Audio Engineering Society
Convention Paper 10365
Presented online at the 148th Convention
2020 June 2 – 5,

Comparison of LMS-based adaptive audio filters for identification

Kristóf Horváth and Balázs Bank

Budapest University of Technology and Economics, Department of Measurement and Information Systems, H-1521 Budapest, Hungary

Correspondence should be addressed to Kristóf Horváth (hkristof@mit.bme.hu)

ABSTRACT

In the field of audio signal processing, logarithmic frequency resolution IIR filters, such as fixed-pole parallel filters and Kautz filters, are often used. These proven structures can efficiently approximate the frequency resolution of hearing, which is a highly desired property in audio applications. In recursive adaptive filtering however, the FIR structure with LMS algorithm is the most common. Since the linear frequency resolution of FIR filters is not well-suited for audio applications, in this paper we explore the possibility of combining the logarithmic frequency resolution IIR filters with the LMS algorithm. To this end the LMS algorithm is applied to fixed-pole parallel and Kautz filters, and the resulting structures are compared in terms of convergence properties.

1 Introduction

In adaptive filtering, the least mean squares (LMS) algorithm is a popular choice because of its simplicity and global convergence. It is most commonly used with finite impulse response (FIR) filters because of their ease of implementation. However, it is possible to use the LMS algorithm with infinite impulse response (IIR) filters too, if the poles of the filter are fixed at predetermined values [1]. Additionally, IIR filters typically require fewer parameters to model a given response, as opposed to FIR filters.

In audio filtering, logarithmic frequency resolution is highly desired when modeling a transfer function. To achieve this, specialized filter design methodologies have been developed, including warped filters [2], second-order fixed-pole parallel filters [3], and Kautz filters [4]. The fixed-pole parallel and Kautz filters can also have a FIR section, which makes them well-suited for modeling non-minimumphase impulse responses [5].

The usual application for adaptive audio filters is to compensate a given response (equalization) or to reduce the additive noise present (noise canceling) [6–8]. These applications all contain an adaptive filter that identifies a given signal path. Thus, as a first step in comparing logarithmic frequency resolution IIR filters in adaptive context this paper explores the identification properties of different IIR filter structures using the LMS algorithm. For simplicity, we assume single-input and single-output (SISO) systems in our investigations.

Accordingly, the LMS algorithm is applied to fixed-pole parallel and Kautz filters including their delayed variants, and the resulting adaptive IIR filters are compared to each other and to the common FIR-LMS filters. In the examples both minimumphase and non-

minimumphase systems are identified, to show the difference between the IIR filters with and without additional FIR sections.

2 The LMS Algorithm

The LMS algorithm is a well-known adaptive filter design method, that has made its way into several textbooks [6,9,10]. In this section we give a short summary both about its basic and normalized variant.

The block scheme of the LMS filter can be found in Fig. 1. The system to be identified is marked by $H(z)$, the common input is denoted by $u(k)$. The outputs of the modeled system and the adaptive filter are marked by $y(k)$ and $\hat{y}(k)$ respectively. For FIR filters, the block denoted by ∇ is a delay line, and for IIR filters, it implements the poles of the filter. Its tap outputs, denoted by $\mathbf{x}(k)$, are used as the base functions in the resulting adaptive linear combiner structure.

The LMS algorithm is a stochastic grade descent method where the coefficients are adapted based on the current error in time [9]. It uses the estimate of the mean square error (MSE) gradient vector from the available data, to make successive corrections to the filter coefficients in the direction opposite to the gradient vector. This iterative procedure eventually converges to minimum mean square error.

To understand the LMS algorithm, the Wiener filter needs to be addressed first. The output of the Wiener filter is a linear combination of the base functions in the following form:

$$\hat{\mathrm{y}}(k) = \sum_{n=0}^{P-1} w_n \cdot \mathrm{x_n}(k) = \mathbf{w}^\top \cdot \mathbf{x}(k), \tag{1}$$

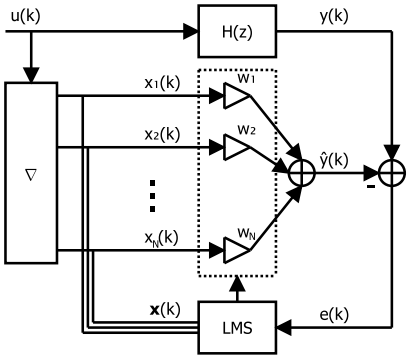

Fig. 1: LMS-based adaptive filter used for identification.

where $\mathbf{w}$ denotes the filter coefficients (weights) in vector form, and $\mathbf{x}(k)$ consists of the base functions at sample k.

The objective function is chosen as the expected value of the squared error:

$$\begin{aligned} V &= \mathbb{E}\{e^2(k)\} = \mathbb{E}\{(y(k) - \mathbf{w}^\top \cdot \mathbf{x}(k))^2\} \\ &= \mathbb{E}\{y^2(k)\} - 2\mathbf{w}^\top \mathbb{E}\{y(k)\mathbf{x}(k)\} \\ &+ \mathbf{w}^\top \mathbb{E}\{\mathbf{x}(k)\mathbf{x}^\top(k)\}\mathbf{w}. \end{aligned} \tag{2}$$

The minimum of the objective function can be found where the derivative, which equals to the gradient vector ($\mathbf{g}$), becomes zero:

$$\begin{aligned} \frac{\partial V}{\partial \mathbf{w}} = \mathbf{g} &= -2\mathbb{E}\{y(k)\mathbf{x}(k)\} + 2\mathbb{E}\{\mathbf{x}(k)\mathbf{x}^\top(k)\}\mathbf{w} \\ &= -2\mathbf{p} + 2\mathbf{R}\mathbf{w} = 0. \end{aligned} \tag{3}$$

After rearranging, the optimal coefficients of the Wiener filter can be calculated as:

$$\mathbf{w}_\mathrm{o} = \mathbf{R}^{-1} \cdot \mathbf{p}, \tag{4}$$

where $\mathbf{R} = \mathbb{E}\{\mathbf{x}(k) \cdot \mathbf{x}^\top(k)\}$, and $\mathbf{p} = \mathbb{E}\{\mathrm{y}(k) \cdot \mathbf{x}(k)\}$.

The optimum can also be calculated recursively, which leads to the steepest-descent method. This is especially useful when $H(z)$ is time-varying. The gradient vector always points to the opposite direction of the minimum, therefore by making successive corrections to the coefficient vector, the optimal weights can be found:

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) - \mu \mathbf{g}_\mathbf{w}(k) \\ &= \mathbf{w}(k) + 2\mu(\mathbf{p} - \mathbf{R}\mathbf{w}(k)), \end{aligned} \tag{5}$$

where $\mathbf{g}_\mathbf{w}(k)$ denotes the gradient vector at $\mathbf{w}(k)$, and μ is the step size parameter.

Note that in Eq. (5), $\mathbf{R}$ and $\mathbf{p}$ are expected values of a matrix and a vector, therefore their computation is not possible without knowing the stochastic properties of the input. However, they can be estimated using the following unbiased estimators at sample k:

$$\hat{\mathbf{R}}(k) = \frac{1}{L}\sum_{n=0}^{L-1} \mathbf{x}(k-n) \cdot \mathbf{x}^\top(k-n), \tag{6}$$

$$\hat{\mathbf{p}}(k) = \frac{1}{L}\sum_{n=0}^{L-1} \mathrm{y}(k-n) \cdot \mathbf{x}(k-n), \tag{7}$$

where L denotes the number of samples used for calculating the estimator. By choosing $L = 1$, the LMS

algorithm uses instantaneous unbiased estimates for $\mathbf{R}$ and $\mathbf{p}$:

$$\hat{\mathbf{R}}(k) = \mathbf{x}(k)\cdot\mathbf{x}^{\top}(k), \quad (8)$$

$$\hat{\mathbf{p}}(k) = \mathrm{y}(k)\cdot\mathbf{x}(k). \quad (9)$$

By substituting these estimates to Eq. (5), the coefficient adaptation formula becomes

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu e(k)\mathbf{x}(k). \quad (10)$$

It is important to note that the initial value of filter coefficients ($w(0)$) is arbitrary.

The tap output vector $\mathbf{x}(k)$ is unique for every filter structure. For FIR filters, it consists of the elements of a delay line; for other structures it can be deduced from the fact that the filter output is formulated as a linear combination of the tap outputs: $\hat{y}(k) = \mathbf{w}^{\top}\cdot\mathbf{x}(k)$.

It was proven that the eigenvalues of matrix $\mathbf{R}$ have an impact on the stability and convergence of the LMS algorithm [10]. To ensure stability, bounds can be prescribed for the step-size parameter:

$$0 < \mu < \frac{1}{\lambda_{max}}, \quad (11)$$

where λ_{max} denotes the largest eigenvalue of $\mathbf{R}$. By satisfying the above condition, the adaptive filter becomes stable regardless of how the base functions are generated.

The convergence properties (e.g. convergence time, residual error) are also connected with the eigenvalues of $\mathbf{R}$. Maximum convergence speed can be achieved by choosing the step size as:

$$\mu = \frac{1}{\lambda_{max} + \lambda_{min}}, \quad (12)$$

where λ_{max} and λ_{min} are the largest and the smallest eigenvalues of $\mathbf{R}$, respectively. Generally, larger step size leads to faster convergence to optimal filter weights. Therefore if λ_{max} is close to λ_{min}, the maximum convergence speed can be achieved. In filters where the tap outputs are independent and have the same output power, $\mathbf{R}$ is a diagonal matrix having the lowest possible eigenvalue-spread, for a given input.

The main drawback of the plain LMS algorithm is that the effective parameter step size scales with the input, which can cause instability in the adaption. As a remedy, the Normalized-LMS (NLMS) method is used, which normalizes the step size using the power of the input [9]:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\frac{e(k)\mathbf{x}(k)}{\alpha + \mathbf{x}^{\top}(k)\mathbf{x}(k)}, \quad (13)$$

where α is a small positive number used to avoid the denominator to become zero. By normalizing the step size, the stability condition for NLMS becomes

$$0 < \mu < 2. \quad (14)$$

Table 1: Number of arithmetic operations (multiplications, additions and divisions) required for the LMS algorithm when the number of tap outputs of the filter is P.

	Multiplication	Addition	Divisions
LMS	$P+1$	P	0
Normalized-LMS	$2P+1$	$2P+1$	1

The choice of the adaptive algorithm has an effect on the computational demand. Tab. 1 contains the number of arithmetic operations required for implementing the two variants of the LMS algorithm. Note that the normalized variant require more than two times the computational demand compared to the plain LMS algorithm. However, the stability criterion is simpler for the former, therefore in this paper we use the NLMS algorithm for implementing adaptive filters.

3 Adaptive IIR Filters

Adaptive IIR filters typically require fewer parameters compared to adaptive FIR filters, however, early research showed that adaptively varying both the poles and zeros can lead to suboptimal performance caused by multimodal error surfaces [11] or because they need to satisfy a strict positive real condition to ensure stability [12].

Alternatively, the poles of the IIR filter can be fixed at predetermined values, which preserves the linearity in parameters and leads to well-behaved adaptation properties [1].

4 Fixed-pole parallel and Kautz filters

In audio signal processing, fixed-pole filters are commonly used. The Kautz and the fixed-pole parallel filters are proven to have equivalent transfer functions when designed off-line [3], as their tap outputs span the same space. Additionally, the base functions of the Kautz filter are orthonormal, which results in convergence properties similar to that of FIR filters [1].

The general structure of the parallel second-order filter can be found in Fig. 2. The second-order sections can be implemented as either direct-form, or other structures [13]. Note that the structure of the second-order sections have direct impact on the parameters, for example, implementing a given pole-zero set using Chamberlin structure [13] will lead to different coefficients compared to a direct-form structure. This difference in coefficients affects the convergence properties if the second-order section is used in an adaptive filter realization.

For testing the parallel filter we first apply a direct-form structure, depicted in Fig. 3. Because the sections in parallel filters have real zeros, the term b_2 is always 0. In order to normalize the output power of the tap outputs, we use the structure in Fig. 4. In this case the two base functions of the second-order sections are highly correlated (delayed version of a bandpass filter output), therefore we expect poor convergence in the LMS algorithm. As an improvement, we suggest the use of an orthogonal second-order structure (Fig. 5), as it is equivalent to a second-order Kautz filter, thus, its two tap outputs are orthogonal. The parameters a_1 and a_2 are the same as in the direct form. The p and q normalizing terms were chosen so that the tap outputs will

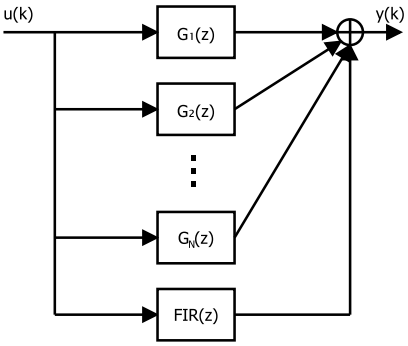

Fig. 2: Fixed-pole parallel filter having N conjugate complex pole pairs with optional FIR section.

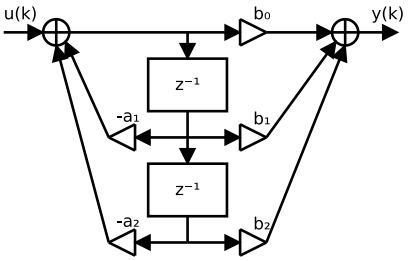

Fig. 3: Direct-form 2 second-order structure.

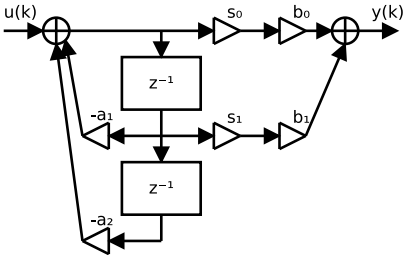

Fig. 4: Direct-form 2 second-order structure used for implementing adaptive fixed-pole parallel filters. The terms s_0 and s_1 are chosen so that the tap outputs will have the same output power when the input is white noise.

have the same output power when the input is white noise. The fixed-pole parallel filter that uses this new structure is termed "orthogonal parallel filter".

There are two variants of Kautz filters: the first uses complex arithmetic, while the second uses real values [4]. Later in the comparisons we assume that the real Kautz filter is implemented, and its second-order sections are realized using the usual direct-form 2 structure (Fig. 3), as it has the lowest computational demand [13].

Note that the choice of second-order structure, as well as the choice of the filter has an impact on computational demand too. According to Table 2, at high filter orders when implemented using direct-form second-order sections the fixed-pole parallel filter needs approximately 47% fewer operations when implemented using direct-form sections and 35% fewer operations when realized using orthogonal second-order sections, compared to the Kautz filter. This may be taken into consideration when implementing in embedded systems.

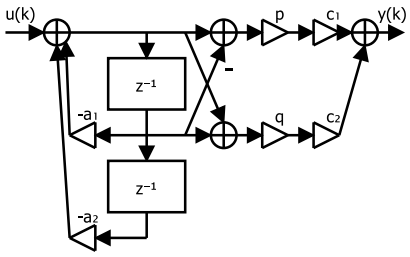

Fig. 5: Orthogonal second-order structure, with normalizing terms p and q.

Table 2: Number of arithmetic operations (multiplications and additions) required for the tested adaptive IIR filters having M-tap FIR sections and N conjugate-complex pole pairs implemented using direct-form 2 (DF2) or orthogonal second-order sections.

	Mult.	Add.
Fixed-pole parallel (DF2)	$6N+M$	$3N+M-1$
Fixed-pole parallel (orth.)	$6N+M$	$5N+M-1$
Kautz filter (DF2)	$9N+M+2$	$8N+M+1$
Delayed fixed-pole par. (DF2)	$6N+M$	$3N+M-1$
Delayed fixed-pole par. (orth.)	$6N+M$	$5N+M-1$
Delayed Kautz (DF2)	$9N+M+2$	$8N+M+1$

5 The delayed parallel and Kautz filters

The fixed-pole parallel filter has an optional parallel FIR path that is beneficial for modeling non-minimumphase systems since it allows efficiently modeling the rising part of impulse responses [14]. However, it has been shown in [5, 15] that this parallel FIR part can lead to numerical problems in real world applications (e.g., lower resolution fixed-point arithmetic), and a delayed variant has been proposed where the IIR sections are delayed to avoid overlap with the FIR part, as shown in Fig. 6.

Since in adaptive filtering we aim to have the base functions as independent as possible, we use this delayed variant in our investigations. This assures that the cross correlations between the FIR taps and the IIR parts are zero.

Similarly to the fixed-pole parallel filter, the Kautz filter can also have a FIR part by forcing some of the poles

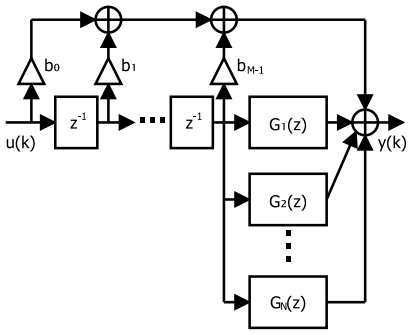

Fig. 6: Delayed parallel second-order filter having N conjugate complex pole pairs and M FIR taps.

to zero. This idea has been proposed earlier in literature, as it can help the modeling of non-minimumphase transfer functions [4, 16].

Analogous to the delayed fixed-pole parallel filter, we suggest to call the structure in Fig. 7 as "delayed Kautz filter". The structure can be deduced by placing the part of the Kautz backbone containing the sections with poles at the origin to the input of the filter. This way, the first sections of the Kautz filter are reduced to a FIR filter, while the other sections can be implemented as the real Kautz filter. Note that this topology does not require more operations than an independent FIR and a Kautz filter implementation.

The most important feature of the delayed Kautz filter is the orthogonality and unity power of the tap outputs. This means that when used with the LMS algoritm, the **R** matrix of the adaptive filter is a unity matrix, when the input is white noise.

6 Comparisons

For comparing the different filter structures, we used the NLMS algorithm as the method for system identification (Fig. 1). The input was white noise uniformly distributed in range $[-1;+1]$. The system to be identified was implemented using a 10000-tap long FIR filter, whose coefficients were based on actual impulse response measurements.

Seven filters were compared: two fixed-pole second-order parallel filters without FIR section (implemented using direct-form or orthogonal second-order sections), a Kautz filter, a delayed second-order parallel filter (implemented using both second-order structures), a

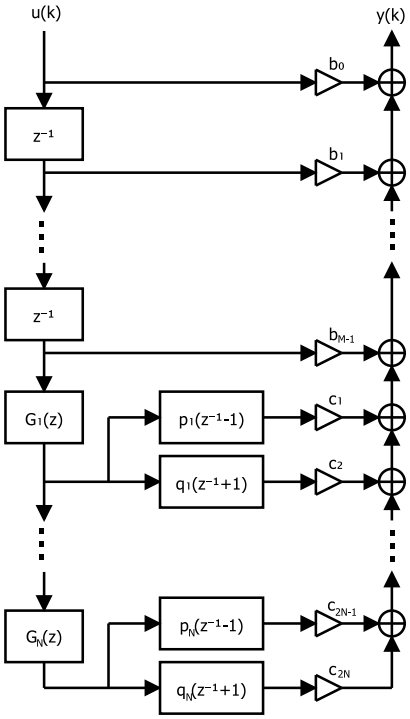

Fig. 7: Structure of the delayed Kautz filter. The FIR part has M taps, followed by the (real) Kautz implementation, which has N conjugate complex pole pairs.

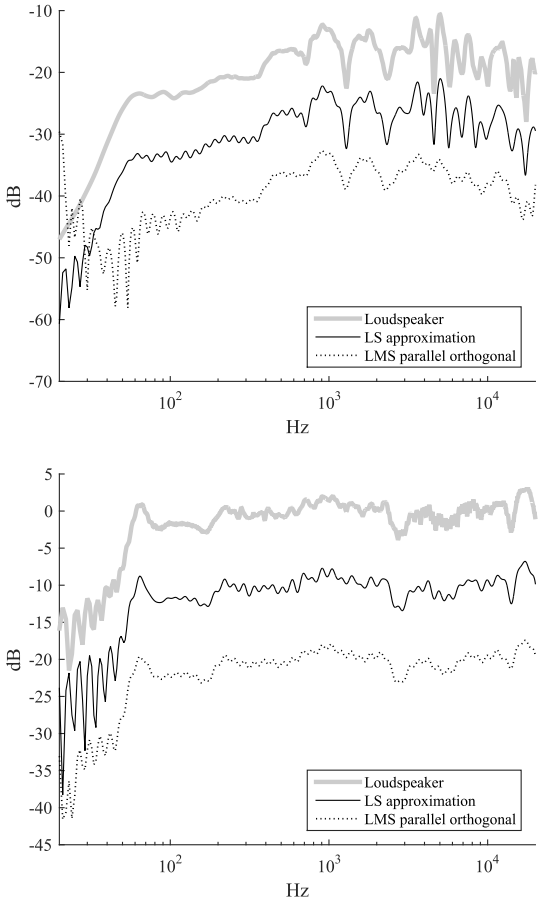

Fig. 8: Magnitude plots of the example transfer functions (grey lines). Top: minimumphase one-way loudspeaker; bottom: non-minimumphase two-way loudspeaker. The LS approximations of the non-delayed filters are plotted using thin black lines. The magnitude responses of the fixed-pole parallel filters using orthogonal second-order sections, after 65536 samples, are also plotted (dotted lines).

delayed Kautz filter and as a reference, a FIR filter. All of the filters had 100 free parameters for adaptation: the non-delayed IIR filters had 50 fixed conjugate complex pole pairs, and the delayed filters had 40 fixed conjugate complex pole pairs and 20 FIR taps. The pole frequencies were chosen according to a logarithmic scale between 20 Hz and 20 kHz, assuming 44.1 kHz sampling frequency. The quality factors of the poles were set so that the neighboring sections had their magnitude responses cross at their -3 dB point [17]. Before the adaptation, all filters were initialized to approximate a unity gain frequency response via an off-line least squares design.

Note that for fixed-pole parallel filters, the higher the number of poles, the sharper the peaks of the second-order sections, which leads to lower cross-correlation between the outputs of the individual sections. This means that the convergence properties of the Kautz and fixed-pole parallel filters become closer and closer when the filter order is increased.

To evaluate the differences in convergence, we calculated a mean transfer function error (MTFE) on a logarithmic frequency scale. First, we evaluated the transfer functions of both the system to be identified and the adaptive filter at a given time point during the adaption process. Then, the transfer functions were smoothed and sampled at 100 points per octave between 20 Hz and 20 kHz, assuming $f_\mathrm{s} = 44.1$ kHz sampling rate. The resulting sampled transfer functions were conver-

Table 3: Number of arithmetic operations required for implementing LMS-based filters for the examples. The total number of adaptive coefficients is 100. The results contain the computational demand of the Normalized-LMS algorithm too.

	Arithmetic operations
LMS FIR	604
LMS Fixed-pole parallel (DF2)	852
LMS Fixed-pole parallel (orth.)	952
LMS Kautz filter (DF2)	1256
LMS Delayed fixed-pole par. (DF2)	802
LMS Delayed fixed-pole par. (orth.)	882
LMS Delayed Kautz (DF2)	1126

ted to magnitude responses and subtracted from each other, and then the absolute differences were summed. This metric was calculated at every 256 samples for all structures. For each of the filters, the μ step-size parameter is tuned in a way that the curves would be parallel with each other on the first 12800 samples, if possible. For reference, the MTFE plots also contain the mean transfer function error of the offline designed filters, based on the LS method.

In our investigation, we used two example transfer functions for testing the algorithms: a minimumphase one-way loudspeaker (Fig. 8 top), and a larger, two-way loudspeaker with non-minimumphase response (Fig. 8 bottom). In the figures, we marked the results of the off-line LS designs as well as the magnitude responses of the adaptive (non-delayed) fixed-pole parallel filters that are implemented using orthogonal second-order sections, as examples.

As a first comparison, we used the response of the minimumphase one-way loudspeaker (Fig. 8 top) as the modeled system. The loudspeaker impulse response was processed using the `rceps` function in MATLAB, to get a truly minimumphase impulse response. The MTFE plots of the filters identifying this transfer function can be found in Fig. 9 and 10. It can be seen that amongst the non-delayed filters the Kautz filter has the fastest convergence and lowest mean magnitude error, but the delayed filters perform even better. Their off-line least-squares approximation also has lower MTFE compared to the non-delayed IIR filters.

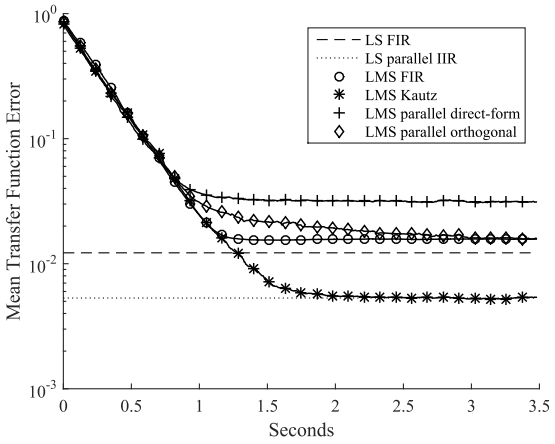

Fig. 9: Mean transfer function error by time, for a minimumphase one-way loudspeaker response. Only the non-delayed filter versions are displayed here.

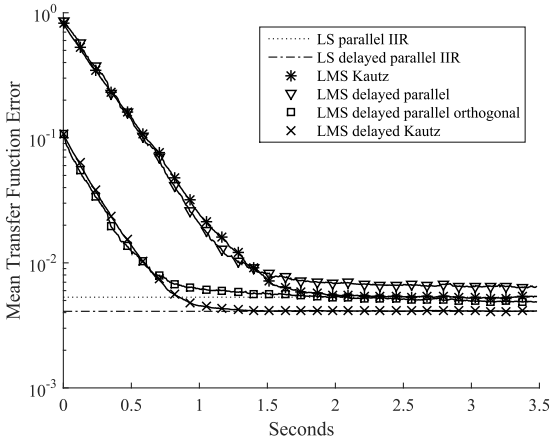

Fig. 10: Mean transfer function error by time, for a minimumphase one-way loudspeaker response. For comparison, the adaption curve of the Kautz filter is displayed among the delayed filters.

Next, the response of the non-minimumphase two-way loudspeaker (Fig. 8 bottom) was identified. As can be seen in Fig. 11, the FIR filter has the highest remaining error. This is caused by the longer impulse response of this loudspeaker compared to that of Fig. 9, which does not fit in the same FIR filter length. According to Fig. 12, the delayed filters with FIR sections have

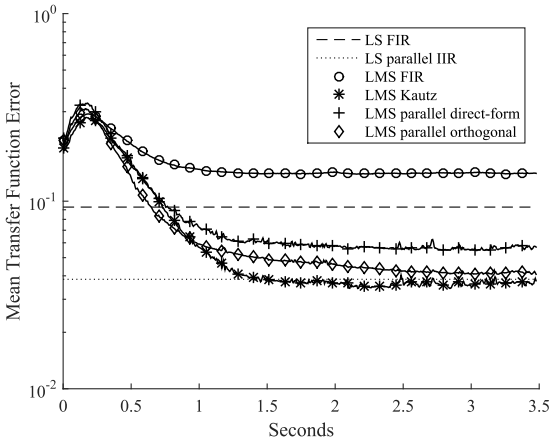

Fig. 11: Mean transfer function error by time, for a non-minimumphase two-way loudspeaker response. Only the non-delayed filter versions are displayed here.

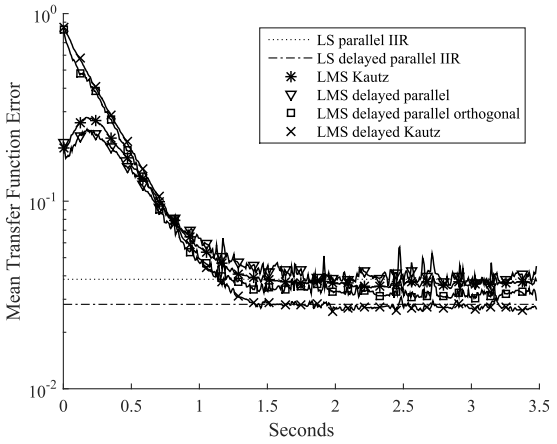

Fig. 12: Mean transfer function error by time, for a non-minimumphase two-way loudspeaker response. For comparison, the adaption curve of the Kautz filter is displayed among the delayed filters.

lower MTFE compared to the non-delayed IIR filters. Amongst them, the delayed Kautz filter has the lowest remaining MTFE, closely followed by the delayed fixed-pole parallel filter with orthogonal sections.

Based on the MTFE plots in Figures 9–12, we recommend to use the delayed Kautz filter structure in LMS-based adaptive systems: it has the lowest remaining mean transfer function error and the fastest convergence among the examined structures. The computational demand of the compared structures were calculated according to Tab. 1 and 2, and displayed in Tab. 3. It can be seen in Tab. 3 that the delayed filters have lower computational demand compared to their non-delayed counterparts, given the filters have the same amount of free parameters.

7 Conclusion

This paper compared LMS-based adaptive implementations of the most common fixed-pole IIR filters used in audio, including their delayed versions, and also presented a new parallel filter structure with orthogonal second-order sections. Based on the comparison, we suggest to use the delayed Kautz structure in LMS-based adaptive audio filters, if its computational demand can be satisfied. As an alternative with lower number of arithmetic operations we recommend the delayed fixed-pole parallel filter, with its second-order sections implemented using the orthogonal structure.

Future research includes testing of the same filters in equalization applications and topologies, such as the Filtered-x LMS algorithm, with extension to linear time-variant systems. Other structures are possible too, including the resonator-based filter [18]. Further comparisons should include the Recursive Least Squares (RLS) algorithm [10] and frequency-domain adaptive algorithms [19] too.

References

[1] G. A. Williamson and S. Zimmermann, "Globally convergent adaptive IIR filters based on fixed pole locations," *IEEE transactions on signal processing*, vol. 44, no. 6, pp. 1418–1427, 1996.

[2] A. Härmä, M. Karjalainen, L. Savioja, V. Välimäki, U. K. Laine, and J. Huopaniemi, "Frequency-warped signal processing for audio applications," *J. Audio Eng. Soc.*, vol. 48, no. 11, pp. 1011–1031, Nov. 2000.

[3] B. Bank, "Audio equalization with fixed-pole parallel filters: An efficient alternative to complex smoothing," *J. Audio Eng. Soc.*, vol. 61, no. 1/2, pp. 39–49, Jan. 2013.

[4] T. Paatero and M. Karjalainen, "Kautz filters and generalized frequency resolution: Theory and au-

dio applications," *J. Audio Eng. Soc.*, vol. 51, no. 1–2, pp. 27–44, Jan./Feb. 2003.

[5] B. Bank and J. O. Smith, "A delayed parallel filter structure with an FIR part having improved numerical properties," in *Proc. 136nd AES Conv., Preprint No. 9084*, Berlin, Germany, Apr. 2014.

[6] B. Widrow and E. Walach, *Adaptive inverse control, reissue edition: a signal processing approach*. John Wiley & Sons, 2008.

[7] V. Välimäki and J. Reiss, "All about audio equalization: Solutions and frontiers," *Applied Sciences*, vol. 6, no. 5, p. 129, 2016.

[8] M. Guldenschuh, "Least-mean-square weighted parallel IIR filters in active-noise-control headphones," in *2014 22nd European Signal Processing Conference (EUSIPCO)*. IEEE, 2014, pp. 1367–1371.

[9] S. S. Haykin, B. Widrow, and B. Widrow, *Least-mean-square adaptive filters*. Wiley Online Library, 2003, vol. 31.

[10] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*. Springer, 1997.

[11] S. Stearns, "Error surfaces of recursive adaptive filters," *IEEE Transactions on Circuits and Systems*, vol. 28, no. 6, pp. 603–606, 1981.

[12] C. Johnson, M. Larimore, J. Treichler, and B. Anderson, "SHARF convergence properties," *IEEE Transactions on Circuits and Systems*, vol. 28, no. 6, pp. 499–510, 1981.

[13] K. Horváth and B. Bank, "Optimizing the numerical noise of parallel second-order filters in fixed-point arithmetic," *Journal of the Audio Engineering Society*, vol. 67, no. 10, pp. 763–771, 2019.

[14] B. Bank, "Direct design of parallel second-order filters for instrument body modeling," in *Proc. Int. Computer Music Conf.*, Copenhagen, Denmark, Aug. 2007, pp. 458–465.

[15] ——, "Converting infinite impulse response filters to parallel form [tips & tricks]," *IEEE Signal Processing Magazine*, vol. 35, no. 3, pp. 124–130, 2018, doi: https://doi.org/10.1109/MSP.2018.2805358.

[16] M. Karjalainen and T. Paatero, "Equalization of loudspeaker and room responses using Kautz filters: Direct least squares design," *EURASIP J. on Advances in Sign. Proc., Spec. Iss. on Spatial Sound and Virtual Acoustics*, vol. 2007, p. 13, 2007, article ID 60949, doi:10.1155/2007/60949.

[17] B. Bank, "Audio equalization with fixed-pole parallel filters: An efficient alternative to complex smoothing," in *Proc. 128th AES Conv., Preprint No. 7965*, London, UK, May 2010.

[18] G. Péceli, "Resonator-based digital filters," *IEEE Transactions on Circuits and Systems*, vol. 36, no. 1, pp. 156–159, 1989.

[19] J. J. Shynk, "Frequency-domain and multirate adaptive filtering," *IEEE Signal processing magazine*, vol. 9, no. 1, pp. 14–37, 1992.