LOW-COMPLEXITY ALGORITHMS FOR ECHO CANCELLATION IN AUDIO CONFERENCING SYSTEMS

Christian Schüldt

Blekinge Institute of Technology
Doctoral Dissertation Series No. 2012:13

School of Engineering

Low-Complexity Algorithms for Echo Cancellation in Audio Conferencing Systems

Christian Schüldt

Blekinge Institute of Technology doctoral dissertation series
No 2012:13

Low-Complexity Algorithms for Echo Cancellation in Audio Conferencing Systems

Christian Schüldt

Doctoral Dissertation in
Telecommunications

School of Engineering
Blekinge Institute of Technology
SWEDEN

School of Engineering
Publisher: Blekinge Institute of Technology,
SE-371 79 Karlskrona, Sweden
Printed by Printfabriken, Karlskrona, Sweden 2012
ISBN: 978-91-7295-242-3
ISSN 1653-2090
urn:nbn:se:bth-00541

Preface

This doctoral thesis summarizes my work in the field of echo cancellation in audio conferencing systems with focus on algorithms requiring low computational resources. The research has been carried out in a joint collaboration between Blekinge Institute of Technology in Karlskrona, Konftel AB in Umeå and Limes Audio AB in Umeå. The thesis is comprised of an introduction followed by six independent parts:

Part	
I	An Improved Deviation Measure for Two-Path Echo Cancellation
II	Evaluation of an Improved Deviation Measure for Two-Path Echo Cancellation
III	A Delay-Based Double-Talk Detector
IV	Robust Low-Complexity Transfer Logic for Two-Path Echo Cancellation
V	Adaptive Filter Length Selection for Acoustic Echo Cancellation
VI	A Low-Complexity Delayless Selective Subband Adaptive Filtering Algorithm

Acknowledgments

First of all, I thank my assistant advisor, friend and mentor Dr. Fredric Lindström, who is the main reason why I became engaged in this particular field of research. His drive, inspiration and enthusiasm have helped me immensely.

Secondly, I thank Prof. Ingvar Claesson, who is the other reason for me engaging in this field of research, for providing guidance as well as great scientific insights.

I would also like to thank my colleagues (none mentioned, none forgotten) at Blekinge Institute of Technology, Konftel AB and Limes Audio AB for their support. My journey towards the doctoral degree would not have been the same without you.

Finally, I would like to thank my girlfriend Linn Ristborg for all the love and support throughout the years.

Christian Schüldt
Stockholm, August 2012

Contents

Publication list 3

Introduction 9
A brief description of the audio transmission path and echoes in a typical telephone call 10
Acoustic echo cancellation 11
Controlling the adaptive filtering process 16
Two-path echo cancellation 17
Residual echo suppression 18
Computational complexity reduction 18

Thesis summary 21

Part I
An Improved Deviation Measure for Two-Path Echo Cancellation ..31

Part II
Evaluation of an Improved Deviation Measure for Two-Path Echo Cancellation 43

Part III
A Delay-Based Double-Talk Detector 55

Part IV
Robust Low-Complexity Transfer Logic for Two-Path Echo Cancellation 83

Part V
Adaptive Filter Length Selection for Acoustic Echo Cancellation ... 97

Part VI
A Low-Complexity Delayless Selective Subband Adaptive Filtering Algorithm 125

Publication list

Part I has been published as:

C. Schüldt, F. Lindstrom and I. Claesson, "An Improved Deviation Measure for Two-Path Echo Cancellation," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 305-308, Dallas, TX, March 2010.

Part II has been published as:

C. Schüldt, F. Lindstrom and I. Claesson, "Evaluation of an Improved Deviation Measure for Two-Path Echo Cancellation," In *Proceedings of International Workshop on Acoustic Echo and Noise Control (IWAENC)*, Tel Aviv, Israel, September 2010.

Part III has been published as:

C. Schüldt, F. Lindstrom and I. Claesson, "A Delay-Based Double-Talk Detector," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1725-1733, February 2012.

Part IV has been published as:

C. Schüldt, F. Lindstrom and I. Claesson, "Robust Low-Complexity Transfer Logic for Two-Path Echo Cancellation," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 173-176, Kyoto, Japan, March 2012.

Part V has been published as:

C. Schüldt, F. Lindstrom, H. Li, and I. Claesson, "Adaptive Filter Length Selection for Acoustic Echo Cancellation," *Signal Processing*, vol. 89, no. 6, pp. 1185-1194, June 2009.

Part VI has been published as:

C. Schüldt, F. Lindstrom and I. Claesson, "A Low-Complexity Delayless Selective Subband Adaptive Filtering Algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 12, pp. 5840-5850, August 2008.

Other publications in conjunction with the thesis

M. Borgh, M. Berggren, C. Schüldt. F. Lindstrom and I. Claesson, "An Improved Adaptive Gain Equalizer for Noise Reduction with Low Speech Distortion,", *EURASIP Journal on Audio, Speech, and Music Processing*, 2011:7, doi: 0.1186/1687-4722-2011-7, August 2011.

M. Berggren, M. Borgh, C. Schüldt. F. Lindstrom and I. Claesson, "Low-Complexity Network Echo Cancellation Approach for Systems Equipped with External Memory,", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 8, pp. 2506-2515, doi:10.1109/TASL.2011.2144972, April 2011.

C. Schüldt, F. Lindstrom and I. Claesson "A Distortion Reducing Subband Limiter Implementation for Conference Phones," In *Proceedings of IEEE International Conference on Consumer Electronics*, Las Vegas, NV, January 2008.

F. Lindstrom, C. Schüldt, M. Långström and I. Claesson, "A Method for Reduced Finite Precision Effects in Parallel Filtering Echo Cancellation," *IEEE Transactions on Circuits and Systems Part I: Regular Papers*, vol. 54, pp. 2011-2018, September 2007.

F. Lindstrom, C. Schüldt and I. Claesson, "An Improvement of the Two-Path Algorithm Transfer Logic for Acoustic Echo Cancellation," *IEEE Transactions on Audio, Speech and Language Signal Processing*, vol. 15, pp. 1320-1326, May 2007.

F. Lindstrom, C. Schüldt and I. Claesson, "A Hybrid Acoustic Echo Canceller and Suppressor," *Signal Processing*, vol. 87, pp. 739-749, April 2007.

F. Lindstrom, C. Schüldt and I. Claesson, "Efficient Multichannel NLMS Implementation for Acoustic Echo Cancellation," *EURASIP Journal on Audio, Speech, and Music Processing*, Article ID 78439, 6 pages, doi:10.1155/2007/78439, January 2007.

C. Schüldt, F. Lindstrom and I. Claesson "A Combined Implementation of Echo Suppression, Noise Reduction and Comfort Noise in a Speaker Phone Application," In *Proceedings of IEEE International Conference on Consumer Electronics*, Las Vegas, NV, January 2007.

C. Schüldt, F. Lindstrom and I. Claesson, "Low-Complexity Adaptive Filtering Implementation for Acoustic Echo Cancellation," In *Proceedings of IEEE TENCON*, Hong Kong, November 2006.

F. Lindstrom, C. Schüldt and I. Claesson, "Reusing Data During Speech Pauses in an NLMS-based Acoustic Echo Canceller," In *Proceedings of IEEE TENCON*, Hong Kong, November 2006.

F. Lindstrom, C. Schüldt, M. Dahl and I. Claesson, "Improving the Performance of a Low-Complexity Doubletalk Detector by a Subband Approach," In *Proceedings of the Third IEEE International Conference on Systems, Signals & Devices*, vol. III, Sousse, Tunisia, March 2005.

Patents filed

C. Schüldt and F. Lindstrom, "Method and device for microphone selection," Sweden Application Serial No. 1150031-1, filed on January 19, 2011. Patent pending.

F. Lindstrom, C. Schüldt and I. Claesson, "Device and method for controlling damping of residual echo," Sweden Application Serial No. 0901012-5, filed on July 20, 2009. PCT Application Serial No. PCT/SE2010/050676, filed on June 17, 2010. U.S. Application Serial No. 13/384554, filed on January 17, 2012.

Introduction

The problem with echoes in telephone systems has been present ever since the beginning. To avoid acoustic echo from the loudspeaker to be picked up by the microphone, early telephones comprised a loudspeaker that was to be held close to the ear with one hand and a separate microphone to be held close to the mouth with the other hand [1]. This design eventually evolved into a handset containing both the loudspeaker and the microphone, to be held only with one hand. In the early 1900s, loudspeaker telephones primarily intended for managers in an office environment, eliminating the need of a handset and thus allowing the user to have his or her hands free while communicating, were introduced [2, 3].

In addition to the acoustic echo, electrical *line-/network echoes* originating from impedance mismatches in the telephone network also occur. The impedance mismatches stem from the fact that the *local loop*, i.e. the circuit connecting the local telephone subscriber/user with the central telephone telephone exchange/switch, varies in impedance for different subscribers depending on the wire length and type of telephone. Moreover, since two wires are used for the local loop (for economic reasons) and four wires are used for connections between telephone exchange offices (due to the need of amplification to compensate for signal loss in the long cables), a 2/4-wire converter also called a *hybrid* is used and essentially all the significant electrical echoes on the telephone network arise at these hybrids [4, 5].

To combat the problems of both the acoustic- and electrical echoes, primitive voice-controlled switching was initially used [2, 6]. This voice-controlled switching mechanism, denoted *echo suppressor*, allows only one person to speak at a time, a so-called *half-duplex* solution, since the audio in the other direction had to be suppressed due to the present echo. An echo suppressor for electrical echoes works reasonably well in situations with low round-trip delay (below 100 milliseconds [4]) and high signal-to-noise ratio, while an acoustic echo suppressor also requires a well damped room since the reverberation time determines how fast the suppressor can switch without allowing echo to slip through. Increased round-trip delay means that each participant will have to wait longer for a response, which significantly increases the number of *double-talk* occurrences, i.e. situations where both parties are speaking simultaneously. Since the basic echo suppressor only allows audio in one direction at a time, a double-talk situation will mean that one party is muted which in turn significantly reduces both the intelligibility as well as the listener com-

fort. A modification of the classic echo suppressor, denoted *center-clipping*, was proposed [8] to somewhat aid this problem by allowing audio to flow in both directions simultaneously only if the energies of the signals are above a threshold. The assumption is that the echo is significantly lower than the speech signal, so if the signal contains only echo the signal energy is below the threshold, and if the signal contains speech (with or without echo) the signal energy is above the threshold. A signal containing mixed speech and echo will be allowed to pass through, but since the speech is assumed to be stronger than the echo, the echo will be somewhat masked. Unfortunately, the assumption that the echo is significantly weaker than the speech is rarely true in acoustic echo cancellation scenarios. In fact, it is common that the echo is between 20 - 30 dB stronger than the speech. In such scenarios *echo cancellation*, based on adaptive filter theory [7, 9], is necessary to allow a conversation.

A brief description of the audio transmission path and echoes in a typical telephone call

Figure 1 shows a simple audio transmission path scheme of a typical telephone call between a handset telephone and a conference phone. The speech of the B-side, denoted the *far-end talk*, is picked up by the microphone of the B-side handset and transmitted over the subscriber line to the hybrid. The signal is then transmitted to the *near-end side* (the conference phone) over the communication network. Finally, the signal is presented on the near-end side loudspeaker. As can be seen in the figure, there are two echo sources in this signal path; the hybrid at the far-end subscriber line and the hybrid inside the conference phone. To prevent these echoes from being heard on the conference phone loudspeaker, echo cancellation blocks for removing the echoes are used, as can be seen in the figure.

In the opposite direction, A-side speech as well as acoustic echo from the loudspeaker is picked up by the microphone and passed through the acoustic echo canceller inside the conference phone which removes the echo. The speech is then sent over the communication network to the far-end side and presented on the loudspeaker of the handset telephone. In this signal path there are two echo sources; the conference telephone loudspeaker at the near-end side and the hybrid at the near-end subscriber line. There are also corresponding echo cancellers.

The most significant differences between line-/network echoes and acoustic echoes are that the amount of returning line-/network echo is limited by

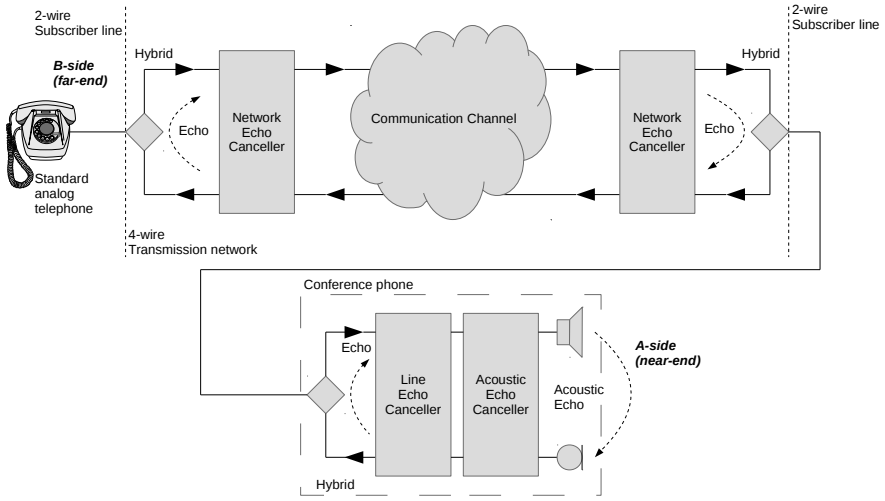

Figure 1: Scheme illustrating audio signal flow and echoes in an audio conferencing setup.

regulations and recommendations [10] and that the transfer function of the line-/network echo is sparse, while the acoustic echo typically has a non-sparse transfer function with an exponentially decaying envelope [11]. Moreover, acoustic echo cancellation in most cases requires longer filters and additional control mechanisms due to poor speech-to-echo ratio as compared to line-/network echo cancellation, and is generally considered a more difficult problem. In the following, a more detailed description of acoustic echo cancellation is presented. However, the same fundamental adaptive filtering principles apply for both types of echo cancellers.

Acoustic echo cancellation

First, consider a digital signal $x(k)$ sampled at 8 kHz, where k is the sample index, passed to a loudspeaker, as shown in figure 2. Sound waves emitted by the loudspeaker propagate in the room and are attenuated and reflected by the air itself as well as by walls, furniture and objects. A simple linear model of the acoustic echo can then be formed as a sum of more or less attenuated and time-delayed versions of the loudspeaker signal. This model is named

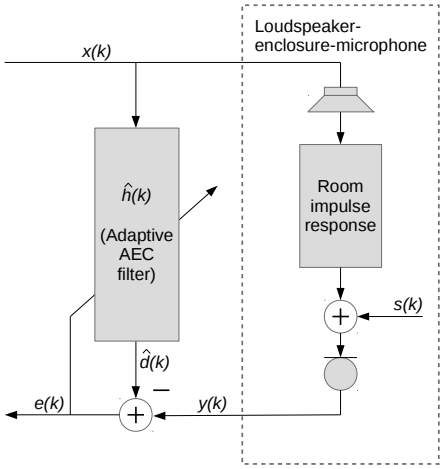

Figure 2: Acoustic echo cancellation using adaptive filtering.

the *room impulse response* in figure 2. It should be noted that this impulse response can change significantly due to minor changes in the room such as e.g. doors opening or closing or even temperature variations in the air. Such a change is called an *echo path change* and is discussed in a later section. For the following discussion, however, it is assumed that the room impulse response is stationary.

Also present on the microphone is near-end speech and noise, denoted $s(k)$ in the figure. The digital microphone signal can then be expressed as

$$y(k) = s(k) + \sum_{i=0}^{\infty} h_i x(k-i), \tag{1}$$

where h_i represents the attenuation of the loudspeaker signal reflection as received on the microphone after i samples. Plotting estimates of h_i measured in a room against the parameter i typically gives a result similar to what is shown in figure 3, i.e. an estimate of the room impulse response. What is visualized in the plot of figure 3 is first the intrinsic delay in the *loudspeaker-enclosure-microphone* (LEM) system, resulting in the amplitude being approximately zero for the first 100 samples, i.e. $h_i \approx 0$ for $i < 100$. This is followed by

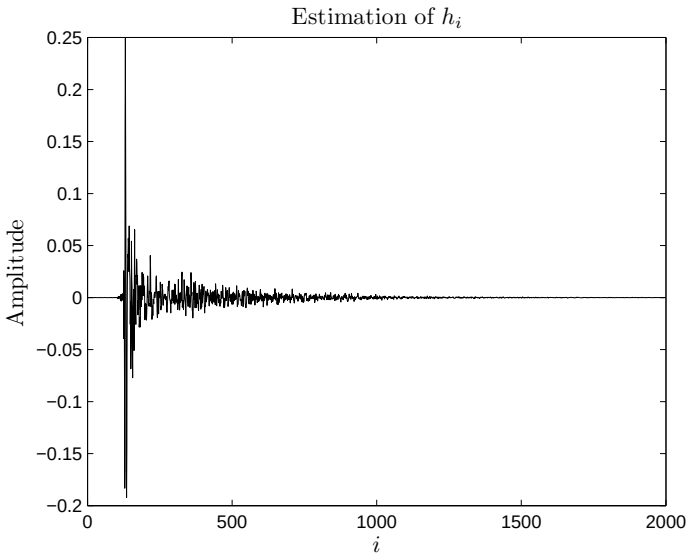

Figure 3: Estimated impulse response (transfer function) of a typical room.

a few samples of large magnitude representing sound traveling straight from the loudspeaker to the microphone without, or with just a few, reflections. Then as i increases, the sound reaching the microphone is more and more attenuated.

The fundamental idea of the echo cancellation approach is to generate a replica of the estimated echo, $\hat{d}(k)$, which is then subtracted from the microphone signal, as opposed to the echo suppression approach in which the microphone signal is multiplied by a gain factor. By subtracting the estimated echo, the near-end speech $s(k)$ component of the microphone signal can remain virtually unaffected if the echo replica is an accurate estimation of the echo component of the microphone signal. The echo canceller generates the echo replica as

$$\hat{d}(k) = \sum_{i=0}^{N-1} \hat{h}_i x(k-i), \tag{2}$$

where N is the model order and $\hat{h}_i$ is an estimate of h_i. This echo replica is then subtracted from the microphone signal to form an echo-cancelled micro-

phone signal

$$e(k) = y(k) - \hat{d}(k). \tag{3}$$

To make the equations more compact, a vector notation is typically used. In this case, the vectors $\mathbf{x}(k) = [x(k), x(k-1), \cdots, x(k-N+1)]^T$, $\hat{\mathbf{h}}(k) = [\hat{h}_0, \hat{h}_1, \cdots, \hat{h}_{k-N+1}]^T$ and $\mathbf{h} = [h_0, h_1, \cdots, h_{N-1}]^T$, where $[\cdot]^T$ denotes vector transpose and assuming that $|h_i| \quad \forall i \geq N$ is small enough to neglect, can be used to combine and rewrite equations (1), (2) and (3) as

$$\begin{aligned} e(k) &= \mathbf{h}^T\mathbf{x}(k) + s(k) - \hat{\mathbf{h}}^T(k)\mathbf{x}(k) \\ &= (\mathbf{h} - \hat{\mathbf{h}}(k))^T\mathbf{x}(k) + s(k). \end{aligned} \tag{4}$$

From equation (4) it can clearly be seen that if $\hat{\mathbf{h}}(k) \approx \mathbf{h}$, the first term will be ≈ 0, i.e. the echo will be cancelled, and $e(k) \approx s(k)$, i.e. the near-end speech will be virtually unaffected. This will (at least in theory) allow both parties to speak simultaneously without any attenuation, a so-called *full-duplex* solution.

Regarding the model order (adaptive filter length) N; a too short filter will obviously not be able to model the full echo path of the setup, resulting in poor cancellation performance. On the other hand, a too long filter uses an unnecessary amount of memory for storing the filter coefficients as well as computational resources, which could perhaps be better used for something else. It is also well known that a long filter converges slower than a short one [12]. In practice, N is often set as large as allowed by the given memory and computational resources, or set adaptively using a variable filter-length algorithm [13, 14]. In part V of this thesis, a variable filter-length algorithm is proposed and evaluated.

Now, what remains is the actual adaptation of the adaptive filter $\hat{\mathbf{h}}(k)$, so that $\hat{\mathbf{h}}(k) \approx \mathbf{h}$ can be achieved. This adaptation process is typically recursive in order to minimize the computational complexity. Several different filter adaptation algorithms have been presented, whereof the normalized least mean square (NLMS) [15, 16, 17, 18] is one of the most popular owing to its ease of implementation, low computational complexity and robustness to fix-point implementation issues. The NLMS update equation can be derived using the principle of minimum disturbance [18], i.e. from one iteration to the next, the weight vector of an adaptive filter should be changed in a minimal manner, subject to a constraint imposed on the updated filter's input. Expressed analytically, this means that

$$\min_{\hat{\mathbf{h}}(k+1)} ||\hat{\mathbf{h}}(k) - \hat{\mathbf{h}}(k+1)||^2 \tag{5}$$

subject to

$$\hat{\mathbf{h}}^T(k+1)\mathbf{x}(k) = y(k), \tag{6}$$

assuming that $s(k) \approx 0$. This optimization problem can be solved using the method of Lagrange multipliers. The Lagrange function is set as

$$\Lambda(\hat{\mathbf{h}}(k+1), \lambda) = ||\hat{\mathbf{h}}(k) - \hat{\mathbf{h}}(k+1)||^2 + \lambda(\hat{\mathbf{h}}^T(k+1)\mathbf{x}(k) - y(k)), \tag{7}$$

and calculating the derivatives with respect to $\hat{\mathbf{h}}(k+1)$ and λ gives

$$\begin{aligned} \frac{\partial\Lambda(\hat{\mathbf{h}}(k+1), \lambda)}{\partial\lambda} &= \hat{\mathbf{h}}^T(k+1)\mathbf{x}(k) - y(k), \\ \frac{\partial\Lambda(\hat{\mathbf{h}}(k+1), \lambda)}{\partial\hat{\mathbf{h}}(k+1)} &= 2\hat{\mathbf{h}}(k+1) - 2\hat{\mathbf{h}}(k) + \lambda\mathbf{x}(k). \end{aligned} \tag{8}$$

Setting both derivatives in equation (8) equal to 0 and solving for first $\hat{\mathbf{h}}(k+1)$ and then λ gives the well-known NLMS update equation [18]

$$\hat{\mathbf{h}}(k+1) = \hat{\mathbf{h}}(k) + \mu\frac{e(k)\mathbf{x}(k)}{\mathbf{x}^T(k)\mathbf{x}(k)}, \tag{9}$$

where a normalized step-size parameter $0 < \mu < 1$ has been added for controlling the adaptation. For small μ the adaptation is slow but robust to disturbances, and for μ close to 1 the adaptation is fast but sensitive to disturbances. Control of this parameter is discussed in a following section.

From a geometric perspective, the updating of the adaptive filter can be seen as moving from one point to another in an N-dimensional space. In the case of the NLMS, a filter update constitutes a movement along the *regression vector* $\mathbf{x}(k)$. Moreover, in the NLMS updating case each update is independent, meaning that movement in the N-dimensional space is far from optimal, especially for highly colored input signals where the regression vectors used for different updates are almost parallel. A more efficient adaptive filtering method in terms of convergence is recursive least squares (RLS) [18], which minimizes a weighted sum of the square of all output errors, as opposed to the NLMS which minimizes the expected value of the current squared error. In a sense, the RLS depends on the signals themselves, whereas the NLMS depends on their statistics. The RLS provides a much faster convergence rate than the NLMS, but at the cost of much higher computational complexity and sensitivity to round off errors occurring in fix-point implementations. An intermediate solution, in terms of both convergence speed and computational

complexity, is the affine projection (AP) algorithm [19]. A fast implementation of the AP algorithm called fast affine projection [20] has also been presented, reducing the computational complexity almost to that of the NLMS, except for a matrix inversion.

A family of *proportionate* type adaptive filtering algorithms have also been proposed, targeted for systems with sparse impulse response, i.e. mainly for line- and network echo cancellation. The main idea is to distribute the available adaptation "energy" unevenly among the filter coefficients [21], aiming to concentrate the adaptation to filter coefficients that benefit most from the update. A number of proportionate type approaches have been proposed, targeted for NLMS [21, 22] as well as AP [23].

Controlling the adaptive filtering process

In the previous section, it was assumed that $s(k) \approx 0$ during the adaptation of the filter, i.e. that there is virtually no near-end speech or noise present on the microphone. In case of significant near-end disturbance during filter adaptation, the filter runs the risk of diverging. To avoid this problem, the normalized step-size parameter μ can be used for controlling the adaptation speed. In practical acoustic echo cancellation applications, two mechanisms are normally used:

- Regulation of μ based on the amount of echo in relation to the stationary noise level on the microphone [24, 25, 26]. In practice this is fairly uncomplicated to achieve since the stationary noise level can be estimated when the loudspeaker is silent (during far-end speech pauses). The advantage over using a fixed step-size parameter is that the adaptation can be allowed to be fast when the echo is strong due to a severely misaligned filter, and then reduced as the filter converges and the echo approaches the stationary noise level.

- A mechanism for detecting non-stationary disturbances such as near-end speech, i.e. a *double-talk detector*, which completely halts the adaptation by setting $\mu = 0$ in case of such disturbances, in order to prevent divergence [27, 28, 29, 30].

One of the most basic double-talk detectors is the Geigel detector [27], which compares the loudspeaker and microphone energies. If the energy picked up by the microphone is larger than the energy going out to the loudspeaker, the extra microphone energy must come from a near-end talker in

the room, hence a double-talk situation is detected and the adaptation of the filter is halted. Other, more recent approaches to the double-talk detection problem have been to use e.g. power comparison using cepstral techniques [24] and coherence and cross-correlation-based approaches [28, 29].

It is of utmost importance that the double-talk detector functions as intended in order to achieve high audio quality. If the double-talk detector is configured to be too sensitive, halting of the filter adaptation could occur in situations where the acoustic environment changes abruptly (e.g. movement of the loudspeaker and/or the microphone), i.e. in situations where adaptation is most needed, the so-called *dead-lock problem*. On the other hand, if the double-talk detector is not sensitive enough, near-end speech might not be detected in some situations which could lead to poor cancellation performance and possibly even to divergence of the adaptive filter.

Two-path echo cancellation

To avoid the dead-lock problem discussed in the previous section, the *two-path* echo cancellation approach has been presented [31, 32, 33]. The basic idea, illustrated in figure 4, is to have two echo cancellation filters. One of the filters is denoted the *background filter* and is continuously updated, even during double-talk. The other filter is denoted the *foreground filter* and is fixed and produces the echo-cancelled output. A control mechanism, the *transfer logic*, determines when the background filter is better adjusted to the room impulse response, and in such an event the background filter coefficients are copied into the foreground filter. Owing to this structure, the dead-lock problem is prevented at the cost of additional complexity in the form of an additional filtering and the transfer logic.

The transfer logic constitutes a set of conditions that have to be satisfied before copying the filter coefficients. One common condition is that the magnitude of the output error from the background filter must be less than that from the foreground filter. However, in some double-talk situations, the adaptive filter can actually cancel a minor part of the near-end speech [34, 33], causing the transfer logic to erroneously classify the background filter as better adjusted to the echo path than the foreground filter. An approach to reduce this problem, the use of so-called *delayed filtering*, is presented in parts I-IV of this thesis.

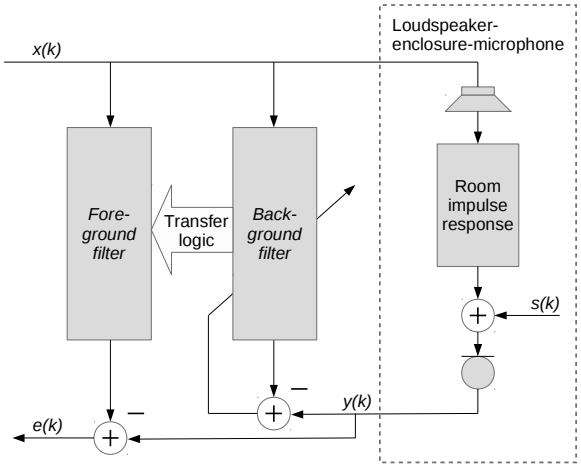

Figure 4: Two-path acoustic echo cancellation scheme.

Residual echo suppression

Unfortunately, in many situations the echo canceller does not completely remove the echo. For acoustic echo cancellation typically 20 - 30 dB of the echo can be cancelled, while up to 50 - 60 dB might be required to remove in order to avoid audible artifacts [11]. The reason why the echo canceller does not completely remove the echo is due to non-linearities in the echo path such as loudspeaker distortion, enclosure vibrations [35], or due to the fact that the adaptive filter might not have had sufficient time to adapt. Thus, in practice additional control of the residual echo (the remaining echo after echo cancellation) is required. One basic approach is to estimate the the amount of residual echo in different frequency subbands and use Wiener-filtering to remove the residual echo [36]. Other, more sophisticated approaches, are based on the psychoacoustic properties of human hearing [37, 38].

Computational complexity reduction

In almost all practical echo cancellation implementations computational complexity is an important factor to take into account in the development phase. For example, in a consumer electronic device such as a conference telephone

it is desired to have as inexpensive components as possible and the use of inexpensive components inevitably implies a digital signal processor (DSP)/central processing unit (CPU) with limited computational resources. Hence, there is a need for echo cancellation algorithms requiring low computational resources.

Perhaps the most common approach to reduce the computational complexity of echo cancellation implementations is to use a subband approach [11], where the signals are passed through a filterbank employing downsampling. An adaptive echo cancellation filter is used in each subband. The key factor for complexity reduction in this case is the downsampling, resulting in shorter filters not updating as often as their traditional long fullband counterpart. The major downside of straight-forward subband adaptive filtering is the delay introduced by the analysis and synthesis filterbanks [11]. A low delay is important for many reasons, e.g. as discussed earlier long delays mean that each participant will have to wait long for a response, which significantly increases the number of double-talk occurrences and reduces the comfort as the speech cannot flow naturally. Moreover, since a longer echo delay requires more attenuation for a maintained level of acceptance [39], the duplex will be reduced due to the increased level of residual echo suppression that is required. A solution to avoid delay introduced by straight-forward subband adaptive filtering is delayless subband adaptive filtering [40, 41], where the adaptive subband filters at regular intervals are merged together to form a fullband filter which in turn produces a delayless echo-cancelled output.

Another method for computational complexity reduction is partial- or selective updating, where only a subsection of the adaptive filter is updated at each instant. A trivial approach is *periodic updating* [42], where the updating of the adaptive filter is restricted to every M:th sample. A similar approach is *partial updating*, where only a part of all N filter coefficients are updated at each instant. Several methods for choosing which coefficients to update at a specific instant have been proposed [43, 44].

An approach combining delayless subband adaptive filtering with an efficient partial updating scheme is presented in part VI of this thesis.

Thesis summary

This doctoral thesis consists of six parts. Part I describes an adaptive filter deviation measure for two-path echo cancellation. This deviation measure is evaluated more thoroughly in part II, where experiments with a wide range of signals and parameter settings are carried out. Part III uses the same basic idea of the approach in part I and II for double-talk detection. In part IV, the adaptive filter deviation measure in part I and the double-talk detector in part III are combined into a complete transfer logic scheme for two-path echo cancellation.

Part V presents a method to adaptively determine the number of adaptive filter coefficients required in an acoustic echo cancellation setup, while part VI presents a subband-based low-complexity approach to echo cancellation where only one subband filter is updated at each instant.

Part I — An Improved Deviation Measure for Two-Path Echo Cancellation

A vital part of the two-path echo cancellation scheme is the estimation of the adaptive filter misalignment. Traditionally, this is done simply by observing the magnitude of the output error. However, the magnitude of the output error does not completely reflect the filter misalignment due to the problem of near-end signal cancellation. This part presents an improved deviation measure by introducing a time-lag so that the near-end signal disturbance is reduced. The advantages of the proposed approach are shown both analytically and through simulations.

Part II — Evaluation of an Improved Deviation Measure for Two-Path Echo Cancellation

Two important parameters to consider when using the approach described in part I are the time-lag (delay) parameter and the step-size parameter of the adaptive filter. In this part, extensive simulations are performed for different parameter settings in order to study how different settings affect the performance of the deviation measure in practice. It is shown that a negative time-lag consistently, regardless of the step-size parameter setting, seems to give better performance than without any lag. The advantage in performance is reduced with a reduced step-size parameter setting.

Part III — A Delay-Based Double-Talk Detector

In this part, the time-lag approach ("delayed filtering") used in parts I and II is utilized for normalized cross-correlation-based double-talk detection. It is first shown that having a fixed echo cancellation filter, which is a common approach in objective evaluation techniques for double-talk detectors, gives significantly different results compared to a more realistic approach with an adaptive echo cancellation filter. Then, realistic simulations with an adaptive echo cancellation filter are performed and comparisons of the proposed approach with two other normalized cross-correlation-based double-talk detectors are made. Experiments are also carried out with real recorded signals.

Part IV — Robust Low-Complexity Transfer Logic for Two-Path Echo Cancellation

For a complete two-path echo cancellation approach, a set of rules for determining how to transfer the filter coefficients between the two filters (transfer logic) is required. Part IV combines the deviation measure in part I and the double-talk detector in part III, together with step-size control and a polyphase subband structure into a full two-path transfer logic scheme. Extensive simulations show that the proposed transfer logic is more robust to double-talk than the conventional method, while also exhibiting slightly improved performance during a change of the echo-path.

Part V — Adaptive Filter Length Selection for Acoustic Echo Cancellation

In an acoustic echo cancellation system, the order (length) of the adaptive filter will significantly affect the echo cancellation performance. A short filter will adapt more quickly, but perhaps not fully cancel the echo due to insufficient length. A long filter, on the other hand, will adapt slower and can cause additional echo due to mismatch of superfluous coefficients. Furthermore, different types of rooms require different filter lengths for optimal acoustic echo cancellation performance.

This part presents an approach for adaptively adjusting the length of the echo cancellation filter. Off-line calculations using recorded speech signals show the behavior in real situations and a comparison with another state-of-the-art variable filter-length algorithm shows the advantages of the proposed method.

Part VI — A Low-Complexity Delayless Selective Subband Adaptive Filtering Algorithm

Subband adaptive filtering is a method for reduced complexity and improved narrowband signal robustness as compared to traditional fullband adaptive filtering. However, the downside of subband methods is the signal delay introduced by the filterbanks. A solution to this problem is *delayless* subband adaptive filtering, where the individual subband adaptive filters are used to construct a fullband filter providing a delayless output. The downside is that the computational cost of constructing the fullband filter is substantial. For further reduction of the computational cost, part VI presents a procedure where only one adaptive subband filter is updated at each instant. This by itself results in lower computational cost, but also allows modification of the fullband filter construction, resulting in further computational complexity reduction.

Bibliography

[1] *Scientific American*, New York, October 6, 1877.

[2] K. V. Tahvanainen, "A revolutionary speaker phone," *The history of Ericsson*, `http://www.ericssonhistory.com/templates/Ericsson/Article.aspx?id=2095&ArticleID=1369&CatID=360&epslanguage=EN`, Accessed 8th August 2012.

[3] W. F. Clemency, F. F. Romanow, A. F. Rose, "The Bell System speakerphone," *AIEE Transactions*, vol. 76(I), pp. 148-153, 1957.

[4] M. M. Sondhi, D. A. Berkley, "Silencing echoes on the telephone network," *Proceedings of the IEEE*, vol. 68, no. 8, August 1980.

[5] M. M. Sondhi, "The history of echo cancellation," *IEEE Signal Processing Magazine*, vol. 23, no. 5, pp. 95-102, September 2006.

[6] A. B. Clark, R. C. Mathes, "Echo suppressors for long telephone circuits," *Transactions of the American Institute of Electrical Engineers*, vol. 44, pp. 481-490, April 1925.

[7] B. Widrow, M. E. Hoff. "Adaptive switching circuits," *In IRE WESCON Convention Record*, vol. 4, pp. 96-104, 1960.

[8] O. M. M. Mitchell, D. A. Berkley, "A full-duplex echo suppressor using center clipping," *The Bell System Technical Journal*, vol. 50, no. 5, pp. 1619-1630, May-June 1971.

[9] M. M. Sondhi, "An adaptive echo canceler," *The Bell System Technical Journal*, vol. 46, pp. 497-510, March 1967.

[10] "G.168 Digital network echo cancellers," *ITU-T Recommendation*, ITU-T, 2002.

[11] E. Hänsler, G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach*, Wiley, 2004.

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

[13] T. Usagawa, H. Matsuo, Y. Morita, M. Ebata, "A new adaptive algorithm focused on the convergence characteristics by colored input signal: variable tap length LMS," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. EA75-A, no. 11, pp. 1493-1499, 1992.

[14] Y. Gong, C. F. N. Cowan, "An LMS style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400-2407, 2005.

[15] J. I. Nagumo, A. Noda, "A learning method for system identification," *IEEE Transactions on Automatic Control*, vol. AC-12, pp. 282-287, 1967.

[16] A. E. Albert, L. A. Gardner, *Stochastic Approximation and Nonlinear Regression*, MIT Press, Cambridge, MA, 1967.

[17] R. R. Bitmead, B. D. O. Anderson, "Performance of adaptive estimation algorithms in dependent random environments," *IEEE Transactions on Automatic Control*, vol. AC-25, pp. 788-794, 1980.

[18] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, 4th edition, 2002.

[19] K. Ozeki, T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electronics and Communication in Japan*, vol. 67-A, pp. 126-132, 1984.

[20] S. L. Gay, S. Tavathia, "The fast affine projection algorithm," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 3023-3026, May 1995.

[21] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 5, September 2000.

[22] J. Benesty, S. L. Gay, "An improved PNLMS algorithm," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 1881-1884, May 2002.

[23] T. Gänsler, J. Benesty, S. L. Gay, M. M. Sondhi, "A robust proportionate affine projection algorithm for network echo cancellation," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 793-796, June 2000.

[24] A. Mader, H. Puder, G. U. Schmidt, "Step-size control for acoustic echo cancellation filters - an overview," *Signal Processing*, vol. 80, pp. 1697-1719, September 2000.

[25] J. Benesty, H. Rey, L. R. Vega, S. Tressens, "A nonparametric VSS NLMS algorithm," *IEEE Signal Processing Letters*, vol. 13, no. 10, pp. 581-584, October 2006.

[26] T. Aboulnasr, K. Mayyas, "A robust variable step-size LMS-type algorithm: analysis and simulations," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 631-639, March 1997.

[27] D. L. Duttweiler, "A twelve-channel digital echo canceler," *IEEE Transactions on Communications*, vol. 26, pp. 647-653, May 1978.

[28] T. Gänsler, M Hansson, C.-J. Ivarsson, G. Salomonsson, "A double-talk detector based on coherence," *IEEE Transactions on Communications*, vol. 44, pp. 1421-1427, November 1996.

[29] J. Benesty, D. R. Morgan, J. H. Cho, "A new class of doubletalk detectors based on cross-correlation," *IEEE Transactions on Speech and Audio Process.*, vol. 8, pp. 168-172, March 2000.

[30] P. Åhgren, "On system identification and acoustic echo cancellation," Ph.D. dissertation, Uppsala University, 2004.

[31] K. Ochiai, T. Araseki, T. Ogihara, "Echo canceler with two echo path models," *IEEE Transactions on Communications*, vol. COM-25, no. 6, pp. 8-11, June 1977.

[32] Y. Haneda, S. Makino, J. Kojima, S. Shimauchi, "Implementation and evaluation of an acoustic echo canceller using the duo-filter control system," In *Proceedings of IWAENC International Workshop on Acoustic Echo and Noise Control*, pp. 79-82, June 1995.

[33] F. Lindstrom, C. Schüldt, I. Claesson, "An improvement of the two-path algorithm transfer logic for acoustic echo cancellation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 4, pp. 1320-1326, May 2007.

[34] F. Lindstrom, M. Dahl, I. Claesson, "The two-path algorithm for line echo cancellation," In *Proceedings of IEEE TENCON*, vol. A, pp. 637-640, November 2004.

[35] A. N. Birkett, R. A. Goubran, "Limitations of handsfree acoustic echo cancellers due to nonlinear loudspeaker distortion and enclosure vibration effects," In *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 103-106, October 1995.

[36] V. Turbin, A. Gilloire, P. Scalart, "Comparison of three post-filtering algorithms for residual acoustic echo reduction," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 307-310, May 1997.

[37] S. Gustafsson, R. Martin, P. Jax, P. Vary, "A psychoacoustic approach to combined acoustic echo cancellation and noise reduction," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 245-256, May 2002.

[38] X. Lu, B. Champagne, "A centralized acoustic echo canceller exploiting masking properties of the human ear," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 377-380, April 2003.

[39] K. Shenoi, *Digital Signal Processing in Telecommunications*, Prentice-Hall, 1995.

[40] D. R. Morgan, J. C. Thi, "A delayless subband adaptive filter architecture," *IEEE Transactions on Signal Processing*, vol. 43, no. 8, pp. 1819-1830, 1995.

[41] J. Huo, S. Nordholm and Z. Zang, "New weight transform schemes for delayless subband adaptive filtering," In *Proceedings of Global Telecommunications Conference*, vol. 1, pp. 197-201, 2001.

[42] S. S. Douglas, "Adaptive filters employing partial updates," *IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing*, vol. 44, no. 3, pp. 209-216, 1997.

[43] T. Aboulnasr, K. Mayyas, "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Transactions on Signal Processing*, vol. 47, no. 5, pp. 1421-1424, 1999.

[44] P. A. Naylor, W. Sherliker, "A short-sort M-MAX NLMS partial-update adaptive filter with applications to echo cancellation," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 373-376, 2003.

Part I

An Improved Deviation Measure for Two-Path Echo Cancellation

Part I is reprinted, with permission, from

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson, "An Improved Deviation Measure for Two-Path Echo Cancellation," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 305-308, Dallas, TX, March 2010.

An Improved Deviation Measure for Two-Path Echo Cancellation

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson

Abstract

Parallel adaptive filters have been proposed for echo cancellation to solve the dead-lock problem, occurring when the echo is detected as near-end speech after a severe echo-path change; causing the updating of the adaptive filter to halt. To control the parallel filters and monitor their performance, estimates of the filter deviation (i.e. the squared norm of the filter mismatch vector) are typically used.

This paper presents a modification of a filter mismatch estimator. The proposed modification requires slightly more computational resources than the original measure, but provides a significant improvement in terms of robustness during double-talk. This is shown both analytically and through simulations.

1 Introduction

In systems using adaptive filters for echo cancellation, it is of outmost importance to have a mechanism controlling the adaptation of the filter to avoid divergence in the case of local disturbances. Such mechanism is commonly referred to as a *double-talk detector* (DTD), with the purpose to differentiate between situations where only echo is present (single-talk) and situations where echo and local disturbances are present (double-talk). Several DTDs have been proposed, such as the Geigel detector [1] and detectors based on correlation [2] and coherence [3]. However, a problem related to all DTDs is the *dead-lock* problem occurring when the echo is detected as a local disturbance, preventing the adaptive filter from updating when it is in fact needed. This can happen after a severe change of the echo-path (i.e. an *echo-path change*). In an acoustic echo cancellation environment, an echo-path change

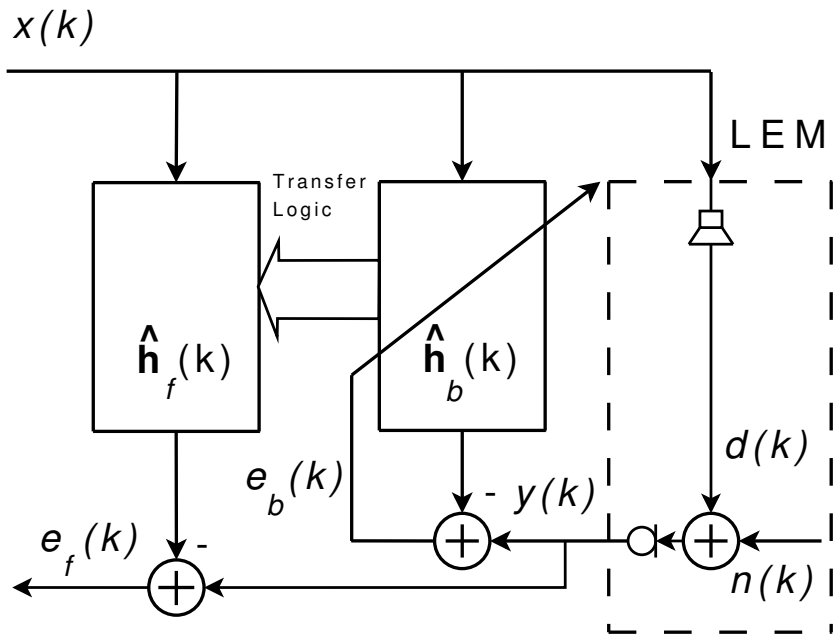

Figure 1: Scheme of the two-path algorithm in an acoustic environment.

constitutes a change in the acoustic environment such as dislocation of the loudspeaker and/or microphone or people moving in the room.

As a solution to the dead-lock problem, at the cost of an extra adaptive filter, the two-path algorithm has been proposed [4, 5, 6]. This paper presents an improved filter deviation measure intended for the two-path algorithm. The proposed filter deviation measure is compared to the one presented in [5], both analytically and through simulations. It is shown that the proposed measure has similar desirable properties as the measure in [5], while being much more robust in double-talk situations.

2 Two-path echo cancellation

A scheme illustrating the two-path echo cancellation approach in an acoustic echo cancellation environment is shown in figure 1. The updating of the background filter, $\hat{\mathbf{h}}_\mathbf{b}(k) = [\hat{h}_{b_0}(k), \hat{h}_{b_1}(k), \cdots, \hat{h}_{b_{N-1}}(k)]^T$, of length N could

be performed with a variety of algorithms, and is in this paper performed with the normalized least mean square (NLMS) [7] owing to its simplicity, according to

$$\begin{aligned} e_b(k) &= y(k) - \hat{\mathbf{h}}_\mathbf{b}(k)^T \mathbf{x}(k) \\ \hat{\mathbf{h}}_\mathbf{b}(k+1) &= \hat{\mathbf{h}}_\mathbf{b}(k) + \mu \frac{e_b(k)\mathbf{x}(k)}{\mathbf{x}(k)^T\mathbf{x}(k) + \epsilon}, \end{aligned} \tag{1}$$

where $x(k)$ is the loudspeaker signal, $y(k)$ is microphone signal, $\mathbf{x}(k) = [x(k), x(k-1), \cdots, x(k-N+1)]^T$ is the regressor vector, μ is the step-size control variable, ϵ is a regularization term to avoid division by zero and k is the sample index. $[\cdot]^T$ denotes transpose.

The foreground filter, denoted $\hat{\mathbf{h}}_\mathbf{f}(k) = [\hat{h}_{f_0}(k), \hat{h}_{f_1}(k), \cdots, \hat{h}_{f_{N-1}}(k)]^T$, gives the output error

$$e_f(k) = y(k) - \hat{\mathbf{h}}_\mathbf{f}(k)^T \mathbf{x}(k). \tag{2}$$

Updating of the foreground filter is done by copying the filter coefficients of the background filter. At which time instances this copying is performed is controlled by the transfer logic. The transfer logic is a set of conditions which should be fulfilled in order to initiate copying of the filter. Typical transfer logic conditions, in addition to trivial conditions such as sufficient loudspeaker and microphone energy, are [4, 5, 6]

1. $\frac{\sigma^2_{e_f}(k)}{\sigma^2_{e_b}(k)} > T_1$ (background filter must produce a lower output error signal than the foreground filter)

2. $\frac{\sigma^2_x(k)}{\sigma^2_{e_b}(k)} > T_2$ (acoustic coupling and echo return loss enhancement must be lower than T_2)

where T_1 and T_2 are thresholds and $\sigma^2_x(k)$, $\sigma^2_y(k)$, $\sigma^2_{e_b}(k)$, $\sigma^2_{e_f}(k)$ denote the short-time energy of the loudspeaker signal, microphone signal, background filter error signal and foreground filter error signal, respectively.

2.1 Filter deviation

During double-talk the background filter can occasionally produce lower output error than the foreground filter due to the cancellation of near-end speech [6]. This means that the first transfer logic condition presented in the previous

section is not always reliable, imposing the need of additional conditions for updating the foreground filter.

One such condition based on the filter deviation has recently been proposed [5], where the adaptive (background) filter deviation is estimated as

$$\nu_b(k) = \left| \frac{r_{e_b y}(k) - \sigma^2_{e_b}(k)}{\sigma^2_y(k) - r_{e_b y}(k)} \right| = \left| \frac{r_{\hat{y}e_b}(k)}{r_{\hat{y}y}(k)} \right|, \tag{3}$$

where $r_{e_b y}(k) = \mathbb{E}[e_b(k)y(k)]$, $r_{\hat{y}e_b}(k) = \mathbb{E}[\hat{y}(k)e_b(k)]$, $r_{\hat{y}y}(k) = \mathbb{E}[\hat{y}(k)y(k)]$, $\hat{y}(k) = \hat{\mathbf{h}}_\mathbf{b}(k)^T\mathbf{x}(k)$ and $\mathbb{E}[\cdot]$ denotes expectation (ensemble average).

The microphone signal is modeled as

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

where the unknown echo-path $\mathbf{h} = [h_1, h_2, \cdots, h_{N-1}]^T$ is of length N, i.e. same length as the adaptive filters, and $n(k)$ is near-end noise and/or speech. Using equations (4) and (1) and assuming that $x(n)$ and $n(k)$ are zero mean and uncorrelated, yields that equation (3) can be written as

$$\nu_b(k) = \left| \frac{\left(\mathbf{h} - \hat{\mathbf{h}}_\mathbf{b}(k)\right)^T \mathbf{R}_{\mathbf{xx}} \hat{\mathbf{h}}_\mathbf{b}(k) + \rho_b(k)}{\mathbf{h}^T \mathbf{R}_{\mathbf{xx}} \hat{\mathbf{h}}_\mathbf{b}(k) + \rho_b(k)} \right|, \tag{5}$$

where $\mathbf{R}_{\mathbf{xx}} = \mathbb{E}[\mathbf{x}(k)\mathbf{x}(k)^T]$ and $\rho_b(k) = \mathbb{E}[\hat{y}(k)n(k)]$. It should be noted that in [5] $\hat{y}(k)$ and $n(k)$ are assumed to be uncorrelated, leading to $\rho_b(k) = 0$. In this case equation (5) provides an estimate of the filter deviation, resulting in $\nu_b(k) \approx 0$ when $\mathbf{h} \approx \hat{\mathbf{h}}_\mathbf{b}(k)$ (i.e. when the adaptive filter is well adjusted to the echo-path) and $\nu_b(k) \gg 0$ when $\mathbf{h} \not\approx \hat{\mathbf{h}}_\mathbf{b}(k)$.

3 Proposed deviation measure

The problem with the described filter deviation estimator in equation (5) is that during double-talk, the disturbing near-end speech present in the microphone signal $y(k)$ and the adaptive filter error signal $e_b(k)$ will corrupt the filter update (see equation (1)). This means that the signal $\hat{y}(k+1)$ will indeed be correlated with $n(k)$. Thus, if $n(k)$ is non-white, which certainly is the case for speech signals, the term $\rho_b(k)$ will *not* be 0, causing the previously described deviation estimate to be inaccurate.

Because of this problem, an alternative filter deviation estimator

$$\nu_{b_D}(k) = \left| \frac{r_{\hat{y}e_{b_D}}(k)}{r_{\hat{y}y_D}(k)} \right|, \tag{6}$$

where $r_{\hat{y}e_{b_D}}(k) = \mathbb{E}[\hat{y}_D(k)e_b(k-D)]$, $r_{\hat{y}y_D}(k) = \mathbb{E}[\hat{y}_D(k)y(k-D)]$, $\hat{y}_D(k) = \hat{\mathbf{h}}_\mathbf{b}(k)^T\mathbf{x}(k-D)$ and D is a delay constant, is proposed.

Using equations (4) and (1), equation (6) can be rewritten as

$$\nu_{b_D}(k) = \left| \frac{\left(\mathbf{h} - \hat{\mathbf{h}}_\mathbf{b}(k-D)\right)^T \mathbf{R}_{\mathbf{xx_D}} \hat{\mathbf{h}}_\mathbf{b}(k) + \rho_{b_D}(k)}{\mathbf{h}^T \mathbf{R}_{\mathbf{xx_D}} \hat{\mathbf{h}}_\mathbf{b}(k) + \rho_{b_D}(k)} \right|, \tag{7}$$

where $\mathbf{R}_{\mathbf{xx_D}} = \mathbb{E}[\mathbf{x}(k-D)\mathbf{x}(k-D)^T]$ and $\rho_{b_D}(k) = \mathbb{E}[\hat{y}_D(k)n(k-D)]$. The significant difference between equations (7) and (5) lies in the terms $\rho_{b_D}(k)$ and $\rho_b(k)$. As discussed earlier, in the event of disturbing near-end speech ($n(k)$), the near-end speech will disturb the filter update, imposing correlation between $\hat{y}(k+1)$ and $n(k)$. Since the autocorrelation of a speech signal usually decrease rapidly as the lag increases [8], it is obvious that $|\rho_{b_D}(k)|$ is more likely to be lower than $|\rho_b(k)|$, resulting in a more accurate filter deviation estimator.

The extra computational cost for this is one additional filtering operation, calculating $\hat{y}_D(k)$. The constant D should be chosen as large as possible to ensure a low disturbance term $|\rho_{b_D}(k)|$. However, increasing D also increases the memory requirement for storing old samples of the signals.

4 Simulations

Speech signals sampled at 8 kHz, shown in figure 2, were used in the simulations. The echo signal $d(k)$ was generated by convolution with a $N = 2000$ coefficient impulse response measured in a normal office. A stationary noise signal $w(k) \sim \mathcal{N}(0, 10^{-6})$ was added to $s(k)$, forming the near-end disturbance signal $n(k) = s(k) + w(k)$. The final microphone signal was then calculated as $y(k) = d(k) + n(k)$. Exponential recursive weighting was used to obtain approximations of the ensemble averages used in the transfer logic conditions [5] as

$$\begin{aligned}
\hat{r}_{\hat{y}e_b}(k) &= \lambda\hat{r}_{\hat{y}e_b}(k-1) + (1-\lambda)\hat{y}(k)e_b(k) \\
\hat{r}_{\hat{y}e_f}(k) &= \lambda\hat{r}_{\hat{y}e_f}(k-1) + (1-\lambda)\hat{y}(k)e_f(k) \\
\hat{r}_{\hat{y}y}(k) &= \lambda\hat{r}_{\hat{y}y}(k-1) + (1-\lambda)\hat{y}(k)y(k) \\
\hat{r}_{\hat{y}e_{b_D}}(k) &= \lambda\hat{r}_{\hat{y}e_{b_D}}(k-1) + (1-\lambda)\hat{y}_D(k)e_b(k-D) \\
\hat{r}_{\hat{y}e_{f_D}}(k) &= \lambda\hat{r}_{\hat{y}e_{f_D}}(k-1) + (1-\lambda)\hat{y}_D(k)e_f(k-D) \\
\hat{r}_{\hat{y}y_D}(k) &= \lambda\hat{r}_{\hat{y}y_D}(k-1) + (1-\lambda)\hat{y}_D(k)y(k-D)
\end{aligned} \tag{8}$$

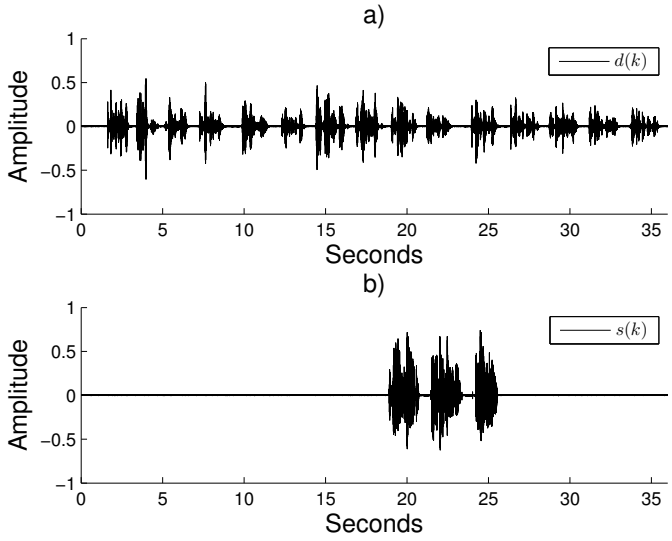

Figure 2: Signals used in the simulations. Plot (a) shows the echo signal $d(k)$ and plot (b) shows the near-end speech signal $s(k)$.

where the forgetting factor was set to $\lambda = 0.9995$.

Equations (3) and (6) can then be expressed in terms of the estimates in (8), with a regularization parameter $\epsilon_a = 0.001$ introduced to avoid division by zero, yielding the filter deviation estimators used in the simulations as

$$\begin{aligned}
\hat{\nu}_b(k) &= \left| \frac{\hat{r}_{\hat{y}e_b}(k)}{\hat{r}_{\hat{y}y}(k) + \epsilon_a} \right|, \\
\hat{\nu}_f(k) &= \left| \frac{\hat{r}_{\hat{y}e_f}(k)}{\hat{r}_{\hat{y}y}(k) + \epsilon_a} \right|, \\
\hat{\nu}_{b_D}(k) &= \left| \frac{\hat{r}_{\hat{y}e_{b\,D}}(k)}{\hat{r}_{\hat{y}y_D}(k) + \epsilon_a} \right|, \\
\hat{\nu}_{f_D}(k) &= \left| \frac{\hat{r}_{\hat{y}e_{f\,D}}(k)}{\hat{r}_{\hat{y}y_D}(k) + \epsilon_a} \right|.
\end{aligned} \tag{9}$$

The delay constant was set to $D = 32$.

Two different cases were simulated, a double-talk scenario where the microphone signal was calculated as described previously with double-talk occurring from 19 seconds to 26 seconds and an echo-path change scenario where the impulse response was changed at 19 seconds by moving all filter coefficients one step to the left (i.e. $h_{i-1} = h_i, \quad i = \{1, \cdots, N-1\}, \quad h_{N-1} = 0$). In the echo-path change scenario, no near-end speech was present (i.e. $s(k) = 0$).

To isolate the performance of the deviation measures, the two-path transfer logic was set to transfer the background filter into the foreground filter at all instances from 0 seconds up to 19 seconds and thereafter halt the update of the foreground filter. This means that the background filter and the foreground filter will be identical from 0 seconds to 19 seconds and after that the background filter will either diverge (in the double-talk scenario) or converge to the new echo-path (in the echo-path change scenario). Since the foreground filter is fixed after 19 seconds, it will either stay converged (in the double-talk scenario) or be misadjusted to the new echo-path (in the echo-path change scenario).

Figure 3 shows the results from the double-talk scenario. Plot (a) shows the output from the deviation estimator described in [5], plot (b) shows the output from the proposed deviation estimator and plot (c) shows the actual normalized deviation calculated as $\mathcal{D}_f = ||\mathbf{h} - \hat{\mathbf{h}}_\mathbf{f}(k)||_2/||\mathbf{h}||_2$ (foreground filter) and $\mathcal{D}_b = ||\mathbf{h} - \hat{\mathbf{h}}_\mathbf{b}(k)||_2/||\mathbf{h}||_2$ (background filter), respectively. Plot (d) shows a magnified version of plot (a). All plotted signals are truncated so that their maximum value cannot exceed 1 for the sake of clarity. It can be seen from plots (a) and (d) that the deviation estimator in [5] does not accurately describe the true deviation shown in plot (c) during double-talk. Further, the background filter even seems to perform better than the foreground filter occasionally, even though it is severely misadjusted. The proposed deviation estimator in plot (b), on the other hand, does not suffer from this problem and models the true deviation in plot (c) much better.

Figure 4 shows the results from the echo-path change scenario. As can be seen, the deviation estimator in [5] shown in plot (a) and the proposed deviation estimator in plot (b), both show similar behavior, i.e. both the foreground- and background filter deviation rise after the echo path change and as the background filter converges, the estimation of the background filter deviation falls.

Thus, the simulations demonstrate that the proposed deviation estimator is more robust than the deviation estimator in [5] during double-talk, while having the same desirable properties after an echo-path change.

5 Conclusions

This paper has proposed a deviation estimator targeted for two-path echo cancellation. The proposed estimator is an extension of the deviation estimator presented in [5] and is showing similar desirable properties in an echo-path change situation in addition to greatly improved robustness during double-talk. The cost of the improved robustness is increased computational complexity (an additional filtering operation). Comparisons between the original deviation estimator and the proposed estimator have been made analytically and through simulations.

References

[1] D.L. Duttweiler, "A twelve-channel digital echo canceler," *IEEE Transactions on Communications*, vol. COM-26, pp. 647–653, May 1978.

[2] J. Benesty, D.R. Morgan, and J.H. Cho, "A new class of doubletalk detectors based on cross-correlation," *IEEE Transactions on Speech and Audio Processing*, vol. 8, pp. 168–172, March 2000.

[3] T. Gansler, M. Hansson, C.-J. Ivarsson, and G. Salomonsson, "A double-talk detector based on coherence," *IEEE Transactions on Communication*, vol. 44, pp. 1421–1427, November 1996.

[4] K. Ochiai, T. Araseki, and T. Ogihara, "Echo canceler with two echo path models," *IEEE Transactions on Communications*, vol. COM-25, no. 6, pp. 8–11, June 1977.

[5] M.A. Iqbal and S.L. Grant, "Novel and efficient download test for two path echo canceller," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2007, pp. 167–170.

[6] F. Lindstrom, C. Schüldt, and I. Claesson, "An improvement of the two-path algorithm transfer logic for acoustic echo cancellation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, pp. 1320–1326, May 2007.

[7] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, 4th edition, 2002.

[8] L.R. Rabiner and R.W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, 1978.

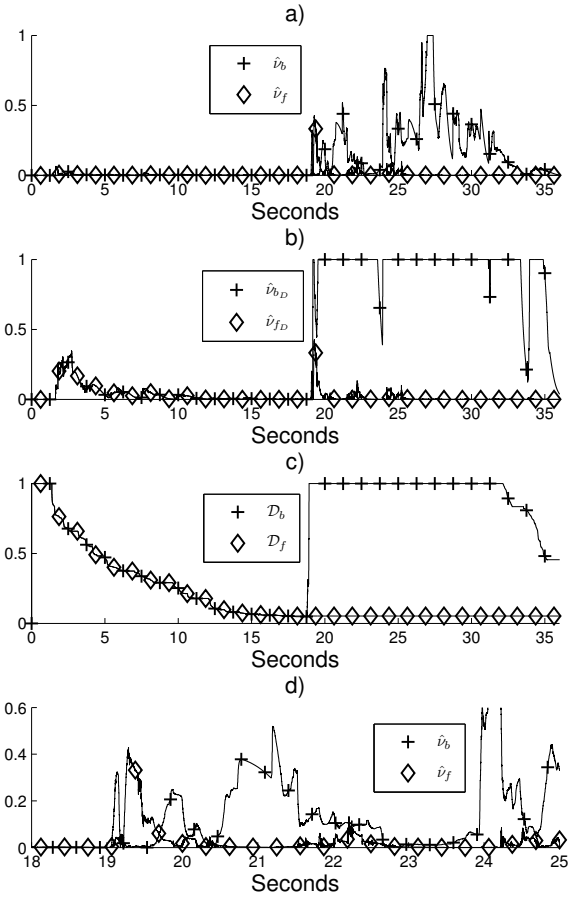

Figure 3: Adaptive filter deviation estimates in a double-talk situation. Plot (a) shows the output from the estimator used in [5], plot (b) shows the output from the proposed estimator, and plot (c) shows the true deviation, respectively. Plot (d) shows a magnified version of plot (a). By comparing plots (a), (d) and (c), it can be seen that output from the estimator in [5] is not reliable during double-talk.

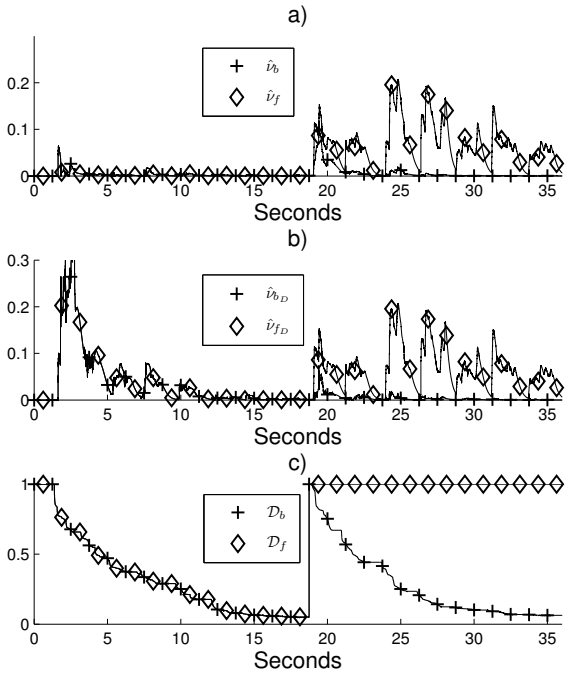

Figure 4: Adaptive filter deviation estimates in a echo-path change situation. Plot (a) shows the output from the estimator used in [5], plot (b) shows the output from the proposed estimator, plot (c) shows the true deviation, respectively. It can be seen that both estimators show similar desirable properties after an echo-path change.

Part II

Evaluation of an Improved Deviation Measure for Two-Path Echo Cancellation

Part II is reprinted, with permission, from

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson, "Evaluation of an Improved Deviation Measure for Two-Path Echo Cancellation," In *Proceedings of International Workshop on Acoustic Echo and Noise Control (IWAENC)*, Tel Aviv, Israel, September 2010.

Evaluation of an Improved Deviation Measure for Two-Path Echo Cancellation

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson

Abstract

The two-path algorithm is a well-known approach for overcoming the dead-lock problem in echo cancellation systems. Typically, a fixed foreground filter is producing the echo cancelled output while a continuously updating background filter adapts to the echo-path. When the background filter is considered to perform better than the foreground filter, the coefficients of the background filter are copied into the foreground filter. To determine which filter is better adjusted to the true echo-path, a filter deviation measure can be used.

Recently, a method which introduces a delay in the calculation of the filter deviation measure, yielding a more reliable estimate has been proposed. However, a thorough evaluation of the effect of different delay settings has not yet been performed.

Thus, in this paper a number of simulations with different delay parameter settings are carried out to show how this parameter affects the overall performance of the filter deviation measure.

1 Introduction

The use of parallel adaptive filters, the so-called *two-path approach* [1], is frequent in many adaptive filter based echo cancellation systems to overcome the dead-lock problem. The dead-lock problem occurs when a change of the echo-path is mistaken for local noise, causing the adaptive filter control to halt the updating of the adaptive filter. In the two-path approach, two parallel adaptive filters are used: a continuously updating *background filter* and a fixed *foreground filter*. The fundamental idea is to constantly compare the echo

cancellation performance of the background- and the foreground filter and to copy the background filter coefficients into the foreground filter whenever the background filter is considered to be better adjusted to the echo-path than the foreground filter.

A straight-forward approach for determining which filter is better adjusted to the echo-path is to compare the output errors from the filters [1, 2]. However, during *double-talk*, i.e. situations where both the signal driving the filter as well as the near-end signal are active simultaneously, cancellation of the near-end signal can occur, making the output error an unreliable measure for determining how well adjusted the filter is to the echo-path [3]. In [4] an alternative filter deviation measure was introduced, although this measure also suffers from problems due to cancellation of near-end speech in double-talk situations. As a solution to this problem, [5] introduced a delay in the calculation of the filter deviation measure to obtain a more reliable estimate. However, setting of the delay parameter and its effect on the performance of the algorithm was not entirely evaluated - in essence the paper just concluded that a large delay was better than no delay (i.e. the method in [4]).

The main purpose of this paper is to show how different settings of the delay parameter affects the performance of the algorithm. It is concluded that having a delay is not always better than no delay. It is also concluded that a negative delay parameter on the other hand always seem to give better performance than no delay and that the step-size parameter in the adaptive filter update strongly affects the performance of the filter deviation measure.

2 Two-path echo cancellation

In this paper, the two-path approach is considered in an acoustic echo cancellation environment where the foreground filter is denoted as $\hat{\mathbf{h}}_\mathbf{f}(k) = [\hat{h}_{f_0}(k), \hat{h}_{f_1}(k), \cdots, \hat{h}_{f_{N-1}}(k)]^T$ and the updating background filter as $\hat{\mathbf{h}}_\mathbf{b}(k) = [\hat{h}_{b_0}(k), \hat{h}_{b_1}(k), \cdots, \hat{h}_{b_{N-1}}(k)]^T$, where N is the filter length and k is the sample index. The echo cancelled signals are calculated by subtracting the filtered echo estimates from the microphone signal $y(k)$ according to

$$e_f(k) = y(k) - \hat{y}_f(k), \tag{1}$$

where $\hat{y}_f(k) = \hat{\mathbf{h}}_\mathbf{f}(k)^T \mathbf{x}(k)$ and

$$e_b(k) = y(k) - \hat{y}_b(k), \tag{2}$$

where $\hat{y}_b(k) = \hat{\mathbf{h}}_\mathbf{b}(k)^T\mathbf{x}(k)$, $e_f(k)$ is the output error from the foreground filter, $e_b(k)$ is the output from the background filter, $\mathbf{x}(k) = [x(k), x(k-1), \cdots, x(k-N+1)]^T$ is the regressor vector and $x(k)$ is the loudspeaker signal.

Updating of the adaptive filter can be achieved with a number of algorithms. In this paper, as well as in [3, 4, 5], normalized least mean square (NLMS) was used in order to simplify the analysis. Updating of the background filter is thus performed as

$$\hat{\mathbf{h}}_\mathbf{b}(k+1) = \hat{\mathbf{h}}_\mathbf{b}(k) + \mu\frac{e_b(k)\mathbf{x}(k)}{\mathbf{x}(k)^T\mathbf{x}(k)+\epsilon}, \tag{3}$$

where μ is the step-size control variable and ϵ is a regularization term to avoid division by zero [6].

The foreground filter is updated by copying the coefficients of the background filter into the foreground filter. When this copying is performed is controlled by the *transfer logic*, which essentially is a set of conditions that should be fulfilled in order to initiate a filter copying operation. Typical transfer logic conditions, in addition to trivial conditions such as sufficient loudspeaker and microphone energy, are [1, 3, 4]

- $\frac{\sigma^2_{e_f}(k)}{\sigma^2_{e_b}(k)} > T_1$ (i.e. the background filter must produce a lower output error signal than the foreground filter)
- $\frac{\sigma^2_x(k)}{\sigma^2_{e_b}(k)} > T_2$ (i.e. the acoustic coupling and echo return loss enhancement must be lower than T_2)

where T_1 and T_2 are thresholds and $\sigma^2_x(k)$, $\sigma^2_y(k)$, $\sigma^2_{e_b}(k)$, $\sigma^2_{e_f}(k)$ denote the short-time energy of the loudspeaker signal, microphone signal, background filter error signal and foreground filter error signal, respectively.

3 Filter deviation measure

The problem related to comparing output errors of the filters is that during double-talk, cancellation of near-end speech by the updating background filter can occur [3], making the output error from the background filter less than then output error from the foreground filter, even though the background filter is misadjusted due to updating during near-end speech. This means that the

first transfer logic condition in the previous section is not always reliable, imposing the need of additional conditions for updating the foreground filter.

In [5] the (background) filter deviation measure

$$\nu_{b_D}(k) = \left| \frac{r_{\hat{y}e_{b_D}}(k)}{r_{\hat{y}y_D}(k)} \right|, \tag{4}$$

where $r_{\hat{y}e_{b_D}}(k) = \mathbb{E}[\hat{y}_D(k)e_b(k-D)]$, $r_{\hat{y}y_D}(k) = \mathbb{E}[\hat{y}_D(k)y(k-D)]$, $\hat{y}_D(k) = \hat{\mathbf{h}}_\mathbf{b}(k)^T\mathbf{x}(k-D)$, D is a delay constant and $\mathbb{E}[\cdot]$ denotes expectation (ensemble average) was introduced.

Assuming that the microphone signal can be modeled as

$$y(k) = \mathbf{h}^T\mathbf{x}(k) + n(k), \tag{5}$$

where the (unknown) echo-path $\mathbf{h} = [h_1, h_2, \cdots, h_{N-1}]^T$ is of length N, i.e. same length as the adaptive filters, and $n(k)$ is near-end noise and/or speech, allows combination of equations (4), (5) and (3) as [5]

$$\nu_{b_D}(k) = \left| \frac{\left(\mathbf{h} - \hat{\mathbf{h}}_\mathbf{b}(k-D)\right)^T \mathbf{R}_{\mathbf{xx_D}} \hat{\mathbf{h}}_\mathbf{b}(k) + \rho_{b_D}(k)}{\mathbf{h}^T\mathbf{R}_{\mathbf{xx_D}}\hat{\mathbf{h}}_\mathbf{b}(k) + \rho_{b_D}(k)} \right|, \tag{6}$$

where $\mathbf{R}_{\mathbf{xx_D}} = \mathbb{E}[\mathbf{x}(k-D)\mathbf{x}(k-D)^T]$ and $\rho_{b_D}(k) = \mathbb{E}[\hat{y}_D(k)n(k-D)]$. By setting $D = 0$ one obtains the filter deviation estimate proposed in [4]. Further, in [4] it was assumed that $\rho_{b_D}(k) = 0$. In that case it can clearly be seen that $\nu_{b_D}(k) \approx 0$ if $\mathbf{h} \approx \hat{\mathbf{h}}_\mathbf{b}(k)$ (i.e. if the adaptive background filter is well adjusted to the echo-path) and $\nu_b(k) \gg 0$ if $\mathbf{h} \not\approx \hat{\mathbf{h}}_\mathbf{b}(k)$.

Thus, the resulting additional transfer logic condition is then $\nu_b(k) < \nu_f(k)$, hence the deviation measure must indicate that the background filter is better adjusted to the echo path than the foreground filter for an update of the foreground filter to occur.

However, as pointed out in [5], if $n(k)$ is non-white (i.e. $\mathbb{E}[n(k)n(k+l)] \neq 0 \quad \forall l \neq 0$) and $D = 0$ it can be seen that $\rho_{b_D}(k) \neq 0$ by observing the adaptive filtering equations (2) and (3). Due to the characteristics of speech, it was argued in [5] that $|\rho_{b_D}(k)|$ is more likely to be lower for $D \neq 0$ than for $D = 0$, resulting in a more accurate filter deviation estimator.

However, in the simulations in [5] the variable D was simply set to $D = 32$ without further evaluation. Because of this, the following sections will show how different settings of the delay parameter D affects the performance of the described filter deviation measure.

It should be noted that although $D < 0$ indicate non-causality, delaying the respective signals by D before performing the filtering and adaptive filter updating will have the same effect and also allow real-time implementation.

4 Simulations

To evaluate the performance of the algorithm for different settings of D, two sets of speech signals sampled at 8 kHz were used. The first set consisted of a speech signal from a male speaker used as the driving loudspeaker signal $x(k)$ and a speech signal from a female speaker used as near-end speech signal $s(k)$. The second set consisted of a speech signal from a female speaker used as the driving loudspeaker signal $x(k)$ and a speech signal from a male speaker used as near-end speech signal $s(k)$. A total of 8 different signal constellations (simulation scenarios), with 36 seconds duration each, was created by using both signal sets and varying the starting time of the near-end speech to occur after either 16, 19, 21 or 24 seconds. The results presented in this paper are the average results form the 8 different simulation scenarios. In all scenarios the same impulse response, a filter $\mathbf{h}$ of length $N = 500$ measured in a normal office, was used. The driving loudspeaker signal $x(k)$ was convoluted with $\mathbf{h}$ to obtain the echo signal $d(k)$ and the simulated microphone signal was then formed by summing the signals $d(k)$, $s(k)$ as well as a local stationary noise signal $w(k) \sim \mathcal{N}(0, 10^{-6})$, i.e. $y(k) = d(k) + s(k) + w(k)$.

To isolate the performance of the deviation measure, the two-path transfer logic was in each simulation scenario set to transfer the background filter into the foreground filter at all instances from 0 seconds up to the sample index where near-end speech starts and thereafter halt the update of the foreground filter. This means that the background filter and the foreground filter will be identical up to the occurrence of near-end speech (double-talk) and then the background filter will diverge while the foreground filter stays converged. This is the same procedure as in [5].

Exponential recursive weighting was used to obtain approximations of the ensemble averages used in the deviation estimates [4] as

$$\begin{aligned} \hat{r}_{\hat{y}e_{b_D}}(k) &= \lambda \hat{r}_{\hat{y}e_{b_D}}(k-1) + (1-\lambda)\hat{y}_D(k)e_b(k-D) \\ \hat{r}_{\hat{y}e_{f_D}}(k) &= \lambda \hat{r}_{\hat{y}e_{f_D}}(k-1) + (1-\lambda)\hat{y}_D(k)e_f(k-D) \\ \hat{r}_{\hat{y}y_D}(k) &= \lambda \hat{r}_{\hat{y}y_D}(k-1) + (1-\lambda)\hat{y}_D(k)y(k-D) \end{aligned} \tag{7}$$

where the forgetting factor was set to $\lambda = 0.9995$, which is the same as in [5].

Two measures were used to evaluate the performance: *the number of errors* defined as the number of sample indices where $\nu_{b_D}(k) < \nu_{f_D}(k)$ during double-talk, i.e. the number of sample indices where the algorithm *falsely* indicates that the background filter is better adjusted to the echo-path than the foreground filter (despite having diverged due to double-talk), and *the averaged difference between the foreground- and background filter deviation measure* defined as

$$\frac{1}{N_d - p_i} \sum_{i=p_i}^{N_d} (\nu_{b_D}(i) - \nu_{f_D}(i)) \tag{8}$$

where p_i is the sample index where near-end speech (and thus double-talk) starts and N_d is the length (in samples) of the double-talk sequence. The averaged results over the 8 different simulation scenarios and for three different step-size parameter settings are shown in figures 1 and 2, respectively. Interesting to note is that the results for the individual scenarios where fairly equal, i.e. no results from a single scenario was very different from the average results.

5 Results

By observing the upper and middle plot of figure 1, representing simulations with step-size parameter $\mu = 0.95$ and $\mu = 0.5$ respectively, it can clearly be seen that for $D < 0$ the number of occasions where the deviations measure falsely indicates that the background filter is better adjusted than the foreground filter during double-talk decreases as the time-lag D decreases. Interesting to note is also that $D = 1$ gives the worst result for $\mu = 0.95$ (upper plot) and $D = 2$ gives the worst result for $\mu = 0.5$ (middle plot). Also interesting to note is that by observing the lower plot of figure 1, representing simulations with step-size parameter $\mu = 0.1$, it can be seen that the deviation measure performs significantly better for $D < 0$ than for $D > 0$ in this case. It can be speculated that this is due to the relatively slow convergence of the adaptive background filter and the properties of speech: it seems that for this step-size setting the adaptive filter captures the characteristics of the near-end speech, making $|\rho_{b_D}(k)| \gg 0$. It is clear that increasing D up to at least 64 does not improve the performance. (Of course, increasing D enough will indeed improve the performance since speech is only considered stationary up to about 20 ms [6].) For relatively large step-sizes on the other hand, it might be that the adaptive filter does not get a chance to adjust to the near-end

speech owing to fluctuation of the NLMS update vector.

Figure 2 shows the the averaged difference between the foreground- and background filter deviation measure calculated as equation (8) averaged over all 8 simulation scenarios. It can be seen that for step-size parameter $\mu = 0.95$ and $\mu = 0.5$ (upper and middle plot), increasing $|D|$ sufficiently gives an improved margin between $\nu_{b_D}(i)$ and $\nu_{b_D}(i)$ during double-talk, which is highly desirable. However, the margin seems to decrease with the step-size parameter - which is expected since an adaptive filter with a large step-size diverges more rapidly than an adaptive filter with a small step-size. Similar conclusions as for figure 1 can be drawn from figure 2, i.e. that using $D < 0$ for $\mu = 0.1$ gives significant performance improvement over using $D > 0$, while for large step-sizes it does not seem to matter as long as $|D|$ is fairly large.

Thus, the conclusion to be drawn from the simulations is that $D < 0$ always seem to give better performance than $D = 0$ and a smaller D always seem to give better performance than a larger D if D is negative, regardless of the step-size. For positive values of D however, starting with $D = 0$, the performance seem to worsen up to a point (depending on the step-size) if D increases, and then improve again if D increases further.

6 Conclusions

In [4] an adaptive filter deviation measure for use in two-path echo cancellation was proposed. This adaptive filter deviation measure was improved in [5] by introducing a delay D in the calculation. This paper has evaluated how different settings of D affects the performance in practice, using simulations with speech signals. It has been concluded that that setting $D < 0$ consistently seem to give better performance than $D = 0$ and for smaller step-sizes ($\mu < 0.5$) the sign of D is more important for the performance than for larger step-sizes. A setting of $D > 0$ could also give better performance than $D = 0$, but not necessarily, as this depends on the actual value of D and the adaptive filter step-size parameter.

Acknowledgment

The funding from the Swedish Knowledge Foundation (KKS) is gratefully acknowledged.

References

[1] K. Ochiai, T. Araseki, and T. Ogihara, "Echo canceler with two echo path models," *IEEE Transactions on Communications*, vol. COM-25, no. 6, pp. 8–11, June 1977.

[2] Y. Haneda, S. Makino, J. Kojima, and S. Shimauchi, "Implementation and evaluation of an acoustic echo canceller using the duo-filter control system," in *Proceedings of International Workshop on Acoustic Echo and Noise Control*, June 1995, pp. 79–82.

[3] F. Lindstrom, C. Schüldt, and I. Claesson, "An improvement of the two-path algorithm transfer logic for acoustic echo cancellation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, pp. 1320–1326, May 2007.

[4] M. Iqbal and S. Grant, "Novel and efficient download test for two path echo canceller," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, October 2007, pp. 167–170.

[5] C. Schüldt, F. Lindstrom, and I. Claesson, "An improved deviation measure for two-path echo cancellation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2010, pp. 305–308.

[6] E. Hänsler and G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach*. Wiley, 2004.

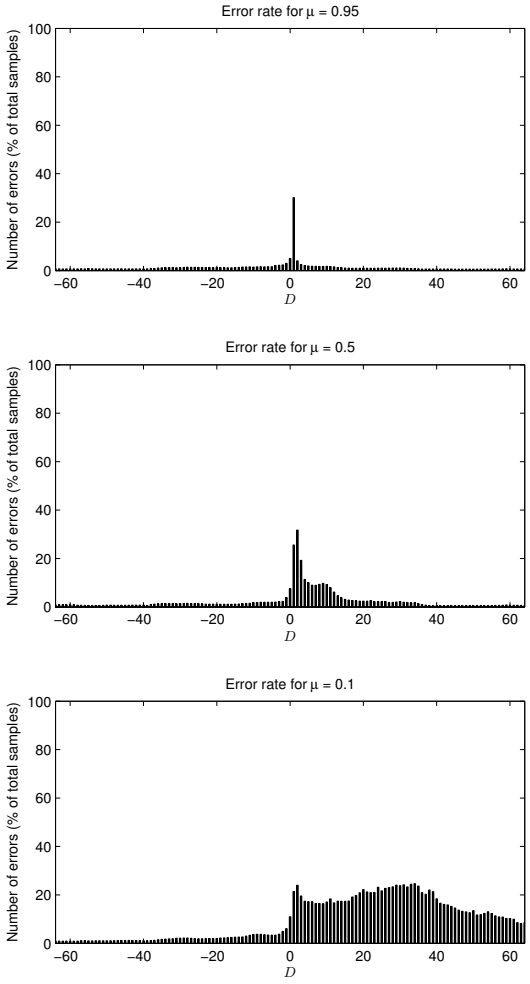

Figure 1: Number of errors, defined as the number of sample indices where $\nu_{b_D}(k) < \nu_{f_D}(k)$ during double-talk, in percent for three different step-sizes ($\mu = \{0.95, 0.5, 0.1\}$) and a number of different time-lags, i.e. settings of D. The plots show the average results of 8 different simulations.

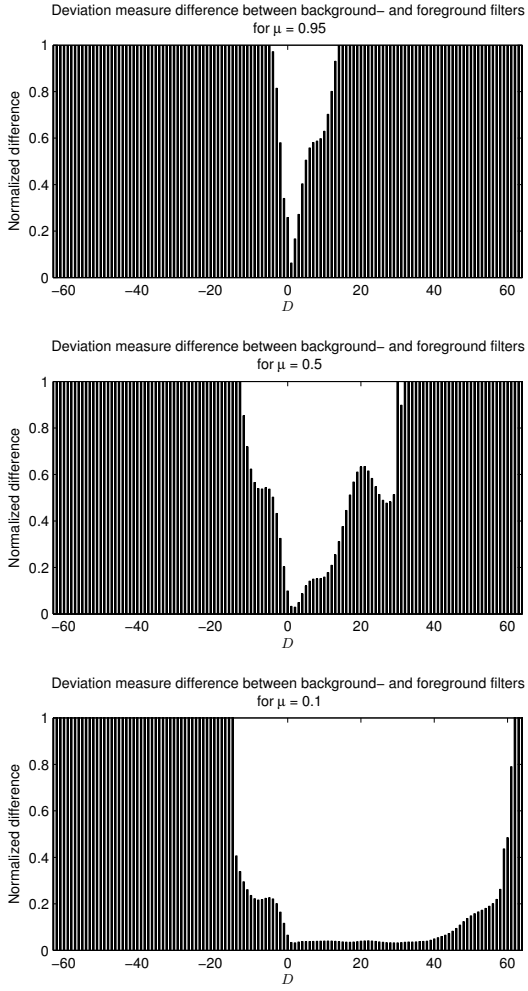

Figure 2: Deviation measure difference between background and foreground filters for three different step-sizes ($\mu = \{0.95, 0.5, 0.1\}$) and a number of different time-lags, i.e. settings of D. The plots show the average results of 8 different simulations.

Part III

A Delay-based Double-talk Detector

Part III is reprinted, with permission, from

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson, "A Delay-based Double-talk Detector," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1725-1733, February 2012.

A Delay-based Double-talk Detector

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson

Abstract

When an adaptive filter is used for echo cancellation, it is essential to prevent the filter from diverging in situations when the echo signal is contaminated with near-end disturbance, i.e. during double-talk. This paper presents an extension of a previously proposed double-talk detector for improved performance. It is shown that the computational complexity of the proposed detector is lower than that of the well-used normalized cross correlation (NCC) double-talk detector, at the cost of performance. Further, it is shown that there can be a significant performance difference, in terms of detecting double-talk, between having a fixed echo cancellation filter, which is a common strategy in objective evaluation techniques, and an adaptive filter, which is more close to realistic conditions.

1 Introduction

The purpose of an echo canceller is to remove echo of a known output (far-end) signal from an input signal. This is in practice typically achieved with an adaptive finite impulse response (FIR) filter set to model the echo path, creating a replica of the echo which is then subtracted from the input signal. To prevent the adaptive filter from diverging when local (near-end) disturbance is present, a double-talk detector (DTD) can be used.

A scheme of an adaptive echo cancellation filter controlled by a double-talk detector is shown in figure 1. In this case, $\mathbf{h} = [h_0, h_1, \cdots, h_{N-1}]^T$ is the unknown echo path and $\hat{\mathbf{h}}(k) = [\hat{h}_0(k), \hat{h}_1(k), \cdots, \hat{h}_{N-1}(k)]^T$ is the adaptive filter, both assumed, for the sake of simplicity, to be of length N and k is the sample index. Also, $\mathbf{h}$ is considered time-invariant or slowly changing for the sake of simplicity. The driving far-end signal $x(k)$ is filtered with the echo path, forming an echo which in turn is summed with local near-end noise and/or speech $v(k)$, yielding the input signal $y(k) = \mathbf{h}^T\mathbf{x}(k) + v(k)$, where

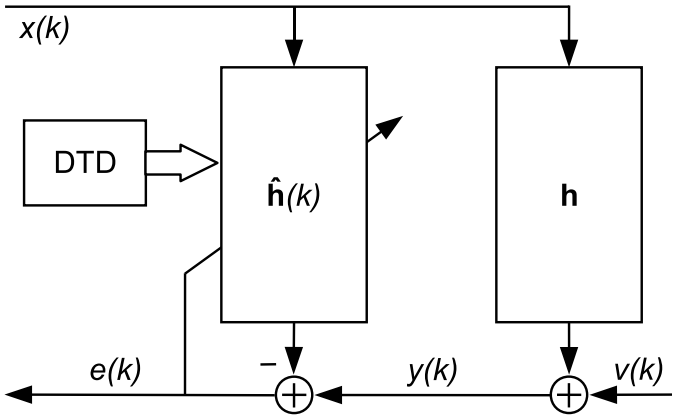

Figure 1: A generic echo canceller controlled by a DTD.

$\mathbf{x}(k) = [x(k), x(k-1), \cdots, x(k-N+1)]^T$ is the regressor vector. The echo is subtracted from the input signal $y(k)$ to obtain an echo cancelled signal

$$e(k) = y(k) - \hat{\mathbf{h}}^T(k)\mathbf{x}(k). \tag{1}$$

Updating of the adaptive filter $\hat{\mathbf{h}}(k)$ can be achieved in many ways [1]. In this paper, the normalized least mean square (NLMS) algorithm is used owing to its simplicity. An NLMS filter update is performed as

$$\hat{\mathbf{h}}(k+1) = \hat{\mathbf{h}}(k) + \mu \frac{e(k)\mathbf{x}(k)}{\mathbf{x}^T(k)\mathbf{x}(k) + \epsilon}, \tag{2}$$

where ϵ is a regularization term to avoid division by zero and μ is the step-size control parameter [1].

In the case of a significant near-end signal, $v(k)$, risking to interfere with the update of the adaptive filter, the updating of the adaptive filter should be halted to avoid filter divergence. Halting of the filter update in case of near-end disturbance is commonly handled by a DTD. Typically, the DTD calculates a detection statistic ξ, and double-talk is said to be active when ξ is lower than some threshold T. Thus, the filter is updated normally according to equation (2) when $\xi > T$ and when $\xi \leqslant T$ the update is halted.

Perhaps the most basic double-talk detector is the Geigel detector [2], which compares the far-end and the near-end signal and decides that double-talk is present when the near-end energy is larger than the far-end energy. Other, more recent approaches have been power comparison using cepstral analysis [3] as well as coherence and cross-correlation based techniques [4, 5].

It should be noted that in addition to the mentioned adaptive filter and double-talk detector, a complete echo cancellation solution typically also require components for residual echo removal, feedback estimation, estimation of filter misalignment and rescue detection to prevent the filter from longlasting misadjustment [3, 6]. The focus of this paper is however only the problem of correlation-based double-talk detection. It should also be noted that the proposed solution, and basically any type of DTD, could be used together with a parallel "two-path" adaptive filter structure [7, 3, 8] for preventing erroneous filter updating and for controlling the residual echo suppression.

The outline of the paper is as follows. In section 2 normalized cross-correlation based double-talk detection is briefly described and the fundamental problem with the so-called *MECC* detector is shown analytically. Section 3 introduces the proposed double-talk detector denoted D-MECC and discusses practical implementation issues in section 3.1 and computational complexity in section 3.2. Then, in section 4 double-talk detector evaluation is discussed, and the problem of trying to separate the DTD from the adaptive filtering algorithm is shown. In this section it is also shown by an example that the claim that the performance of the MECC DTD and the NCC DTD are exactly similar [9] does not hold when the DTD operates together with an adaptive echo cancellation filter. Simulations to compare performance of the different DTDs are then described in sections 5 and 6, showing that the detection performance of the proposed D-MECC lies between that of MECC and the more computationally demanding NCC. Finally, conclusions are presented in section 7.

2 Normalized cross-correlation based double-talk detection

The normalized cross correlation (NCC) double-talk detector, presented in [5], uses the detection statistic

$$\xi_{\mathrm{NCC}} = \sqrt{\mathbf{r}_{\mathbf{x}y}^T (\sigma_y^2 \mathbf{R}_{\mathbf{xx}})^{-1} \mathbf{r}_{\mathbf{x}y}}, \tag{3}$$

where $\mathbf{r}_{\mathbf{x}y} = \mathbb{E}\left[\mathbf{x}(k)y(k)\right]$ is the cross correlation vector between $\mathbf{x}(k)$ and $y(k)$, $\mathbf{R}_{\mathbf{xx}} = \mathbb{E}\left[\mathbf{x}(k)\mathbf{x}^T(k)\right]$ is the autocorrelation matrix of $x(k)$, $\sigma_y^2 = \mathbb{E}\left[y^2(k)\right]$ is the variance of $y(k)$ (assuming zero mean) and $\mathbb{E}\left[\cdot\right]$ denotes expected value. One of the main advantages with this detection statistic is that it achieves normalization in the sense that ξ_{NCC} is 1 when no near-end disturbance is present and $0 < \xi_{\text{NCC}} < 1$ when near-end disturbance is present.

It can be seen that $\mathbf{R}_{\mathbf{xx}}^{-\mathbf{1}}\mathbf{r}_{\mathbf{x}y} = \mathbf{h}$ and when in a converged state it is clear that $\mathbf{h} \approx \hat{\mathbf{h}}(k)$. Thus, a common way to reduce the computational complexity is to substitute $\mathbf{R}_{\mathbf{xx}}^{-\mathbf{1}}\mathbf{r}_{\mathbf{x}y}$ for $\hat{\mathbf{h}}(k)$ in equation (3) [5, 10, 11, 9]. Equation (3) can then be rewritten as

$$\xi_{\text{NCC}} \approx \sqrt{\frac{\mathbf{r}_{\mathbf{x}y}^T\hat{\mathbf{h}}(k)}{\sigma_y^2}}. \tag{4}$$

To further reduce the computational complexity, one can use the approximation [11]

$$\mathbf{r}_{\mathbf{x}y}^T\hat{\mathbf{h}}(k) \approx r_{y\hat{y}}, \tag{5}$$

where $r_{y\hat{y}} = \mathbb{E}\left[y(k)\hat{y}(k)\right]$. Using this approximation and removing the square root, one obtains the double-talk detection statistic [11, 9]

$$\xi_{\text{MECC}} = \frac{r_{y\hat{y}}}{\sigma_y^2} = 1 - \frac{r_{ye}}{\sigma_y^2}, \tag{6}$$

where $r_{ye} = \mathbb{E}\left[y(k)e(k)\right]$. In [11], this double-talk detector is denoted Cheap-NCR variant 2 (and the double-talk detector corresponding to equation (3) in this paper is denoted Cheap-NCR variant 1). The approximation in equation (5) was also used in [9], and in that paper the corresponding detector was denoted MECC. Worth noting is that no clear distinction between NCC and MECC in terms of performance is made in neither [11] nor [9]. In fact [9] even states that the performance of NCC and MECC are exactly similar.

By combining equations (1) and (6), assuming that $x(k)$ and $v(k)$ are independent and zero mean and using that $y(k) = \mathbf{h}^T\mathbf{x}(k) + v(k)$ and $\mathbf{r}_{\mathbf{x}y} = \mathbf{h}^T\mathbf{R}_{\mathbf{xx}}$, it can be seen that

$$\xi_{\text{NCC}}^2 \approx \frac{\mathbf{h}^T\mathbf{R}_{\mathbf{xx}}\hat{\mathbf{h}}(k)}{\mathbf{h}^T\mathbf{R}_{\mathbf{xx}}\mathbf{h} + \sigma_v^2}, \tag{7}$$

$$\xi_{\text{MECC}} = \frac{\mathbf{h}^T\mathbf{R}_{\mathbf{xx}}\hat{\mathbf{h}}(k) + \rho(k)}{\mathbf{h}^T\mathbf{R}_{\mathbf{xx}}\mathbf{h} + \sigma_v^2}, \tag{8}$$

where $\rho(k) = \mathbb{E}\left[\hat{\mathbf{h}}^T(k)\mathbf{x}(k)v(k)\right]$. If the adaptive filter does not update during near-end disturbance $v(k)$ it is clear that the adaptive filter coefficients and $v(k)$ are independent, and thus $\rho(k) = 0$ and $\xi^2_{\mathrm{NCC}} \approx \xi_{\mathrm{MECC}}$. Since $\mathbf{h} \approx \hat{\mathbf{h}}(k)$ when the filter is converged, it can be seen that $\xi_{\mathrm{MECC}} \approx 1$ when no near-end disturbance is present and $\xi_{\mathrm{MECC}} \ll 1$ during strong near-end noise. However, in a situation where the DTD misses the near-end disturbance and updates the filter, the performance of the MECC DTD will deteriorate due to the influence of $\rho(k)$ in equation (8).

The nature of $\rho(k)$ will entirely depend on the filter adaptation algorithm. In the case of NLMS, $\rho(k)$ can be evaluated as follows. By using the NLMS filter update equation (2) rewritten as

$$\begin{aligned}\hat{\mathbf{h}}^T(k) &= \hat{\mathbf{h}}^T(k-1) + \\ &+ \mu\left(\mathbf{h} - \hat{\mathbf{h}}(k-1)\right)^T \frac{\mathbf{x}(k-1)\mathbf{x}^T(k-1)}{\mathbf{x}^T(k-1)\mathbf{x}(k-1)} + \\ &+ \mu v(k-1)\frac{\mathbf{x}^T(k-1)}{\mathbf{x}^T(k-1)\mathbf{x}(k-1)} \end{aligned} \tag{9}$$

(assuming $\epsilon = 0$ for simplicity) to expand the expression for $\rho(k)$, one obtains

$$\begin{aligned}\rho(k) &= \mathbb{E}\left[\hat{\mathbf{h}}^T(k)\mathbf{x}(k)v(k)\right] = \\ &= \mathbb{E}\left[\hat{\mathbf{h}}^T(k-1)\mathbf{X}_1\mathbf{x}(k)v(k)\right] + \\ &+ \mu\mathbb{E}\left[v(k-1)v(k)\right]\mathbb{E}\left[\frac{\mathbf{x}^T(k-1)\mathbf{x}(k)}{\mathbf{x}^T(k-1)\mathbf{x}(k-1)}\right], \end{aligned} \tag{10}$$

where $\mathbf{X}_i = \left(\mathbf{I} - \mu\frac{\mathbf{x}(k-i)\mathbf{x}^T(k-i)}{\mathbf{x}^T(k-i)\mathbf{x}(k-i)}\right)$ and $\mathbf{I}$ is the $N \times N$ identity matrix. It should be noted that

$$\mu\mathbb{E}\left[\mathbf{h}^T\frac{\mathbf{x}(k-1)\mathbf{x}^T(k-1)}{\mathbf{x}^T(k-1)\mathbf{x}(k-1)}\mathbf{x}(k)v(k)\right] = 0, \tag{11}$$

owing to the independence and zero-mean of $x(k)$ and $v(k)$ and the fact that $\mathbf{h}$ is considered constant. Again, using the NLMS filter update equation (9)

to expand equation (10) yields the expression

$$\begin{aligned}\rho(k) &= \mathbb{E}\left[\hat{\mathbf{h}}^T(k-2)\mathbf{X}_2\mathbf{X}_1\mathbf{x}(k)v(k)\right] + \\ &+ \mu\mathbb{E}\left[v(k-2)v(k)\right]\mathbb{E}\left[\frac{\mathbf{x}^T(k-2)\mathbf{X}_1\mathbf{x}(k)}{\mathbf{x}^T(k-2)\mathbf{x}(k-2)}\right] \\ &+ \mu\mathbb{E}\left[v(k-1)v(k)\right]\mathbb{E}\left[\frac{\mathbf{x}^T(k-1)\mathbf{x}(k)}{\mathbf{x}^T(k-1)\mathbf{x}(k-1)}\right]. \qquad (12)\end{aligned}$$

It can thus be seen that continuing to expand the expression for $\rho(k)$ using the NLMS filter update equation (9) gives

$$\begin{aligned}\rho(k) &= \mathbb{E}\left[\hat{\mathbf{h}}^T(k-M)\left(\prod_{i=1}^{M}\mathbf{X}_i\right)\mathbf{x}(k)v(k)\right] + \\ &+ \mu\sum_{i=1}^{M}\mathbb{E}\left[v(k-i)v(k)\right]\times \\ &\times \mathbb{E}\left[\frac{\mathbf{x}^T(k-i)}{\mathbf{x}^T(k-i)\mathbf{x}(k-i)}\left(\prod_{j=0}^{i-1}\mathbf{X}_j\right)\mathbf{x}(k)\right], \qquad (13)\end{aligned}$$

where $\mathbf{X}_0 = \mathbf{I}$ and M is the number of expansions. By considering $k = 0$ as the starting index and thus $\hat{\mathbf{h}}(0)$ as the initial adaptive filter vector, it is clear that the first term is 0 since $x(k)$ and $v(k)$ are assumed to be independent and zero-mean. (Also, the adaptive filter typically has all coefficients set to zero initially.) The relation $k = M$ will hold for all $k > 0$, since the reference is always the chosen starting point $\rho(0) = 0$. Hence, the resulting expression becomes

$$\begin{aligned}\rho(k) &= \mu\sum_{i=1}^{k}\mathbb{E}\left[v(k-i)v(k)\right]\times \\ &\times \mathbb{E}\left[\frac{\mathbf{x}^T(k-i)}{\mathbf{x}^T(k-i)\mathbf{x}(k-i)}\left(\prod_{j=0}^{i-1}\mathbf{X}_j\right)\mathbf{x}(k)\right], \\ &k \geqslant 1. \qquad (14)\end{aligned}$$

Several conclusions can be drawn from equation (14). First of all, it is obvious that the disturbance $\rho(k)$ is directly proportional to the step-size parameter

μ. Further, if any of the signals $x(k)$ and $v(k)$ are white, then $\rho(k) = 0$. In the case of speech, the magnitude of the first factor in equation (14) is likely to decrease as i increases, since the autocorrelation of a speech signal usually decrease rapidly as the lag increases [12, 13, 14]. Further, since all eigenvalues of $\mathbf{X}_i$ are non-negative and $\leqslant 1$, the magnitude of the eigenvalues of the matrix resulting of the product $\prod_{j=0}^{i-1} \mathbf{X}_j$ are monotonically decreasing as i increases. This is intuitive since it can be argued that recent activity should have more influence on the disturbance than earlier activity.

3 Proposed double-talk detector

The detection statistic proposed in this paper is based on the same idea as in [12], where a delay is introduced to reduce the influence of near-end disturbance in a filter deviation measure, although in this paper the idea is used in a DTD context. The proposed detection statistic is

$$\xi_{\text{D-MECC}} = 1 - \frac{r_{y_D e_D}}{\sigma^2_{y_D}}, \tag{15}$$

where $r_{y_D e_D} = \mathbb{E}\left[y(k-D)e_D(k)\right]$, $\sigma^2_{y_D} = \mathbb{E}\left[y(k-D)y(k-D)\right]$ and $e_D(k) = y(k-D) - \hat{\mathbf{h}}^T(k)\mathbf{x}(k-D)$.

As in the previous section, using equations (1) and (6), equation (15) can be rewritten as

$$\xi_{\text{D-MECC}} = \frac{\mathbf{h}^T \mathbf{R}_{\mathbf{x_D x_D}} \hat{\mathbf{h}}(k) + \rho_D(k)}{\mathbf{h}^T \mathbf{R}_{\mathbf{x_D x_D}} \mathbf{h} + \sigma^2_{v_D}}, \tag{16}$$

where $\rho_D(k) = \mathbb{E}\left[\mathbf{x}^T(k-D)\hat{\mathbf{h}}(k)v(k-D)\right]$ and $\mathbf{R}_{\mathbf{x_D x_D}} = \mathbb{E}\left[\mathbf{x}(k-D)\mathbf{x}^T(k-D)\right]$. Using the same recursive approach, inserting the NLMS update equation equation (9), as previously for $\rho(k)$, an expression for $\rho_D(k)$ can be obtained as

$$\begin{aligned}
\rho_D(k) &= \mu \sum_{i=1}^{k} \mathbb{E}\left[v(k-i)v(k-D)\right] \times \\
&\times \mathbb{E}\left[\frac{\mathbf{x}^T(k-i)}{\mathbf{x}^T(k-i)\mathbf{x}(k-i)} \left(\prod_{j=0}^{i-1} \mathbf{X}_j\right) \times \right. \\
&\times \left. \mathbf{x}(k-D)\right], \quad k \geqslant 1.
\end{aligned} \tag{17}$$

It is obvious that $\rho(k) = \rho_D(k)$ for $D = 0$. As in [12], it can be argued that $|\rho_D(k)| < |\rho(k)|$ should hold in most cases since the auto-correlation for speech decreases as the lag increases. For $D < 0$ this is trivial to realize by comparing equations (14) and (17). For $D > 0$ however, as shown in [13], it does not always hold that $|\rho_D(k)| < |\rho(k)|$. For example, $D = 1$ and $\mu = 1$ will result in the first term of $\rho_D(k)$ being $\mathbb{E}\left[v^2(k-1)\right]$ which is likely to give even worse performance than $D = 0$. Nevertheless, as D increases, the disturbance term $|\rho_D(k)|$ is likely to decrease since then the largest first factor $\mathbb{E}\left[v^2(k-i)\right]$ will be multiplied with a second factor of smaller magnitude (since, as argued before, the magnitude of the eigenvalues of the matrix resulting of the product $\prod_{j=0}^{i-1} \mathbf{X}_j$ are monotonically decreasing as i increases). This agrees well with the simulated results in [13].

To illustrate the behavior of $\rho_D(k)$, simulations to estimate the components of equation (17) were conducted. The signal $v(k)$ was chosen as $v(k) = 0.9v(k-1) + w_1(k)$ where $w_1(k) \sim \mathcal{N}(0,1)$ and the signal $x(k)$ was chosen as $x(k) = 0.95x(k-1)+w_2(k)$ where $w_2(k) \sim \mathcal{N}(0,1)$. The parameters μ and N were set to 0.95 and 16, respectively. The reason for choosing a comparatively short filter length was to avoid excessive computations. Ensemble averages were taken over 10^5 runs and the results are shown in figures 2, 3 and 4. From the figures, it can clearly be seen that a low D reduces the magnitude of $\rho_D(k)$.

Thus, $\xi_{\text{D-MECC}}$ should then be a better choice for detection statistic than ξ_{MECC} for $D < 0$. A setting of $D = -32$ was chosen in this paper.

3.1 Practical considerations

In practice, $\mathbf{r}_{\mathbf{x}y}$, r_{ye} and σ_y^2 can be estimated using a running average over a time-window [5, 11] or exponential recursive weighting [9, 12] as

$$\begin{aligned} \hat{\mathbf{r}}_{\mathbf{x}y}(k) &= \lambda\hat{\mathbf{r}}_{\mathbf{x}y}(k-1) + (1-\lambda)\mathbf{x}(k)y(k), \\ \hat{r}_{ye}(k) &= \lambda\hat{r}_{ye}(k-1) + (1-\lambda)y(k)e(k), \\ \hat{\sigma}_y^2(k) &= \lambda\hat{\sigma}_y^2(k-1) + (1-\lambda)y^2(k), \end{aligned} \tag{18}$$

where $\hat{\mathbf{r}}_{\mathbf{x}y}(k)$ is to approximate $\mathbf{r}_{\mathbf{x}y}$, $\hat{r}_{ye}(k)$ is to approximate r_{ye} and $\hat{\sigma}_y^2(k)$ is to approximate σ_y^2, respectively, and λ is a forgetting factor. This is the approach used hereinafter, with a forgetting factor $\lambda = 0.995$ used for all three DTD and for all simulations. Variables used in the proposed detection statistic are estimated analogously. The variable λ determines the trade-off between sensitivity and robustness, i.e. a small forgetting factor results in

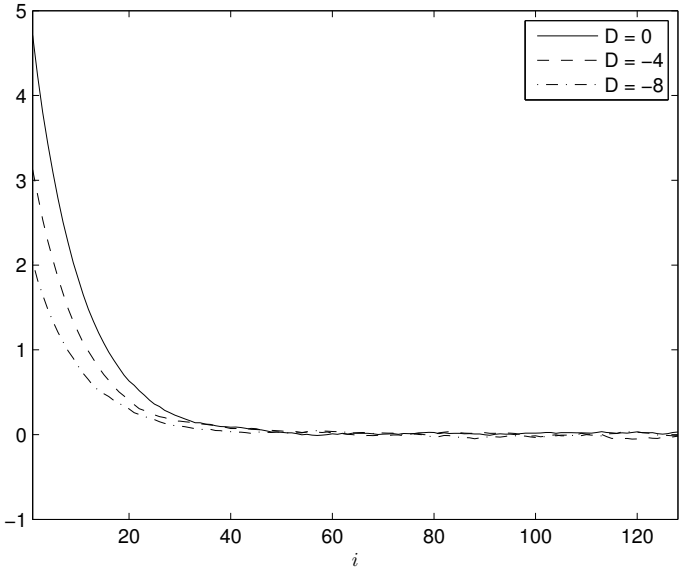

Figure 2: Estimation of $\mathbb{E}\left[v(k-i)v(k-D)\right]$ for $v(k) = 0.9v(k-1) + w_1(k)$ where $w_1(k) \sim \mathcal{N}(0,1)$, $\mu = 0.95$ and $N = 16$. Ensemble average is taken over 10^5 runs.

averages that change rapidly over time and quickly adapt to changes, while a large forgetting factor gives averages which are more consistent (robust). Hence, a small λ would imply rapid but not so accurate detection of double-talk, while a large λ would imply more accurate, but less rapid double-talk detection.

As discussed in the previous section, the proposed algorithm is based on a delay D. One approach for implementation with $D < 0$ is to introduce a delay $|D|$ in the signal path of $y(k)$, yielding a causal process. The proposed DTD then operates on the "early" signals $y(k-D)$ and $\mathbf{x}(k-D)$ preceding the delay, and the adaptive filter operates on the delayed signals $y(k)$ and $\mathbf{x}(k)$. The downside of this implementation is of course the delay introduced in the signal path, but since $|D|$ typically is comparatively small this delay is acceptable in some applications such as Voice-over-IP endpoints. If the delay

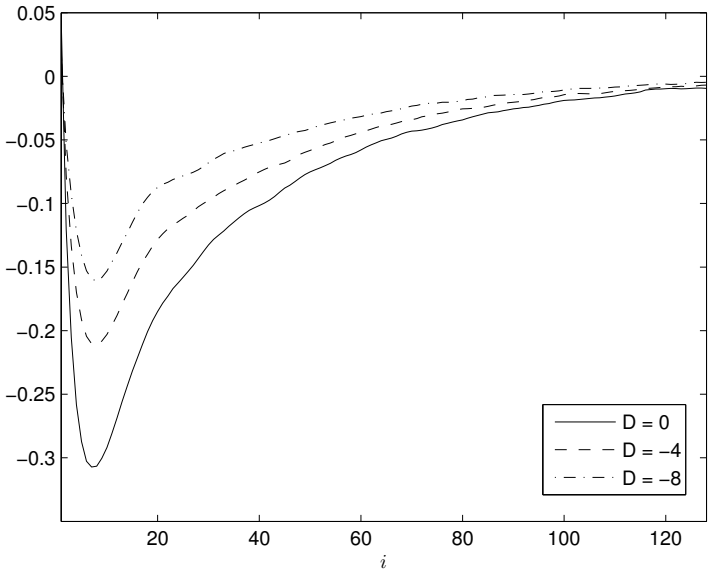

Figure 3: Estimation of $\mathbb{E}[\frac{\mathbf{x}^T(k-i)}{\mathbf{x}^T(k-i)\mathbf{x}(k-i)} \left(\prod_{j=0}^{i-1} \mathbf{X}_j\right) \mathbf{x}(k-D)]$ for $x(k) = 0.95x(k-1) + w_2(k)$ where $w_2(k) \sim \mathcal{N}(0,1)$, $\mu = 0.95$ and $N = 16$. Ensemble average is taken over 10^5 runs.

is not acceptable (for instance, the ITU-T recommendation G-168 establishes that the delay in the "receive path" should not exceed 250 μs), an alternative approach would be to store D previous versions of the adaptive filter $\hat{\mathbf{h}}(k)$. On the other hand, this significantly increases the amount of required memory.

Nevertheless, a third, much more efficient, approach is to calculate $\hat{\mathbf{h}}^T(k)\mathbf{x}(k-D)$ directly using knowledge about previous filter updates. First of all the index is changed using $n = k-D$ for the sake of clarity, yielding $\hat{\mathbf{h}}^T(n+D)\mathbf{x}(n)$. It should be kept in mind that only the case of $D < 0$ is considered here. First of all, inserting the NLMS update equation (2) into the expression for

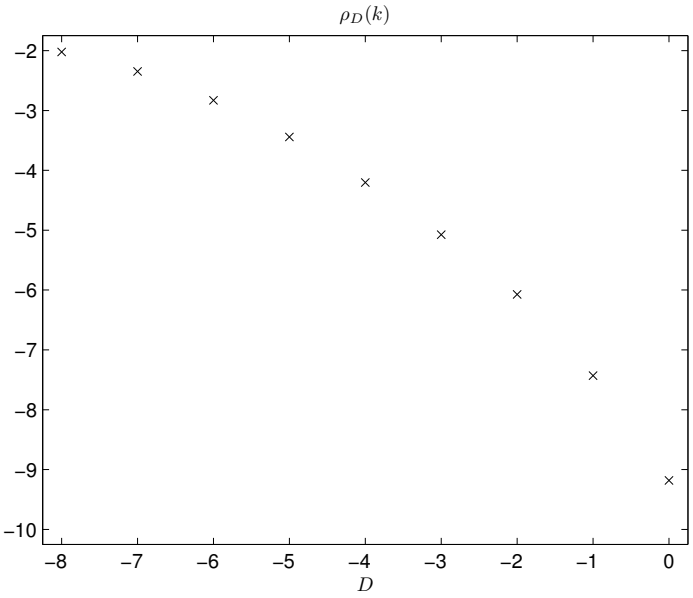

Figure 4: Estimation of $\rho_D(k)$ for $v(k) = 0.9v(k-1) + w_1(k)$ where $w_1(k) \sim \mathcal{N}(0,1)$ and $x(k) = 0.95x(k-1) + w_2(k)$ where $w_2(k) \sim \mathcal{N}(0,1)$, $\mu = 0.95$ and $N = 16$. Ensemble average is taken over 10^5 runs.

$\hat{\mathbf{h}}^T(n)\mathbf{x}(n)$ gives

$$\begin{aligned}\hat{\mathbf{h}}^T(n)\mathbf{x}(n) &= \Big(\hat{\mathbf{h}}(n-1) + \beta(n-1)\mathbf{x}(n-1)\Big)^T \mathbf{x}(n) \\ &= \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n) + \beta(n-1)\mathbf{x}^T(n-1)\mathbf{x}(n), \qquad (19)\end{aligned}$$

where $\beta(n) = \mu\frac{e(n)}{\mathbf{x}^T(n)\mathbf{x}(n)+\epsilon}$. Continuing to recursively expand equation (19) using the NLMS update equation (2) yields

$$\hat{\mathbf{h}}^T(n)\mathbf{x}(n) = \hat{\mathbf{h}}^T(n+D)\mathbf{x}(n) + \sum_{i=1}^{|D|}\beta(n-i)\alpha_i(n), \qquad (20)$$

where $\alpha_i(n) = \mathbf{x}^T(n-i)\mathbf{x}(n)$. It is thus obvious that

$$\hat{\mathbf{h}}^T(n+D)\mathbf{x}(n) = \hat{\mathbf{h}}^T(n)\mathbf{x}(n) - \boldsymbol{\beta}^T(n)\boldsymbol{\alpha}^T(n), \qquad (21)$$

where the vectors $\boldsymbol{\beta}(n) = [\beta(n-1), \beta(n-2), \cdots, \beta(n-|D|)]^T$ and $\boldsymbol{\alpha}(k) = [\alpha_1(n), \alpha_2(n), \cdots, \alpha_{|D|}(n)]^T$ are both of length $|D|$. It should be noted that $\hat{\mathbf{h}}^T(n)\mathbf{x}(n)$ is calculated in the adaptive filtering update and is thus available without additional computational cost. It should also be noted that this approach introduces no signal delay and avoids storing of previous filter coefficients.

Since the proposed DTD uses an "old" copy of the adaptive filter, $\hat{\mathbf{h}}(n+D)$, it is in a sense more sensitive to an echo path change than the related methods. This is likely to become apparent in situations with short filters, large $|D|$ and abrupt changes of the echo-path. A straight forward approach to avoid problems related to this, such as e.g. dead-lock, is to use the proposed DTD together with a parallel two-path adaptive filter structure [7, 3, 8].

3.2 Computational complexity

In terms of computational complexity, the NCC double-talk detector requires $2N + 2$ multiplications and $N + 1$ additions just for calculating $\hat{\mathbf{r}}_{\mathbf{x}y}(k)$ and $\hat{\sigma}_y^2(k)$ in equation (18). Further, an additional N multiplications and additions as well as one division are required for evaluating equation (6) (ignoring the square-root), resulting in a total of $3N + 2$ multiplications, $2N + 1$ additions and one division per evaluation, i.e. typically per input sample.

Using similar exponential recursive weighing as in equation (18), it is clear that ξ_{MECC} can be evaluated very efficiently, using a total of 4 multiplications, two additions, one subtraction and one division per evaluation.

As a comparison, D-MECC with the direct approach of storing previous filter coefficients or introducing a signal delay requires, in addition to the complexity of the MECC, N multiplications, N additions and one subtraction to calculate $e_D(k)$. On the other hand, D-MECC with the approach of calculating $\hat{\mathbf{h}}^T(k)\mathbf{x}(k-D)$ as explained in the previous section requires evaluation of equation (21) together with one subtraction (for obtaining $e_D(k)$) in addition to the complexity of the MECC.

The vector $\boldsymbol{\beta}(n)$, used in the scalar product in equation (21) can be obtained at practically no additional computational cost, since $\beta(n)$ which is calculated in the NLMS update equation (2) just has to be delayed/stored in the vector. The elements of the vector $\boldsymbol{\alpha}(k)$ can be obtained recursively at low computational cost as $\alpha_i(n) = \alpha_i(n-1) - x(n-N)x(n-N-i) + x(n)x(n-i)$, i.e. using just two multiplications, one addition and one subtraction per element. Thus, in addition to the complexity of the MECC, D-MECC with

the approach of calculating $\hat{\mathbf{h}}^T(k)\mathbf{x}(k-D)$ requires $3|D|$ multiplications and $3|D|+1$ additions/subtractions per evaluation.

4 Evaluation of double-talk detection performance

An objective technique for evaluating the performance of DTDs based on receiver operating characteristics (ROC) was presented in [10] and has since been used in numerous publications. The technique is carried out by first selecting a *probability of false alarm*, P_{f}, i.e. the probability of declaring detection when double-talk is not present, and finding appropriate detection thresholds $\{T_{\mathrm{NCC}}, T_{\mathrm{MECC}}, T_{\mathrm{D\text{-}MECC}}\}$ that correspond to the selected P_{f} setting by using speech signals where double-talk is not present. Then, simulations using speech signals with double-talk are carried out, using the respective thresholds corresponding to the chosen P_{f} setting, and the *probability of miss*, P_{m}, i.e. the probability of failing to detect double-talk when double-talk is present, is calculated. The probability of miss is evaluated over a range of near-end to acoustic echo ratios (NER) to give an indication of the performance of the double-talk detection algorithms in different situations. For a more detailed description of the evaluation technique, the reader is referred to [10].

One important aspect of the evaluation technique, as originally presented, is that the adaptive filter is assumed to be converged throughout the simulation. Thus, a fixed filter with a pre-determined misalignment at -30 dB, generated by perturbing the actual room response samples, is used [10]. Naturally, this is done to remove the dependence of the adaptive algorithm. However, as will later become apparent, this dependence can in some cases be crucial for the performance of the DTD. Thus, in those cases, using a fixed filter will not reflect the true performance of the double-talk detector in a real environment together with an actual adapting filter.

In figure 5, the detection statistics for the three considered DTDs; NCC, MECC and D-MECC in a simulation with a fixed filter, at a pre-determined misalignment at -30 dB, are shown. The far-end signal was a colored stationary signal generated as $x(n) = 0.9x(n-1)+w(n)$, where $w(n) \sim \mathcal{N}(0, 4\times10^{-4})$ and the near-end signal was a speech signal, becoming active after 16200 samples, corresponding to approximately 2 seconds with 8 kHz sample rate. As can be seen from figure 5, all three double-talk detection statistics perform

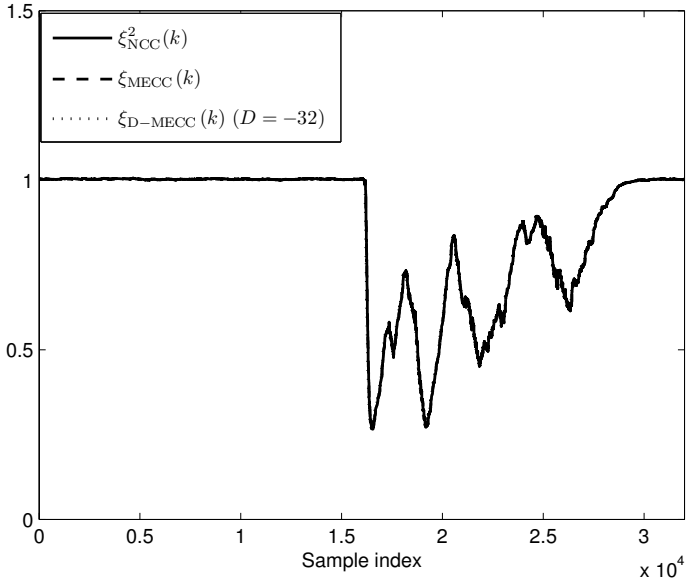

Figure 5: Comparison of DTD statistics in a situation with a fixed echo cancellation filter. Double-talk occurs after sample index 16200.

identical. This is indeed in agreement with the results of [9].

Shown in figure 6 is the resulting detection statistics from the same simulation as described above, with the single difference that a constantly updating NLMS-based adaptive filter with step-size parameter $\mu = 1$ was used instead of a fixed filter. The filter was allowed to converge to a steady-state before activating the detection statistics, and is still updating as double-talk occurs, simulating a situation where the DTD fails to detect double-talk or a parallel filter ("two-path") implementation where the DTD is coupled to the constantly updating background filter [8]. From figure 6, it can be seen that all compared detection statistics behave very differently during double-talk, when the adaptive filter is updating. Worth noting is that the detection statistic for MECC in this case is significantly worse than the NCC and D-MECC detection statistics - a threshold setting of 0.8 would in this case mean that the MECC completely misses to detect the double-talk. This situation is shown

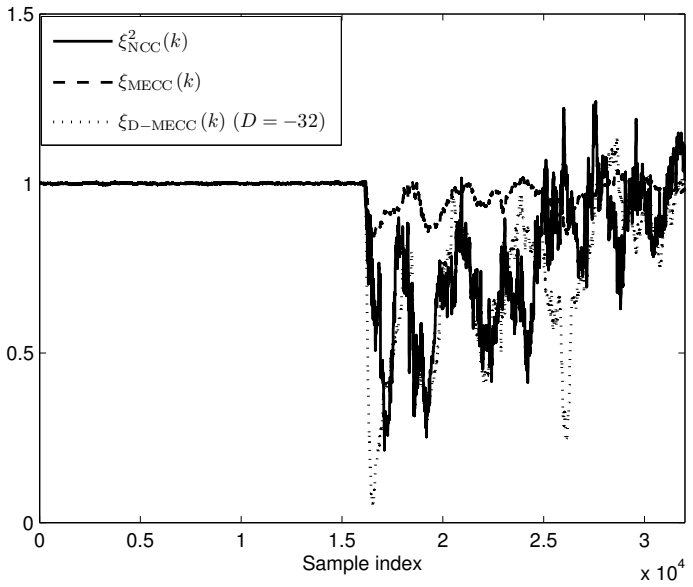

Figure 6: Comparison of DTD statistics in a situation with a constantly updating adaptive echo cancellation filter. Double-talk occurs after sample index 16200.

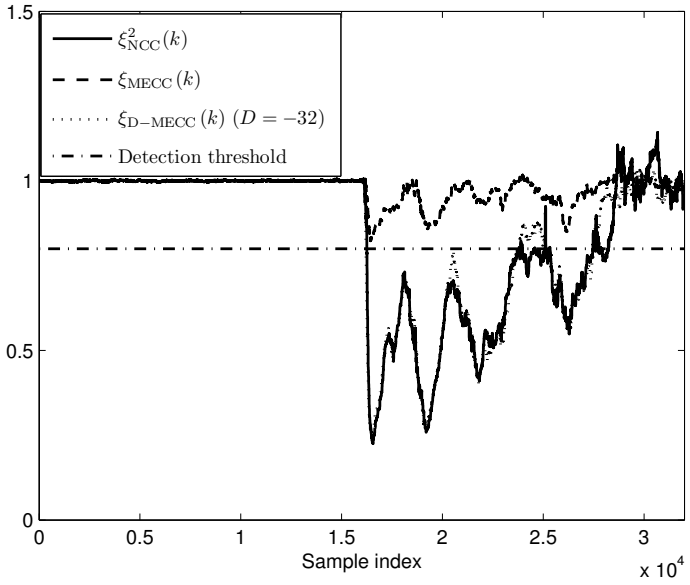

Figure 7: Comparison of DTD statistics in a situation where the adaptive echo cancellation filter is halted when the DTD statistic is < 0.8. Double-talk occurs after sample index 16200.

in figure 7 where it clearly can be seen that MECC fails to detect doubletalk, while the detection statistics for NCC and D-MECC both behave similar to the situation with no filter updates in figure 5.

Further, a double-talk detection threshold of 0.8 would also mean that both NCC and D-MECC detect doubletalk approximately at the same time, after $25 - 75$ samples (depending on exactly at which sample near-end speech is considered active). However, using the implementation of D-MECC with the delay D in the signal path of $y(k)$, as described in section 3.1, will result in a delayed update of the adaptive filter, i.e. the DTD will operate on $y(k - D)$ and the adaptive filter will operate on $y(k)$. Thus, the adaptive filter will in the D-MECC case update $|D|$ less iterations during actual double-talk than NCC (since the NCC DTD operates on $y(k)$), resulting in a reduced risk of a misadjusted filter.

In conclusion; NCC, MECC and D-MECC all show identical performance in the case of a fixed filter. In a more realistic scenario with an adaptive filter however, it is apparent that the performance of the three algorithms are very different. Therefore, further simulations to compare the performance of the algorithms are performed.

5 Simulations

To evaluate the performance of the double-talk detectors, the evaluation method in [10] was used, with the modification that an adaptive NLMS-based filter was used for echo cancellation instead of a fixed filter. The adaptive filter was constantly updating with step-size parameter $\mu = 0.95$ during far-end single-talk and was halted when the evaluated algorithm declared double-talk. The length of the adaptive filter was set to $N = 500$, which was the same length as the fixed echo-generating filter which was obtained by measurement in a standard office. The forgetting factor was set to $\lambda = 0.995$ for all averages, see section 3.1.

Like in [10], the far-end signal with duration of 12.5 seconds was from a male talker, sampled at 8 kHz and four different speech signals (two male and two female) of approximately 2 s each and also sampled at 8 kHz, were used as near-end speech. The near-end speech was set to occur at four different positions in time (at 6.25 s, 7.5 s, 8.75 s or 10 s) within the 12.5 s far-end speech. Independent flat spectrum noise with different intensity was added to the near-end signal, resulting in three different cases with echo-to-noise ratio (ENR) of 10 dB, 20 dB and 30 dB. In all cases the adaptive filter did reach

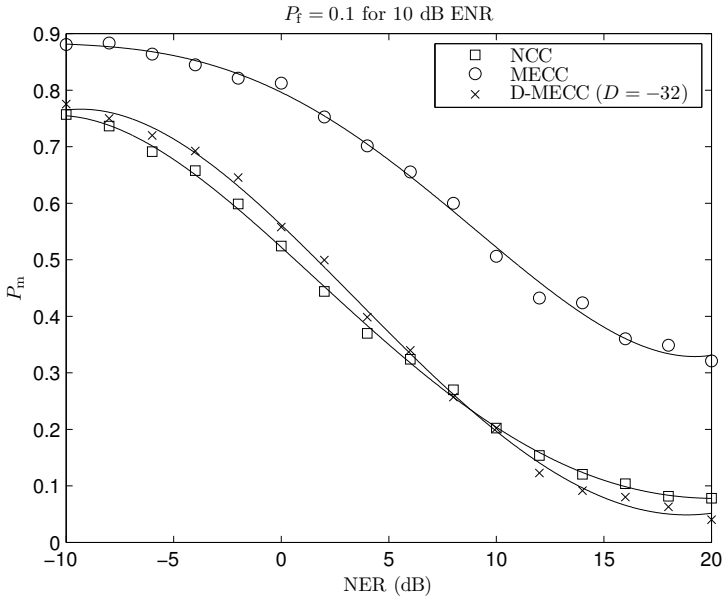

Figure 8: Performance of the three DTDs for $P_f = 0.1$ with 10 dB echo-to-noise ratio (ENR).

a steady-state in less than 5 seconds and after this the double-talk evaluation was initiated. Hence the adaptive filter misalignment depended only on the near-end noise intensity during the double-talk evaluation.

5.1 Results

The simulation results for $P_f = 0.1$ over a range of NERs and ENRs are shown in figures 8, 9 and 10. It can be seen that the NCC detector performs significantly better than the MECC detector, while the performance of the proposed D-MECC detector lies close to that of NCC for low NERs and relatively high misalignment (NER 10 dB shown in figure 8) and goes down towards (and occasionally surpasses) NCC for high NERs and for less misalignment (NER 20 dB and 30 dB shown in figures 9 and 10).

Simulations were also performed for $P_f = 0.3$, and the results are visible

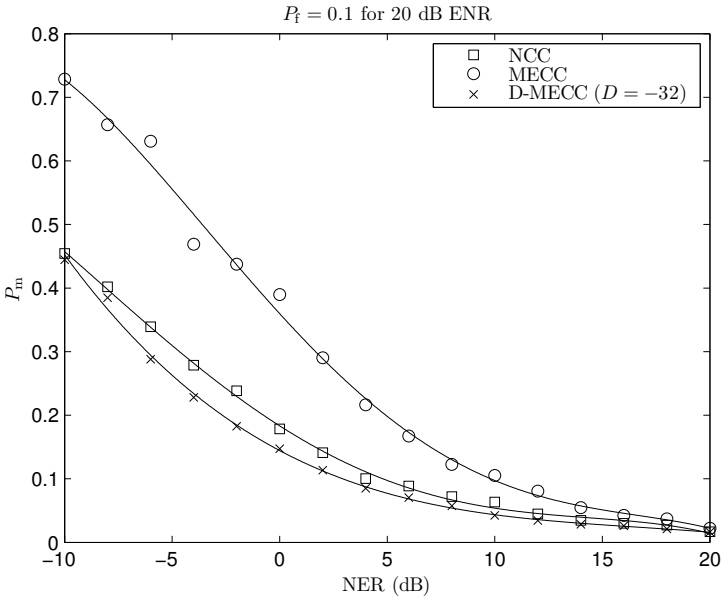

Figure 9: Performance of the three DTDs for $P_f = 0.1$ with 20 dB echo-to-noise ratio (ENR).

in figures 11, 12 and 13. In these cases NCC still shows the best performance, although the D-MECC performance is close. MECC shows the worst performance.

It should be noted that the difference in performance between MECC and NCC decrease for increased ENR and NER. The performance difference also seems to be smaller for $P_f = 0.3$ than for $P_f = 0.1$. This is probably owing to the fact that the adaptive filter is halted more often in these cases (as the DTD detects double-talk more frequently), and once the adaptive filter is halted, the performance of NECC and MECC are identical in theory given the same adaptive filter.

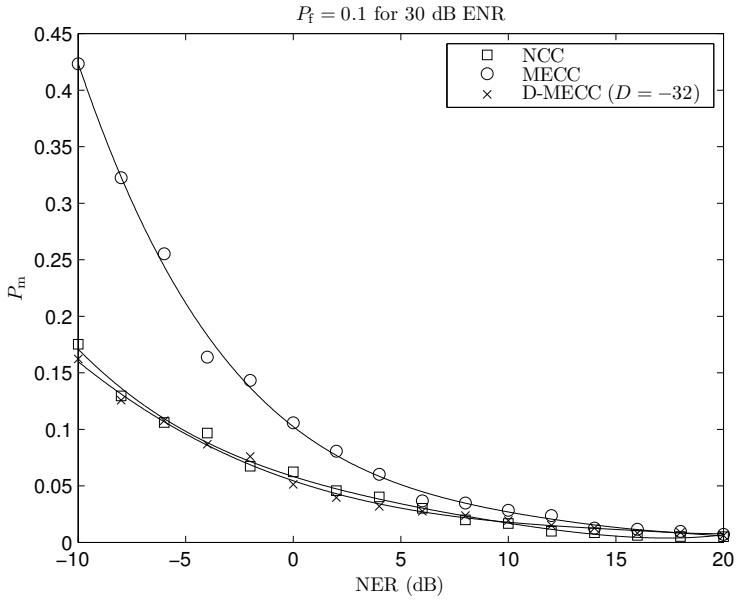

Figure 10: Performance of the three DTDs for $P_f = 0.1$ with 30 dB echo-to-noise ratio (ENR).

6 Experiments with recorded signals

Experiments were also carried out with signals recorded in a small office. The loudspeaker and microphone were placed with approximately 50 cm distance from each other on a desk and the loudspeaker volume was set so that the ENR was approximately 24 dB. In this case it was also necessary to increase the adaptive filter length to $N = 1000$ in order to capture most of the echo tail (all other parameters were unchanged). As in the simulations in section 5, the adaptive filter was allowed to converge for 5 seconds before double-talk was applied. Double-talk was applied in the same manner as previously.

The results of the experiments are shown in figures 14 and 15. Figure 14 shows the result for $P_f = 0.1$ and figure 15 shows the result for $P_f = 0.3$. It can be seen that all three DTDs in figure 14 show very similar results to the simulation with ENR set to 20 dB displayed in figure 9. The reason for the

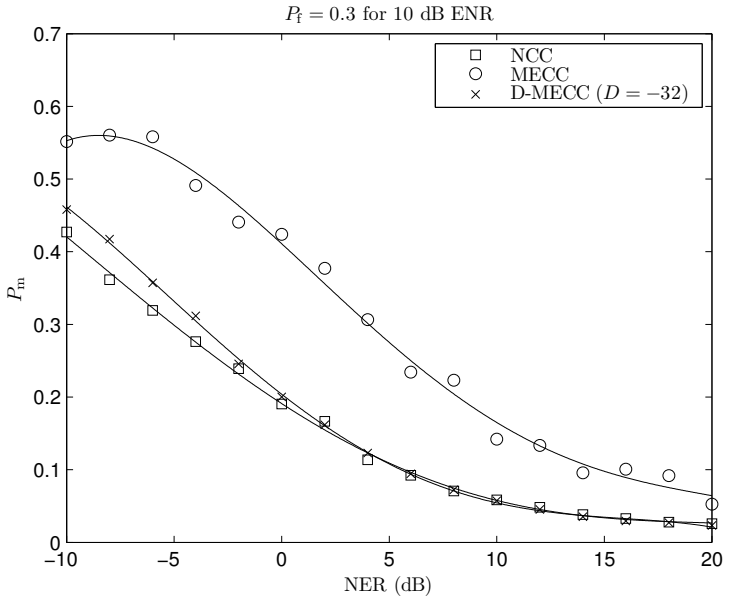

Figure 11: Performance of the three DTDs for $P_f = 0.3$ with 10 dB echo-to-noise ratio (ENR).

experiments not showing improved results, despite an ENR increase of 4 dB, is probably due to the colored background noise in the case of the recordings.

As in the simulations, NCC and D-MECC are fairly similar, while MECC exhibits worse performance.

It is thus clear that in realistic situations, with an adaptive NLMS filter which updates when the DTD does not indicate double-talk, the NCC detector is superior to the MECC detector. The proposed D-MECC detector performs overall significantly better than the MECC detector and slightly worse than the NCC detector. On the other hand, in terms of computational complexity, MECC has by far the lowest, followed by D-MECC, while the NCC requires the most, see section 3.1.

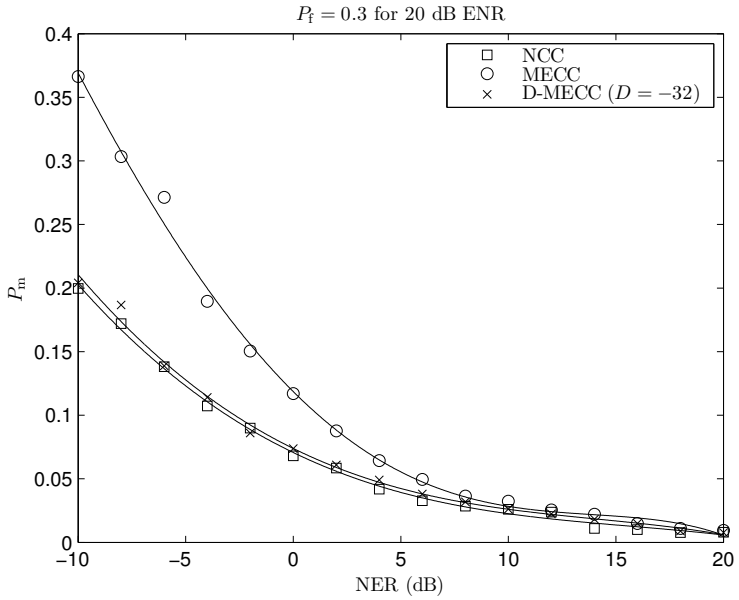

Figure 12: Performance of the three DTDs for $P_f = 0.3$ with 20 dB echo-to-noise ratio (ENR).

7 Conclusions

In [9] is it claimed that the performance of the MECC double-talk detector and the NCC double-talk detector are exactly similar. In this paper it was shown that this holds only under the assumption of a fixed echo cancellation filter. In a realistic situation with an adaptive NLMS filter updating when the DTD does not indicate double-talk, the MECC performs significantly worse than the NCC detector. This has been verified by simulations. Further, a novel DTD named D-MECC with computational complexity slightly higher than the MECC but much lower than NCC, has been proposed. It has been shown through simulations that the D-MECC performance is significantly better than MECC and appears to become comparable to that of NCC when the adaptive filter misalignment is low.

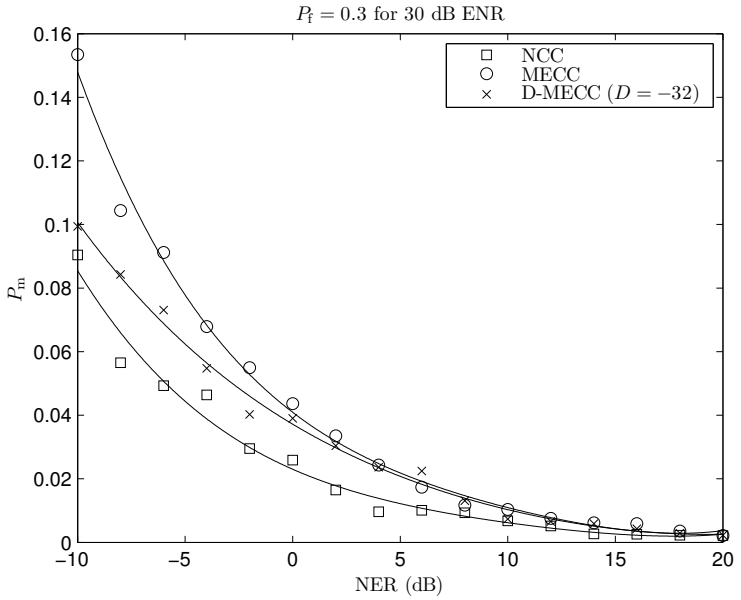

Figure 13: Performance of the three DTDs for $P_f = 0.3$ with 30 dB echo-to-noise ratio (ENR).

References

[1] S. Haykin, *Adaptive Filter Theory*, 4th ed. Prentice-Hall, 2002.

[2] D. Duttweiler, "A twelve-channel digital echo canceler," *IEEE Transactions on Communications*, vol. COM-26, pp. 647–653, May 1978.

[3] A. Mader, H. Puder, and G. U. Schmidt, "Step-size control for acoustic cancellation filters - an overview," *Signal Processing*, vol. 80, pp. 1697–1719, 2000.

[4] T. Gansler, M. Hansson, C.-J. Ivarsson, and G. Salomonsson, "A double-talk detector based on coherence," *IEEE Transactions on Communication*, vol. 44, pp. 1421–1427, November 1996.

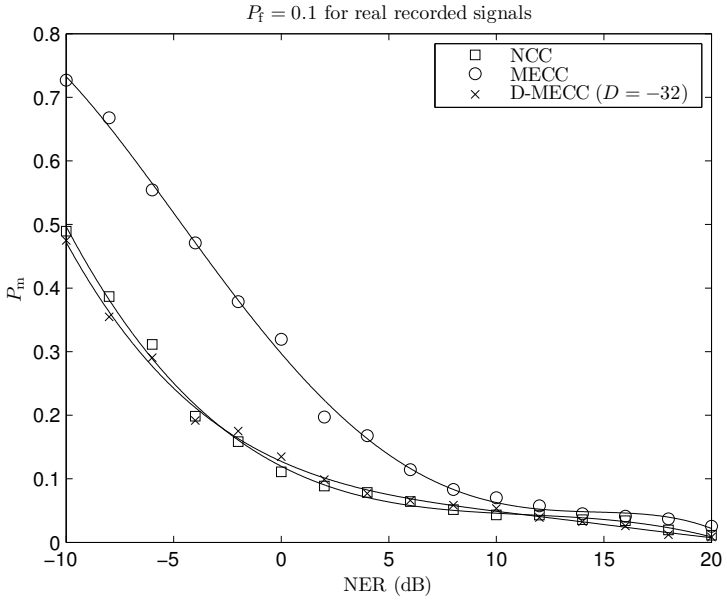

Figure 14: Performance of the three DTDs for $P_f = 0.1$ with recorded signals.

[5] J. Benesty, D. Morgan, and J. Cho, "A new class of doubletalk detectors based on cross-correlation," *IEEE Transactions on Speech and Audio Processing*, vol. 8, pp. 168–172, March 2000.

[6] E. Hänsler and G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach*. Wiley, 2004.

[7] K. Ochiai, T. Araseki, and T. Ogihara, "Echo canceler with two echo path models," *IEEE Transactions on Communications*, vol. COM-25, no. 6, pp. 8–11, June 1977.

[8] F. Lindstrom, C. Schüldt, and I. Claesson, "An improvement of the two-path algorithm transfer logic for acoustic echo cancellation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, pp. 1320–1326, May 2007.

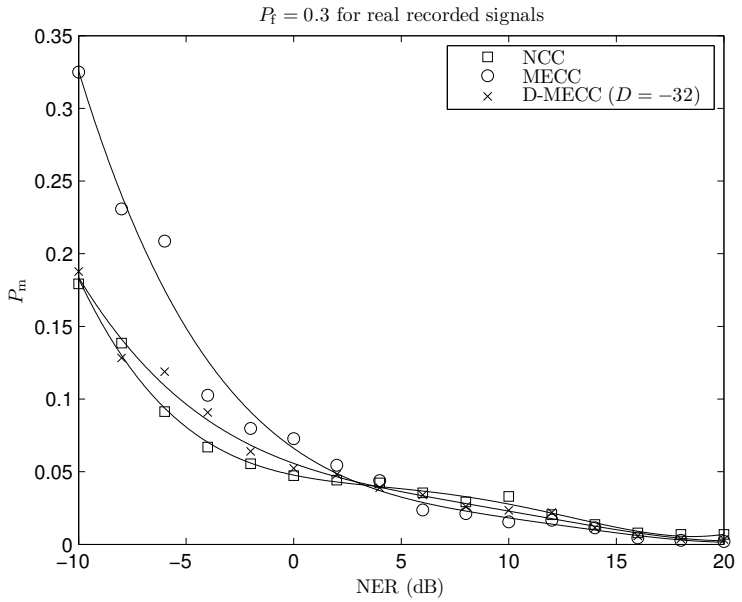

Figure 15: Performance of the three DTDs for $P_f = 0.3$ with recorded signals.

[9] M. Iqbal, J. Stokes, and S. Grant, "Normalized double-talk detection based on microphone and AEC error cross-correlation," in *Proceedings of IEEE International Conference on Multimedia and Expo*, July 2007, pp. 360–363.

[10] J. H. Cho, D. R. Morgan, and J. Benesty, "An objective technique for evaluating doubletalk detectors in acoustic echo cancelers," *IEEE Transactions on Speech and Audio Processing*, vol. 7, pp. 718–724, November 1999.

[11] P. Åhgren, "On system identification and acoustic echo cancellation," Ph.D. dissertation, Uppsala University, 2004.

[12] C. Schüldt, F. Lindstrom, and I. Claesson, "An improved deviation measure for two-path echo cancellation," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2010, pp. 305–308.

[13] ——, "Evaluation of an improved deviation measure for two-path echo cancellation," in *Proceedings of IWAENC International Workshop on Acoustic Echo and Noise Control*, September 2010.

[14] L. Rabiner and R. Schafer, *Digital Processing of Speech Signals*. Prentice-Hall, 1978.

Part IV

Robust Low-Complexity Transfer Logic for Two-Path Echo Cancellation

Part IV is reprinted, with permission, from

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson, "Robust Low-Complexity Transfer Logic for Two-Path Echo Cancellation," In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 173-176, Kyoto, Japan, March 2012.

Robust Low-Complexity Transfer Logic for Two-Path Echo Cancellation

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson

Abstract

A well used approach for echo cancellation is the two-path method, where two adaptive filters in parallel are utilized. Typically, one filter is continuously updated, and when this filter is considered better adjusted to the echo-path than the other filter, the coefficients of the better adjusted filter is transferred to the other filter. When this transfer should occur is controlled by the transfer logic.

This paper proposes transfer logic that is both more robust and more simple to tune, owing to fewer parameters, than the conventional approach. Extensive simulations show the advantages of the proposed method.

1 Introduction

A common approach in many speech communication applications where echo arise is to use the two-path adaptive filter structure [1], consisting of two echo cancellation filters, here denoted the *background filter* (BG filter) and the *foreground filter* (FG filter), respectively. The BG filter is continuously updated and the FG filter, which is producing the echo cancelled output, is fixed until the BG filter is considered to be better adjusted to the echo-path than the FG filter. When this occurs, the BG filter coefficients are transferred to the FG filter. Fundamental to the two-path filter structure is the *transfer logic*, which determines when the BG filter coefficients should be transferred to the FG filter. In the original two-path filter approach [1] several conditions are used for controlling the FG filter update. The main condition is that the output error magnitude of the BG filter must be less than that of the FG filter for a FG filter update to take place. However, due to cancellation of local speech during double-talk, this conditions is sometimes fulfilled despite the

BG filter being severely misadjusted [2]. Additional and alternative transfer logic control conditions have also been proposed [2, 3].

This paper presents a simple, yet robust transfer logic approach for two-path echo cancellation using the delay based filter deviation measure presented in [4] and the delay based double-talk detector presented in [5]. A more efficient way of calculating the delay based measures than in the original papers is also presented. It is shown through extensive simulations that the proposed transfer logic is more robust to erroneous FG filter updating during double-talk than previous approaches.

2 Two-path filtering

The two-path filtering scheme considered in this paper constitutes a constantly adapting BG filter and a fixed FG filter producing the echo cancelled output, as explained in the previous section. In this paper, as in [1, 2, 3, 4], the normalized least mean square (NLMS) is used to update the BG filter according to

$$\begin{aligned} e_b(k) &= y(k) - \hat{\mathbf{h}}_\mathbf{b}(k)^T \mathbf{x}(k) \\ \hat{\mathbf{h}}_\mathbf{b}(k+1) &= \hat{\mathbf{h}}_\mathbf{b}(k) + \mu \frac{e_b(k)\mathbf{x}(k)}{\mathbf{x}(k)^T\mathbf{x}(k) + \epsilon}, \end{aligned} \tag{1}$$

where $e_b(k)$ is the BG filter error signal, $\hat{\mathbf{h}}_\mathbf{b}(k) = [\hat{h}_{b_0}(k),$ $\hat{h}_{b_1}(k), \cdots, \hat{h}_{b_{N-1}}(k)]^T$ is the adaptive BG filter of length N, $x(k)$ is the driving signal fed to the echo-path (e.g. loudspeaker signal in case of acoustic echo cancellation (AEC), or signal fed to the telephone network in case of line echo cancellation (LEC)), $y(k)$ is the echo contaminated input signal (microphone signal in case of AEC, or signal from the telephone network in case of LEC), $\mathbf{x}(k) = [x(k), x(k-1), \cdots, x(k-N+1)]^T$ is the regressor vector, μ is the step-size control variable, ϵ is a regularization term to avoid division by zero and k is the sample index. $[\cdot]^T$ denotes transpose.

The FG filter, denoted $\hat{\mathbf{h}}_\mathbf{f}(k) = [\hat{h}_{f_0}(k), \hat{h}_{f_1}(k), \cdots, \hat{h}_{f_{N-1}}(k)]^T$, gives the output error

$$e_f(k) = y(k) - \hat{\mathbf{h}}_\mathbf{f}(k)^T \mathbf{x}(k), \tag{2}$$

and is updated by copying the BG filter coefficients, i.e. $\hat{\mathbf{h}}_\mathbf{b}(k)$ into the FG filter $\hat{\mathbf{h}}_\mathbf{f}(k)$. This is performed when the BG filter is considered to be better adjusted to the echo-path than the FG filter.

The process of determining when the FG filter should be updated is controlled by the transfer logic, which constitutes a set of conditions which has to be fulfilled for an update to take place. Typical transfer logic conditions are [1, 2, 3]

1. $\sigma_x^2(k) > T_1$ (sufficient excitation energy must exist)
2. $\sigma_y^2(k) > T_2$ (sufficient echo/near-end signal energy must exist)
3. $\frac{\sigma_{e_f}^2(k)}{\sigma_{e_b}^2(k)} > T_3$ (the BG filter must produce lower output error than the FG filter)
4. $\frac{\sigma_x^2(k)}{\sigma_{e_b}^2(k)} > T_4$ (sufficient echo cancellation and acoustic isolation must be present)

where T_1, T_2, T_3 and T_4 are thresholds and $\sigma_x^2(k)$, $\sigma_y^2(k)$, $\sigma_{e_b}^2(k)$, $\sigma_{e_f}^2(k)$ denote the short-time energy of the corresponding signals.

Additional transfer-logic conditions can be double-talk detectors of either Geigel-type [1] or based on the normalized cross-correlation [3] according to

$$1 - \frac{r_{ye_b}(k)}{\sigma_y^2(k)} > T_5, \tag{3}$$

where $r_{ye_b}(k) = \mathbb{E}\left[y(k)e_b(k)\right]$ and T_5 is a threshold. $\mathbb{E}[\cdot]$ denotes expectation (ensemble average; which in practice is approximated using time-averaging). Conditions related to the filter misalignment has also been presented [2, 3]. In [2], an artificial delay of L was introduced in the signal path of $y(k)$, causing the L first coefficients of the impulse response to be zero. This means that the first L coefficients of the impulse response are known (to be zero) and since the misalignment spreads over the whole filter [2], an estimate of the total filter misalignment can be made. While the solution in [2] works well for a fullband echo canceller, problems arise when trying to implement it for subband echo cancellation. The reason is that although the fullband echo-path is causal, the subband filters are not, due to temporal spreading, see [6] and references therein. Another filter misalignment condition has been proposed in [3] as

$$\left|\frac{r_{\hat{y}_f e_f}(k)}{r_{\hat{y}_f y}(k)}\right| > \left|\frac{r_{\hat{y}_b e_b}(k)}{r_{\hat{y}_b y}(k)}\right| \tag{4}$$

where $r_{\hat{y}_f e_f}(k) = \mathbb{E}[\hat{\mathbf{h}}_\mathbf{f}(k)^T \mathbf{x}(k) e_f(k)]$, $r_{\hat{y}_f y}(k) = \mathbb{E}[\hat{\mathbf{h}}_\mathbf{f}(k)^T \mathbf{x}(k) y(k)]$, $r_{\hat{y}_b e_b}(k) = \mathbb{E}[\hat{\mathbf{h}}_\mathbf{b}(k)^T \mathbf{x}(k) e_b(k)]$ and $r_{\hat{y}_b y}(k) = \mathbb{E}[\hat{\mathbf{h}}_\mathbf{b}(k)^T \mathbf{x}(k) y(k)]$.

3 Proposed approach

First of all, it should be noted that the transfer logic conditions in [1, 2, 3] all contain numerous thresholds and require careful tuning. This can be problematic if the echo canceller is expected to function well in different environments. Further, there are some problems with the previous approaches. Perhaps most notably is the problem of the filter misalignment measure in [3] (Equation (4) in this paper). This problem was illustrated and discussed in [4]. Moreover, in [4] an improved filter misalignment measure was also presented, but only as a stand-alone measure, i.e. not as a part of a complete two-path transfer logic solution.

Here, the improved deviation measure in [4] is incorporated in a complete two-path transfer logic solution together with the double-talk detector presented in [5]. It is also shown that the deviation measure, as well as the double-talk measure, can be calculated much more efficiently than originally presented.

It has been shown that the deviation measure in Equation (4) is not reliable for the BG filter during double-talk and an improved (BG filter) deviation measure was presented as [4]

$$\nu_{b_D}(k) = \left| \frac{r_{\hat{y}_{b_D} e_{b_D}}(k)}{r_{\hat{y}_{b_D} y}(k)} \right|, \tag{5}$$

where $r_{\hat{y}_{b_D} e_{b_D}}(k) = \mathbb{E}[\hat{y}_{b_D}(k)(y(k) - \hat{y}_{b_D}(k))]$, $r_{\hat{y}_{b_D} y}(k) = \mathbb{E}[\hat{y}_{b_D}(k) y(k)]$, $\hat{y}_{b_D}(k) = \hat{\mathbf{h}}_{\mathbf{b}}(k + D)^T \mathbf{x}(k)$ and D is a delay constant. The purpose of the delay is to avoid cancellation of near-end speech by the constantly updating BG filter. For more details regarding this matter, the reader is referred to [4] and [7].

The first proposed transfer logic condition is

$$\sigma_x^2(k) > T_1 \tag{6}$$

according to the traditional scheme (see previous section). This condition is also coupled with the NLMS update of the BG filter so that Equation (6) is true if the BG filter is adapted.

The deviation measure in Equation (5) is used in the second proposed transfer logic condition as

$$\left| \frac{r_{\hat{y}_f e_f}(k)}{r_{\hat{y}_f y}(k)} \right| > \left| \frac{r_{\hat{y}_{b_D} e_{b_D}}(k)}{r_{\hat{y}_{b_D} y}(k)} \right|. \tag{7}$$

Furthermore, the third transfer logic condition involves double-talk detection and is [5]

$$1 - \frac{r_{ye_{b_D}}}{\sigma_y^2} > T_5, \tag{8}$$

where $r_{ye_{b_D}} = \mathbb{E}[y(k)(y(k) - \hat{y}_{b_D}(k))]$. (The reader is referred to [5] for more details.)

The fourth and final condition of the proposed transfer logic is a straightforward output error comparison according to

$$\sigma_{e_f}^2(k) > \sigma_{e_{b_D}}^2(k), \tag{9}$$

where $\sigma_{e_{b_D}}^2(k) = \mathbb{E}[(y(k) - \hat{y}_{b_D}(k))^2]$ i.e. the squared output error magnitude of the (delayed) BG filter must be lower than that of the FG filter.

Hence, the proposed transfer logic involves only three tuning parameters: T_1, T_5 and D.

3.1 Practical considerations and complexity

The essence of the proposed transfer logic is the calculation of the echo estimate $\hat{y}_{b_D}(k)$ which should be decoupled as much as possible from the current adaptive filter update, using the delay D. It has been shown that a negative D achieves this decoupling better than a positive D [4] for the filter deviation measure. A straight-forward approach to calculate $\hat{y}_{b_D}(k)$ is to store all old filters and perform a filtering operation. However, a much more efficient solution is presented below.

Inserting the NLMS update Equation (1) into the expression for $\hat{y}(k)$ gives

$$\begin{aligned} \hat{y}_b(k) &= \mathbf{x}(k)^T \hat{\mathbf{h}}_\mathbf{b}(k) \\ &= \mathbf{x}(k)^T \left(\hat{\mathbf{h}}_\mathbf{b}(k-1) + \beta(k-1)\mathbf{x}(k-1) \right) \\ &= \hat{y}_{-1}(k) + \beta(k-1)\mathbf{x}(k)^T\mathbf{x}(k-1), \end{aligned} \tag{10}$$

where $\hat{y}_{-1}(k) = \mathbf{x}(k)^T \hat{\mathbf{h}}_\mathbf{b}(k-1)$ and $\beta(k) = \mu \frac{e_b(k)}{\mathbf{x}(k)^T\mathbf{x}(k)+\epsilon}$. From Equation (10) it can be seen that continuing to expand the expression using the NLMS update Equation (1) yields

$$\hat{y}_b(k) = \hat{y}_{b_D}(k) + \sum_{i=1}^{|D|} \beta(k-i)\alpha_i(k), \tag{11}$$

where $\alpha_i(k) = \mathbf{x}(k)^T\mathbf{x}(k-i)$. Thus it is clear that $\hat{y}_{b_D}(k)$ can be calculated as

$$\hat{y}_{b_D}(k) = \hat{y}_b(k) - \boldsymbol{\beta}(k)^T\boldsymbol{\alpha}(k), \tag{12}$$

where the vectors $\boldsymbol{\beta}(k) = [\beta(k-1), \beta(k-2), \cdots, \beta(k-|D|)]^T$ and $\boldsymbol{\alpha}(k) = [\alpha_1(k), \alpha_2(k), \cdots, \alpha_{|D|}(k)]^T$ are both of length $|D|$.

Since the echo estimate $\hat{y}_b(k)$ is calculated in the adaptive filtering update procedure, what remains for obtaining $\hat{y}_{b_D}(k)$ is the scalar product $\boldsymbol{\beta}(k)^T\boldsymbol{\alpha}(k)$. This scalar product requires $|D|$ multiplications and additions. The vector $\boldsymbol{\beta}(k)$ can be obtained at practically no additional computational cost, since $\beta(k)$ is calculated in the NLMS update Equation (1) just has to be delayed/stored in the vector. The elements of the vector $\boldsymbol{\alpha}(k)$ can be obtained recursively at low computational cost as $\alpha_i(k) = \alpha_i(k-1) - x(k-N)x(k-N-i) + x(k)x(k-i)$, i.e. using just two multiplications, one addition and one subtraction. In total, this means that calculating $\hat{y}_{b_D}(k)$ using the proposed method requires $3|D|$ multiplications and the same number of additions/subtractions. This should be compared to the straight-forward approach in [4], requiring N multiplications and the same number of additions. Since typically $|D| \ll N$, the proposed approach saves significant computational cost.

4 Simulations and results

The performance of the proposed transfer logic was verified through simulation with speech signals sampled at 8 kHz. Evaluation was performed for a DTF-modulated polyphase filterbank [6] with 32 subbands and a decimation ratio of 16. The number of prototype filter coefficients was 128. Each subband contained an individual setup of FG and BG filters with $N = 64$, and the BG filter was constantly updated when there was enough driving signal energy using the NLMS with variable step-size according to [8]. The two-path transfer logic of each subband was independent from the other subbands. This setup was used for both compared methods.

Evaluation of the proposed approach was made through comparison with the transfer logic presented in [3] which basically is the same as in [1] (see Section 2) with the addition of the filter misalignment condition in Equation (4) and the double-talk detector in Equation (3) instead of a Geigel detector. This approach was denoted the *conventional transfer logic* and the thresholds were $T_1 = 10^{-8}$ (corresponding to the threshold for BG filter NLMS updating), $T_2 = 10^{-10}$, $T_3 = 2$, $T_4 = 2$ and $T_5 = 0.95$. The proposed approach

uses only a total of four conditions, Equations (6), (7), (8) and (9), with three parameters: T_1 and T_5 which were set, as for the conventional transfer logic, to 10^{-8} and 0.95, respectively and $D = -4$, corresponding to a "full-band delay setting" of -64 with the selected decimation ratio [7].

For a BG-to-FG filter transfer to take place, the transfer logic conditions must be true for 50 consecutive subband samples, corresponding to 100 ms. This was used for both the conventional and the proposed method.

The ensemble averages used in the transfer logic conditions were estimated through time-averaging exponential recursive weighting [3, 4], e.g.

$$\hat{r}_{ye_b}(k+1) = \lambda \hat{r}_{ye_b}(k) + (1-\lambda) y(k) e_b(k) \tag{13}$$

and similarly for all other averages. The forgetting factor λ was set to 0.95.

To obtain the signal $y(k)$, the driving signal $x(k)$ was filtered with a known impulse response $\mathbf{h} = [h_0, h_1, \cdots, h_{N_f-1}]^T$ of length $N_f = 1024$ obtained through measurement in a small office. A flat spectrum noise signal was also added to $y(k)$, giving an echo-to-noise ratio of approximately 30 dB.

Two sets of scenarios were used: an echo-path change situation and double-talk. In both scenarios, the same driving speech signal of length 12 s was used. The performance of the different transfer logics was evaluated using the corresponding fullband FG filter misalignment evaluated as $10\log_{10}||\mathbf{h} - \hat{\mathbf{f}}_{\mathbf{f}}(k)||^2/||\mathbf{h}||^2$ where $\hat{\mathbf{f}}_{\mathbf{f}}(k)$ is a fullband filter constructed from all subband FG filters.

4.1 Echo-path change

In this scenario, the known impulse response $\mathbf{h}$ was changed after 6 s by shifting all filter coefficients one step to the left (i.e. $h_{i-1} = h_i, \quad i = \{1, \cdots, N_f - 1\}, \quad h_{N_f-1} = 0$). The resulting filter misalignment for the BG filter, the FG filter with the conventional transfer logic and the FG filter with the proposed transfer logic are shown in Figure 1. It can be seen that the proposed transfer logic allows the FG filter to track the adaptation of the BG filter slightly better than the conventional approach.

4.2 Double-talk

For the double-talk evaluation, a similar setup as in [5] where four different speech signals (two male and two female) of approximately 2 s each were used as near-end speech. The near-end speech was set to occur at four different

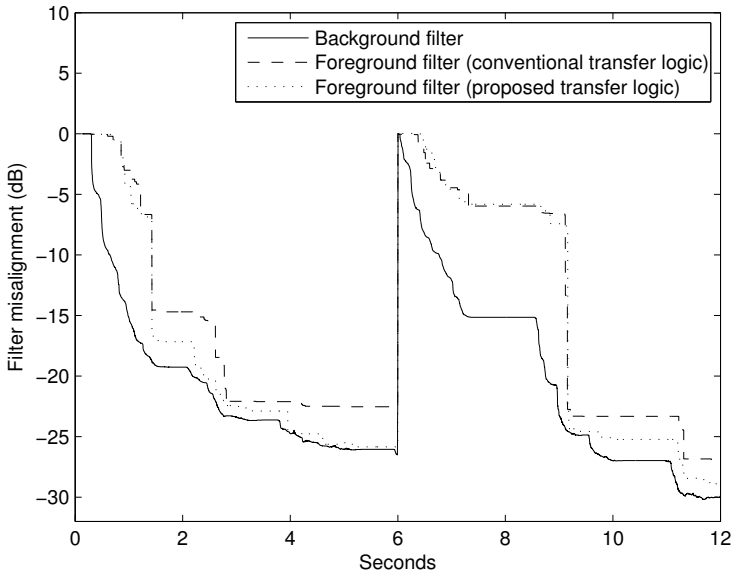

Figure 1: Misalignment of the background filter and the foreground filters of both compared methods. An echo-path change occurs after 6 s.

positions in time within the 12 s far-end speech, yielding a total of 16 simulations. The gain of the near-end speech was also varied to give a range of different near-end speech to echo ratios (NER).

The transfer logic performance was evaluated by comparing the FG filter misalignment during double-talk. For each of the 16 simulations, the maximum FG filter deviation during double-talk was noted. The ensemble mean of the maximum FG filter deviation during double-talk was then calculated, together with the overall maximum and minimum misalignment. This procedure was carried out for a range of NERs and the results are shown in Figures 2 and 3. By comparing the figures it can clearly be seen that the proposed method is more robust than the conventional transfer logic as the FG filter is constantly kept at a lower level during double-talk.

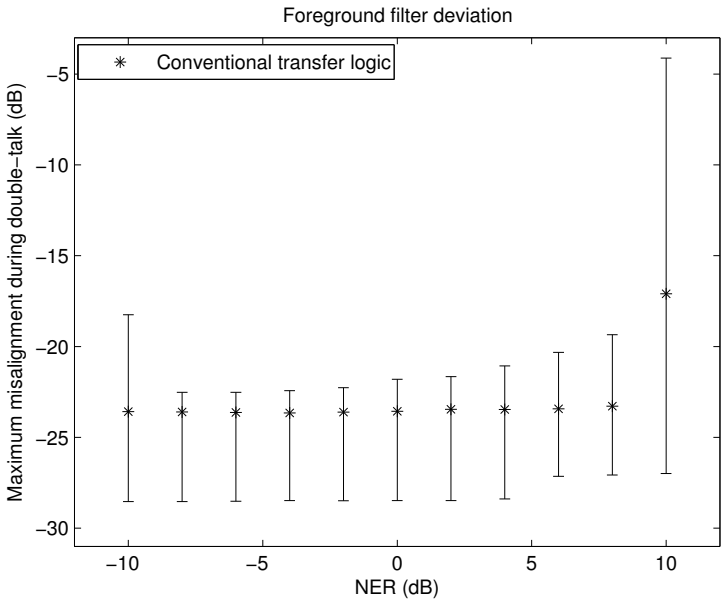

Figure 2: Ensemble averages, together with the maximum and minimum, of the maximum foreground filter misalignment during double-talk for different near-end-to-echo ratios (NER) for the conventional transfer logic.

5 Conclusions

An improved transfer logic scheme for two-path echo cancellation, based on the delay-based deviation measure in [4] and the delay based double-talk detector in [5], has been proposed. Extensive simulations have shown that the proposed transfer logic is more robust to double-talk than the conventional method, while also exhibiting slightly improved performance during a change of the echo-path.

References

[1] K. Ochiai, T. Araseki, and T. Ogihara, "Echo canceler with two echo path models," *IEEE Transactions on Communications*, vol. COM-25, no.

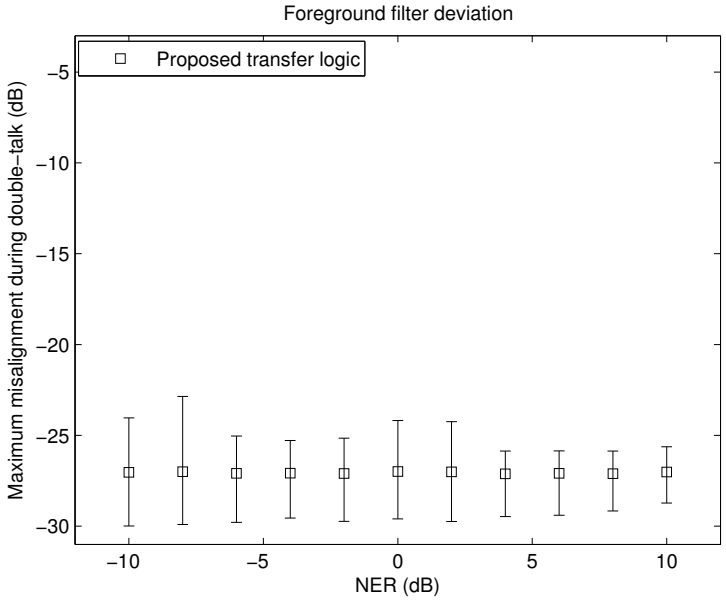

Figure 3: Ensemble averages, together with the maximum and minimum, of the maximum foreground filter misalignment during double-talk for different near-end-to-echo ratios (NER) for the proposed transfer logic.

6, pp. 8–11, June 1977.

[2] F. Lindstrom, C. Schüldt, and I. Claesson, "An improvement of the two-path algorithm transfer logic for acoustic echo cancellation," *IEEE Trans. on Audio, Speech and Language Proc.*, vol. 15, pp. 1320–1326, May 2007.

[3] M.A. Iqbal and S.L. Grant, "Novel and efficient download test for two path echo canceller," in *Proc. of IEEE WASPAA*, 2007, pp. 167–170.

[4] C. Schüldt, F. Lindstrom, and I. Claesson, "An improved deviation measure for two-path echo cancellation," in *Proc. of IEEE ICASSP*, March 2010, pp. 305–308.

[5] C. Schüldt, F. Lindstrom, and I. Claesson, "A delay-based double-talk detector," *IEEE Trans. on Audio, Speech and Language Proc.*, vol. 20, no. 6, pp. 1725–1733, 2012.

[6] E. Hänsler and G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach*, Wiley, 2004.

[7] C. Schüldt, F. Lindstrom, and I. Claesson, "Evaluation of an improved deviation measure for two-path echo cancellation," in *Proc. of IWAENC*, September 2010.

[8] J. Benesty, H. Rey, L.R. Vega, and S. Tressens, "A nonparametric VSS NLMS algorithm," *IEEE Signal Processing Letters*, vol. 13, no. 10, pp. 581–584, October 2006.

Part V

Adaptive Filter Length Selection for Acoustic Echo Cancellation

Part V is reprinted, with permission, from

Christian Schüldt, Fredric Lindstrom, Haibo Li, Ingvar Claesson, "Adaptive Filter Length Selection for Acoustic Echo Cancellation," *Signal Processing*, vol. 89, no. 6, pp. 1185-1194, June 2009.

Adaptive Filter Length Selection for Acoustic Echo Cancellation

Christian Schüldt, Fredric Lindstrom,
Haibo Li, Ingvar Claesson

Abstract

The number of coefficients in an adaptive finite impulse response filter based acoustic echo cancellation setup is an important parameter, affecting the overall performance of the echo cancellation. Too few coefficients give undermodelling and too many cause slow convergence and an additional echo due to the mismatch of the extra coefficients.

This paper proposes a method to adaptively determine the filter length, based on estimation of the mean square deviation. The method is primarily intended for identifying long non-sparse systems, such as a typical impulse response from an acoustic setup. Simulations with band limited flat spectrum signals are used for verification, showing the behavior and benefits of the proposed algorithm. Furthermore, off-line calculation using recorded speech signals show the behavior in real situations and comparison with another state-of-the-art variable filter-length algorithm shows the advantages of the proposed method.

1 Introduction

Adaptive finite impulse response (FIR) filter algorithms, in particular the least mean square (LMS) and variants thereof, such as the normalized least mean square (NLMS), have been extensively studied [1, 2]. The (N)LMS, as well as other adaptive filter algorithms such as recursive least squares (RLS) and affine projection algorithms (APA) have also been used in a variety of echo cancelling applications [3].

It is well known that a short adaptive filter converges faster than a long [2], although a long adaptive filter is often necessary to model real systems. For example the reverberation time of an ordinary office is typically in the order

of several hundred milliseconds [3], requiring an adaptive filter length of thousands of coefficients (assuming a sampling frequency of 8kHz) in an acoustic echo cancelling (AEC) application. Further, in a subband implementation, different lengths of the adaptive subband filters might be desireable since the reverberation time generally is different for different subbands in an acoustic environment [3]. For AEC implementations in a mobile device, e.g. a conference phone, adaptive filter length could be of interest since the acoustic echo canceller may have to operate in a large variety of acoustic environments.

An intuitive adaptive filter length approach is to start with a short filter which is gradually increased up to a maximum length, to improve the convergence speed [4]. Convergence performance analysis of this type of variable length LMS shows that it is less dependent on the eigenvalue spread of the input correlation matrix than the standard LMS, but is more affected by the relative values of the system coefficients [5].

A number of more elaborate variable length adaptive algorithms have also been proposed. For example, a gradient descent based approach with a cost function based on the mean squared error (MSE) [6], the use of three parallel adaptive filters of different lengths [7, 8] and splitting the adaptive filter in different segments and basing the length adaptation on the MSE output from the different segments [9]. A somewhat related approach [10] targeted for longer adaptive filters and acoustic echo cancellation also divides the filter into different segments, but in this case the segments are part of a long fixed length filter. This approach does not compare output errors from different filter segments, but rather uses the overall degree of convergence for determining which filter segment to update at each instant.

Another, more recent variable tap-length algorithm based on MSE output from different filter segments is the the pseudo fractional tap-length (N)LMS (FT-(N)LMS) [11], which combines the traditional segmented filter approach with a gradient descent based method.

In an acoustic echo cancellation setup, impairments of the echo cancellation performance caused by non-stationary local noise as well as non-linear effects of non-ideal components, such as loudspeakers and amplifiers, are common. This will affect the performance of a variable tap-length algorithm. Further, all mentioned approaches are based on the MSE, which can be unreliable in a speech based echo cancellation implementation since in situations with local disturbing signals, minor cancellation of the disturbing signal can occur due to the highly non-stationary nature of speech [12, 13].

This paper stresses the problem of an adaptive filter with non-optimal length. Typical examples in an acoustic echo cancellation application are

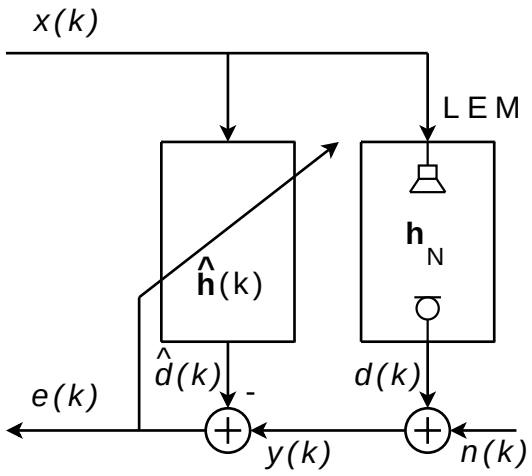

Figure 1: Adaptive filter setup, where $x(k)$ denotes the loudspeaker signal, $y(k)$ the microphone signal, $n(k)$ the near end noise and $d(k)$ and $\hat{d}(k)$ the acoustic echo and estimated acoustic echo, respectively. The loudspeaker-enclosure-microphone (LEM) system is described by $\mathbf{h}_N$ and the adaptive filter is described by $\hat{\mathbf{h}}(k)$.

shown and discussed. An adaptive length NLMS algorithm, based on estimation of the mean square deviation (MSD), for identifying long (several hundred coefficients) impulse responses is presented and tested through simulations. Comparison with another state-of-the art variable tap-length algorithm, FT-NLMS, using real recorded signals shows the advantages of the proposed adaptive length algorithm.

In this paper, the proposed algorithm is presented in conjunction with the NLMS, although essentially any other adaptive filtering method could be used. The reason for considering NLMS is because of its simplicity and relatively well known behavior.

2 Normalized least mean square adaptive filtering

The system is modeled as shown in figure 1, where $\mathbf{h}_N = [h_0, h_1, \cdots, h_{N-1}]^T$ is a vector describing the impulse response of a loudspeaker-enclosure-microphone (LEM) system, where N is the filter length and $[\cdot]^{\mathrm{T}}$ denotes transpose. The adaptive filter is described by $\hat{\mathbf{h}}(k) = [\hat{h}(k)_0, \hat{h}(k)_1, \cdots, \hat{h}(k)_{M-1}]^{\mathrm{T}}$, where M is the filter length and k is the sample index.

The acoustic echo from the LEM system $d(k) = \mathbf{x}_N(k)^{\mathrm{T}}\mathbf{h}_N$, where $\mathbf{x}_N(k) = [x(k), x(k-1), \cdots, x(k-N+1)]^{\mathrm{T}}$ and $x(k)$ is the loudspeaker signal, is added to a local noise signal $n(k)$ to form the microphone signal $y(k) = d(k) + n(k)$. The output from the adaptive filter, $\hat{d}(k) = \mathbf{x}(k)^{\mathrm{T}}\hat{\mathbf{h}}(k)$, where $\mathbf{x}(k) = [x(k), x(k-1), \cdots, x(k-M+1)]^{\mathrm{T}}$, is subtracted from the microphone signal $y(k)$ to form the output error signal from the adaptive filter as

$$e(k) = d(k) - \hat{d}(k) + n(k). \tag{1}$$

Updating of the adaptive filter is then performed as

$$\hat{\mathbf{h}}(k+1) = \hat{\mathbf{h}}(k) + \mu \frac{e(k)\mathbf{x}(k)}{\mathbf{x}(k)^{\mathrm{T}}\mathbf{x}(k) + \epsilon}, \tag{2}$$

where μ is the step-size control and ϵ is a regularization parameter.

3 Effects of filter order mismatch

Selecting the number of adaptive filter coefficients in an AEC application is a non-trivial task. In the selection process one typically accounts for the nature of the LEM system (if known), computational complexity, memory requirements and desired performance and chooses the number of adaptive filter coefficients given these parameters. The problem in a portable AEC solution, implemented in a mobile phone or another kind of mobile device, is that the acoustic environment, and thus also the LEM system, can vary significantly over time.

3.1 Too short filter

A too short filter will not be able to model the LEM, resulting in degraded echo cancellation performance. Figure 2, plot (a) demonstrates the problem

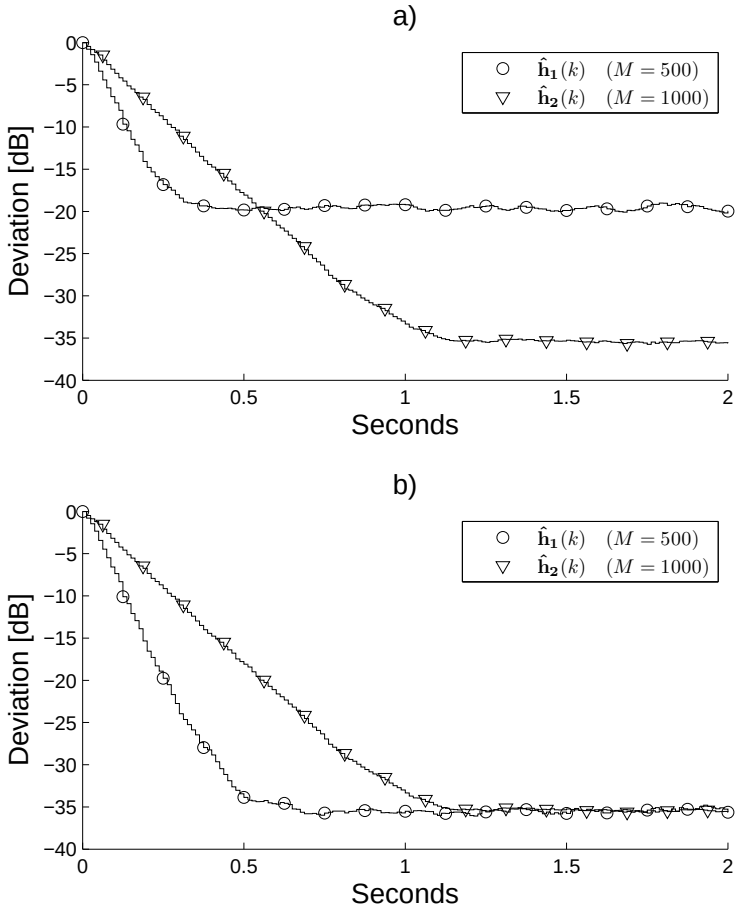

Figure 2: Normalized filter deviation for two adaptive filters, $\hat{\mathbf{h}}_1$ and $\hat{\mathbf{h}}_2$, with different length ($M = 500, 1000$). The system to be estimated, $\mathbf{h}_N$, is of order $N = 1000$ in plot (a) and $N = 500$ in plot (b). Bandlimited flat spectrum noise is used as input signal.

of insufficient modelling (in terms of normalized squared deviation [1]), with two adaptive filters, $\hat{\mathbf{h}}_\mathbf{1}(k)$ and $\hat{\mathbf{h}}_\mathbf{2}(k)$ of different lengths, $M_1 = 500$ and $M_2 = 1000$, respectively, estimating a LEM impulse response with length $N = 1000$. As can be seen, $\hat{\mathbf{h}}_\mathbf{2}(k)$ converges, but $\hat{\mathbf{h}}_\mathbf{1}(k)$ remains at a higher steady state deviation due to undermodelling.

3.2 Too long filter

The obvious drawback of a too long filter is slow convergence, shown in figure 2, plot (b) with two adaptive filters, $\hat{\mathbf{h}}_\mathbf{1}(k)$ and $\hat{\mathbf{h}}_\mathbf{2}(\mathbf{k})$ of different lengths, $M_1 = 500$ and $M_2 = 1000$, respectively, estimating a LEM system with length $N = 500$. Both adaptive filters eventually converge to the same steady-state deviation, but the too long filter $\hat{\mathbf{h}}_\mathbf{2}(k)$ converges slower than the filter with the same number of coefficients as the LEM system ($\hat{\mathbf{h}}_\mathbf{1}(k)$).

Moreover, since the error spreads over all filter coefficients [14] in the event of a mismatch, the adaptive filter itself will introduce an echo in this case. The magnitude and delay of this echo will depend on the significance of the mismatch, which in turn depends on near-end disturbances and non-linearities, as well as the filter length. This is demonstrated in figure 3, where a short noise burst (100 samples, shown in plot (a)) is fed to the converged adaptive filters $\hat{\mathbf{h}}_\mathbf{1}(k)$ and $\hat{\mathbf{h}}_\mathbf{2}(k)$ from figure 2, plot (b). Figure 3, plot (b) shows the output error from filter $\hat{\mathbf{h}}_\mathbf{1}(k)$ and plot (c) shows the output error from filter $\hat{\mathbf{h}}_\mathbf{2}(k)$.

As can be seen when comparing plots (b) and (c), the extra coefficients in the too long filter introduce an additional echo (from sample index 850 to sample index 1350), even though the filter has converged to a steady-state. In a real AEC implementation, this residual echo could very well be audible, especially if the filter is not properly converged, and must be removed with a residual echo suppressor or similar. A too long filter (plot (c)), will thus force the residual echo suppressor to be active for a longer period than for the shorter filter in plot (b), in this case an additional 500 samples (62.5ms with sampling frequency 8kHz). Since duplex constitutes a significant part of the perceived audio quality of a real AEC implementation, it is important that the residual echo suppressor is active for as short period as possible. Thus, a too long filter will cause a deterioration of the perceived audio quality.

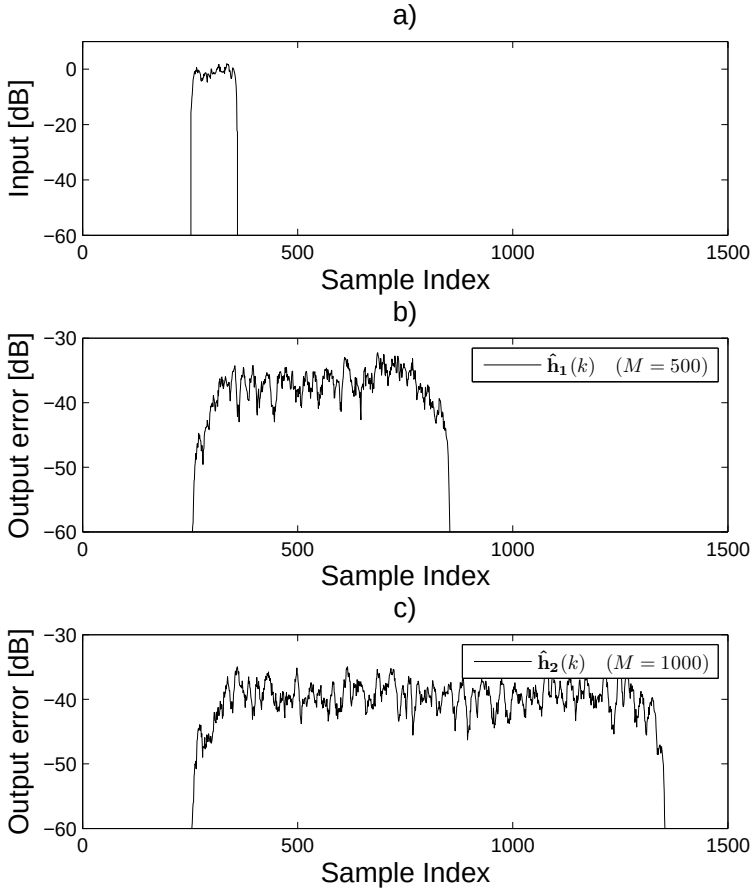

Figure 3: Plots (b) and (c) show short term average of the output error from two adaptive filters, $\hat{\mathbf{h}}_\mathbf{1}$ and $\hat{\mathbf{h}}_\mathbf{2}$, with different length ($M = 500, 1000$) during a noise burst (shown in plot (a)). The system to be estimated, $\mathbf{h}_N$, is of order $N = 500$.

4 Fractional tap-length algorithm

The fractional tap-length algorithm [11] is a MSE-based hybrid method, combining a segmented filter with a gradient descent approach. Defining $M_f(k)$ as the pseudo fractional tap-length at sample index k, which can take positive real values, the following adaptation rule is used

$$M_f(k+1) = (M_f(k) - \alpha) - \gamma\Big(e^2(k) - e_\Delta^2(k)\Big), \tag{3}$$

where α is a leaky factor, solving the "wandering" problem common to gradient descent algorithms, γ is a scaling constant, $e(k)$ is the output error produced by the full length filter $\hat{\mathbf{h}}(k)$ of length $M(k)$ and $e_\Delta(k)$ is the output error produced by a decimated version of $\hat{\mathbf{h}}(k)$ with length $M(k) - \Delta$, where Δ is a constant. The "true" filter length, $M(k)$, is then updated through

$$M(k+1) = \begin{cases} [M_f(k)] & \text{if } |M(k) - M_f(k)| > \delta \\ M(k) & \text{otherwise.} \end{cases} \tag{4}$$

Here δ is a constant and $[\cdot]$ denotes rounding to the nearest integer. For a more detailed description and analysis of the fractional tap-length algorithm, the reader is referred to [11].

5 The proposed algorithm

Like stated earlier, previous methods for adaptive filter length have been based on the MSE, which might be unreliable in a speech based echo cancellation implementation since in situations with local disturbing signals, minor cancellation of the disturbing signal can occur due to the non-stationary nature of speech [12, 13]. Instead, this paper proposes an adaptive filter length algorithm for acoustic echo cancellation based on estimation of the MSD.

Introducing an artificial delay of the error signal $e(k)$ prior to updating the adaptive filter will cause the first coefficients of the LEM system to be estimated to be zero. Since this means that a part of the system is known, it is possible to use this to estimate the deviation of the adaptive filter [1, 3]. This technique has previously been used for various purposes such as algorithms controlling the step-size of the NLMS [14, 15] and controlling the transfer logic in the two-path adaptive filter scheme [12]. In this paper, the artificial delay is used for adjusting the length of the adaptive filter modelling a system with a non-sparse impulse response, such as a typical LEM system.

By comparing the squared sum of the first adaptive filter coefficients corresponding to the artificial delay,

$$\mathcal{D}_F(k) = \sum_{i=0}^{D-1} \hat{h}_i^2(k) \tag{5}$$

where D is the length of the artificial delay, to the squared sum of the last coefficients,

$$\mathcal{D}_L(k) = \sum_{i=M-D-1}^{M-1} \hat{h}_i^2(k), \tag{6}$$

a decision is made wether the filter length should be increased, decreased or remain as it is. Naturally, the first filter section (containing only zeros corresponding to the artificial delay D in the ideal case) is of the same length D as the artificial delay. For straight forward comparison, the length of the last filter section is also chosen as D.

The condition for adjusting the adaptive filter length is

$$M(k+1) = \begin{cases} M(k) + D & \text{if } \frac{\mathcal{D}_L(k)}{\mathcal{D}_F(k)} > T_1 \text{ and } \frac{\mathcal{D}_F(k)}{\hat{\xi}_D(k)} > T_2 \\ M(k) - D & \text{if } \frac{\mathcal{D}_L(k)}{\mathcal{D}_F(k)} < T_3 \\ M(k) & \text{otherwise.} \end{cases} \tag{7}$$

where T_1, T_2 and T_3 are thresholds and $\hat{\xi}_D(k)$ is an estimate of the steady state mean square deviation for a filter section of length D.

When the adaptive filter is too short, the squared sum of the last coefficients will be sufficiently larger than the squared sum of the first coefficients, motivating the "increase" condition in equation (7) ($\frac{\mathcal{D}_L(k)}{\mathcal{D}_F(k)} > T_1$). When the filter is too long, the squared sum of the last coefficients is not sufficiently larger than the squared sum of the first coefficients, triggering the "decrease" condition ($\frac{\mathcal{D}_L(k)}{\mathcal{D}_F(k)} < T_3$).

The motivation for the auxiliary condition in equation (7) ($\frac{\mathcal{D}_F(k)}{\hat{\xi}_D(k)} > T_2$) is that since the error of each filter coefficient is time varying and might be different (although having the same statistical properties [14]), $\frac{\mathcal{D}_L(k)}{\mathcal{D}_F(k)}$ is also time varying even when the filter is fully converged. Thus, without the auxiliary condition the threshold T_1 must be set high to prevent the filter length from increasing occasionally despite $M(k) \geqslant N$ during situations with low signal-to-noise ratio (SNR). The auxiliary condition prevents the filter

from becoming too long by assuring that the filter length is increased only when the mismatch of the first filter coefficients (corresponding to the artificial delay) is significantly larger than the theoretical steady state deviation.

The steady state mean square deviation of a sufficient length adaptive filter under the assumption of white gaussian signals is $\frac{\mu}{2-\mu}\frac{\sigma_n^2}{\sigma_x^2}$ [3], where σ_n^2 is the variance of the local noise signal $n(k)$ and σ_x^2 is the variance of the input signal $x(k)$. This is used to calculate $\hat{\xi}_D(k)$ as

$$\hat{\xi}_D(k) = \frac{D}{M(k)}\frac{\mu}{2-\mu}\frac{\hat{\sigma}_n^2}{\hat{\sigma}_x^2}, \tag{8}$$

where $\hat{\sigma}_n^2$ is an estimate of the local noise variance and $\hat{\sigma}_x^2$ is an estimate of the input signal variance. In a real situation, estimation of the loudspeaker signal variance $\hat{\sigma}_x^2$ is straight-forward and the noise variance $\hat{\sigma}_n^2$ can be estimated with for example minimum statistic techniques [3].

Although the steady state mean square deviation expression presented above is only valid for white gaussian signals, it has been shown through experiments with other type of signals that equation (8) can be used also in these cases if the thresholds are set accordingly, see sections 5.1 and 7.

In practice the proposed algorithm is as follows. Every Kth sample, equations (5) and (6) (the squared sums) are calculated and length adaptation is performed according to equation (7). An average of the squared sums over several time instances could of course also be used, which would decrease the fluctuations, but would on the other hand also decrease the convergence speed. Further, at each length increase, the first D filter coefficients are forced to zero as a re-initialization of the squared sum in equation (5), i.e. to prevent accumulation of mismatch errors.

5.1 Setting of the thresholds

The thresholds T_1 and T_3 determines the biggest allowed difference between the squared sum of the D first filter coefficients, i.e. the filter mismatch in a sense, and the squared sum of the D last filter coefficients. It is desired to set T_1 as low as possible without causing over-estimation of the filter length and to set T_3 as high as possible without causing under-estimation of the filter length. In practice, this can be achieved through experimentation.

The threshold T_2 is then relatively easy to set, in comparison, since it determines at which point to stop increasing the filter length. For example, in a simulated environment with stationary noise T_2 can be set low since the

achievable steady-state squared deviation in practice is near the theoretical steady state mean square deviation. However, in a real situation with non-linearities and non-stationary local disturbances, this might not be achievable and in this case the threshold T_2 needs to be raised to a corresponding level.

Typical values for T_1, T_2 and T_3 in real acoustic situations (see section 7) are 1 dB, 12 dB and 0 dB, respectively.

6 Simulations

Three systems with different impulse responses $\mathbf{h}_{N_1}$, $\mathbf{h}_{N_2}$ and $\mathbf{h}_{N_3}$ with lengths $N_1 = 500$, $N_2 = 1000$ and $N_3 = 1500$ (shown in figure 4, where the artificial delay $D = 100$ also is present) were fed with a zero mean bandlimited flat spectrum signal with variance $\sigma_x^2 = 1$ as input signal $x(k)$, giving three output signals $d_1(k)$, $d_2(k)$ and $d_3(k)$. A noise signal $n(k)$ (independent from $x(n)$), also zero mean bandlimited flat spectrum but with variance $\sigma_n^2 = 0.0125$ was added to each output signal, forming $y_i(k) = d_i(k) + n(k)$, where $i = 1, 2, 3$. In all simulations the inverse SNR was calculated directly as $\frac{\hat{\sigma}_n^2}{\hat{\sigma}_x^2} = \frac{\sigma_n^2}{\sigma_x^2}$. The proposed algorithm was then set to estimate each system by feeding it with $x(k)$ and $y_1(k)$, yielding the estimate $\hat{\mathbf{h}}_{M_1}(k)$, then $x(k)$ and $y_2(k)$ yielding $\hat{\mathbf{h}}_{M_2}(k)$ and finally $x(k)$ and $y_3(k)$ yielding $\hat{\mathbf{h}}_{M_3}(k)$. The initial length $M(0)$ of the adaptive filter was 500 for all three cases and all signals have the sampling frequency 8kHz. Other parameters were as shown in table 1. The result can be seen in figure 5, where the normalized filter deviation of $\hat{\mathbf{h}}_{M_1}(k)$ is shown in plot (a_1), $\hat{\mathbf{h}}_{M_2}(k)$ in plot (a_2) and $\hat{\mathbf{h}}_{M_3}(k)$ in plot (a_3) and the length of the adaptive filters $\hat{\mathbf{h}}_{M_1}(k)$, $\hat{\mathbf{h}}_{M_2}(k)$ and $\hat{\mathbf{h}}_{M_3}(k)$ are shown in plots (b_1), (b_2) and (b_3), respectively. In the evaluation of the normalized filter deviation, the tail of the shorter vector (of $\mathbf{h}$ and $\hat{\mathbf{h}}(k)$) is zero padded to the length of $\max\{N, M(k)\}$ before calculating

$$\frac{\sum_{j=0}^{\max\{N,M(k)\}-1}(h_j - \hat{h}_j(k))^2}{\sum_{j=0}^{N-1} h_j^2}. \tag{9}$$

As can be seen in figure 5, the proposed algorithm manages to estimate the length of the unknown impulse response well, although there is a slight over-estimate of the length i.e. $M(k) = N + D$ for two of the systems (see plots (b_1) and (b_2)), during a short period right before convergence is reached. This is because $\mathcal{D}_F(k)$ has not reached the noise floor yet and thus, due to fluctuations the condition $\frac{\mathcal{D}_L(k)}{\mathcal{D}_F(k)} > T_1$ will become true and trigger a length increase.

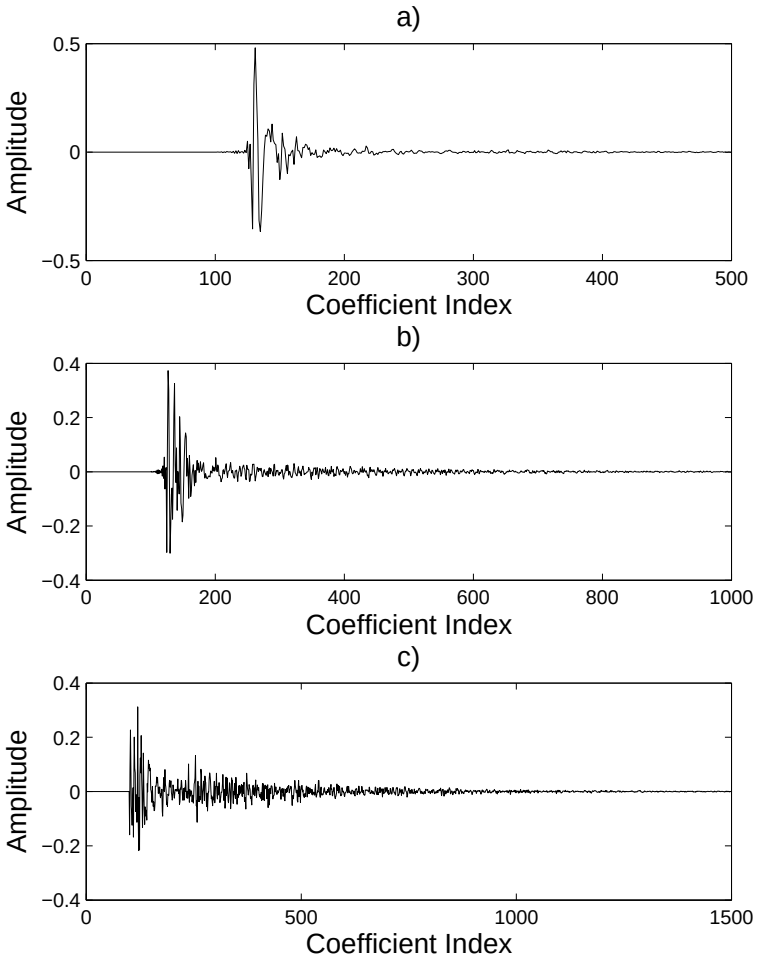

Figure 4: The three impulse responses used in the simulations.

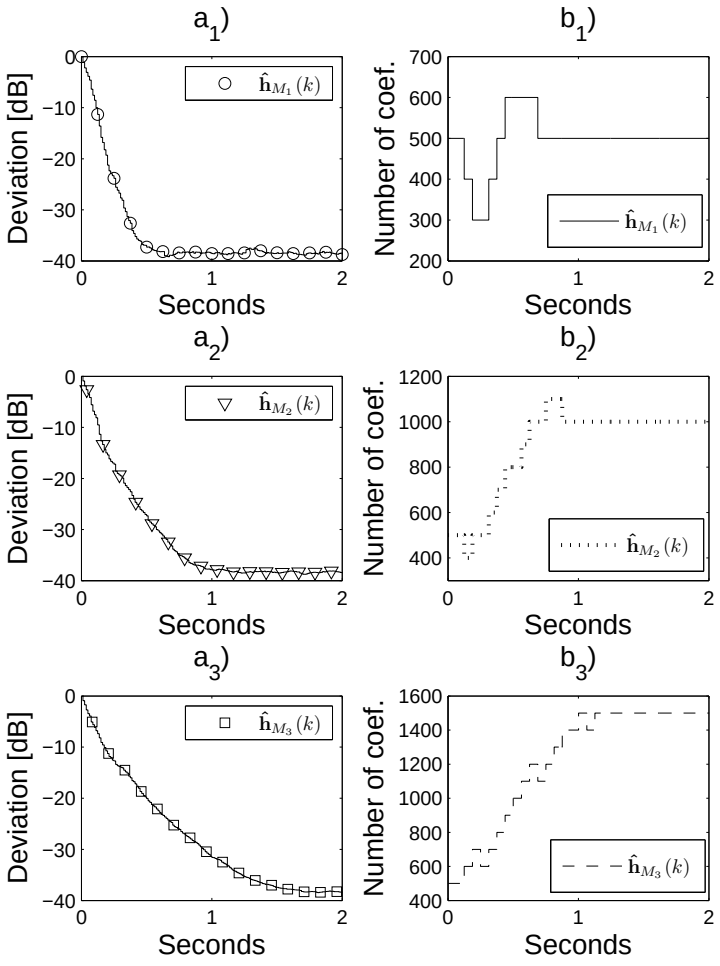

Figure 5: Plots (a_1),(a_2) and (a_3) show normalized squared deviation and plots (b_1),(b_2) and (b_3) show the corresponding number of coefficients for the proposed algorithm for three different lengths of the unknown LEM system (500, 1000 and 1500, respectively). A flat spectrum signal was used as input.

However, when convergence has been reached, $\mathcal{D}_F(k) \approx \hat{\xi}_D(k)$ and $\mathcal{D}_F(k) \approx \mathcal{D}_L(k)$ will cause a decrease in filter length, resulting in $M(k) = N$. This is illustrated in figure 6 and 7. $\mathcal{D}_F(k)$, $\mathcal{D}_L(k)$ and $\hat{\xi}_D(k)$ are shown in figure 7, plot (a) and the conditions for increasing the filter length (equation (7)) are shown in plot (b), respectively. The systems in figure 5) plots (b_1) and (b_2) as well as figure 6 decreases the filter length a brief period during initial convergence. This is due to the fact that the last filter taps have not had chance to converge (i.e. grow larger than the first filter taps), causing the condition $\frac{\mathcal{D}_L(k)}{\mathcal{D}_F(k)} < T_3$ to temporarily become true and cause a length decrease.

Figures 8 and 9 show the performance of the proposed algorithm during an echo path change situation. The adaptive filter has converged to the initial system $\mathbf{h}_{N_2}$, which is changed to the shorter impulse response $\mathbf{h}_{N_1}$ after 1 second. The figure shows that the proposed algorithm also handles a situation where both the impulse response itself as-well as the length are abruptly changed (after 1 second). As can be seen, the algorithm immediately responds to the change and gradually adjusts the filter length to the new impulse response.

7 Real system

The proposed algorithm was also evaluated in a real acoustic system were the driving signal was chosen as speech (see figure 10 plot (a)) and fed to a loudspeaker and the microphone signal was obtained through input from a microphone. The setup was placed in two rooms with different acoustic environments; one well damped room with short reverberation time (room 1), i.e. short impulse response, and one room with many hard sound reflecting

Parameter	Value
D	100
K	500
μ	0.95
ϵ	0.001
T_1	1 dB
T_2	3 dB
T_3	0 dB

Table 1: Simulation parameter settings.

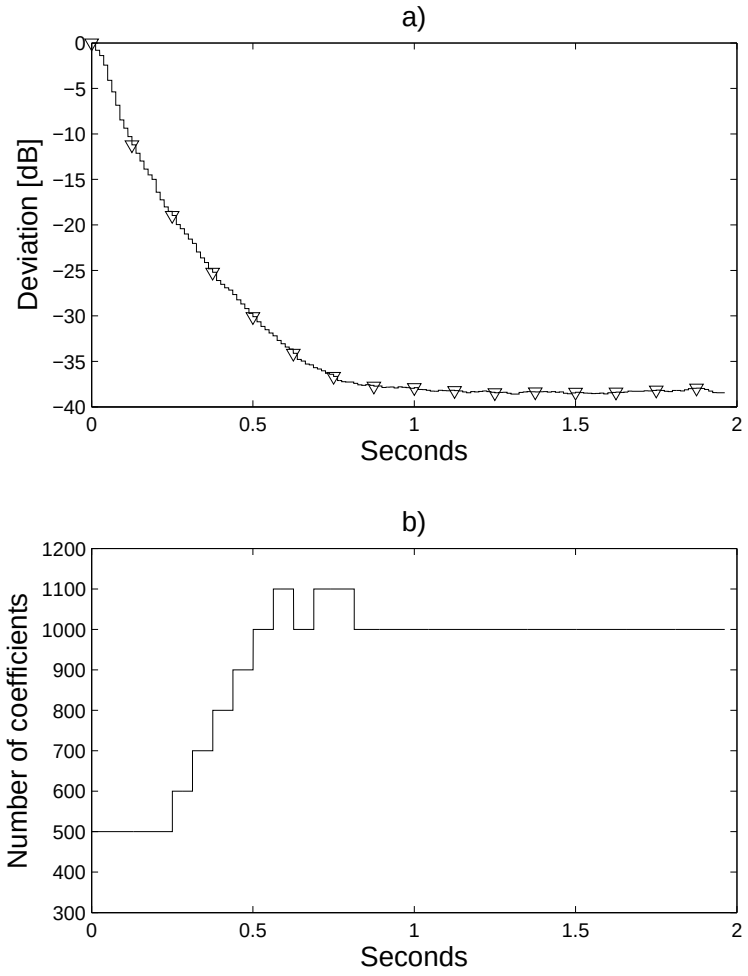

Figure 6: Normalized squared deviation (plot (a)) and number of coefficients (plot (b)) for the proposed algorithm with flat spectrum signal as input signal and $\hat{\mathbf{h}}_{N_2}$ as the LEM impulse response.

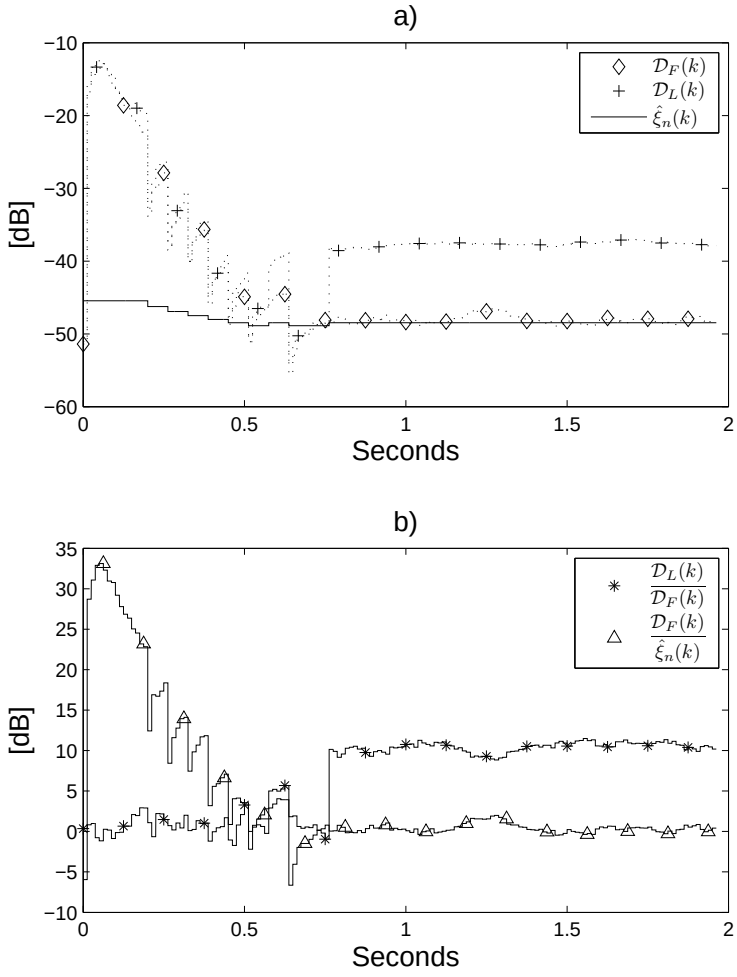

Figure 7: Plot (a) shows squared sums of different parts of the filter (equations (5) and (6)) and plot (b) shows the conditions for altering the filter length. Flat spectrum signal as input signal and $\mathbf{h}_{N_2}$ as the LEM impulse, as in figure 6.

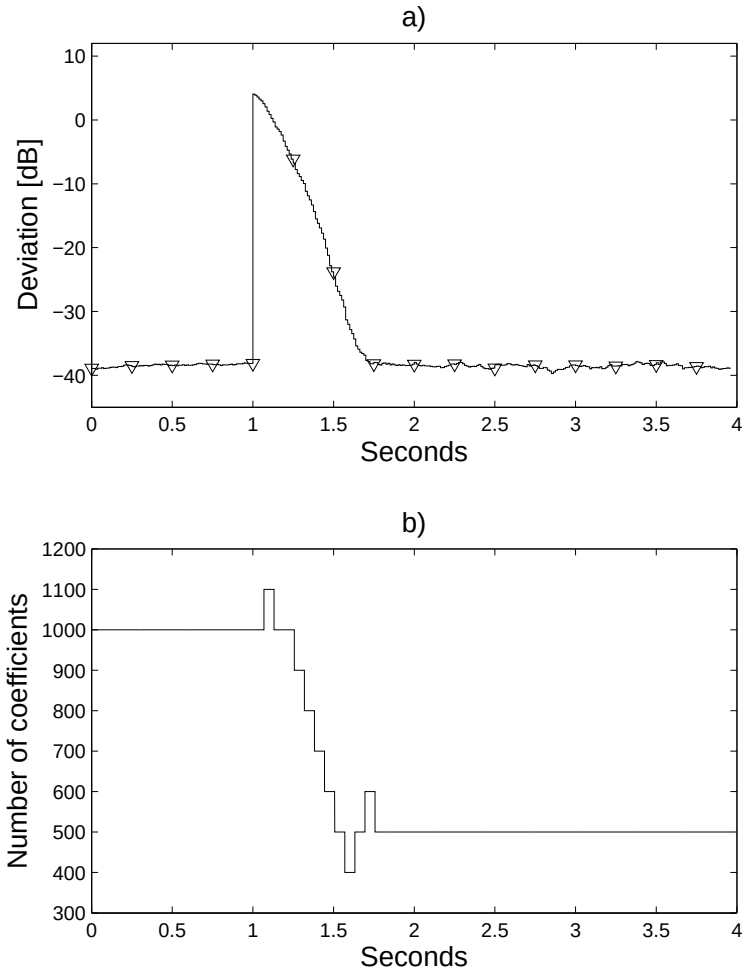

Figure 8: Plot (a) shows mean square deviation, plot (b) shows the number of coefficients for the proposed algorithm with a flat spectrum input signal in an echo path change situation, where switching from filter $\mathbf{h}_{N_2}$ to $\mathbf{h}_{N_1}$ after 1 second.

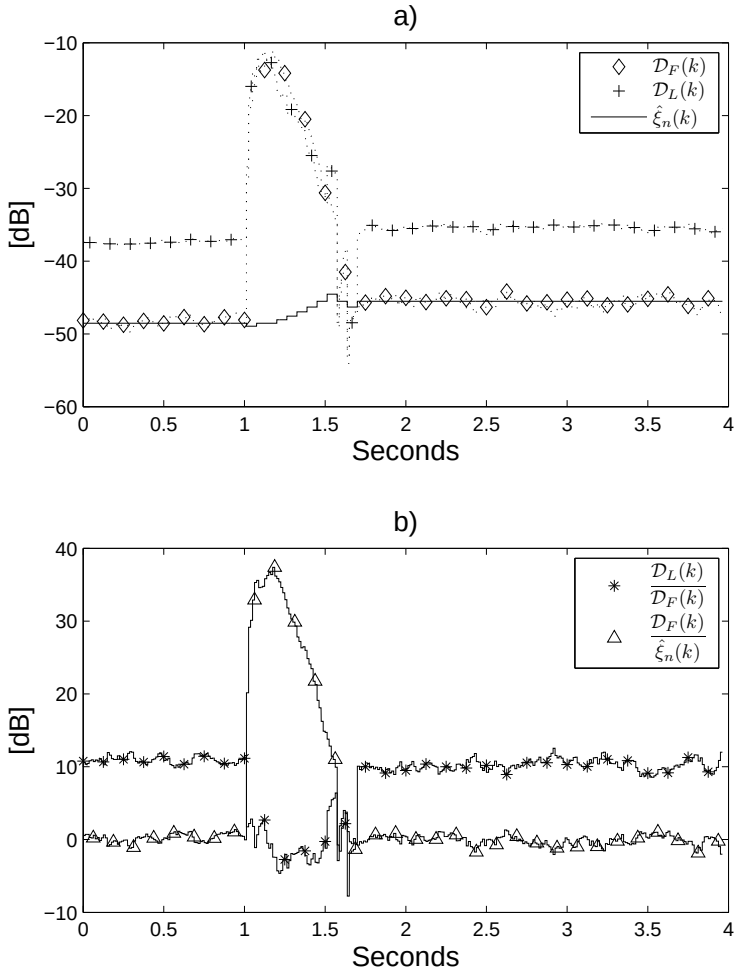

Figure 9: Plot (a) shows squared sums of different parts of the filter and plot (b) shows the conditions for altering the filter length for the proposed algorithm with flat spectrum input signal in an echo path change situation as in figure 8.

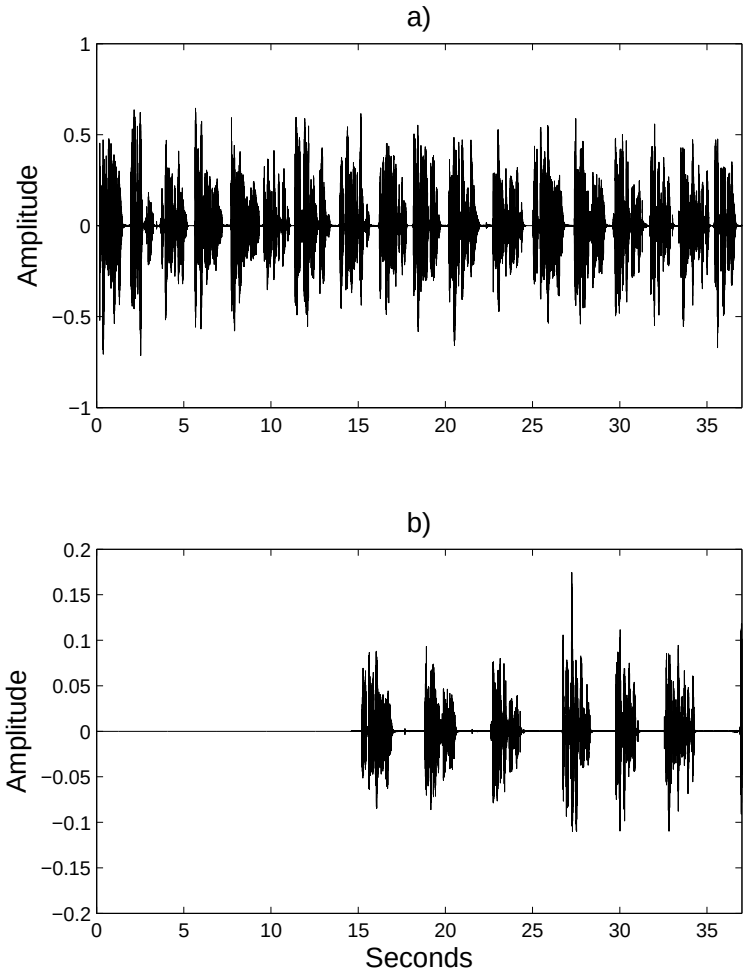

Figure 10: Plot (a) shows the loudspeaker signal used in the real systems experiments and plot (b) shows the disturbing near end speech signal used in one of the experiments.

surfaces causing a longer reverberation time (room 2). Due to the limited acoustic echo cancellation level in practice, 20-30 dB [3], the threshold T_2 have to be raised to compensate for this. In this case, T_2 was set to 12 dB.

Figures 11 and 12 show the results from evaluations in room 1 and room 2, respectively. Plot (b) in both figures show the residual echo from the proposed method together with the residual echo from the FT-NLMS, using using parameters shown in table 2. As expected, both the proposed algorithm and the FT-NLMS increase the number of filter coefficients for room 2 (figure 12), which is the more reverberant room, while using considerably fewer coefficients for the damped room 1, figure 11. It is obvious that the FT-NLMS over-estimates the number of filter coefficients for room 1 (figure 12) compared to the proposed algorithm, as there is basically no difference in the amount of echo cancellation between the two methods in this case. However, for room 2, the FT-NLMS, although showing slightly faster convergence, under-estimates the number of filter coefficients compared to the proposed algorithm, as can be seen again by comparing the amount of echo cancellation shown in figure 12.

The main problem of the FT-NLMS in this case is the lack of an adjustable tuning-parameter to get rid of the over- and under-estimation problems. Increasing γ would decrease the problem of under-estimating the filter length for room 2, but would on the other hand increase the amount of over-estimation for room 1. Adjusting α have similar effect, i.e. increasing this parameter reduces the over-estimation for room 1, but increases the under-estimation problem in room 2.

For room 1, using a larger number of filter coefficients than necessary obviously means non-optimal utilization of computational resources. Another disadvantage is the spreading of the residual echo in time, as discussed earlier in section 3.2.

Further, to test the robustness of the two variable tap-length algorithms, a local speech signal (shown in figure 10 plot (b)) was added to the microphone signal from room 1 (the damped room) as a non-stationary disturbance. Fig-

Parameter	Value
Δ	100
γ	256
α	0.004
δ	1

Table 2: FT-NLMS parameter settings.

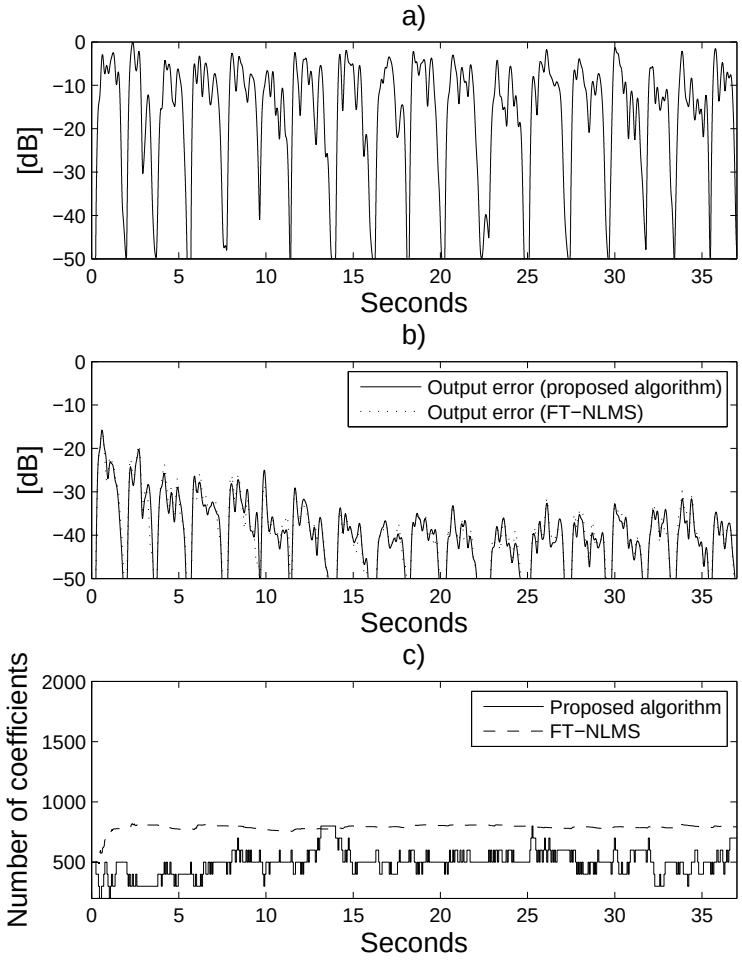

Figure 11: Plot (a) shows smoothed average of the microphone signal and plot (b) shows the smoothed average of the residual echo from the proposed algorithm as well as from the FT-NLMS. Plot (c) shows the number of coefficients used by the proposed algorithm and the FT-NLMS, respectively. Evaluation was performed using a real system (room 1).

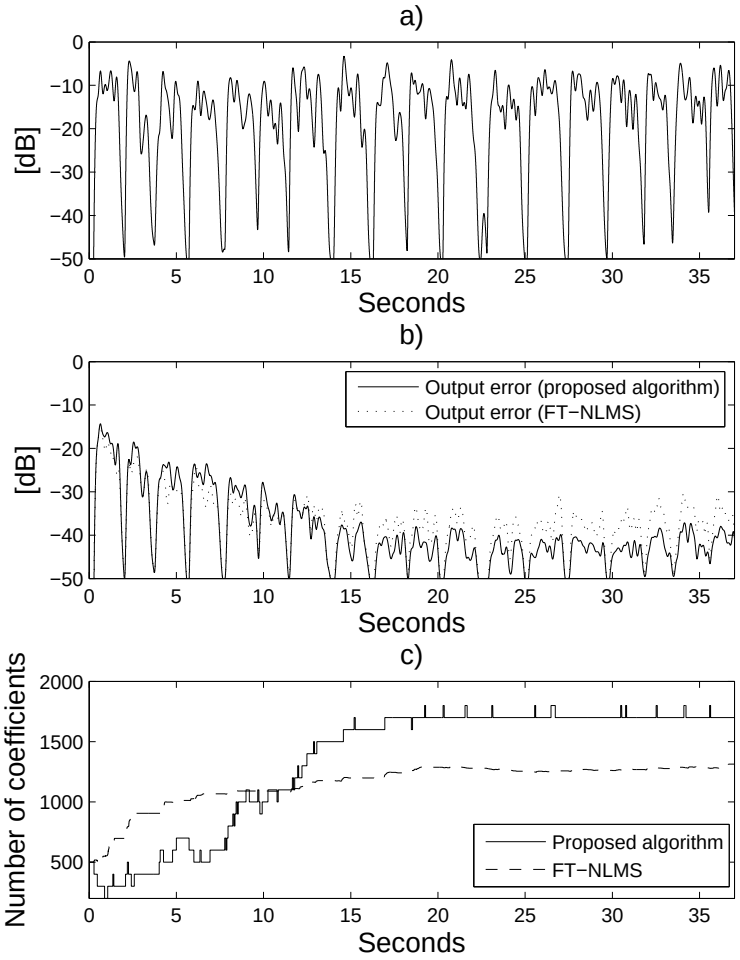

Figure 12: Plot (a) shows smoothed average of the microphone signal and plot (b) shows the smoothed average of the residual echo from the proposed algorithm as well as from the FT-NLMS. Plot (c) shows the number of coefficients used by the proposed algorithm and the FT-NLMS, respectively. Evaluation was performed using a real system (room 2).

ure 13 shows the result from this experiment and it can clearly be seen that the FT-NLMS does not handle the disturbance well. The near end speech is added after approximately 15 seconds and at this point the filter length of the FT-NLMS starts growing rapidly, while the proposed algorithm exhibits only slightly different behavior from the undisturbed case shown in figure 11.

Obviously, in a real acoustic echo cancellation system, there would also be a mechanism for detecting local disturbance, i.e. a doubletalk detector [3], which halts filter- and length-updating but this is a difficult problem and a badly tuned doubletalk detector could easily miss detecting local disturbance occasionally.

8 Conclusions

This paper has addressed the problem of modelling the impulse response of a system having N coefficients with an adaptive filter having M coefficients in the case of $M \neq N$ for an acoustic echo cancellation implementation. The downsides of having either a too short or a too long adaptive filters have been discussed. Moreover, an adaptive filter length algorithm based on estimation of the mean square deviation has been proposed. The algorithm has been verified through simulations and real off-line calculations with band limited flat spectrum signals and speech and compared to an existing variable tap-length algorithm, FT-NLMS. The results show that the proposed algorithm is more robust and has better tuning possibilities for acoustic echo cancellation environments, as compared to the FT-NLMS.

References

[1] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, 4th edition, 2002.

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

[3] E. Hänsler and G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach*, Wiley, 2004.

[4] Z. Pritzker and A. Feuer, "Variable length stochastic gradient algorithm," *IEEE Transactions on Signal Processing*, vol. 39, no. 4, pp. 997–1001, 1991.

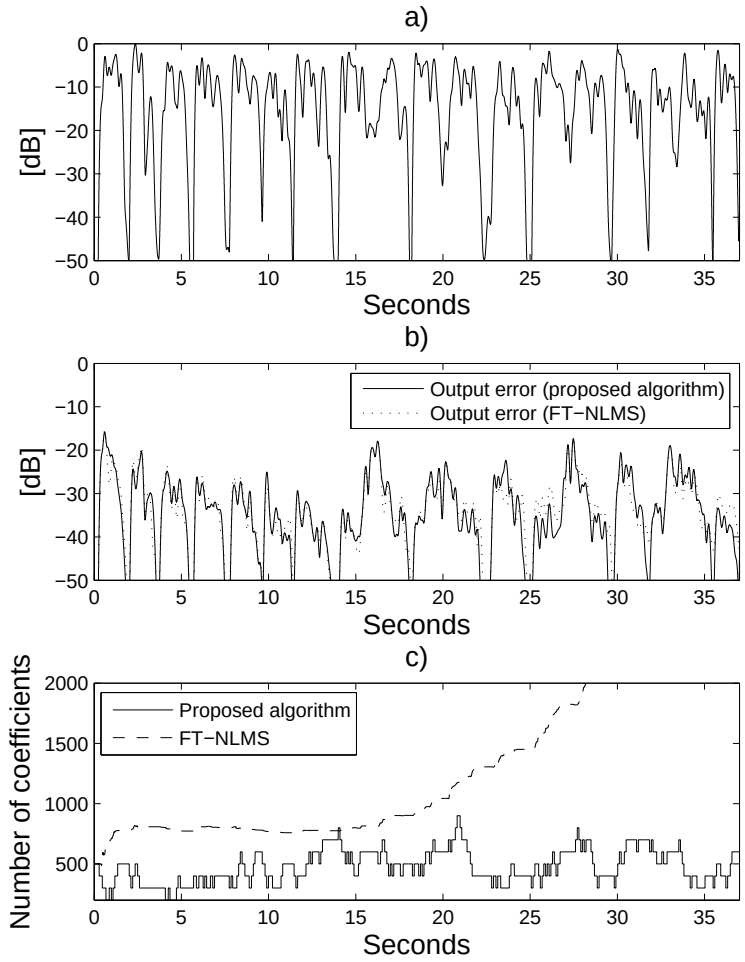

Figure 13: Similar to figure 11, with the difference that a local disturbing speech signal is added to the microphone after 15 seconds in this experiment.

[5] V.H. Nascimento, "Improving the initial convergence of adaptive filters: variable-length LMS algorithms," *14th International Conference on Digital Signal Processing*, vol. 2, pp. 667–670, 2002.

[6] Yuantao Gu, Kun Tang, and Huijuan Cui, "LMS algorithm with gradient descent filter length," *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 305–307, March 2004.

[7] R.C. Bilcu, P. Kuosmanen, and K. Egiazarian, "A new variable length LMS algorithm: theoretical analysis and implementations," *9th International Conference on Electronics, Circuits and Systems*, vol. 3, pp. 1031–1034, 2002.

[8] R.C. Bilcu, P. Kuosmanen, and K. Egiazarian, "On length adaptation for the least mean square adaptive filters," *Signal Processing*, vol. 86, no. 10, pp. 3089–3094, 2006.

[9] F. Riera-Palou, J.M. Noras, and D.G.M. Cruickshank, "Linear equalisers with dynamic and automatic length selection," *Electronics Letters*, vol. 37, no. 25, pp. 1553–1554, 2001.

[10] T. Usagawa, H. Matsuo, Y. Morita, and M. Ebata, "A new adaptive algorithm focused on the convergence characteristics by colored input signal: Variable tap length LMS," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. EA75-A, no. 11, pp. 1493–1499, 1992.

[11] Y. Gong and C.F.N. Cowan, "A LMS style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400–2407, 2005.

[12] F. Lindstrom, C. Schuldt, and I. Claesson, "An improvement of the two-path algorithm transfer logic for acoustic echo cancellation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 4, pp. 1320–1326, 2007.

[13] F. Lindstrom, M. Dahl, and I. Claesson, "The two-path algorithm for line echo cancellation," *Proc. of IEEE Tencon*, pp. 637–640, November 2004.

[14] S. Yamamoto and S. Kitayama, "An adaptive echo canceller with variable step gain method," *Trans. IECE Japan*, vol. 65, pp. 1–8, June 1982.

[15] A. Mader, H. Puder, and G. U. Schmidt, "Step-size control for acoustic cancellation filters - an overview," *Signal Processing*, vol. 80, pp. 1697–1719, 2000.

Part VI

A Low-Complexity Delayless Selective Subband Adaptive Filtering Algorithm

Part VI is reprinted, with permission, from

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson, "A Low-Complexity Delayless Selective Subband Adaptive Filtering Algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 12, pp. 5840-5850, December 2008.

A Low-Complexity Delayless Selective Subband Adaptive Filtering Algorithm

Christian Schüldt, Fredric Lindstrom, Ingvar Claesson

Abstract

Adaptive filters of significant order, requiring high computational complexity, are necessary in many applications such as acoustic echo cancellation and wideband active noise control. Successful approaches to lessen the computational complexity of such filters are subband methods, and partial updating schemes where only a part of the filter is updated at each instant. To avoid the time delay introduced by the subband-splitting, delayless structures which reconstructs a fullband filter, producing delayless output, from the adaptive subband filters have been proposed.

This paper proposes a delayless subband adaptive filter partial updating scheme, where the general idea is to only update the most misadjusted subband filter(s). Analysis in terms of mean square deviation is presented and shows that the fullband filter convergence speed is significantly increased, even for flat spectrum signals, as compared to traditional periodic subband filter update with the same computational complexity. Echo cancellation simulations with an artificial system to verify the analysis, using both flat spectrum signals and speech, is also presented, as well as off-line calculations using signals from a real system.

1 Introduction

Adaptive finite impulse response (FIR) filters is a vital component in many echo cancellation- and system estimation arrangements. The general idea is to feed the same input signal to both the system to be estimated and the adaptive filter, and using the difference of the respective outputs produced, i.e. the output error, as a measure of estimation performance. The output error is used for updating the adaptive filter. Perhaps the most frequently

used adaptive filter updating algorithm is the (normalized) least mean square ((N)LMS) [1], owing to its ease of implementation, low complexity and robustness to fix-point arithmetic implications. One drawback of the (N)LMS is however slow convergence speed, especially in the case of colored input signals. To speed up the filter convergence, at the cost of increased computational complexity, algorithms such as the recursive least squares (RLS), affine projection (AP) [1], and its computationally more efficient approximation fast affine projection (FAP) [2] have been proposed.

Another approach for both increased convergence speed, mainly in the case of colored input signals, and reduced computational complexity is subband adaptive filtering [3]. This can be performed either in a transform domain [4, 5], or in the time domain [6, 7]. Other means for reduced complexity include partial updating algorithms, where the idea is to avoid updating of all filter coefficients at each time instant. Periodic NLMS performs the filter update only at periodical sample intervals, while the sequential NLMS updates only a part of all coefficients at every sample in a sequential manner. In essence, although having different stability properties, the convergence performance of these two methods are similar [8]. Other suggested methods for better convergence performance have been e.g. choosing a subset of the regressor vector containing the largest coefficients [9] and block based regressor vector methods [10, 11]. Several combinations of subband structures and partial updating algorithms have also been proposed. These have either been based on sequential updating [12] or used the magnitude of the regressor vectors in the respective subbands as selection criterion [4, 13].

A disadvantage of conventional subband structures is the delay introduced in the signal path by the filterbanks. To avoid this issue, *delayless* subband adaptive filter architectures have been proposed [6], where the general idea is to reconstruct a fullband filter from the adaptive subband filters. The reconstructed fullband filter is then used to produce the fullband output. Thus, the signal filtering is performed using the fullband filter, avoiding the delay introduced by the filterbanks, while adaptive filters are adapted in the subbands.

This paper proposes a delayless subband adaptive filter partial updating scheme, based on the idea to update only the subband filters which are most misadjusted. The outline of the paper is as follows: In section 2, the proposed subband filtering method and filterbank structure are described and in section 3, the conventional delayless subband NLMS is described. Section 4 shows the relation between the fullband filter mean square deviation and the mean square deviation of the individual subband filters. Results from this sec-

tion is then used in section 5, where the proposed selective subband updating scheme is derived. Theoretical analysis of the proposed algorithm, periodic NLMS and full updating scheme is presented in section 6. The computational complexity of the proposed algorithm is presented in section 7. Section 8 verifies the analytical results through simulations using flat spectrum signals, colored stationary signals and speech with an artificial system in an echo cancellation application and in section 9, the proposed algorithm is subjected to speech signals recorded in a real setup in an office.

2 Polyphase filterbank structure

The delayless subband structure used in this paper is essentially the same as in [6], where the subband signals are obtained by convolution with a frequency shifted prototype lowpass filter [3]. The prototype filter used here is designed using the fast converging iterative least squares method provided by [14]. Thus, in the case of dividing the input signal $x(k)$ (see figure 1) into M subbands, the signal in subband $m \in \{0, \cdots, M-1\}$ will be

$$x_m(n) = \sum_{i=0}^{K-1} x(k-i) g_i e^{j\frac{2\pi m}{M} i}, \tag{1}$$

where n is the decimated subband sample index, R is the decimation ratio, $k = Rn$ is the fullband sample index, g_i is the i:th prototype filter coefficient, and K is the number of prototype filter coefficients. Rearranging the summation index in equation (1) according to

$$\begin{aligned} i &= sM + q \qquad && q \in \{0, \cdots, M-1\} \\ & && s \in \{0, \cdots, S-1\}, \end{aligned} \tag{2}$$

where $K = SM$ gives

$$x_m(n) = \sum_{q=0}^{M-1} e^{j\frac{2\pi m}{M} q} \sum_{s=0}^{S-1} x(Rn - sM - q) g_{sM+q}. \tag{3}$$

Thus, the subband filtering can be implemented very efficiently through M convolutions (each of length S) and one inverse FFT (fast fourier transform) every R:th fullband sample [3].

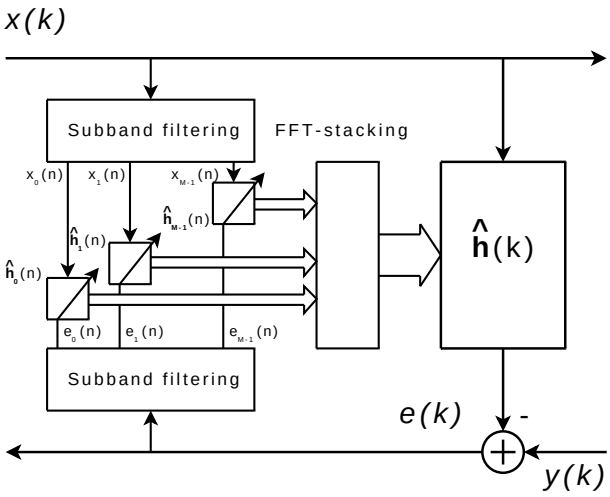

Figure 1: Closed loop delayless subband adaptive filtering configuration.

3 Subband normalized least mean square adaptive filtering implementation

For the closed loop case, which is considered in this paper, the non-delayed fullband output error is calculated directly as (see figure 1)

$$e(k) = y(k) - \hat{\mathbf{h}}(k)^T \mathbf{x}(k), \tag{4}$$

where $y(k) = d(k) + w(k)$, and $d(k)$ is the signal to be estimated, $w(k)$ is the local noise, $\hat{\mathbf{h}}(k) = [\hat{h}_0(k), \cdots, \hat{h}_{N-1}(k)]^T$ is the fullband adaptive filter, and $\mathbf{x}(k) = [x(k), \cdots, x(k-N+1)]^T$ is the regressor vector of length N. $[\cdot]^T$ denotes transpose. The fullband error $e(k)$ is partitioned into subbands $e_m(n)$, just as the input signal $x(k)$, see equation (1).

Then, NLMS updating of subband filter $\hat{\mathbf{h}}_m(n) = [\hat{h}_{m,0}(n), \cdots, \hat{h}_{m,N_M-1}(n)]^T$ of length N_M is performed as

$$\hat{\mathbf{h}}_m(n+1) = \hat{\mathbf{h}}_m(n) + \beta_m e_m^*(n) \mathbf{x}_m(n), \tag{5}$$

where

$$\beta_m = \frac{\mu_m}{||\mathbf{x}_m(n)||^2 + \epsilon}, \tag{6}$$

and μ_m is a step-size control parameter, ϵ is a regularization parameter [1], and * denotes complex conjugate. However, in the case of real fullband signals, which is what is considered in this paper, it is only necessary to update the $M/2+1$ first subband filters owing to Hermitian symmetry.

Transformation of the $M/2+1$ subband filters, in the case of $R = M/2$, is then performed through a technique called FFT-2 stacking [15], which is a refinement of the FFT-stacking technique suggested by [6]. The FFT-2 stacking is performed by taking a $2N_M$-point FFT of each subband filter and then stacking the DFT (discrete Fourier transform)-coefficients as

$$\hat{H}(l) = \begin{cases} \hat{H}_{[lM/2N]}(l \bmod 4N/M) & l \in [0, N) \\ 0 & l = N \\ \hat{H}^*(2N - l) & l \in (N, 2N) \end{cases} \tag{7}$$

where $\hat{H}(l)$ denotes DFT-coefficient l of the fullband filter and $\hat{H}_m(l \bmod 4N/M)$ denotes DFT-coefficient l modulo $4N/M$ of subband filter m, respectively. In this case $[\cdot]$ denotes rounding towards nearest integer.

Finally, the fullband filter is the N first samples of the $2N$-point inverse FFT of $\hat{H}(l)$.

4 Fullband- and subband filter deviation

This section describes the relation between the fullband filter mean square deviation and the mean square deviation of the individual subband filters in the FFT-2 stacking case. This relation is then used in the following section, where the proposed algorithm is derived.

The fullband filter deviation vector is defined as $\mathbf{v}(k) = \mathbf{h}_{\text{opt}} - \hat{\mathbf{h}}(k) = [v_0(k), \cdots, v_{N-1}(k)]^T$, where $\mathbf{h}_{\text{opt}}$ describes the unknown system to be estimated. It is assumed that $\hat{\mathbf{h}}(k)$ and $\mathbf{h}_{\text{opt}}$ are of equal length N. By taking the $2N$-point FFT of the adaptive filter and the optimal filter, respectively, the corresponding expressions can be obtained in the DFT-domain as $\text{DFT}_{2N}\{\mathbf{v}(k)\} = V(l) = H_{\text{opt}}(l) - \hat{H}(l)$. Defining the $2N$-point DFT of the N coefficient vector $\mathbf{v}(k)$ as

$$V(l) = \sum_{q=0}^{N-1} v_q(k) e^{-j\frac{2\pi l}{2N}q} \tag{8}$$

where $l \in \{0, \cdots, 2N\}$, and thus the inverse formula as

$$v_q(k) = \frac{1}{2N} \sum_{l=0}^{2N-1} V(l) e^{j\frac{2\pi q}{2N} l} \tag{9}$$

gives the mean square deviation (MSD) as

$$\mathcal{D}(k) = \mathbb{E}[||\mathbf{v}(k)||^2] = \mathbb{E}[\frac{1}{2N} \sum_{l=0}^{2N-1} |V(l)|^2], \tag{10}$$

according to Parseval's relation. $\mathbb{E}[\cdot]$ denotes expectation. Similarly, the MSD of subband filter m is defined as

$$\mathcal{D}_m(k) = \mathbb{E}[||\mathbf{v}_m(k)||^2] = \mathbb{E}[\frac{1}{2N_M} \sum_{l=0}^{2N_M-1} |V_m(l)|^2], \tag{11}$$

where $V_m(l)$ is the $2N_M$-point FFT of $\mathbf{v}_m(k)$, which in turn is the subband deviation vector for band m.

Now, examining the effect of the FFT-2 stacking procedure, equation (7), on the MSD, it is clear that $V(l)$ can be seen as being built up by stacked versions of $V_m(l)$. However, not all frequency coefficients of $V_m(l)$ are used to build up $V(l)$. In fact, it can be seen by examining equation (7) that only half of the coefficients of $V_m(l)$ are used. Moreover, due to the 2-times oversampling, there is a frequency overlap between the adaptive subband filters. Considering ideal subband filtering, it can be assumed that the overlapping frequency bins of two neighboring adaptive filters are approximately equal. This means that

$$\frac{1}{2} \sum_{m=0}^{M-1} \sum_{l=0}^{2N_M-1} \mathbb{E}[|V_m(l)|^2] \approx \sum_{l=0}^{2N-1} \mathbb{E}[|V(l)|^2], \tag{12}$$

and inserting equation (11) gives

$$\begin{aligned} \mathcal{D}(k) &= \mathbb{E}[||\mathbf{v}(k)||^2] = \\ &\frac{1}{2N} \sum_{l=0}^{2N-1} \mathbb{E}[|V(l)|^2] \approx \frac{N_M}{2N} \sum_{m=0}^{M-1} \mathcal{D}_m(k). \end{aligned} \tag{13}$$

From equation (13) it can be seen that the mean square deviation of the fullband filter is proportional to the sum of the mean square deviation of the subband filters. The proportionality constant will depend on the FFT-scaling and the amount of oversampling.

5 Proposed selective subband updating

By allowing only a subset of the adaptive subband filters to update at each instant, reduction of the computational complexity can be achieved. In this particular approach, the updating of only *one* subband filter each sample instant will be considered. The proposed scheme is shown in figure 2.

Besides the reduced computational complexity achieved through absent filter updates, the FFT-2 stacking in this case can be modified for further reduced complexity. Since only one subband filter has changed since the last subband sample, it is only necessary to compute the $2N_M$-point FFT of the corresponding filter for the stacking. Further, when constructing the fullband filter, instead of performing the $2N$-point FFT of the whole filter, it is possible to consider the difference between the fullband filter from the previous update and the currently updated fullband filter, i.e.

$$\mathbf{c}(k) = \hat{\mathbf{h}}(k) - \hat{\mathbf{h}}(k-R), \tag{14}$$

and thus in the DFT-domain

$$C(l) = \hat{H}(l) - \hat{H}_p(l) \qquad l \in \{0, \cdots, 2N-1\}, \tag{15}$$

where $\mathrm{DFT}_{2N}\{\hat{\mathbf{h}}(k-R)\} = \hat{H}_p(l) \quad l \in \{0, \cdots, 2N-1\}$. Obviously, $C(l)$ will be 0 for all l which corresponds to an unchanged subband filter (see equation (7)). Hence, $C(l)$ will only contain N_M non-zero components, which means that the inverse FFT of $C(l)$, i.e. $\mathbf{c}(k)$, can be computed very efficiently. Finally, the updated fullband filter is obtained through $\hat{\mathbf{h}}(k) = \hat{\mathbf{h}}(k-R) + \mathbf{c}(k)$. This technique is denoted *FFT-difference stacking*, see figure 2.

For selecting which subband filter to update at each instant, a periodic selection scheme where the filters are sequentially selected in a round-robin manner, or a random selection scheme where the filters are selected randomly could be used. However, as obviously a low fullband filter deviation is desired in as few updates as possible, m should be chosen as the subband corresponding to the largest current deviation reduction. This is since the fullband mean square filter deviation is proportional to the sum of the mean square deviation of the subband filters, as given by equation (13). The general idea is similar to the multichannel reasoning in [16] and the buffering technique described in [17].

The square deviation change in one subband m, using equation (5), is given as

$$||\mathbf{v}_m(n+1)||^2 = ||\mathbf{v}_m(n) - \beta_m(n) e_m^*(n) \mathbf{x}_m(n)||^2. \tag{16}$$

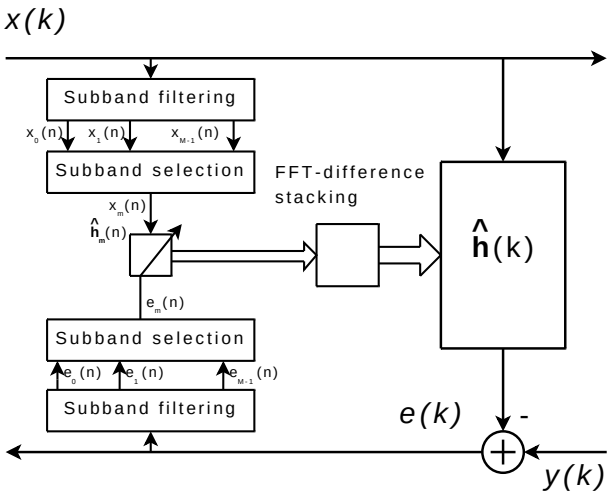

Figure 2: Proposed adaptive filtering configuration.

By inserting equation (6), assuming no regularization is used, i.e. $\epsilon = 0$, and the definition

$$t_m(n) = \mathbf{v}_m^H(n)\mathbf{x}_m(n) \tag{17}$$

into equation (16), the following expression is obtained [1]

$$\begin{aligned} ||\mathbf{v}_m(n+1)||^2 &= ||\mathbf{v}_m(n)||^2 \quad + \\ + \quad \mu_m^2 \frac{|e_m(n)|^2}{||\mathbf{x}_m(n)||^2} &- 2\mu_m \frac{\mathrm{Re}\{e_m^*(n)t_m(n)\}}{||\mathbf{x}_m(n)||^2}. \end{aligned} \tag{18}$$

Assuming that that $x_m(n)$ and $w_m(n)$, i.e. the driving subband signal and the local subband noise, are independent and zero mean, and using the relation

$$e_m(n) = t_m(n) + w_m(n), \tag{19}$$

the difference in mean square deviation from one update to the next is given by

$$\begin{aligned} &\mathcal{D}_m(n+1) - \mathcal{D}_m(n) = \\ &\quad \mu_m(\mu_m - 2)\mathbb{E}\Big[\frac{|t_m(n)|^2}{||\mathbf{x}_m(n)||^2}\Big] + \mu_m^2 \mathbb{E}\Big[\frac{|w_m(n)|^2}{||\mathbf{x}_m(n)||^2}\Big]. \end{aligned} \tag{20}$$

Further, assuming small fluctuations in the input energy $||\mathbf{x}_m(n)||^2$ from one iteration to the next gives [1]

$$\mathcal{D}_m(n+1) - \mathcal{D}_m(n) = \mu_m(\mu_m - 2)\frac{\mathbb{E}[|t_m(n)|^2]}{\mathbb{E}[||\mathbf{x}_m(n)||^2]} + \mu_m^2 \frac{\mathbb{E}[|w_m(n)|^2]}{\mathbb{E}[||\mathbf{x}_m(n)||^2]}. \quad (21)$$

From this expression it can be seen that the first term contributes to a deviation reduction under the assumption that $0 < \mu_m < 2$. Moreover, a large $\mathbb{E}[|t_m(n)|^2]$ gives a significant deviation decrease. On the other hand, a large noise level $\mathbb{E}[|w_m(n)|^2]$ counteracts the deviation decrease and could even cause an increase.

Based on these observations and with equation (13) in mind, it is clear that updating the subband filter corresponding to the largest output error magnitude, disregarding the noise, will cause the largest possible fullband filter deviation reduction. To minimize the impact of the noise, stationary noise could be estimated in each subband and then subtracted from the output error prior to deciding which subband to update. Estimation of this noise could be done in a number of ways, e.g. using minimum statistics, or schemes with fast and slow estimators [3]. Non-stationary noise could e.g. in an echo cancellation scenario be detected by a *doubletalk detector* (DTD)[3]. The DTD would then indicate when it is safe to update the filter. Details on noise estimation is, however, out of scope for this paper.

Once the estimated subband noise level, denoted $\hat{\sigma}^2_{w_m}(n)$, is obtained, this variable can be subtracted from the squared subband error to obtain an estimate of $t^2_m(n)$. However, as a margin for minor noise estimation errors, a constant T_n is multiplied with $\hat{\sigma}^2_{w_m}(n)$ prior to the subtraction. Thus,

$$\hat{t}^2_m(n) = e^2_m(n) - T_n\hat{\sigma}^2_{w_m}(n). \quad (22)$$

From equation (22), it can be seen that the setting of T_n controls how much influence the noise is allowed to have on the selection of which subband filter to update.

Finally, which subband filter to be updated is decided through

$$i = \arg\max_m \frac{\hat{t}^2_m(n)}{||\mathbf{x}_m(n)||^2} \quad m \in \{0, \cdots, M/2\}. \quad (23)$$

Thus, the proposed algorithm could be summarized as follows

1: Estimate the noise level $\hat{\sigma}^2_{w_m}(n)$ in all subbands.

2: Before every filter update, calculate an estimate of $t_m^2(n)$ as in equation (22) for each subband, where T_n is a pre-defined constant.
3: Calculate equation (23) and update subband filter $\hat{\mathbf{h}}_i(k)$.
4: Perform FFT-difference stacking (as described in the beginning of this section) to construct an updated fullband filter.

6 Mean Square Deviation Analysis

To analytically show the benefits of the proposed algorithm, mean square deviation expressions for uncorrelated input samples for the standard NLMS, periodic NLMS and the proposed updating method are presented and compared.

6.1 Subband mean square deviation for NLMS

Considering only the deviation of a single constantly updating subband m and inserting $e_m^* = \mathbf{x}_m^H(n)\mathbf{v}_m(n) + w_m^*(n)$ and equation (6) into equation (16) and taking expectation gives

$$\mathbb{E}[\mathbf{v}_m^H(n+1)\mathbf{v}_m(n+1)] = \mathbb{E}\left[\left|\left|\left(\mathbf{I} - \mu_m \frac{\mathbf{x}_m(n)\mathbf{x}_m^H(n)}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}\right)\mathbf{v}_m(n) - \mu_m \frac{w_m^*(n)\mathbf{x}_m(n)}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}\right|\right|^2\right], \quad (24)$$

where $\mathbf{I}$ is the identity matrix with dimensions $N_M \times N_M$. Again using the assumption that $x_m(n)$ and $w_m(n)$ are uncorrelated and zero mean allows reduction to

$$\begin{aligned}\mathbb{E}[\mathbf{v}_m^H(n+1)\mathbf{v}_m(n+1)] = & \\ \mathbb{E}\big[\mathbf{v}_m^H(n)\big(\mathbf{I} - \mu_m(2-\mu_m)\frac{\mathbf{x}_m(n)\mathbf{x}_m^H(n)}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}\big)\mathbf{v}_m(n)\big] + & \\ + \quad \mu_m^2 \mathbb{E}\big[\frac{|w_m(n)|^2}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}\big]. & \quad (25)\end{aligned}$$

Using the independence assumption [18], i.e. that $\mathbf{x}_m(n)$ and $\mathbf{v}_m(n)$ are independent, and assuming that the individual entries of $\mathbf{x}_m(n)$ are uncorrelated allows separate evaluation of $\mathbb{E}[\frac{\mathbf{x}_m(n)\mathbf{x}_m^H(n)}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}]$ as $\mathbb{E}[\frac{|x_m(n)|^2}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}]\mathbf{I}$ and equa-

tion (25) can be rewritten as

$$\begin{aligned}\mathcal{D}_m(n+1) = & \\ & \left(1-\mu_m(2-\mu_m)\mathbb{E}\big[\frac{|x_m(n)|^2}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}\big]\right)\mathcal{D}_m(n) + \\ + \quad & \mu_m^2\mathbb{E}\big[\frac{|w_m(n)|^2}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}\big]. \end{aligned} \tag{26}$$

The assumption of small input signal energy fluctuations from one iteration to the next allows the approximation $\mathbb{E}[\frac{|x_m(n)|^2}{\mathbf{x}_m^H(n)\mathbf{x}_m(n)}] \approx \frac{\mathbb{E}[|x_m(n)|^2]}{\mathbb{E}[\mathbf{x}_m^H(n)\mathbf{x}_m(n)]}$ [1] which leads to

$$\mathcal{D}_m(n+1) = \left(1-\frac{\mu_m(2-\mu_m)}{N_M}\right)\mathcal{D}_m(n) + \frac{\mu_m^2}{N_M}\frac{\sigma_{w_m}^2}{\sigma_{x_m}^2}, \tag{27}$$

where $\sigma_{x_m}^2 = \mathbb{E}[|x_m(n)|^2]$ and $\sigma_{w_m}^2 = \mathbb{E}[|w_m(n)|^2]$. It is then obvious that by letting $n \to \infty$, the steady state deviation becomes

$$\mathcal{D}_m(\infty) = \frac{\mu_m}{2-\mu_m}\frac{\sigma_{w_m}^2}{\sigma_{x_m}^2}. \tag{28}$$

6.2 Subband mean square deviation for periodic NLMS

Now, in the case of periodic updating every $P = M/2+1$:th subband sample, the expression for the periodic NLMS becomes

$$\mathcal{D}_m(n+P) = \left(1-\frac{\mu_m(2-\mu_m)}{N_M}\right)\mathcal{D}_m(n) + \frac{\mu_m^2}{N_M}\frac{\sigma_{w_m}^2}{\sigma_{x_m}^2}. \tag{29}$$

From equation (29) it can be seen that the steady state in equation (28) also holds for the periodic NLMS, and that stability is ensured for $0 < \mu_m < 2$ just as for equation (27). It is clear that the convergence speed will be reduced with a factor P, still under the assumption of uncorrelated input samples, compared to the full updating scheme.

6.3 Subband mean square deviation for the proposed algorithm

For the proposed algorithm, the updating scheme of subband m will depend on the input signal as well as the state of the adaptive filters in the other subbands. Assuming that the subband filters will update approximately equally

often, the considered filter $\hat{\mathbf{h}}_m$ will on average update every P:th subband sample, just as for the periodic approach. However, the statistical distribution of the input will be different; in the periodic case the original distribution of the input samples is maintained, but not for the proposed approach. In the event of an update of subband m, and disregarding the noise, $|e_i(n)|^2 \quad i \in \{0, \cdots, M/2\}$ is largest for $i = m$.

If it is assumed that the error contribution from the filter mismatch is significantly larger than the local noise, and the spectral content of the input signal $x_m(n)$ is essentially flat over a frequency band larger than that occupied by each element of the deviation vector $\mathbf{v}_m(n)$, the mean square error can be approximated as [1]

$$\begin{aligned} \mathbb{E}[|e_m(n)|^2] &= \mathbb{E}[|\mathbf{v}_m^H(n)\mathbf{x}_m(n)|^2] + \mathbb{E}[|w_m(n)|^2] \\ &\approx \mathbb{E}[||\mathbf{v}_m(n)||^2]\mathbb{E}[|x_m(n)|^2] \\ &= \mathcal{D}_m(n)\mathbb{E}[|x_m(n)|^2]. \end{aligned} \tag{30}$$

Further, the assumption of all subband filters on average updating equally often gives that the filter deviation of all subband filters are approximately equal, i.e. $\mathcal{D}_i(n) \approx \mathcal{D}_j(n) \quad i, j \in \{0, \cdots, M/2\}$. Thus, defining $e_{\text{cur}}(n)$ as the error of the filter which is updated at the current subband sample index n, the mean square error of the filter to be updated is

$$\mathbb{E}[|e_{\text{cur}}(n)|^2] = \mathcal{D}_{\text{cur}}(n)\mathbb{E}[\max_i |x_i(n)|^2] \quad i \in \{0, \cdots, M/2\}. \tag{31}$$

Now, again considering the subband m, it is obvious that $\mathbb{E}[|e_{\text{cur}}(n)|^2] = \mathbb{E}[|e_m(n)|^2]$ when updating subband filter m. Also, under the assumption given above, each subband filter is updated approximately every P:th subband sample. Using this, and inserting equation (31) into equation (25) yields

$$\mathcal{D}_m(n+P) = \left(1 - \frac{\mu_m(2-\mu_m)}{N_M}\frac{\sigma_{x_\infty^2}}{\sigma_{x_m}^2}\right)\mathcal{D}_m(n) + \frac{\mu_m^2}{N_M}\frac{\sigma_{w_m}^2}{\sigma_{x_m}^2}, \tag{32}$$

where $\sigma_{x_\infty}^2 = \mathbb{E}[\max_i |x_i(n)|^2] \quad i \in \{0, \cdots, M/2\}$. From equation (32) it can be seen that since obviously $\frac{\sigma_{x_\infty}^2}{\sigma_{x_m}^2} \geqslant 1$, the convergence speed of the proposed updating scheme is in general higher than for the periodic NLMS (equation (29)). It can also be seen that if $\frac{\sigma_{x_\infty}^2}{\sigma_{x_m}^2} < N_M$, the stability is ensured for $0 < \mu_m < 2$. The first condition holds under essentially all practical circumstances since typically N_M is fairly large (especially for acoustic echo cancellation) while $\sigma_{x_\infty}^2$ and $\sigma_{x_m}^2$ generally are in the same order of magnitude.

However, equation (32) only describes the the deviation of the proposed algorithm during the initial converging phase when the filter mismatch component of the squared error $|e_m(n)|^2$ (see equation (30)) is larger than the noise component, i.e. when $\mathcal{D}_m(n) > \frac{\sigma^2_{w_m}}{\sigma^2_{x_m}}$. In this case, the selection of which subband filter to update will be optimal in the sense that the subband filter which will contribute most to reducing the total deviation will be updated. After convergence, however, the selection of which subband filter to update will also depend on the noise components in the different bands. Assuming that the noise influence will disturb the subband selection so that the selection will be in a totally random manner gives that equation (29) better describes the proposed algorithm after convergence. Incorporating this property into equation (32) yields

$$\mathcal{D}_m(n+P) = \left(1 - \frac{\mu_m(2-\mu_m)}{N_M} q_m(n)\right) \mathcal{D}_m(n) + \frac{\mu_m^2}{N_M} \frac{\sigma^2_{w_m}}{\sigma^2_{x_m}}, \qquad (33)$$

where

$$q_m(n) = \begin{cases} \frac{\sigma_{x^2_\infty}}{\sigma^2_{x_m}} & \text{if} \quad \mathcal{D}_m(n) > \frac{\sigma^2_{w_m}}{\sigma^2_{x_m}} \\ 1 & \text{otherwise.} \end{cases} \qquad (34)$$

6.4 Reconstructed fullband filter mean square deviation

Updating one adaptive subband filter will, since the individual bands in the oversampled filterbank (shown in figure 3) are overlapping, also affect the future input error to its neighbors due to the closed-loop structure (see figure 2). This means that the convergence speed of the neighboring adaptive filters will be affected. Experiments have shown that for a high order ($K = 8192$) near-ideal 2-times oversampled filterbank with 100% frequency overlap (i.e. two filter bands overlap at every frequency), equation (13) holds fairly well. In this case, the spectrum of the signal will be fairly flat within each subband if the fullband signal spectrum is flat. However, this is not true if the filter order is changed so that the amount of overlap is decreased (e.g. as in figure 3). As the filter bands become narrower, although remaining flat in the frequency region corresponding to the components used in the FFT-2-stacking, the convergence speed of the fullband filter is increased. It seems like this amount of increase can be up to a factor of about 3/2 in practice, which should be taken account for by multiplying the right hand side of equation (13) by 2/3. Mathematical analysis of this effect is a subject of further studies.

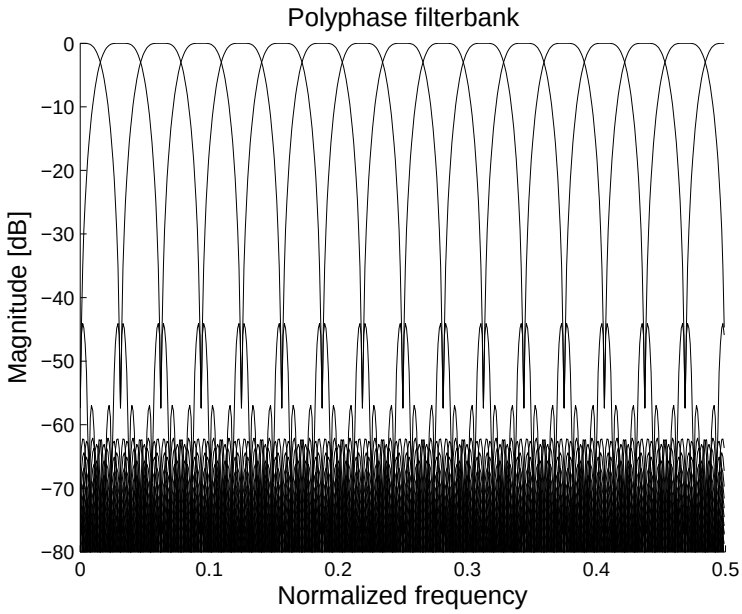

Figure 3: Frequency response of polyphase FFT filterbank.

7 Computational complexity

The delayless adaptive filtering can be divided into four steps; subband filtering, subband filter updating, FFT-2 stacking and the signal path fullband filtering. For the sake of simplicity, the computational complexity considered here is measured in multiplications and divisions per sample.

7.1 Subband filtering

For M subbands, the 2-times oversampled polyphase FFT requires one K-coefficient prototype filter convolution and one M point FFT for each set of $M/2$ input samples. An M-point real FFT requires about $M \log_2 M$ multiplications, giving a total of [6]

$$2(K/M + 2\log_2 M) \tag{35}$$

real multiplications per input sample.

7.2 Subband filter updating

Since the input signal is real, only $M/2 + 1$ subbands need to be considered. Updating one subband adaptive filter requires $4N_M$ multiplications and one division. In the full updating case, the number of multiplications per sample then is

$$(M/2 + 1)8N_M/M \tag{36}$$

and for the case where only one subband is updated, i.e. the periodic scheme and the proposed algorithm,

$$8N_M/M. \tag{37}$$

In the full updating case, $M/2 + 1$ real divisions are also required by the subband filter updating (see equation (5)). The proposed algorithm which updates only one subband requires only one division for updating. However, an additional $M/2 + 1$ real divisions are required for the subband selection in equation (23), resulting in a total of $M/2+2$ real divisions. Thus, the number of divisions per input sample is $1+2/M$ in the full updating case and $1+4/M$ for the proposed algorithm.

Furthermore, for the proposed algorithm, there are $M/2+1$ comparisons/max-operations (for finding the largest value) and $M/2 + 1$ multiplications (for calculating the squared errors), yielding

$$1 + 2/M \tag{38}$$

additional multiplications per input sample. The imposed need of noise estimation in each subband by the proposed algorithm has less importance to the complexity, since the estimation is performed when there is no input signal present (i.e. only local noise), hence not at the same time as the adaptive filter updating.

7.3 FFT-2 stacking

The $2N_M$-point complex FFT for each subband, which is originally of length N_M and zeropadded up to $2N_M$, requires $4N_M \log_2(2N_M)$ real multiplications. Since the echo path is assumed to be real, only $M/2+1$ subbands need to be considered. Thus, in the full updating case, a total of $(M/2+1)4N_M \log_2(2N_M)$ real multiplications is required for calculating all the DFT coefficients, which are stacked to form the fullband filter. The fullband filter is then calculated using an complex-to-real inverse FFT, requiring $N \log_2(2N)$ real multiplications (half of a full complex-to-real inverse FFT, since only the N first time domain filter coefficients are calculated). This gives the total number of real multiplications for the FFT-2 stacking as

$$(M/2+1)4N_M \log_2(2N_M) + N \log_2(2N). \tag{39}$$

For the periodic updating scheme and the proposed algorithm, only one subband filter is updated each subband sample. This means that only one subband filter has changed since the last sample, and it is possible to use the FFT-2-difference stacking previously described. Hence, it is only necessary to compute one $2N_M$-point real FFT for the updated subband filter, i.e. $4N_M \log_2(2N_M)$ real multiplications are required. Further, when constructing the fullband filter, the number of real multiplications needed is $N \log_2 N_M$, since only N_M components are non-zero, and only the N first coefficients are considered. The total number of real multiplications for the FFT-2-difference stacking is then

$$4N_M \log_2(2N_M) + N \log_2 N_M. \tag{40}$$

Since the FFT-stacking is relatively computationally demanding, [6] suggested to only perform this stacking every N/J input samples. This is motivated through the fact that the fullband filter cannot change much faster than the length of its impulse response. The computational complexity of the FFT-2 stacking is then

$$J\left((M/2+1)4N_M \log_2(2N_M) + N \log_2(2N)\right)/N, \tag{41}$$

which can be rewritten as

$$J\left(\log_2(2N_M) + \log_2(2N) + \frac{4N_M \log_2 2N_M}{N}\right) \tag{42}$$

real multiplications per input sample.

The FFT-2-difference stacking can of course also be performed only every N/J input samples, yielding the total number of real multiplications per input sample as

$$J\left(4\frac{N_M}{N}\log_2(2N_M) + \log_2 N_M\right). \tag{43}$$

7.4 Signal path fullband filtering

The signal path fullband filtering can be performed through *fast convolution* [19]. The fullband filter coefficients are divided into L blocks, where the first block is processed with direct convolution, resulting in N/L real multiplications per input sample and the remaining blocks are processed in the DFT-domain. The DFT-domain processing requires a $2N/L$-point real FFT of each of the remaining $L-1$ blocks, as well as one $2N/L$-point real FFT of the block of N/L input samples, $(L-1)N/L$ complex multiplications in the frequency domain and one $2N/L$-point complex-to-real inverse FFT. This gives [6]

$$N/L + 2(L-1)\log_2(2N/L) + 4(L-1) + 4\log_2(2N/L) \tag{44}$$

real multiplications per input sample, which reduces to

$$N/L + 2(L+1)\log_2(2N/L) + 4(L-1) \tag{45}$$

real multiplications per input sample.

7.5 Examples

Considering a fullband filter of length $N = 512$, $M = 32$ subbands and a prototype filter of length $K = 128$ and a decimation ratio of $R = 16$ and $J = 4$, the computational complexity in the full updating case will be $18 + 136+142+218 = 514$ (equations (35), (36), (42) and (45)) real multiplications per input sample, whereas for the case where only one subband is updated every subband sample will be $18 + 8 + \frac{33}{32} + 26 + 218 \approx 271$ (equations (35), (37), (38), (43) and (45)) real multiplications per input sample. The number

of divisions per input sample for the full updating case is $\frac{33}{32}$ and $\frac{9}{8}$ (see section 7.2) for the proposed algorithm. Additionally, $\frac{33}{32}$ max-operations per sample are required for the proposed algorithm.

As can be seen, the significant difference in computational complexity between the considered methods lies in the number of multiplications per input sample. Hence, the proposed algorithm almost halves the computational complexity in this case. For $J = 16$, i.e. updating of the fullband filter after each subband update, the full updating scheme requires $18 + 136 + 568 + 218 = 940$ real multiplications per input sample, while the proposed algorithm requires $18 + 8 + \frac{33}{32} + 104 + 218 \approx 349$ real multiplications per input sample.

It is clear that in these cases, the proposed algorithm requires significantly less complexity as compared to the full updating delayless subband approach. For further comparison, the conventional fullband (N)LMS requires $2N = 1024$ multiplications per sample. Thus, in this case the proposed algorithm requires only about one fourth of the fullband (N)LMS complexity for $J = 4$ and one third of the fullband (N)LMS complexity for $J = 16$.

8 Simulations

To verify the results obtained in the previous section, various simulations were performed. In all simulations, the sampling frequency was 8 kHz. As a first setup, an "ideal" single-input-multiple output (SIMO) setup with four different FIR-filters, each of length 512, were studied. The coefficients of the four filters were realizations of zero mean gaussian random variables with variance 1. A bandlimited flat spectrum signal was used as input (hence, independent input samples as assumed in the analysis in section 6), with zero mean and variance 1. Four independent zero mean bandlimited flat spectrum signals with variance $\sigma^2_{w_m} = 2.5 \times 10^{-3}$ were used as local noise signals. To estimate the SIMO-setup, four adaptive filters of length $N = 512$ were used. Two different updating methods were then compared, one employing the periodic updating schedule and one with the proposed updating scheme. Considering this setup in terms of four different "subbands", i.e. $m = \{0, \cdots, 3\}$, the parameters were $\sigma^2_{x_m} = 1$ and $\sigma^2_{x_\infty} \approx 2.47$ (estimated through Monte Carlo simulation). The squared deviation of one filter (or "subband" $m = 0$), calculated as $||\mathbf{v}_m(n)||^2 = ||\mathbf{h}_{m,\mathrm{opt}} - \hat{\mathbf{h}}_m(n)||^2$ is shown in figure 4. As can be seen, the estimated deviations, obtained through equation (29) and equation (33), follow the simulated deviations fairly well. Moreover, the proposed algorithm converges faster than the periodic updating scheme, and

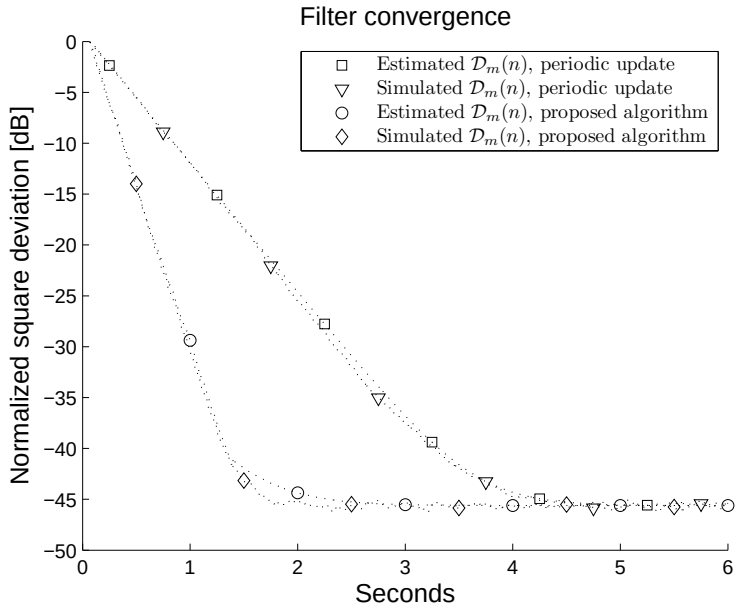

Figure 4: Comparison between the estimated and simulated squared deviation of the proposed methods and the periodic NLMS, respectively. Simulations were performed with an ideal setup, i.e. independent input samples and orthogonal filter coefficient vectors. Estimated deviations were obtained through equation (29) for the periodic updating scheme and equation (33) for the proposed algorithm.

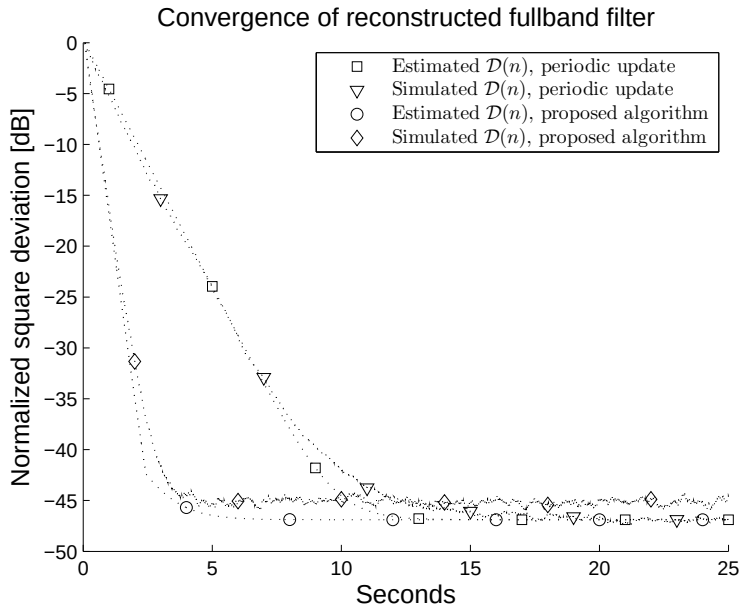

Figure 5: Comparison between the estimated and simulated squared fullband filter deviation of the proposed methods and the periodic NLMS, respectively, in a situation with a flat spectrum signal as input. 32 subbands were used, out of which only one subband filter is updated at each time instant.

reaches the same steady state, as expected. Behavior of filters $m = \{1, 2, 3\}$ is identical.

Next, the performance of the two updating schemes, the periodic and the proposed, were compared in an actual subband setup. In this case the number of subband were chosen as $M = 32$ with a decimation ratio of $M/2 = 16$. A band limited flat spectrum signal with zero mean and variance 1 were used as input signal, resulting in $\frac{\sigma_{x_\infty}^2}{\sigma_{x_m}^2} \approx 3.6$ (again, estimated through Monte Carlo simulation). In this case, the squared deviation of the fullband filters are compared, since the optimal subband filters are not directly available due to the delayless structure. Estimation of the fullband MSD for the periodic and the proposed updating scheme, respectively, is as described in section 6.4. Results are shown in figure 5, again showing the faster convergence of the

Parameter	Value
N	512
R	16
M	32
N_M	32
S	128
J	16
μ	0.5
ϵ	10^{-6}
T_n	0.1

Table 1: Simulation parameter settings.

proposed method compared to the periodic updating. However, the simulated deviation curves show a deviant behavior compared to the estimated curves. This happens below approximately -30 dB, at which point the influence of the subband filter band edges on the adaptive filter deviation start to become significant. The reduction in convergence speed is believed to be caused by the small eigenvalues associated with the band edges of the subband filters [6]. Moreover, the steady state divergence is slightly higher (about 2 dB) for the proposed method. The reason for this is not clear at the moment.

Simulations with real speech signals and real recorded stationary ambient noise with different characteristics were used in further evaluation of the proposed algorithm. Important to note is that the spectrum of the noise in this case is highly non-flat, i.e. the amount of noise in each subband is very different. The noise was recorded in a standard office and originates mainly from computer fans and air-conditioners. Parameters were as shown in table 1.

A simple voice activity detector (VAD) [3] was set to operate in each subband, assuring update only when sufficient echo-to-noise ratio in the subband. In practice, this means that $\hat{t}_m^2(n)$ for all subbands was calculated for both algorithms. Thus, if $\hat{t}_m^2(n)$ is negative, the next subband for which $\hat{t}_m^2(n)$ is not negative is updated instead for the periodic NLMS. If $\hat{t}_m^2(n)$ is negative for all subbands, no subband is updated. For the proposed algorithm, a corresponding approach implies not updating any subband if the largest $\hat{t}_m^2(n)$ is negative.

The results, in the form of squared averaged output error, calculated as $\tilde{e}(n) = \frac{1}{N_i}\sum_{i=1}^{N_i}|e(n-i)|^2$, with $N_i = 4000$ (and $\tilde{y}(n) = \frac{1}{N_i}\sum_{i=1}^{N_i}|y(n-i)|^2$) from both updating methods, are shown in the upper plot of figure 6. Clearly,

it can be seen that the proposed algorithm converges faster than the periodic updating scheme also in this case. In this simulation, comparison with a full updating scheme is also presented. It could also be seen that the convergence of the proposed algorithm is comparable (only slightly slower) than the full updating scheme, albeit much lower computational complexity. In the lower plot of figure 6, the update distribution among the different subband filters for the proposed algorithm is presented. One observation that can be made from this plot is that at the very beginning (0-2 seconds), the updates are primarily concentrated to lower bands. This is since the largest output error magnitudes are originating from those subbands in this case. Then, as the lower bands converge, producing lower output error magnitudes, the higher bands have a chance to update.

This property is also shown in figures 7 and 8. The upper plot of figure 7 shows a simulation with a flat spectrum signal filtered through a bandpass filter, and the lower plot shows the update distribution among the different subband filters for the proposed algorithm in this simulation. It can be seen that the proposed algorithm initially concentrates the updates to the middle bands (bands 7 to 9), containing the largest error magnitudes at this point. The upper plot of figure 8 shows a simulation with a flat spectrum signal filtered through a bandstop filter, and the lower plot shows the update distribution among the different subband filters for the proposed algorithm in this simulation. In this case, the updates are concentrated to the uppermost and lowermost subbands, not allowing subband 8, corresponding to the notch of the stopband filter, to update. Subband 7 and 9, corresponding to the band edges, are allowed to update after about 1 second, i.e. after the other bands have reached a sufficient level of convergence.

9 Off-line calculations

Evaluation using real signals, recorded in a normal office with a loudspeaker and a microphone, were also performed. In this case, the adaptive filter length was changed to $N = 1024$ and thus $N_M = 64$ to be able to model the long impulse response of the room. Moreover, the parameters $\mu = 0.75$ and $\epsilon = 10^{-5}$ were used. The result is presented in figure 9, where the averaged square output error from the full updating, periodic and the proposed scheme is shown. Like in the previous simulations, the full updating scheme displays only slightly faster convergence compared to the proposed scheme, while the convergence of the periodic updating scheme is significantly slower.

10 Conclusions

In this paper, a method for reduced computational complexity of delayless subband adaptive filters has been presented. An analytical expression for the mean square deviation for the proposed algorithm in a situation with uncorrelated input samples has also been presented, and verified through simulations. Comparison between the proposed algorithm and a periodic updating scheme, in both artificial situations with flat spectrum signals and speech, as well as in real situations with speech signals in an acoustic echo cancellation setup, shows the advantage of the proposed algorithm in terms of convergence speed. Moreover, comparison with a full updating delayless scheme shows that the proposed solution exhibits only slightly slower convergence, while requiring only about half of the computational complexity.

Implementation of the proposed algorithm in a fixed point environment would be straight-forward, except possible implications in the balance between keeping sufficient precision and avoid saturation of the FFTs and the inverse FFTs. However, it is believed that this can be handled through appropriate (perhaps dynamic) scaling, and is a subject of further study.

References

[1] S. Haykin, *Adaptive Filter Theory*, 4th ed. Prentice-Hall, 2002.

[2] S. L. Gay and S. Tavathia, "The fast affine projection algorithm," *In proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 3023–3026, 1995.

[3] E. Hänsler and G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach.* Wiley, 2004.

[4] K. Dogancay and O. Tanrikulu, "Generalized subband decomposition LMS algorithm employing selective partial updates," *In proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 1377–1380, 2002.

[5] S. Hosur and A. Tewfik, "Wavelet transform domain adaptive FIR filtering," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 617–630, 1997.

[6] D. Morgan and J. Thi, "A delayless subband adaptive filter architecture," *IEEE Transactions on Signal Processing*, vol. 43, no. 8, pp. 1819–1830, 1995.

[7] H. Huang and C. Kyriakakis, "Real-valued delayless subband affine projection algorithm for acoustic echo cancellation," *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 259–262, 2004.

[8] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing*, vol. 44, no. 3, pp. 209–216, 1997.

[9] T. Aboulnasr and K. Mayyas, "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Transactions on Signal Processing*, vol. 47, no. 5, pp. 1421–1424, 1999.

[10] K. Dogancay and O. Tanrikulu, "Adaptive filtering with selective partial updates," *IEEE Trans. on Circuits and Systems - II: Analog and Digital Signal Processing*, vol. 48, no. 8, pp. 762–769, 2001.

[11] T. Shertler, "Selective block update of NLMS type algorithms," *In proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, 1998.

[12] R. Brennan and H. Sheikhzadeh, "Advances in subband adaptive filtering using a low-resource oversampled filterbank implementation," *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1473–1477, 2003.

[13] S. Attallah, "The wavelet transform-domain LMS adaptive filter with partial subband-coefficient updating," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 53, no. 1, pp. 8–12, 2006.

[14] S. Weiss, M. Harteneck, and R. Stewart, "On implementation and design of filter banks for subband adaptive systems," *IEEE Workshop on Signal Processing Systems*, pp. 172–181, 1998.

[15] J. Huo, S. Nordholm, and Z. Zang, "New weight transform schemes for delayless subband adaptive filtering," *Global Telecommunications Conference, GLOBECOM*, vol. 1, pp. 197–201, 2001.

[16] F. Lindstrom, C. Schuldt, and I. Claesson, "Efficient multichannel NLMS implementation for acoustic echo cancellation," *EURASIP Journal on Audio, Speech, and Music Processing*, 2007, article ID 78439, 6 pages, doi:10.1155/2007/78439.

[17] C. Schuldt, F. Lindstrom, and I. Claesson, "Low-complexity adaptive filtering implementation for acoustic echo cancellation," *In proceedings of IEEE TENCON*, November 2006.

[18] W. Gardner, "Learning characteristics of stochastic-gradient-descent algorithms: A general study, analysis, and critique," *Signal Processing*, vol. 6, no. 2, pp. 113–133, 1984.

[19] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing.* Prentice-Hall, 1989.

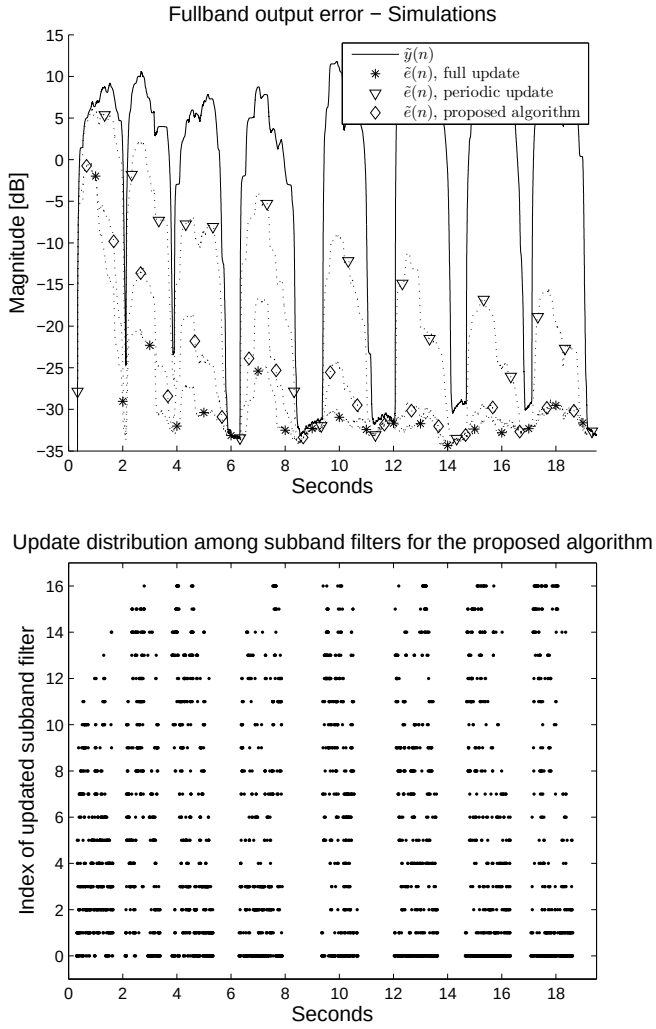

Figure 6: Upper plot shows the output error from the proposed algorithm, the periodic updating scheme and full updating, respectively, with a speech signal as input and simulated impulse response. Lower plot shows the update distribution among subband filters for the proposed algorithm. A dot denotes and update of the corresponding filter.

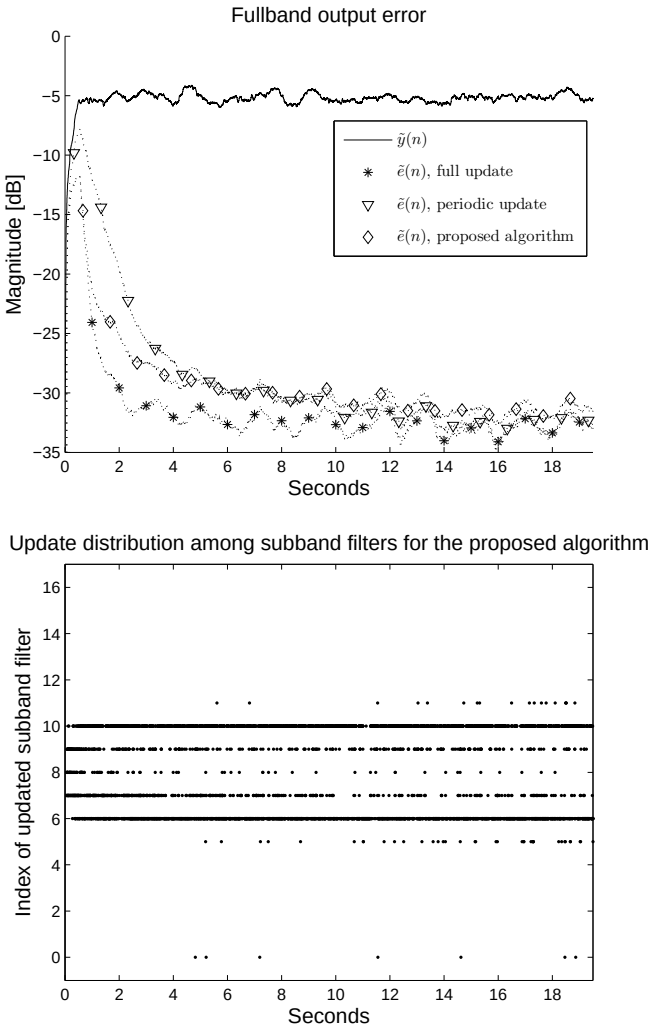

Figure 7: Upper plot shows the output error from the proposed algorithm, the periodic updating scheme and full updating, respectively, with a flat spectrum signal filtered through a bandpass filter as input and simulated impulse response. Lower plot shows the update distribution among subband filters for the proposed algorithm. A dot denotes and update of the corresponding filter.

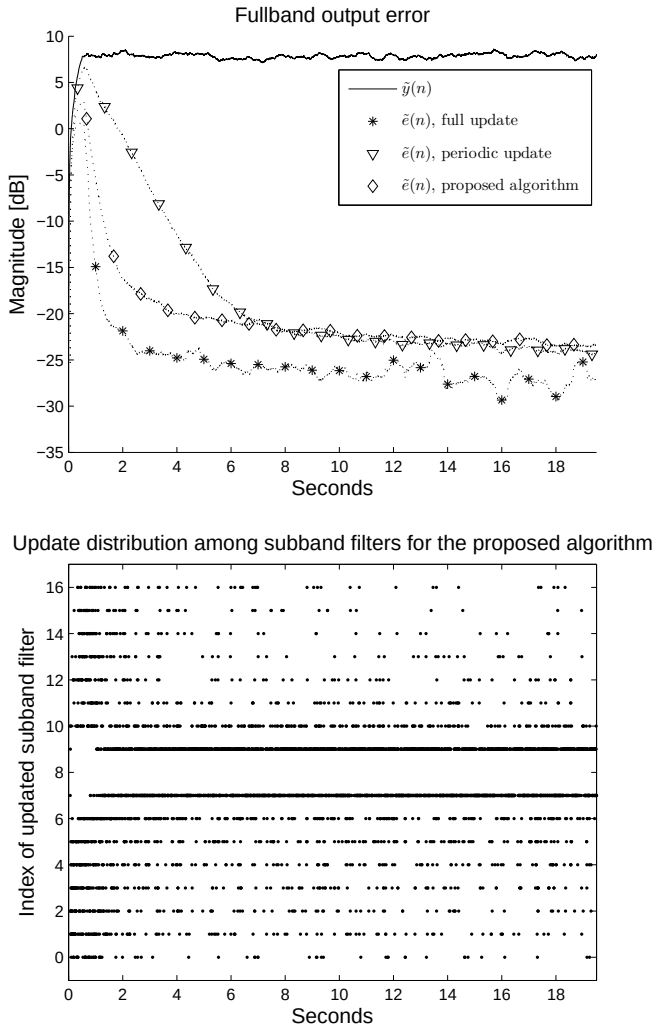

Figure 8: Upper plot shows the output error from the proposed algorithm, the periodic updating scheme and full updating, respectively, with a flat spectrum signal filtered through a bandstop filter as input and simulated impulse response. Lower plot shows the update distribution among subband filters for the proposed algorithm. A dot denotes and update of the corresponding filter.

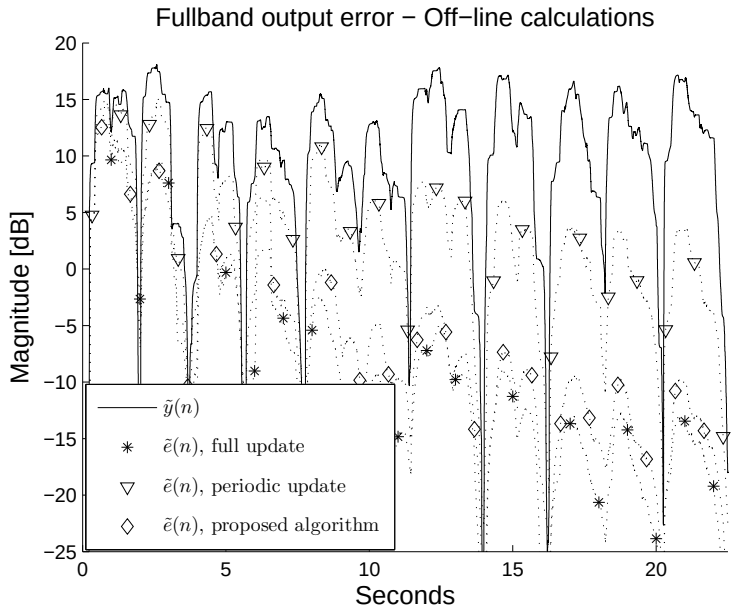

Figure 9: Output error from the proposed algorithm and the periodic updating scheme, respectively, with a speech signal as input and "real" impulse response, i.e. the signals were recorded from an acoustic setup in a normal office.

ABSTRACT

Ever since the birth of the telephony system, the problem with echoes, arising from impedance mismatch in 2/4-wire hybrids, or acoustic echoes where a loudspeaker signal is picked up by a closely located microphone, has been ever present. The removal of these echoes is crucial in order to achieve an acceptable audio quality for conversation. Today, the perhaps most common way for echo removal is through cancellation, where an adaptive filter is used to produce an estimated replica of the echo which is then subtracted from the echo-infested signal.

Echo cancellation in practice requires extensive control of the filter adaptation process in order to obtain as rapid convergence as possible while also achieving robustness towards disturbances. Moreover, despite the rapid advancement in the computational capabilities of modern digital signal processors there is a constant demand for low-complexity solutions that can be implemented using low power and low cost hardware.

This thesis presents low-complexity solutions for echo cancellation related to both the actual filter adaptation process itself as well as for controlling the adaptation process in order to obtain a robust system. Extensive simulations and evaluations using real world recorded signals are used to demonstrate the performance of the proposed solutions.

ISSN 1653-2090

2012:13

ISBN: 978-91-7295-242-3