ITERATED COEFFICIENT UPDATES OF PARTITIONED BLOCK FREQUENCY DOMAIN SECOND-ORDER VOLTERRA FILTERS FOR NONLINEAR AEC

Marcus Zeller and Walter Kellermann

Chair of Multimedia Communications and Signal Processing,
University of Erlangen-Nuremberg,
Cauerstr. 7, 91058 Erlangen, Germany
{zeller,wk}@LNT.de

ABSTRACT

This paper presents the benefits of iterated coefficient updates for the adaptation of partitioned block frequency domain second-order Volterra filters when applied to nonlinear acoustic echo cancellation. In order to increase the convergence speed of an NLMS algorithm with separate kernel normalization, each input frame is used for several coefficient updates. This procedure effectively accelerates the convergence of the employed adaptive Volterra filters and is shown to be superior to processing with increased data overlap. The advantages of this novel approach are illustrated by experimental results for noise and speech input and guidelines for determining suitable numbers of iterations for the filter kernels are given.

Index Terms— iterative methods, Volterra filters, echo suppression, adaptive signal processing

1. INTRODUCTION

The task of acoustic feedback suppression is vital to a variety of applications and appropriate algorithms are well-established. However, if the acoustic echo path cannot be modelled by linear components alone, nonlinear acoustic echo cancellation (NLAEC) becomes desirable. The basic scenario of such an NLAEC is depicted in Fig. 1.

In [1] it has been shown that nonlinear distortions which originate in small-scale, low-cost loudspeakers driven at high volume can be compensated adequately by Volterra filters (VF) of second order. However, the convergence of such adaptive structures is significantly slowed down by correlated input and insufficient excitation of the LMS-type coefficient updates. The latter is especially true for the quadratic Volterra kernel which models the nonlinear components of the echo signal that are highly dependent on the signal's amplitude and therefore usually excited only intermittently. Due to the non-stationary nature of speech, the signal power varies strongly for different segments of the signal and thus it seems desirable to fully exploit the excitation power of the nonlinear distortions in order to increase the speed of convergence.

This contribution proposes to employ an iterative update procedure as already applied for the case of linear filtering [2]. By doing so, the same input frame of an overlap-save block processing is repeatedly filtered in order to adjust the adaptive coefficients significantly more than in the single-update case. At first, an introduction into the structure of partitioned block second-order Volterra filters is presented in Section 2 and a concise overview of the conventional frequency domain NLMS adaptation for these nonlinear filters is given in Section 3. The extension towards an iterated update procedure is presented in Section 4 along with some considerations concerning the computational efforts. Section 5 presents experimental results for noise and speech input.

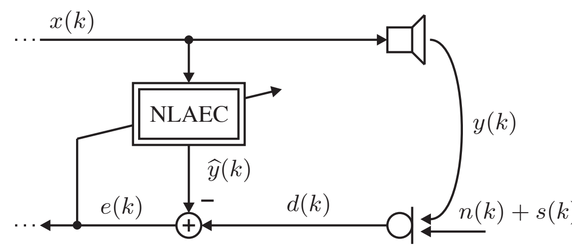

Fig. 1. NLAEC scenario where the nonlinear echo $y(k)$ is to be compensated by an adaptive second-order Volterra filter

2. PRELIMINARIES

As background for the iterated update approach, we briefly review the efficient frequency domain realization of adaptive second-order VFs and the corresponding adaptation techniques. However, for a thorough understanding of these *partitioned block frequency domain adaptive Volterra filters (PBFDAVF)*, the reader is referred to [1] for a detailed presentation.

According to [1], the DFT domain output block $\widehat{Y}_\nu(m)$ of the PBFDAVF at time frame ν is given as superposition of all corresponding linear and quadratic kernel partition outputs

$$\begin{aligned}\widehat{Y}_\nu(m) &= \widehat{Y}_{\nu,1}(m) + \widehat{Y}_{\nu,2}(m) \\ &= \sum_{b_1=0}^{B_1-1} \widehat{Y}_{\nu,b_1}(m) + \sum_{b_{21}=0}^{B_2-1} \sum_{b_{22}=0}^{B_2-1} \widehat{Y}_{\nu,b_{21},b_{22}}(m), \qquad (1)\end{aligned}$$

where the partition size N and the number of filter partitions B_1, B_2 are chosen such that $N_1 = B_1 N$ and $N_2 = B_2 N$ holds for the total memory lengths N_1, N_2 of the VF. Thereby, the output of the partition b_1 of the linear kernel reads

$$\widehat{Y}_{\nu,b_1}(m) = \widehat{H}_{\nu,b_1}(m)\, X_{\nu,b_1}(m), \qquad (2)$$

which corresponds to the well-known fast convolution by multiplication of the 1D-DFT $\widehat{H}_{\nu,b_1}(m)$ of the filter partition and the input

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under contract number KE890/2.

DFT block $X_{\nu,b_1}(m)$. On the other hand, the quadratic kernel's partition outputs are specified by

$$\widehat{Y}_{\nu,b_{21},b_{22}}(m) \;=\; \frac{1}{M}\sum_{\widetilde{m}=0}^{M-1}\widehat{H}_{\nu,b_{21},b_{22}}\big(\widetilde{m},[m-\widetilde{m}]_M\big)\; X_{\nu,b_{21}}(\widetilde{m})\,X_{\nu,b_{22}}\big([m-\widetilde{m}]_M\big), \tag{3}$$

where multi-dimensional filtering techniques are applied as derived in [3] due to the 2D-DFTs $\widehat{H}_{\nu,b_{21},b_{22}}\big(m_{21},m_{22}\big)$. Note that $[...]_M$ denotes a modulo operation w.r.t. the DFT size M.

The input data of the overlap-save method is extracted for all necessary blocks b and frame indices ν as

$$x_{\nu,b}(\kappa) \;:=\; x\big(\nu\,L+\kappa-(M-L)-b\,N\big) \tag{4}$$

where the time index $0\le\kappa\le M-1$ results in an effective frame shift by L samples. This shift is required to be smaller or equal to N and may also be expressed by means of a so-called *overlap factor* $\rho := \frac{N}{L}$ which specifies the amount of overlapping samples between successive frames. The resulting time frames of the filter output are finally obtained by

$$\widehat{y}_\nu(\kappa) \;=\; \mathrm{IDFT}_M\Big\{\widehat{Y}_\nu(m)\Big\} \tag{5}$$

$$\widehat{y}(\nu\,L+l) \;=\; \widehat{y}_\nu\big(M-L+l\big) \tag{6}$$

where $0\le l\le L-1$ captures only the L most recent time instants of the output frame $\widehat{y}_\nu(\kappa)$. This results from the fact that the DFT domain multiplications in (2) and (3) correspond to circular convolutions in the time domain.

In order to express the PBFDAVF operation in a compact manner, (1) is reformulated in vector notation which yields

$$\begin{aligned}\widehat{Y}_\nu(m) \;&=\; \Big[\widehat{\mathbf{H}}^{\mathrm{T}}_{\nu,1}(m),\widehat{\mathbf{H}}^{\mathrm{T}}_{\nu,2}(m)\Big]\Big[\mathbf{X}^{\mathrm{T}}_{\nu,1}(m),\mathbf{X}^{\mathrm{T}}_{\nu,2}(m)\Big]^{\mathrm{T}}\\ &=\; \widehat{\mathbf{H}}^{\mathrm{T}}_{\nu}(m)\,\mathbf{X}_\nu(m),\end{aligned} \tag{7}$$

where the contributing bins of the filter partitions are described by

$$\widehat{\mathbf{H}}_{\nu,1}(m) \;:=\; \Big[...,\widehat{H}_{\nu,b_1}(m),...\Big]^{\mathrm{T}} \tag{8}$$

$$\widehat{\mathbf{H}}_{\nu,2}(m) \;:=\; \Big[...,\widehat{H}_{\nu,b_{21},b_{22}}(\widetilde{m},\big[m-\widetilde{m}\big]_M),...\Big]^{\mathrm{T}} \tag{9}$$

and relevant bins of the input spectra are captured as

$$\mathbf{X}_{\nu,1}(m) \;:=\; \Big[...,X_{\nu,b_1}(m),...\Big]^{\mathrm{T}} \tag{10}$$

$$\mathbf{X}_{\nu,2}(m) \;:=\; \frac{1}{M}\Big[...,X_{\nu,b_{21}}(\widetilde{m})\,X_{\nu,b_{22}}(\big[m-\widetilde{m}\big]_M),...\Big]^{\mathrm{T}}. \tag{11}$$

Note that the necessary division by M is incorporated into the definition of $\mathbf{X}_{\nu,2}(m)$ for presentational convenience.

3. CONVENTIONAL NLMS ADAPTATION

As can be seen by the definitions in (8) and (9), the Volterra filtering of (7) is linear w.r.t. its kernel bins and therefore a frequency domain adaptation of the VF can be performed by applying standard LMS approaches [4]. At first, we regard the samples of the ν-th residual error frame

$$\widetilde{e}_\nu(\kappa) \;=\; \widetilde{d}_\nu(\kappa)-\widehat{y}_\nu(\kappa) \tag{12}$$

where the frames of the microphone reference are extracted analogously to (4) as

$$\widetilde{d}_\nu(\kappa) \;:=\; d\big(\nu\,L+\kappa-(M-L)\big). \tag{13}$$

Due to the temporal aliasing, which is contained in $\widehat{y}_\nu(k)$ and is caused by the overlap-save technique, the desired error frames $e_\nu(\kappa)$ are furthermore constructed such that

$$e_\nu(\kappa) \;:=\; \begin{cases} 0\;, & 0\le\kappa\le M-N-1\\ \widetilde{e}_\nu(\kappa), & M-N\le\kappa\le M-1\end{cases}, \tag{14}$$

and thus the first samples are discarded. Correspondingly, the error spectra of (6) are merely based on the N most recent time instants of the filter output:

$$E_\nu(m) \;=\; \mathrm{DFT}_M\Big\{e_\nu(\kappa)\Big\}. \tag{15}$$

Note that in case of overlapped processing with $\rho>1$, (14) provides a robust estimation of the DFT domain error $E_\nu(m)$, since it is always based on the last N samples of (12), although only the most recent $L<N$ time instants contain new information.

If an instantaneous estimate of the mean squared error (MSE) gradient is employed [4], the general LMS update equation for both kernels ($p=1,2$) of the PBFDAVF is given by

$$\widehat{\mathbf{H}}_{\nu+1,p}(m) \;=\; \widehat{\mathbf{H}}_{\nu,p}(m)+\mu_{\nu,p}(m)\,\mathcal{C}\Big\{E_\nu(m)\,\mathbf{X}^*_{\nu,p}(m)\Big\} \tag{16}$$

which affects all DFT domain coefficients which contribute to the frequency bin m. Here, * denotes conjugate complex and $\mathcal{C}\{...\}$ represents a constraint function comprising the cascade of an IDFT, a windowing operation and a subsequent DFT which ascertains the constraint of zero-padded time domain filter partitions [1]. If the step sizes are chosen such that

$$\mu_{\nu,1}(m) \;\equiv\; \mu_{\nu,2}(m) \;=\; \frac{\alpha}{S_\nu(m)+\delta} \tag{17}$$

with $0<\alpha<2$ and a regularization constant δ to prevent numerical problems, (16) yields a *jointly normalized update (JNLMS)*. This is due to the fact that the $S_\nu(m)$ represent the smoothed subband powers of the input spectra of both Volterra kernels according to

$$S_\nu(m) \;=\; S_{\nu,1}(m)+S_{\nu,2}(m) \tag{18}$$

which is furthermore composed of the individual powers

$$S_{\nu,p}(m) \;=\; \lambda_p\,S_{\nu-1,p}(m)+(1-\lambda_p)\,\big|\big|\mathbf{X}_{\nu,p}(m)\big|\big|_2^2 \tag{19}$$

for $p=1,2$ and the forgetting factors λ_p chosen smaller than one.

Apparently, this JNLMS update is dominated by an adaptation of the linear Volterra kernel as $S_{\nu,2}(m) << S_{\nu,1}(m)$. This is a consequence of the DFT bin products and the division by M in (11) and yields a rather slow convergence of the quadratic kernel due to its weak excitation. On the other hand, this implies a noticeable impact on the complete adaptation progress as well, since the rarely compensated nonlinear components prevent the adaptive VF from converging to the unknown nonlinear system. A remedy to this is given by performing *separately normalized updates (SNLMS)*, where (16) is applied with a kernel-dependent step size

$$\mu_{\nu,p}(m) \;=\; \frac{\alpha_p}{S_{\nu,p}(m)+\delta} \tag{20}$$

for the linear ($p=1$) and the quadratic kernel ($p=2$), respectively. Nevertheless, since the misadjustment of the strongly excited linear kernel also acts as a distortion for the quadratic kernel, $\mu_{\nu,2}(m)$ has to be selected small enough to ensure a stable adaptation. However, this results in the drawback of a relatively slow convergence behaviour of the SNLMS as well and thus methods for accelerated adaptation are desirable.

4. ITERATED COEFFICIENT UPDATES

In the following, we will now apply an iterative update approach as performed in [2] to an NLAEC scenario by means of a second-order PBFDAVF. In order to exploit the excitation power within each frame more efficiently, the usual SNLMS is repeated another $R-1$ times per frame which results in a total of R coefficient update iterations in each kernel. The introduced extensions of this adaptation employing a *joint number of iterations (SNLMS-JI)* will be discussed in the following.

Due to repeated coefficient update in each frame ν, the general update equation (16) is reformulated according to

$$\widehat{\mathbf{H}}_{\nu,p}^{(r+1)}(m) \;=\; \widehat{\mathbf{H}}_{\nu,p}^{(r)}(m) + \mu_{\nu,p}(m)\,\mathcal{C}\Big\{E_\nu^{(r)}(m)\,\mathbf{X}_{\nu,p}^{*}(m)\Big\}, \tag{21}$$

which is valid for $0 \le r \le R-1$ up to a specified number R of allowed iterations. To ascertain a continuous adaptation towards the unknown nonlinear system, the initial and final coefficient sets are given by

$$\widehat{\mathbf{H}}_{\nu,p}^{(0)}(m) \;=\; \widehat{\mathbf{H}}_{\nu-1,p}^{(R)}(m) \tag{22}$$

$$\widehat{\mathbf{H}}_{\nu+1,p}^{(0)}(m) \;=\; \widehat{\mathbf{H}}_{\nu,p}^{(R)}(m) \tag{23}$$

for each frame ν respectively. As can be seen from (21), the same input data $\mathbf{X}_{\nu,p}(m)$ is used for each of the iterations, so that the filter output after the r-th iteration is given by:

$$\widehat{Y}_\nu^{(r)}(m) \;=\; \Big[\widehat{\mathbf{H}}_\nu^{(r)}(m)\Big]^{\mathrm{T}}\mathbf{X}_\nu(m), \tag{24}$$

$$\widehat{y}_\nu^{(r)}(\kappa) \;=\; \mathrm{IDFT}_M\Big\{\widehat{Y}_\nu^{(r)}(m)\Big\}. \tag{25}$$

Accordingly, the residual error after the r-th coefficient update thus reads

$$\widetilde{e}_\nu^{(r)}(\kappa) \;=\; \widetilde{d}_\nu(\kappa) - \widehat{y}_\nu^{(r)}(\kappa). \tag{26}$$

and is further reduced compared to the a-priori error $\widetilde{e}_\nu^{(0)}(\kappa)$ [2], because it is already based on the refined echo estimate $\widehat{y}_\nu^{(r)}(\kappa)$. Constructing the desired frame $e_\nu^{(r)}(\kappa)$ analogously to (14), the corresponding DFT spectrum of the intermediate block error of (26) is calculated as

$$E_\nu^{(r)}(m) \;=\; \mathrm{DFT}_M\Big\{e_\nu^{(r)}(\kappa)\Big\}, \tag{27}$$

and can then be used to obtain another coefficient update (21).

Repeating the above procedure for all R iterations in each frame ν thus yields the a-posteriori error $\widetilde{e}_\nu^{(R-1)}(\kappa)$. Consequently, the MSE of this final error block will be significantly lower than that of $\widetilde{e}_\nu^{(0)}(\kappa)$, due to the fact that the input data $\mathbf{X}_\nu(m)$ and $\widetilde{d}_\nu(\kappa)$ of the current frame are exploited several times by the SNLMS-JI. On the other hand, this implies that the second-order VF is not only adapted towards the unknown nonlinear system, but that each $\widetilde{e}_\nu^{(r)}(\kappa)$ is also minimized w.r.t. the short-time signal characteristics to some extent. However, this behaviour has also been reported in [2] for scenarios of linear AEC and may be considered quite beneficial for real-world applications within noisy environments as will be demonstrated by the experimental results of Section 5.

Moreover, it has been shown in [5] that for the case of partitioned block linear filtering, the employed iterated adaptation using subband normalization and constrained coefficient updates has a potential risk of stability problems for certain data sets and $R \to \infty$. Nevertheless, considering the small number of iterations (e.g. $R \le 10$) for practical realizations of adaptive second-order VFs, these stability issues may be neglected.

As can be seen from (21) to (27), the iterated coefficient update has in principle about R times the complexity of a single-update SNLMS algorithm. However, comparing the iterated adaptation procedure to an SNLMS update with $\rho = R$, we find that the SNLMS-JI is slightly less complex since for a reference length of N samples, only a single DFT operation is sufficient to create the input spectra of the $\mathbf{X}_{\nu,p}(m)$ whereas R transforms are required in case of a processing with overlapping frames.

Regarding the large number of filter coefficients which contribute to $\widehat{\mathbf{H}}_{\nu,2}(m)$ in (9), the algorithmic complexity of the SNLMS-JI is mainly determined by the quadratic kernel of the adaptive VF. Thus it is desirable to perform a different number of update iterations R_p for each of the Volterra kernels ($p = 1, 2$). By doing so, the number of coefficient update iterations of the SNLMS algorithm may be adjusted to some optimum setting of R_1, R_2 in terms of complexity and convergence speed.

The corresponding filter update equation of this *separately iterated (SNLMS-SI)* algorithm then reads

$$\widehat{\mathbf{H}}_{\nu,p}^{(r_p+1)}(m) \;=\; \widehat{\mathbf{H}}_{\nu,p}^{(r_p)}(m) + \mu_{\nu,p}(m)\,\mathcal{C}\Big\{E_\nu^{(r_p)}(m)\,\mathbf{X}_{\nu,p}^{*}(m)\Big\}, \tag{28}$$

where $0 \le r_p \le R_p - 1$ controls the kernel-dependent number of iterations. Note that the transitions of the filter coefficient sets as in (22), (23) are performed accordingly.

Taking the computational demands of the quadratic kernel filtering into account, it is suitable to choose the number of iterations for the linear kernel greater than those of the quadratic kernel, i.e. in practice $R_1 \ge R_2$ holds. Assuming such a setting of the iteration parameters, we find that the filter output for the common iterations of both kernels with $0 \le r \le R_2 - 1$ is obtained as given by (24). However, the PBFDAVF for any further iteration $R_2 \le r \le R_1 - 1$ reads

$$\widehat{Y}_\nu^{(r)}(m) \;=\; \Big[\widehat{\mathbf{H}}_{\nu,1}^{(r)}(m)\Big]^{\mathrm{T}}\mathbf{X}_{\nu,1}(m) + \Big[\widehat{\mathbf{H}}_{\nu,2}^{(R_2-1)}(m)\Big]^{\mathrm{T}}\mathbf{X}_{\nu,2}(m), \tag{29}$$

since the adaptation of the quadratic kernel is stopped and only the linear kernel is adjusted furthermore. Note that the kernel output corresponding to the coefficient set $\widehat{\mathbf{H}}_{\nu,2}^{(R_2-1)}(m)$ can efficiently be stored in memory and thus no quadratic filtering has to be performed for the iterations exceeding R_2 in frame ν.

5. EXPERIMENTAL RESULTS

We will now present simulation results for an NLAEC system as depicted in Fig. 1 applying the proposed iterated coefficient update for stationary noise input (coloured, Laplacian) and male speech. Thereby, the echo path of $y(k)$ has been modelled by a second-order VF with $N_1 = 320$ and $N_2 = 64$, where a power ratio of linear to nonlinear components of 20 dB has been achieved. The echo canceller is designed as a second-order PBFDAVF of the same filter kernel lengths with a common partition size $N = 64$ (implying $B_1 = 5$, $B_2 = 1$) accordingly.

All experiments have been conducted using step sizes $\alpha_1 \equiv \alpha_2 = 0.3$, forgetting factors $\lambda_1 = 0.94, \lambda_2 = 0.72$ and a regularization constant $\delta = 0.001$. Since no step size control is regarded, $d(k)$ is generated for single-talk situations (i.e. $s(k) = 0$) only, using some amount of additive white noise such that an SNR of 30 dB is obtained. Therefore, all presented curves display the *total ERLE*

$$\text{ERLE}(k)\ [\text{dB}] \ := \ 10 \cdot \log_{10}\left(\frac{\mathcal{E}\{d^2(k)\}}{\mathcal{E}\{e^2(k)\}}\right) \qquad (30)$$

which is a common measure for the achieved echo cancellation. Note that this evaluation is based on the real-world signals $d(k)$ and $e(k)$ as they will be perceived by a far-end listener in noisy environments, i.e. when $n(k) \neq 0$.

Fig. 2 shows the results of an NLAEC with the aforementioned noise (top) and speech input (bottom). In both cases, we compare a plain SNLMS adaptation with $\rho = 1$ to an SNLMS using $\rho = 4$ and an SNLMS-JI algorithm which has no overlapping data, but employs $R = 4$ iterations per frame. As can be noticed, the iterated coefficient updates clearly outperform the SNLMS algorithm in terms of convergence speed and steady-state echo cancellation. Furthermore, the achieved gain is even above the SNR threshold of 30 dB, which is due to the inherent exploitation of the short-time characteristics of the noise $n(k)$ which is also contained in $d(k)$ [2].

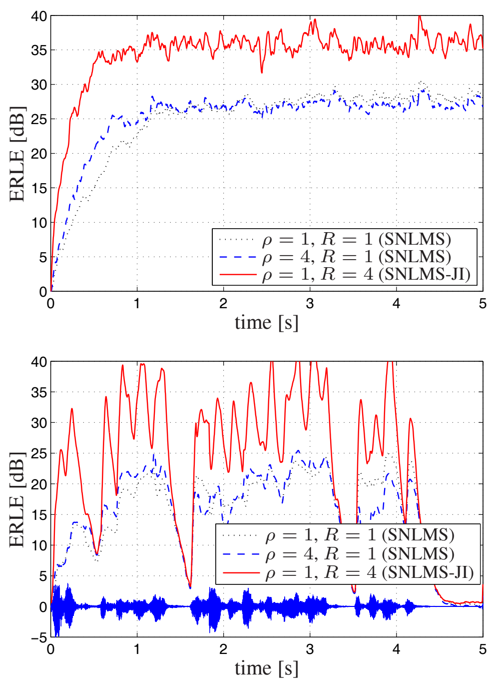

Fig. 2. ERLE results for noise (top) and speech input (bottom) of a second-order PBFDAVF using SNLMS and SNLMS-JI adaptation

In order to investigate the SNLMS-SI with a kernel-dependent number of iterations, Fig. 3 illustrates the corresponding ERLE results for speech input, where $\rho = 1$ has been set for all evaluated algorithms. Interestingly, it indicates that an iterated adaptation with $R_1 = 4$ and $R_2 = 2$ is comparable to an algorithmic version which performs update iterations only for the quadratic kernel, but with a higher complexity ($R_2 = 4$). On the other hand, comparing these results with Fig. 2, we find that the achieved ERLE gain approaches the results of the SNLMS-JI algorithm with a joint number of $R = 4$ iterations for both Volterra kernels and is noticeably higher than for the single-update adaptation.

By this, we conclude that in case of an SNLMS-SI algorithm, controls for a consistent adaptation of both Volterra kernels are beneficial in order to avoid large cross-kernel distortions which might inhibit proper convergence. However, this result indicates that the computational demands of the iterated SNLMS may be reduced significantly by using a separate number of iterations for each kernel and lowering R_2 for the adaptation of the quadratic Volterra kernel. Moreover, if the complexity of these iterated updates is to be reduced further, fast realizations of this technique have to be taken into account, as also available for the partitioned block adaptive filtering within linear AEC scenarios [2].

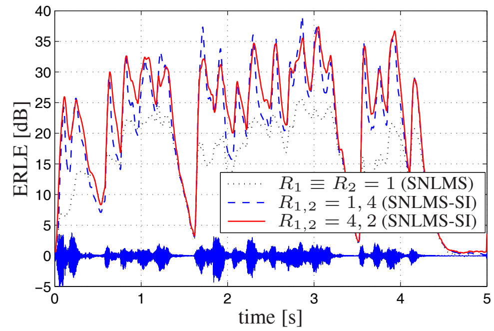

Fig. 3. ERLE results for speech input comparing SNLMS and SNLMS-SI adaptation of the PBFDAVF

6. CONCLUSIONS

We have proposed an iterated version of the partitioned block adaptive Volterra filter in the DFT domain. By repeating the coefficient update several times for a given data frame, the excitation of the filter kernels is considerably increased and thus higher adaptation speeds are obtained. Furthermore, we have shown that even higher steady-state echo attenuation can be observed for both noise and speech inputs since the filter adaptation is accompanied with an optimization towards the short-time signal characteristics in each frame. Although the complexity of this algorithm is roughly proportional to the number of applied iterations, the computational demands are still below a comparable approach with increased frame overlap. Moreover, the number of calculations can be further reduced by selecting a different number of update iterations for the linear and the quadratic Volterra kernel, respectively.

7. REFERENCES

[1] F. Küch and W. Kellermann, "Partitioned block frequency-domain adaptive second-order Volterra filter," *IEEE Trans. on Signal Processing*, vol. 53, no. 2, pp. 564–575, Feb. 2005.

[2] K. Eneman and M. Moonen, "Iterated partitioned block frequency-domain adaptive filtering for acoustic echo cancellation," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 11, no. 2, pp. 143–158, March 2003.

[3] V. J. Mathews and G. L. Sicuranza, *Polynomial Signal Processing*, New York: John Wiley and Sons, 2000.

[4] S. Haykin, *Adaptive Filter Theory*, New Jersey: Prentice Hall, 2002 (Fourth Edition).

[5] K. Eneman and M. Moonen, "On iterating the partitioned block frequency-domain adaptive filter," in *Technical Report 00-127*, Katholieke Universiteit Leuven, Dec. 2000.