Design of Least-Squares and Minimax Composite Filters

Wu-Sheng Lu, *Life Fellow, IEEE*, and Takao Hinamoto, *Life Fellow, IEEE*

***Abstract*— We study a class of composite FIR filters (C-filters), each is composed of a prototype filter and a shaping filter in cascade, where the shaping filter is constructed by cascading several complementary comb filters. In particular, the problems of designing C-filters that are optimal in least-squares, equiripple passband and lease-squares stopband, and minimax sense are formulated, and three algorithms for designing such linear-phase FIR C-filters are proposed. The algorithms are based on an alternating optimization strategy in that the prototype and shaping filters are optimized in separate steps, which are coupled and carried out in a sequential manner to yield a satisfactory design. Design examples are presented to illustrate the algorithms and demonstrate the performance of the C-filters relative to their conventional FIR counterparts.**

***Index Terms*— Least-square FIR filters, minimax FIR filters, composite FIR filters.**

I. Introduction

A COMPOSITE filter (C-filter) refers to a digital filtering system that is composed of explicit individual modules, often called subfilters, which are connected in cascade or parallel, or a mixture of both. Well known classes of C-filters include interpolated FIR filters [1]–[3], frequency-response-masking filters [4], and their variants, see e.g. [5]–[11]. Over the years, analysis and design of C-filters have attracted a great deal of research interest primarily because of their ability to offer improved computational efficiency relative to their conventional counterparts when appropriate subfilter structures are chosen and their parameters are optimized in accordance with certain design criteria [12]. Recently, C-filters composed of a prototype filter and a shaping filter, connected in cascade, are shown to offer certain advantages over conventional counterparts [13], [14]. In the case of FIR C-filters [13], the shaping filter is constructed by cascading several complementary comb filters (CCFs) of the form $(1+z^{-l})^{k_l}$ where $1 \le l \le L$ and $k_l \ge 0$ are integers. As such, the shaping filter only requires several adders and memory units to implement and is free of multiplications. Yet, as demonstrated in [13], the shaping filter

Manuscript received June 8, 2017; revised September 1, 2017; accepted November 5, 2017. Date of publication December 4, 2017; date of current version February 15, 2018. This paper was recommended by Associate Editor H. Johansson. *(Corresponding author: Wu-Sheng Lu.)*

W.-S. Lu is with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC V8W 3P6, Canada (e-mail: wslu@ece.uvic.ca).

T. Hinamoto is with the Faculty of Engineering, Hiroshima University, Higashi-Hiroshima 739-8527, Japan (e-mail: hinamoto@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSI.2017.2772345

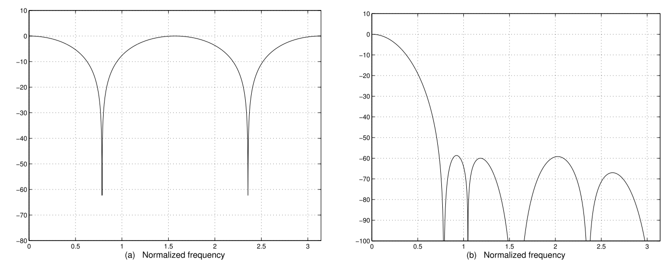


Fig. 1. Amplitude responses of (a) $1+z^{-4}$ and (b) $(1+z^{-1})^4(1+z^{-2})^4(1+z^{-3})^1(1+z^{-4})^2$.

is capable of effectively improving filter performance. As an illustration, Fig. 1(a) depicts the amplitude response of $1+z^{-4}$ (a scaling factor 0.5 has been used in the figure to normalize the filter gain at $\omega = 0$ to unity). By cascading several CCFs with appropriate powers k_l, a shaping filter can offer much reduced stopband energy while retaining decent passband gain. Fig. 1(b) depicts the amplitude response of a shaping filter $\prod_{l=1}^{4}(1+z^{-l})^{k_l}$ with $\{k_l,\ l = 1, 2, 3, 4\} = \{4, 4, 1, 2\}$ where the filter gain at $\omega = 0$ has been normalized to unity.

In spite of the design insights provided in [13], currently available design procedures for this type of composite filters lack a systematic methodology to incorporate adequate design criteria, problem formulations, and solution techniques together to produce optimal composite FIR filters in either least squares or minimax sense. In this paper, we present new algorithms for the design of C-filters, where the potential of the filter structure proposed in [13] is further explored. Specifically, the design of the prototype filter $H_p(z)$ as well as shaping filter $H_s(z)$ are formulated using rather different criteria in order for $H(z) = H_p(z)H_s(z)$ to achieve optimality in least squares (LS), equiripple passbands and least-squares stopbands (EPLSS), and minimax sense, respectively. FIR filters of these types are at the core of many digital signal processing systems, and find broad applications as documented in [12] and [15]–[21]. The main technical challenge encountered in the design procedure is largely related to the dealings with the resulting nonconvex formulations, each involving a mixture of a group of continuous design variables with a group of integer design variables that are supposed to be jointly optimized. The underlining principle in the development of the new design algorithms is a sequential optimization strategy that deals with the design of $H_p(z)$ and $H_s(z)$ separately, yet these two design steps are coupled by alternately performing

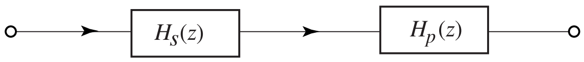


Fig. 2. A composite filter.

them until the algorithm converges. A strong motivation to adopt such a design methodology is that the design parameters from the prototype filter are continuous but those from the shaping filter are discrete, and these two sets of design variables are separate from each other due to the filter's cascade structure. Moreover, in all three design algorithms, the design steps concerning optimizing the prototype filter can be formulated as convex problems while the design step for updating the shaping filter can be effectively addressed either using discrete gradient or via convex optimization. Numerical examples are presented to illustrate the proposed design methods and demonstrate their performance.

The paper is organized as follows. Sec. II provides background of the C-filters in terms of structure and frequency-domain characterization which serves a starting point in formulating the LS, EPLSS, and minimax design problems. Section III starts by describing the design strategy and devotes its main body to the development of three design algorithms for the C-filters that are optimal in LS, EPLSS, and minimax sense, respectively. Three design examples with comparisons are presented in Sec. IV to demonstrate the performance of the proposed design methods. The present paper is an extended version of our work in [22].

II. Problem Formulation and Design Strategy

A. Composite Filters

We consider a C-filter that is composed of two subfilters, known as *prototype* filter $H_p(z)$ and shaping filter $H_s(z)$ respectively, that are connected in cascade as shown in Fig. 2. Thus the transfer function of the C-filter assumes the form

$$H(z) = H_p(z) \cdot H_s(z) \tag{1}$$

In this paper, we examine a class of linear-phase FIR C-filters where the prototype filter is an FIR filter of length N with transfer function

$$H_p(z) = \sum_{n=0}^{N-1} h_n z^{-n} \tag{2}$$

that largely determines the characteristics of $H(z)$, while the shaping filter is a CCF of the form

$$H_s(z) = \prod_{l=1}^{L} (1 + z^{-l})^{k_l} \tag{3}$$

where k_l are nonnegative integers, which reshapes the prototype filter for improved performance without substantial increase in implementation complexity. The frequency response can be expressed as

$$H(\boldsymbol{x}, \boldsymbol{y}, \omega) = H_p(\boldsymbol{x}, \omega) H_s(\boldsymbol{y}, \omega)$$

with

$$H_p(\boldsymbol{x}, \omega) = \sum_{n=0}^{N-1} h_n e^{-jn\omega} = e^{-j\tau_p\omega}[\boldsymbol{x}^T \boldsymbol{c}(\omega)] \tag{4}$$

where $\boldsymbol{x}$ is a coefficient vector determined by the impulse response of $H_p(z)$, $\tau_p = (N-1)/2$ is the group delay of $H_p(z)$, and $\boldsymbol{c}(\omega)$ is a vector with trigonometric components; and

$$H_s(\boldsymbol{y}, \omega) = \prod_{l=1}^{L} (1 + e^{-jl\omega})^{k_l} = e^{-j\tau_s\omega} \prod_{l=1}^{L} \left(2\cos\frac{l\omega}{2}\right)^{k_l} \tag{5}$$

where $\boldsymbol{y} = [k_1 \ k_2 \ \ldots \ k_L]^T$ and $\tau_s = \frac{1}{2}\sum_{l=1}^{L} l \cdot k_l$. It follows that the zero-phase frequency response of the C-filter is given by

$$A(\boldsymbol{x}, \boldsymbol{y}, \omega) = \boldsymbol{x}^T \boldsymbol{c}(\omega) \prod_{l=1}^{L} \left(2\cos\frac{l\omega}{2}\right)^{k_l} \tag{6}$$

and the group delay of the C-filter is given by $\tau = \tau_p + \tau_s$. The behaviour of shaping filter $H_s(z)$ is determined by the number L of CCFs used and the power k_l for each CCF. As long as $k_l > 0$, the first notch of a single CCF $(1 + z^{-l})^{k_l}$ over the normalized baseband $[0, \ \pi]$ occurs at $\omega = \pi/l$.

In the algorithms deveoloped below, the shaping filter and prototype filter are *jointly* optimized to achieve the design objectives.

B. Problem Formulation

For clarity of presentation, in the rest of the paper we focus on linear-phase lowpass FIR C-filters. Given a desired lowpass frequency response $H_d(\omega)$, prototype filter length N, number of CCF's L, and a bound M on the total group delay, we seek to find a linear-phase FIR C-filter $H(z)$ of form (1)–(3) such that its frequency response $H(\omega)$ optimally approximates $H_d(\omega)$ in a certain sense. Specifically we present three formulations where the optimal approximation is realized either in the least squares (LS) sense in both passband and stopband; or in the sense of minimum peak-to-peak amplitude ripple in passband and constrained least-squares in stopband (which is known as an equiripple passband and least-squares stopband (EPLSS) design); or in the minimax sense in both passband and stopband. In all design formulations, the design variables are the real-valued vector $\boldsymbol{x}$ representing the impulse response of the prototype filter $\{h_n, n = 0, \ 1, \ldots, \ N-1\}$ and vector $\boldsymbol{y}$ representing nonnegative integer powers k_l, $l = 1, 2, \ldots, L$ for the shaping filter.

C. Design Strategy

As will become transparent shortly, a key term involved in the proposed design formulation is the amplitude response of C-filter, $A(\boldsymbol{x}, \boldsymbol{y}, \omega)$, given by (6). We see from (6) that the design variables $\boldsymbol{x}$ and $\boldsymbol{y}$ in $A(\boldsymbol{x}, \boldsymbol{y}, \omega)$ are separate from each other. In addition, $A(\boldsymbol{x}, \boldsymbol{y}, \omega)$ depends on $\boldsymbol{x}$ linearly, but on $\boldsymbol{y}$ highly nonlinearly. Also note that the components of variable $\boldsymbol{y}$ are constrained to be *nonnegative* integers. Under these circumstances, it is intuitively natural to update variables $\boldsymbol{x}$ and $\boldsymbol{y}$ separately, and the updates are carried out in an alternate fashion. This approach leads to a sequential procedure where in each step one of the variables, say $\boldsymbol{x}$, is updated while the other variable, $\boldsymbol{y}$, is held fixed, and the solution so produced, say $\boldsymbol{x}_k$, is held fixed in the next step when variable $\boldsymbol{y}$ is updated to $\boldsymbol{y}_{k+1}$, and so on. This procedure continues until a stopping criterion is met and the last pair of updates $(\boldsymbol{y}_K, \boldsymbol{x}_K)$ is taken to be the optimal design.

III. Design Method

A. An Algorithm for the LS Design

An LS design produces a C-filter that reaches minimum approximation error in L_2 norm over frequency region $\Omega = \Omega_p \cup \Omega_a$ where Ω_p and Ω_a denote passband and stoppband of the filter, respectively. In the context of C-filter designs, such minimization shall be carried out subject to an upper bound on the total group delay $\tau = \tau_p + \tau_s$ for the C-filter to be useful in practice. The design problem can therefore be formulated as

$$\begin{aligned} &\underset{\boldsymbol{x},\boldsymbol{y}}{\text{minimize}} \int_{\omega\in\Omega} W(\omega)|H(\boldsymbol{x},\boldsymbol{y},\omega) - H_d(\omega)|^2 d\omega \\ &\text{subject to: } \tau_p + \tau_s \le M \end{aligned}$$

where $W(\omega) \ge 0$ weighs the squared error $|H(\boldsymbol{x},\boldsymbol{y},\omega) - H_d(\omega)|^2$ and M is an upper bound for the total group delay of the C-filter.

Let $H_d(\omega) = e^{-\tau\omega} A_d(\omega)$ with $A_d(\omega)$ the desired amplitude response, the integrand of the objective function becomes $W(\omega)[A(\boldsymbol{x},\boldsymbol{y},\omega) - A_d(\omega)]^2$. With a given filter length N for the prototype filter, its group delay τ_p is fixed to $(N-1)/2$, and the constraint on the total group delay is reduced to $\tau_s \le D$ where $\tau_s = \frac{1}{2}\sum_{l=1}^{L} l \cdot k_l$ and $D = M - (N-1)/2$. Hence the LS design is formulated as

$$\underset{\boldsymbol{x},\boldsymbol{y}}{\text{minimize}} \int_{\omega\in\Omega} W(\omega)[A(\boldsymbol{x},\boldsymbol{y},\omega) - A_d(\omega)]^2 d\omega \quad (7a)$$
$$\text{subject to: } k_l \ge 0 \text{ for } l = 1,\ 2,\ \ldots, L \quad (7b)$$
$$\frac{1}{2}\sum_{l=1}^{L} l \cdot k_l \le D \quad (7c)$$

Using (6), the objective function in (7a) (up to a constant which plays no role in the optimization) can be expressed explicitly in terms of design variables $\boldsymbol{x}$ and $\boldsymbol{y}$ as

$$f_{ls}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{x}^T \boldsymbol{Q}(\boldsymbol{y})\boldsymbol{x} - 2\boldsymbol{x}^T \boldsymbol{q}(\boldsymbol{y})$$

where

$$\boldsymbol{Q}(\boldsymbol{y}) = \int_{\Omega} W(\omega)\hat{\boldsymbol{c}}(\boldsymbol{y},\omega)\hat{\boldsymbol{c}}(\boldsymbol{y},\omega)^T d\omega \quad (8)$$
$$\boldsymbol{q}(\boldsymbol{y}) = \int_{\Omega} W(\omega)A_d(\omega)\hat{\boldsymbol{c}}(\boldsymbol{y},\omega) d\omega \quad (9)$$

and

$$\hat{\boldsymbol{c}}(\boldsymbol{y},\omega) = \boldsymbol{c}(\omega)\prod_{l=1}^{L}\left(2\cos\frac{l\omega}{2}\right)^{k_l} \quad (10)$$

By letting $\boldsymbol{i}_L = [1\ 2\ \ldots\ L]^T$, the constraint in (7c) becomes $\boldsymbol{i}_L^T \boldsymbol{y} \le 2D$. As a result, problem (7) is now expressed as

$$\underset{\boldsymbol{x},\boldsymbol{y}}{\text{minimize}}\ f_{ls}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{x}^T \boldsymbol{Q}(\boldsymbol{y})\boldsymbol{x} - 2\boldsymbol{x}^T \boldsymbol{q}(\boldsymbol{y}) \quad (11a)$$
$$\text{subject to: } \boldsymbol{y} \ge 0 \quad (11b)$$
$$\boldsymbol{i}_L^T \boldsymbol{y} \le 2D \quad (11c)$$

Following the design strategy outlined in Sec. II.C, problem (11) is solved by a sequential procedure where design variables $\boldsymbol{x}$ and $\boldsymbol{y}$ are updated in an alternate manner:

- Solve (11) with $\boldsymbol{y}$ fixed to $\boldsymbol{y} = \boldsymbol{y}_k$
  With $\boldsymbol{y} = \boldsymbol{y}_k$ that satisfies (11b) and (11c), these constraints can be neglected and (11a) becomes
  $$\underset{\boldsymbol{x}}{\text{minimize}}\ f_{ls}(\boldsymbol{x},\boldsymbol{y}_k) = \boldsymbol{x}^T \boldsymbol{Q}(\boldsymbol{y}_k)\boldsymbol{x} - 2\boldsymbol{x}^T \boldsymbol{q}(\boldsymbol{y}_k) \quad (12)$$
  From (8), $\boldsymbol{Q}(\boldsymbol{y}_k)$ is clearly positive definite, hence (12) is an unconstrained problem with a strictly convex quadratic objective function. Consequently, the stationary point of $f_{ls}(\boldsymbol{x},\boldsymbol{y}_k)$ given by
  $$\boldsymbol{x}_k = \boldsymbol{Q}(\boldsymbol{y}_k)^{-1}\boldsymbol{q}(\boldsymbol{y}_k) \quad (13)$$
  is the global minimizer of (12).
- Update $\boldsymbol{y}$ with $\boldsymbol{x}$ fixed to $\boldsymbol{x} = \boldsymbol{x}_k$
  Problem (11) in this case becomes
  $$\underset{\boldsymbol{y}}{\text{minimize}}\ f_{ls}(\boldsymbol{x}_k,\boldsymbol{y}) = \boldsymbol{x}_k^T \boldsymbol{Q}(\boldsymbol{y})\boldsymbol{x}_k - 2\boldsymbol{x}_k^T \boldsymbol{q}(\boldsymbol{y}) \quad (14a)$$
  $$\text{subject to: } \boldsymbol{y} \ge 0 \quad (14b)$$
  $$\boldsymbol{i}_L^T \boldsymbol{y} \le 2D \quad (14c)$$

A technical difficulty encountered in (14) is that the components of the design variable $\boldsymbol{y}$ must be integers. Here we handle the situation by examining a *discrete gradient*, denoted by $\boldsymbol{g}_k$, of the minimum total LS error with respect to $\boldsymbol{y}$ at $\boldsymbol{y} = \boldsymbol{y}_k$ subject to constraints (14b) and (14c), and this is done as follows.

The update starts at $\boldsymbol{y} = \boldsymbol{y}_k$. Let $\boldsymbol{y}_{k,l} = \boldsymbol{y}_k + \boldsymbol{e}_l$ where $\boldsymbol{e}_l$ is the lth column of the L by L identity matrix. In other words, $\boldsymbol{y}_{k,l}$ is constructed by adding unity to the lth component of $\boldsymbol{y}_k$. If $\boldsymbol{y}_{k,l}$ satisfies (14c), we compute the solution of minimizing $f_{ls}(\boldsymbol{x},\boldsymbol{y}_{k,l})$ using (13) and denote it as $\boldsymbol{x}_{k,l}$, the lth component of the discrete gradient is assigned to

$$g_k(l) = f_{ls}(\boldsymbol{x}_{k,l},\boldsymbol{y}_{k,l}) - f_{ls}(\boldsymbol{x}_k,\boldsymbol{y}_k) \quad (15)$$

If $\boldsymbol{y}_{k,l}$ violates (14c), then $g_k(l)$ is set to zero. The above process goes for l from 1 to L, and the discrete gradient so constructed is normalized to $\tilde{\boldsymbol{g}}_k = \boldsymbol{g}_k/\|\boldsymbol{g}_k\|_\infty$ so that the largest magnitude of the components in $\tilde{\boldsymbol{g}}_k$ is set to the unity. Now $\boldsymbol{y}_k$ is updated to

$$\boldsymbol{y}_{k+1} = \texttt{max}[\texttt{round}(\boldsymbol{y}_k - \alpha\tilde{\boldsymbol{g}}_k), \boldsymbol{0}] \quad (16)$$

where operation "round" ensures an integer-component $\boldsymbol{y}_{k+1}$, operation "max" ensures $\boldsymbol{y}_{k+1}$ to satisfy (14b), and scalar parameter $\alpha \in (0, 1]$ is used to ensure that satisfies (14c). Since $\tilde{\boldsymbol{g}}_k$ approximates the gradient of the L_2 error with respect to variable $\boldsymbol{y}$ at $\boldsymbol{y} = \boldsymbol{y}_k$, (16) is seen as a steepest-descent based update for variable $\boldsymbol{y}$ subject to (14b), (14c), as well as the integer constraint.

Algorithm 1 LS Design of C-Filters

Step 1 input $\boldsymbol{y}_0$, (N, L, D), ω_p, and ω_a.
Step 2 for $k = 0, 1, \ldots$
  (i) fix $\boldsymbol{y} = \boldsymbol{y}_k$, evaluate $\boldsymbol{Q}(\boldsymbol{y}_k)$ and $\boldsymbol{q}(\boldsymbol{y}_k)$ using (8) and (9), and compute $\boldsymbol{x}_k$ using (13);
  (ii) fix $\boldsymbol{x} = \boldsymbol{x}_k$, compute discrete gradient $\boldsymbol{g}_k$, and perform (16) to obtain $\boldsymbol{y}_{k+1}$;
  (iii) if $\boldsymbol{y}_k \neq \boldsymbol{y}_{k+1}$, set $k = k+1$ and repeat from step 2(i), otherwise go to Step 3.
Step 3 output $\boldsymbol{x}^* = \boldsymbol{x}_k$, $\boldsymbol{y}^* = \boldsymbol{y}_k$.

The algorithm terminates when $\boldsymbol{y}_{k+1}$ is identical to $\boldsymbol{y}_k$, and in this case the kth iterate $\{\boldsymbol{x}_k, \boldsymbol{y}_k\}$ is taken to be the final design. The algorithm described above is outlined as Algorithm 1.

The convergence of the algorithm is based on the following facts. (a) For a fixed $\boldsymbol{y} = \boldsymbol{y}_k$, the problem in (12) is convex which admits a unique solution $\boldsymbol{x}_k$ as seen in (13) (which is the global minimizer of problem (12)); (b) with variable $\boldsymbol{x}$ fixed to $\boldsymbol{x}_k$, variable $\boldsymbol{y}$ is updated from $\boldsymbol{y}_k$ to $\boldsymbol{y}_{k+1}$ by further reducing the objective function; (c) when the two steps described above are repeated, one obtains sequence of pairs $\{\boldsymbol{x}_k, \ \boldsymbol{y}_k\}$ for $k = 0, \ 1, \ \ldots$ at which the objective function is *monotonically decreasing*; (d) as a result, the procedure described above will terminate in finite number of rounds because the total number of points $\{\boldsymbol{y}_k, k = 0, 1, \ldots\}$ satisfying constraint (14c) is finite, and hence the algorithm is guaranteed to converge. On the other hand, the objective function in (11a) as a function of $(\boldsymbol{x}, \ \boldsymbol{y})$ is *not* convex, hence the original problem in (11) is a nonconvex problem. Consequently, the solution produced by Algorithm 1 can only be claimed to be a local minimizer. A remedy for this problem is to try different initial point $\boldsymbol{y}_0$ and select the one which yield the best local solution. This technique is found effective especially for the design of composite filters of moderate order because the number of feasible initial points $\boldsymbol{y}_0$ is rather limited in these cases.

B. An Algorithm for the EPLSS Design

An EPLSS design aims at producing a filter with minimum peak-to-peak amplitude ripple in passband subject to constraints on filter's energy. The design objective is achieved by solving the constrained problem

$$\underset{\boldsymbol{x},\boldsymbol{y}}{\text{minimize}} \ \max_{\omega\in\Omega_p} |A(\boldsymbol{x}, \boldsymbol{y}, \omega) - A_d(\omega)| \tag{17a}$$

$$\text{subject to:} \ \int_{\omega_a}^{\pi} |A(\boldsymbol{x}, \boldsymbol{y}, \omega)|^2 d\omega \le e_a \tag{17b}$$

$$\max_{\omega\in\Omega_a} |A(\boldsymbol{x}, \boldsymbol{y}, \omega)| \le \delta_a \tag{17c}$$

$$\text{same as (7b)} \tag{17d}$$

$$\text{same as (7c)} \tag{17e}$$

where ω_a denotes the stopband edge, e_a, and δ_a are constants representing upper bounds for stopband energy and peak filter gain in stopband, respectively.

While the algorithm proposed below also adopts a sequential procedure similar to Algorithm 1, considerably different techniques have been utilized in both design phases.

• *Solve (17) with $\boldsymbol{y}$ fixed to $\boldsymbol{y} = \boldsymbol{y}_k$*

With a fixed $\boldsymbol{y}_k$ that satisfies (17d) and (17e), these constraints are neglected in this part of the design. By introducing an upper bound δ_p for the objective function as an auxiliary variable, the problem at hand becomes

$$\text{minimize} \ \delta_p \tag{18a}$$

$$\text{s.t.} \ |A(\boldsymbol{x}, \boldsymbol{y}_k, \omega) - A_d(\omega)| \le \delta_p \quad \text{for } \omega \in \Omega_p \tag{18b}$$

$$\int_{\omega_a}^{\pi} |A(\boldsymbol{x}, \boldsymbol{y}_k, \omega)|^2 d\omega \le e_a \tag{18c}$$

$$|A(\boldsymbol{x}, \boldsymbol{y}_k, \omega)| \le \delta_a \ \text{for } \omega \in \Omega_a \tag{18d}$$

Using (6), (18b)–(18d) are reduced to

$$|\boldsymbol{x}^T \hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega) - A_d(\omega)| \le \delta_p \quad \text{for } \omega \in \Omega_p$$

$$\boldsymbol{x}^T \boldsymbol{Q}_a(\boldsymbol{y}_k)\boldsymbol{x} \le e_a$$

$$|\boldsymbol{x}^T \hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega)| \le \delta_a \quad \text{for } \omega \in \Omega_a$$

where $\hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega)$ is defined in (10) and

$$\boldsymbol{Q}_a(\boldsymbol{y}_k) = \int_{\omega_a}^{\pi} \hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega)\hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega)^T d\omega$$

For realistic implementation, one has to deal with a finite set of constraints and this is accomplished by replacing sets Ω_p and Ω_a with their discrete counterparts $\hat{\Omega}_p = \{\omega_{p1}, \omega_{p2}, \ldots, \omega_{pK_1}\}$ and $\hat{\Omega}_a = \{\omega_{a1}, \omega_{a2}, \ldots, \omega_{aK_2}\}$, respectively, where the frequency grids are placed uniformly over the respective bands. Following these, problem (18) is simplified to

$$\text{minimize} \quad \delta_p \tag{19a}$$

$$\text{s.t.} \ |\boldsymbol{x}^T \hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega) - A_d(\omega)| \le \delta_p \ \text{for } \omega \in \hat{\Omega}_p \tag{19b}$$

$$\boldsymbol{x}^T \boldsymbol{Q}_a(\boldsymbol{y}_k)\boldsymbol{x} \le e_a \tag{19c}$$

$$|\boldsymbol{x}^T \hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega)| \le \delta_a \ \text{for } \omega \in \hat{\Omega}_a \tag{19d}$$

With respect to variables $(\delta_p, \boldsymbol{x})$, the objective function and constraints (19b) and (19d) are linear, while (19c) is a convex quadratic constraint because $\boldsymbol{Q}_a(\boldsymbol{y}_k)$ is positive definite. Consequently, (19) is a convex problem whose global solution can be computed using reliable and convenient solvers [23], [24]. We denote the solution of (19) by $(\delta_k, \boldsymbol{x}_k)$.

• *Update $\boldsymbol{y}$ with $\boldsymbol{x}$ fixed to $\boldsymbol{x} = \boldsymbol{x}_k$*

With $\boldsymbol{x}_k$ obtained by solving (19) held fixed, we consider the stopband energy given by

$$J(\boldsymbol{y}) = \int_{\omega_a}^{\pi} |H(\boldsymbol{x}_k, \boldsymbol{y}, \omega)|^2 d\omega$$

$$= \int_{\omega_a}^{\pi} |\boldsymbol{x}_k^T \boldsymbol{c}(\omega)|^2 \prod_{l=1}^{L} \left(4\cos^2 \frac{l\omega}{2}\right)^{k_l} d\omega \tag{20}$$

Our strategy to update $\boldsymbol{y}$ is via minimizing $J(\boldsymbol{y})$ with respect to $\boldsymbol{y}$ subject to several relevant constraints which shall become transparent shortly.

As noticed earlier, a technical difficulty to utilize continuous optimization to optimize $\boldsymbol{y}$ is that the components of $\boldsymbol{y}$ must be integers. Unlike the LS design in Sec. III.A where this problem is addressed using the notion of discrete gradient, here the problem is handled by extending the k_ls from the domain of nonnegative integers to that of nonnegative reals. When an optimizing $\boldsymbol{y}$ with non-integer components is obtained, an integer update for $\boldsymbol{y}_k$ can be found by rounding its components to nearest integers.

The gradient and Hessian of $J(\boldsymbol{y})$ are evaluated by computing

$$\frac{\partial J(\boldsymbol{y})}{\partial k_i} = \int_{\omega_a}^{\pi} |H(\boldsymbol{x}_k, \boldsymbol{y}, \omega)|^2 \log\left(4\cos^2\frac{i\omega}{2}\right) d\omega$$
$$\frac{\partial J(\boldsymbol{y})}{\partial k_i \partial k_j} = \int_{\omega_a}^{\pi} |H(\boldsymbol{x}_k, \boldsymbol{y}, \omega)|^2 \log\left(4\cos^2\frac{i\omega}{2}\right) \cdot \log\left(4\cos^2\frac{j\omega}{2}\right) d\omega$$

Note that with an arbitrary column vector $\boldsymbol{v}$ of length L, we have

$$\boldsymbol{v}^T \nabla^2 J(\boldsymbol{y})\boldsymbol{v} = \int_{\omega_a}^{\pi} |H(\boldsymbol{x}_k, \boldsymbol{y}, \omega)|^2 \left(\sum_{i=1}^{n} v_i \log\left(4\cos^2\frac{i\omega}{2}\right)\right)^2 d\omega \geq 0$$

hence $J(\boldsymbol{y})$ is *convex*. This motivates us to consider a convex quadratic approximation of $J(\boldsymbol{y})$ as

$$\hat{J}(\boldsymbol{y}, \boldsymbol{y}_k) = J(\boldsymbol{y}_k) + \boldsymbol{d}_k^T \nabla J(\boldsymbol{y}_k) + \tfrac{1}{2}\boldsymbol{d}_k^T \nabla^2 J(\boldsymbol{y}_k)\boldsymbol{d}_k$$

where $\boldsymbol{d}_k = \boldsymbol{y} - \boldsymbol{y}_k$, and update $\boldsymbol{y}$ by minimizing $\hat{J}(\boldsymbol{y}, \boldsymbol{y}_k)$ subject to constraints (17d), (17e), and an additional constraint to ensure the performance of the C-filter over the passband, especially at passband edge ω_p:

$$1 - d_p \leq |H(\boldsymbol{x}_k, \boldsymbol{y}, \omega_p)| = |\boldsymbol{x}_k^T \boldsymbol{c}(\omega_p)| \prod_{l=1}^{L} |2\cos\frac{l\omega_p}{2}|^{k_l} \leq 1 + d_p$$

with a small $d_p > 0$. The last constraint is highly nonlinear with respect to $\boldsymbol{y}$, but fortunately it is equivalent to

$$\log(1 - d_p) \leq c + \sum_{l=1}^{L} k_l \log|2\cos\frac{l\omega_p}{2}| \leq \log(1 + d_p) \quad (21)$$

with $c = \log|\boldsymbol{x}_k^T \boldsymbol{c}(\omega_p)|$, which is linear with respect to $\boldsymbol{y}$. Based on above analysis, we propose to update $\boldsymbol{y}$ by solving the *convex* problem

$$\min \tfrac{1}{2}(\boldsymbol{y} - \boldsymbol{y}_k)^T \nabla^2 J(\boldsymbol{y}_k)(\boldsymbol{y} - \boldsymbol{y}_k) + (\boldsymbol{y} - \boldsymbol{y}_k)^T \nabla J(\boldsymbol{y}_k) \quad (22a)$$
$$\text{subject to: (17d), (17e), and (21)} \quad (22b)$$

and then rounding its solution to a nearest integer solution which is named as $\boldsymbol{y}_k$ again. In this way, problem (22) is solved a number of times and the solution of the last iteration is taken to be $\boldsymbol{y}_{k+1}$.

The algorithm terminates when $\boldsymbol{y}_{k+1}$ is found no difference from $\boldsymbol{y}_k$, and in this case the kth iterate $\{\boldsymbol{x}_k, \boldsymbol{y}_k\}$ is taken to be the final design. The algorithm described above is outlined as Algorithm 2.

Algorithm 2 EPLSS C-Filters

Step 1 input $\boldsymbol{y}_0$, (N, L, D), ω_p, ω_a, δ_a, e_a, d_p, and K.
Step 2 for $k = 0, 1, \ldots$
- (i) fix $\boldsymbol{y} = \boldsymbol{y}_k$ and solve (19) for $\boldsymbol{x}_k$;
- (ii) fix $\boldsymbol{x} = \boldsymbol{x}_k$, perform K iterations of (22) to obtain $\boldsymbol{y}_{k+1}$;
- (iii) if $\boldsymbol{y}_k \neq \boldsymbol{y}_{k+1}$, set $k = k + 1$ and repeat from step 2(i), otherwise go to Step 3.

Step 3 output $\boldsymbol{x}^* = \boldsymbol{x}_k$, $\boldsymbol{y}^* = \boldsymbol{y}_k$.

An argument similar to that in Sec. III.A can be made for the convergence of Algorithm 2 with a difference that the solution from Step 2 (i) is not in a closed form but is obtained using a convex programming solver which yields a unique global minimizer of problem (19).

C. An Algorithm for the Minimax Design

A minimax C-filter achieves weighted minimum peak-to-peak ripples in both passband and stopband subject to an upper bound on the total group delay. The design problem can be formulated as

$$\underset{\boldsymbol{x},\boldsymbol{y}}{\text{minimize}} \ \max_{\omega \in \Omega_p} |A(\boldsymbol{x}, \boldsymbol{y}, \omega) - A_d(\omega)| \quad (23a)$$
$$\text{subject to: same as (7b)} \quad (23b)$$
$$\text{same as (7c)} \quad (23c)$$

The algorithm proposed below adopts a two-phase alternate updating strategy as outlined in Sec. II.C.

- *Solve (23) with $\boldsymbol{y}$ fixed to $\boldsymbol{y} = \boldsymbol{y}_k$*

By assuming $\boldsymbol{y}_k$ satisfies (23b) and (23c) so that these constraints can be ignored in this phase of the design, and introducing an upper bound for the objective function, (23) becomes

$$\text{minimize } \delta \quad (24a)$$
$$\text{s.t. } W(\omega)|A(\boldsymbol{x}, \boldsymbol{y}_k, \omega) - A_d(\omega)| \leq \delta \text{ for } \omega \in \boldsymbol{\Omega} \quad (24b)$$

where $\boldsymbol{\Omega} = \boldsymbol{\Omega}_p \bigcup \boldsymbol{\Omega}_a$. Using (6), (24b) is reduced to

$$W(\omega)|\boldsymbol{x}^T \hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega) - A_d(\omega)| \leq \delta \text{ for } \omega \in \boldsymbol{\Omega}$$

This in conjunction with replacing set $\boldsymbol{\Omega}$ by its discrete counterpart $\hat{\boldsymbol{\Omega}} = \hat{\boldsymbol{\Omega}}_p \bigcup \hat{\boldsymbol{\Omega}}_a$ with $\hat{\boldsymbol{\Omega}}_p = \{\omega_{p1}, \omega_{p1}, \ldots, \omega_{pK_1}\}$ and $\hat{\boldsymbol{\Omega}}_a = \{\omega_{a1}, \omega_{a1}, \ldots, \omega_{aK_2}\}$, the problem at hand is led to

$$\text{minimize } \delta \quad (25a)$$
$$\text{s.t. } W(\omega)|\boldsymbol{x}^T \hat{\boldsymbol{c}}(\boldsymbol{y}_k, \omega) - A_d(\omega)| \leq \delta \text{ for } \omega \in \hat{\boldsymbol{\Omega}} \quad (25b)$$

which is a linear programming problem with respect to variables $\{\delta, \boldsymbol{x}\}$. We denote its solution by $\{\delta_k, \boldsymbol{x}_k\}$.

- *Update $\boldsymbol{y}$ with $\boldsymbol{x}$ fixed to $\boldsymbol{x} = \boldsymbol{x}_k$*

With $\boldsymbol{x}$ held fixed to $\boldsymbol{x} = \boldsymbol{x}_k$, problem (23) becomes

$$\text{minimize } \delta \quad (26a)$$
$$\text{s.t. } W(\omega)|\boldsymbol{x}^T \hat{\boldsymbol{c}}(\boldsymbol{y}, \omega) - A_d(\omega)| \leq \delta \text{ for } \omega \in \hat{\boldsymbol{\Omega}} \quad (26b)$$
$$\boldsymbol{y} \geq 0 \quad (26c)$$
$$\boldsymbol{i}_L^T \boldsymbol{y} \leq 2D \quad (26d)$$

Algorithm 3 Minimax Design of C-Filters

Step 1 input $\boldsymbol{y}_0$, (N, L, D), ω_p, and ω_a.
Step 2 for $k = 0, 1, \ldots$
(i) fix $\boldsymbol{y} = \boldsymbol{y}_k$, compute $\boldsymbol{x}_k$ by solving (25);
(ii) fix $\boldsymbol{x} = \boldsymbol{x}_k$, compute discrete gradient $\boldsymbol{g}_k$, and perform (28) to obtain $\boldsymbol{y}_{k+1}$;
(iii) if $\boldsymbol{y}_k \neq \boldsymbol{y}_{k+1}$, set $k = k+1$ and repeat from step 2(i), otherwise go to Step 3.
Step 3 output $\boldsymbol{x}^* = \boldsymbol{x}_k,\ \boldsymbol{y}^* = \boldsymbol{y}_k$.

And we update $\boldsymbol{y}$ from the current state of $\boldsymbol{y} = \boldsymbol{y}_k$ so as to reduce the bound δ in (26b) subject to constraints (26c) and (26d). Unlike the case of EPLSS designs where a convex closed-form expression of a performance measure (as seen in (20)) is available, the upper bound δ does not admit a closed-form expression in terms of variable $\boldsymbol{y}$. Under the circumstance we follow an approach similar to that employed in the LS design to define a *discrete gradient*, denoted by $\boldsymbol{g}_k$, of the minimum upper bound δ with respect to $\boldsymbol{y}$ at $\boldsymbol{y}_k$ subject to constraints (26c) and (26d), and this is done as follows.

The update starts at $\boldsymbol{y} = \boldsymbol{y}_k$. Let $\boldsymbol{y}_{k,l} = \boldsymbol{y}_k + \boldsymbol{e}_l$ where $\boldsymbol{e}_l$ is the lth column of the L by L identity matrix. If $\boldsymbol{y}_{k,l}$ satisfies (26d), we compute the solution of problem (25) for $\boldsymbol{y} = \boldsymbol{y}_{k,l}$ and denote it as $\{\delta_{k,l}, \boldsymbol{x}_{k,l}\}$, the lth component of the discrete gradient is assigned to

$$\boldsymbol{g}_k(l) = \delta_{k,l} - \delta_k \tag{27}$$

If $\boldsymbol{y}_{k,l}$ violates (26d), then $\boldsymbol{g}_k(l)$ is set to zero. The above process goes for l from 1 to L, and the discrete gradient so constructed is normalized to $\tilde{\boldsymbol{g}}_k = \boldsymbol{g}_k / \|\boldsymbol{g}_k\|_\infty$ so that the largest magnitude of the components in $\tilde{\boldsymbol{g}}_k$ is set to the unity. Now $\boldsymbol{y}_k$ is updated using $-\tilde{\boldsymbol{g}}_k$ to

$$\boldsymbol{y}_{k+1} = \mathtt{max}[\mathtt{round}(\boldsymbol{y}_k - \alpha \tilde{\boldsymbol{g}}_k), \boldsymbol{0}] \tag{28}$$

where operation "round" ensures an integer-component $\boldsymbol{y}_{k+1}$, operation "max" ensures $\boldsymbol{y}_{k+1}$ to satisfy (26c), and scalar parameter $\alpha \in (0,\ 1]$ is used to ensure that $\boldsymbol{y}_{k+1}$ satisfies (26d). Since $\tilde{\boldsymbol{g}}_k$ approximates the gradient of the minimum upper bound δ with respect to variable $\boldsymbol{y}$ at $\boldsymbol{y} = \boldsymbol{y}_k$, (28) is essentially a steepest-descent based update for variable $\boldsymbol{y}$ subject to (26c), (26d), as well as the integer constraint.

The algorithm terminates when $\boldsymbol{y}_{k+1}$ is identical to $\boldsymbol{y}_k$, and in this case the kth iterate $\{\boldsymbol{x}_k, \boldsymbol{y}_k\}$ is taken to be the final design. The algorithm described above is outlined as Algorithm 3.

An argument similar to that in Sec. III.A can be made for the convergence of Algorithm 3 with a difference that the solution from Step 2(i) is not in a closed form but is obtained using a linear programming solver which yields a unique global minimizer of problem (25).

D. Selection of Parameters L, N, and D

Parameters L, N, and D are a part of the input for the three design algorithms described above. Below we provide some guidelines for the selection of these parameters. Following the discussion in Sec. II.A, parameter L is related to the first notch of the shaping filter, which occurs at π/L, hence L should be selected to obey $\pi/L > \omega_p$ where ω_p denotes the passband edge of the lowpass filter to be designed. Consequently, π/ω_p can be used as an upper bound in selecting L. Concerning the selection of N and D, we recommend to start with an empirical observation that the total group delay $\tau_p + \tau_s$ of the C-filter should lie in a range comparable to the group delay $\hat{\tau}$ of a conventional linear-phase FIR filter of length $\hat{N}$ that achieves practically the same approximation accuracy as the C-filter. This relationship, $\tau_p + \tau_s \approx \hat{\tau}$, leads to

$$\frac{N-1}{2} + \frac{1}{2}\sum_{l=1}^{L} l \cdot k_l \approx \frac{\hat{N}-1}{2}, \quad \text{i.e.,} \ \sum_{l=1}^{L} l \cdot k_l \approx \hat{N} - N$$

This means that upper bound D should be slightly larger than $(\hat{N} - N)/2$. For illustration, assume a conventional lowpass linear-phase FIR filter of length $\hat{N} = 175$ is found to satisfy required approximation accuracy and we want to design a linear-phase C-filter with a prototype filter of (reasonably) reduced length, say $N = 151$, then parameter D should be selected to satisfy $D \geq 12$.

E. Design of other Types of Composite Filters

This paper focuses on composite lowpass filters because they can be utilized as initial filter models to build composite highpass, bandpass, and bandstop filters. Let $H(z) = H_p(z)H_s(z)$ be a lowpass composite filter with normalized passband edge ω_p and stopband edge ω_a. Then it can be readily verified that

(i) Composite transfer function $H_{high}(z) = H_p(-z)H_s(-z)$ represents a highpass filter with stopband edge $\pi - \omega_a$ and passband edge $\pi - \omega_p$. This is because

$$H_{high}(e^{j\omega}) = H(-e^{j\omega}) = H(e^{j(\omega-\pi)})$$

hence the frequency response $H_{high}(e^{j\omega})$ can be produced by shifting that of the lowpass filter $H(e^{j\omega}$ by π which yields a highpass frequency response.

(ii) Composite transfer function $H_{bs}(z) = H_p(z^2)H_z(z^2)$ represents a bandstop filter with the first passband edge $\omega_p/2$, first stopband edge $\omega_a/2$, second stopband edge $\pi - \omega_a/2$, and second passband edge $\pi - \omega_p/2$. This is because

$$H_{bs}(e^{j\omega}) = H(e^{j2\omega})$$

so the frequency response $H_{bs}(e^{j\omega})$ is produced by squeezing that of $H(e^{j\omega})$ over the entire baseband (from 0 to 2π) into a half of the frequency region (from 0 to π).

(iii) Composite transfer function

$$H_{bp}(z) = H_p(-z^2)H_s(-z^2)$$

represents a bandpass filter with the first stopband edge $(\pi - \omega_a)/2$, first passband edge $(\pi - \omega_p)/2$, second passband edge $(\pi + \omega_p)/2$, and second stopband edge $(\pi + \omega_a)/2$. This is because

$$H_{bp}(e^{j\omega}) = H(-e^{j2\omega}) = H(e^{j2(\omega-\pi/2)})$$

Thus the frequency response $H_{bp}(e^{j\omega})$ is produced by first producing a bandstop frequency response as in case (ii) and then shifting it by $\pi/2$.

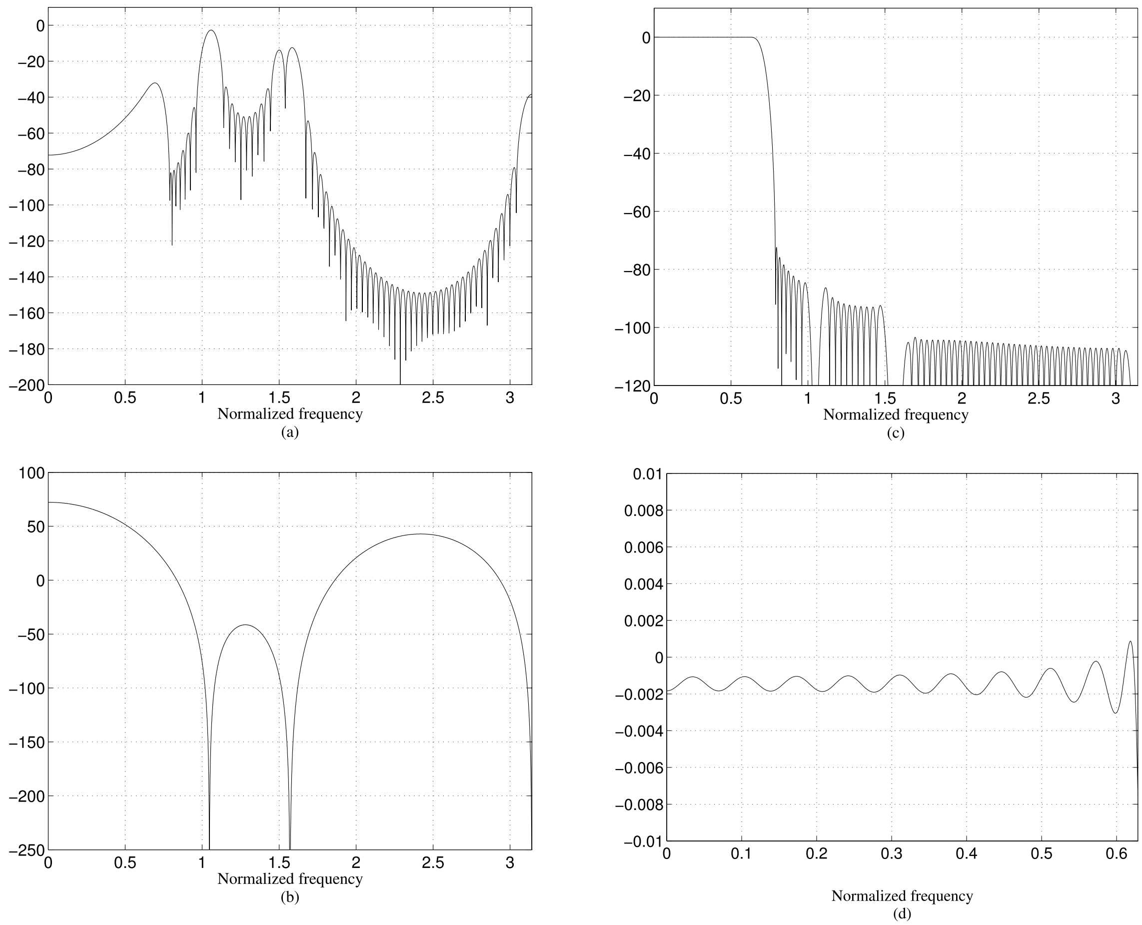


Fig. 3. Amplitude responses of (a) prototype filter $H_p(z)$, (b) shaping filter $H_s(z)$, (c) C-filter $H(z)$ over the entire baseband and (d) C-filter $H(z)$ over the passband for Example 1.

IV. Design Examples and Comparisons

In this section, we present three design examples to illustrate the respective design algorithms developed in Sec. III. Each design is evaluated in terms of approximation accuracy achieved in the frequency bands of interest. The performance of each design will be compared with its conventional FIR counterpart in terms of approximation accuracy and implementation complexity. Interpolated and frequency-response-masking FIR filters and their variants such as those documented in [1]–[11] are well known for their computationally efficiency, however this efficiency is achieved at increased implementation cost of complicated filter structures including much involved filter topology and additional function modules such as up-sampling etc.. By contrast the C-filters investigated in this paper possess a much simpler structure that differs from conventional FIR filters merely by a trivially-structured shaping filter. For the sake of comparison fairness, therefore, the design examples presented below are examined relative to conventional linear-phase LS, EPLSS, and minimax FIR filters.

Example 1: We illustrate Algorithm 1 by applying it to design a linear-phase lowpass LS C-filter with normalized passband edge $\omega_p = 0.2\pi$ and stopband edge $\omega_a = 0.25\pi$. The total L_2 error

$$E_2^{(\text{total})} = \int_0^{\omega_p} |A(\omega) - 1|^2 d\omega + \int_{\omega_a}^{\pi} |A(\omega)|^2 d\omega$$

is required to be no greater than 5×10^{-9}. The length of $H_p(z)$ was set to $N = 151$. With $L = 3$, $D = 14$, $W(\omega) \equiv 1$ for $\omega \in \Omega_p \bigcup \Omega_a$ and $\boldsymbol{y}_0 = [1\ 1\ 1]^T$, it took the algorithm eight iterations to converge to a solution $\{\boldsymbol{x}^*, \boldsymbol{y}^*\} = \{\boldsymbol{x}_8, \boldsymbol{y}_8\}$ with $\boldsymbol{y}^* = [1\ 6\ 5]^T$. The amplitude responses of the prototype filter $H_p(z)$, shaping filter $H_s(z)$ and C-filter $H(z)$ are shown in Fig. 3. The total LS error of the C-filter obtained was found to be $E_2^{(total)} = 4.9041 \times 10^{-9}$, and the group delay $\tau = (N-1)/2 + D = 89$. The number of multiplications M per output sample is $M = 76$. For comparison, a conventional linear-phase LS FIR filter [25] with comparable performance was designed. This design was obtained by applying Algorithm 1 with a single iteration where variable $\boldsymbol{y}_0$ was set to $\boldsymbol{y}_0 = \boldsymbol{0}$.

With $L = 3$ and $D = 14$, there are a total of 864 nonzero $\boldsymbol{y}$ vectors that satisfy constraints (11b) and (11c). An exhaustive search for a globally optimal solution was conducted by trying out every qualified $\boldsymbol{y}$ as the initial point $\boldsymbol{y}_0$. The optimal $\boldsymbol{y}$ was found to be $\boldsymbol{y}^{**} = [3\ 5\ 5]^T$ which led to an LS C-filter with $E_2^{(total)} = 4.8484 \times 10^{-9}$, representing 1.2% improvement in $E_2^{(total)}$ over the design obtained by Algorithm 1.

For comparison, with filter length $N = 175$, the total L_2 error of a conventional linear-phase LS FIR filter was found to be $E_2^{(total)} = 5.1967 \times 10^{-9}$. The group delay of the filter and the number of multiplications per output sample are $\tau = 87$ and $M = 88$, respectively.

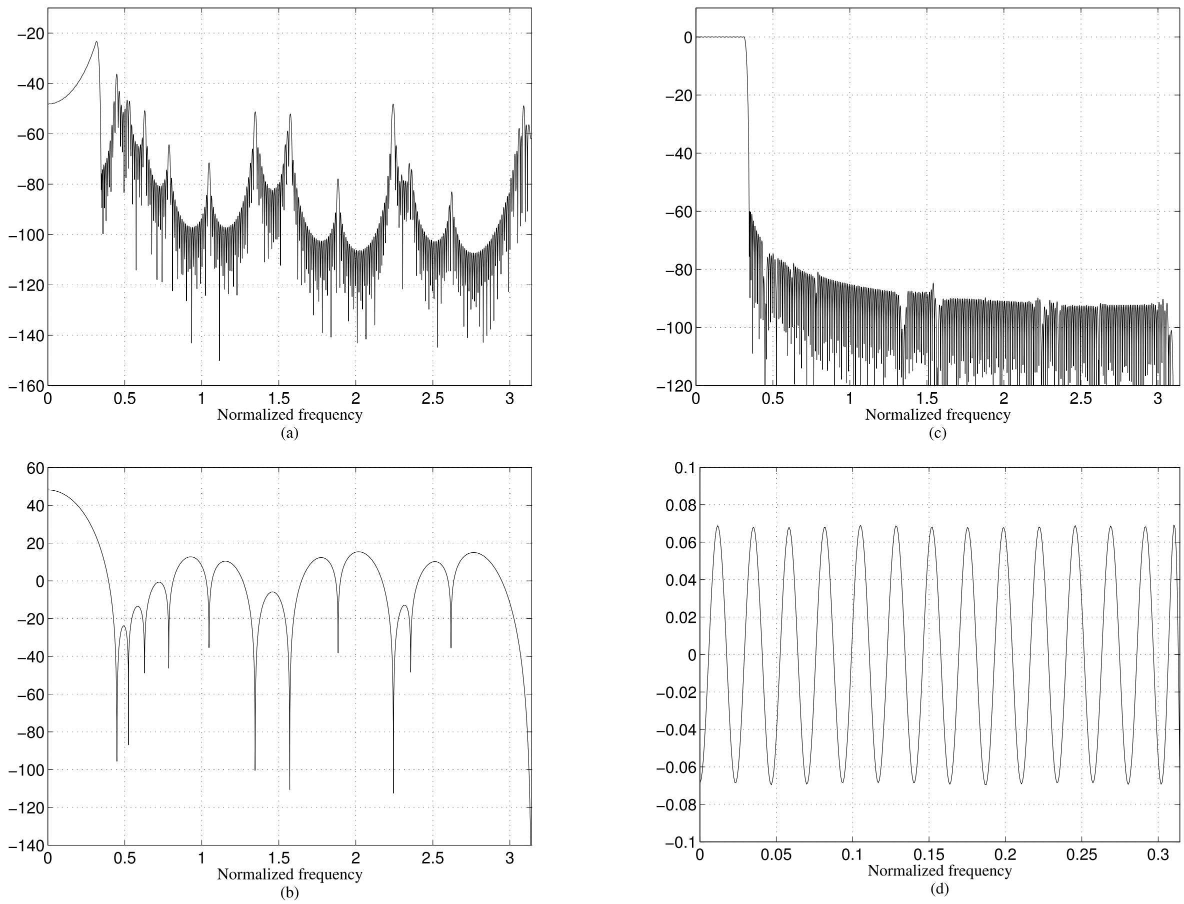


Fig. 4. Amplitude responses of (a) prototype filter $H_p(z)$, (b) shaping filter $H_s(z)$, (c) C-filter $H(z)$ over the entire baseband and (d) C-filter $H(z)$ over the passband for Example 2.

Example 2: We now apply Algorithm 2 to design a linear-phase lowpass EPLSS C-filter with sharp transition band specified by $\omega_p = 0.1\pi$ and $\omega_a = 0.11\pi$ whose performance is to be evaluated by its peak-to-peak ripple A_p in dB over the passband, minimum stopband attenuation A_a in dB, stopband energy E_a defined by

$$E_a = \int_{\omega_a}^{\pi} |A(\omega)|^2 d\omega$$

the number of multiplications M per output sample, and group delay τ. The passband ripple A_p was required to be no greater than 0.14 dB, the minimum stopband attenuation A_a is required to be below -60 dB (i.e. $\delta_a = 0.001$), and stopband energy E_a is required to be no greater than 2.2×10^{-8}. The length of $H_p(z)$ was set to $N = 519$. With $L = 7$, D = 18, $\boldsymbol{y}_0 = [1\ 1\ 1\ 1\ 1\ 1\ 1]^T$, $d_p = 0.08$, and $e_a = 5 \times 10^{-4}$, problem (19) was solved and its solution $\boldsymbol{x}_0$ together with $\boldsymbol{y}_0$ defines a C-filter achieving $A_p = 0.1681$, $A_a = -60$, and $E_a = 2.15 \times 10^{-8}$. With $\boldsymbol{x}_0$ held fixed, $K = 5$ iterations of (22) were performed to obtain an integer solution $\boldsymbol{y}_1 = [1\ 1\ 1\ 1\ 1\ 1\ 2]^T$. Since $\boldsymbol{y}_1 \neq \boldsymbol{y}_0$, problem (19) was solved again with $\boldsymbol{y}$ fixed to $\boldsymbol{y}_1$, where e_a was adjusted in order to obtain a solution $\boldsymbol{x}_1$ with practically the same peak gain in the stopband and stopband energy as $\boldsymbol{x}_0$ so that the two iterates $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ can be compared with each other in terms of peak passband ripple. With $e_a = 2.95 \times 10^{-4}$, the C-filter specified by $(\boldsymbol{x}_1, \boldsymbol{y}_1)$ achieved $A_p = 0.1387$, $A_a = -60.1083$ and $E_a = 2.1217 \times 10^{-8}$. We then run (22) again with $\boldsymbol{x}$ fixed to $\boldsymbol{x}_1$ and $K = 5$ that yields an integer solution $\boldsymbol{y}_2 = [1\ 1\ 1\ 1\ 1\ 1\ 2]^T$. Since $\boldsymbol{y}_2 = \boldsymbol{y}_1$, the algorithm is terminated and $(\boldsymbol{x}_1, \boldsymbol{y}_1)$ is claimed as the solution. The group delay and the number of multiplications M per output sample are $\tau = 276.5$ and $M = 260$, respectively. The amplitude responses of prototype filter $H_p(z)$, shaping filter $H_s(z)$, and C-filter $H(z)$ are shown in Fig.4.

For comparison, a conventional linear-phase EPLSS FIR filter with comparable performance was designed. This design was obtained by applying Algorithm 2 with a single iteration where variable $\boldsymbol{y}_0$ was set to $\boldsymbol{y}_0 = \boldsymbol{0}$. With filter length $N = 551$, the conventional EPLSS filter achieved $A_p = 0.1442$, $A_a = -60.1208$ and $E_a = 2.2140 \times 10^{-8}$. The group delay and the number of multiplications M per output sample are $\tau = 275$ and $M = 276$, respectively.

Example 3: Finally, Algorithm 3 was applied to design a linear-phase lowpass minimax C-filter with normalized passband edge $\omega_p = 0.15\pi$ and stopband edge $\omega_a = 0.18\pi$. It is required that the peak-to-peak passband ripple A_p is no greater than 0.01 dB and the minimum stopband attenuation is no greater than -64.8 dB. The length of $H_p(z)$ was set to $N = 221$. With $L = 5$, $D = 14$, $W(\omega) = 1$ for

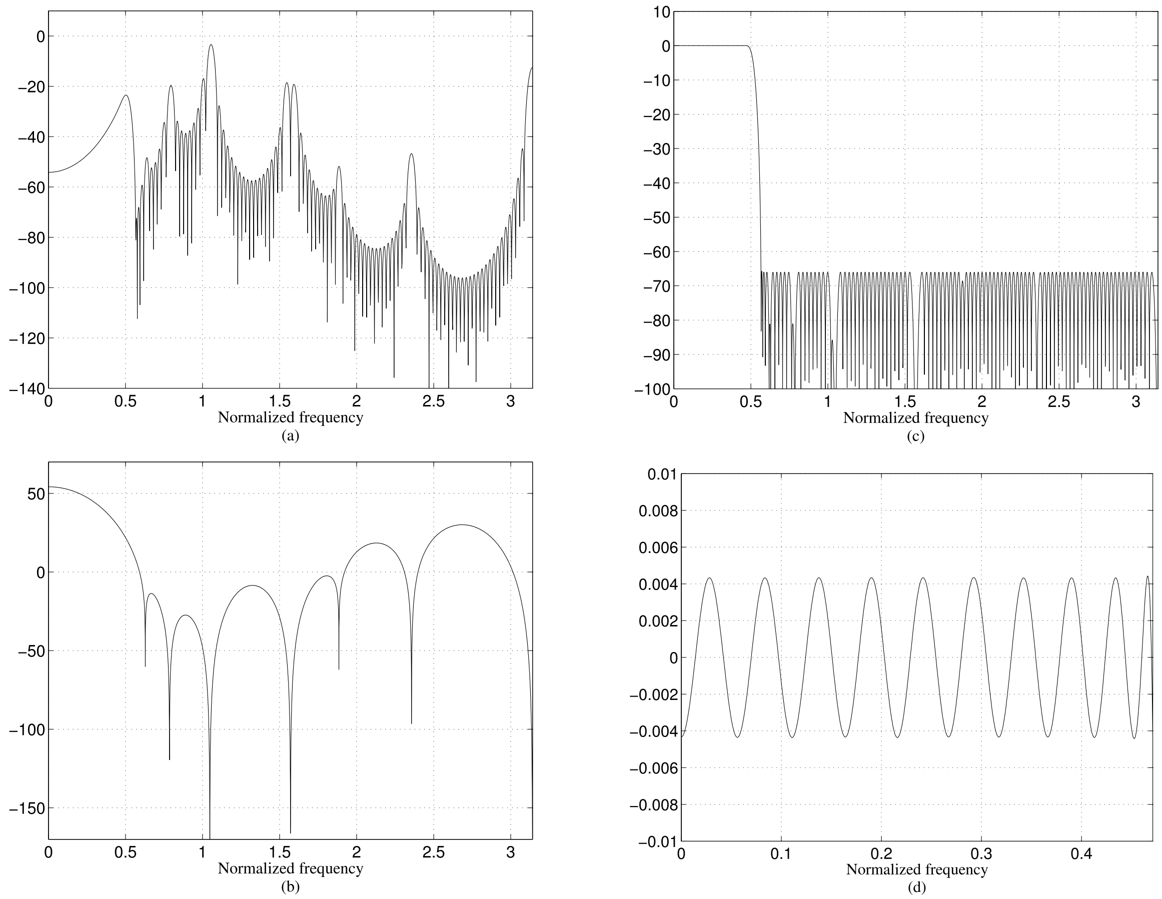


Fig. 5. Amplitude responses of (a) prototype filter $H_p(z)$, (b) shaping filter $H_s(z)$, (c) C-filter $H(z)$ over the entire baseband and (d) C-filter $H(z)$ over the passband for Example 3.

$\omega \in \Omega_p \bigcup \Omega_a$, and $\boldsymbol{y}_0 = [0\ 1\ 0\ 0\ 0]^T$, it took Algorithm 3 four iterations to converge to a solution $\{\boldsymbol{x}^*, \boldsymbol{y}^*\} = \{\boldsymbol{x}_5, \boldsymbol{y}_5\}$ with $\boldsymbol{y}^* = [0\ 3\ 3\ 2\ 1]^T$, hence a group delay $\tau = (N-1)/2 + D = 124$. The C-filter obtained yields $A_p = 0.0088$ dB and $A_a = -65.7571$ dB. The amplitude responses of the prototype filter $H_p(z)$, shaping filter $H_s(z)$ and C-filter $H(z)$ are shown in Fig. 5. The number of multiplications M per output sample that the C-filter requires is $M = 111$.

With $L = 5$ and $D = 14$, there are a total of 4048 nonzero $\boldsymbol{y}$ vectors that satisfy constraints (26c) and (26d). An exhaustive search for a globally optimal solution was conducted by trying out every qualified $\boldsymbol{y}$ as the initial point $\boldsymbol{y}_0$. The optimal $\boldsymbol{y}$ was found to be $\boldsymbol{y}^{**} = [0\ 3\ 3\ 2\ 1]^T$ which coincides with $\boldsymbol{y}^*$ obtained by Algorithm 3. Therefore, the C-filter produced by the proposed algorithm happens to be a global solution.

For comparison, a linear-phase FIR filters with equiripple passbands and stopbands obtained using the Parks-McClellan algorithm [15] with comparable performance was designed. This design was obtained using MATLAB function `firpm.m`, where filter length was set to $N = 247$. The group delay of the Parks-McClellan filter and the number of multiplications per output sample are $\tau = 123$ and $M = 124$, respectively.

In the three designs presented above, it is observed that with comparable approximation error and group delay the C-filters are found to be computationally more efficient relative to their conventional counterparts in terms of multiplications per output sample. We remark that the improved computational efficiency is gained at a cost of using an adequately designed shaping filter which however only requires several adders and memory units to implement and is free of multiplications.

V. Conclusion

In this paper, the potential of the C-filters proposed in [13] is further explored. Formulation and design algorithms for C-filters that are optimal in LS, EPLSS, and minimax sense are developed. It is demonstrated that with comparable approximation error and group delay the C-filters are computationally more efficient relative to their conventional counterparts in terms of multiplications per output sample.

References

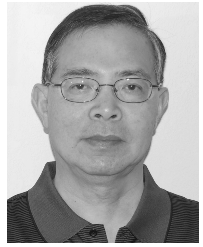
[1] Y. Neuvo, D. Cheng-Yu, and S. K. Mitra, "Interpolated finite impulse response filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, no. 3, pp. 563–570, Jun. 1984.

[2] T. Saramäki, Y. Neuvo, and S. K. Mitra, "Design of computationally efficient interpolated FIR filters," *IEEE Trans. Circuits Syst.*, vol. CS-35, no. 1, pp. 70–88, Jan. 1988.

[3] Y. Zou and T. Saramäki, "Design of computationally efficient narrowband linear-phase FIR filters," in *Proc. IEEE MELECON*, Dubrovnik, Croatia, May 2004, pp. 255–259.

[4] Y. C. Lim, "Frequency-response masking approach for the synthesis of sharp linear phase digital filters," *IEEE Trans. Circuits Syst.*, vol. CS-33, no. 4, pp. 357–364, Apr. 1986.

[5] Y. C. Lim and Y. Lian, "The optimum design of one- and two-dimensional FIR filters using the frequency response masking technique," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 40, no. 2, pp. 88–95, Feb. 1993.
[6] Y. C. Lim and Y. Lian, "Frequency-response masking approach for digital filter design: Complexity reduction via masking filter factorization," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 41, no. 8, pp. 518–525, Aug. 1994.
[7] T. Saramäki and H. Johansson, "Optimization of FIR filters using frequency-response masking approach," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2001, pp. 177–180.
[8] W.-S. Lu and T. Hinamoto, "Optimal design of frequency-response-masking filters using semidefinite programming," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 50, no. 4, pp. 557–568, Apr. 2003.
[9] Y. Liu and Z. Lin, "Optimal design of frequency-response masking filters with reduced group delays," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 6, pp. 1560–1570, Jul. 2008.
[10] J. Yli-Kaakinen and T. Saramäki, "An efficient algorithm for the optimization of FIR filters synthesized using the multistage frequency-response masking approach," *Circuits, Syst., Signal Process.*, vol. 30, no. 1, pp. 157–183, Feb. 2011.
[11] W.-S. Lu and T. Hinamoto, "A unified approach to the design of interpolated and frequency-response-masking FIR filters," *IEEE Trans. Circuits Syst. I, Reg. Paper*, vol. 63, no. 12, pp. 2257–2266, Dec. 2016.
[12] S. K. Mitra, *Digital Signal Processing—A Computer-Based Approach*, 3rd ed. New York, NY, USA: McGraw-Hill, 2006.
[13] D. Shiung, Y.-Y. Yang, and C.-S. Yang, " Improving FIR filters by using cascade techniques tips & tricks," *IEEE Signal Process. Mag.*, vol. 33, no. 3, pp. 108–114, May 2016.
[14] D. Shiung, Y.-Y. Yang, and C.-S. Yang, "Cascading tricks for designing composite filters with sharp transition bands," *IEEE Signal Process. Mag.*, vol. 33, no. 1, pp. 151–157 and 162, Jan. 2016.
[15] T. W. Parks and J. H. McClellan, "A program for the design of linear phase finite impulse response digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, no. 3, pp. 195–199, Aug. 1972.
[16] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1975.
[17] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1983.
[18] J. W. Adams, "FIR digital filters with least-squares stopbands subject to peak-gain constraints," *IEEE Trans. Circuits Syst.*, vol. 39, no. 4, pp. 376–388, Apr. 1991.
[19] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1992.
[20] S. K. Mitra and J. F. Kaiser, Eds. *Handbook for Digital Signal Processing*. Hoboken, NJ, USA: Wiley, 1993.
[21] A. Antoniou, *Digital Filters: Analysis, Design, and Applications*, 2nd ed. McGraw-Hill, 1993.
[22] W.-S. Lu and T. Hinamoto, "Design of composite filters with equiripple passbands and least-square stopbands," in *Proc. Int. Symp. Circuits Syst.*, Baltimore, MD, USA, May 2017, pp. 1910–1913.
[23] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," Dept. Econometrics, Tilburg Univ., Tilburg, The Netherlands, Aug. 1998.
[24] M. Grant and S. Boyd. *CVX: MATLAB Software for Disciplined Convex Programming*. Version 2.0. Accessed: Sep. 2013. [Online]. Available: http://cvxr.com/cvx
[25] T. W. Parks and C. S. Burrus, *Digital Filter Design*. Hoboken, NJ, USA: Wiley, 1987.



Wu-Sheng Lu (S'81–M'85–SM'90–F'99–LF'12) received the bachelor's degree in mathematics from Fudan University, Shanghai, China, in 1964, the M.S. degree in electrical engineering, and the Ph.D. degree in control science from the University of Minnesota, Minneapolis, USA, in 1983 and 1984, respectively. He was a Post-Doctoral Fellow with the University of Victoria, Victoria, BC, Canada, in 1985, and a Visiting Assistant Professor with the University of Minnesota from 1986 to 1987. Since 1987, he has been with the Electrical and Computer Engineering Department, University of Victoria, where he currently is a Professor. His current research interests include analysis and design of digital filters, signal and image processing, and the methods of convex optimization. He has co-authored with A. Antoniou of *Two-Dimensional Digital Filters* (Marcel Dekker, 1992) and *Practical Optimization: Algorithms and Engineering Applications* (Springer, 2007). He served as an Editor for the *Canadian Journal of Electrical and Computer Engineering* and an Associate Editor for several journals, including the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I and II, the *International Journal of Multidimensional Systems and Signal Processing*, and the *Journal of Circuits, Systems, and Signal Processing*. He is a fellow of the Engineering Institute of Canada and a Registered Professional Engineer in British Columbia, Canada.

Takao Hinamoto (M'77–SM'84–F'01–LF'11) received the B.E. degree from Okayama University, Okayama, Japan, in 1969, the M.E. degree from Kobe University, Kobe, Japan, in 1971, and the Dr. Eng. degree from Osaka University, Osaka, Japan, in 1977, all in electrical engineering.

From 1972 to 1988, he was with the Faculty of Engineering, Kobe University. From 1979 to 1981, he was a Visiting Member of Staff with the Department of Electrical Engineering, Queen's University, Kingston, ON, Canada, on leave from Kobe University. From 1988 to 1991, he was a Professor of electronic circuits with the Faculty of Engineering, Tottori University, Tottori, Japan. From 1992 to 2009, he was a Professor of electronic control with the Department of Electrical Engineering, Hiroshima University, Hiroshima, Japan. Since 2009, he has been an Emeritus Professor of Hiroshima University and a Teaching Member of Staff with the Hiroshima Institute of Technology, Hiroshima. His research interests include digital signal processing, system theory, and control engineering. He has authored over 400 papers in these areas. He has co-edited and co-authored *Two-Dimensional Signal and Image Processing* (Tokyo, Japan: SICE, 1996). He is a fellow of the Institute of Electronics, Information and Communication Engineers and the Society of Instrument and Control Engineers.