Computationally Efficient Active Noise Reduction in Headsets

J. Graf[1,*] and E. Reithmeier[1,†]

[1] *Institute for Measurement and Automatic Control*
Nienburger Strasse 17, 30167 Hanover, Germany

In order to improve the passive attenuation of hearing protection headsets, active noise reduction (ANR) techniques are usually applied. These ANR-techniques accomplish the active attenuation of the disturbing acoustical noise using an out-of-phase antinoise. The antinoise destructively interferes with the disturbing noise close to the humans ear drum. The generation of the antinoise can be conducted using different control strategies. However, due to variable system plants, often adaptive control strategies are chosen. Even though such adaptive systems effectively attenuate the disturbing noise in a wide frequency range, a major disadvantage is the computational effort linked to the large amount of controller parameters. The controller parameters have to be updated by the adaptive algorithm and the resulting computational effort makes the application of expensive digital signal processors unavoidable. For this reason, no commercial products realizing adaptive broadband techniques are on the market yet.
In this paper, a partially-adaptive control approach is introduced which permits the reduction of the computational effort in comparison to conventional and fully adaptive ANR-controllers. The noise reduction performance as well as the computational efficiency of the proposed control strategy is presented.

Keywords: ANR-headset, computationally efficient active noise reduction, partially-adaptive IIR-filter

INTRODUCTION

Active noise reduction (ANR) headsets are commonly used in surroundings of high level noise. For example, a typical field of application is the hearing protection of pilots or employees who work e.g. on oil platforms or in engine rooms. The active attenuation of the disturbing noise is commonly accomplished by an ANR-controller. Even though effective but also complex ANR-control strategies have been developed [1] [2] [3], contemporary commercial ANR-Headsets primary use the well known non-adaptive feedback control approach. The advantage of this approach is its simple and economic implementation. However, a disadvantage is the fact that the controller is not able to adapt to different noise spectra. In contrast, adaptive ANR-systems accomplish the adaptation to different noise spectra but suffer from the algorithm's high computational complexity. In order to provide sufficient processing power, fast digital signal processors are required which cause reasonable production costs as well as high power consumption. Therefore, the implementation of non-adaptive feedback controllers is still the most reasonable compromise between noise attenuation performance, controller complexity and power consumption.
However, due to time delay, the feedback control approach especially enables the attenuation of low frequency noise. If the attenuation of higher frequency components is required, the feedforward control technique is more appropriate. In case of circum-aural headsets, the feedforward approach is particularly sensitive to plant variations resulting from changing noise source directions or ear-cup leakage. In order to account for these plant variations, the implementation of an adaptive feedforward controller is suggested in the literature [4] and [5].
In the framework of a project called COMDAC (COMmunication headset with Digital Adaptive noise Cancellation) an ANR-headset was developed that permits the attenuation of broadband noise up to about 3000Hz. The COMDAC-algorithm uses a combination of an adaptive feedforward controller implementing 140 adaptive parameters linked to a non-adaptive standard feedback control loop. In order to reduce the computational effort of the COMDAC fully adaptive feedforward controller, in this paper a partially-adaptive feedforward controller is introduced. Similar to the COMDAC-controller, an additional non-adaptive feedback loop is used to improve the noise reduction especially in the low frequency range.
In the next section, the computational complexity of the conventional and fully adaptive feedforward controller as applied in the framework of the COMDAC-project is examined. Section 2 introduces the partially-adaptive feedforward controller and describes the combined partially-adaptive feedforward and non-adaptive feedback control strategy. The active attenuation performance of the designed controller in comparison to the COMDAC-controller is presented in section 3 and concluding remarks are given in section 4.

* Electronic address: graf@imr.uni-hannover.de
† Electronic address: reithmeier@imr.uni-hannover.de

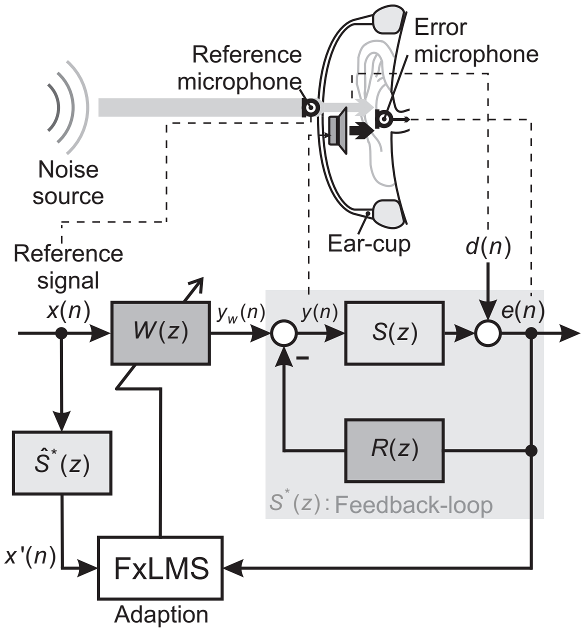

Figure 1. Combined ANR-controller consisting of a non-adaptive feedback loop linked to an adaptive feedforward controller.

1. COMPLEXITY OF THE CONVENTIONAL IMPLEMENTATION

In this section, the computational complexity linked to the conventional adaptive feedforward controller as depicted in figure 1 is examined. In the upper part of the figure, the schematic drawing of the ear-up is depicted including the reference microphone and the error microphone. The compensation loudspeaker is located in front of the ear-canal inside the ear-cup and it outputs the out-of-phase antinoise $y(n)$ that is generated by the ANR-controller. The block diagram of the combined controller is depicted in the lower part of figure 1. The reference signal $x(n)$ is recorded by the reference microphone and inputted to the adaptive feedforward filter $W(z)$. The error signal is processed by the non-adaptive feedback controller $R(z)$ that is implemented as a transfer function with 10 zeros and 10 poles:

$$R(z) = \frac{b_0 + b_1 z^{-1} + \ldots + b_{10} z^{-10}}{1 - a_1 z^{-1} - \ldots - a_{10} z^{-10}}. \tag{1.1}$$

The outputs of both controllers are combined and the actuating variable $y(n)$ is outputted to the compensation loudspeaker.

Since the adaptive feedforward controller is realized as an FIR-filter, a convex performance surface results and thus the adaptation of the controller parameters $\mathbf{w(n)}$ can be accomplished using the well known Filtered-x-Least-Mean-Squares algorithm (FxLMS) [4], which is illustrated in table 1.

In the following we focus on the computational complexity of the control algorithm. A computationally expensive calculation is the calculation of the filter-output in step 2:

$$y_w(n) = \mathbf{w}^T(n) \cdot \mathbf{x}(n). \tag{1.2}$$

In this equation, the vector $\mathbf{x}(n)$ contains the time series of the reference signal and $\mathbf{w}(n)$ denotes the adaptive parameter vector of length L:

$$\mathbf{x}(n) = [x(n)\ x(n-1)\ \ldots\ x(n-L+1)]^T,$$
$$\mathbf{w}(n) = [w_0(n)\ w_1(n)\ \ldots\ w_{L-1}(n)]^T. \tag{1.3}$$

Another expensive calculation occurs in conjunction with the parameter update of step 6:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \mathbf{x}'(n) e(n). \tag{1.4}$$

The parameter update necessitates the time series of the filtered reference signal $\mathbf{x}'(n)$ which is obtained by filtering the reference with a model of the closed loop secondary path $\hat{S}^*(z)$ (step 5). Here, the model $\hat{S}^*(z)$ is realized as a transfer function with 6 zeros and 6 poles. Thus, step 5 of table 1 can be written in the time discrete domain:

$$x'(n) = b_0 x(n) + b_1 x(n-1) + \ldots + b_6 x(n-6)$$
$$+ a_1 x'(n-1) + \ldots + a_6 x'(n-6). \tag{1.5}$$

In this equation the parameters $b_0 \ldots b_6$ denote the numerator and the parameters $a_1 \ldots a_6$ represent the denominator of the model $\hat{S}^*(z)$. Thus, the calculation according to equation 1.5 necessitates 13 multiplications and 12 additions.

However, the computational complexity of the FxLMS-algorithm is primarily determined by the equations 1.2 and 1.4. Hence, the complexity of the algorithm is directly linked to the amount of adaptive parameters L.

In order to calculate equation 1.2, in every single sample interval L multiplications as well as $L-1$ additions are necessary. According to equation 1.4,

Initialization: $\mathbf{w}(0) = 0$
Calculate for each sample $n = 0, 1, 2, \ldots$: 1) Input of the reference $x(n)$ 2) Calculation of the filter output: $y_w(n) = \mathbf{w}^T(n)\mathbf{x}(n)$ 3) Output of the actuating variable $y_w(n)$ 4) Input of the error microphone $e(n)$ 5) Filtering of the reference signal: $x'(n) = \mathcal{Z}^{-1}\{\hat{S}^*(z)X(z)\}$ 6) Update of the parameter vector: $\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\mathbf{x}'(n)e(n)$

Table 1. FxLMS-algorithm.

	Additions	Multiplications
Step 2	139	140
Step 5	12	13
Step 6	140	141
FB-controller	20	21
Sum	311	315

Table 2. Required mathematical operations for the conventional implementation of the combined controller. Amount of adaptive parameters: $L = 140$.

the update of L parameters requires $L + 1$ multiplications as well as L additions. The necessary effort for the calculation of these two equations decisively determines the computational complexity of the adaptive feedforward controller. In addition to the computation of the adaptive feedforward controller, the output of the feedback controller has to be computed. The non-adaptive feedback controller is implemented as an IIR-filter with 10 zeros and 10 poles and thus, additional 21 multiplications and 20 additions are required. All necessary computations linked to the described control approach are summarized in table 2.

2. PARTIALLY-ADAPTIVE FEEDFORWARD AND NON-ADAPTIVE FEEDBACK CONTROL

According to figure 2, perfect compensation of the disturbing noise is achieved if:

$$e(n) = 0 \quad \rightarrow \quad u(n) = -d(n). \tag{2.1}$$

Thus, the optimal feedforward filter is given in the z-domain as [2]:

$$W(z) = -\frac{P_2(z)}{P_1(z) \cdot S^*(z)}. \tag{2.2}$$

In this equation, $P_2(z)$ models the transfer function of the propagation behavior from the noise source to the error microphone and $P_1(z)$ represents the acoustical transfer function from the noise source to the reference microphone respectively. Due to changing plants $P_1(z)$ and $P_2(z)$, the feedforward filter $W(z)$ is permanently adapted by the

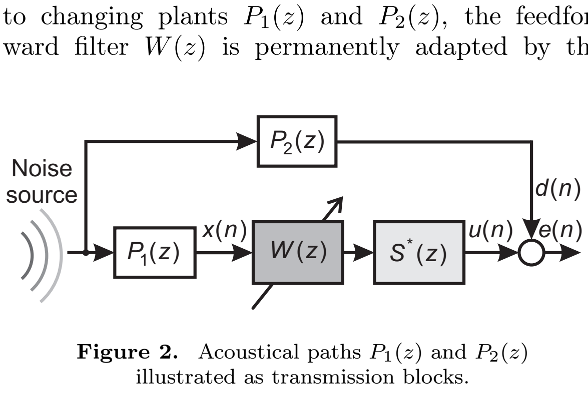

Figure 2. Acoustical paths $P_1(z)$ and $P_2(z)$ illustrated as transmission blocks.

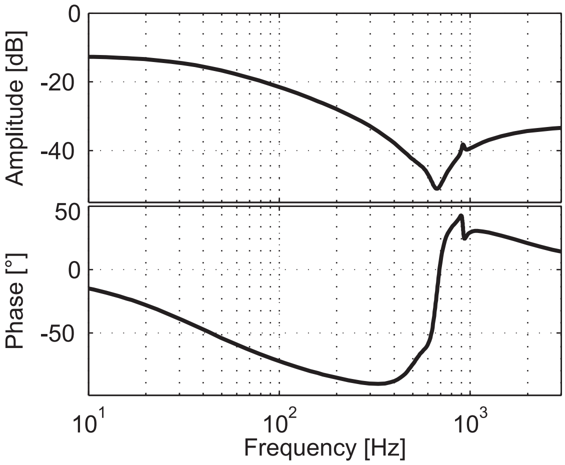

Figure 3. Bode plot of the non-adaptive part of the partially-adaptive feedforward filter.

FxLMS-algorithm (section 1). However, for the practical implementation of the adaptive feedforward controller, an algorithm with reduced computational complexity compared to the conventional feedforward-system of figure 1 is favorable. In order to assess the saving potential regarding the computational complexity, the optimal feedforward transfer function 2.2 was designed for different noise source directions and ear-cup leakages. In conjunction with the implementation of different $W(z)$ as non-adaptive IIR-Filters, it turned out that the poles and zeros of the filters are primary located within the right-hand side of the outer part of the unit circle. The location of the poles is especially determined by the characteristic decay of the ear-cup's passive attenuation at approx. 100 Hz, which is similar for all examined optimal feedforward transfer functions $W(z)$. This fact suggests itself the design of a non-adaptive transfer function that realizes the mentioned non-variable decay. A capable transfer function is found by averaging all measured feedforward filters. The obtained averaged transfer function represents the non-adaptive part $W_{na}(z)$ of the partially-adaptive feedforward controller. The bode plot of the IIR-filter $W_{na}(z)$ is depicted in figure 3.

2.1. Serial or Parallel Interconnection of the non-Adaptive and the Adaptive Feedforward Part

So far, it has not been discussed in which manner the non-adaptive part $W_{na}(z)$ of the feedforward controller has to be linked to the adaptive part $W(z)$. As depicted in figure 4, a serial or a parallel interconnection is possible. Under consideration of a serial interconnection, the entire transfer function of the partially adaptive feedforward controller can

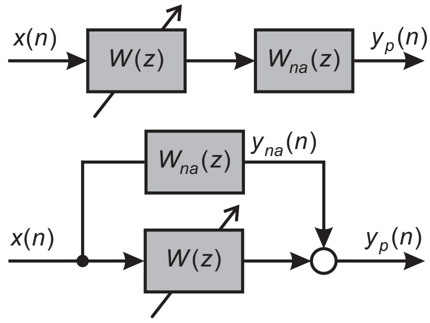

Figure 4. Possible interconnections of the non-adaptive and the adaptive feedforward filter.

be written as:

$$\frac{Y_p(z)}{X(z)} = \underbrace{\frac{b_0 + b_1 z^{-1} + \ldots + b_M z^{-M}}{a_0 - a_1 z^{-1} - \ldots - a_N z^{-N}}}_{\text{non-adaptive part } W_{na}(z)} \cdot \underbrace{(w_0 + \ldots + w_{L-1} z^{-L+1})}_{\text{adaptive part } W(z)}. \qquad (2.3)$$

This equation shows that the zeros of the partially adaptive feedforward filter are affected by the adaptive part $W(z)$.
Now, the following definitions are considered:

$$\begin{aligned}
\mathbf{b} &= [b_0\ b_1\ \ldots\ b_M\ \underbrace{0\ \ldots\ 0}_{L-M-1\text{ zeros}}\ ]^T, \\
\mathbf{a} &= [a_1\ a_2\ \ldots\ a_N]^T,\ a_0 = 1, \\
\mathbf{w}(n) &= \begin{bmatrix} b_0 w_0 \\ b_0 w_1 + b_1 w_0 \\ b_0 w_2 + b_1 w_1 + b_2 w_0 \\ \vdots \\ b_M w_{L-1} \end{bmatrix}, \\
\mathbf{x}(n) &= [x(n)\ x(n-1)\ \ldots\ x(n-L+1)]^T, \\
\mathbf{y}_p(n-1) &= [y_p(n-1)\ y_p(n-2)\ \ldots\ y_p(n-N)]^T.
\end{aligned}$$

Using these definitions, the output of the partially-adaptive filter can be written in the time discrete domain as follows:

$$y_p(n) = \mathbf{w}^T(n)\mathbf{x}(n) + \mathbf{a}^T \mathbf{y}_p(n-1). \qquad (2.5)$$

In case of the serial filter-structure, the non-adaptive filter $W_{na}(z)$ can be seen as a part of the secondary path. Experiments showed that according to this change of the secondary path, the adaptation of the parameters $W(z)$ results in a controller that insufficiently reduces the disturbing noise.
In contrast to the serial interconnection, the parallel linking according the lower part of figure 4 is more suitable. The filter output of the partially-adaptive feedforward filter in case of the parallel interconnection can be written in the z-domain as:

$$\frac{Y_p(z)}{X(z)} = \underbrace{\frac{b_0 + b_1 z^{-1} + \ldots + b_M z^{-M}}{a_0 - a_1 z^{-1} \ldots - a_N z^{-N}}}_{\text{non-adaptive part } W_{na}(z)} + \underbrace{(w_0 + w_1 z^{-1} \ldots + w_{L-1} z^{-L+1})}_{\text{adaptive part } W(z)}. \qquad (2.6)$$

Similar to equation 2.5, the vector

$$\mathbf{w}(n) = \begin{bmatrix} a_0 w_0 \\ a_0 w_1 + a_1 w_0 \\ a_0 w_2 + a_1 w_1 + a_2 w_0 \\ \vdots \\ a_N w_{L-2} + a_{N-1} w_{L-1} \\ a_N w_{L-1} \end{bmatrix} \qquad (2.7)$$

is redefined and the partially-adaptive IIR-filter's output follows:

$$y_p(n) = [\mathbf{b}^T + \mathbf{w}(n)^T]\mathbf{x}(n) + \mathbf{a}^T \mathbf{y}_{na}(n-1). \qquad (2.8)$$

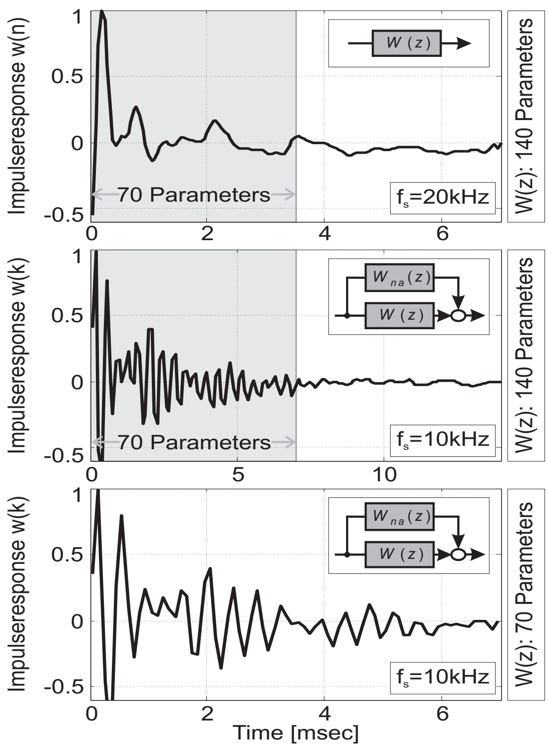

Figure 5. Impulse responses. Upper part: Conventional implementation, 140 parameter. Middle part: Partially-adaptive implementation, 140 parameters. Lower part: Partially-adaptive implementation, 70 parameters.

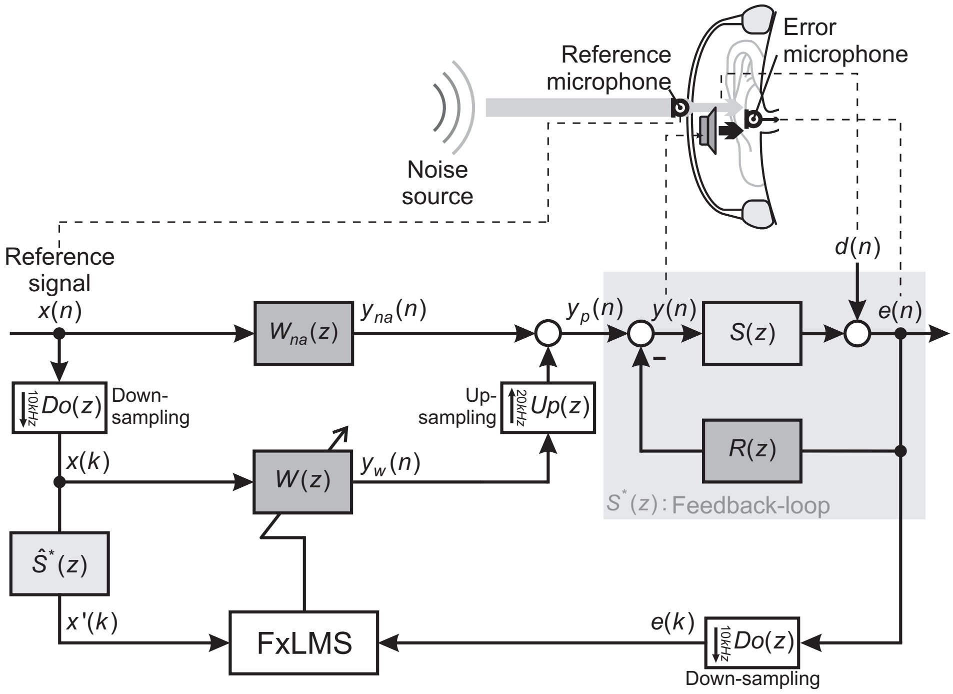

Figure 6. Block diagram of the combined non-adaptive feedback and partially-adaptive feedforward controller.

2.2. *Impulse Response and Sampling Frequency*

Since $W_{na}(z)$ summarizes the invariant system behavior in a non-adaptive transfer function, it can be interpreted as a partial solution of the optimal feedforward filter 2.2 and thus the adaptation problem of the adaptive filter is simplified. The simplified problem enables the reduction of the adaptive filter parameters in comparison to the conventional adaptive feedforward approach according to figure 1. It turned out that solely a reduction of the parameters is not sufficient. This can be explained by the examination of the impulse response of $W(z)$ as depicted in figure 5. The upper part of the figure shows the impulse response of the conventional and fully adaptive feedforward filter using a sampling frequency $f_s = 20\,\text{kHz}$ and an amount of 140 adaptive parameters. The grey highlighted area denotes the impulse response in case of using 70 parameters. It can be seen that the impulse response is not sufficiently decayed using merely 70 parameters. Thus, the noise reduction performance would deteriorate by solely a reduction of the adaptive parameters. In order to lengthen the impulse response of the 70 parameter filter to the original length, the sampling frequency has to be reduced to $f_s = 10\,\text{kHz}$. This in fact results in less time-resolution of the impulse response. However, in case of the partially-adaptive feedforward controller, the reduced time resolution is neutralized by the non-adaptive filter $W_{na}(z)$.
The impulse response of $W(z)$ in case of the partially-adaptive controller using 140 parameter and a sampling frequency of $f_s = 10\,\text{kHz}$ is shown in the middle part of figure 5. Even though the impulse response is subject to enormous oscillations, it is sufficiently decayed using only 70 parameters. This means that further 70 parameters have no significant influence on the transfer behavior of the converged filter $W(z)$. Hence, the sampling frequency of $f_s = 10\,\text{kHz}$ in combination with 70 adaptive parameters is sufficient for the implementation of the adaptive filter $W(z)$. The impulse response of the 70-parameter-filter $W(z)$ is shown in the lower part of the figure.
Remark: For discrete time signals which are sampled with a frequency of $f_s = 10\,\text{kHz}$, the discrete time variable k is used and for signals that are sampled with $f_s = 20\,\text{kHz}$, the discrete time variable n is used respectively.

2.3. *Combining the non-Adaptive and the Adaptive Controller*

Considering the described partially-adaptive feedforward system and the linked feedback control loop, the complete ANR-controller operates with two sampling frequencies. The non-adaptive feedforward part as well as the feedback controller operates at a sampling frequency of 20 kHz while the adaptive feedforward part requires a sampling frequency of 10 kHz. Such a control systems is commonly referred to as a multirate control system. The block diagram

of the combined ANR-controller is shown in figure 6.
In order to link the controllers, a sampling frequency conversion is required. In the block diagram, the sampling rate conversions are depicted as transmission blocks named "up-sampling" and "down-sampling" respectively.
The actuation variable $y(n)$ that is outputted to the compensation loudspeaker is provided by the following control law:

$$y(n) = \mathcal{Z}^{-1}\{[Do(z)W(z)Up(z) - W_{na}(z)]X(z) - R_{FB}(z)E(z)\}. \tag{2.9}$$

3. RESULTS

3.1. Active Noise Reduction

In order to verify the noise reduction performance of the proposed controller, the ANR-headset is exposed to broadband disturbing noise and the active attenuation is measured on a test-head. The test-head is equipped with an integrated ear-simulator. The involved ear-microphone is used to measure the noise reduction performance of the control algorithm.
Figure 7 shows the noise reduction performance in case of broadband noise. It can be seen that the noise reduction of the combined controller realizing the partially-adaptive feedforward system is comparable to the conventional implementation (COMDAC-algorithm).
However, the implementation of the partially-adaptive feedforward controller results in an adaptive feedforward controller with a reduced amount of adaptive parameters. The computational savings that are achieved with this control strategy are discussed in the following paragraph.

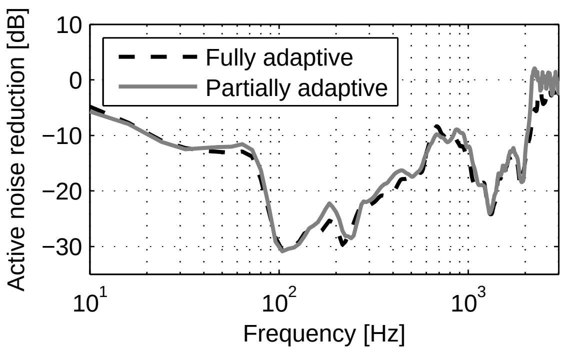

Figure 7. ANR-performance of the conventional implementation according to figure 1 in comparison to the implementation with the combined partially-adaptive ANR-controller.

3.2. Computational Savings

The computational effort that is linked to the feedback controller as well as the filtering of the reference signal (step 5) is not affected by the introduction of the partially-adaptive feedforward controller. Furthermore, the output of the non-adaptive filter $W_{na}(z)$ additionally has to be computed. However, the computational complexity linked to equation 1.2 (step 2) as well as the filter update of equation 1.4 (step 6) is reduced in comparison to the conventional implementation of figure 1. Hence, a reasonable reduction of the computation time is achieved. The required computations of the entire partially-adaptive ANR-controller is summarized in table 3. It can be seen that in comparison to the conventional and fully adaptive feedforward approach 127 multiplications as well as 128 additions can be saved in every single sampling interval. Hereby, the saved calculations refer to mathematical operations for one single ear-cup. Since in real headset applications the algorithm has to be computed for two ear-cups, the double amount of computations can be saved.

	Additions	Multiplications
Step 2	69	70
Step 5	12	13
Step 6	70	71
FB-controller	20	21
non-Adap. FF-part	12	13
Sum	183	188

Table 3. Required mathematical operations for the combined non-adaptive feedback and partially-adaptive feedforward ANR-controller. Amount of adaptive parameters: $L = 70$.

4. CONCLUSION

Adaptive ANR-systems commonly realize complex algorithms that often require more than 100 adaptive filter taps in order to effectively reduce broadband disturbing noise. This amount of adaptive filter parameters results in considerable computational effort. Thus, fast and expensive digital signal processors are necessary. The high power consumption of these digital signal processors results in limitations regarding the battery life and thus in short operation time of the ANR-controller. In order to extend the operation time, in this paper a partially-adaptive feedforward controller is suggested that permits the saving of 50% of the adaptive parameters in comparison to a fully adaptive feedforward controller. The partially-adaptive feedforward controller is linked to a non-adaptive feedback controller and the noise reduction performance is compared to the conventional adaptive feedforward control approach. The measurements show that the broadband noise reduction of the proposed controller is comparable to the conventional and fully adaptive ANR-controller.

5. LITERATURE

1. Wolter, B., Peissig, J., Foudhaili, H., Reithmeier, E., "Combined feedback and adaptive feedforward active noise control in headsets", Proc. 36th International Congress and Exhibition on Noise Control Engineering, Internoise, Istanbul, 2007.
2. Foudhaili, H., "Kombinierte Feedback- und adaptive Feedforward-Regelung fuer die aktive Laermreduktion in einem Kommunikations-Headset", Shaker Verlag, Aachen, 2008.
3. Reithmeier, E., Graf, J., "Combination of non-Adaptive and Adaptive Control Strategies for Active Noise Reduction in Headsets", Proc. 9th International Conference on Motion and Vibration Control, Movic, Munich, 2008.
4. Kuo, S.M., Morgan, D.R., "Active Noise Control Systems - Algorithms and DSP Implementations", Wiley-Interscience Publication, New York, 1996.
5. Elliott, S.J., "Signal processing for active control", Academic Press, San Diego, 2001.