OPEN ACCESS Freely available online
B. Bank, "Warped, Kautz, and
Fixed-Pole Parallel Filters: A Review"
J. Audio Eng. Soc., vol. 70, no. 6, pp. 414–434, (2022 June).
DOI: https://doi.org/10.17743/jaes.2022.0016

Warped, Kautz, and Fixed-Pole Parallel Filters: A Review

BALÁZS BANK, *AES Associate Member*
(bank@mit.bme.hu)

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

In audio signal processing, the aim is the best possible sound quality for a given computational complexity. For this, taking into account the logarithmic frequency resolution of hearing is a good starting point. The present paper provides an overview on warped, Kautz, and fixed-pole parallel filters and demonstrates that they are all capable of achieving logarithmic-like frequency resolution, providing much more efficient filtering or equalization compared to straightforward finite impulse response (FIR) or infinite impulse response (IIR) filters. Besides presenting the historical development of the three methods, the paper discusses their relations and provides a comparison in terms of accuracy, computational requirements, and design complexity. The comparison includes loudspeaker–room response modeling and equalization examples.

0 INTRODUCTION

Modeling or equalizing a given transfer function is one of the most often used applications of digital filters in the field of audio. A typical example is to correct the non-ideal frequency response of loudspeakers, ranging from speakers in mobile devices, computer speakers, and car audio to large-scale public address systems [1, 2]. Often the transfer function of the room is also equalized together with the loudspeaker [3–6]. Other applications include the modeling of the same systems for simulation purposes [7, 8], modeling and equalization of headphones [9], modeling of head-related transfer functions for 3D audio [10–12], or the synthesis of musical instrument sounds [13–15], to name a few. For all these applications, a digital filter that achieves the best sound quality at a given computational cost has to be designed.

As in audio, the final judge of quality is the human ear; it seems logical to take into account some of the properties of the auditory system during filter design. One such property that has been used since the early times of audio is the logarithmic-like frequency resolution of hearing. For example, graphic equalizers that are used to manually tune the response of an audio system or change the timbre of sound recordings have bands with center frequencies evenly distributed in the logarithmic scale.

Moreover, audio transfer functions are almost always displayed in a logarithmic frequency scale and they are often smoothed to some fractional octave resolution. Note that traditionally smoothing has been applied to magnitude responses only, but with digital processing it has become possible to smooth the complex frequency responses, with the advantage that a smoothed impulse response can be reconstructed from the smoothed transfer function [16, 3].

It is now a common practice to design loudspeaker and room equalizers based on the logarithmically smoothed response instead of the original. This is not only motivated by perceptual principles, but also coming from the fact the fine details of the original response are highly position-dependent. Therefore, equalizing this overly detailed response for one measurement point usually worsens the response at other listening positions, an artifact greatly reduced by proper smoothing prior to filter design [17, 3, 6].

As a result, when designing a digital filter for audio purposes, it seems logical to design the filter so that the error is distributed evenly in the logarithmic scale, and indeed, this is the approach most often taken in the literature when designing filters for audio applications.

Note that various auditory frequency scales exist, like the Bark, mel, or Equivalent Rectangular Bandwidth (ERB) [18, 19] scales (which are all actually quite close to the

logarithmic scale) that are sometimes used for audio applications. Since in audio the logarithmic scale is by far the most commonly used, the examples of this paper will use the logarithmic scale. Nevertheless, all the discussed methods can be parameterized so that they allow filter design on frequency scales other than logarithmic.

0.1 Limitations of General Purpose Filter Design Techniques

Note that the term *filter design* is often used in a strict sense for designing low-pass, high-pass, band-pass, or band-reject filters with a certain passband ripple and stopband attenuation [20, 21]. In this work, however, the term *filter design* is used as a general method for approximating an arbitrary impulse response or frequency response.

Modeling or equalizing a transfer function by a finite impulse response (FIR) or infinite impulse response (IIR) filter is a common task in many fields of digital signal processing. Therefore, a wide range of filter design techniques exist that at first glance should be also appropriate for audio applications. However, these general filter design methods have an important property that makes them less than optimal for audio. Namely, they have linear frequency resolution, meaning that the error of the filter is distributed evenly in the linear frequency scale. Linear frequency resolution is inherent in FIR filters, since their frequency response is given as the discrete Fourier transform (DFT) of their impulse response, leading to a transfer function vector with linearly spaced frequency bins. Thus, the frequency resolution Δf is directly determined by the length of the filter N and sampling frequency f_s, given as the resolution of the DFT: $\Delta f = f_s/N$ [note that smaller $\Delta(f)$ means higher resolution].

Also, many IIR filter design algorithms (e.g., Prony [20] and Steiglitz-McBride [22]) minimize the error between the target impulse response and filter response in the mean squared sense, which, by Parseval's theorem, is equivalent to minimizing the mean squared error between the target and filter frequency response in a linear frequency scale. Some frequency-domain IIR filter design methods allow the use of a weighting function (e.g., the frequency-domain Steiglitz-McBride algorithm [23] or `invfreqz` function in MATLAB [24]), or their target frequency scale can be made logarithmic instead of linear. So in theory, it should be feasible to achieve a logarithmic frequency resolution by these methods. However, the logarithmic scale is so distorted compared to its linear counterpart that this is not a viable solution in practice, as has also been noted in [25].

The following examples will demonstrate the difficulties of achieving logarithmic frequency resolution with general IIR filter design methods. The example case is modeling a loudspeaker–room response. In Fig. 1(a) a 100th-order IIR filter is designed by the frequency-domain Steiglitz-McBride algorithm [23]. The target is a measured loudspeaker–room response. The target is made minimum-phase to ease filter design; this is a common practice in audio because by giving up the perceptually less important phase specification, better magnitude match is achieved when designing IIR filters. The target points are linearly distributed in frequency: 10,000 points from 0 Hz to half of the sample rate $f_s/2 = 22{,}050$ Hz. Note that the sample rate $f_s = 44{,}100$ Hz is one of the most commonly used in audio, and this sample rate will be used for all the examples of this work. Because of the linear distribution of target frequency points, it is expected that the error will be linearly distributed in frequency. This is indeed visible in the left column of Fig. 1(a), where the filter response (thick solid line) follows the target (thin gray line) by the same accuracy for all frequencies, which is also indicated by the fact that the frequencies of the filter poles are evenly distributed in the linear frequency scale [see the crosses in the left side of Fig. 1(a)]. When the same curves are plotted in a logarithmic scale in the right column of Fig. 1(a), a different picture can be seen, and the lack of modeling ability at low frequencies is immediately apparent.

The next step is to apply frequency weighting. In the example of Fig. 1(b), a weighting of $W(f) = 1/f^2$ is used; otherwise, the design is the same as for Fig. 1(a). (Note that $W(f)$ is limited at $f = 20$ Hz to avoid the very large weights at the otherwise irrelevant infrasonic frequencies.) It can be seen in the logarithmic scale Fig. 1(b) (right column) that the fit is only slightly improved at low frequencies, which is quite surprising because the lowest frequencies get 10^6 times larger weight in the error compared to the highest ones.

As a next trial, the filter design is based on a target response whose frequency points are logarithmically spaced. In Fig. 1(c), the target (thin gray line) was resampled to a logarithmic scale with 100 bins per octave from 20 Hz to $f_s/2 = 22{,}050$ Hz, giving 1,011 specification points. No weighting is used, because the logarithmic resampling of the frequency grid already ensures that the error is minimized in the logarithmic scale. Perhaps somewhat surprisingly, as shown by the practically unchanged low frequencies in Fig. 1(c) compared with Fig. 1(a), the inaccuracy in the low frequencies persists, and the design shows no visible improvement.

Finally, the logarithmic spacing of the target frequency points is kept, but logarithmic smoothing is also applied, which, in practice, removes the details at high frequencies in favor of placing more effort on the low frequencies, which are given more importance on a logarithmic scale. In Fig. 1(d), a sixth-octave smoothing has been used, and it is apparent in the right column of Fig. 1(d) that the resolution of the target (thin gray line) is logarithmic, indeed. Although the low-frequency modeling is improved somewhat, the results are still disappointing below 500 Hz.

From the examples it appears that although in theory both the weighted design and the design based on a logarithmically spaced specification should result in a filter with logarithmic frequency resolution—not to mention the design based on a logarithmically smoothed target—this is not the outcome in practice. The examples used only the frequency-domain Steiglitz-McBride algorithm [23], but similar results may be observed with other methods, like `invfreqz` in MATLAB. This complies with the findings of [25] in that traditional IIR design techniques us-

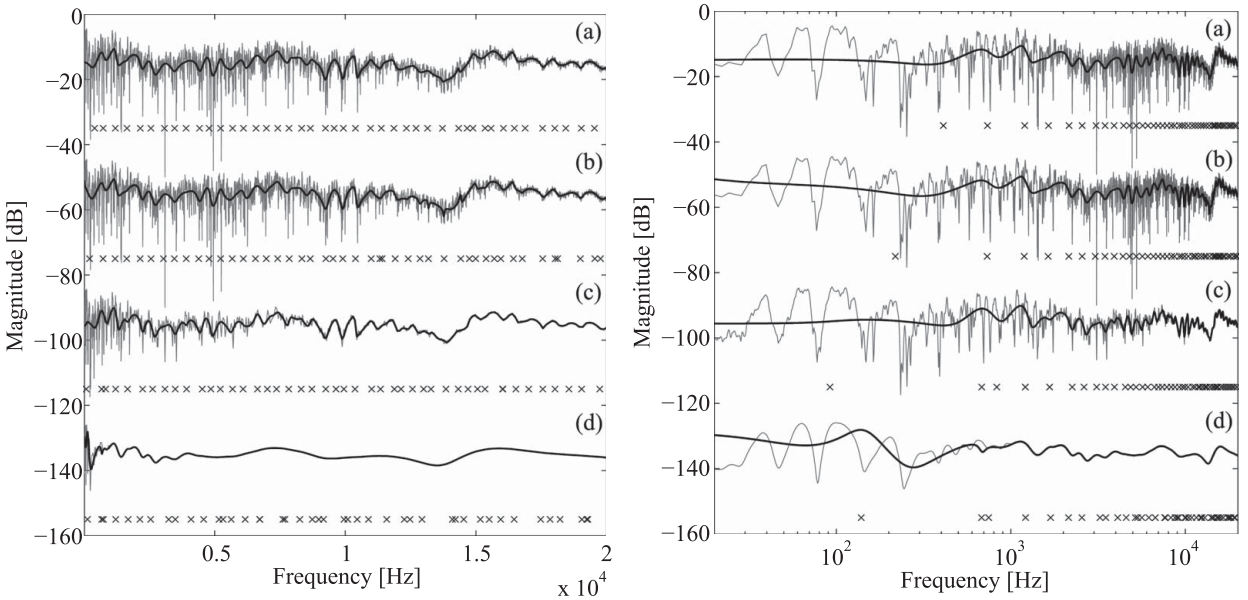

Fig. 1. Different variations of 100th-order infinite impulse response (IIR) filter design using the frequency-domain Steiglitz-McBride algorithm. The minimum-phase loudspeaker–room response is displayed by thin gray lines, and the modeled responses by thick solid lines. The versions include designs based on linearly spaced target frequency points (a) without weighting and (b) with $1/f^2$ weighting, (c) a design using logarithmically spaced target frequency points, and (d) using a logarithmically smoothed target response. The same responses are plotted in linear (left column) and logarithmic (right column) frequency scale. The crosses below the corresponding magnitude responses indicate the pole frequencies of the filters. The curves are offset for clarity.

ing weighted least-squares minimization may fail to converge when the target frequency points are logarithmically spaced.

The reasons for this are most probably of numerical nature. Logarithmic frequency resolution would require a very high pole density at low frequencies with poles near the unit circle, and this cannot be implemented by direct form IIR filters even at double precision. Traditional IIR filter design algorithms estimate the parameters of a direct form IIR filter; they thus cannot give such a set of coefficients that would lead to the desired high pole density at low frequencies, because such a set of coefficients does not exist in the space of available numbers because of finite precision.

0.2 Specialized Filter Structures for Achieving Logarithmic Frequency Resolution

Recognizing these limitations, specialized IIR filter design techniques that allow a more flexible distribution of modeling detail as a function of frequency have been developed. It is interesting to note that all these methods use special filter structures instead of the direct (rational) form used for general purpose IIR filters, and the filters are designed directly in these special forms. This work will focus on warped [26], Kautz [27], and fixed-pole parallel filters [28].

It has to be mentioned that another way to address the problem of logarithmic frequency resolution comes from manually tuned graphic or parametric equalizers that exist from the early times of audio. Automatic techniques have been developed that iteratively tune the parameters of parametric equalizers by a nonlinear optimization process (see, e.g., [2, 29, 30]). This approach is advantageous when there is a need to manually fine-tune the filter response by listening, since the parameters of the equalization filters are perceptually meaningful and well understood by sound engineers.

Also, in automatically tuned parametric equalizers interpolation between different filter settings is relatively easily achieved. A drawback is that the special form of the filter sections reduces the degrees of freedom (center frequency, Q-factor, and gain instead of the four coefficients of a general second-order section) and thus may result in lower accuracy for the same filter order (see [31] and Fig. 3 for a comparison example). Additionally, these techniques allow magnitude equalization only; while warped, Kautz or parallel filters are also able to model or equalize the phase behavior if desired. Therefore, the automatic tuning of parametric equalizers will not be further discussed here. For an overview, the reader is referred to [30].

1 WARPING

The earliest perceptually motivated design technique is based on frequency warping. The idea can be traced back to the paper of Constantinides [32], which proposes the substitution of unit delays in digital filters with all-pass filters in order to change the filter type. Such a transformation can be low-pass–low-pass (change in cut-off frequency), low-pass–high-pass, low-pass–band-pass, or low-pass–band-reject, similarly to the spectral transformations used in the design of analog filters.

The first application of the all-pass transform for achieving nonlinear frequency resolution was proposed by Oppenheim et al. [33], where a non-uniform DFT was obtained

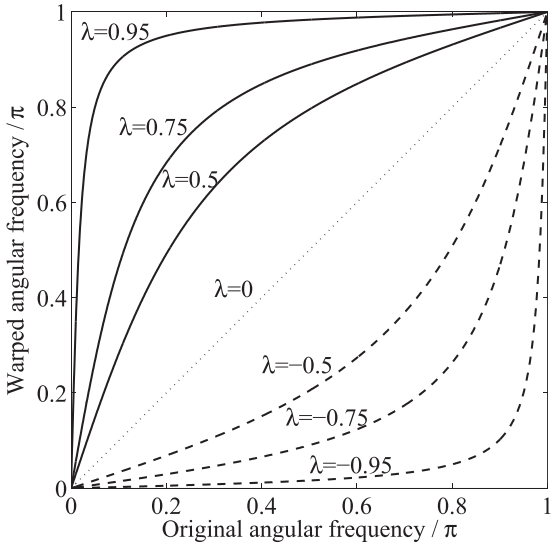

Fig. 2. The frequency mapping function $\nu(\vartheta)$ of Eq. (2) for various warping parameters.

by passing the input signal through an all-pass chain and using the outputs of the stages as the input of an ordinary FFT operation. Strube [34] has applied frequency warping for approximating the frequency resolution of the human auditory system in linear predictive coding of speech.

The use of frequency warping as a means of approximating logarithmic frequency resolution for IIR filters was proposed in [24, 25], and an extensive overview on the subject was presented in [26].

1.1 The Effect of Warping

The basic idea of warped filters is that the unit delay z^{-1} of traditional FIR or IIR filters is replaced by an all-pass filter:

$$z^{-1} \leftarrow D(z) = \frac{z^{-1} - \lambda}{1 - \lambda z^{-1}}. \tag{1}$$

The transformation of the frequency axis is equal to the phase response of the first-order all-pass filter, resulting in the frequency mapping

$$\tilde{\vartheta} = \nu(\vartheta) = \arctan \frac{(1 - \lambda^2)\sin(\vartheta)}{(1 + \lambda^2)\cos(\vartheta) - 2\lambda}, \tag{2}$$

where ϑ is the original and $\tilde{\vartheta}$ is the warped angular frequency in radians [26]. This transformation is displayed for various λ values in Fig. 2.

Accordingly, a filter originally having the transfer function of $H(\vartheta)$ will have the transfer function of $H(\nu(\vartheta))$ after substituting its delay elements by the first-order all-pass filter of Eq. (1). For increasing the resolution at low frequencies, which is required for achieving a logarithmic scale, positive λ values are used. It can be seen in Fig. 2 that in this case, the region around zero frequency will span a much larger frequency region in the warped domain, which means increased resolution.

The change of the frequency resolution is related to the slope of the mapping function in Fig. 2, that is, the derivative

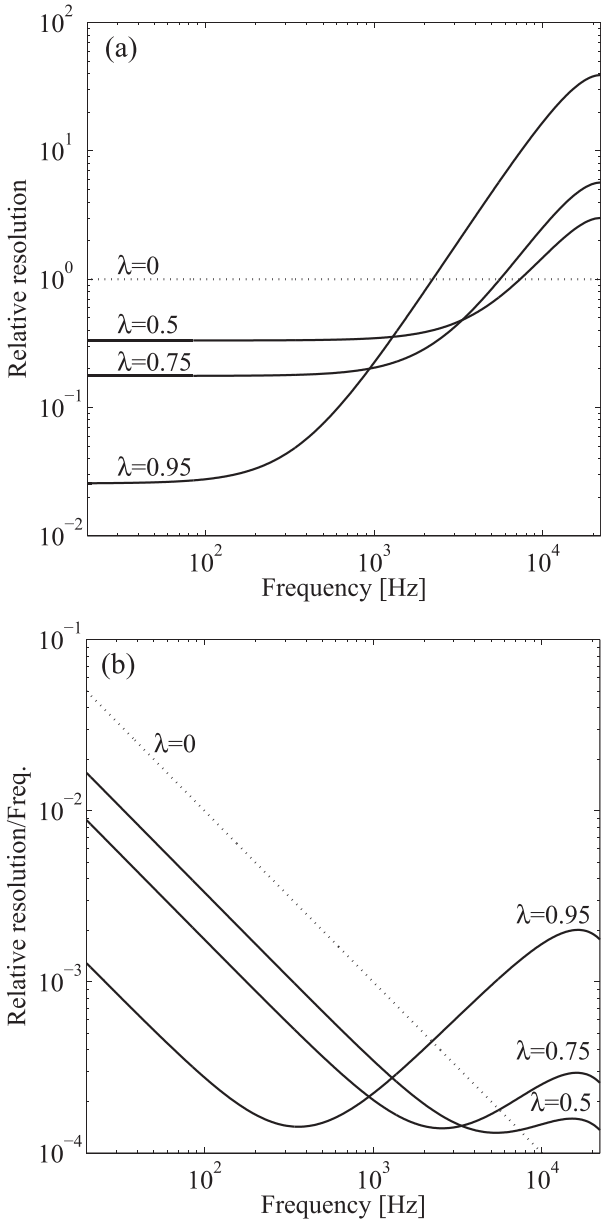

Fig. 3. The change of frequency resolution due to warping: (a) $\Delta\tilde{f}$ according to Eq. (3) for various warping parameters and (b) "logarithmic resolution" $\Delta\tilde{f}/f$. The resolution prior to warping is $\Delta f = 1$ in both figures. Smaller values correspond to a higher resolution.

of Eq. (2). The steeper the slope, the larger the increase in frequency resolution is. Accordingly, if an FIR or IIR filter has a local resolution described by $\Delta f(f)$, then its warped variant will have the resolution characterized by

$$\Delta\tilde{f}(f) = \frac{1 + \lambda^2 - 2\lambda\cos(\vartheta)}{1 - \lambda^2}\Delta f, \tag{3}$$

where $\vartheta = 2\pi f/f_s$ [35]. For FIR filters, $\Delta f = f_s/N$, where N is the filter order. For IIR filters, such an exact number cannot be computed; nevertheless, Eq. (3) still shows how the resolution is mapped when the IIR filter is implemented using all-pass filters instead of unit delays.

The relative resolution is shown in Fig. 3(a) for various

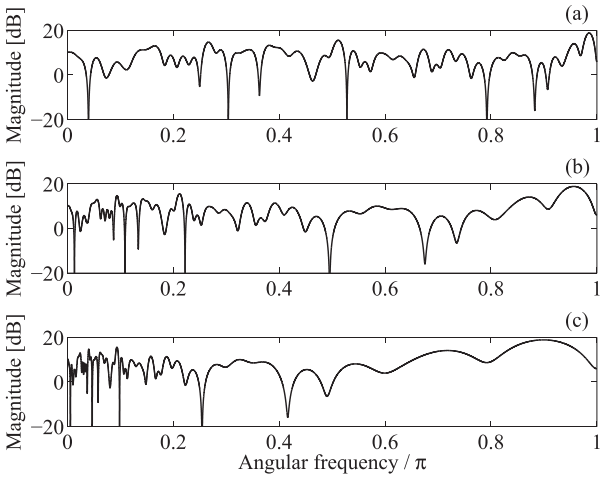

Fig. 4. The frequency response of a 100th-order finite impulse response filter with random coefficients (a) and the same filter having its unit delays replaced by all-pass filters of the form Eq. (1) with (b) $\lambda = 0.5$ and (c) $\lambda = 0.75$.

λ values, and the dotted line $\lambda = 0$ corresponds to no warping. The plot was computed using Eq. (3) with $\Delta f = 1$ and $f_s = 44.1$ kHz. It can be seen in Fig. 3(a) that the warped implementations with $\lambda > 0$ increase the resolution (decrease $\Delta\tilde{f}$) at low frequencies, at the expense of lower resolution (larger $\Delta\tilde{f}$) at high frequencies. This tradeoff is understandable because the degrees of freedom in the filter are unchanged; thus, an improvement at a specific frequency band will lead to poorer performance at another band.

Since the aim is to approximate logarithmic frequency resolution, it makes sense to plot the same curves divided by frequency. Accordingly, in Fig. 3(b), $\Delta\tilde{f}(f)/f$ is plotted for $\Delta f = 1$. Again, the dotted line with $\lambda = 0$ shows what happens with an ordinary (not warped) FIR or IIR filter, and a logarithmic frequency resolution would correspond to a horizontal line in the figure. It can be seen that although none of the λ values achieve fully logarithmic resolution (none of the curves are horizontal lines), there are some frequency bands for each λ where this is relatively well achieved.

The relation of warping to psychoacoustic scales (Equivalent Rectangular Bandwidth [ERB], Bark, and Greenwood) is discussed in [26]. Additionally, [18] has given an analytical expression for λ as a function of sampling rate to match the Bark scale. However, similarly to the logarithmic scale, an exact match is not possible because of the limited degrees of freedom (a single λ parameter).

The warping effect is demonstrated in Fig. 4, where Fig. 4(a) displays the original frequency response of an arbitrary FIR filter having random coefficients and Figs. 4(b) and 4(c) display the filter response when the unit delays are exchanged for first-order all-pass filters with $\lambda = 0.5$ and $\lambda = 0.75$, respectively. It can be seen in Fig. 4(a) that the original FIR filter has even (linear) resolution, that is, the detail is evenly distributed in the linear frequency scale. However, when frequency warping is applied, the transfer function gradually shifts toward lower frequencies with increasing λ, meaning that the level of detail is higher at low frequencies compared to high frequencies.

2 FILTER DESIGN

The basic idea of warped filter design is that the filter specification in the time-domain or frequency-domain is pre-distorted with the inverse of the warping effect of the filter. Then any traditional filter design technique can be used to design an FIR or IIR filter, and finally when the filter is implemented by using first-order all-pass elements, the filter response is mapped back to the right frequencies. A very appealing property of warped filters is that the embedded filter design step is the same as for ordinary FIR or IIR filters.

2.1 Frequency-Domain Design

The inverse mapping $\nu^{-1}(\tilde{\vartheta})$ is defined so that $\vartheta = \nu^{-1}(\nu(\vartheta))$. If the mapping function was computed by using λ in Eq. (2), then the inverse mapping could be obtained by using $-\lambda$ in the same function Eq. (2) [26].

The steps of filter design are the following:

1. **Pre-warping of the target frequency response**. The filter specification is transformed by the inverse mapping function $\nu^{-1}(\tilde{\vartheta})$. Mathematically, this mapping is described by

$$\tilde{H}_t(\tilde{\vartheta}_n) = H_t(\nu^{-1}(\tilde{\vartheta}_n)). \tag{4}$$

   In practice this can be done by some suitable interpolation, where there is direct control over the density of the target frequency points in the warped domain. A simpler alternative solution is to move the original specification points $H_t(\vartheta_n)$ to the frequencies $\tilde{\vartheta}_n = \nu(\vartheta_n)$ but leave their magnitude and phase values unchanged.
2. **Frequency-domain filter design**. An FIR or IIR filter is designed based on the pre-warped target $\tilde{H}_t(\tilde{\vartheta})$ by any of the available filter design methods, just as with ordinary FIR or IIR filters. This leads to the filter $\tilde{H}(\tilde{\vartheta})$.
3. **Filter implementation**. This is actually not part of the design process, but when the warped filter is implemented by substituting the unit delays of the FIR or IIR filter designed in step 2 by the first-order all-pass filters, the frequency response of the filter will be mapped back to match that of the original specification. Thus, the transfer function is automatically shifted from $\tilde{H}(\tilde{\vartheta})$ to $\tilde{H}(\vartheta)$ so that

$$H(\vartheta) = \tilde{H}(\nu(\vartheta)). \tag{5}$$

The steps of the filter design are illustrated by a minimum-phase loudspeaker–room response modeling example. The sixth-octave smoothed version of a loudspeaker–room response is shown in Fig. 5(a) in the

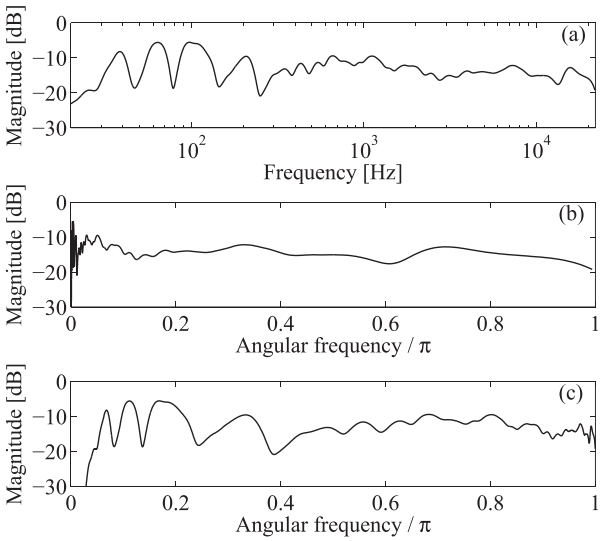

Fig. 5. Frequency warping of a smoothed loudspeaker response: (a) original response plotted in logarithmic scale, (b) the same response plotted in linear scale, and (c) the frequency response pre-warped using $\lambda = -0.95$, again in linear scale.

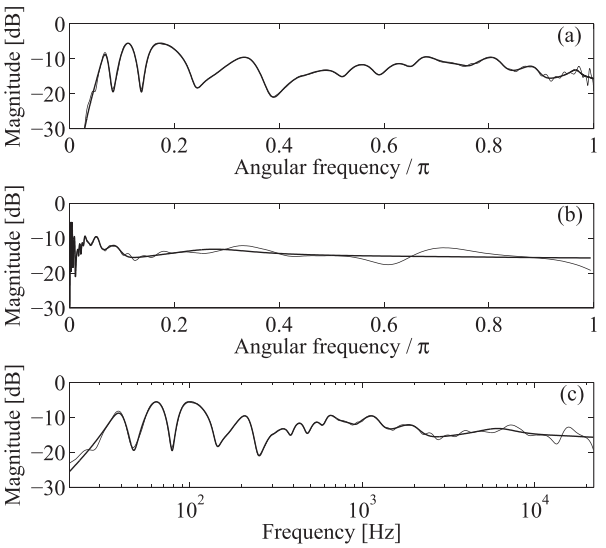

Fig. 6. Warped infinite impulse response (IIR) filter design based on a smoothed loudspeaker response: (a) the filter response designed based on the pre-warped target, (b) the filter response after substituting the unit delays with first-order all-passes ($\lambda = 0.95$), and (c) the same response plotted in logarithmic scale. The thick lines show the frequency response of the 32nd-order IIR filter, and the thin lines display the target response.

logarithmic scale, where the even distribution of detail is apparent. When shown in the linear scale in Fig. 5(b), the low-frequency detail is concentrated to a very narrow region, which would be impossible to model with traditional, linear frequency resolution filters. The response after pre-warping with $\lambda = -0.95$ is displayed in Fig. 5(c) in linear frequency scale, showing a distribution of detail quite similar to the logarithmic plot Fig. 5(a).

Next, a 32nd-order IIR filter is designed based on the warped specification by the use of the frequency-domain Steiglitz–McBride method [23]. The filter response $\tilde{H}(\tilde{\vartheta})$ is shown by a solid line in Fig. 6(a) together with the target $\tilde{H}_\mathrm{t}(\tilde{\vartheta})$ with a thin line [this is the same as Fig. 5(c)]. It can be seen that the filter follows the loudspeaker–room response very accurately.

When the IIR filter is implemented using all-pass elements in place of the unit delays, its frequency response gets automatically de-warped. This is shown by the solid line in Fig. 6(b) together with the target response (thin line). The same filter response $H(\vartheta)$ is displayed in the logarithmic scale in Fig. 6(c), showing a very good match.

2.2 Time-Domain Design

The steps of the design are the following:

1. **Pre-warping of the target impulse response**. In the time domain, the design of warped filters starts with warping the target impulse response $h_\mathrm{t}(n)$ by the use of an all-pass chain with $-\lambda$ [26]. This is done in practice by sending a unit pulse to the input of a warped FIR (WFIR) filter having $-\lambda$ warping parameter in the all-pass sections. The filter coefficients are set to be equal to the original target impulse response $b_n = h_\mathrm{t}(n)$, and the output of the filter is the pre-warped target response $\tilde{h}_\mathrm{t}(n)$. Note that a finite target impulse response $h_\mathrm{t}(n)$ results in a pre-warped target $\tilde{h}_\mathrm{t}(n)$ of infinite length, since it is the sum of the responses of (all-pass) IIR filters.
2. **Time-domain filter design**. WFIR filters can be simply obtained by truncating or windowing the warped target response $\tilde{h}_\mathrm{t}(n)$, just as what would be done when modeling an infinite impulse response with a finite length filter. Similarly, warped IIR (WIIR) filters are designed by traditional filter design algorithms (e.g., linear predictive coding, Prony, and Steiglitz-McBride) using this pre-warped $\tilde{h}_\mathrm{t}(n)$.
3. **Filter implementation**. When the filter designed in step 2 is implemented with first-order all-pass filters in place of the unit delays, its impulse response is automatically mapped back to be in accordance with the original (un-warped) target impulse response.

The steps of the time-domain design are illustrated in Fig. 7 for modeling a piano soundboard response. The target impulse response is displayed in Fig. 7(a). After pre-warping, shown in Fig. 7(b), the low-frequency content gets compressed toward the beginning of the response, whereas the high-frequency content stretches out. Then, as the simplest example of warped filter design, a WFIR filter is obtained by truncating the pre-warped response to $N = 200$ samples, displayed in Fig. 7(c). When implemented as a WFIR filter with first-order all-pass filters in place of the unit delays, the response stretches back to match the original target, shown in Fig. 7(d). It can be seen that the filter response Fig. 7(d) follows the target Fig. 7(a) at high frequencies only in the beginning of the response, whereas at low frequencies, it models entirely. This is expected because WFIR

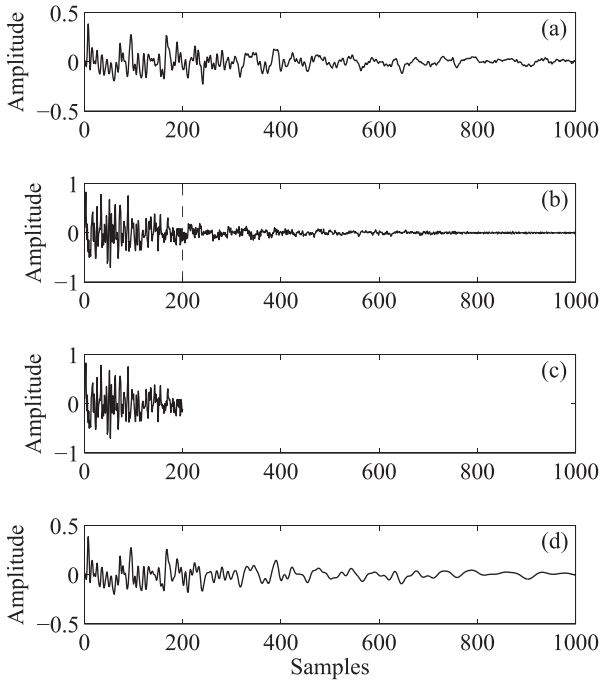

Fig. 7. Time-domain warped finite impulse response (WFIR) filter design based on a piano soundboard response: (a) target impulse response, (b) target response after pre-warping ($\lambda = -0.7$), (c) WFIR coefficients given as the first 200 samples of the pre-warped response, and (d) impulse response of the WFIR filter implemented with $\lambda = 0.7$.

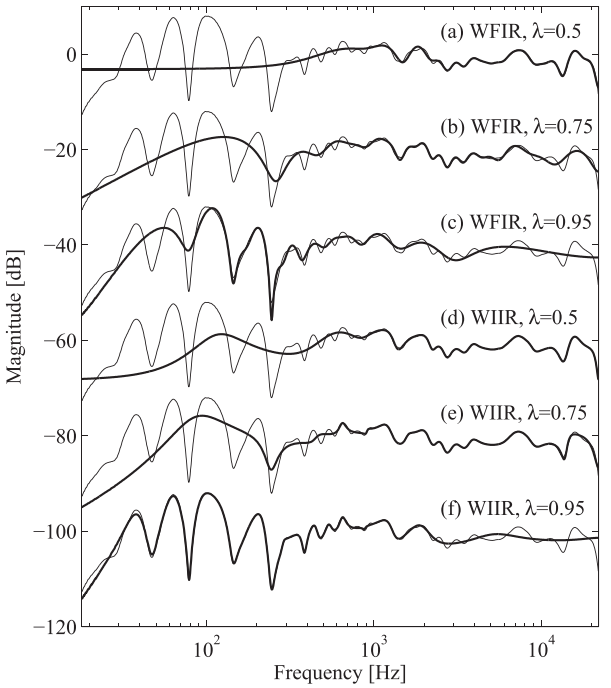

Fig. 8. Modeling a sixth-octave smoothed minimum-phase loudspeaker–room response with (a)–(c) 32nd-order warped finite impulse response (WFIR) filters and (d)–(f) 32nd-order warped infinite impulse response (WIIR) filters using various warping parameters $\lambda = 0.5, 0.8, 0.95$. The target response is displayed by thin lines, and the filter responses are shown by thick lines. The curves are offset for clarity.

filter design can be essentially considered as the frequency-dependent windowing of the target impulse response [36].

3 FILTER DESIGN EXAMPLES

Fig. 8 shows various warped filter designs based on the same sixth-octave smoothed loudspeaker–room response where the standard IIR filter design methods failed to provide logarithmic frequency resolution in Fig. 1(d). The first three curves Fig. 8(a)–8(c) correspond to WFIR filters with various λ values designed by truncating the warped impulse response. It can be seen that increasing λ shifts the region with detailed modeling in accordance with the resolution curves of Fig. 3(b). It can also be seen that none of the λ values provide even distribution of the modeling accuracy in the logarithmic frequency scale.

Figs. 8(d)–8(f) display WIIR filter designs using the Steiglitz-McBride method [22] with the same λ values as for the WFIR filters. Here again, increasing λ shifts the region of accurate modeling to low frequencies. Compared to the WFIR examples, the WIIR filters provide a better fit because they can redistribute the modeling detail by their poles. Coming from this, the region of accurate modeling is wider than for WFIR filters. However, there is still no such λ value where the accuracy is evenly distributed in the full audio bandwidth.

4 IMPLEMENTATION

Warped filters can either be implemented as special filter structures incorporating all-pass elements or be "dewarped" to traditional direct-form, cascade, or parallel realizations.

4.1 Implementation With Special Filter Structures

The WFIR filters have a similar structure to FIR filters, but the unit delays are replaced by the all-pass filter $D(z)$ as in Eq. (1). That is, the WFIR filter is an all-pass chain, where the signals between the first-order all-pass blocks are tapped and weighted by the FIR coefficients b_k. This is displayed in Fig. 9(a). Because of the all-pass elements, WFIR filters are actually IIR filters, and only their structure and design resemble that of FIR filters [26].

In contrast to WFIR filters, the implementation of WIIR filters is less straightforward. This is because for WIIR filters, the replacement of unit delays by $D(z)$ leads to delay-free loops, and the filter structure has to be modified for practical implementation [37–40]. For example, [38] proposes a simple algorithm to compute the denominator coefficients of the WIIR filter. With this technique, the all-pass structure is preserved; only the first all-pass filter is replaced by a first-order low-pass filter. In [40], an alternative realization is proposed, which fully preserves the all-pass structure of the filter network. This structure

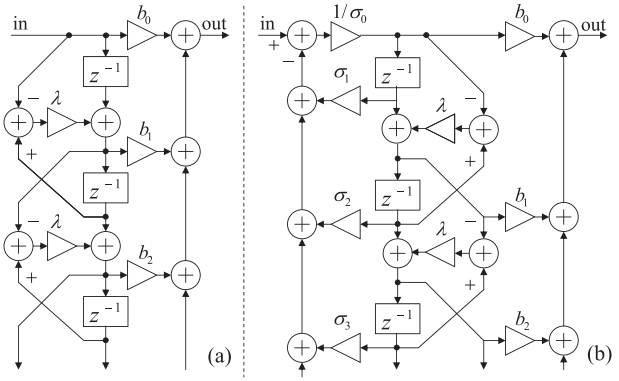

Fig. 9. Warped finite impulse response (a) and warped infinite impulse response (b) implementation using special filter structures [1].

is displayed in Fig. 9(b). Because of the specialized filter structures, WFIR and WIIR filters typically require two to four times higher computational time on digital signal processors compared with normal FIR and IIR filters of the same order [26].

4.2 De-Warping to Direct-Form Filters

Substituting the first-order all-pass transfer function Eq. (1) into the transfer function of the FIR or IIR filter leads to a direct-form IIR filter that can be implemented much more efficiently compared to the special warped structures [37, 24, 26]. The disadvantage of de-warping the filter to a direct-form realization is that it may lead to numerical issues (instability, transfer function errors, and increased quantization noise) because of pole clustering at low frequencies. It has been found that such problems may arise above filter orders of 20–30 even when using double precision floating point arithmetic [26].

4.3 De-Warping to Cascade or Parallel Sections

Another, numerically better-behaving option is to de-warp the filter to a cascade or parallel second-order structure. The idea is to first break up the transfer function of the warped filter to series or parallel second-order sections and then de-warp the sections separately [39]. Finally, the filter is implemented in this series or parallel form.

Whereas [39] presents the formulas of de-warping a second-order section, in [41] the problem is addressed by first finding the poles $\tilde{p}_k$ and zeros $\tilde{m}_k$ of the WIIR filter and then de-warping them by the expression

$$p_k = \frac{\tilde{p}_k + \lambda}{1 + \lambda \tilde{p}_k}, \quad m_k = \frac{\tilde{m}_k + \lambda}{1 + \lambda \tilde{m}_k}. \tag{6}$$

Finally, the filter is implemented as a series of second-order sections, computed from the de-warped poles p_k and zeros m_k [41]. De-warping to second-order sections is a very effective way of implementing warped filters because it leads to the same computational complexity as de-warping to direct-form filters but can also be used with high filter orders (the numerical problems coming from pole clustering are avoided).

4.4 Extensions of Basic Warping Techniques

It has been seen in Fig. 3(b) that there is no single λ value that would result in a constant resolution in the logarithmic scale. On the contrary, each λ focuses the resolution around a certain frequency region. Therefore, a straightforward development of the warped filter concept is to use different warping parameters for the different frequency regions.

4.4.1 Combination With Traditional FIR or IIR Filters

A special case of multi-band warping is when one of the bands is a traditional FIR or IIR filter (that is, for that band $\lambda = 0$). This is motivated by the fact that straightforward FIR and IIR filters can be implemented more efficiently compared to their warped counterparts. Since the resolution of traditional FIR and IIR filter design methods is linear, they are best suited for modeling or equalization of the high-frequency region of the transfer function. This can also be deduced from Fig. 3(b) showing that traditional filters (dotted line) have improved logarithmic resolution (smaller $\Delta f/f$) as frequency increases. Such a combined warped and linear equalizer was presented in [42]. The frequency band is split to two by a crossover network, which includes a low-pass and high-pass filter. The high-frequency part of the signal is processed by an FIR filter, and the low-frequency part by a WFIR filter. A similar approach has been proposed in [35] with the important difference that the FIR and WFIR filters are in cascade, eliminating the need for a crossover network. A generalization of the method was presented in [43] where, besides FIR and WFIR filters, combined IIR and WIIR equalization is also employed.

4.4.2 Multiple Warped Filters

The first paper using multiple warped filters is [44], which proposes the use of a three-band equalizer, where different λ values are chosen in the three branches to maximize the warping effect for each WFIR filter. Additionally, the middle band incorporates decimation and interpolation so that the processing is done at a reduced sample rate. The three bands are separated by a crossover network composed of a low-pass, band-pass, and high-pass filter.

A multi-band warping technique has also been developed for computing the pole positions of the fixed-pole parallel filter [31], and the first part of the method can be directly used as a WIIR filter design. In this method, the transfer function is split to two bands, and two WIIR filters are designed with such λ values that lead to maximal modeling resolution in the center of their corresponding bands. Then the two WIIR filters are combined (connected in series).

4.4.3 Custom Warping

So far, the frequency warping function Eq. (2) has been used, which gives a limited freedom because of a single parameter λ. Improved results were obtained by combining multiple warped filters with different λ values. However, a question arises if it is possible to use a truly logarithmic frequency mapping prior to filter design. The answer is

that, if implementing the filter as the previously discussed WFIR or WIIR structures, logarithmic frequency mapping is not possible because that will lead to a warping characteristic described by Eq. (2). A workaround to this is to use pole-zero de-warping instead of the all-pass transform. In "custom warping" [45], the filter design is done in the frequency domain, starting with transforming the target response by an arbitrary smooth and monotonic mapping function—for the current purposes, it should be a logarithmic function. Then an IIR filter is designed based on this transformed specification, and its poles and zeros are found. The pole and zero frequencies are transformed back with the inverse of the logarithmic mapping function, and the pole/zero radii are computed based on the derivative of this inverse mapping. Finally, a post-optimization is run to reduce the approximation errors made during de-warping, and the filter is implemented as parallel set of second-order filters.

5 KAUTZ AND PARALLEL FILTERS

It has been seen that warped filters (especially WIIR filters) provide a much better approximation to logarithmic frequency resolution compared with straightforward IIR filters shown in SEC. 0.1. However, when the full audio band from 20 Hz to 20 kHz has to be modeled or equalized, a single λ parameter is insufficient because either the high or low frequencies will lack modeling detail (see Fig. 8). This can be improved using multiple λ values, as was outlined in SEC. 4.4.

As a generalization, the question arises if it would be possible to construct the WFIR and WIIR filters in such a way that every all-pass filter has different λ values. This question has been investigated by Tyril et al. [41], who has proposed the use of warped individual z FIR (WizFIR) filters. No systematic procedure has been given for choosing the different λ parameters for the various sections; rather, they were set by trial and error, and it was found that the performance is slightly improved compared with that of normal WFIR filters. However, for the same computational complexity, WizFIR filters are outperformed by normal WIIR filters having a single λ value [41]. Thus, the use of WizFIR filters is not encouraged. The authors have also added that the benefit in using individual λ values in WIIR filters is doubtful because WIIR filters can have different poles anyway.

A mathematically better founded alternative to WizFIR filters is the use of Kautz filters, which are indeed very similar to a WFIR filter structure with different λ values, with an additional feature of orthonormal basis functions [27]. Before discussing how Kautz filters can be used to achieve logarithmic frequency resolution, the history of the method is first reviewed.

5.1 Laguerre and Kautz Models

Traditionally, Laguerre and Kautz models were proposed for system identification. These models reconstruct the system response as a linear combination of orthonormal basis functions. In the case of Laguerre models [46], the orthonormalization procedure is started from identical first-order low-pass transfer functions $1/(1 - pz^{-1})$ with a pole at p. This gives the following set of orthonormal functions:

$$L_k(z) = \frac{\sqrt{1-p^2}}{1-pz^{-1}} \left(\frac{z^{-1}-p}{1-pz^{-1}} \right)^{k-1}, \tag{7}$$

for $k = 1, 2, \ldots K$. The term in the parenthesis corresponds to an all-pass filter; indeed, Laguerre models lead to the same filter structure as WFIR filters, and the only difference is the normalization term $\sqrt{1-p^2}/(1-pz^{-1})$, which is simply a low-pass filter at the input of the all-pass backbone.

A straightforward generalization of Laguerre filters is when the orthonormalization process is started from first-order low-pass filters having different poles (note that the poles p_k can also be complex). For continuous-time systems, the concept was introduced by Kautz [47], and the corresponding discrete-time orthonormal sequences were first presented by Broome [48]. The orthonormal polynomials take the following form [27]:

$$G_k(z) = \frac{\sqrt{1-p_k\overline{p}_k}}{1-p_kz^{-1}} \prod_{j=1}^{k-1} \frac{z^{-1}-\overline{p}_j}{1-p_jz^{-1}}, \tag{8}$$

for $k = 1, \ldots K$, where $\overline{p}_k$ is the complex conjugate of p_k. (Note that in [27] the indexing starts from $k = 0$.) Again, the filter can be implemented as a tapped all-pass backbone, but now the poles of the filter are different. Therefore, the first-order low-pass normalization terms $\sqrt{1-p_k\overline{p}_k}/(1-p_kz^{-1})$ have to be implemented separately after the tapping points of the backbone. This is similar to the WizFIR filter of [41] with the added first-order low-pass filters at the tap outputs.

Eq. (8) results in complex sequences (impulse responses) for complex poles. Also, such a model would result in a filter with complex coefficients. However, in practice, most applications require the modeling of real impulse responses and the use of filters with real coefficients. For such systems, complex poles always appear in complex conjugate pairs p_i and $\overline{p}_i$, and the complex pole pairs can be combined to form second-order sections. For a pole pair p_i and $\overline{p}_i$, a pair of real valued basis functions $G_i^+(z)$ and $G_i^-(z)$ is obtained as follows [48, 27]:

$$\begin{aligned} A_i(z) &= \frac{1}{(1-p_1z^{-1})(1-\overline{p}_1z^{-1})} \times \\ &\quad \times \prod_{j=2}^{i} \frac{(z^{-1}-p_{j-1})(z^{-1}-\overline{p}_{j-1})}{(1-p_jz^{-1})(1-\overline{p}_jz^{-1})} \\ G_i^+(z) &= C_i^+(1+z^{-1})A_i(z) \\ G_i^-(z) &= -C_i^-(1-z^{-1})A_i(z), \end{aligned} \tag{9}$$

for $i = 1, 2, \ldots I$. In Eq. (9), C_i^+ and C_i^- are normalization constants computed from the pole set p_i [48, 27]. Note that since

$$A_i(z) = A_{i-1}(z) \frac{(z^{-1}-p_{i-1})(z^{-1}-\overline{p}_{i-1})}{(1-p_iz^{-1})(1-\overline{p}_iz^{-1})}, \tag{10}$$

each $A_i(z)$ can be implemented by filtering the previous term $A_{i-1}(z)$ with a second-order filter, meaning that the $A_i(z)$ part

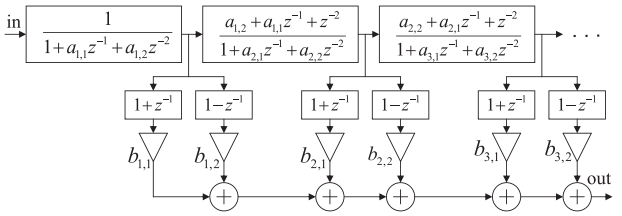

Fig. 10. The structure of the Kautz filter with real coefficients.

can be implemented as a backbone composed of second-order stages. Then the signal is tapped between the sections and filtered by the first-order numerator terms $C_i^+(1+z^{-1})$ and $-C_i^-(1-z^{-1})$ to obtain the outputs $G_i^+(z)$ and $G_i^-(z)$ [48]. The total filter response is the linear combination of these outputs with weights w_k. The block diagram of the Kautz structure in this efficient real form is displayed in Fig. 10, where $a_{i,1} = -2\mathrm{Re}\{p_i\}$, $a_{i,2} = |p_i|^2$, $b_{i,1} = w_{2i-1}C_i^+$, and $b_{i,2} = -w_{2i}C_i^-$.

Both for Laguerre and Kautz models, the impulse response of a system $h_\mathrm{t}(n)$ is modeled as a linear combination of basis functions $x_k(n)$, which are obtained as the inverse–z transform of $L_k(z)$ for Laguerre and $G_i^+(z)$ and $G_i^-(z)$ for Kautz filters:

$$h(n) = \sum_{k=1}^{K} w_k x_k(n), \tag{11}$$

where w_k is the weights. The goal is to estimate w_k such that the model response $h(n)$ is closest to the target $h_\mathrm{t}(n)$. Since the $x_k(n)$ sequences are orthonormal, the optimal solution in the mean-squared sense is given by the scalar product

$$w_k = \sum_{n=0}^{N} h_\mathrm{t}(n)x_k(n), \tag{12}$$

requiring much less computations compared to solving the usual least-squares (LS) equations required for non-orthogonal basis functions. This complexity can be decreased even more by noting that the scalar product of Eq. (12) is equivalent to convolving the time-reversed target $h_\mathrm{t}(-n)$ with $x_k(n)$ and taking the output for $n = 0$ [48, 27]. Convolution with $x_k(n)$ is actually done by filtering $h_\mathrm{t}(-n)$ with the same recursive Laguerre or Kautz filter structure that is used for modeling, and w_k is simply obtained by reading the tap outputs at time $n = 0$.

Nowadays solving a linear LS problem is considered as one of the simplest optimization problems; thus, the orthonormality of Laguerre and Kautz basis functions has lost some of its attractiveness. However, for some cases such as adaptive filtering, orthonormality is still highly beneficial because it leads to faster convergence [49, 50].

5.2 Kautz Filters for Audio Applications

Using Kautz filters for audio equalization was first proposed by Paatero et al. [51, 27]. In these papers, the poles of the Kautz filter are either set manually based on the measured response or based on a (possibly warped) filter design using a somewhat complicated procedure. This procedure adapts the Brandenstein-Unbehauen method [52]

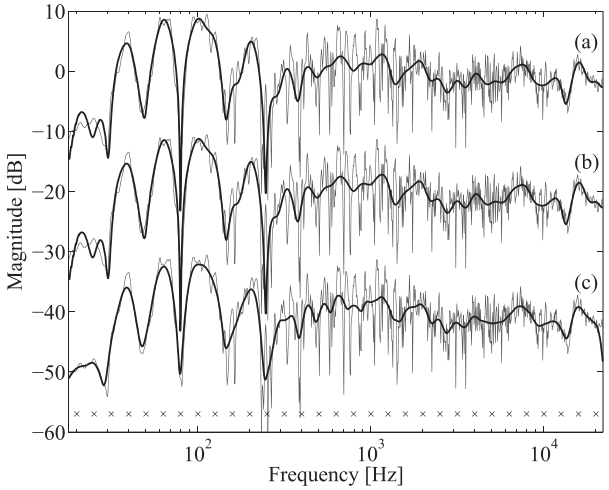

Fig. 11. Modeling a loudspeaker–room response with a 62nd-order Kautz filter (a) and 62nd-order parallel filter (b) having the same set of poles (three poles per octave from 20 Hz to 20 kHz). The pole frequencies are indicated by crosses in the bottom of the figure. The thin gray lines show the minimum-phase target response, and the thick lines the filter response. In (c), the thick line is the sixth-octave smoothed version of the target (thin gray line). The curves are offset for clarity.

to the Kautz filters, which is an iterative LS optimization technique similar to the Steiglitz-McBride algorithm [22]. However, probably the most important step in making Kautz filters easily usable for audio filter design was when Karjalainen and Paatero [4] recognized that by setting the pole frequencies according to a logarithmic frequency scale and setting the pole radii as an exponentially damping sequence, truly logarithmic frequency resolution can be achieved.

Fig. 11(a) shows a Kautz filter design when the pole frequencies are distributed uniformly in the logarithmic scale: a total of 31 pole pairs have been used, meaning three pole frequencies per octave. It can be seen that the modeling detail is now evenly distributed in the logarithmic scale, in contrast to WFIR and WIIR designs shown in Fig. 8. This can be confirmed by comparing the Kautz filter response of Fig. 11(a) to the sixth-octave smoothed version of the transfer function displayed in Fig. 11(c), showing a great similarity. Since pole positioning for Kautz filters is essentially the same task as for parallel filters, SEC. 6 will discuss the topic further.

5.3 Fixed-Pole Parallel Filters

It may be concluded that Kautz filters provide an attractive way for constructing logarithmic frequency resolution filters. However, the combined cascade-parallel nature of the Kautz filter visible in Fig. 10 requires more computation compared with filters implemented in direct, cascade, or parallel form for the same filter order.

A question then arises whether it is possible to keep the direct control over frequency resolution with simpler filter structures. The experiments of the present author have shown that the key for such a desired performance does not actually lie in the Kautz filter structure, but in the fixed-pole

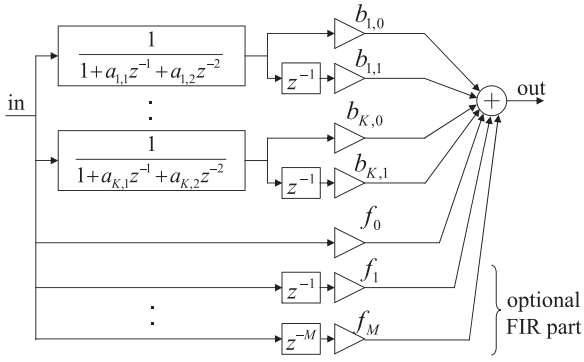

Fig. 12. Structure of the parallel second-order filter.

design. It has been found that if the pole set is the same as for the Kautz filter, practically equivalent results can be achieved if the filter is designed in a parallel second-order form, and thus the concept of the fixed-pole parallel filter was born [53].

Implementing IIR filters in the form of parallel second-order sections has been used traditionally because of its better quantization noise performance compared to direct-form filters, similarly to series biquads [54, 21, 55]. The parameters of the second-order sections are then determined from the direct form IIR filters, e.g., by the partial fraction expansion or a similar algorithm [54, 21, 56]. The novelty of the present methodology lies in the fact that instead of converting from a direct-form IIR filter, the parallel second-order filter bank is designed directly, and that by the choice of the pole frequencies, direct control over the frequency resolution of the design is gained similarly to the Kautz filter.

The complex form of the fixed-pole parallel filter ("parallel filter" in short) is actually related to decomposing a rational transfer function $H(z) = B(z)/A(z)$ to partial fractions:

$$H(z) = \sum_{i=1}^{P} c_i \frac{1}{1 - p_i z^{-1}} + \sum_{m=0}^{M} f_m z^{-m}, \tag{13}$$

where p_i is the poles, either real valued or forming conjugate pairs, if the system has a real impulse response. The second sum in Eq. (13) is the FIR filter part of order M. Note that Eq. (13) assumes that there is no pole multiplicity; otherwise terms of higher order would also appear [57].

The resulting filter can be implemented directly as in Eq. (13), forming parallel first-order complex filters. However, it is more practical to combine the complex pole pairs to a common denominator resulting in a parallel set of second-order sections with real valued coefficients:

$$H(z) = \sum_{k=1}^{K} \frac{b_{k,0} + b_{k,1} z^{-1}}{1 + a_{k,1} z^{-1} + a_{k,2} z^{-2}} + \sum_{m=0}^{M} f_m z^{-m}, \tag{14}$$

where K is the number of second-order sections. The filter structure is depicted in Fig. 12.

For most modeling or equalization tasks, there is no need for the FIR part. On the other hand, for non-decaying responses where the peak of the target response is not in the beginning, using the FIR path for the early, rising part of the response improves modeling accuracy for a given computational complexity [53]. Related to this, it must be mentioned that the original complex [Eq. (13)] and real [Eq. (14)] forms of the parallel filter were constructed so that they are in line with the forms used in partial fraction expansion [54, 21], with the FIR part of the impulse response overlapping the IIR part. Meanwhile, it has turned out that this overlap can lead to a dynamic range limitation in real-world applications using limited word lengths (e.g., 24-bit fixed-point) [58]. This can be simply avoided by using the "delayed parallel filter," where the IIR part is delayed to start after the FIR response. While here, the original (non-delayed) parallel filter design will be discussed; the delayed parallel filter can be designed very similarly, or its coefficients can be easily converted from those of the non-delayed form [58].

5.4 Parameter Estimation for Parallel Filters

The filter weights $b_{k,0}$, $b_{k,1}$, and f_m can be estimated both in the time and frequency domains; here, the more flexible frequency-domain variant will be outlined [59]. (For the time domain design, see [53, 28], whereas [59] compares the time-domain and frequency-domain designs.) Substituting $z^{-1} = e^{-j\vartheta_n}$ into Eq. (14) for a finite set of ϑ_n angular frequencies yields

$$H(\vartheta_n) = \sum_{k=1}^{K} \frac{b_{k,0} + b_{k,1} e^{-j\vartheta_n}}{1 + a_{k,1} e^{-j\vartheta_n} + a_{k,2} e^{-j2\vartheta_n}} + \sum_{m=0}^{M} f_m e^{-jm\vartheta_n}, \tag{15}$$

which is linear in its free parameters $b_{k,0}$, $b_{k,1}$, and f_m. (The coefficients $a_{k,1}$ and $a_{k,2}$ are constant because of the fixed pole set.)

Because of its linearity, Eq. (15) can be written in a matrix form

$$\mathbf{h} = \mathbf{M}\mathbf{p}, \tag{16}$$

where $\mathbf{p} = [b_{1,0}, b_{1,1}, \ldots b_{K,0}, b_{K,1}, f_0 \ldots f_M]^T$ is a column vector composed of the free parameters. The columns of the modeling matrix $\mathbf{M}$ contain the transfer functions of the second-order sections $1/(1 + a_{k,1}e^{-j\vartheta_n} + a_{k,2}e^{-j2\vartheta_n})$ and their delayed versions $e^{-j\vartheta_n}/(1 + a_{k,1}e^{-j\vartheta_n} + a_{k,2}e^{-j2\vartheta_n})$ for the ϑ_n angular frequencies. The last columns of $\mathbf{M}$ are the transfer functions of the FIR part $e^{-jm\vartheta_n}$ for $m = [0 \ldots M]$. Finally, $\mathbf{h} = [H(\vartheta_1) \ldots H(\vartheta_N)]^T$ is a column vector composed of the resulting frequency response.

Now the task is to find the optimal parameters $\mathbf{p}_{\text{opt}}$ such that $\mathbf{h} = \mathbf{M}\mathbf{p}_{\text{opt}}$ is closest to the target frequency response $\mathbf{h}_t = [H(\vartheta_1)_t \ldots H_t(\vartheta_N)]^T$. If the error is evaluated in the mean squares sense

$$e_{\text{LS}} = \sum_{n=1}^{N} |H(\vartheta_n) - H_t(\vartheta_n)|^2 = (\mathbf{h} - \mathbf{h}_t)^H (\mathbf{h} - \mathbf{h}_t), \tag{17}$$

and the minimum of Eq. (17) is found by the LS solution:

$$\mathbf{p}_{\text{opt}} = \mathbf{M}^{+}\mathbf{h}_{\text{t}},$$
$$\mathbf{M}^{+} = (\mathbf{M}^{H}\mathbf{M})^{-1}\mathbf{M}^{H}, \tag{18}$$

where $\mathbf{M}^{+}$ is the Moore-Penrose pseudoinverse, and $\mathbf{M}^{H}$ is the conjugate transpose of $\mathbf{M}$. Note that $\mathbf{M}^{+}$ can be precomputed if the pole set (frequency resolution) is fixed.

Eq. (18) assumes a filter specification $H_{\text{t}}(\vartheta_n)$ given for the full frequency range $\vartheta_n \in [-\pi, \pi]$. Thus, the design can be used for obtaining filters with complex coefficients, since the frequency specification is not constrained to be conjugate-symmetric. However, in most cases, the interest is in filters with real coefficients: in this case, it must be ensured that $H_{\text{t}}(-\vartheta_n) = \overline{H}_{\text{t}}(\vartheta_n)$, where $\overline{H}_{\text{t}}$ is the complex conjugate of H_{t} (an alternative approach using a one-sided specification is outlined in [59]).

A clear benefit of designing the fixed-pole parallel filter in the frequency domain is that this allows adding different weights to the different frequency points [59]. In this case, the error becomes

$$e_{\text{WLS}} = \sum_{n=1}^{N} W(\vartheta_n)|H(\vartheta_n) - H_{\text{t}}(\vartheta_n)|^2 = \\ = (\mathbf{h} - \mathbf{h}_{\text{t}})^{H}\mathbf{W}(\mathbf{h} - \mathbf{h}_{\text{t}}), \tag{19}$$

where $W(\vartheta_n)$ is the weight for the ϑ_n frequency and $\mathbf{W}$ is the weighting matrix having $W(\vartheta_n)$ in its diagonal and zeros elsewhere. The minimum is obtained by the weighted-LS (WLS) solution:

$$\mathbf{p}_{\text{opt}} = (\mathbf{M}^{H}\mathbf{W}\mathbf{M})^{-1}\mathbf{M}^{H}\mathbf{W}\mathbf{h}_{\text{t}}. \tag{20}$$

As for the design complexity, frequency-domain design requires significantly less computations compared with the time-domain design when the target frequency response is given in the logarithmic scale, and it is also numerically better conditioned [59]. Therefore, it is advised to convert time-domain target responses to the frequency domain before filter design.

Another advantage of frequency-domain design compared with the time-domain version is that it is usable also when only a magnitude specification is given. For that, [59] proposes a simple iterative procedure where the design is started from a minimum-phase target computed from the given magnitude specification, and the phase is updated in each iteration based on the previous filter approximation.

Note that this procedure is similar to "magnitude-priority" filter design [60], where the iterative process is started from the original phase specification. This technique is preferred when the phase specification is given, but matching the magnitude has higher importance. The method models or equalizes both the magnitude and phase in those frequency regions where this is feasible with the given filter order, whereas if this is not possible, it modifies the phase specification to allow for a better magnitude match.

5.5 Mathematical Equivalence of the Kautz and Parallel Filters

Figs. 11(a) and 11(b) show Kautz and parallel filter designs using the same pole set having 31 pole pairs distributed uniformly in the logarithmic scale. As can be seen, the same filter response arises for both filters.

This can be explained by the fact that the Kautz basis functions seen in Eq. (8) are the orthonormalized versions of decaying complex exponentials, which are actually the basis functions of the complex parallel filter Eq. (13). A formal proof for the equivalence has been presented in [28] based on the partial fraction expansion of the complex Kautz basis functions Eq. (8).

If the parameters of the Kautz filter in its complex form Eq. (8) are given in a vector $\mathbf{w} = [w_1, \ldots, w_K]^T$, the parameter vector of the parallel filter $\mathbf{c} = [c_1, \ldots, c_K]^T$ in Eq. (13) can be obtained by the matrix multiplication

$$\mathbf{c} = \mathbf{K}\mathbf{w}, \tag{21}$$

where the conversion matrix $\mathbf{K}$ is given as

$$K_{i,k} = \sqrt{1 - p_k\overline{p}_k} \prod_{j=1, j\neq i}^{k} \frac{1}{p_i - p_j} \prod_{j=1}^{k-1}(1 - \overline{p}_j p_i) \quad \text{for } i \leq k,$$
$$K_{i,k} = 0 \quad \text{for } i > k. \tag{22}$$

This allows the conversion of an already designed Kautz filter to the computationally more efficient parallel second-order form. Note that the parallel FIR part (f_m) in Eq. (13) is zero in this case, and the conversion assumes having no pole multiplicity in the Kautz filter. An inverse mapping of the parameters (parallel to Kautz conversion) is of course also possible by using $\mathbf{w} = \mathbf{K}^{-1}\mathbf{c}$.

Basically, this proves that the basis functions of the parallel and Kautz filters span the same approximation space, and converting between the two filters is merely a change of basis. Therefore, approximating a target response using any error norm (e.g., the L_2 norm in LS design) will lead to the same filter response in both cases for a given pole set p_k.

Although the complex forms of the Kautz and parallel filters have been related, since the real forms Eqs. (9) and (14) are mathematically equivalent to the complex ones, the results are valid for the more practical real forms as well. This means that pole positioning techniques developed for the parallel filter can also be used for the Kautz filter and vice versa. Also, the smoothing properties of the two filters can be discussed jointly.

5.6 Controlling the Frequency Resolution: Relations to Transfer Function Smoothing

The thick line in Fig. 11(c) shows the sixth-octave complex smoothed version of the target (thin gray line). By observing the results of parallel and Kautz filters with a logarithmic pole distribution in Figs. 11(a) and 11(b), it is apparent that the effect of filter design is similar to that of fractional-octave complex smoothing of transfer functions. This smoothing behavior has been systematically analyzed

in [28] for the parallel filter, but it is equally valid for the Kautz filter because of their equivalent approximation for the same pole set.

It has been shown in [28] that for a linear (uniform) set of pole frequencies, the impulse response of the parallel filter can be approximated by the windowed (truncated) version of the target impulse response. In the frequency domain, the parallel filter response thus becomes the target frequency response smoothed (convolved) by a sinc function. It is also shown that if the pole density is different in different frequency regions, the length of the time-domain windowing will also be different, leading to a different amount of smoothing in the frequency domain.

It has been demonstrated in [28] that having α pole frequencies per octave leads to smoothing by a sinc function having the main lobe width of $1/\alpha$ octave. The main lobe of the sinc function is actually quite similar to a Hann window often used in complex smoothing, and for smoothing with a Hann window, it is the half with of the Hann window which determines the frequency resolution [28]. As a result, placing α pole frequencies per octave is comparable to smoothing the target response to $1/(2\alpha)$ octave resolution. This is indeed visible in Fig. 11 where having three complex conjugate pole pairs per octave leads to filter responses similar to the 6th-octave smoothed version of the target.

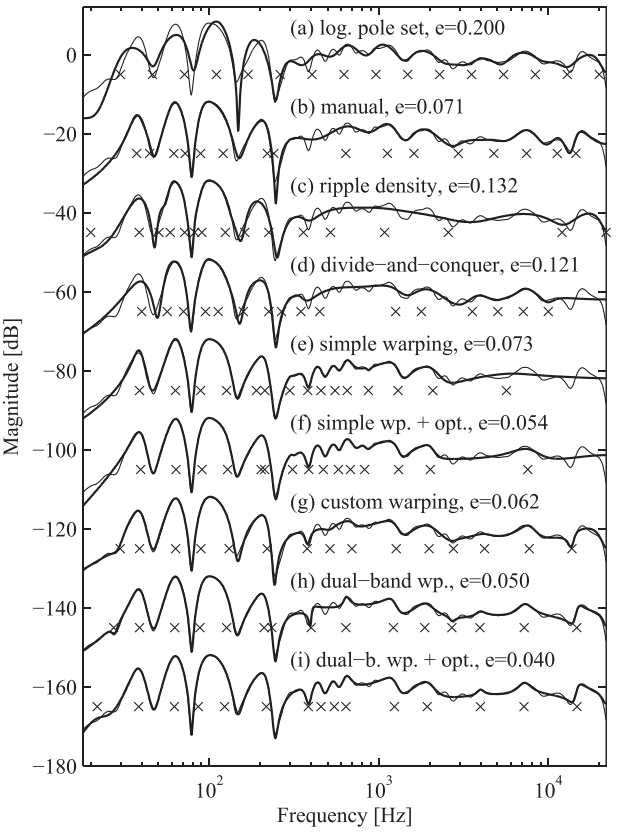

Fig. 13. Comparison of different pole positioning techniques for modeling a sixth-octave smoothed minimum-phase loudspeaker–room response using a fixed-pole parallel filter. The pole positioning techniques used are (a) logarithmic pole set, (b) manual pole positioning, and pole positioning based on (c) ripple density and the (d) divide-and-conquer approach. Next, pole positioning based on (e) standard warped infinite impulse response filter design, (f) the same with post-optimization, (g) custom warping, (h) multi-band warping, and finally, (i) multi-band warping with post-optimization. The target response is shown by thin lines. The pole frequencies are displayed by crosses. The number of pole pairs (or second-order sections) is 16 in all cases, giving a total filter order of 32. The curves are offset for clarity. The square root of the mean squared error computed in logarithmic scale is given as e = ... for each curve.

6 POLE POSITIONING STRATEGIES FOR KAUTZ AND PARALLEL FILTERS

The various pole positioning techniques can be put into two main categories, depending on the relation of the system and model order, in other words, if one wishes to model or equalize the system precisely or only approximately.

6.1 Pole Positioning Based on the Transfer Function

In some cases, one wishes to model a system as accurately as possible. For example, for artificial reverberation [61] or instrument body modeling in physics-based sound synthesis [53], one would like to keep the long decay and reverberant character of the measured response, not only its effect on the timbre. In this case, the original (unsmoothed) response should be used as a target for filter design. This will naturally require high filter orders for high-order systems, and the filter poles should correspond to system poles for best accuracy. This is, in a way, related to the field of system identification because the parameters of the model have a direct connection to physical reality.

A straightforward approach is that an IIR filter is designed based on the target response, and the poles of this IIR filter are used as the poles of the parallel filter. Here the accuracy is determined by how well the original IIR filter design approximates the target response. As discussed in SEC. 0, standard IIR filters have a linear frequency resolution, so for audio, one would rather design a WIIR filter, find the poles, and de-warp them using Eq. (6). In this case, the frequency resolution is controlled by the warping parameter, and the response of the Kautz or parallel filter is practically the same as that of the warped filter from which its poles originate, as will be seen later in Fig. 13(e).

This approach is suggested in [27], where the Brandenstein-Unbehauen IIR filter design algorithm [52] is run on the warped impulse response, and then the poles of the resulting IIR filter are de-warped by Eq. (6) and used as Kautz filter poles. Alternatively, other IIR filter design techniques can be also be used for determining the (warped) IIR filter from which the poles are taken [53]. The present author has made various tests using the Brandenstein-Unbehauen technique and found that it practically gives the same approximation error as the Steiglitz-McBride method [22]. An advantage of the latter is that it is readily available in MATLAB Signal Processing Toolbox as the `stmcb` function.

SEC. 4.4 discussed that the accuracy of the WIIR filter design can be improved by using multiple λ values or

custom warping. This improvement of course translates to pole positioning for the parallel and Kautz filters as well. A comparison of these techniques is given in [62], showing that dual-band and custom warping have a similar performance and both of them significantly outperform the pole positioning based on a straightforward single-band WIIR filter design. In the experience of the present author, these methods allow estimating high-order ($N \approx 1{,}000$) models without problems (see, e.g., [45]). Note that at such high orders, (warped) IIR filter design and the subsequent root finding of the denominator may lead to a few unstable poles; however, these poles can be discarded or flipped inside the unit circle before estimating the Kautz or parallel filter weights.

Maestre et al. [63] have further developed the pole positioning method based on WIIR design by using an iterative procedure where the poles obtained from a single-band WIIR design using the Prony or Steiglitz-McBride technique are post-optimized with a gradient descent algorithm. For modeling reverberant room responses that require high filter orders ($N \approx 1{,}000$), the technique is used in sub-bands to keep the optimization process manageable [61].

Recently, an interesting pole positioning technique has been presented in [8] for the application of sparse approximation of highly reverberant room impulse responses by Kautz filters. Instead of aiming at modeling all the details of the room impulse, the proposed technique tries to fit the most prominent resonances by iteratively testing a set of candidate pole positions and adding the one that decreases the approximation error the most to the final pole set. The examples demonstrate that the method provides a stable approximation of room impulse responses with filter orders up to the range of $N \approx 1{,}000$. However, the pole frequencies and damping factors cannot be estimated as accurately as with the IIR filter design–based approaches because they are chosen from a predefined pole set [8].

6.2 Pole Positioning Based on a Predetermined Pole Set

In contrast to the case discussed in SEC. 6.1, one may aim at modeling or equalizing only the most important perceptual features of the transfer function (e.g., how it affects the timbre), which is better described by its logarithmically smoothed version, as already discussed in SEC. 0. Now one can take advantage of the fact that the resolution of the Kautz or parallel filter design can be directly controlled by the pole density, as discussed in SEC. 5.6. For this, a predetermined (e.g., logarithmically distributed) set of poles needs to be used based on the modeling resolution desired. Because in this case, the poles of the filter are unrelated to the original system, this is more like a nonparametric approach.

For logarithmic frequency resolution, if one wishes to achieve a result similar to $1/\beta$ octave smoothing, $2/\beta$ octave pole frequency distances are necessary. Table 1 displays the pole densities required to achieve the most typical fractional-octave resolutions used in transfer function smoothing. It also lists the number of pole frequencies $K = 10(\beta/2) + 1$, assuming a design with the ten octaves of the full audio bandwidth, and the total filter order, $2K$.

Table 1. Pole densities, number of pole frequencies, and filter orders for typical values of fractional-octave resolutions using a predetermined pole set. The pole frequencies span ten octaves from 20 to 20,480 Hz.

Fractional octave resolution	1	1/3	1/6	1/12	1/24
Pole frequencies per octave	1/2	3/2	3	6	12
Total no. pole frequencies	6	16	31	61	121
Filter order	12	32	62	122	242

In [28], a general set of pole equations is presented, which can determine the pole radii for an arbitrary set of pole frequencies. Once the pole frequencies f_k are predefined, the digital angular frequencies

$$\theta_k = 2\pi \frac{f_k}{f_s} \tag{23}$$

are computed, with f_s being the sample rate.

Next, the bandwidth of the kth second-order section $\Delta\theta_k$ is computed from the neighboring pole frequencies

$$\begin{aligned} \Delta\theta_k &= \tfrac{\theta_{k+1}-\theta_{k-1}}{2} \quad \text{for } k = [2, .., K-1] \\ \Delta\theta_1 &= \theta_2 - \theta_1 \\ \Delta\theta_K &= \theta_K - \theta_{K-1}. \end{aligned} \tag{24}$$

Then the poles p_k are obtained as

$$p_k = e^{-\frac{\Delta\theta_k}{2}} e^{j\theta_k}. \tag{25}$$

Eq. (25) sets the pole radii $|p_k|$ in such a way that the transfer functions of the Kautz or parallel filter sections cross approximately at their –3 dB points. For a filter having a real impulse response, the poles must be in complex-conjugate pairs; therefore, the pole set needs to be extended by the complex-conjugate version $\overline{p}_k$ of the poles p_k.

An example design for both the Kautz and parallel filters has been already shown having three poles per octave in Fig. 11, leading to sixth-octave resolution as expected from Table 1.

Of course the pole frequency series does not need to be strictly logarithmic; for example, it is possible to obtain higher resolution by having larger pole density at some specific frequency regions. It is also possible to choose the pole frequencies completely freely but still compute the pole radii with the above Eqs. (23)–(25). This is useful for manual pole positioning when the user can add or remove poles to specific frequency regions while checking the result in real time. The parallel filter homepage (see SEC. 9) includes an interactive MATLAB/Octave script `parfiltdemo.m`, where the interested reader is invited to experiment with adding/removing poles by a click of a mouse.

6.3 Pole Positioning Based on the Smoothed Response

Although using a predetermined (e.g., logarithmic) pole set is well suited to modeling higher-order systems with low-order filters as seen in SEC. 6.2, it is still possible to decrease the filter order for the same accuracy. The basic idea is that the target response is smoothed to the required resolution (which can be different in the various frequency regions), and then this smoothed response is used to determine the optimal pole set of the filter. For this, the most straightforward approach is to design a WIIR filter based on this smoothed response by any of the methods discussed in SEC. 6.1.

Note that since the (warped) IIR filter is designed based on the smoothed version of the target, the poles and zeros of the Kautz or parallel filter will not have a direct connection with the system poles and zeros. So in effect, this approach is similar to that of a predetermined pole set (SEC. 6.2) with the difference that now the frequency resolution is not controlled by the direct choice of the poles but by the resolution of the smoothing applied to the original transfer function. The reason why this can lead to lower filter orders compared with the predetermined (e.g., logarithmic) pole set is because some regions of the system response may be smoother than others, and a strictly logarithmic frequency resolution may waste some computational resources at these already smooth regions. For example, a typical loudspeaker-room response has larger deviations at low frequencies compared to the high ones after smoothing [see Fig. 11(c)].

In addition to the WIIR filter design variants discussed in SEC. 6.1, it is also possible to determine the pole positions based on the raggedness of the smoothed transfer function. The ripple-density–based technique [62] places more poles to those frequency regions where the transfer function has more ripples. Although this does not provide as good accuracy as the WIIR-based methods, it has the benefit of being a significantly simpler design algorithm. Recently, a similarly simple approach has been presented in [9] using a divide-and-conquer approach by iteratively placing more and more poles in the regions of large deviations, best-suited for low-order ($N \approx 10$) filter design.

6.4 Pole Positioning Comparison

Here some of the pole positioning methods presented will be compared on the same sixth-octave smoothed loudspeaker–room response modeling example as used in Fig. 8. The filter order is also kept the same to make the figures directly comparable. The numbers next to the curves in Fig. 13 display the square root of the mean squared error calculated on a logarithmic frequency grid using 100 points per octave. Note that since the difference is computed between the complex target and filter responses, a low number not only indicates a good match in magnitude, but in phase as well.

Fig. 13(a) shows the modeling using a predetermined pole set with strictly logarithmic pole positioning. The even distribution of modeling detail in the logarithmic frequency scale is immediately apparent. Note that when using a predetermined pole set, smoothing the target prior to filter design is not necessary, as seen in Fig. 11. A smoothed target is used here only to make the comparison with other pole positioning approaches more coherent.

Fig. 13(b) displays the result of manual pole positioning using the `parfiltdemo.m` script (see SEC. 9). It can be seen that quite a good fit can be achieved with some human interaction. Note that manual tuning of the poles is much simpler than it seems because it is only the pole frequencies that are adjusted by the user, and the pole radii are computed automatically according to Eqs. (24) and (25). For the present example, the author has spent about 2 min of moving around the pole frequencies.

It is also possible to determine the pole frequencies automatically while keeping the idea of computing the pole radii based on the distance of the neighboring pole frequencies using Eqs. (24) and (25). The modeling using a ripple-density–based pole frequency set [62] is shown in Fig. 13(c), and Fig. 13(d) displays the result of the divide-and-conquer approach [9]. Although the fit is improved compared with that of a strictly logarithmic pole set [Fig. 13(a)], none of the approaches can reach the accuracy of manual tuning.

The four methods displayed in Figs. 13(a)–13(d) are all similar in that the pole radii are determined from the pole frequencies by Eqs. (24) and (25). Next, more complex approaches that optimize the pole radii independently from the pole frequencies will be examined.

In the example of Fig. 13(e), a 32nd-order IIR filter is designed by the frequency-domain Steiglitz-McBride method [23] based on the warped version of the smoothed system response with $\lambda = 0.95$. It can be seen in Fig. 13(e) that the warping-based pole positioning provides an accurate modeling from 100 Hz to 2 kHz, but it fails to provide the same accuracy in the full audio frequency range. Note also that the transfer function is practically the same as that of the WIIR design of Fig. 8(f). This is expected because the poles of a WIIR filter designed using the same λ and filter order are being used. The results of the warping-based design with post-optimizing the poles as suggested in [63] are shown in Fig. 13(f). The fit improves at high frequencies, but the distribution of the detail is still not even in the logarithmic scale.

The design using poles obtained by custom warping is displayed in Fig. 13(g), showing that the frequency resolution is spread much more evenly in the logarithmic scale compared with simple warping, and thus an excellent modeling performance is achieved. In Fig. 13(h), dual-band warping is applied. The split frequency is 500 Hz, and the two warping parameters are $\lambda_{\mathrm{LF}} = 0.986$ and $\lambda_{\mathrm{HF}} = 0.65$. The filter orders are 16 in both the low and high bands. The performance is excellent, similarly to the custom warping case.

Although [63] uses simple warping, of course the post-optimization to a pole set coming from custom or dual-band warping can also be applied. The latter is shown in Fig. 13(i), where the post-optimization is able to correct the inaccuracies at the crossover frequency of the dual-

warped design [see at 500 Hz in Fig. 13(h)]. Accordingly, custom or dual-band warping with post-optimization can be considered the state of the art in pole positioning for parallel or Kautz filters.

7 DISCUSSION

7.1 Computational Complexity

When looking at Fig. 9, it can be seen that WFIR filters require four multiply and accumulate (MAC) and two addition (ADD) operations per second-order section, and for WIIR filters, six MAC and two ADD instructions are needed. On general purpose processors, where no MAC instruction is available, this leads to four multiplications and six ADDs for the WFIR case, and six multiplications and eight ADDs for the WIIR case. This is roughly double compared with a straightforward second-order IIR filter, and in practice, the computational complexity becomes even larger because of the overhead needed because of the relatively complicated filter structures of Fig. 9.

On the other hand, when the WFIR or WIIR filters are de-warped to series second-order sections as discussed in SEC. 4.3, the computational complexity becomes the same as for straightforward IIR filters, that is, four MACs per second-order section or four multiplications and four ADDs on general purpose processors.

As for the Kautz filter, it can be deduced from Fig. 10 that the Kautz backbone requires four MAC operations per second-order section, similarly to series biquads (except the first section that needs two MACs). Then the $(1 + z^{-1})$ and $(1 - z^{-1})$ terms need two ADDs, and the weights $b_{i,1}$ and $b_{i,2}$ and the corresponding output summing require two MAC operations. This adds up to six MACs and two ADDs, whereas on general purpose processors where MAC instructions are not available, eight ADDs and six multiplications are needed per second-order section. For the fixed-pole parallel filter, four MAC instructions, or four multiplications and ADDs, are needed, similarly to WFIR or IIR filters de-warped to series second-order sections.

For minimizing the number of arithmetic operations, the series second-order implementation of warped filters and the fixed pole parallel filter seem to be the best two contenders. However, coming from the fact that the fixed-pole parallel filter has a fully parallel structure, additional computational savings can be achieved on parallel architectures such as graphic processing units [64] compared to the series structure.

Note that for low-pass, high-pass, band-pass, and band-reject filters, the series form is preferred because the coefficient quantization has much smaller effect on its zeros than for the parallel form [21]. However, for typical transfer functions used for modeling or equalization where the exact position of the zeros is less critical, the coefficient quantization effects of the parallel implementation become comparable to the series form. On the other hand, it has been found that the parallel form has smaller round-off noise compared to the series second-order sections [55, 65].

As a result, if a filter or equalizer is designed as a WIIR filter, it is advisable to compute its poles, de-warp them, and design a fixed-pole parallel filter based on these poles, because this will lead to the lowest computational complexity and best numerical performance (the filter response remains the same, as discussed in SEC. 6.4).

Note that WFIR filters have multiple poles at λ; thus, they cannot be converted to second-order parallel sections—rather, they should be implemented in the series form or using the more numerically robust all-pass structure of Fig. 9(a).

Kautz filters should also be converted to fixed-pole parallel filters by using the equations presented in SEC. 5.5 for significantly reduced computational complexity. The only case when the Kautz backbone implementation is preferred over parallel second-order sections is adaptive filtering, like an LMS algorithm, since in that case, orthogonality leads to faster convergence [49, 50]. As a compromise between complexity and convergence, [50] presents an "orthogonal" form for the parallel filter.

7.2 Design Complexity and Accuracy

Although Figs. 8 and 13 already provide a good comparison because the same target and filter order is used, here an additional example is presented. The thick line in Fig. 14(a) displays the same smoothed loudspeaker–room response that has been modeled in Figs. 8 and 13, but now filters are being designed so that the equalized response matches a flat target displayed by thin lines in Fig. 14. It can be seen in Figs. 14(b) and 14(c) that neither a 1,000th-order FIR filter nor a 32nd-order IIR filter can equalize the response when designed in a traditional manner, e.g., by the Steiglitz-McBride algorithm (`stmcb` command in MATLAB).

An easy way of designing equalizers with logarithmic-like resolution is WFIR filter design: both the system response and target response are pre-warped with an all-pass chain, and then an FIR equalizer is estimated between these with an LS design, e.g., using the Steiglitz-McBride algorithm with a numerator order set to zero. Already this relatively simple procedure produces much better results compared with those of straightforward IIR filters, as can be seen in Fig. 14(d).

It is similarly simple to design a Kautz or parallel filter using a predetermined (e.g., logarithmic) pole set. Note that for equalizer design the scalar product cannot be computed for Kautz filters. Instead, an LS estimation needs to be done [4] exactly the same way as for parallel filters. The filter weights are computed in closed form by the LS equations in both cases as in Eq. (18), albeit with different $\mathbf{M}$ matrices [4, 59]. Fig. 14(e) shows an example with a stepwise logarithmic pole frequency set having larger pole density in the more problematic low-frequency region. Compared to WFIR filters, a more even distribution of equalization error can be observed. Therefore, as a starting point for logarithmic frequency resolution filter design, the present author suggests the use of Kautz or parallel filters with a logarithmic pole set.

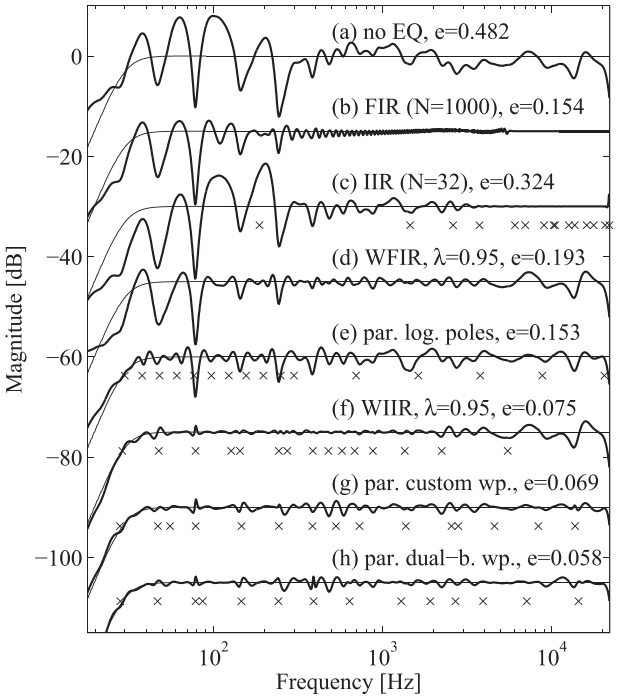

Fig. 14. Comparison of different filter design techniques for equalizing a sixth-octave smoothed minimum-phase loudspeaker–room response. Thin lines: target response. Thick lines: (a) the smoothed loudspeaker–room response equalized by a (b) finite impulse response (FIR) filter, (c) straightforward infinite impulse response (IIR) filter, (d) warped FIR filter using $\lambda = 0.95$, (e) fixed-pole parallel filter with stepwise logarithmic pole frequencies, (f) warped IIR filter using $\lambda = 0.95$, (g) custom warping, and (h) dual-band warping. The filter orders are $N = 32$ for all cases, except for the FIR filter where $N = 1{,}000$. The pole frequencies are displayed by crosses. The curves are offset for clarity. The square root of the mean squared error computed in logarithmic scale is given as e = ... for each curve. WFIR, warped FIR; WIIR, warped IIR.

Interestingly, for this particular example, neither the ripple-based pole positioning [62] nor divide-and-conquer approach [9] provides a significant benefit over the stepwise logarithmic pole set (curves not shown). As a next step in design complexity and modeling accuracy, one may choose from WIIR filters and Kautz or parallel filter designs using the poles from a WIIR filter. Especially the custom or dual-band warping techniques provide a very even distribution of modeling detail in the logarithmic scale, thus providing excellent accuracy, as already seen in Figs. 13(g) and 13(h). The performance benefit is similar for the equalizer case as well, as displayed in Figs. 14(g) and 14(h), showing a more even distribution of equalization error compared to single-band warping in Fig. 14(f).

The equalization performance could be most probably improved by post-optimization, similarly to Fig. 13(i). However, the method proposed in [63] has been developed for the modeling case, so it needs to be modified to fit the equalization task as well. Although this extension should be relatively straightforward, it is left for future work.

In any case, post-optimization significantly increases the design complexity compared with the rest of the filter design methods that typically require 10 to 100 ms in MATLAB; the design examples in Figs. 13(f) and 13(i) need 5–10 s to be computed in MATLAB on an average PC. This is still acceptable for most applications and becomes a bottleneck only when the filter needs to be redesigned in real time based on quick measurements or some user interaction, such as for a graphic equalizer [66].

Note that for simple (single-band) WIIR design and dual-band warping, the frequency response of the Kautz or parallel filter is practically the same as that of the WIIR filter(s), from where the poles originate. However, from the implementation point of view, the parallel filter will lead to lower computational complexity (compared with the special WIIR structure) or better quantization noise performance (compared with the series second-order implementation), as already discussed in SEC. 7.1.

As for choosing between Kautz or parallel filters, the resulting transfer function is the same for the same set of poles, and the design complexity is also not very much different. The LS solution of the parallel filter (especially the frequency-domain variant) is very robust and efficient; thus, the benefit of the orthonormality for the Kautz filter vanishes for non-adaptive cases. Given that Kautz filters should be converted to the parallel form for lower filtering complexity, it is probably simpler to design the filter initially in the parallel form to avoid the need of conversion.

Although this paper has presented single-channel modeling and equalization examples only, it should be mentioned that multichannel equalization or crosstalk cancellation is also possible at the logarithmic frequency scale. So far, WFIR [67, 68] and fixed-pole parallel filters [69] have been proposed for this purpose, both of which provide a computational benefit compared with straightforward FIR multiple-input–multiple-output equalizers.

8 CONCLUSION

This paper has presented an overview on warped, Kautz, and fixed-pole parallel filter design for achieving quasi-logarithmic frequency resolution desirable for audio applications. It has demonstrated that already the simplest techniques, like WFIR design or parallel filter design using a logarithmic pole set, provide significant advantages compared with straightforward FIR or IIR filter design. The performance is further improved by more sophisticated WIIR-based pole positioning methods. It is also argued that for most filter/equalizer design cases, parallel filters provide the best accuracy for a given computational complexity, whereas the orthonormality of Kautz filters is advantageous for adaptive filtering.

9 LIST OF MATLAB/OCTAVE TOOLBOXES

Warped filter design toolbox:
`http://legacy.spa.aalto.fi/software/warp/`.
Kautz filter design toolbox:
`http://legacy.spa.aalto.fi/software/kautz/kautz.htm`.

Parallel filter design toolbox:
`http://www.mit.bme.hu/~bank/parfilt/`.

10 ACKNOWLEDGMENT

This research was funded by the National Research, Development, and Innovation Fund of Hungary under Grant TKP2021-EGA-02. The author is thankful to Esteban Maestre and Panagiotis Zachos for computing the pole positions for the post-optimization and divide-and-conquer examples displayed in Fig. 13. He is also grateful to the reviewers and Prof. Julius O. Smith for their helpful comments.

11 REFERENCES

[1] M. Karjalainen, E. Piirilä, A. Järvinen, and J. Huopaniemi, "Comparison of Loudspeaker Equalization Methods Based on DSP Techniques," *J. Audio Eng. Soc.*, vol. 47, no. 1/2, pp. 14–31 (1999 Feb.).

[2] 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 (2006 Dec.).

[3] M. Karjalainen, T. Paatero, J. N. Mourjopoulos, and P. D. Hatziantoniou, "About Room Response Equalization and Dereverberation," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 183–186 (New Paltz, NY) (2005 Oct.). https://doi.org/10.1109/ASPAA.2005.1540200.

[4] M. Karjalainen and T. Paatero, "Equalization of Loudspeaker and Room Responses Using Kautz Filters: Direct Least Squares Design," *EURASIP J. Adv. Signal Process.*, vol. 2007, paper 060949 (2007). https://doi.org/10.1155/2007/60949.

[5] J. Hollebon, F. M. Fazi, and M. F. Simón Gálvez, "A Multiple Listener Crosstalk Cancellation System Using Loudspeaker Dependent Regularization," *J. Audio Eng. Soc.*, vol. 69, no. 3, pp. 191–203 (2021 Mar.). https://doi.org/10.17743/jaes.2020.0067.

[6] S. Cecchi, A. Carini, and S. Spors, "Room Response Equalization—A Review," *Appl. Sci.*, vol. 8, no. 1, paper 16 (2018 Jan.). https://doi.org/10.3390/app8010016.

[7] S. Bharitkar and C. Kyriakakis, "Loudspeaker and Room Response Modeling With Psychoacoustic Warping, Linear Prediction, and Parametric Filters," presented at the *121st Convention of the Audio Engineering Society* (2006 Oct.), paper 6982.

[8] G. Vairetti, E. De Sena, M. Catrysse, et al., "A Scalable Algorithm for Physically Motivated and Sparse Approximation of Room Impulse Responses With Orthonormal Basis Functions," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 25, no. 7, pp. 1547–1561 (2017 Jul.). https://doi.org/10.1109/TASLP.2017.2700940.

[9] G. Kamaris, P. Zachos, and J. Mourjopoulos, "Low Filter Order Digital Equalization for Mobile Device Earphones," *J. Audio Eng. Soc.*, vol. 69, no. 5, pp. 297–308 (2021 May.). https://doi.org/10.17743/jaes.2021.0003.

[10] J. Mackenzie, J. Huopaniemi, V. Välimäki, and I. Kale, "Low-Order Modeling of Head-Related Transfer Functions Using Balanced Model Truncation," *IEEE Signal Process. Lett.*, vol. 4, no. 2, pp. 39–41 (1997 Feb.). https://doi.org/10.1109/97.554467.

[11] C. J. Liu and S. F. Hsieh, "Common-Acoustic-Poles/Zeros Approximation of Head-Related Transfer Functions," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 3341–3344 (Salt Lake City, UT) (2001 May). https://doi.org/10.1109/ICASSP.2001.940374.

[12] G. Ramos, M. Cobos, B. Bank, and J. A. Belloch, "A Parallel Approach to HRTF Approximation and Interpolation Based on a Parametric Filter Model," *IEEE Signal Process. Lett.*, vol. 24, no. 10, pp. 1507–1511 (2017 Oct). https://doi.org/10.1109/LSP.2017.2741724.

[13] B. Bank and H.-M. Lehtonen, "Perception of Longitudinal Components in Piano String Vibrations," *J. Acoust. Soc. Am.*, vol. 128, no. 3, pp. EL117–EL128 (2010 Sep.). https://doi.org/10.1121/1.3453420.

[14] E. Maestre, G. P. Scavone, and J. O. Smith, "Joint Modeling of Bridge Admittance and Body Radiativity for Efficient Synthesis of String Instrument Sound by Digital Waveguides," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 25, no. 5, pp. 1128–1139 (2017 May). https://doi.org/10.1109/TASLP.2017.2689241.

[15] K. Kobayashi, D. Takeuchi, M. Iwamoto, K. Yatabe, and Y. Oikawa, "Parametric Approximation of Piano Sound Based on Kautz Model With Sparse Linear Prediction," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 626–630 (Calgary, Canada) (2018 Apr.). https://doi.org/10.1109/ICASSP.2018.8461547.

[16] P. D. Hatziantoniou and J. N. Mourjopoulos, "Generalized Fractional-Octave Smoothing of Audio and Acoustic Responses," *J. Audio Eng. Soc.*, vol. 48, no. 4, pp. 259–280 (2000 Apr.).

[17] P. G. Craven and M. A. Gerzon, "Practical Adaptive Room and Loudspeaker Equalizer for Hi-Fi Use," presented at the *92nd Convention of the Audio Engineering Society* (1992 Mar.), paper 3346.

[18] J. O. Smith and J. S. Abel, "Bark and ERB Bilinear Transform," *IEEE Trans. Speech Audio Process.*, vol. 7, no. 6, pp. 697–708 (1999 Nov.). https://doi.org/10.1109/89.799695.

[19] E. Zwicker and H. Fastl, *Psychoacoustics: Facts and Models* (Springer-Verlag, Heidelberg, Germany, 1990).

[20] T. W. Parks and C. S. Burrus, *Digital Filter Design* (Wiley, New York, NY, 1987).

[21] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing* (Prentice-Hall, Upper Saddle River, NJ, 1999), 2nd ed.

[22] K. Steiglitz and L. McBride, "A Technique for the Identification of Linear Systems," *IEEE Trans. Autom. Control*, vol. 10, no. 4, pp. 461–464 (1965 Oct.). https://doi.org/10.1109/TAC.1965.1098181.

[23] L. B. Jackson, "Frequency-Domain Steiglitz-McBride Method for Least-Squares IIR Filter De-

sign, ARMA Modeling, and Periodogram Smoothing," *IEEE Signal Process. Lett.*, vol. 15, pp. 49–52 (2008). https://doi.org/10.1109/LSP.2007.910320.

[24] III J. O. Smith, *Techniques for Digital Filter Design and System Identification With Application to the Violin*, Ph.D. thesis, Stanford University, Stanford, CA (1983 Jun.). https://ccrma.stanford.edu/papers/techniques-digital-filter-design-and-system-identification-with-application-violin.

[25] M. Waters and M. B. Sandler, "Least Squares IIR Filter Design on a Logarithmic Frequency Scale," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 1, pp. 635–638 (Chicago, IL) (1993 May). https://doi.org/10.1109/ISCAS.1993.393801.

[26] A. Härmä, M. Karjalainen, L. Savioja, et al., "Frequency-Warped Signal Processing for Audio Applications," *J. Audio Eng. Soc.*, vol. 48, no. 11, pp. 1011–1031 (2000 Nov.).

[27] T. Paatero and M. Karjalainen, "Kautz Filters and Generalized Frequency Resolution: Theory and Audio Applications," *J. Audio Eng. Soc.*, vol. 51, no. 1/2, pp. 27–44 (2003 Feb.).

[28] 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 (2013 Jan.).

[29] H. Behrends, A. von dem Knesebeck, W. Bradinal, P. Neumann, and U. Zölzer, "Automatic Equalization Using Parametric IIR Filters," *J. Audio Eng. Soc.*, vol. 59, no. 3, pp. 102–109 (2011 Mar.).

[30] G. Vairetti, E. De Sena, M. Catrysse, et al., "An Automatic Design Procedure for Low-Order IIR Parametric Equalizers," *J. Audio Eng. Soc.*, vol. 66, no. 11, pp. 935–952 (2018 Nov.). https://doi.org/10.17743/jaes.2018.0049.

[31] B. Bank and G. Ramos, "Improved Pole Positioning for Parallel Filters Based on Spectral Smoothing and Multiband Warping," *IEEE Signal Process. Lett.*, vol. 18, no. 5, pp. 299–302 (2011 Mar.). https://doi.org/10.1109/LSP.2011.2124456.

[32] A. G. Constantinides, "Spectral Transformations for Digital Filters," *Proc. Inst. Electr. Eng.*, vol. 117, pp. 1585–1590 (1970 Aug.).

[33] A. Oppenheim, D. Johnson, and K. Steighlitz, "Computation of Spectra With Unequal Resolution Using the Fast Fourier Transform," *Proc. IEEE*, vol. 59, no. 2, pp. 299–301 (1971 Feb.). https://doi.org/10.1109/PROC.1971.8146.

[34] H. W. Strube, "Linear Prediction on a Warped Frequency Scale," *J. Acoust. Soc. Am.*, vol. 68, no. 4, pp. 1071–1076 (1980 Oct.). https://doi.org/10.1121/1.384992.

[35] G. Ramos, J. J. López, and B. Pueo, "Cascaded Warped-FIR and FIR Filter Structure for Loudspeaker Equalization With Low Computational Cost Requirements," *Digit. Signal Process.*, vol. 19, no. 3, pp. 393–409 (2009 May). https://doi.org/10.1016/j.dsp.2008.01.003.

[36] M. Karjalainen and T. Paatero, "Frequency-Dependent Signal Windowing," in *Proceedings of the IEEE Workshop Applications of Signal Processing to Audio and Acoustics*, pp. 35–38 (New Paltz, NY) (2001 Oct.). https://doi.org/10.1109/ASPAA.2001.969536.

[37] D. H. Johnson, "Variable Digital Filters Having a Recursive Structure," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 27, no. 1, pp. 98–99 (1979 Feb.). https://doi.org/10.1109/TASSP.1979.1163196.

[38] K. Steiglitz, "A Note on Variable Recursive Filters," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 28, no. 1, pp. 111–112 (1980 Feb.). https://doi.org/10.1109/TASSP.1980.1163341.

[39] S. Ahuja and S. D. Roy, "Variable Digital Filters," *IEEE Trans. Circ. Syst.*, vol. 27, no. 9, pp. 836–838 (1980 Sep.). https://doi.org/10.1109/TCS.1980.1084884.

[40] M. Karjalainen, A. Härmä, and U. K. Laine, "Realizable Warped IIR Filters and Their Properties," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 2205–2208 (Munich, Germany) (1997 Apr.). https://doi.org/10.1109/ICASSP.1997.599488.

[41] M. Tyril, J. A. Pedersen, and P. Rubak, "Digital Filters for Low-Frequency Equalization," *J. Audio Eng. Soc.*, vol. 49, no. 1/2, pp. 36–43 (2001 Feb.).

[42] P. Wang, W. Ser, and M. Zhang, "A Dual-Band Equalizer for Loudspeakers," *J. Audio Eng. Soc.*, vol. 48, no. 10, pp. 917–921 (2000 Oct.).

[43] G. Ramos, J. J. López, and B. Pueo, "Combination of Warped and Linear Filter Structures for Loudspeaker Equalization," presented at the *124th Convention of the Audio Engineering Society* (2008 May), paper 7401.

[44] P. Wang, S. Wee, and Z. Ming, "Multiband Warped Filter Equalizer Design for Loudspeaker Systems," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. II913–II916 (Istanbul, Turkey) (2000 Jun.). https://doi.org/10.1109/ICASSP.2000.859109.

[45] B. Bank, "Warped IIR Filter Design With Custom Warping Profiles and Its Application to Room Response Modeling and Equalization," presented at the *130th Convention of the Audio Engineering Society* (2011 May), paper 8415.

[46] T. Oliveira e Silva, "Laguerre Filters – An Introduction," *Revista do Detua*, vol. 1, no. 3, pp. 237–248 (1995 Jan.).

[47] W. Kautz, "Transient Synthesis in the Time Domain," *Trans. IRE Prof. Group Circuit Theory*, vol. CT-1, no. 3, pp. 29–39 (1954 Sep.). https://doi.org/10.1109/TCT.1954.1083588.

[48] P. W. Broome, "Discrete Orthonormal Sequences," *J. ACM*, vol. 12, no. 2, pp. 151–168 (1965 Apr.). https://doi.org/10.1145/321264.321265.

[49] I. Salama and J. E. Cousseau, "Comparison of Orthonormal Adaptive FIR and IIR Filter Realizations," in *Proceedings of the IEEE Symposium on Advances in Digital Filtering and Signal Processing*, pp. 77–81 (Victoria, Canada) (1998 Jun.). https://doi.org/10.1109/ADFSP.1998.685699.

[50] K. Horváth and B. Bank, "Comparison of LMS-Based Adaptive Audio Filters for Identification," presented at the *148th Convention of the Audio Engineering Society* (2020 May), paper 10365.

[51] T. Paatero, M. Karjalainen, and A. Härmä, "Modeling and Equalization of Audio Systems Using Kautz Filters," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 3313–3316 (Salt Lake City, UT) (2001 Feb.). https://doi.org/10.1109/ICASSP.2001.940367.

[52] H. Brandenstein and R. Unbehauen, "Least-Squares Approximation of FIR by IIR Digital Filters," *IEEE Trans. Signal Process.*, vol. 46, no. 1, pp. 21–30 (1998 Jan.). https://doi.org/10.1109/78.651163.

[53] B. Bank, "Direct Design of Parallel Second-Order Filters for Instrument Body Modeling," in *Proceedings of the International Computer Music Conference*, vol. 1, pp. 458–465 (Copenhagen, Denmark) (2007 Aug.).

[54] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing* (Prentice-Hall, Englewood Cliffs, NJ, 1975).

[55] W. Chen, "Performance of Cascade and Parallel IIR Filters," *J. Audio Eng. Soc.*, vol. 44, no. 3, pp. 148–158 (1996 Mar.).

[56] M. Price, S. Holden, and M. Sandler, "Accurate Parallel Form Filter Synthesis," *Electron. Lett.*, vol. 32, no. 22, pp. 2066–2067 (1996 Oct.).

[57] J. O. Smith, *Introduction to Digital Filters With Audio Applications* (W3K Publishing, Stanford, CA, 2007). https://ccrma.stanford.edu/∼jos/filters/.

[58] B. Bank and J. O. Smith, "A Delayed Parallel Filter Structure With an FIR Part Having Improved Numerical Properties," presented at the *136th Convention of the Audio Engineering Society* (2014 Apr.), paper 9084.

[59] B. Bank, "Logarithmic Frequency Scale Parallel Filter Design With Complex and Magnitude-Only Specifications," *IEEE Signal Process. Lett.*, vol. 18, no. 2, pp. 138–141 (2011 Feb.). https://doi.org/10.1109/LSP.2010.2093892.

[60] B. Bank, "Magnitude-Priority Filter Design," *J. Audio Eng. Soc.*, vol. 62, no. 7/8, pp. 485–492 (2014 Jul.). https://doi.org/10.17743/jaes.2014.0030.

[61] E. Maestre, J. S. Abel, J. O. Smith, and G. P. Scavone, "Constrained Pole Optimization for Modal Reverberation," in *Proceedings of the 20th International Conference on Digital Audio Effects*, pp. 381–388 (Edinburgh, UK) (2017 Sep.).

[62] B. Bank, "Loudspeaker and Room Response Equalization Using Parallel Filters: Comparison of Pole Positioning Strategies," in *Proceedings of the AES 51st International Conference: Loudspeakers and Headphones* (2013 Aug.), paper 1-4.

[63] E. Maestre, G. P. Scavone, and J. O. Smith, "Design of Recursive Digital Filters in Parallel Form by Linearly Constrained Pole Optimization," *IEEE Signal Process. Lett.*, vol. 23, no. 11, pp. 1547–1550 (2016 Nov.). https://doi.org/10.1109/LSP.2016.2605626.

[64] J. A. Belloch, B. Bank, L. Savioja, A. Gonzalez, and V. Välimäki, "Multi-Channel IIR Filtering of Audio Signals Using a GPU," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 6692–6696 (Florence, Italy) (2014 May). https://doi.org/10.1109/ICASSP.2014.6854895.

[65] B. Bank and K. Horváth, "Quantization Noise of Warped and Parallel Filters Using Floating Point Arithmetic," presented at the *142nd Convention of the Audio Engineering Society* (2017 May), e-Brief 337.

[66] 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 (2014 Dec.). https://doi.org/10.1109/TASLP.2014.2354241.

[67] O. Kirkeby, P. Rubak, L. G. Johansen, and P. A. Nelson, "Implementation of Cross-Talk Cancellation Networks Using Warped FIR Filters," in *Proceedings of the AES 16th International Conference: Spatial Sound Reproduction*, (1999 Mar.), paper 16-031.

[68] J. Jeong, J. Lee, Y. Park, J.-T. Kim, and D. Youn, "Design and Implementation of IIR Crosstalk Cancellation Filters Approximating Frequency Warping," presented at the *118th Convention of the Audio Engineering Society* (2005 May), paper 6490.

[69] B. Bank, "Multichannel Equalization and Crosstalk Cancellation Using Fixed-Pole IIR Filters," *J. Audio Eng. Soc.*, vol. 66, no. 11, pp. 901–909 (2018 Nov.).

THE AUTHOR

Balázs Bank

Balázs Bank is an associate professor at the Department of Measurement and Information Systems, Budapest University of Technology and Economics (BUTE). He received his M.Sc. and Ph.D. degrees in Electrical Engineering from BUTE, Hungary, in 2000 and 2006, respectively. 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 research assistant at BUTE. In 2001 he visited the Department of Information Engineering, University of Padova. In 2007 he returned to the Acoustics Laboratory of Helsinki University of Technology for a year, and in 2008 he was with the Department of Computer Science, Verona University, Italy. Since 2009 he has been with the BUTE. He was an Associate Editor for *IEEE Signal Processing Letters* in 2013–2016 and *IEEE Signal Processing Magazine* in 2018–2022. He is a Guest Editor for the current *JAES* special issue "Audio Filter Design." His research interests include physics-based sound synthesis and filter design for audio applications. He is an AES and IEEE member.