) \tag{1}
\]

where \( \mathbf{x}(k) = [x(k), \ldots, x(k-N+1)]^T \). Analogously for the background filter

\[
e_b(k) = y(k) - \hat{d}_b(k) = y(k) - \mathbf{w}_b(k)^T \mathbf{x}(k) \tag{2}
\]

where \( \mathbf{w}_b(k) = [w_{b,0}(k), \ldots, w_{b,N-1}(k)]^T \). The background filter is continuously updated using the normalized least mean square (NLMS) algorithm

\[
\mathbf{w}_b(k+1) = \mathbf{w}_b(k) + \frac{\mu e_b(k) \mathbf{x}(k)}{||\mathbf{x}(k)||^2 + \epsilon} \tag{3}
\]
where \( ||x(k)||^2 = x(k)^T x(k) \) is the squared Euclidean norm, \( \mu \) is the step size control parameter, and \( \epsilon \) is a regularization constant introduced to avoid division by zero [18].

The reason for using NLMS in this paper is that its performance and behavior are well known. Further, the use of the NLMS algorithm facilitates comparison to related papers, e.g., [6]. For acoustic echo cancellation requiring a large number of filter coefficients (\( N > 1000 \)), the full-band NLMS is not an optimal scheme due to its slow convergence. Examples of other more suitable algorithms, e.g., subband, frequency domain and affine projection methods, can be found in, e.g., [1], [18]. The method proposed in this paper is, however, not limited to NLMS-based two-path cancellation but can be used in conjunction with essentially any other adaptive algorithm for the background filter update.

### A. Transfer Logic

If the background filter is estimated to be better tuned than the foreground filter, its filter coefficients are copied to the foreground filter. This is controlled by comparisons between the short-term powers of the signals \( x(k), y(k), e_f(k), \) and \( e_b(k) \). In the original two-path algorithm, the update conditions for the foreground filter [11] are basically as given by

\[
\frac{P_y(k)}{P_x(k)} < T_{y,x} \quad (4)
\]

\[
\frac{P_{e_b}(k)}{P_y(k)} < T_{e_b,y} \quad (5)
\]

\[
\frac{P_{e_f}(k)}{P_{e_f}(k)} < T_{e_f,e_f} \quad (6)
\]

where \( T_{y,x}, T_{e_b,y}, T_{e_f,e_f} \) are thresholds and the power estimate is given by, e.g.,

\[
P_x(k) = \frac{1}{M} \sum_{i=0}^{M-1} x^2(k - i) \quad (7)
\]

where \( M \) is the update interval. In the transfer logic of the original two-path algorithm, the foreground filter updating is performed at every \( M \)th sample in order to reduce computational complexity and memory requirement.

The filter \( w_f(k) \) is updated with the filter \( w_b(k) \) if all of the conditions (4), (5), and (6) are true. Condition (4) is basically the classical Geigel DTD [9]. Condition (5) implies that no updating is done when \( w_b(k) \) is considered to perform poorly in terms of echo cancellation. Condition (6) is satisfied when filter \( w_f(k) \) produces a smaller error signal \( e_f(k) \) compared to \( e_b(k) \).

Intuitively, condition (6) can be seen as the core condition, determining if \( w_f(k) \) is better tuned than \( w_f(k) \), while conditions (4) and (5) are used to avoid erroneous updates during doubletalk.

Since the check for update is only performed every \( M \) sample, there is an intrinsic delay in convergence of \( M \) samples. Thus, the value of \( M \) should be chosen considering the convergence rate of the adaptive algorithm. For example, a too high value of \( M \) will not yield any significant extra reduction in complexity but will slow down convergence.

Often in an acoustic environment, the acoustic coupling between the microphone and the loudspeaker makes it hard or even impossible to detect present near-end speech by comparing average energy of the loudspeaker and microphone signal [7]. Thus, for acoustic echo cancellation in general, condition (4) is not suitable as a doubletalk detector.

Condition (5) estimates the reduction of the echo as it passes the AEC. Since one acoustic environment can vary greatly from another (in terms of loudspeaker to microphone distance, room reverberation, nonlinearities in the involved components, etc.), the practically achievable echo cancellation also differs significantly from situation to situation and is hard to predict. This makes condition (5) impractical, since an acoustic situation with an echo reduction level that is higher than the \( T_{e_b,y} \) threshold would lead to absent foreground filter updates.

As argued above, the use of conditions (4) and (5) has major drawbacks in an acoustic environment since these were originally intended to be used for line echo cancellation, where the echo cancellation performance is fairly predictable and the received echo relatively low. It is therefore suggested to replace both conditions with

\[
P_{e_b}(k) < T_{e_b,x} P_x(k) \quad (8)
\]

where \( T_{e_b,x} \) is a threshold. This condition is used in conjunction with the two-path algorithm in [14] and is basically a DTD operating on \( x(k) \) and \( e_b(k) \) [6]. From a slightly different perspective, condition (8) can be seen as the core DTD and the two-path algorithm as a complement which prevents deadlock in an echo path change situation, compare further with the “shadow” filter discussion in [6].

The foreground filter update is then given by

\[
\begin{cases} 
  w_f(k) = w_b(k), & \text{if } (6) \text{ AND } (8) \text{ are TRUE} \\
  w_f(k), & \text{otherwise.} 
\end{cases} \quad (9)
\]

Doubletalk becoming active just a few samples prior to the update check could lead to divergence of the filter \( w_f(k) \). However, since only a few samples are affected by doubletalk, this might pass undetected, and the diverged filter coefficients are copied into filter \( w_f(k) \). The situation can be avoided by requiring the copy conditions to be true for two consecutive \( M \) sample periods and updating the filter \( w_f(k) \) with an \( M \) sample delayed version \( w_b(k - M) \).

We denote the solution described in this section (9) as the conventional two-path (CTP) solution.

### B. Threshold Settings

Setting the thresholds \( T_{e_b,x} \) and \( T_{e_b,e_f} \) is nontrivial and crucial for overall system performance and typically involves a tradeoff between convergence speed and stability. To allow as smooth and fast convergence as possible, it is desirable to set the thresholds \( T_{e_b,x} \) and \( T_{e_b,e_f} \) high, which in practice means close to 0 dB. On the other hand, setting these thresholds low reduces the risk of erroneous copying of the filter coefficients during doubletalk.

The condition (8) estimates the total echo return loss, and the foreground filter is not updated until the total echo return loss is below \( T_{e_b,x} \). Thus, setting the threshold \( T_{e_b,x} \) too low might lead to the filter not being updated at all. A sensible threshold setting will depend on the application.
Since speech is a highly nonstationary correlated signal, it can occur during doubletalk, that the background filter manages to cancel a significant part of the microphone signal (and thus a cancellation of the near-end speech as well), without the background filter being well tuned. It is therefore a risk of wrongly adapted filter coefficients being copied into the foreground filter. To eliminate the risk of divergence, the threshold \( T_{e_b,e_f} \) must thus be set at a sufficiently low level.

The proposed overall approach is to set \( T_{e_b,e_f} \) as low as the application allows, which might be just a few decibels below 0 dB, and thereafter set \( T_{e_b,e_f} \) as high as possible, while still ensuring performance during doubletalk.

The condition on \( T_{e_b,e_f} \) which fulfills the robustness requirement will thus imply a reduction of convergence speed. In this paper, a complementary update condition is proposed that will help increase the convergence speed without compromising the robustness, see Section III.

### III. Proposed Improvement of the Two-Path Transfer Logic

A measure of filter convergence is the deviation (or system distance) [1], [18]. The normalized square deviation (NSD) \( D_{w_b}(k) \) of filter \( w_b(k) \) from a LEM impulse response \( h_{LEM} = [h_{LEM,0}, \ldots, h_{LEM,N-1}]^T \) is given by

\[
D_{w_b}(k) = \frac{\sum_{i=0}^{N-1} (h_{LEM,i} - w_{b,i}(k))^2}{||h_{LEM}||^2}. \tag{10}
\]

The NSD of filter \( w_f(k) \), \( D_{w_f}(k) \) can be calculated analogously. Ideally, \( D_{w_f}(k) \) should be updated when

\[
D_{w_b}(k) < D_{w_f}(k). \tag{11}
\]

However, it is possible for the short-time power of the error signal from the adapting background filter to be lower than the short-time power of the error signal from the fixed foreground filter (i.e., \( P_{e_b}(k) < P_{e_f}(k) \)) even though the foreground filter is a better model of the echo path (i.e., \( D_{w_b}(k) > D_{w_f}(k) \)). This can occur during doubletalk due to near cancellation of the near-end speech, or due to far-end single talk due to the nonstationary nature of speech [16] (also see Section IV in this paper for experimental verification). As a result, erroneous filter copying could occur, which in turn leads to reduced echo cancellation and ultimately cause system divergence. The solution to this is, as mentioned in the previous section, to keep \( T_{e_b,e_f} \) at a sufficiently low level. A low value of \( T_{e_b,e_f} \) introduces the problem of a slower convergence of the foreground filter, which is the major drawback of the two-path algorithm [5], as well as (in some cases) a higher steady-state error.

This paper proposes a modification which reduces the problem of a slower convergence without compromising the performance during doubletalk. The improvement consists of an additional update condition [as a complement to condition (6)] based on the estimated squared deviation.

In order to obtain this estimate, an artificial delay of \( L \) samples is inserted into the signal path of \( y(k) \) just before the subtraction yielding \( e_b(k) \), see Fig. 2. Note that this does not delay the output signal. Moreover, the background filter length is increased by \( L \) samples, resulting in an extended background filter \( w_e(k) \), according to

\[
w_e(k) = \begin{bmatrix} w_z(k) \\ w_b(k) \end{bmatrix} \tag{12}
\]

where \( w_z(k) = [w_{z,0}(k), \ldots, w_{z,L-1}(k)]^T \). This assures that the optimal values of the first \( L \) coefficients (i.e., \( w_z(k) \)) of filter \( w_e(k) \) are zero. According to [1] and [17], the NLMS algorithm spreads the error evenly among the filter coefficients. Therefore, the norm of the extension coefficients can be used as an un-normalized, signal energy weighted, estimate of the filter deviation \( \hat{D}_{w_b}(k) \)

\[
\hat{D}_{w_b}(k) = ||w_z(k)||^2. \tag{13}
\]

Setting \( L \) too low will yield a poor estimator. However, the extension of the background filter implies increased memory and complexity requirements, which is directly dependent on \( L \), see Section IV. Thus, there is a tradeoff situation.

The deviation estimate in (13) is based on the assumption that the NLMS algorithm spreads the error evenly among the filter coefficients, as mentioned earlier. However, the proposed algorithm is not limited to the NLMS algorithm but can be used in conjunction with any adaptive algorithm that fulfills this property. Note that in [1] and [17], the above deviation estimate method is used for determining the optimal step-size. In this paper, the same method is instead used to improve the update control of the two-path algorithm.

The foreground filter \( w_f(k) \) consists of a previous copy of the background filter. Thus, the deviation estimate of the foreground filter \( \hat{D}_{w_f}(k) \) is given by

\[
\hat{D}_{w_f}(k+1) = \begin{cases} \hat{D}_{w_b}(k), & \text{if the foreground filter is updated} \\ \hat{D}_{w_f}(k), & \text{otherwise} \end{cases} \tag{14}
\]

If \( \hat{D}_{w_f}(k) \) is less than \( \hat{D}_{w_f}(k) \), the background filter is better in the estimated deviation sense and an update should be per-
formed. Thus, the following additional update condition is proposed: update if

$$\frac{\hat{D}_{w_0}(k)}{\hat{D}_{w_f}(k)} < T_{b,f}$$

(15)

where $T_{b,f}$ is a threshold.

The proposed condition (15) is combined with the previous conditions (6) and (8) to form a foreground filter update condition in the following manner:

$$w_f(k + 1) = \begin{cases} w_b(k), & \text{if (6) OR (15) AND (8) is TRUE} \\ w_f(k), & \text{otherwise}. \end{cases}$$

(16)

It might seem that condition (15) can replace the previous condition (6). However, this is not the case. A change of the echo path (by translation of the loudspeaker or microphone for example) might introduce a new LEM system which is harder to estimate than the previous. Then, $\hat{D}_{w_0}(k)$ will be larger than $\hat{D}_{w_f}(k)$, (since $\hat{D}_{w_f}(k)$ is relative to the previous echo path) until the first foreground filter update, which must then be triggered by condition (6).

The virtues of introducing the new condition (15) is that the filter update can be performed more often, hence, resulting in better convergence of the two-path algorithm, and in some cases a lower steady-state error. We denote the proposed scheme the improved two-path (ITP) solution.

IV. EVALUATION

In the evaluation, a typical speech signal is used as the input signal $x(k)$, see Fig. 3. The signal $y(k)$ is obtained through

$$d(k) = \begin{cases} x(k)^T h_1, & \text{if } k < I \\ x(k)^T h_2, & \text{otherwise} \end{cases}$$

(17)

$$y(k) = \begin{cases} d(k) + b(k), & \text{if } k < J \\ d(k) + b(k) + \alpha s(k), & \text{otherwise} \end{cases}$$

(18)

where $h_1 = [h_{1,0}, \ldots, h_{1,N-1}]^T$ and $h_2 = [h_{2,0}, \ldots, h_{2,N-1}]^T$ are FIR models of two different LEM systems, corresponding to two different spatial positions of the microphone, as in [2]. $b(k)$ is an ambient background noise signal with energy level 30 dB below the energy level of $d(k)$. Further, $s(k)$ is a bursty speech signal, and $I$ and $J$ are indices controlling the occurrence of echo path change and doubletalk, respectively. The parameter $\alpha$ controls the near-end speech level, and the sampling frequency is set to 8 kHz. A practical AEC implementation typically achieves about 30-dB echo cancellation or more under favorable conditions [1], although performance in a difficult environment, e.g., with a lot of movements, can be significantly worse. The threshold $T_{e_{b,e}}$ is set to $-18$ dB to allow a margin for the AEC under these conditions. The thresholds, $T_{e_{b,e_1}}$, is set to $-12$ dB, which (under the given conditions) is the highest possible setting that still guarantees robust performance during doubletalk. The threshold value for $T_{e_{b,e_2}}$ was found through extensive simulations, by varying the parameters $I$, $J$, $\alpha$, and $\mu$. These parameters were varied in the ranges $[0, 480,000]$, $[0, 480,000]$, $[0, 0.1]$, $[0.0625, 0.5]$, respectively. The default settings, $I_0$, $J_0$, $\alpha_0$, and $\mu_0$, of these parameters as well as the settings of other parameters are summarized in Table I, i.e., the settings for the illustrated examples in all figures are as in Table I if not stated otherwise. The step-size $\mu_0$ was determined through simulations as the value giving the fastest convergence without risking divergence.

The proposed solution was also implemented on a fix-point processor [19]. The extension of the background filter by $L$ coefficients implies an increased memory requirement of $L$ elements. Further, the complexity is increased by $L$ multiplications and additions for the filtering and $L$ multiplications and additions for the NLMS-update. The evaluation of (13) is performed only every $M$th sample in conjunction with the other update check calculations. At these sample instances, the update of the NLMS is omitted. In the fix-point processor implementation the squared sum in (13) was replaced with a sum of absolute values in order to reduce complexity. This replacement had no significant impact on the performance and the complexity reduction implied that the update could be fitted and no extra complexity was added. Thus, the increased complexity when implementing the proposed algorithm in about $2L$ extra additions and $2L$ extra multiplications, as compared to the conventional two-path algorithm.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>1800</td>
</tr>
<tr>
<td>$M$</td>
<td>2000</td>
</tr>
<tr>
<td>$L$</td>
<td>50</td>
</tr>
<tr>
<td>$T_{e_{b,e}}$</td>
<td>$-18$ dB</td>
</tr>
<tr>
<td>$T_{e_{b,e_1}}$</td>
<td>$-12$ dB</td>
</tr>
<tr>
<td>$T_{b,f}$</td>
<td>0dB</td>
</tr>
<tr>
<td>$\mu_0$</td>
<td>0.5</td>
</tr>
<tr>
<td>$\alpha_0$</td>
<td>0.25</td>
</tr>
<tr>
<td>$I_0$</td>
<td>225000</td>
</tr>
<tr>
<td>$J_0$</td>
<td>250000</td>
</tr>
</tbody>
</table>

Fig. 3. Speech signals, far-end speech (upper plot) and near-end speech (lower plot). Doubletalk is present from about 28 s, as shown in the lower plot. (a) Far-end speech. (b) Near-end speech.
V. RESULTS

A simulated doubletalk situation using the conventional solution (CTP) for two different \( T_{\text{e}_F} \) threshold settings is shown in Fig. 4. This figure illustrates the tradeoff between convergence and robustness to doubletalk in the conventional two-path algorithm. As can be seen in Fig. 4(a), the \( T_{\text{e}_F} = -6 \) dB threshold is too high, since the foreground filter deviation increases during doubletalk, interval 28–60 s. This is prevented with the \( T_{\text{e}_F} = -12 \) dB threshold, shown in the lower plot (b), but at the cost of slower foreground filter convergence and a slightly larger steady-state deviation. The slower convergence can be observed by comparing the upper and lower plot in the interval 5–10 s, where the foreground filter deviation in the upper plot better follows the deviation of the converging background filter. The larger steady-state deviation can be observed in the interval 12–28 s, i.e., in the lower plot the foreground filter deviation does not reach the \( -29 \) dB deviation level of the background filter.

Fig. 5 illustrates the previously discussed problem of only considering the filter output errors in the two-path transfer logic. Note that \( P_{\text{e}_o}(k) \) [solid line, Fig. 5(b)] occasionally is lower than \( P_{\text{e}_f}(k) \) [dashed line, Fig. 5(a)] during the doubletalk period despite the fact that the foreground filter [dashed line, Fig. 5(a)] is better tuned than the background filter [solid line, Fig. 5(a)]. The figure thus demonstrates that it is possible for the output error from the background filter to be smaller than the corresponding error from the foreground filter, despite the foreground filter being a more accurate model (in the normalized squared deviation sense) of the echo path. This again justifies the proposed control logic improvement of the ITP-solution.

Fig. 6 illustrates how the performance varies with different values of \( L \). It is shown that setting \( L \) too low might lead to reduced improvement in convergence speed of the proposed solution. Since the complexity increases with an increased value of \( L \), there is thus a tradeoff situation between performance and complexity. The optimal choice of \( L \) will depend on the application at hand.

The ITP and CTP solutions were evaluated in a large number of doubletalk simulations for different values of \( J, \alpha \) and \( \mu \), with other parameter values as shown in Table I. Both solutions were robust during doubletalk, i.e., none of them diverged during the simulations.
Likewise, a large number of echo path change simulations were conducted for different values of $I$ and $\mu$. The simulations clearly demonstrated the improved performance of the ITP solution. A series of simulations for three different values of the step-size $\mu$ with $I = 2.5 \times 10^5$, i.e., echo path change occur at 31 s, are depicted in Fig. 7. The improved convergence rate of the proposed ITP solution can be observed in Fig. 7(a)–(c), in that the ITP foreground filter (dotted line) better follows the converging background filter (solid line), as compared to the CTP foreground filter (dashed line).

Fig. 8 demonstrates the functionality of the proposed algorithm implemented on a fix-point processor operating in real-time. In the plot of Fig. 8(a), the convergence of the foreground filter can be observed, echo path change occurs at about 28 s. In Fig. 8(b), the doubletalk robustness can be observed again, doubletalk is present from about 28 s. During doubletalk, the background filter (solid line) performs poorly, only about 15-dB echo cancellation. The foreground filter (dashed line), which generates the output signal, continues to yield a low residual echo during doubletalk. In the experiments depicted in Fig. 8, the acoustic echo was obtained through a real LEM system. The echo path change is obtained by a translation of the microphone. In the doubletalk case, the near-end speech signal $s(k)$ was not added acoustically, but electronically. This was done in order to be able to evaluate only residual echo during doubletalk, i.e., to be able to remove the near-end speech $s(k)$ from the echo cancelled signal.

VI. CONCLUSION

This paper proposed a transfer logic solution for a two-path algorithm for acoustic echo cancellation and shown convergence speed improvement compared to previous solutions. This is achieved while maintaining the robustness, which is one of the main advantages of using two echo cancellation filters. Supporting results were obtained through experiments with both simulated and real signals.
ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their helpful and constructive comments, Analog Devices for technical support, and the Swedish Knowledge Foundation (KKS) for funding.

REFERENCES