International Journal of Innovative
Computing, Information and Control
Volume **21**, Number **4**, August **2025**

 ISSN 1349-4198
pp. **1025–1038**

STABILITY-GUARANTEED TUNABLE FILTER DESIGN EXPLOITING NOVEL CASCADE STRUCTURE WITH LP-ERROR MINIMIZATION

Tian-Bo Deng

Faculty of Science
Toho University
Miyama 2-2-1, Funabashi, Chiba 274-8510, Japan
deng@is.sci.toho-u.ac.jp

Received February 2025; revised May 2025

Abstract. *This paper reveals a novel cascade structure for implementing and designing even-order tunable filters. The novel structure cascades second-order (2nd-order) building blocks, and it offers high modularity and features low quantization errors in hardware implementations. Based on this novel cascade structure, an Lp-error-minimization strategy is developed for designing the even-order cascade-structured tunable filter. To grapple with the stability guarantee of the cascade structure, the paper further reveals a new stabilizing function used to convert the parameters in the building block's denominator to the stabilizing functions so that the stability is definitely guaranteed. A tunable highpass-filter exploiting the novel structure is designed for exemplifying the Lp-error-minimization strategy and validating the stabilizing-function-based stability-guarantee. The highpass design simulations demonstrate that the designed highpass tunable filter is extremely accurate, whose average Lp-error is just 0.000012658. Furthermore, the resulting highpass tunable filter has the maximum pole radius 0.9588, which is much less than unity. Hence, the highpass tunable filter also exhibits a sufficiently large stability margin.*
Keywords: Cascade structure, Even-order tunable filter, Lp-error minimization, Stability, Stabilizing function

1. **Introduction.** A tunable filter can instantly change frequency-domain characteristics, including phase response or gain (amplitude) response, or even both phase and gain responses [1-10]. Thanks to the high flexibility, there is no necessity for the filter user to start redesigning a new filter all over again when some frequency-domain characteristics need to be updated. This is achieved by updating filter's parameters which are usually represented as the functions of tuning parameters, where tuning parameters denote those used to tune the frequency-domain characteristics. Thus far, a great variety of schemes have been proposed, aiming to design and implement both non-recursive and recursive tunable filters. In particular, a stability-guarantee measure must be taken seriously to prevent the instability of recursive tunable filters [11-14]. This is because a recursive tunable filter may fall into instability when the filter coefficients are suddenly updated. The updated coefficient values may no longer meet the stability condition [15], which results in an unstable tunable filter. Therefore, it is tremendously important to make sure that the stability is always guaranteed during the tuning of frequency-domain characteristics.

As far as the filter structure is concerned, most existing tunable filter designs utilize direct-form models (transfer functions). However, such direct-form transfer functions greatly suffer quantization errors (round-off noises) when they are implemented using hardware [15]. Hence, it is preferable to employ a structure that mitigates round-off

DOI: 10.24507/ijicic.21.04.1025

noises in hardware implementations. Moreover, it is preferable to utilize a filter structure that has high modularity, which can be implemented by using the same basic modules (building blocks). Such a modularity facilitates hardware implementations.

This paper aims to present several novel schemes that contribute to optimally designing and efficiently implementing a recursive tunable filter.

- This paper reveals a novel cascade structure for designing and implementing a tunable filter with recursive structure. As compared to the existing works that exploits the direct-form structure [16, 17], this novel configuration cascades only the second-order (2nd-order) modules (blocks), which features high modularity, and thus eases the filter implementations. Furthermore, employing this cascade structure to implement the tunable filter mitigates the quantization errors induced in hardware implementations [15].
- This cascade structure contains the 2nd-order blocks as its basic building blocks, which facilitates hardware implementations. However, those basic blocks involve feedback loops and may become unstable. To ensure that the whole cascade system is stable, it is mandatory to guarantee that every single 2nd-order block is stable. To fulfill this objective, the paper develops a novel stabilizing function for guaranteeing the stability of the 2nd-order blocks. The core target is to utilize the stabilizing function to express the blocks' denominator coefficients, and those expressions absolutely satisfy the stability requirement. Therefore, this stabilizing function plays a critical role in guaranteeing the stability.
- The paper develops an Lp-error-minimization solution to design an even-order tunable filter using the cascade structure. The Lp-error-minimization strategy utilizes nonlinear programming to minimize the Lp-errors, leading to an even-order Lp-error-minimized tunable filter. Combining the stability-guarantee strategy with the Lp-error-minimization produces an even-order cascade-structured tunable filter with definitely guaranteed stability.

Based on the novel cascade structure, this paper showcases the design of a tunable highpass-filter employing the Lp-error-minimization alongside the stabilizing function for stability-guarantee. As compared with the design exploiting the direct-type structure [16, 17], the highpass design simulations demonstrate that using the novel cascade structure can produce an Lp-error-minimized tunable filter, which features high modularity and absolutely guaranteed stability.

2. **Tunable Filter Employing Novel Cascade Structure.**

The ultimate target of designing a tunable filter aims to achieve the best approximation of a prescribed ideal gain response $G(\omega, \nu)$, where ω means frequency. Also, the included parameter ν tunes $G(\omega, \nu)$. That is, the gain specification $G(\omega, \nu)$ changes as ν varies. Therefore, $G(\omega, \nu)$ represents an ideal tunable gain response. To approximate $G(\omega, \nu)$, a frequency-domain mathematical model called transfer function is utilized. This paper presents a novel cascade structure used for modelling the tunable filter. This section first details the novel cascade structure, and then reveals the Lp-error-minimization methodology for approximating the tunable specification $G(\omega, \nu)$ based on the cascade structure. Moreover, a new stabilizing function is developed, which is incorporated into the Lp-error-minimization scheme so as to guarantee the stability of the cascade structure.

2.1. **Novel fixed cascade structure and stability.**

Before investigating the design of a tunable filter for approximating the tunable gain specification $G(\omega, \nu)$, we first begin with presenting the novel cascade structure

$$H(z) = \prod_{k=1}^{K} B_k(z) \tag{1}$$

that cascades the 2nd-order modules (building blocks)

$$B_k(z) = \frac{b_{k0} + b_{k1}z^{-1} + b_{k2}z^{-2}}{1 + q_{k1}z^{-1} + q_{k2}z^{-2}} \tag{2}$$

with

$$\begin{aligned} &b_{k0} \neq 1, \text{ for } k = 1 \\ &b_{k0} = 1, \text{ for } k > 1. \end{aligned} \tag{3}$$

The cascade-form transfer function (mathematical model) $H(z)$ involves the 2nd-order building blocks $B_k(z)$ that have unchangeable coefficients. Thus, the whole cascade structure $H(z)$ holds unchangeable frequency-domain characteristics. The 2nd-order blocks $B_k(z)$ contain feedback paths, and those feedback paths may cause instability. Therefore, it is necessary to ensure the stability of all the blocks $B_k(z)$. As long as $B_k(z)$ are designed to be stable, the whole cascade configuration of $H(z)$ is also stable. Indeed, the stability of $B_k(z)$ only rests on their denominator coefficients.

To render $H(z)$ stable, we must constrain the $B_k(z)$'s denominator coefficients to satisfy

$$\begin{cases} |q_{k2}| < 1 \\ |q_{k1}| < 1 + q_{k2}. \end{cases} \tag{4}$$

This condition requires that (q_{k1}, q_{k2}) must be constrained within the triangle, which is dubbed the stability triangle [15]. As noted above, only the coefficients q_{k2} and q_{k1} impact $B_k(z)$'s stability.

To design the cascade filter $H(z)$ in Equation (1), it is necessary to determine the coefficients of all the building blocks $B_k(z)$. That is, we must optimize the coefficients

$$\begin{aligned} \boldsymbol{b} &= \begin{bmatrix} b_{10} & b_{11} & b_{12} & b_{21} & b_{22} & \cdots & b_{K1} & b_{K2} \end{bmatrix}^T \\ \boldsymbol{q} &= \begin{bmatrix} q_{11} & q_{12} & q_{21} & q_{22} & \cdots & q_{K1} & q_{K2} \end{bmatrix}^T \end{aligned} \tag{5}$$

so as to approximate a prescribed ideal response adequately. However, it must be noted that optimizing coefficients directly is risky. This is because the direct optimization may produce the coefficients that violate the condition in Equation (4). To fix this problem, this paper employs a parameter-conversion scheme in ensuring the stability. More specifically, the technique utilizes the stabilizing function

$$f(x) = \begin{cases} \sin(\lambda x), & |\lambda x| < \pi/2 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

to express the parameters q_{k2} and q_{k1} as

$$\begin{cases} q_{k2} = f(x_{k2}) \\ q_{k1} = f(x_{k1})(1 + q_{k2}) = f(x_{k1})[1 + f(x_{k2})]. \end{cases} \tag{7}$$

Here, x_{k2} and x_{k1} are the variables that are newly introduced to replace the original variables q_{k2} and q_{k1} through using the stabilizing function $f(x)$. In other words, q_{k2} and q_{k1} are replaced as the functions of x_{k2} and x_{k1}. As a consequence, the new knowns x_{k2} and x_{k1} need to be determined. To replace q_{k2} and q_{k1}, we use $f(x)$ in Equation (6), and Figure 1 shows the stabilizing function $f(x)$ with $\lambda = 0.1$.

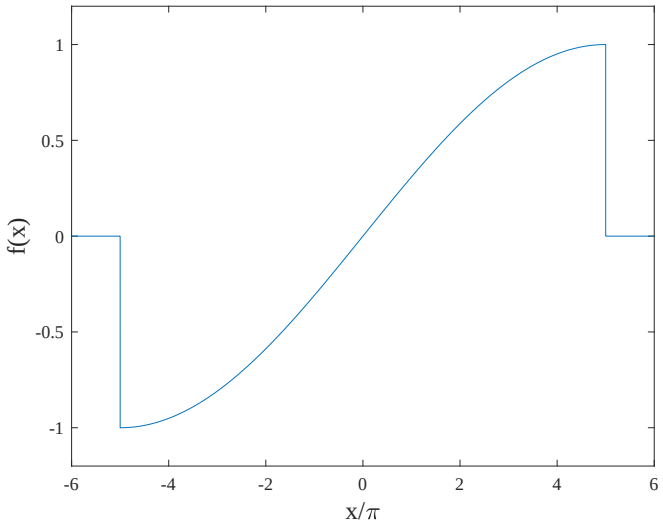

FIGURE 1. Stabilizing function

As shown below, we can justify the rationale for using the expressions in Equation (7) to ensure the condition in Equation (4) satisfied. By taking the absolute values of the expressions in Equation (7), we can simply confirm

$$\begin{aligned} |q_{k2}| &= |f(x_{k2})| < 1 \\ |q_{k1}| &= |f(x_{k1})(1+q_{k2})| = |f(x_{k1})|(1+q_{k2}) < 1+q_{k2} \end{aligned} \tag{8}$$

for any variables x_{k2} and x_{k1}. Thus, the expressions in Equation (7) justify that no constraints need to be imposed on $\{x_{k1}, x_{k2}\}$. In other words, no values of $\{x_{k1}, x_{k2}\}$ violate the condition in Equation (4).

By using the parameter expressions in Equation (7), we can replace $B_k(z)$ in Equation (2) as

$$B_k(z) = \frac{b_{k0} + b_{k1}z^{-1} + b_{k2}z^{-2}}{1 + f(x_{k1})[1 + f(x_{k2})]z^{-1} + f(x_{k2})z^{-2}}. \tag{9}$$

The blocks in Equation (9) contain new variables $\{x_{k1}, x_{k2}\}$, and thus the overall configuration $H(z)$ in Equation (1) contains the parameters

$$\begin{aligned} \boldsymbol{b} &= \begin{bmatrix} b_{10} & b_{11} & b_{12} & b_{21} & b_{22} & \cdots & b_{K1} & b_{K2} \end{bmatrix}^T \\ \boldsymbol{x} &= \begin{bmatrix} x_{11} & x_{12} & x_{21} & x_{22} & \cdots & x_{K1} & x_{K2} \end{bmatrix}^T. \end{aligned} \tag{10}$$

It should be noted that the tunable filter design needs to find the optimal values of those parameters. For a specific value of ν, $G(\omega, \nu)$ defines a specific ideal gain response. Such a specification can be approximated by using the actual gain response of $H(z)$, where the converted expressions in Equation (9) are adopted for ensuring the stability. The approximation requires optimizing the parameters in Equation (10). That is, after replacing the parameters as Equation (7), we simply need to determine the coefficients in Equation (10). By exploiting the stabilizing function $f(x)$ to convert the parameters as Equation (7), we can optimize the parameters in Equation (10) without concerning the stability.

The above technique plays a critical role in obtaining a recursive tunable filter with ensured stability. The details on designing a stabilized recursive tunable filter using the novel structure are given in the next subsection, where the cascade-structured tunable filter holds the coefficients being the functions of ν. Generally speaking, when ν varies, the coefficient values also vary, which may induce instability. To grapple with this problem,

the next subsection adopts the aforementioned idea to produce a stability-guaranteed cascade-structured tunable filter. The resulting denominator coefficients reduce to more complicated functions of ν after $f(x)$ is employed to express the denominator parameters of the tunable building blocks.

2.2. **Tunable cascade model and stability.** The fixed cascade model $H(z)$ in Equation (1) can be generalized to the tunable one as

$$H(z,\nu)=\prod_{k=1}^{K}B_k(z,\nu) \tag{11}$$

where

$$B_k(z,\nu)=\frac{b_{k0}(\nu)+b_{k1}(\nu)z^{-1}+b_{k2}(\nu)z^{-2}}{1+q_{k1}(\nu)z^{-1}+q_{k2}(\nu)z^{-2}} \tag{12}$$

are tunable 2nd-order blocks with the coefficients expressed as the functions of ν. Similar to the fixed cascade model, the coefficients $b_{k0}(\nu)$ feature

$$b_{k0}(\nu)=\begin{cases}\text{function of }\nu, & \text{for } k=1\\ 1, & \text{for } k>1.\end{cases} \tag{13}$$

To design the tunable filter $H(z,\nu)$ whose gain response approaches the ideal $G(\omega,\nu)$, we must find the changeable coefficients

$$\begin{aligned}\boldsymbol{b}(\nu)&=\begin{bmatrix}b_{10}(\nu) & b_{11}(\nu) & b_{12}(\nu) & b_{21}(\nu) & b_{22}(\nu) & \cdots & b_{K1}(\nu) & b_{K2}(\nu)\end{bmatrix}^T\\ \boldsymbol{q}(\nu)&=\begin{bmatrix}q_{11}(\nu) & q_{12}(\nu) & q_{21}(\nu) & q_{22}(\nu) & \cdots & q_{K1}(\nu) & q_{K2}(\nu)\end{bmatrix}^T\end{aligned} \tag{14}$$

that are the functions of ν. For the 2nd-order tunable blocks $B_k(z,\nu)$ to be stable, $q_{k2}(\nu)$ and $q_{k1}(\nu)$ must fulfill the requirement

$$\begin{cases}|q_{k2}(\nu)|<1\\ |q_{k1}(\nu)|<1+q_{k2}(\nu)\end{cases} \tag{15}$$

for every single value of ν. That is, as ν varies, $q_{k2}(\nu)$ and $q_{k1}(\nu)$ must meet the above condition. The next subsection reveals a 2-stage procedure that expresses $q_{k2}(\nu)$ and $q_{k1}(\nu)$ as the functions of ν so as to render the 2nd-order tunable blocks $B_k(z,\nu)$ always stable.

2.3. **Two-stage procedure and stability guarantee.** This subsecton details how to find the coefficients in Equation (14), while the resulting $q_{k1}(\nu)$, $q_{k2}(\nu)$ ensure the stability of $B_k(z,\nu)$. The two stages include designing a series of cascade-structured fixed filters and then fitting polynomials to the optimized parameter values of the fixed filters.

Stage-1: Evenly discretize the parameter $\nu\in[\nu_{\min},\nu_{\max}]$ and get L equidistant samples ν_l, $l=1,2,\ldots,L$. Each sample ν_l leads to the sampled gain-specification $G(\omega,\nu_l)$. For a specific ν_l, we design a cascade-structured filter

$$H^{(l)}(z)=\prod_{k=1}^{K}B_k^{(l)}(z) \tag{16}$$

with

$$\begin{aligned}B_k^{(l)}(z)&=\frac{b_{k0}+b_{k1}z^{-1}+b_{k2}z^{-2}}{1+q_{k1}z^{-1}+q_{k2}z^{-2}}\\ &=\frac{b_{k0}+b_{k1}z^{-1}+b_{k2}z^{-2}}{1+f(x_{k1})[1+f(x_{k2})]z^{-1}+f(x_{k2})z^{-2}}\end{aligned} \tag{17}$$

whose gain approximates $G(\omega, \nu_l)$. In Equation (17), the denominator coefficients use the conversions in Equation (7), rendering $H^{(l)}(z)$ stable. The approximation of each $G(\omega, \nu_l)$ requires optimizing the coefficients in Equation (10) such that the approximation errors are as small as possible. In this paper, the Lp-error is utilized as the metric to measure the approximation errors. That is, the design seeks to optimize the coefficients in Equation (10) such that the Lp-error

$$\mathcal{E}_p(\boldsymbol{b}, \boldsymbol{x}) = \left\{ \sum_{m=1}^{M} w(\omega_m) |e(\omega_m)|^p \right\}^{1/p} \tag{18}$$

is minimized, where

$$e(\omega_m) = G(\omega_m, \nu_l) - H^{(l)}(\omega_m) \tag{19}$$

represents the gain-response error, $H^{(l)}(\omega_m)$ represents the gain response of $H^{(l)}(z)$ at frequency ω_m, and the factor

$$w(\omega_m) = \begin{cases} 1, & \omega_m \in \text{passband and stopband} \\ 0, & \omega_m \in \text{transition band} \end{cases} \tag{20}$$

weights the gain-response error $e(\omega_m)$. It should be emphasized that minimizing $\mathcal{E}_p(\boldsymbol{b}, \boldsymbol{x})$ requires solving a nonlinear minimization problem. In this paper, we exploit the minimization solver *fminsearch* in MATLAB to carry out the minimization.

By designing $H^{(l)}(z)$ to approximate $G(\omega, \nu_l)$ one after another, we yield a set of $\{\boldsymbol{b}, \boldsymbol{x}\}$. The resulting $\{\boldsymbol{b}, \boldsymbol{x}\}$ are summarized to two matrices

$$\mathcal{B} = \begin{bmatrix} b_{10}^{(1)} & b_{11}^{(1)} & b_{12}^{(1)} & b_{21}^{(1)} & b_{22}^{(1)} & \cdots & b_{K1}^{(1)} & b_{K2}^{(1)} \\ b_{10}^{(2)} & b_{11}^{(2)} & b_{12}^{(2)} & b_{21}^{(2)} & b_{22}^{(2)} & \cdots & b_{K1}^{(2)} & b_{K2}^{(2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ b_{10}^{(L)} & b_{11}^{(L)} & b_{12}^{(L)} & b_{21}^{(L)} & b_{22}^{(L)} & \cdots & b_{K1}^{(L)} & b_{K2}^{(L)} \end{bmatrix} \tag{21}$$

and

$$\mathcal{X} = \begin{bmatrix} x_{11}^{(1)} & x_{12}^{(1)} & x_{21}^{(1)} & x_{22}^{(1)} & x_{31}^{(1)} & x_{32}^{(1)} & \cdots & x_{K1}^{(1)} & x_{K2}^{(1)} \\ x_{11}^{(2)} & x_{12}^{(2)} & x_{21}^{(2)} & x_{22}^{(2)} & x_{31}^{(2)} & x_{32}^{(2)} & \cdots & x_{K1}^{(2)} & x_{K2}^{(2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ x_{11}^{(L)} & x_{12}^{(L)} & x_{21}^{(L)} & x_{22}^{(L)} & x_{31}^{(L)} & x_{32}^{(L)} & \cdots & x_{K1}^{(L)} & x_{K2}^{(L)} \end{bmatrix}. \tag{22}$$

In Equations (21) and (22), the l-th rows denote the optimized values of $\{\boldsymbol{b}, \boldsymbol{x}\}$ for $H^{(l)}(z)$. That is, those values represent the parameter values of $H^{(l)}(z)$, $l = 1, 2, \ldots, L$, and the gain response of $H^{(l)}(z)$ fits $G(\omega, \nu_l)$. For example, the first rows of $\mathcal{B}$, $\mathcal{X}$ represent the optimized $\{\boldsymbol{b}, \boldsymbol{x}\}$ respectively for $H^{(1)}(z)$ whose gain response fits $G(\omega, \nu_1)$.

Stage-2: After the matrices in Equations (21) and (22) are obtained, an individual polynomial is fitted to each column by minimizing the sum of squared errors. This reduces to the least-squares curve fitting, leading to the fitting polynomials

$$\begin{aligned} \boldsymbol{b}(\nu) &= \begin{bmatrix} b_{10}(\nu) & b_{11}(\nu) & b_{12}(\nu) & b_{21}(\nu) & b_{22}(\nu) & \cdots & b_{K1}(\nu) & b_{K2}(\nu) \end{bmatrix}^T \\ \boldsymbol{x}(\nu) &= \begin{bmatrix} x_{11}(\nu) & x_{12}(\nu) & x_{21}(\nu) & x_{22}(\nu) & x_{31}(\nu) & x_{32}(\nu) & \cdots & x_{K1}(\nu) & x_{K2}(\nu) \end{bmatrix}^T. \end{aligned} \tag{23}$$

After the curve fitting, $q_{k2}(\nu)$ and $q_{k1}(\nu)$ are produced using the conversions

$$\begin{cases} q_{k2}(\nu) = f(x_{k2}(\nu)) \\ q_{k1}(\nu) = f(x_{k1}(\nu))[1 + q_{k2}(\nu)] = f(x_{k1}(\nu))[1 + f(x_{k2}(\nu))]. \end{cases} \tag{24}$$

Ultimately, we yield the tunable-filter coefficients

$$\boldsymbol{b}(\nu) = \begin{bmatrix} b_{10}(\nu) & b_{11}(\nu) & b_{12}(\nu) & b_{21}(\nu) & b_{22}(\nu) & \cdots & b_{K1}(\nu) & b_{K2}(\nu) \end{bmatrix}^T$$

$$\boldsymbol{q}(\nu) = \begin{bmatrix} q_{11}(\nu) & q_{12}(\nu) & q_{21}(\nu) & q_{22}(\nu) & q_{31}(\nu) & q_{32}(\nu) & \cdots & q_{K1}(\nu) & q_{K2}(\nu) \end{bmatrix}^T.$$

In the above functions, although $x_{k2}(\nu)$ and $x_{k1}(\nu)$ are polynomials in ν, $q_{k2}(\nu)$ and $q_{k1}(\nu)$ are no longer polynomials. Instead, they become more complicated functions of ν. As justifying in Equation (8), we can confirm in a similar way that arbitrary polynomials $x_{k1}(\nu)$ and $x_{k2}(\nu)$ render the tunable-filter $H(z, \nu)$ in Equation (11) stable, regardless of the value of ν.

To evaluate the minimization accuracy, the average Lp-error (Lp-error per frequency sample) is used as the metric, which is defined by

$$\begin{aligned} \overline{\mathcal{E}}_p(\boldsymbol{b}, \boldsymbol{x}) &= \frac{\mathcal{E}_p(\boldsymbol{b}, \boldsymbol{x})}{M} \\ &= \frac{\left\{ \sum_{m=1}^{M} w(\omega_m) |e(\omega_m)|^p \right\}^{1/p}}{M} \end{aligned} \tag{25}$$

with $e(\omega_m)$ being the gain-response error defined in Equation (19).

3. **Tunable Highpass-Filter Simulations.** Let us consider the approximation of the tunable highpass gain-specification

$$G(\omega, \nu) = \begin{cases} 0 & \omega \in [0, \omega_s] \\ \dfrac{\omega - \omega_s}{\omega_p - \omega_s} & \omega \in [\omega_s, \omega_p] \\ 1 & \omega \in [\omega_p, \pi] \end{cases} \tag{26}$$

with

$$\begin{aligned} &\omega_p = 0.50\pi + \nu \\ &\omega_s = \omega_p - 0.05\pi \\ &\nu \in [-0.20\pi, 0.20\pi]. \end{aligned}$$

In Equation (26), ω_p denotes passband edge-frequency, while ω_s denotes stopband edge-frequency. Clearly, both ω_p and ω_s are tunable, while the transition-band keeps its bandwidth unchanged (0.05π).

By following the 2-stage procedure, we first get L equidistant samples from the interval $\nu \in [-0.20\pi, 0.20\pi]$ with $L = 21$. Figure 2 shows the corresponding highpass specifications $G(\omega, \nu_l)$. Then, those highpass specifications are individually approximated by using 21 cascade-structured filters $H^{(l)}(z)$. The design parameters are

$$K = 3, \quad p = 20, \quad M = 1001. \tag{27}$$

Here, $K = 3$ means that the filter order is 6, while $p = 20$ is employed in minimizing the error $\mathcal{E}_p(\boldsymbol{b}, \boldsymbol{x})$ in Equation (18). Moreover, $M = 1001$ denotes the number of discretized frequencies. Considering the filter complexity as well as design accuracy, the basic idea to select K is that the filter order ($2K$) should be chosen as low as possible, but without significantly degrading the approximation accuracy. The design simulations given below

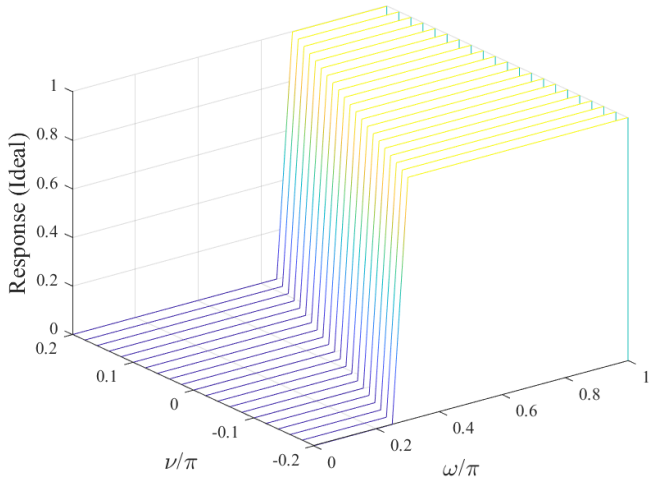

FIGURE 2. Discretized tunable highpass specifications

justify that choosing $K = 3$ can yield a sufficiently accurate highpass design. Moreover, the stabilizing function $f(x)$ in Equation (6) is employed with $\lambda = 0.1$.

The parameter values for initiating designing the first highpass-filter $H^{(1)}(z)$ are

$$\begin{bmatrix} b_{10} \\ b_{11} \\ b_{12} \\ b_{21} \\ b_{22} \\ b_{31} \\ b_{32} \end{bmatrix} = \begin{bmatrix} -0.147201456151267 \\ 1.007773405305439 \\ -2.123655462415750 \\ -0.504586405514010 \\ -1.270594449808660 \\ -0.382584802707648 \\ 0.648679262048621 \end{bmatrix} \tag{28}$$

$$\begin{bmatrix} x_{12} \\ x_{11} \\ x_{22} \\ x_{21} \\ x_{32} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 0.825727149241758 \\ -1.014943642680137 \\ -0.471069912683167 \\ 0.137024874130050 \\ -0.291863375753573 \\ 0.301818555261006 \end{bmatrix}. \tag{29}$$

This nonlinear minimization ends if the relative Lp-error is smaller than 10^{-5}. After $H^{(1)}(z)$ is produced, the optimized parameter values in Equations (28) and (29) are exploited to initiate the design of the next one $H^{(2)}(z)$. The same iteration continues to the end of designing the last $H^{(L)}(z)$.

Figure 3 depicts the gain responses of the obtained 21 fixed highpass-filters $H^{(l)}(z)$, and Figure 4 illustrates the trajectories (q_{k1}, q_{k2}) for the three building blocks $B_1^{(l)}(z)$, $B_2^{(l)}(z)$, and $B_3^{(l)}(z)$, where the trajectories move as ν_l varies. Figure 4 verifies that those trajectories always remain inside the triangles, which implies that the parameter expressions in Equation (7) constrain $H^{(l)}(z)$ to remain stable. That is, utilizing the stabilizing function $f(x)$ to express the parameters in Equation (7) renders the blocks $B_k^{(l)}(z)$ stable.

To measure the minimization accuracy, we use the average Lp-error defined in Equation (25). The design simulations show that the average Lp-error is

$$\overline{\mathcal{E}}_p(\boldsymbol{b}, \boldsymbol{x}) = 0.000011736. \tag{30}$$

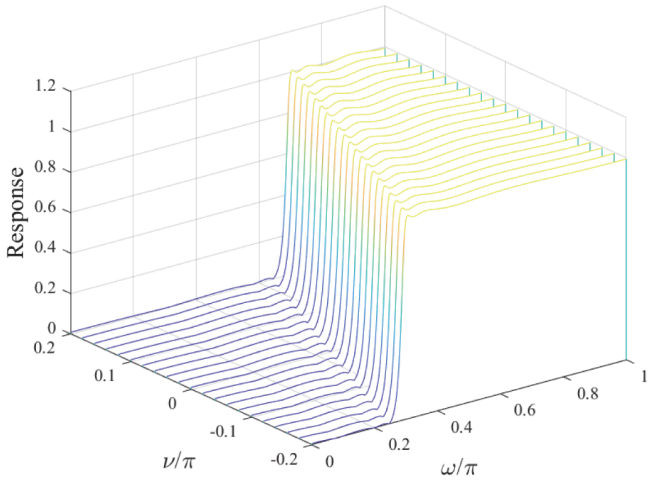

FIGURE 3. Responses of the fixed highpass-filters $H^{(l)}(z)$

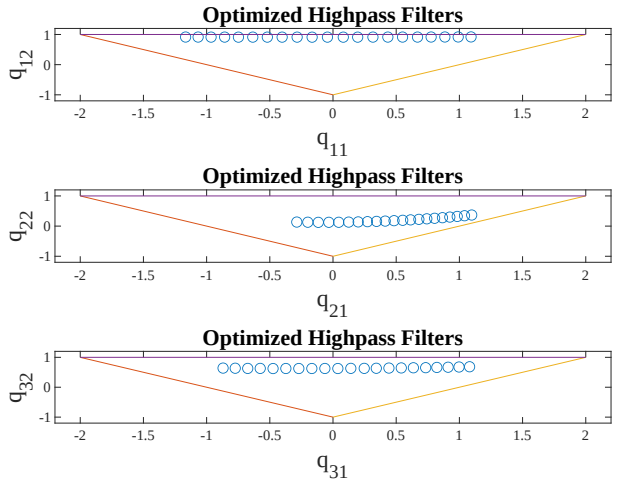

FIGURE 4. Stability check for the fixed highpass-filters $H^{(l)}(z)$

To get the fitting polynomials in Equation (23), we employ the fourth-order polynomials in fitting the parameter values in Equations (21) and (22). Figure 5 and Figure 6 show those polynomials, respectively. Those polynomials directly relate to the coefficients (14) of the cascade-structured tunable highpass-filter $H(z, \nu)$.

Figure 7 draws the tunable highpass responses of $H(z, \nu)$, where more samples of $\nu \in [-0.20\pi, 0.20\pi]$ are taken, ν_l $(l = 1, 2, \ldots, 41)$. To assess the approximation accuracy, we also plot the corresponding highpass specifications in Figure 8. The differences between Figure 7 and Figure 8 signify the highpass-response errors (unweighted errors) as plotted in Figure 9. Furthermore, Figure 10 draws the design errors after those in Figure 9 are weighted by $w(\omega)$. In computing the weighted errors, the errors in transition-bands are ignored (regarded as zero). This is because the frequency-domain characteristics in transition bands are insignificant. The average Lp-error of the weighted errors shown in Figure 10 is

$$\overline{\mathcal{E}}_p(\boldsymbol{b}, \boldsymbol{x}) = 0.000012658. \tag{31}$$

It is clear that the average Lp-error in Equation (30) and that in Equation (31) are nearly the same. This justifies that using polynomial fitting to get the coefficients of the cascade-type tunable highpass-filter $H(z, \nu)$ causes negligent (insignificant) deterioration of the approximation accuracy.

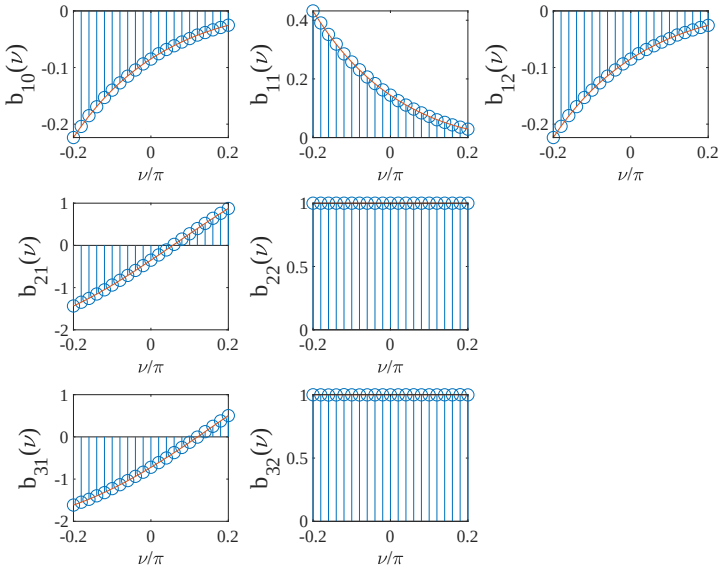

FIGURE 5. Polynomials for fitting the numerator parameters

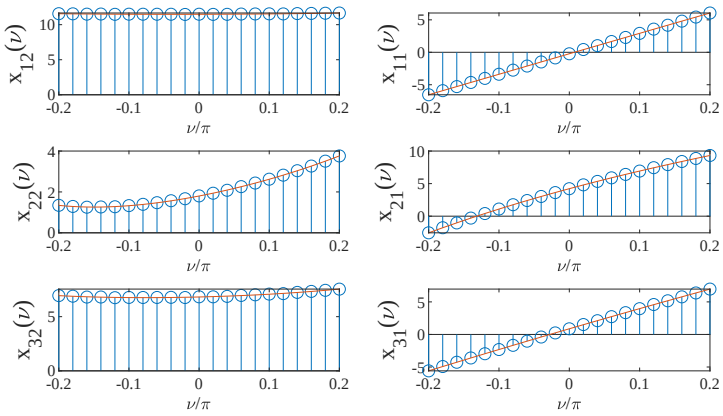

FIGURE 6. Polynomials for fitting the denominator parameters

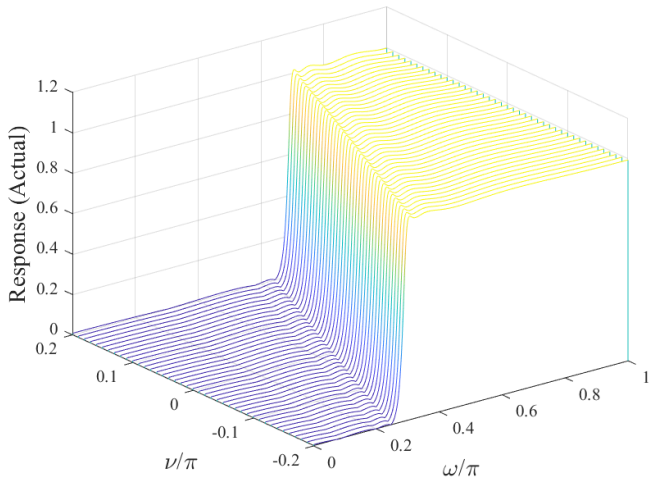

FIGURE 7. Highpass responses of the tunable highpass-filter $H(z, \nu)$

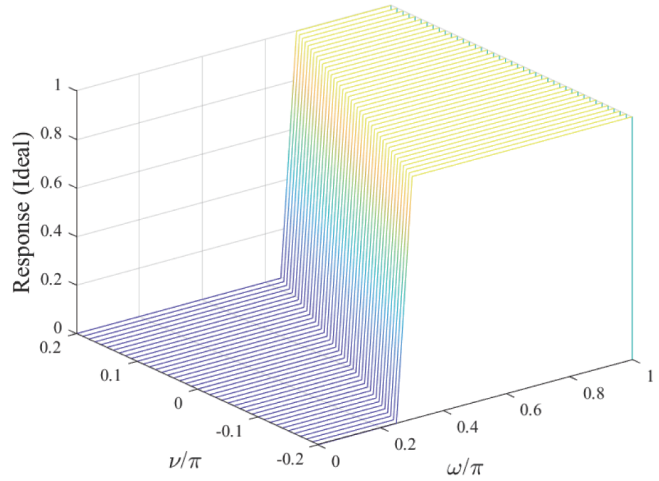

FIGURE 8. Desired tunable highpass responses

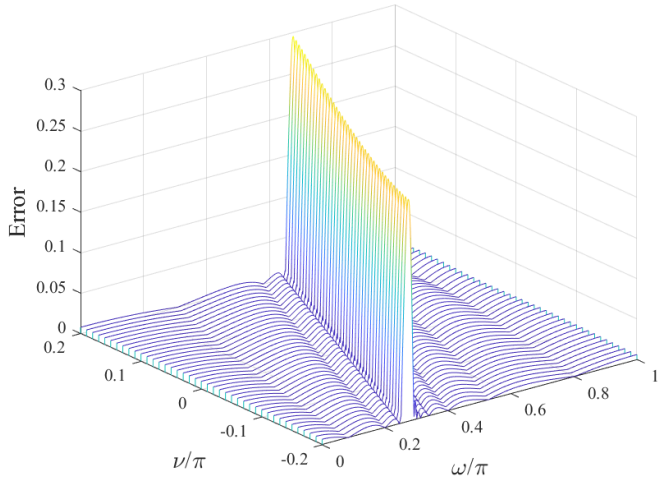

FIGURE 9. Highpass-response errors (unweighted)

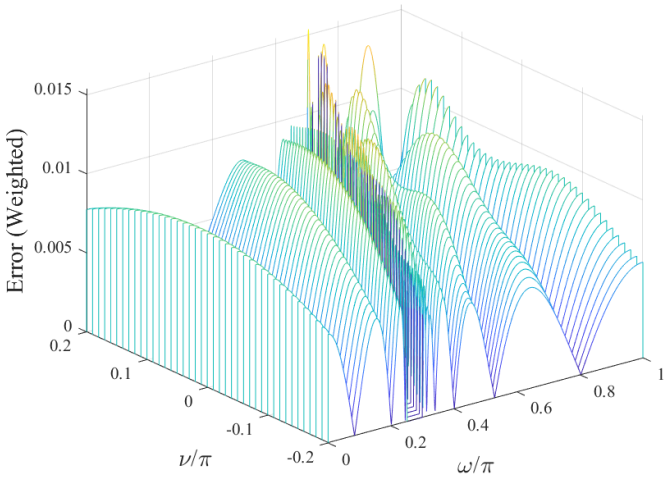

FIGURE 10. Highpass-response errors (weighted)

To check whether or not the tunable highpass-filter $H(z,\nu)$ is stable, Figure 11 details the trajectories $(q_{k1}(\nu), q_{k2}(\nu))$ for the three blocks $B_1(z,\nu)$, $B_2(z,\nu)$, and $B_3(z,\nu)$. Figure 11 shows that all the trajectories keep moving within the triangles. This validates that using the stabilizing function $f(x)$ to convert the polynomials $x_{k2}(\nu)$, $x_{k2}(\nu)$ to $q_{k2}(\nu)$, $q_{k2}(\nu)$ renders the tunable highpass-filter $H(z,\nu)$ always stable. That is, the resultant tunable highpass-filter $H(z,\nu)$ never becomes unstable with changes in ν.

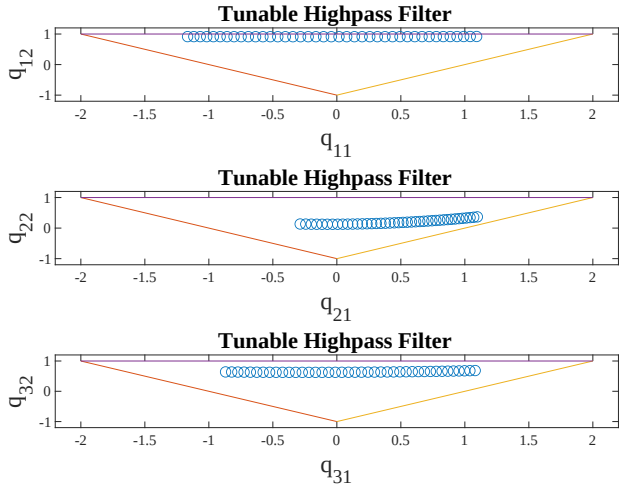

FIGURE 11. Stability triangles for the tunable highpass-filter $H(z,\nu)$

The guarantee of stability can be further demonstrated by confirming that all the poles of $B_k(z,\nu)$ have magnitudes (absolute values) less than one. Figure 12 shows the pole movements of the tunable blocks $B_1(z,\nu)$, $B_2(z,\nu)$, and $B_3(z,\nu)$, where the magnitudes of all the poles never exceed one. The simulation results have verified that the largest pole

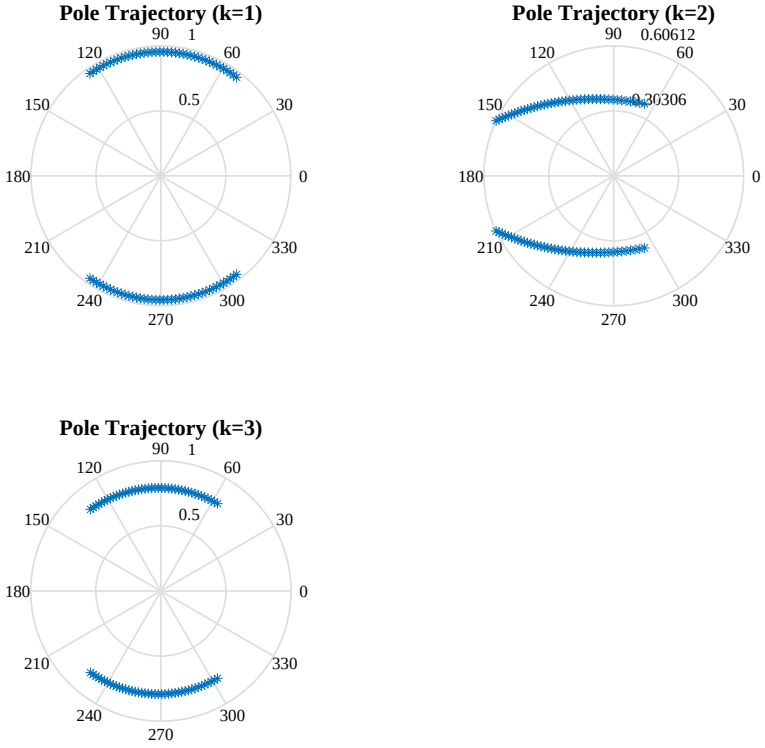

FIGURE 12. Pole movements of the tunable highpass-filter $H(z,\nu)$

radius is 0.9588. Thus, the resulting tunable highpass-filter $H(z, \nu)$ holds a sufficiently large stability margin. This signifies that the conversions (24) exploiting the stabilizing function $f(x)$ in Equation (6) render the tunable highpass-filter $H(z, \nu)$ robustly stable, regardless of the changes in ν.

4. **Conclusions.** The paper has investigated a novel cascade structure for designing and implementing cascade-structured tunable filters. As compared to the designs using direct-form structures [16, 17], this newly developed cascade model cascades only the 2nd-order blocks, taking the 2nd-order blocks as its basic building modules. This enables the cascade filter to be efficiently implemented by using only the 2nd-order blocks as basic implementation blocks. As a result, this novel structure features high modularity and easy hardware implementations. Since cascade structures suffer less coefficient quantization errors [15], this novel cascade structure can mitigate the quantization errors induced in hardware implementations. Therefore, the adoption of this cascade structure benefits both hardware implementation with low quantization errors and easy hardware implementations.

Based on the novel cascade structure, this paper has also investigated a 2-stage methodology for designing the cascade-structured tunable filter. The design minimizes the Lp-error, while keeping the stability ensured. The stability is ensured by exploiting the new stabilizing function to express the blocks' parameters as the stabilizing functions, rendering the stability definitely guaranteed.

A cascade-structured tunable highpass-filter has exemplified that the novel cascade structure can be successfully exploited to produce a tunable filter with its stability ensured and the Lp-error minimized. The design simulations of the tunable highpass-filter have shown that the average Lp-error (0.000012658) is significantly small, and that the resultant tunable highpass-filter holds a sufficiently large stability margin, whose largest pole radius (0.9588) is much less than one. Thus, the designed tunable highpass-filter exhibits high accuracy and is also robustly stable. As future works and research directions, we should investigate how to employ the novel cascade structure to achieve other types of tunable filters based on other error metrics.

REFERENCES

[1] T. I. Laakso, V. Valimaki, M. Karjalainen and U. K. Laine, Splitting the unit delay: Tools for fractional delay filter design, *IEEE Signal Process. Mag.*, vol.13, no.1, pp.30-60, 1996.

[2] T.-B. Deng, Discretization-free design of variable fractional-delay FIR digital filters, *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol.48, no.6, pp.637-644, 2001.

[3] T.-B. Deng and Y. Lian, Weighted-least-squares design of variable fractional-delay FIR filters using coefficient symmetry, *IEEE Trans. Signal Process.*, vol.54, no.8, pp.3023-3038, 2006.

[4] Y.-D. Huang, S.-C. Pei and J.-J. Shyu, WLS design of variable fractional-delay FIR filters using coefficient relationship, *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol.56, no.3, pp.220-224, 2009.

[5] T.-B. Deng, Minimax design of low-complexity even-order variable fractional-delay filters using second-order cone programming, *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol.58, no.10, pp.692-696, 2011.

[6] T.-B. Deng, Decoupling minimax design of low-complexity variable fractional-delay FIR digital filters, *IEEE Trans. Circuits Syst. I: Regular Papers*, vol.58, no.10, pp.2398-2408, 2011.

[7] P. Soontornwong and S. Chivapreecha, Pascal-interpolation-based noninteger delay filter and low-complexity realization, *Radioengineering*, vol.24, no.4, pp.1002-1012, 2015.

[8] P. Soontornwong, T.-B. Deng and S. Chivapreecha, Low-complexity and high-modularity structure for implementing transient-free Pascal-delay filter, *IEEE Trans. Signal Process.*, vol.65, no.23, pp.6233-6243, 2017.

[9] P. Sutthikarn, S. Chivapreecha, A. Trirat and T. Jongsataporn, A simple tunable biquadratic digital bandpass filter design for spectrum sensing in cognitive radio, *Proc. of IEEE ECTI-CON 2020*, Phuket, Thailand, pp.71-75, 2020.

[10] T.-B. Deng, Closed-form design and efficient implementation of variable digital filters with simultaneously tunable magnitude and fractional-delay, *IEEE Trans. Signal Process.*, vol.52, no.6, pp.1668-1681, 2004.
[11] R. Zarour and M. M. Fahmy, A design technique for variable digital filters, *IEEE Trans. Circuits Syst.*, vol.36, no.11, pp.1473-1478, 1989.
[12] A. O. Hussein and M. M. Fahmy, Design of 2-D linear phase variable recursive digital filters for parallel form implementation, *IEE Proceedings-G*, vol.138, no.3, pp.335-340, 1991.
[13] T.-B. Deng, Design of recursive 1-D variable filters with guaranteed stability, *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol.44, no.9, pp.689-695, 1997.
[14] T.-B. Deng, An improved method for designing variable recursive digital filters with guaranteed stability, *Signal Processing*, vol.81, no.2, pp.439-446, 2001.
[15] L. B. Jackson, *Digital Filters and Signa Processing*, Kluwer Academic Publishers, Amsterdam, 1992.
[16] T.-B. Deng, Typical benchmark specifications for designing stable variable filters using novel unity-bounded functions, *Vietnam Journal of Computer Science*, vol.11, no.1, pp.53-74, 2024.
[17] T.-B. Deng, Stability-guaranteed odd-order variable-bandwidth filters using stabilized odd-order transfer function, *Journal of Information and Telecommunication*, vol.8, no.3, pp.384-398, 2024.

Author Biography

Tian-Bo Deng received the Ph.D. degree in Electronic Engineering from Tohoku University, Sendai, Japan, in 1991. Currently, he is a full professor with the Faculty of Science, Toho University, Japan. From 1998 to 1999, he was also a visiting professor with University of Victoria, Canada. So far, Dr. Deng has received 18 paper awards, including 11 Best Paper Awards. His research interests focus on the design and implementation of variable one-dimensional (1-D) and variable multi-dimensional (M-D) filters. Dr. Deng served as an Editorial Board member for Elsevier's *Signal Processing*, and an Associate Editor for *IEEE Trans. Circuits and Systems II: Express Briefs*.