
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Liski, Juho; Bank, Balazs; Smith, Julius O.; Välimäki, Vesa
Converting Series Biquad Filters into Delayed Parallel Form

Published in:
IEEE Transactions on Signal Processing

DOI:
[10.1109/TSP.2019.2919419](https://doi.org/10.1109/TSP.2019.2919419)

Published: 15/07/2019

Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Liski, J., Bank, B., Smith, J. O., & Välimäki, V. (2019). Converting Series Biquad Filters into Delayed Parallel Form: Application to Graphic Equalizers. *IEEE Transactions on Signal Processing*, 67(14), 3785-3795. Article 8723568. <https://doi.org/10.1109/TSP.2019.2919419>

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

Converting Series Biquad Filters Into Delayed Parallel Form: Application to Graphic Equalizers

Juho Liski , Balázs Bank , *Member, IEEE*, Julius O. Smith , *Member, IEEE*, and Vesa Välimäki , *Fellow, IEEE*

Abstract—Digital filter transfer functions can be converted between the direct form and parallel connections of elementary sections, typically second-order (“biquad”) sections. The conversion from direct to parallel form is performed using a partial fraction expansion, which usually requires long division of polynomials when expanding proper and improper transfer functions. This paper focuses on the conversion of a series of biquad sections to the parallel form, and proposes a novel way to implement the partial-fraction expansion without the use of long division. Additionally, the resulting structure is the delayed parallel form in which the section gains remain small. The new design and previous methods are compared in a case study on graphic equalizer design. The delayed parallel filter is shown to use the same number of operations as the series form during filtering. The conversion of a recently proposed series graphic equalizer into the delayed parallel form leads to an improved parallel graphic equalizer design relative to all known prior approaches. The proposed conversion technique is widely applicable to the design of parallel infinite impulse response filters, which are becoming popular as they are well suited to implementation using parallel computers.

Index Terms—Digital filters, equalizers, IIR filters, polynomials.

I. INTRODUCTION

INFINITE impulse response (IIR) filter transfer functions can be implemented both as series and parallel second-order sections [1]–[4]. This paper presents a new method to convert a series IIR filter into a parallel form using the partial-fraction expansion (PFE). The usefulness of this method is exemplified by a graphic equalizer (EQ), which is easier to design in the series form but often advantageous to implement in the parallel form.

In signal processing, the same IIR filter transfer function can be implemented by various filter structures besides the direct-form variants [1]. Series or parallel second-order (“biquad”)

Manuscript received December 21, 2018; accepted April 29, 2019. Date of publication May 28, 2019; date of current version June 18, 2019. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mark A. Davenport. The work of J. Liski was supported by the Aalto ELEC Doctoral School. The work of B. Bank was supported by the BME Artificial Intelligence FIKP grant of EMMI (BME FIKP-MI/SC). (*Corresponding author: Juho Liski.*)

J. Liski and V. Välimäki are with the Acoustics Laboratory, Department of Signal Processing and Acoustics, Aalto University, FI-00076 Espoo, Finland (e-mail: juho.liski@aalto.fi; vesa.valimaki@aalto.fi).

B. Bank is with the Department of Measurement and Information Systems, Budapest University of Technology and Economics, Budapest 1117, Hungary (e-mail: bank@mit.bme.hu).

J. O. Smith is with the Center for Computer Research in Music and Acoustics, Music Department, Stanford University, Stanford, CA 94305 USA (e-mail: jos@cerma.stanford.edu).

This paper has supplementary downloadable material available at <http://ieeexplore.ieee.org>, provided by the authors.

Digital Object Identifier 10.1109/TSP.2019.2919419

IIR filter sections are most commonly used [5]. The parallel structure has several practical advantages, such as suitability for parallel implementation with GPUs [6] and beneficial quantization characteristics [5]. However, series biquad filter designs are more common, for example, in audio equalizers [7]–[11], mainly because the designs are intuitive and are often based on closed-form or easily solved formulas. In addition, traditional IIR filter design techniques based on the bilinear transform, such as Butterworth, Chebyshev, and Cauer, also give the filter in the pole-zero form that is closely related to the series structure when the complex poles and zeros are paired as second-order sections [12]. Thus, in many applications, converting a filter presented in a series second-order form (or in the equivalent pole-zero form) to parallel second-order sections can be beneficial.

A straightforward approach for such a conversion is to first convert the series form to a direct-form IIR filter by multiplying the numerator and denominator polynomials of the sections. From the direct-form transfer function, the parallel form can be obtained by PFE [1]–[3], [13], [14]. However, for large filter orders and/or poles near the unit circle, the direct-form realization of the transfer function can easily become unstable due to numerical errors. Thus, this procedure is not recommended unless the filter order is relatively small (depending on the poles, filter orders of 10–20 can be converted from series to parallel by this method).

In [4], two methods are proposed for series-to-parallel conversion. The first one is based on a least-squares (LS) filter design inspired by fixed-pole parallel filters [15]. The denominators of the parallel form are the same as those of the series form and, therefore, finding the numerator coefficients of the parallel transfer function is a linear problem. Thus, the numerator parameters are obtained as a LS solution where the error between the impulse response (or the frequency response) of the series and parallel forms is minimized.

The second method proposed in [4] performs the PFE on the series form directly, without computing the direct-form transfer function. To be able to do so, one must factor out one or two zeros to make the transfer function strictly proper, since that is generally required for the PFE [1]. The zeros are then implemented as a first- or second-order FIR filter in series with the parallel structure.

Alternatively, an IIR filter could be designed directly in the parallel form. For example, a direct design method for fixed-pole parallel filters proposed by Bank, in which the zeros are found with an LS solution [16], [17]; a vector-fitting method by Wong *et al.*, which iteratively optimizes the pole positions of an

IIR filter using LS to approximate FIR filter response [18]; and a method proposed by Qi *et al.*, who optimize the coefficients and the poles to design an IIR filter based on target frequency response [19] all lead to parallel first or second-order sections similar to the PFE form. The two latter methods are currently unavailable for graphic equalizer design, but the first one is used for comparison in this paper.

In this work, we propose a novel way to convert a series IIR filter into the parallel form that requires neither an LS design nor the series FIR filter of [4]. This method is inspired by the direct-to-parallel conversion technique of Orfanidis [2], which, in contrast to all the literature the authors are aware of, proposes computing the PFE of proper transfer functions without requiring polynomial long division. This paper shows that the direct-to-parallel conversion can be performed without the long division even for improper transfer functions and demonstrates why this is beneficial for the series-to-parallel conversion. In addition, we convert the filters to the delayed parallel form [3], [4], [20] in which the FIR part of the filter does not interact with the responses of the parallel biquad sections, leading to better numerical behavior.

Finally, we present an application example in which an accurate series graphic EQ is converted into the delayed parallel form. The conversion is simplified in this case by the fact that the denominators of the series design remain the same in the parallel structure, although the poles must be solved for the PFE calculation. Comparison with other parallel graphic EQ designs shows the advantages of the proposed method: fast design, good accuracy of approximation, and efficient implementation.

This paper is organized as follows. Section II reviews the use of the PFE in digital filter design. Section III introduces a new method for performing the PFE without the long division. Section IV presents an application example in which an accurate series design of a graphic EQ is converted to the delayed parallel form. Section V compares the new design with the series form and with previous parallel designs. Finally, conclusions are drawn in Section VI.

II. PARTIAL FRACTION EXPANSION

PFE is used to convert IIR filters into the parallel form [3]. For example, a high-order IIR filter can be presented as a sum of individual pole terms, where the coefficients are the residues of corresponding poles. In real-world digital filters, the coefficients of the first-order parallel terms are often complex valued, and thus the complex-conjugate pole pairs must be combined into second-order sections to obtain real-valued numerator coefficients. If the original transfer function is strictly proper, i.e., the order of the numerator N_{Num} is less than that of the denominator N_{Den} , only parallel first-order transfer function terms are obtained [3]. However, if the transfer function is proper ($N_{\text{Num}} = N_{\text{Den}}$) or improper ($N_{\text{Num}} > N_{\text{Den}}$), an FIR part also emerges with the length of $L = N_{\text{Num}} - N_{\text{Den}} + 1$.

When the original filter transfer function is $H(z) = B(z)/A(z)$, its PFE can be written as [3]

$$H(z) = F_1(z) + \sum_{n=1}^N \frac{r_n}{1 - p_n z^{-1}}, \quad (1)$$

where r_n are the residues corresponding to poles p_n and $F_1(z)$ is the FIR part. We assume distinct poles ($p_n \neq p_m$ for $\forall n \neq m$) throughout this work, but the extension to repeated poles [3] is straightforward. The FIR part of the PFE is usually obtained by long division [3], where the numerator polynomial is divided by the denominator polynomial. After the long division, or if the transfer function is already strictly proper, the residues are obtained using the Heaviside cover-up method [21]:

$$r_n = (1 - p_n z^{-1})H'(z)|_{z=p_n}, \quad (2)$$

where $H'(z) = H(z) - F_1(z)$ is the strictly proper remainder after long division. The term $(1 - p_n z^{-1})$ in the denominator of $H'(z)$ is canceled out to determine the residue r_n of pole p_n .

A. Delayed Parallel Form

Traditionally, the long division in a PFE is performed on polynomials starting with the highest powers of z^{-1} [13]. This leads to a parallel filter in which the FIR part and the IIR part overlap. There is, however, an alternate PFE form featuring a delayed IIR part [3]. When the coefficients are simply reversed during long division, i.e., long division starts with the lowest powers of z^{-1} , the following delayed PFE form is obtained [3]:

$$H(z) = F(z) + z^{-L} \sum_{n=1}^N \frac{\tilde{r}_n}{1 - p_n z^{-1}}, \quad (3)$$

where $F(z)$ is the FIR part of length L and \tilde{r}_n is the residue corresponding to the delayed IIR part. The FIR and IIR parts do not overlap in the impulse response of this form. Note that the FIR part and residues of this form differ from the ones obtained with the traditional PFE, but the resulting transfer functions $H(z)$ are equal. Here, $F(z)$ corresponds directly to the first L samples of the impulse response of $H(z)$. This form has been found to yield better numerical behavior than the form with overlapping FIR and IIR parts [4], [20]. Our intention is to convert series IIR filters to this type of delayed form.

B. PFE of Proper Transfer Functions Without Long Division

Orfanidis has suggested a method to determine the PFE of a proper transfer function without the long division [2]. This method produces the traditional form of the PFE, i.e., the form in which the IIR part is not delayed in comparison to the FIR part. Now, since proper transfer functions are expanded into partial fraction form, the length of the FIR part is $L = N_{\text{Num}} - N_{\text{Den}} + 1 = 1$ and (1) can be written as

$$H(z) = F_1 + \sum_{n=1}^N \frac{r_n}{1 - p_n z^{-1}}, \quad (4)$$

where F_1 is a constant forming the FIR part of the filter (order 0) and $N_{\text{Den}} = N_{\text{Num}}$.

Orfanidis suggests that the IIR part of (4) is considered first by solving for the residues using the Heaviside cover-up method (2) [2]. For this, the poles of the transfer function $H(z) = B(z)/A(z)$ must be known. The n th residue is then obtained by canceling the n th pole and evaluating the remaining transfer function at the pole. Note that Orfanidis applies the method directly to the proper transfer function $H(z)$ [2].

(Section III-A shows that the same residues are obtained in this way as with long division.) Finally, the constant F_1 is computed by evaluating the original transfer function at $z = 0$:

$$F_1 = H(z)|_{z=0}. \quad (5)$$

III. NOVEL PFE CONVERSION METHODS

In this section, the method suggested by Orfanidis is extended to the case of improper transfer functions ($N_{\text{Num}} > N_{\text{Den}}$), and, inspired by it, a novel series-to-parallel conversion producing the delayed PFE form is proposed.

A. Obtaining the PFE of Improper Transfer Functions

Corresponding to the usual procedure, Orfanidis also suggests the use of polynomial long division to obtain the FIR part for the $N_{\text{Num}} > N_{\text{Den}}$ case [2]. Here, we show why the idea of not using long division before PFE is valid for proper transfer functions, and that it can also be extended to the improper case. The derivations shown here are valid for both delayed and non-delayed parallel forms.

An improper transfer function $H(z) = B(z)/A(z)$ in which the order N_{Num} of $B(z)$ is larger than the order N_{Den} of $A(z)$ is usually decomposed to a strictly proper transfer function $H'(z) = B'(z)/A(z)$ and an FIR part $F(z)$ in parallel:

$$H(z) = \frac{B(z)}{A(z)} = F(z) + \frac{B'(z)}{A(z)}. \quad (6)$$

Here, we show that applying the Heaviside cover-up method on the original transfer function $H(z)$ gives the same residues as when it is performed on the strictly proper transfer function $H'(z)$. By using (6), the residues of $H(z)$ are computed as

$$\begin{aligned} r_n &= (1 - p_n z^{-1})H(z)|_{z=p_n} \\ &= (1 - p_n z^{-1})F(z)|_{z=p_n} + (1 - p_n z^{-1})\frac{B'(z)}{A(z)}\Big|_{z=p_n}, \end{aligned} \quad (7)$$

where the first term is zero, since $z = p_n$. The second part is nonzero because $A(z)$ contains the term $(1 - p_n z^{-1})$, and thus r_n actually equals the residue computed from the strictly proper transfer function $H'(z)$ using (2).

This means we obtain the same residues before or after long division. However, we need to find an alternative method to obtain the FIR part $F(z)$. This is particularly simple for the delayed PFE form, because in that form there is no overlap between the FIR part and the impulse-response of the IIR sections. Therefore, the coefficients of the FIR part f_k are simply given by the first L samples of the filter impulse response $h(k)$, $k = 0, 1, \dots, L - 1$. However, to obtain the FIR part of the traditional non-delayed PFE, the first $L = N_{\text{Num}} - N_{\text{Den}}$ samples of the parallel IIR sections may be subtracted from the first L samples of the impulse response of the original filter, since in this case the FIR and IIR parts overlap.

Accordingly, an IIR filter $H(z) = B(z)/A(z)$ having numerator order N_{Num} and denominator order N_{Den} can be converted to parallel second-order form as follows:

- 1) Compute the residues r_n from the original transfer function using the cover-up method (2).

- 2) Combine complex-conjugate pairs of poles p_n and \bar{p}_n and residues r_n and \bar{r}_n to obtain real second-order sections.
- 3) Determine the FIR part $F(z)$ of the length $L = N_{\text{Num}} - N_{\text{Den}}$ by (i) (delayed parallel form) computing the first L samples of the impulse-response $h(k)$ of the original filter $H(z)$ (achieved by simply ‘‘running’’ the filter on a unit-impulse input signal), and setting $f_k = h(k)$, $k = 0, 1, \dots, L - 1$, or (ii) (traditional non-delayed parallel form) computing the first L samples of the impulse response $h(k)$ of the original filter and the first L samples of the parallel IIR part and subtracting the latter from the former.

Finally, note that when converting very high-order ($N_{\text{Den}} > 100$) IIR filters to parallel form, the numerically more robust LS approach is recommended over the Heaviside cover-up method [4].

B. Novel Series-to-Parallel Conversion Without Long Division

Next, we show how a series filter structure can be efficiently converted to the delayed parallel form without long division. The method is illustrated for a series of second-order sections, but it works as well for any series-connected filter sections. The key idea is to represent the original biquad sections non-causally, i.e., as a function of z instead of z^{-1} . This change of representation corresponds to the reversal of the coefficients in the traditional PFE when determining the desired PFE form.

A biquad transfer function may be written in the z domain as

$$H(z) = k_0 \frac{1 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}, \quad (8)$$

where k_0 is a gain factor, b_1 and b_2 are the feedforward coefficients, and a_1 and a_2 are the feedback coefficients. The advantage of the proposed method for the biquad case is that it does not utilize long division, and thus the rational form of the transfer function does not need to be evaluated. Thus, in addition to the case of a single biquad section, the method also works for a filter presented as biquad sections in series, where the number of sections M is arbitrary, i.e., the transfer function of the filter is

$$H(z) = G_0 \prod_{m=1}^M \frac{1 + b_{1,m} z^{-1} + b_{2,m} z^{-2}}{1 + a_{1,m} z^{-1} + a_{2,m} z^{-2}}, \quad (9)$$

where the gain factor G_0 is

$$G_0 = \prod_{m=1}^M k_{0,m}. \quad (10)$$

The structure of such a series filter is shown in Fig. 1(a), and the structure of a single biquad filter section is shown in Fig. 1(b).

The proposed method starts with the non-causal form of (9):

$$H(z) = \left(\frac{z^2}{z^2}\right)^M H(z) = G_0 \prod_{m=1}^M \frac{z^2 + b_{1,m} z + b_{2,m}}{z^2 + a_{1,m} z + a_{2,m}}. \quad (11)$$

Now, the poles are found for each second-order section using the quadratic formula. The number of poles determines the number of residues needed to be calculated. For the residue calculation,

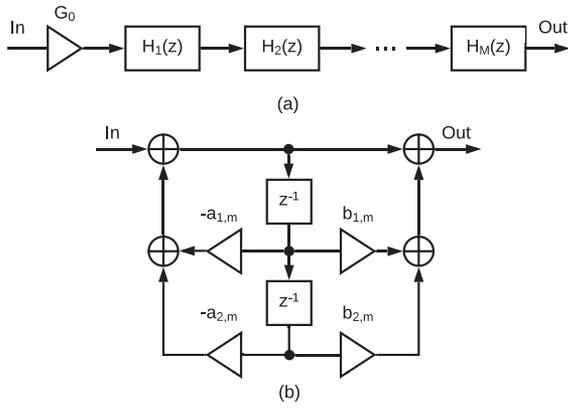


Fig. 1. (a) Series IIR filter structure, and (b) the block diagram of a single second-order section, i.e., a biquad filter. The gain factors of the biquad sections have been consolidated in the overall gain G_0 .

instead of (4), the PFE of $H(z)$ is assumed to equal

$$H(z) = F + \sum_{n=1}^N \frac{\tilde{r}_n}{z - p_n}, \quad (12)$$

where F and \tilde{r}_n correspond to the PFE form including the delayed IIR part. Again, the numerator and denominator are of the same order, which leads to the FIR part having length one. The residues are found first using the Heaviside cover-up method, which now has the form

$$\tilde{r}_n = (z - p_n)H(z)|_{z=p_n}. \quad (13)$$

In the next step, the delay is introduced in the IIR part by converting it back to the standard, causal filter notation:

$$\begin{aligned} H(z) &= F + \sum_{n=1}^N \frac{\tilde{r}_n}{z - p_n} = F + \sum_{n=1}^N \frac{z^{-1}\tilde{r}_n}{z^{-1}(z - p_n)} \\ &= F + z^{-1} \sum_{n=1}^N \frac{\tilde{r}_n}{1 - p_n z^{-1}}. \end{aligned} \quad (14)$$

The order of the delay for the IIR part is seen to be $L = N_{\text{Num}} - N_{\text{Den}} + 1 = 1$ (see (3)), which is the desired result. In order to solve for F , (9) and (14) are combined as

$$\begin{aligned} H(z) &= G_0 \prod_{m=1}^M \frac{1 + b_{1,m}z^{-1} + b_{2,m}z^{-2}}{1 + a_{1,m}z^{-1} + a_{2,m}z^{-2}} \\ &= F + z^{-1} \sum_{n=1}^N \frac{\tilde{r}_n}{1 - p_n z^{-1}}, \end{aligned} \quad (15)$$

where F is the only unknown. In Orfanidis's method, the FIR constant is solved by evaluating $H(0)$. However, this requires positive powers of z , and the equations are already written with negative powers of z . In addition, we know that the value of F is constant, and thus we obtain its value directly by letting $z \rightarrow \infty$ in (15). Now, terms with negative powers of z go to zero (including the sum term on the right-hand side) and the

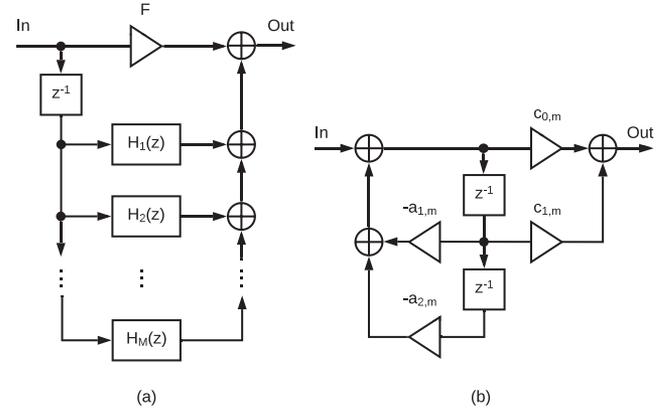


Fig. 2. (a) Parallel filter structure with delayed IIR part, and (b) the block diagram of a single second-order filter.

product term on the left-hand side equals 1, resulting in

$$F = G_0, \quad (16)$$

which is the first sample of the filter impulse response $h(0)$. This procedure applies to the series biquads case, and, if $N_{\text{Num}} > N_{\text{Den}}$, the FIR part is obtained as mentioned in Section III-A.

Since the first-order PFE terms obtained from a real-valued filter may be complex-valued, complex-conjugate pole pairs are combined. Existing real poles are also combined in order to obtain a parallel filter consisting of the FIR part and second-order IIR sections, with at most one section being effectively first order. This results in the same denominators as in the original series biquads and in new numerators with one feedforward coefficient less. One such second-order section $H_p(z)$ can be written as

$$\begin{aligned} H_p(z) &= \frac{r_1}{1 - p_1 z^{-1}} + \frac{r_2}{1 - p_2 z^{-1}} \\ &= \frac{(r_1 + r_2) - (r_1 p_2 + r_2 p_1) z^{-1}}{(1 - p_1 z^{-1})(1 - p_2 z^{-1})}. \end{aligned} \quad (17)$$

The final transfer function of the parallel biquad sections is of the form

$$H(z) = F + z^{-1} \sum_{m=1}^M \frac{c_{0,m} + c_{1,m} z^{-1}}{1 + a_{1,m} z^{-1} + a_{2,m} z^{-2}}, \quad (18)$$

where $c_{0,m} = \tilde{r}_{(2m-1)} + \tilde{r}_{(2m)}$ and $c_{1,m} = -\tilde{r}_{(2m-1)}p_{(2m)} - \tilde{r}_{(2m)}p_{(2m-1)}$. Now, the series IIR filter has been fully converted to the corresponding parallel form without evaluating the full direct-form transfer function.

The parallel structure realizing (18) is illustrated in Fig. 2(a), and the block diagram of the single second-order section is shown in Fig. 2(b). The proposed structure of Fig. 2(a) contains one unit delay more and M extra additions compared to Fig. 1(a). However, compared to the original biquad form (cf. Fig. 1(b)), the structure in Fig. 2(b) contains one feedforward path less, as also implied by (18), which means that it has one addition less per biquad section, or M additions less for the whole filter. It then turns out that all in all the delayed parallel structure requires exactly the same number of additions and multiplications as the series form.

TABLE I
CENTER FREQUENCIES AND BANDWIDTHS FOR THE 31 FILTERS OF THE THIRD-OCTAVE GRAPHIC EQ. THE ADJUSTED BANDWIDTHS OF THE SIX HIGHEST BAND FILTERS ARE SHOWN IN ITALICS

	1	2	3	4	5	6	7	8	9	10	11
f_c (Hz)	19.69	24.80	31.25	39.37	49.61	62.50	78.75	99.21	125.0	157.5	198.4
f_B (Hz)	9.178	11.56	14.57	18.36	23.13	29.14	36.71	46.25	58.28	73.43	92.51
	12	13	14	15	16	17	18	19	20	21	22
f_c (Hz)	250.0	315.0	396.9	500.0	630.0	793.7	1000	1260	1587	2000	2520
f_B (Hz)	116.6	146.9	185.0	233.1	293.7	370.0	466.2	587.4	740.1	932.4	1175
	23	24	25	26	27	28	29	30	31		
f_c (Hz)	3175	4000	5040	6350	8000	10080	12700	16000	20160		
f_B (Hz)	1480	1865	2350	<i>2846</i>	<i>3502</i>	<i>4253</i>	<i>5038</i>	<i>5689</i>	<i>5573</i>		

IV. CASE STUDY: ACCURATE PARALLEL GRAPHIC EQUALIZER DESIGN

In this section, the novel PFE method of Section III is applied to the design of a graphic EQ. Both parallel and series forms are used in graphic EQ design [22]. However, a major difference between the forms concerns the design of the filter coefficient values. Graphic EQ design is a magnitude-only problem, where the target response specifies the filter gain at different frequencies [22], [23]. For a parallel filter structure, the magnitude of the sum depends on the relative phases of the component sections, and thus, the phase response must be specified in some way for the component sections even when the overall phase is not specified [15], [17], [24]. Minimum phase is a typical choice [1] for the overall equalizer phase, since that is in some sense the “easiest” for an IIR filter to provide.

On the other hand, for a series filter structure, a phase-sensitive design is not needed, since the total transfer function is the product of the transfer functions of the sections, and thus, the total magnitude response is the product of the individual magnitude responses, or, in decibels (dB), the dB-magnitude response is the sum of the individual dB-magnitude responses. This simplifies the design significantly [8], [22]. However, the parallel design has favorable properties compared to the series design, such as better dynamic-range distribution and numerical accuracy. Consequently, we show that it pays to convert a series graphic equalizer into the delayed parallel form, as the resulting filter has advantages in comparison to previously published series designs.

A. Third-Octave Graphic Equalizer Design

The series graphic EQ proposed by Välimäki and Liski [11], [25], which we call ACGE (accurate cascade graphic EQ), offers a fast design and accurate results utilizing a single second-order IIR filter section per band. Here, we expand this series filter design method into the superior delayed parallel form.

The ACGE employs a second-order IIR peak/notch filter given by Orfanidis [2] in which the reference gain at dc is set to 1:

$$H(z) = \frac{1 + G\beta - 2 \cos(\omega_c)z^{-1} + (1 - G\beta)z^{-2}}{1 + \beta - 2 \cos(\omega_c)z^{-1} + (1 - \beta)z^{-2}}, \quad (19)$$

where G is the linear peak gain, $\omega_c = 2\pi f_c/f_s$ is the normalized center frequency in radians (the center frequencies in Hertz are shown in Table I for the third-octave EQ), f_s is the sampling

frequency, β is defined as

$$\beta = \begin{cases} \tan(B/2), & \text{when } G = 1, \\ \sqrt{\frac{|G_B^2 - 1|}{|G^2 - G_B^2|}} \tan\left(\frac{B}{2}\right) & \text{otherwise,} \end{cases} \quad (20)$$

and G_B is the linear gain at the edges of bandwidth $B = 2\pi f_B/f_s$. Equation (19) can also be written in the form of (8). In that case, $k_0 = (1 + G\beta)/(1 + \beta)$, $b_1 = -(2 \cos(\omega_c))/(1 + G\beta)$, $b_2 = (1 - G\beta)/(1 + G\beta)$, $a_1 = -(2 \cos(\omega_c))/(1 + \beta)$, and $a_2 = (1 - \beta)/(1 + \beta)$. The standard audio sampling frequency $f_s = 44.1$ kHz is used in this work.

The IIR peak/notch filter in (19) and (20) allows for exact control over the filter bandwidth. We set the bandwidth B_m to equal the difference with the neighboring band center frequencies in order to have precise control of the behavior of the band filter at these points, thus improving the accuracy of the design. This way, the bandwidth of the third-octave EQ filters equal $B_m = (\sqrt[3]{2} - 1/\sqrt[3]{2})\omega_{c,m} \approx 0.4662\omega_{c,m}$. However, due to the filter asymmetry near the Nyquist limit, the bandwidth of the six uppermost filters must be adjusted by hand [25]. The resulting bandwidth values are shown in Table I. Furthermore, (19) and (20) allow for an unusual definition for the filter bandwidth. Traditionally, the bandwidth of a resonance is defined by its -3 -dB points, which refers to 0.707 times the linear gain. Here, however, we can select the dB gain at the bandwidth to be $g_{B,m} = cg_m$, where $0 < c < 1$ is a free design parameter. In the third-octave EQ, $c = 0.4$ is used [25].

The accurate series EQ is based on the self similarity of the filters [11]. Self-similar filters retain their shape in dB when filters with different gain settings are amplitude normalized, as is shown in Fig. 3. Due to the self-similarity, the normalized band filters of the EQ can be used as basis functions in the design process in order to control the interaction between the neighboring filters [11]. This idea was originally proposed by Abel and Berners [8] and Oliver and Jot [10].

The self-similarity is utilized by forming an interaction matrix \mathbf{B} from samples taken from the normalized dB amplitude responses. The interaction matrix further helps to control the interaction among different band filters [8], [10], [11], [25]. A filter of the form (19) is designed for each M bands using a prototype gain g_p , which for the third-octave EQ is 17 dB. The filters are then amplitude normalized by dividing them with the prototype gain, and the dB magnitude of these filters is evaluated in the band center frequencies as well as at the geometric

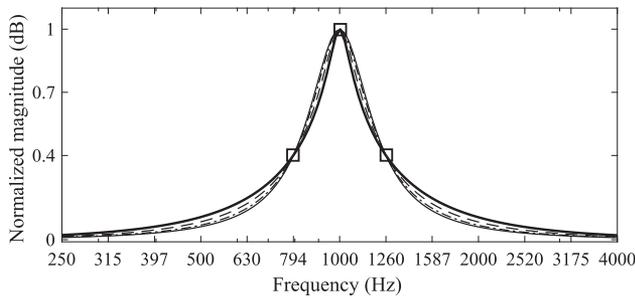


Fig. 3. Normalized magnitude response of the 1-kHz band filter with $c = 0.4$, showing the self-similarity. The different curves have different peak gains: 3 dB (thin solid), 10 dB (dash-dot), 17 dB (dashed), and 24 dB (bold solid line). The squares indicate the points where the responses meet. X-axis labels are the rounded third-octave center frequencies.

mean of these values between them. The inclusion of the extra points between the filter center frequencies decreases the error and improves the behavior of the EQ between the command points [11]. This amounts to 61 frequency points in total, and since there are 31 bands, the size of the interaction matrix \mathbf{B} is 61-by-31.

The interaction matrix is used to solve the optimal dB gains for the band filters in the LS sense by utilizing its inverse matrix \mathbf{B}^{-1} [13]. However, since \mathbf{B} is nonsquare, its pseudoinverse \mathbf{B}^+ is required instead [13], which is written as

$$\mathbf{B}^+ = (\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T. \quad (21)$$

Now, when the user selects the dB command gains for the 31 third-octave bands, a vector \mathbf{t}_1 with $2M - 1$ elements is formed containing the selected gain values in odd rows and their linearly interpolated intermediate values in even rows. The optimal dB gains are then solved with a well known LS solution

$$\mathbf{g} = \mathbf{B}^+ \mathbf{t}_1. \quad (22)$$

Finally, in order to reduce the error of the EQ to match the desired accuracy of 1 dB, one iteration step is required, where a new interaction matrix \mathbf{B}_1 is formed with \mathbf{g} obtained from (22) instead of the prototype gain g_p . The new filter dB gains are calculated as

$$\mathbf{g}_1 = \mathbf{B}_1^+ \mathbf{t}_1 = (\mathbf{B}_1^T \mathbf{B}_1)^{-1} \mathbf{B}_1^T \mathbf{t}_1, \quad (23)$$

which are then converted into linear gains to be used in (19) and (20).

We now have a finalized series EQ of the form (9), where $M = 31$. The next step is to convert this series design into a parallel one, which is done as explained in Section III-B. With the conversion process, we obtain the constant gain F as well as the new numerator coefficients $c_{0,m}$ and $c_{1,m}$. Note that the parallel EQ uses the same denominator coefficients as the series EQ, which we already have, but the poles must still be calculated based on (19) and (20) in order for the Heaviside method to be applicable.

The first step is to write the series EQ non-causally similar to (11). In a MATLAB implementation, this only affects the substitution of the pole values into (13), i.e., we use p_n instead of $1/p_n$. The poles p_n are solved with the quadratic equation or

the `roots` command in MATLAB. Next, the parallel first-order coefficients are obtained with (13). The `polyval` function can be used to implement the residue calculation efficiently in MATLAB, since it allows vector inputs. Thus, the numerator part of a single second-order section of $H(z)$ can be evaluated at all the pole values p_n as a single command. The denominator part is obtained similarly, but the pole cancellation must also be accounted for. Note that as `polyval` treats the input polynomial coefficients as a function of positive powers of z ordered as descending positive powers, the transfer functions of the series sections are already in their non-causal forms without additional operations. After all the second-order sections are evaluated at all the pole values, we solve (13) for all first-order residues \tilde{r}_n . The coefficients for the second-order sections are obtained from (18), and finally, the constant gain is solved with (16).

The obtained parallel EQ produces an impulse response identical to that of the original cascade EQ. Since the original cascade EQ design is minimum-phase, as all of its cascaded biquad filters are minimum-phase, also the parallel EQ obtained with the proposed conversion is minimum-phase.

B. Octave Graphic Equalizer Design

In addition to the third-octave ACGE, an octave version was also proposed [11]. The latter contains some differences when compared to the third-octave case, the largest of which is the number of bands and their center frequencies. The octave version of the ACGE uses 10 bands with the following center frequencies: 31.25, 62.5, 125, 250, 500, 1000, 2000, 4000, 8000, and 16000 Hz. The filter bandwidths have been defined in a similar manner to the third-octave version, i.e., they are the difference between the neighboring center frequencies, which here are $B_m = 1.5\omega_{c,m}$. However, the bandwidths of the three last filters must be adjusted to account for the filter asymmetry at high frequencies, and thus, the bandwidths are 46.88, 93.75, 187.5, 375.0, 750.0, 1500, 3000, 5580, 9360, and 12160 Hz.

The number of bands in the EQ also affects the size of the interaction matrix, which now becomes a 19-by-10 matrix. Since the interaction matrix is again nonsquare, its pseudoinverse is required for the optimal dB-gain calculation. From the two adjustable parameters of the ACGE, namely the prototype gain g_p and parameter c , the latter differs from the third-octave case. In order to achieve the desired accuracy, a value of $c = 0.30$ is used in the octave design [11], [25]. Apart from these differences in matrices and parameters, the conversion to parallel form is performed in the same way as for the third-octave case above.

V. COMPARISON OF GRAPHIC EQUALIZER DESIGNS

In this section, filters obtained using the proposed parallel EQ, called accurate parallel graphic equalizer (APGE), are compared to those produced using the original, cascade EQ design, ACGE [11], [25]. Furthermore, the proposed APGE design is compared with the ACGE and with a state-of-the-art parallel design, the parallel graphic EQ (PGE) presented in [17], [24], in terms of approximation error, design time, and the number of operations per output sample during filtering. The approximation error is evaluated in the same way as in [25], i.e., at the filter center

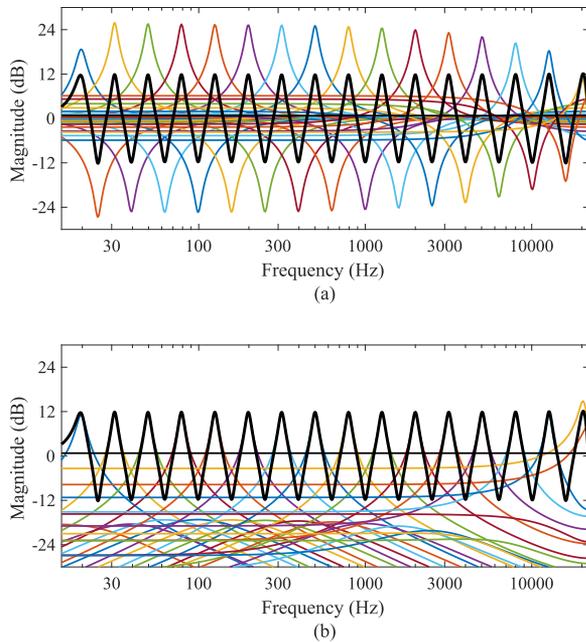


Fig. 4. Zigzag EQ setting with ± 12 dB realized using (a) the series EQ and (b) the proposed delayed parallel EQ (bold lines). The thin black lines show the values of the constant gain.

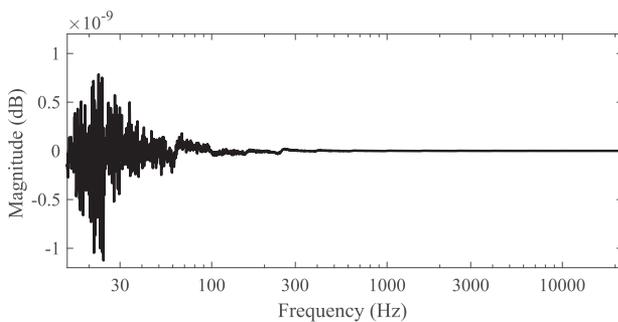


Fig. 5. Magnitude response difference of APGE and ACGE designs is very small, generally less than about 10^{-9} dB. This result was obtained using the ± 12 dB zigzag gain setting, cf. Fig. 4.

frequencies and, if two neighboring command gains are set to the same value, also between these neighboring frequency points.

A. Comparison of Series and Parallel Third-Octave Forms

In order to compare the APGE and the ACGE, the same target EQ curve is designed with both methods. A zigzag setting is used here, where the command gains alternate between ± 12 dB, as shown in Fig. 4. Fig. 4(a) shows the total response of the EQ as well as the responses of the subfilters for the series design, whereas Fig. 4(b) presents the corresponding curves for the delayed parallel design. The total responses produced by these two methods are almost identical, as confirmed by Fig. 5, which shows the difference between the magnitude responses of the two designs. Since the maximum difference resulting from the parallel conversion is on the order of 10^{-9} dB when 64-bit floating point numbers are used, the two filter forms can be stated to have the same magnitude response.

However, differences between the parallel and series form can be observed when the responses of the individual second-order sections, also shown in Fig. 4, are compared: When using the same zigzag command-gain setting, the filters in the series design require a maximum gain of 26 dB to produce the desired gain of 12 dB due to the interaction with the neighboring filters. The interaction is especially noticeable when two neighboring command gains have a large difference. In comparison, in the delayed parallel design, the maximum gain of an individual filter is approximately 14 dB.

B. Comparison With a Previous Parallel Octave Graphic Equalizer of the Same Order

In this section, the magnitude response of the octave ACGE is compared to that of a previous parallel EQ proposed by Chen *et al.* [26], which also comprises a single biquad section per band. Their EQ design utilizes a modified bilinear transform to compensate for the center-frequency shift at high frequencies and a pre-distortion of the quality factors to correct the bandwidths [26]. The filter gains are optimized in the sense of neighboring band leakage by using a gradient algorithm to solve a set of nonlinear equations.

Chen *et al.* give the parameters for two gain-setting cases in [26], which are used here to compare their design with the proposed method. The two cases are shown in Figs. 6 and 7: the first is a constant gain setting with all commands at 5 dB (see Fig. 6(a)), and the second contains amplifications and attenuations of ± 3 dB (see Fig. 6(b)). Both EQ designs by Chen *et al.* are accurate at the filter center frequencies, but the proposed method performs better overall having less undulation between these points, as shown in Figs. 7(a) and 7(b). In the first test case (Figs. 6(a) and 7(a)), the EQ by Chen *et al.* drops approximately 1.7 dB from the flat target, whereas the maximum deviation in the proposed method is approximately 0.24 dB. For the second case, the deviations for the two methods are 1.7 dB and 0.35 dB, respectively.

Hi-fi audio typically strives for a 1-dB accuracy for equalizers [11], [23]. Since, the parallel EQ design proposed by Chen *et al.* fails to achieve this accuracy between the command frequencies, it may be unsuitable for the most demanding hi-fi applications. Furthermore, to our best knowledge, no other previous parallel EQ using a single second-order section per band can achieve the desired accuracy of ± 1 dB. The state-of-the-art parallel EQ reaching the desired accuracy that will be used for further comparisons in the next section requires two second-order sections per band.

C. Comparison With a State-of-the-Art Parallel Graphic Equalizer

The state of the art in parallel EQs was proposed by Rämö *et al.* [17]. Two efficient variants of this parallel graphic EQ are proposed in [24], the weighted and the unweighted design. Both significantly reduce the design time compared to the original version in [17] by optimizing the target phase calculation, and the unweighted design additionally uses a pre-computed pseudoinverse at the expense of design accuracy. We compare

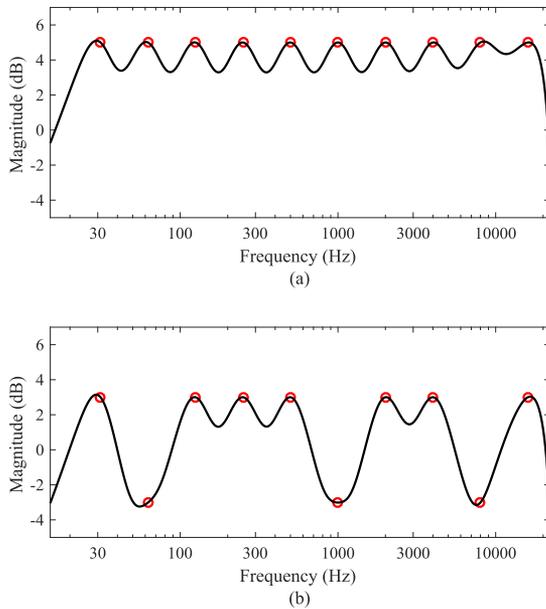


Fig. 6. Magnitude response of the octave EQ proposed by Chen *et al.* [26] with (a) the all-up and (b) the attenuation-amplification command gain setting from [26].

the response of our proposed parallel design with the more accurate weighted variant, since our goal is to fulfill the ± 1 -dB accuracy required for hi-fi applications.

The three test cases are shown in Figs. 8 and 9. The zigzag setting shown first exposes the EQ's ability to create steep transitions between ± 12 dB. Both EQ designs are able to reach the command gains, as shown in Figs. 9(a) and 8(a), producing similar responses everywhere except at very low and high frequencies. Both responses also achieve the desired 1-dB accuracy at the command points. The maximum estimation errors are shown in Table II, and the proposed method performs slightly better with approximately 0.3-dB smaller error.

The second test concerns a known challenging case [25], which in addition to the alternating ± 12 dB command gains contains flat regions, i.e., two or more neighboring commands are at the same position, as shown in Figs. 9(b) and 8(b). In this case, the EQ response must reach the command values at the center frequencies and also stay within the desired 1-dB limit between the command points having the same gain, i.e., in the flat regions. Again, the two responses are almost identical between 20 Hz and 20 kHz, and both achieve the desired accuracy. In this case, the PGE is slightly more accurate having an approximately 0.3-dB smaller maximum error.

The third test case, shown in Figs. 9(c) and 8(c), is a non-extreme setting having multiple types of transitions between the command points. Both EQs produce acceptable results, but the APGE response contains less ripple than that of PGE. In addition, the maximum error is approximately 0.2 dB smaller for the APGE. Thus, the PGE and the APGE designs both produce sufficiently good parallel graphic EQ filters for demanding audio applications.

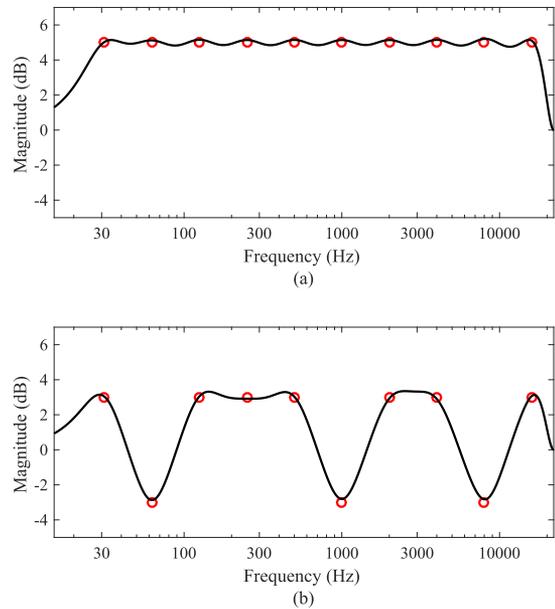


Fig. 7. Magnitude response of the proposed octave EQ design with (a) the all-up and (b) the attenuation-amplification command gain setting from [26].

TABLE II
MAXIMUM ERROR OF TWO THIRD-OCTAVE GRAPHIC EQS IN THREE TEST TARGET SETTINGS. THE BEST RESULT IN EACH CASE IS HIGHLIGHTED

Case	PGE [24]	APGE (proposed)
Zigzag	0.67 dB	0.41 dB
Hard case	0.73 dB	0.98 dB
Non-extreme	0.67 dB	0.52 dB

TABLE III
AVERAGE COMMAND GAIN UPDATE TIMES IN MATLAB

ACGE	PGE	APGE (proposed)
0.52 ms	0.39 ms	0.82 ms

D. Comparison of Computational Performance

Finally, the performance of the APGE is compared to the original series design (ACGE) and the state-of-the-art parallel EQ (PGE) in terms of design time and the computational load of the filtering operation.

Table III lists for each method the average design time, i.e., the gain update time, in MATLAB. Each of these times is an average of 1,000 test runs, where the EQs were designed with random gains between ± 12 dB. The update times for each EQ design have a similar order of magnitude with each being suitable for applications where the gains are frequently updated automatically, such as digital audio workstations. The efficient parallel EQ design by Bank *et al.* [24] is the fastest taking 0.39 ms, and the proposed method is the slowest taking approximately double the design time of the PGE. The series design method ACGE is also faster when compared to the parallel design in terms of gain update time, which is logical, since the proposed parallel design comprises the ACGE design steps followed by additional processing. The series-to-parallel conversion using the PFE method proposed in this paper amounts an increase in

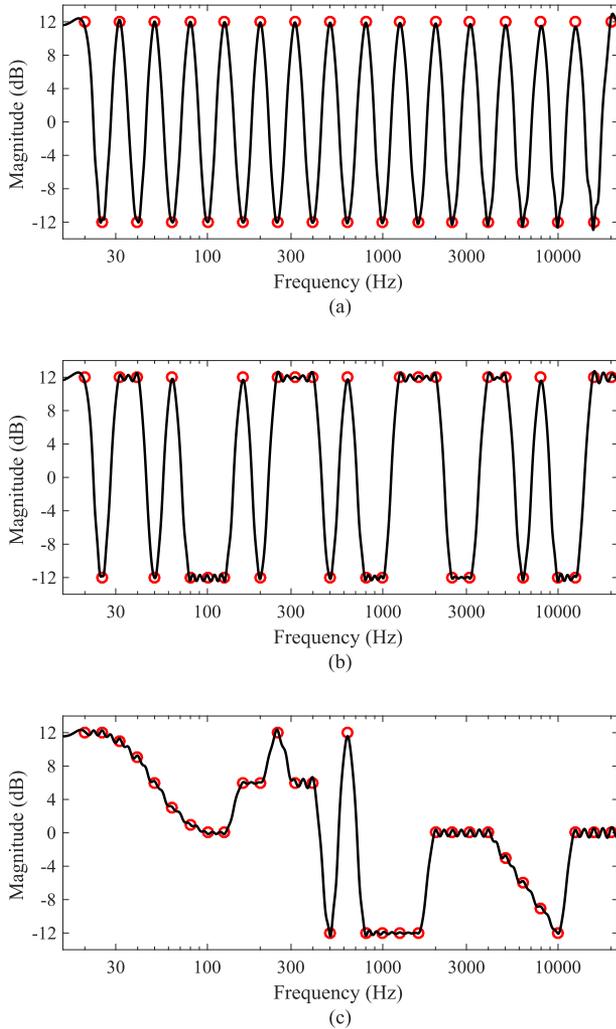


Fig. 8. Magnitude response of the PGE with (a) zigzag, (b) hard, and (c) nonextreme command gain settings.

TABLE IV

NUMBER OF OPERATIONS PER OUTPUT SAMPLE IN THE SERIES AND IN TWO PARALLEL GRAPHIC EQUALIZER STRUCTURES. THE SMALLEST NUMBERS ON EACH ROW ARE HIGHLIGHTED IN BOLD

Operations	ACGE (series)	PGE	APGE (proposed)
Additions	124	248	124
Multiplications	125	248	125
Total	249	496	249

design time of approximately 58% when compared to the ACGE series design.

Note that non-weighted variant of PGE [24] is even more efficient (the average computation time is 0.07 ms), but produces a slightly larger error at the command points (1.1 dB for the non-extreme case) and more ripples between the command points that do not meet the ± 1 -dB accuracy requirement.

Table IV lists for each method the number of operations necessary during real-time filtering. The number of operations for the series EQ can be calculated with the help of Fig. 1(b). A single section performs four additions and four multiplications, and since there are 31 such sections and one more multiplication

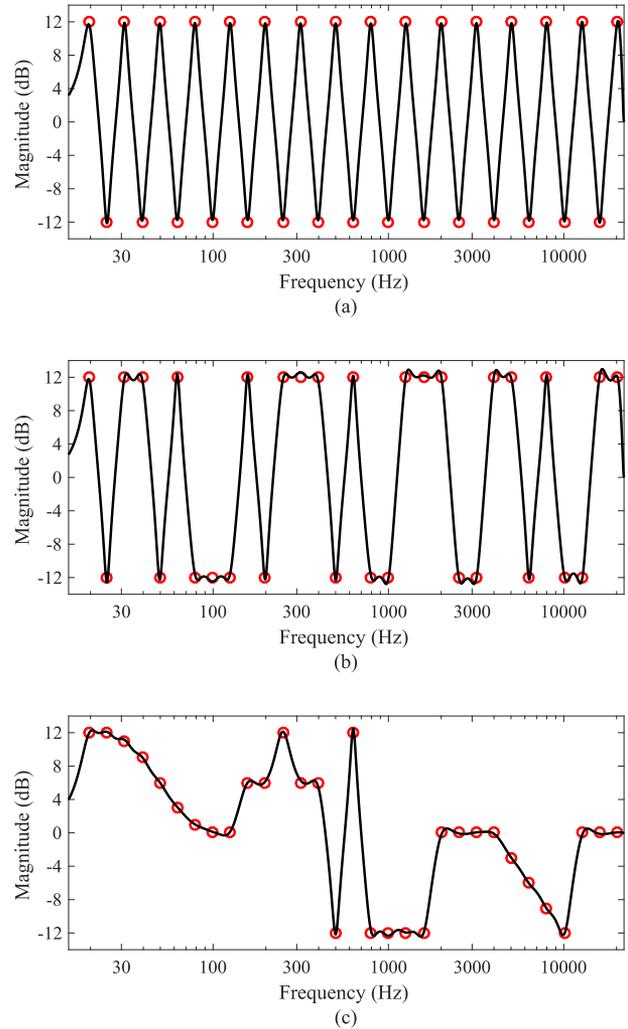


Fig. 9. Magnitude response of the proposed EQ design with (a) zigzag, (b) hard, and (c) non-extreme command gain settings.

(G_0), as seen in Fig. 1(a), the total is 124 additions and 125 multiplications. For the delayed parallel EQ, similar calculations can be made with the help of Fig. 2(b). Here, a single section has three additions and four multiplications, but there are also 31 additions and one multiplication in the overall structure shown in Fig. 2(a). Thus, 31 sections result in 124 additions and 125 multiplications, which is the same as for the series structure.

The structure of the PGE is similar to that in Fig. 2 with the exception that the unit delay is missing in the PGE structure and there is no direct path gain (i.e., $F = 0$). However, it requires twice as many filters as there are bands, and thus, it comprises 62 second-order sections, resulting in 248 additions and 248 multiplications, which are listed in Table IV.

Table IV shows that the proposed converted delayed parallel filter APGE outperforms the PGE structure in computational efficiency, since it uses 50% less operations per sample. The proposed parallel structure has just one second-order section per frequency band, whereas the PGE structure requires two filters for each band and so uses 62 second-order filters in the 31-band

graphic equalizer [17]. The proposed APGE now becomes the parallel graphic equalizer with the lowest order fulfilling the ± 1 dB magnitude tolerance.

VI. CONCLUSION

This paper has introduced a novel method for converting series IIR filters to the delayed parallel form without sacrificing the accuracy of approximation. The method was inspired by the Orfanidis PFE method, which can be implemented without long division of polynomials. The proposed PFE method leads to a delayed parallel IIR filter form, which, in addition to all the desirable properties of parallel digital filter systems, enjoys a smaller dynamic range requirement than the traditional non-delayed case. In addition, Orfanidis's idea has been extended for the case where the order of the numerator is larger than that of the denominator.

A case study on graphic EQs showed that when an accurate series design is converted to the delayed parallel form having a unit delay in front of the biquad sections, the resulting filter is superior to previous parallel equalizer designs: its accuracy is superior, its computational load during filtering is smaller, and it has the design time of the same magnitude when compared to the previous state-of-the-art parallel EQ. In addition, when compared to the series design, the subfilters of the parallel version have smaller maximum gain and the conversion requires only a design overhead of 58% on top of the original series design.

To the best of our knowledge, the proposed APGE filter is the first parallel graphic EQ using a single second-order filter per band to fulfill the standard ± 1 -dB hi-fi requirement. The proposed conversion technique is widely applicable to the design of parallel IIR filters, which are becoming popular because they are well suited to implementation in parallel computers. The relevant MATLAB code is available online [27].

ACKNOWLEDGMENT

The authors would like to thank J. Rämö for helpful comments and Z. Chen and G. S. Geng for e-mail discussions.

REFERENCES

- [1] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*, 3rd ed. Upper Saddle River, NJ, USA: Pearson Higher Education, 2010.
- [2] S. J. Orfanidis, *Introduction to Signal Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.
- [3] J. O. Smith, *Introduction to Digital Filters with Audio Applications*. W3K Publishing, 2008. [Online]. Available: <http://books.w3k.org/>
- [4] B. Bank, "Converting infinite impulse response filters to parallel form," *IEEE Signal Process. Mag.*, vol. 35, no. 3, pp. 124–130, May 2018.
- [5] W. Chen, "Performance of cascade and parallel IIR filters," *J. Audio Eng. Soc.*, vol. 44, no. 3, pp. 148–158, Mar. 1996.
- [6] J. A. Belloch, B. Bank, L. Savioja, A. Gonzalez, and V. Välimäki, "Multi-channel IIR filtering of audio signals using a GPU," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, Florence, Italy, May 2014, pp. 6692–6696.
- [7] F. Keiler and U. Zölzer, "Parametric second- and fourth-order shelving filters for audio applications," in *Proc. IEEE 6th Workshop Multimedia Signal Process.*, Siena, Italy, Sep. 2004, pp. 231–234.
- [8] J. S. Abel and D. P. Berners, "Filter design using second-order peaking and shelving sections," in *Proc. Int. Comput. Music Conf.*, Miami, FL, USA, Nov. 2004.

- [9] G. Ramos and J. J. López, "Filter design method for loudspeaker equalization based on IIR parametric filters," *J. Audio Eng. Soc.*, vol. 54, no. 12, pp. 1162–1178, Dec. 2006.
- [10] R. J. Oliver and J.-M. Jot, "Efficient multi-band digital audio graphic equalizer with accurate frequency response control," in *Proc. Audio Eng. Soc. 139th Conv.*, New York, NY, USA, Oct. 2015.
- [11] V. Välimäki and J. Liski, "Accurate cascade graphic equalizer," *IEEE Signal Process. Lett.*, vol. 24, no. 2, pp. 176–180, Feb. 2017.
- [12] T. W. Parks and C. S. Burrus, *Digital Filter Design*. New York, NY, USA: Wiley, 1987.
- [13] L. B. Jackson, *Digital Filters and Signal Processing*, 3rd ed. Norwell, MA, USA: Kluwer, 1996.
- [14] T. J. Cavicchi, *Digital Signal Processing*, 1st ed. New York, NY, USA: Wiley, 2000.
- [15] B. Bank, "Perceptually motivated audio equalization using fixed-pole parallel second-order filters," *IEEE Signal Process. Lett.*, vol. 15, pp. 477–480, 2008.
- [16] B. Bank, "Direct design of parallel second-order filters for instrument body modeling," in *Proc. Int. Comput. Music Conf.*, Copenhagen, Denmark, Aug. 2007, pp. 458–465.
- [17] J. Rämö, V. Välimäki, and B. Bank, "High-precision parallel graphic equalizer," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 22, no. 12, pp. 1894–1904, Dec. 2014.
- [18] N. Wong and C.-U. Lei, "IIR approximation of FIR filters via discrete-time vector fitting," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1296–1302, Mar. 2008.
- [19] H. Qi, Z. G. Feng, K. F. C. Yiu, and S. Nordholm, "Optimal design of IIR filters via the partial fraction decomposition method," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, to be published.
- [20] B. Bank and J. O. Smith, "A delayed parallel filter structure with an FIR part having improved numerical properties," in *Proc. Audio Eng. Soc. 136th Conv.*, Berlin, Germany, Apr. 2014.
- [21] G. B. Thomas, Jr. and R. L. Finney, *Calculus and Analytic Geometry*, 9th ed. Reading, MA, USA: Addison-Wesley, 1998.
- [22] V. Välimäki and J. D. Reiss, "All about audio equalization: Solutions and frontiers," *Appl. Sci.*, vol. 6, no. 129/5, pp. 1–46, May 2016.
- [23] M. Holters and U. Zölzer, "Graphic equalizer design using higher-order recursive filters," in *Proc. Int. Conf. Digital Audio Effects*, Montreal, QC, Canada, Sep. 2006, pp. 37–40.
- [24] B. Bank, J. A. Belloch, and V. Välimäki, "Efficient design of a parallel graphic equalizer," *J. Audio Eng. Soc.*, vol. 65, no. 10, pp. 817–825, Oct. 2017.
- [25] J. Liski and V. Välimäki, "The quest for the best graphic equalizer," in *Proc. Int. Conf. Digital Audio Effects*, Edinburgh, U.K., Sep. 2017, pp. 95–102.
- [26] Z. Chen, G. S. Geng, F. L. Yin, and J. Hao, "A pre-distortion based design method for digital audio graphic equalizer," *Digital Signal Process.*, vol. 25, pp. 296–302, Feb. 2014.
- [27] J. Liski, B. Bank, J. O. Smith, and V. Välimäki, "Companion page to Converting cascaded biquad filters into delayed parallel form: Application to graphic equalizers," 2019. [Online]. Available: <http://research.spa.aalto.fi/publications/papers/ieee-tsp-2019-biquad/>. Accessed: May 7, 2019.



Juho Liski was born in Helsinki, Finland, in 1989. He received the M.Sc. in Technology degree in electrical engineering in 2016 from the Aalto University School of Electrical Engineering, Espoo, Finland, where he is currently working toward the Doctoral degree, focusing on audio signal processing and equalization.

Since 2015, he has been with the Acoustics Laboratory, Aalto University School of Electrical Engineering.

Mr. Liski is a student member of the Audio Engineering Society and a member of the Acoustical Society of Finland.



Balázs Bank (M'12) received the M.Sc. and Ph.D. degrees in electrical engineering from the Budapest University of Technology and Economics, Budapest, Hungary, in 2000 and 2006, respectively. His doctoral dissertation was about physics-based synthesis of string instrument sounds with an emphasis on the piano.

In the academic year 1999–2000, he was with the Laboratory of Acoustics and Audio Signal Processing, Helsinki University of Technology, Finland. From 2000 to 2006, he was a Ph.D. student and a

Research Assistant with the Department of Measurement and Information Systems, Budapest University of Technology and Economics. In 2001, he visited the Department of Information Engineering, University of Padova. In 2007, he returned to the Acoustics Laboratory, Helsinki University of Technology, for a year, with the support of an FP6 Marie Curie EIF individual fellowship. In 2008, he was with the Department of Computer Science, Verona University, Italy. Since 2009, he has been with the Department of Measurement and Information Systems, Budapest University of Technology and Economics, where he is currently an Associate Professor. His research interests include physics-based sound synthesis and filter design for audio applications.

Prof. Bank is an AES Member. From 2013 to 2016, he was an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS, and is currently an Associate Editor for the IEEE SIGNAL PROCESSING MAGAZINE.



Julius O. Smith (M'76) received the B.S.E.E. degree in control, circuits, and communication from Rice University, Houston, TX, USA, in 1975, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA, in 1978 and 1983, respectively. His Ph.D. research was devoted to improved methods for digital filter design and system identification applied to music and audio systems.

From 1975 to 1977, he was with the Signal Processing Department, ESL, Sunnyvale, CA, USA, on systems for digital communications. From 1982 to

1986, he was with the Adaptive Systems Department, Systems Control Technology, Palo Alto, CA, USA, where he worked in the areas of adaptive filtering and spectral estimation. From 1986 to 1991, he was with NeXT Computer, Inc., responsible for sound, music, and signal processing software for the NeXT computer workstation. After NeXT, he was an Associate Professor at the Center for Computer Research in Music and Acoustics, Stanford University, Stanford, CA, USA, teaching courses and pursuing research related to signal processing techniques applied to music and audio systems. Continuing this work, he is currently a Professor of Music and (by courtesy) Electrical Engineering with Stanford University.

Prof. Smith is a Fellow of the Audio Engineering Society and the Acoustical Society of America. For more information, see <http://ccrma.stanford.edu/~jos/>.



Vesa Välimäki (S'90–M'92–SM'99–F'15) received the M.Sc. in Technology and the Doctor of Science in Technology degrees in electrical engineering from the Helsinki University of Technology (TKK), Espoo, Finland, in 1992 and 1995, respectively.

He was a Postdoctoral Research Fellow with the University of Westminster, London, U.K., in 1996. In 1997–2001, he was a Senior Assistant (cf. Assistant Professor) with TKK. In 2001–2002, he was a Professor of Signal Processing with the Tampere University of Technology, Pori, Finland. In 2006–2007, he was

the Head of the TKK Laboratory of Acoustics and Audio Signal Processing. In 2008–2009, he was a Visiting Scholar with Stanford University. He is currently a Professor of Audio Signal Processing and the Vice Dean for Research in Electrical Engineering with Aalto University, Espoo, Finland. His research interests include digital filter design, audio effects processing, artificial reverberation, sound synthesis, and signal processing for headphones and loudspeakers.

Prof. Välimäki is a Fellow of the Audio Engineering Society and a Life Member of the Acoustical Society of Finland. In 2007–2013, he was a member of the Audio and Acoustic Signal Processing Technical Committee of the IEEE Signal Processing Society and is currently an associate member. He is a Founding Member of the EURASIP Special Area Team in acoustic, sound and music signal processing. He was an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS in 2005–2009 and IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, in 2007–2011. He was in the Editorial Board of the *Research Letters in Signal Processing* and *Journal of Electrical and Computer Engineering*. He was the Lead Guest Editor of a Special Issue of the IEEE SIGNAL PROCESSING MAGAZINE in 2007 and a Special Issue of the IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING in 2010. He was a Guest Editor for the special issues of the IEEE SIGNAL PROCESSING MAGAZINE on signal processing techniques for assisted listening in 2015 and on music signal processing in 2018–2019. He has been a Guest Editor for two special issues published in *Applied Sciences*, one in 2016 and another in 2017–2018. Since 2015, he has been a Senior Area Editor of the IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING. In 2008, he was the Chair of the 11th International Conference on Digital Audio Effects DAFX-08. In 2017, he was the Chair of the International Conference on Sound and Music Computing.