LOW DELAY FILTER FOR ADAPTIVE NOISE REDUCTION

Heinrich W. Löllmann and Peter Vary

Institute of Communication Systems and Data Processing (ind)
RWTH Aachen University, D-52056 Aachen, Germany
{loellmann|vary}@ind.rwth-aachen.de

ABSTRACT

A filter structure is proposed which has a significantly lower signal delay than common frequency-domain filtering by means of an analysis-synthesis filter-bank (AS FB). The recently published filter-bank equalizer (FBE) is a filter(-bank) with a lower signal delay compared to the corresponding AS FB. In this contribution, the signal delay of the FBE is further reduced by approximating the time-domain filter of the FBE by a filter of lower degree. The proposed low delay filter (LDF) achieves a higher amount of noise reduction than an AS FB or FBE with the same signal delay.

1. INTRODUCTION

Noise reduction systems are today part of many real-time speech processing and speech transmission systems, respectively, such as mobile phones, hands-free telephony, tele-conferencing systems or hearing aids. For such systems, a low signal delay is an important necessity for a convenient and pleasant speech communication.

The enhancement of noisy speech is usually performed by means of spectral-domain filtering, i.e., spectral weighting employing an analysis-synthesis filter-bank (AS FB). For this, the DFT filter-bank (FB) is often used which can be efficiently implemented as polyphase network (PPN) AS FB with the discrete Fourier transform (DFT) calculated by the Fast Fourier Transform (FFT), cf. [1]. This method causes a relative high signal delay as an analysis FB and synthesis FB is needed. An approach to reduce the signal delay of a noise reduction based on spectral weighting (spectral subtraction) is presented in [2].

A different approach to decrease the signal delay due to the filtering is to employ the recently proposed filter-bank equalizer (FBE) [3], [4]. This filter(-bank) has a lower signal delay than the corresponding AS FB with the same number of frequency bands and prototype filter length. The amount of noise reduction achieved by the FBE and the corresponding AS FB is similar [5].

An approach to further reduce the signal delay of the FBE is presented in this contribution. The new low delay filter (LDF) is an extension of the uniform FBE, briefly explained in Sec. 2. The concept of the LDF and different LDF types are introduced in Sec. 3. The properties of the LDF and implementation aspects are discussed in Sec. 4. The performance of the noise reduction by using the LDF is investigated in Sec. 5. A summary is given in Sec. 6.

This work was supported by Siemens Com, Munich, Germany.

2. UNIFORM FILTER-BANK EQUALIZER

The filter-bank equalizer (FBE) [3], [4] is an efficient realization of a low delay filter-bank which has a significantly lower signal delay than the corresponding AS FB. The application of the adaptive FBE for noise reduction with reduced signal delay has been investigated in [5]. This filter structure is shown in Fig. 1 for the case of a uniform frequency resolution.

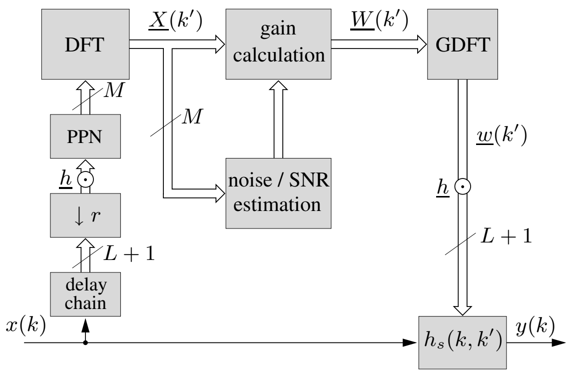

Figure 1: *Adaptive uniform GDFT filter-bank equalizer (FBE) for noise reduction. (Vectors are marked by the underline and the 'circle-dot' denotes an element-wise vector multiplication.)*

The M spectral coefficients $\underline{X}(k')$ are calculated by a DFT polyphase network (PPN) analysis FB. This allows the prototype lowpass filter with impulse response $h(n)$ to be longer than the DFT size M. The variable r denotes the down-sampling rate, where $k = r\,k'$. The calculation of the M spectral gain factors $\underline{W}(k')$ can be done by a common spectral speech estimator, e.g., [6]. The obtained gain factors with $0 \leq W_i \leq 1$ are of zero phase. The generalized discrete Fourier transform (GDFT) of size M provides $L+1$ weighting factors $w_n(k')$, due to its periodicity, with non-zero phase. The filter coefficients of the FBE are finally obtained by

$$h_s(n, k') = h(n)\, w_n(k') \quad ; \quad n = 0, 1, \ldots L\,. \qquad (1)$$

An efficient implementation of the FBE for $L > M$ is given by the PPN FBE [4]. The FBE corresponds to a single time-domain filter with impulse response $h_s(n, k')$, which is obtained by the coefficients $h(n)$ of the prototype lowpass filter and weighting factors $w_n(k')$ adapted in the spectral-domain [4]. On one hand, the FBE needs more multiplications than the corresponding AS FB due to the time-domain filtering at sampling rate. On the other hand, the computation of the gain factors in the spectral-domain is decoupled from the actual filtering in the time-domain. Therefore, no aliasing effects occur and the rate for the computation of the DFT and GDFT (by means of the FFT) is not governed by restrictions for signal reconstruction as for an AS FB. Moreover, the signal delay of the FBE is lower than for the corresponding AS FB since no synthesis FB is needed.

3. LOW DELAY FILTER

3.1. Concept

The signal delay of the (adaptive) FBE shall be reduced. A straight-forward solution is to take a FBE of lower filter degree L. Another approach is to approximate the original time-domain filter of the FBE by a filter of lower degree $P < L$. This new concept will be termed as low delay filter (LDF), and its application for noise reduction is shown in Fig. 2.

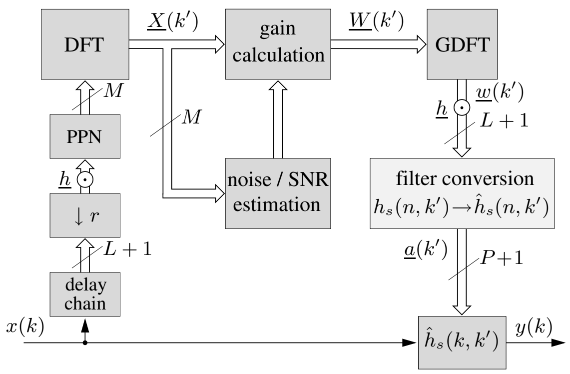

Figure 2: *Low delay filter (LDF) for adaptive noise reduction.*

In contrast to the FBE of Fig. 1, an additional block 'filter conversion' is included which calculates the $P+1$ time-varying coefficients $\underline{a}(k')$ of the LDF from the $L+1$ filter coefficients $h_s(n,k')$ of the FBE. The obtained filter of degree P and with impulse response $\hat{h}_s(n,k')$ can be an FIR filter or IIR filter. In the following, two different filter approximations will be proposed. For the sake of simplicity, the time-dependency of the filter coefficients on k' will be dropped in the sequel.

3.2. Moving-Average Low Delay Filter

The original filter of the FBE can be approximated by an FIR filter of degree $P < L$ following a technique very similar to FIR filter design by windowing, e.g., [1]. The impulse response $h_s(n)$ of Eq. (1) is truncated by a window sequence of length $P+1$ according to

$$\hat{h}_s(n) = h_{\text{MA}}(n) = h_s(n)\,\text{win}_P(n-n_c) \tag{2}$$

with

$$\text{win}_P(n) \begin{cases} \neq 0 & ; \quad 0 \le n \le P \\ = 0 & ; \quad \text{else} \end{cases} . \tag{3}$$

This approximation of the original filter of the FBE by a moving-average (MA) filter results the MA LDF. To ease the treatment, the term (MA) LDF shall refer to the overall system according to Fig. 2, and the term (MA) filter includes only the actual filter with impulse response $\hat{h}_s(n)$. The truncation by a window results a smoothed frequency response of the original filter. The truncation by a rectangular window yields the least-squares approximation error between the impulse response of the original filter and that of the MA filter, e.g., [1]. A more flexible window than the rectangular window is given, e.g., by

$$\text{win}_P(n,\beta) = \begin{cases} \beta + (\beta-1)\cos\left(\frac{2\pi}{P}n\right) & ; \quad 0 \le n \le P \\ 0 & ; \quad \text{else} \end{cases} \tag{4}$$
$$\text{with} \quad 0.5 \le \beta \le 1\,.$$

The rectangular window ($\beta = 1$), the Hanning window ($\beta = 0.5$) and the Hamming window ($\beta = 0.54$) are included as special cases. A non rectangular window results an approximation error which is not optimal in a least-squares sense.

3.3. Auto-Regressive Low Delay Filter

The original FIR filter with impulse response $h_s(n)$ can also be approximated by an IIR filter. The spectral gain factors W_i for noise reduction amplify (mostly) spectral components with high speech energy (high SNR) and vice versa. Thus, these spectral gain factors reflect roughly the (short-term) magnitude spectrum of speech. This motivates the use of an auto-regressive (AR) model, e.g. [7], for the filter approximation.

The system function of an AR filter (allpole filter) of degree P is given by

$$\hat{H}_s(z) = H_{\text{AR}}(z) = \frac{a_0}{1 - \sum\limits_{n=1}^{P} a_n\, z^{-n}}\,. \tag{5}$$

Methods to determine the AR filter coefficients a_n can be taken from parametric spectrum analysis, e.g., [1]. The coefficients of the AR filter are determined by the Yule-Walker equations

$$\begin{bmatrix} \varphi_{\tilde{h}\tilde{h}}(1) \\ \vdots \\ \varphi_{\tilde{h}\tilde{h}}(P) \end{bmatrix} = \begin{bmatrix} \varphi_{\tilde{h}\tilde{h}}(0) & \cdots & \varphi_{\tilde{h}\tilde{h}}(1-P) \\ \vdots & \ddots & \vdots \\ \varphi_{\tilde{h}\tilde{h}}(P-1) & \cdots & \varphi_{\tilde{h}\tilde{h}}(0) \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ \vdots \\ a_P \end{bmatrix} \tag{6}$$

which yields the least-squares error between the original filter and the AR filter approximation. The scaling factor a_0 in Eq. (5) is determined by

$$a_0 = \sqrt{\varphi_{\tilde{h}\tilde{h}}(0) - \sum_{n=1}^{P} a_n\,\varphi_{\tilde{h}\tilde{h}}(-n)} \tag{7}$$

and ensures that the AR filter and the original filter have both the same amplification. The auto-correlation coefficients $\varphi_{\tilde{h}\tilde{h}}$ are calculated here by the weighted auto-correlation method

$$\varphi_{\tilde{h}\tilde{h}}(\lambda) = \sum_{n=0}^{L-|\lambda|} \tilde{h}(n)\,\tilde{h}(n+\lambda) \quad ; \quad 0 \le |\lambda| \le P \tag{8}$$
$$\text{with} \quad \tilde{h}(n) = h_s(n)\,\text{win}_L(n,\beta) \tag{9}$$

and window sequence given, e.g., by Eq. (4). A rectangular window has been taken here to reduce the computational load for evaluating Eq. (8). The auto-correlation method ensures a symmetric Toeplitz structure for the auto-correlation matrix in

Eq. (6) which allows to solve the Yule-Walker equations efficiently by means of the Levinson-Durbin recursion, e.g., [1]. The obtained AR filter is always stable and of minimal phase since the auto-correlation matrix is positive-definite. The use of the covariance method instead of Eq. (8) to calculate the auto-correlation coefficients for Eq. (6) has not been considered here since the obtained AR filter is not necessarily stable.

A general IIR filter (ARMA filter) can also be employed for the LDF. This approach, however, has not been regarded here since the computation of the ARMA filter coefficients is much more complex than for the AR filter (e.g. [1]).

4. IMPLEMENTATION AND PROPERTIES

4.1. Algorithmic Complexity

The algorithmic complexity - in terms of computational complexity and memory consumption - for the presented LDFs is listed in Table 1. The complexity for the gain calculation and

	calculation of $h_s(n,k')$ and MA / AR filtering
multiplications	$\frac{1}{r}(2M\log_2 M + 2L+2)+P+1$
additions	$\frac{1}{r}(3M\log_2 M+L+1-M)+P$
memory	$L+2M+P$
	filter conversion for MA LDF
multiplications	$P+1$
additions	0
memory	0
	filter conversion for AR LDF
multiplications	$\frac{1}{r}\,((P+1)\,(L+4) + P\,(\mathcal{M}_{\text{div}} + \mathcal{M}_{\text{sqrt}}))$
additions	$\frac{1}{r}\,((P+1)\,(L+2) + P\,(\mathcal{A}_{\text{div}} + \mathcal{A}_{\text{sqrt}}))$
memory	$3P$

Table 1: *Algorithmic complexity in terms of required average number of real multiplications and real additions per sample as well as number of delay elements (memory) for a MA LDF and AR LDF of filter degree P.*

SNR estimation in Fig. 2 is independent of the filter type and has not been considered in Table 1. The DFT and GDFT can be calculated by means of the FFT, cf. [1], [5]. The algorithmic complexity for the calculation of the filter coefficients $h_s(n,k')$ and the actual time-domain filtering is equal for both LDFs.

The variable $\mathcal{M}_{\text{div}}$ marks the number of multiplications needed for a division operation, and $\mathcal{M}_{\text{sqrt}}$ represents the number of multiplications needed for a square-root operation, whose values dependent on the numeric procedure and accuracy used to perform these operations. Accordingly, the variables $\mathcal{A}_{\text{div}}$ and $\mathcal{A}_{\text{sqrt}}$ denote the additions needed for a division and square-root operation, respectively. Most of the computational complexity for the AR filter conversion is required to compute the $P+1$ auto-correlation coefficients according to Eq. (8), which is of order PL. A lower computational complexity can be achieved by calculating Eq. (8) by means of the fast convolution or the Rader algorithm [8] with savings dependent on P and L. The MA filter conversion needs no multiplications if a rectangular window is used for Eq. (2).

It should be noted that the AR filter degree can be chosen significantly lower than the MA filter degree for a similar amount of noise reduction as shown later in Sec. 5.

The implementation of the MA / AR filter can be based on different filter structures, e.g., [1]. The choice of the filter structure is important for filters with time-varying coefficients as well as for real filter implementations with finite precision arithmetic. Here, the implementation of the MA / AR filter by the transposed direct form II[1] has been found suitable to avoid filter-ringing. This effect can occur for time-varying filter coefficients and becomes audible by disturbing artifacts. For the direct form, the delay chain in Fig. 2 can be shared by the MA filter such that the MA filter needs not the P delay elements considered in Table 1. The FBE can be regarded as MA LDF with $P = L$. Therefore, the MA filter for $P > M$ can be implemented by a PPN realization similar to the PPN FBE [4]. However, this implementation has not been considered in Table 1 for the sake of simplicity.

4.2. Signal Delay

The signal delay of a filter can be determined for sample-wise processing, e.g., by the average group-delay. A second method used here is to determine the signal delay d_0 according to

$$d_0 = \arg \max_{\lambda \in \mathbb{Z}} \{\varphi_{xy}(\lambda)\} \tag{10}$$

with $\varphi_{xy}(\lambda)$ denoting the cross-correlation sequence between the input sequence $x(k)$ and the output sequence $y(k)$ of the filter. Thereby, the signal delay calculated by this two methods might differ for filters with non-linear phase response.

The signal delay of a MA filter with linear phase and degree P amounts to $P/2$ samples. In contrast, the AR filter regarded in Sec. 5 has a maximal signal delay of only two samples.

5. SIMULATION RESULTS

The described MA LDF and AR LDF have been employed for noise reduction according to Fig. 2 and compared with the FBE of Fig. 1. The noise reduction system is operated at a sampling frequency of 16 kHz. The spectral gain factors are adapted after $r = 128$ samples. The soft-gain MMSE spectral estimator [6] has been used, with an SNR estimation based on minimum statistics [9], to calculate the gain factors. Leopard tank noise from the noisex-92 database has been added to a male speech sequence. The amplification of the noise has been varied to achieve different signal-to-noise ratios (SNRs) for the noisy input speech.

The simulation facilitates to filter the noise and speech sequence separately with filter coefficients adapted for the noisy speech. This allows to calculate the noise attenuation (NA), the speech attenuation (SA), and the cepstral distance (CD) [10]. The SA is the average ratio between the powers of clean speech and processed speech; the NA is the average ratio between the powers of added noise and processed noise. The trade-off between noise attenuation and speech attenuation can be captured by the effective noise attenuation (ENA) which is defined by ENA=NA−SA, where the NA and the SA are expressed by their logarithm. The CD is a frequency-domain distant measure to account for speech distortions. Here, the first 40 cepstral coefficients have been used for the evaluation of the CD. The evaluation of these instrumental measures requires the signal delay d_0.

[1]The nomenclature for different filter structures is inconsistent in literature and those of [1] has been used here.

This has been determined by Eq. (10) using the clean and processed speech due to their stronger correlation. The simulation results are shown in Fig. 3. The highest amount of noise reduc-

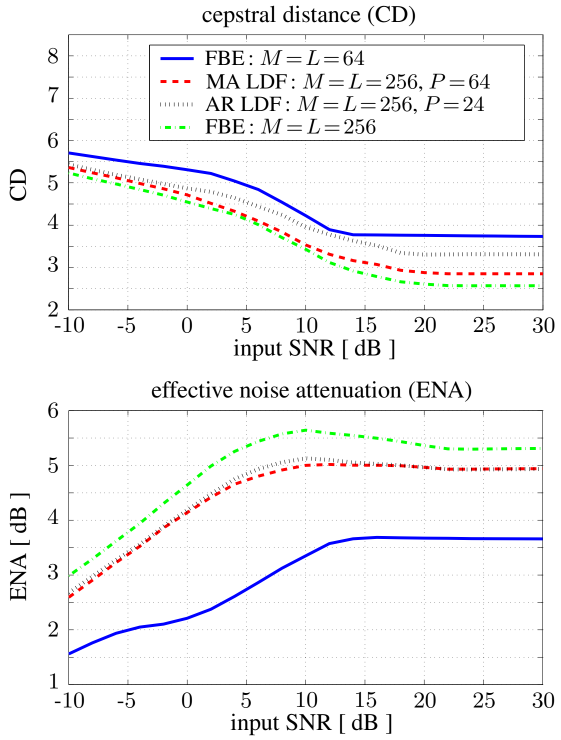

Figure 3: *ENA and CD for different SNRs achieved by means of the LDF and the FBE for tank noise. (The curves for white noise are similar but exhibit a higher ENA.)*

tion - in terms of a low CD and high ENA - is achieved by means of a FBE of degree $L = 256$ and $M = 256$ frequency channels. The approximation of this FBE by a MA LDF or AR LDF, respectively, leads to a slightly decreased noise reduction but significant lower signal delay: The original (linear phase) FBE with $L = M = 256$ causes a signal delay of 128 samples, whereas the (linear phase) MA filter has a delay of 32 samples, and the AR filter has a signal delay of only up to 2 samples dependent on the input SNR. Moreover, the MA LDF has a lower algorithmic complexity than the FBE due to its lower filter degree.

The (linear phase) FBE with $L = M = 64$ has a signal delay of 32 samples which is equal to that of the MA filter. However, the amount of noise reduction achieved by this FBE is lower than for the MA / AR filter. This shows that the LDF approach, which calculates the spectral gain factors with a higher frequency resolution and approximates the resulting time-domain filter by a filter of lower degree, achieves a higher amount of noise reduction than the use of a FBE with a lower degree (lower frequency resolution) to obtain the same signal delay. Moreover, in some cases it might be necessary to adjust the parameters for the gain calculation if the number of frequency channels is altered. This parameter adjustment is not required if the signal delay is reduced by the LDF approach. The higher FFT size M of the regarded MA LDF compared to the FBE with $M = 64$ leads to a moderate increase for the computational complexity as the calculation of $w_n(k')$ is done at a decimated sampling rate.

The LDF has been compared with the FBE here. However, it has been shown in [5] that the commonly used AS FB and the FBE achieve approximately the same noise reduction for the same number of frequency channels M and the same prototype filter length L. The comparison between AS FB and LDF yields similar results for the measured amount of noise reduction, but the the signal delay of the AS FB is much greater than for the corresponding FBE. Thus, the LDF achieves a relative high amount of noise reduction despite a low signal delay. The incorporation of other transforms than the GDFT is easily possible by regarding the generalized FBE [4] as basis for the LDF.

6. CONCLUSIONS

A novel low delay filter (LDF) has been presented which is especially useful for noise reduction systems requiring a low signal delay. The MA LDF is based on an FIR filter and well suited if the algorithmic complexity should be kept low. The AR LDF is based on an IIR filter of minimal phase and achieves a very low signal delay, but requires more computations for the evaluation of its coefficients than the MA LDF. Thus, the presented LDF approach allows to adjust the trade-off of a noise reduction between signal delay, computational complexity and achieved amount of noise reduction in a flexible manner by means of the filter degree and filter type. The LDF achieves a higher amount of noise reduction than a FBE or AS FB with the same signal delay. The use of the LDF for other applications than noise reduction is easily possible and remains as outlook.

7. REFERENCES

[1] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles, Algorithms and Applications*, Prentice-Hall, 1996.

[2] H. Gustafsson, S. E. Nordholm, and I. Claesson, “Spectral Subtraction Using a Reduced Delay Convolution and Adaptive Averaging,” *IEEE Trans. on Speech and Audio Processing*, vol. 9, no. 8, pp. 799–807, Nov. 2001.

[3] P. Vary, “An Adaptive Filter Bank Equalizer for Speech Enhancement,” *accepted for publication in Signal Processing, Elsevier, Special Issue on Applied Speech and Audio Processing*, 2005.

[4] H. W. Löllmann and P. Vary, “Efficient Non-Uniform Filter-Bank Equalizer,” in *Proc. of European Signal Processing Conf. (EUSIPCO)*, Antalya, Turkey, Sept. 2005.

[5] H. W. Löllmann and P. Vary, “Generalized Filter-Bank Equalizer for Noise Reduction with Reduced Signal Delay,” in *Proc. of European Conf. on Speech Communication and Technology (Interspeech - Eurospeech)*, Lisbon, Portugal, Sept. 2005.

[6] R. J. McAulay and M. L. Malpass, “Speech Enhancement Using a Soft-Decision Noise Suppression Filter,” *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 28, no. 2, pp. 137–145, Apr. 1980.

[7] J. R. Deller jr., J. G. Proakis, and J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*, Macmillan Publishing Company, 1993.

[8] C. M. Rader, “An Improved Algorithm for High-Speed Autocorrelation with Application to Spectral Estimation,” *IEEE Trans. on Audio and Electroacoustics*, vol. AU-18, pp. 439–441, Dec. 1970.

[9] R. Martin, “Noise Power Spectral Density Estimation Based on Optimal Smoothing and Minimum Statistics,” *IEEE Trans. on Speech and Audio Processing*, vol. 9, no. 5, pp. 504–512, July 2001.

[10] S. Gustafsson, *Enhancement of Audio Signals by Combined Acoustic Echo Cancellation and Noise Reduction*, Ph.D. thesis, IND, RWTH Aachen University, 1999.