A Unified Approach to the Design of Interpolated and Frequency-Response-Masking FIR Filters

Wu-Sheng Lu[1], *Life Fellow, IEEE* and Takao Hinamoto[2], *Life Fellow, IEEE*

We present a unified approach to the design of two well-known classes of computationally efficient digital filters, namely the interpolated and frequency-response-masking (FRM) FIR filters, that are optimized in minimax sense. The highly non-convex minimax designs of these filters are carried out by jointly optimizing the subfilters involved using a convex-concave procedure (CCP) that yields an iterative and converging process where each iteration involves a convex problem of minimizing a linear function subject to convex quadratic constraints. We present an analysis to explain why CCP is well-suited for the design problems at hand. We also show that the CCP-based design framework is flexible to allow extension to the design of FRM filters that simultaneously promotes sparsity of the filter coefficients to reduce implementation complexity. Design examples with comparisons are presented to demonstrate the performance of the proposed design methods.

***Index Terms*—Interpolated FIR filters; Frequency-response-masking FIR filters; Convex-concave procedure.**

I. Introduction

Interpolated FIR (IFIR) filters [1], [2] and frequency-response-masking (FRM) FIR filters [3] are well-known classes of computationally efficient digital filters because the total number of multipliers and adders required to implement these filters are considerably less than those required by their conventional counterparts [4]. There has been a great deal of work in the literature following the aforementioned original development, see for example [5]-[18] and the references therein. Because of the importance of these filters, it is naturally desirable to develop a simple and unifying design method that is applicable to both of the filter classes. In this paper, we propose such a design technique in that the subfilters involved are jointly optimized in the minimax sense. The core of the proposed design approach is the convex-concave procedure (CCP) [19]-[21] that allows the use of efficient convex optimization to deal with nonconvex design problems where the objective functions assume the form of difference of two convex functions.

The paper is organized as follows. Sec. II provides background of IFIR and FRM filters in terms of their structures and frequency-domain models which serve a starting point in formulating the design problems of interest in the subsequent sections. To make the paper self-contained and reach a wider audience, Sec. II also introduces the basics of convex-concave procedure and several properties that are most relevant to the course of filter designs. In Sections III and IV, the CCP is applied to design IFIR and FRM filters, respectively. Our focus is on the design of original IFIR and single stage FRM filters so as for the reader to sense this general design strategy in a simple and transparent manner. We also explain why the CCP is well-suited that simultaneously promotes sparsity of filter coefficients for the designs at hand. In Sec. V, we present a variant of the CCP-based technique for the design of FRM filters with improved implementation efficiency. Design examples with comparisons are presented in Sec. VI to demonstrate the performance of the proposed design methods. The present paper is a much extended version of the work in [22].

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

T. Hinamoto is with Hiroshima University, Higashi-Hiroshima, 739-8527, Japan, email: hinamoto@ieee.org

II. Preliminaries and Related Work

A. Interpolated FIR Filters

IFIR filters are introduced in [1] and further investigated in [2]. As shown in Fig. 1, an IFIR filter is composed of a cascade of two FIR filters, whose transfer function assumes the form

$$H(z) = F(z^L)M(z) \tag{1}$$

where $L > 0$ is an integer that determines the degree of the filter's sparsity, hence its computational efficiency. Suppose the parent transfer function $F(z)$ represents a lowpass filter, then $F(z^L)$ is a periodic filter whose widths of passband and transition bands are $1/L$-th of those of $F(z)$. Consequently, by cascading $F(z^L)$ with a lowpass $M(z)$ (known as interpolator) that suppresses the undesired passbands of $F(z^L)$, a lowpass FIR filter with narrow passband and sharp transition band can be obtained with improved computational complexity relative to that of a conventional FIR with comparable approximation accuracy.

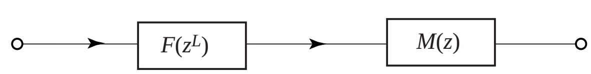

Fig. 1. An IFIR filter.

B. Frequency-Response-Masking Filters

Another class of FIR filters with very narrow transition bands, known as FRM filters, is proposed in [3] and has since been a subject of study [4]–[15]. As illustrated in Fig. 2, a single stage FRM filter has a connected parallel structure where the periodic filter $F(z^L)$ and its delay complementary periodic filers, $z^{-L(N-1)/2} - F(z^L)$, are cascaded with masking filters $M_a(z)$ and $M_c(z)$, respectively, so as to produce an

FIR filter with sharper transition bands. The transfer function of the FRM filter in Fig. 2 is given by

$$H(z) = F(z^L)M_a(z) + [z^{-L(N-1)/2} - F(z^L)]M_c(z) \quad (2)$$

where N denotes the length of $F(z)$ and is assumed to be an odd integer throughout.

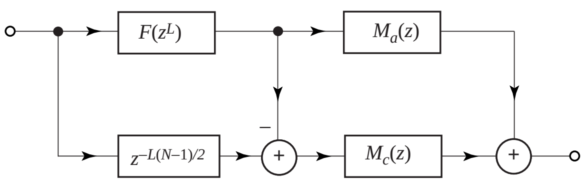

Fig. 2. A single-stage FRM filter.

C. Convex-Concave Procedure (CCP)

As will become clear shortly, jointly optimizing all subfilters involved in an IFIR or FRM filter is not a convex problem. The CCP is a heuristic method of convexifying nonconvex problems. It is known that a function $f(\boldsymbol{x})$ with continuous second-order derivatives can be expressed as a difference of two convex functions [23]. If the function in question has a bounded Hessian, such an expression, namely $f(\boldsymbol{x}) = u(\boldsymbol{x}) - v(\boldsymbol{x})$, can be explicitly constructed [19]. We are thus motivated to consider a nonconvex problem of the form

$$\text{minimize} \quad f(\boldsymbol{x}) = u_0(\boldsymbol{x}) - v_0(\boldsymbol{x}) \quad (3a)$$
$$\text{subject to:} \quad u_j(\boldsymbol{x}) \le v_j(\boldsymbol{x}) \quad \text{for } j = 1,\ 2,\ \ldots,\ q \quad (3b)$$

where $u_j(\boldsymbol{x})$ and $v_j(\boldsymbol{x})$ for $j = 0,\ 1,\ \ldots, q$ are convex. The CCP is an iterative procedure for solving (3). In the kth iteration where iterate $\boldsymbol{x}_k$ is known, CCP performs two steps: (i) convexification of the objective function and constraints at $\boldsymbol{x}_k$ by replacing each $v_j(\boldsymbol{x})$ by its affine approximation

$$\hat{v}_j(\boldsymbol{x}, \boldsymbol{x}_k) = v_j(\boldsymbol{x}_k) + \nabla v_j(\boldsymbol{x}_k)^T(\boldsymbol{x} - \boldsymbol{x}_k) \quad (4)$$

where ∇ denotes the gradient operator.
(ii) Solve the convex problem

$$\text{minimize} \quad \hat{f}(\boldsymbol{x}) = u_0(\boldsymbol{x}) - \hat{v}_0(\boldsymbol{x}) \quad (5a)$$
$$\text{subject to} \quad u_j(\boldsymbol{x}) \le \hat{v}_j(\boldsymbol{x}) \text{ for } j = 1,\ 2,\ \ldots,\ q \quad (5b)$$

One can start CCP by solving (5) with an initial $\boldsymbol{x}_0$ that is feasible for the original problem (3). This means that $u_j(\boldsymbol{x}_0) - v_j(\boldsymbol{x}_0) \le 0$ for $1 \le j \le q$ which in conjunction with the convexity of $v_j(\boldsymbol{x})$ implies that

$$u_j(\boldsymbol{x}_0) - \hat{v}_j(\boldsymbol{x}_0, \boldsymbol{x}_0) = u_j(\boldsymbol{x}_0) - v_j(\boldsymbol{x}_0) \le 0$$

hence $\boldsymbol{x}_0$ is also a feasible point for the convex problem in (5). Also note that if $\boldsymbol{x}_{k+1}$ is produced by solving (5) in the kth iteration, then $\boldsymbol{x}_{k+1}$ is also feasible for the original problem in (3) because the convexity of $v_j(\boldsymbol{x})$ implies that $v_j(\boldsymbol{x}_{k+1}) \ge \hat{v}_j(\boldsymbol{x}_{k+1}, \boldsymbol{x}_k)$ which leads to

$$u_j(\boldsymbol{x}_{k+1}) - v_j(\boldsymbol{x}_{k+1}) \le u_j(\boldsymbol{x}_{k+1}) - \hat{v}_j(\boldsymbol{x}_{k+1}, \boldsymbol{x}_k) \le 0$$

Another desirable property of CCP is that it is a descent method [20], namely the original objective function, $u_0(\boldsymbol{x}) - v_0(\boldsymbol{x})$, decreases monotonically, at the iterates $\{\boldsymbol{x}_k\}$ generated by solving convex problem (5). To see this, note that

$$\begin{aligned} f_k &= u_0(\boldsymbol{x}_k) - v_0(\boldsymbol{x}_k) = u_0(\boldsymbol{x}_k) - \hat{v}_0(\boldsymbol{x}_k, \boldsymbol{x}_k) \\ &\ge u_0(\boldsymbol{x}_{k+1}) - \hat{v}_0(\boldsymbol{x}_{k+1}, \boldsymbol{x}_k) \end{aligned} \quad (6)$$

Since $v_0(\boldsymbol{x})$ is convex, we have $v_0(\boldsymbol{x}_{k+1}) \ge \hat{v}_0(\boldsymbol{x}_{k+1}, \boldsymbol{x}_k)$, hence

$$u_0(\boldsymbol{x}_{k+1}) - \hat{v}_0(\boldsymbol{x}_{k+1}, \boldsymbol{x}_k) \ge u_0(\boldsymbol{x}_{k+1}) - v_0(\boldsymbol{x}_{k+1}) = f_{k+1}$$

which in conjunction with (6) gives $f_{k+1} \le f_k$. It follows that if the objective is bounded from below, then $\{f_k\}$ converges to a finite limit. As expected, the convergence of CCP-produced iterates $\{\boldsymbol{x}_k\}$ cannot be assured in general because it deals with noncovex problems as a heuristic method after all. Nevertheless, it can be shown that under mild conditions $\{\boldsymbol{x}_k\}$ converges to critical points of the original problem [21]. The CCP may be terminated in several ways, including by a certain number of iterations or by monitoring if the difference in objective function between two consecutive iterations is less than a given convergence tolerance.

D. Related Work

The studies of both IFIR and FRM filters began during 1980's [1]–[3]. The performance of IFIR filters was later enhanced [30] and the design method has been supported by MATLAB® [4]. Concerning the design of FRM filters, multistage FRM filters which offer additional reduction of realization complexity was initiated in [3] and further enhanced in [5], [11], [14], and [16]–[18]. Variants and applications of FRM are proposed over the years [6], [7], [15]. In most of the design methods, the subfilters involved are optimized either separately or jointly as nonlinear and nonconvex problems [8], [10]–[18]. In [16], the design of FRM filters is carried out using a batch back-propagation neural network algorithm, this appears to be the first attempt of applying neural networks in the field.

III. Minimax Design of IFIR Filters

A. Problem Formulation

Following (1), let the transfer functions of the parent filter and interpolator of an IFIR filter be given by

$$F(z) = \sum_{n=0}^{N-1} f_n z^{-n} \quad \text{and} \quad M(z) = \sum_{n=0}^{N_i-1} m_n z^{-n} \quad (7)$$

respectively, and the frequency response of an IFIR filter is given by

$$H(e^{j\omega}) = F(e^{jL\omega})M(e^{j\omega}) \quad (8)$$

Assume both $F(z)$ and $M(z)$ are of linear phase response, the zero-phase frequency response of the IFIR filter is given by

$$H_0(\boldsymbol{a}_f, \boldsymbol{a}_m, \omega) = [\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)] \cdot [\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)] \quad (9)$$

where $\boldsymbol{a}_f$ and $\boldsymbol{a}_m$ are coefficient vectors determined by the impulse responses of $F(z)$ and $M(z)$, respectively, $\boldsymbol{t}_f(\omega)$ and $\boldsymbol{t}_m(\omega)$ are vectors with trigonometric components determined by the filter lengths and types (e.g. odd or even length and symmetrical or antisymmetrical filter coefficients). In the case of both $F(z)$ and $M(z)$ being of odd length, for example, we have

$$\boldsymbol{a}_f = \begin{bmatrix} f_{(N-1)/2} \\ 2f_{(N+1)/2} \\ \vdots \\ 2f_{N-1} \end{bmatrix}, \ \boldsymbol{a}_m = \begin{bmatrix} m_{(N_i-1)/2} \\ 2m_{(N_i+1)/2} \\ \vdots \\ 2m_{N_i-1} \end{bmatrix}$$

$$\boldsymbol{t}_f(\omega) = \begin{bmatrix} 1 \\ \cos\omega \\ \vdots \\ \cos[(N-1)\omega/2] \end{bmatrix}, \ \boldsymbol{t}_m(\omega) = \begin{bmatrix} 1 \\ \cos\omega \\ \vdots \\ \cos[(N_i-1)\omega/2] \end{bmatrix}$$

where f_n's and m_n's are from (7).

Let $H_d(\omega)$ be the desired zero-phase response, the frequency-weighted minimax design of an IFIR filter amounts to finding vectors $\boldsymbol{a}_f$ and $\boldsymbol{a}_m$ that solves the problem

$$\underset{\boldsymbol{a}_f,\boldsymbol{a}_m}{\text{minimize}} \max_{\omega\in\Omega} w(\omega)|H_0(\boldsymbol{a}_f, \boldsymbol{a}_m, \omega) - H_d(\omega)| \quad (10)$$

where $w(\omega) > 0$ is a frequency-selective weight over a frequency domain of interest, Ω. Evidently, problem (10) is nonconvex with respect to design variables $\boldsymbol{x}^T = [\boldsymbol{a}_f^T \ \boldsymbol{a}_m^T]$.

B. Convexification of (10) using CCP

By introducing an upper bound δ for the objective function in (10) and treating the bound as an additional design variable, problem (10) becomes

$$\text{minimize } \delta \quad (11a)$$
$$\text{s.t. } [\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)][\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)] \le \delta_w + H_d(\omega) \quad (11b)$$
$$-[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)][\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)] \le \delta_w - H_d(\omega) \quad (11c)$$

for $\omega \in \Omega$, where $\delta_w = \delta/w(\omega)$. Bearing CCP in mind, we add the term $0.5s(\boldsymbol{x}, \omega)$ with

$$s(\boldsymbol{x}, \omega) = (\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega))^2 + (\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega))^2 \quad (12)$$

to both sides of (11b) and (11c) to obtain an equivalent pair of constrains as

$$[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega) + \boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)]^2 \le s(\boldsymbol{x}, \omega) + 2\delta_w + 2H_d(\omega) \quad (13a)$$
$$[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega) - \boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)]^2 \le s(\boldsymbol{x}, \omega) + 2\delta_w - 2H_d(\omega) \quad (13b)$$

Clearly, the functions on both sides of (13a) and (13b) are convex with respect to $\boldsymbol{x}$ and δ, therefore (13) is suitable for application of CCP. We proceed by replacing $s(\boldsymbol{x}, \omega)$ in (13) by its linearization at $\boldsymbol{x}_k$, namely $s(\boldsymbol{x}_k, \omega) + \nabla s(\boldsymbol{x}_k, \omega)^T (\boldsymbol{x} - \boldsymbol{x}_k)$. This gives

$$[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega) + (-1)^i \boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)]^2 \le \eta_i(\boldsymbol{x}, \boldsymbol{x}_k, \omega) \ \ i = 0, 1 \quad (14)$$

with

$$\eta_i(\boldsymbol{x}, \boldsymbol{x}_k, \omega) = \\ s(\boldsymbol{x}_k, \omega) + \nabla s(\boldsymbol{x}_k, \omega)^T (\boldsymbol{x} - \boldsymbol{x}_k) + 2\delta_w + (-1)^i 2H_d(\omega)$$

where for practical purposes ω is taken from $\Omega_d = \{\omega_j, j = 1, 2, \ldots, K\} \subset \Omega$, a finite discrete set of frequency grids that are sufficiently dense and placed uniformly over the frequency region of interest Ω.

In summary, the convex problem to be solved in the kth iteration of CCP is given by

$$\text{minimize } \delta \quad (15a)$$
$$\text{s.t. } [\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega_j) + (-1)^i \boldsymbol{a}_m^T \boldsymbol{t}_m(\omega_j)]^2 \le \eta_i(\boldsymbol{x}, \boldsymbol{x}_k, \omega_j) \quad (15b)$$
$$\text{for } i = 0, \ 1 \text{ and } j = 1, \ 2, \ \ldots, \ K$$

where $\omega_j \in \Omega_d$. It is well known that this problem of minimizing a linear function subject to convex quadratic constraints can be formulated as a semidefinite programming (SDP) or second-order cone programming (SOCP) problem [24], which can be solved efficiently [25]–[27]. The CCP-based algorithm may be terminated after a given number of iterates $\{\boldsymbol{x}_k\}$ have been generated or, when the difference between two consecutive error bounds, namely $|\delta_{k-1} - \delta_k|$, is less than a given convergence tolerance. In either case, the last iterate produced from (15) is taken to be a solution of the design problem. A step-by-step description of the proposed algorithm is given below as Algorithm 1.

Algorithm 1 for IFIR filters

Step 1: input $\boldsymbol{x}_0, (N, N_i), \Omega_d, H_d(\omega)$ for $\omega \in \Omega_d$, $w(\omega)$, and K_i.
Step 2: for $k = 0, \ 1, \ \ldots, \ K_i - 1$
(i) solve (15) for $\boldsymbol{a}_f$ and $\boldsymbol{a}_m$
(ii) construct $\boldsymbol{x}_{k+1} = [\boldsymbol{a}_f^T \ \boldsymbol{a}_m^T]^T$
end
Step 3: output $\boldsymbol{x}^* = \boldsymbol{x}_{K_i}$

C. Remarks on Convexification in (13)–(14)

Note that expressing a nonconvex function as $f(\boldsymbol{x}) = u(\boldsymbol{x}) - v(\boldsymbol{x})$ is not unique. In fact if $f(\boldsymbol{x}) = u(\boldsymbol{x}) - v(\boldsymbol{x})$ holds with $u(\boldsymbol{x})$ and $v(\boldsymbol{x})$ convex, then $f(\boldsymbol{x}) = \tilde{u}(\boldsymbol{x}) - \tilde{v}(\boldsymbol{x})$ also holds with $\tilde{u}(\boldsymbol{x}) = u(\boldsymbol{x}) + w(\boldsymbol{x})$ and $\tilde{v}(\boldsymbol{x}) = v(\boldsymbol{x}) + w(\boldsymbol{x})$ where $w(\boldsymbol{x})$ is an arbitrary convex function, hence both $\tilde{u}(\boldsymbol{x})$ and $\tilde{v}(\boldsymbol{x})$ remain convex. A natural question arising from this observation is how this non-uniqueness affects the CCP. In CCP, a nonconvex constraint $u(\boldsymbol{x}) \le v(\boldsymbol{x})$ is replaced by

$$u(\boldsymbol{x}) \le v(\boldsymbol{x}_k) + \nabla v(\boldsymbol{x}_k)^T (\boldsymbol{x} - \boldsymbol{x}_k) \quad (16)$$

Now if we treat the above nonconvex constraint as $u(\boldsymbol{x}) + w(\boldsymbol{x}) \le v(\boldsymbol{x}) + w(\boldsymbol{x})$ with a nonlinear convex $w(\boldsymbol{x})$ (here we assume $w(\boldsymbol{x})$ is nonlinear, because adding a linear $w(\boldsymbol{x})$ does not affect the CCP at all) and apply CCP, the constraint would be replaced by

$$u(\boldsymbol{x}) + w(\boldsymbol{x}) \le v(\boldsymbol{x}_k) + \nabla v(\boldsymbol{x}_k)^T (\boldsymbol{x} - \boldsymbol{x}_k) \\ + w(\boldsymbol{x}_k) + \nabla w(\boldsymbol{x}_k)^T (\boldsymbol{x} - \boldsymbol{x}_k)$$

i.e.,

$$u(\boldsymbol{x}) + e(\boldsymbol{x}, \boldsymbol{x}_k) \le v(\boldsymbol{x}_k) + \nabla v(\boldsymbol{x}_k)^T (\boldsymbol{x} - \boldsymbol{x}_k) \quad (17)$$

where, due to the convexity of $w(\boldsymbol{x})$, $e(\boldsymbol{x}, \boldsymbol{x}_k) = w(\boldsymbol{x}) - w(\boldsymbol{x}_k) - \nabla w(\boldsymbol{x}_k)^T (\boldsymbol{x} - \boldsymbol{x}_k)$ is convex and always nonnegative.

On comparing (17) with (16), we note that a point $\boldsymbol{x}$ that satisfies constraint (17) also satisfies constraint (16), but the converse does not hold (unless $w(\boldsymbol{x})$ is a linear function which make $e(\boldsymbol{x}, \boldsymbol{x}_k)$ vanish). In other words, adding a redundant convex component $w(\boldsymbol{x})$ to the decomposition $f(\boldsymbol{x}) = u(\boldsymbol{x}) - v(\boldsymbol{x})$ shrinks the feasible region in a CCP-based method, hence imposing the risk of losing good solution candidates.

With above analysis in mind, we now examine the convexification steps made in Sec. III.B. For illustration purposes, here we focus on the treatment of constraint (11b) since the same analysis also applies to (11c). By writing (11b) as

$$\frac{1}{2}[\boldsymbol{a}_f^T \quad \boldsymbol{a}_m^T]\begin{bmatrix} \boldsymbol{0} & \boldsymbol{t}_f(L\omega)\boldsymbol{t}_m(\omega)^T \\ \boldsymbol{t}_m(\omega)\boldsymbol{t}_f(L\omega)^T & \boldsymbol{0} \end{bmatrix}\begin{bmatrix} \boldsymbol{a}_f \\ \boldsymbol{a}_m \end{bmatrix} \leq \delta_w + H_d(\omega) \tag{18}$$

we see that the left-hand side of (18) is a quadratic function with an indefinite Hessian, hence nonconvex. Therefore, constraint (11b) does not fit into the form in (3b), and an adequate convex term needs to be added to both sides of (18) to make CCP applicable. Intuitively, the quadratic expression in (18) suggests to add the term

$$\frac{1}{2}[\boldsymbol{a}_f^T \quad \boldsymbol{a}_m^T]\begin{bmatrix} \boldsymbol{t}_f(L\omega)\boldsymbol{t}_f(L\omega)^T & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{t}_m(\omega)\boldsymbol{t}_m(\omega)^T \end{bmatrix}\begin{bmatrix} \boldsymbol{a}_f \\ \boldsymbol{a}_m \end{bmatrix} \tag{19}$$

which itself is convex and precisely equal to $0.5s(\boldsymbol{x}, \omega)$ (see (12) for the definition of $s(\boldsymbol{x}, \omega)$). In doing so, (18) is led to

$$\frac{1}{2}[\boldsymbol{a}_f^T \quad \boldsymbol{a}_m^T]\begin{bmatrix} \boldsymbol{t}_f(L\omega)\boldsymbol{t}_f(L\omega)^T & \boldsymbol{t}_f(L\omega)\boldsymbol{t}_m(\omega)^T \\ \boldsymbol{t}_m(\omega)\boldsymbol{t}_f(L\omega)^T & \boldsymbol{t}_m(\omega)\boldsymbol{t}_m(\omega)^T \end{bmatrix}\begin{bmatrix} \boldsymbol{a}_f \\ \boldsymbol{a}_m \end{bmatrix} \leq \frac{1}{2}s(\boldsymbol{x}, \omega) + \delta_w + H_d(\omega) \tag{20}$$

where the functions on both sides are convex, hence fitting nicely into (3b). It is important to realize that the function on the left-hand side of (20) is convex but not strictly convex because its Hessian is merely a rank-one matrix, implying that adding anything less than that of (19) will not lead to a formulation suitable for CCP. Clearly, (20) is identical to (13a), hence the above explains the convexification steps in Sec. III.B.

IV. Minimax Design of FRM Filters

A. The Design Problem

Following (2), let the transfer functions of the periodic filter and two masking filters of an FRM filter be given by

$$F(z) = \sum_{n=0}^{N-1} f_n z^{-n}, \; M_a(z) = \sum_{n=0}^{N_a-1} m_n^{(a)} z^{-n} \quad \text{and} \quad M_c(z) = \sum_{n=0}^{N_c-1} m_n^{(c)} z^{-n} \tag{21}$$

respectively, the zero-phase frequency response of a single-stage FRM filter can be expressed as

$$H(\boldsymbol{x}, \omega) = [\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)][\boldsymbol{a}_a^T \boldsymbol{t}_a(\omega) - \boldsymbol{a}_c^T \boldsymbol{t}_c(\omega)] + \boldsymbol{a}_c^T \boldsymbol{t}_c(\omega) \tag{22}$$

where $\boldsymbol{a}_f, \boldsymbol{a}_a$, and $\boldsymbol{a}_c$ are coefficient vectors associated with filters $F(z)$, $M_a(z)$ and $M_c(z)$ in Fig. 2 and $\boldsymbol{t}_f(\omega)$, $\boldsymbol{t}_a(\omega)$ and $\boldsymbol{t}_c(\omega)$ are respective trigonometric vectors determined by the lengths and types of the FIR filters involved. Given a desired zero-phase frequency response $H_d(\omega)$, a frequency-weighted minimax design seeks to find variable vector $\boldsymbol{x} = [\boldsymbol{a}_f^T \quad \boldsymbol{a}_a^T \quad \boldsymbol{a}_c^T]^T$ that solves the problem

$$\underset{\boldsymbol{x}}{\text{minimize}} \max_{\omega \in \Omega} w(\omega)|H(\boldsymbol{x}, \omega) - H_d(\omega)| \tag{23}$$

where $w(\omega) > 0$ is a frequency-selective weight defined over a frequency region of interest, Ω. From (22), it is evident that (23) is a nonconvex problem with respect to design variable $\boldsymbol{x}$.

B. A CCP approach to solving (23)

By bounding the objective in (23) from above by δ and treating the bound as an additional design variable, we arrive at

$$\text{minimize} \;\; \delta \tag{24a}$$

$$\text{s.t.} \; H(\boldsymbol{x}, \omega) - \delta_w - H_d(\omega) \leq 0, \; \omega \in \Omega \tag{24b}$$

$$-H(\boldsymbol{x}, \omega) - \delta_w + H_d(\omega) \leq 0, \; \omega \in \Omega \tag{24c}$$

where $\delta_w = \delta / w(\omega)$. We now take an approach similar to that of Sec. III.B to reformulate (24). By adding the convex term

$$v(\boldsymbol{x}, \omega) = [\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)]^2 + \tfrac{1}{2}[\boldsymbol{a}_a^T \boldsymbol{t}_a(\omega)]^2 + \tfrac{1}{2}[\boldsymbol{a}_c^T \boldsymbol{t}_c(\omega)]^2 \tag{25}$$

to both sides of (24b) and (24c), we obtain an equivalent pair of constraints as

$$u_1(\boldsymbol{x}, \omega) \leq v(\boldsymbol{x}, \omega) \tag{26a}$$

$$u_2(\boldsymbol{x}, \omega) \leq v(\boldsymbol{x}, \omega) \tag{26b}$$

where

$$u_1(\boldsymbol{x}, \omega) = \tfrac{1}{2}(\boldsymbol{a}_{fa}^T \boldsymbol{P}_0 \boldsymbol{a}_{fa} + \boldsymbol{a}_{fc}^T \boldsymbol{Q}_1 \boldsymbol{a}_{fc}) + \boldsymbol{a}_c^T \boldsymbol{t}_c(\omega) - \delta_w - H_d(\omega) \tag{26c}$$

and

$$u_2(\boldsymbol{x}, \omega) = \tfrac{1}{2}(\boldsymbol{a}_{fa}^T \boldsymbol{P}_1 \boldsymbol{a}_{fa} + \boldsymbol{a}_{fc}^T \boldsymbol{Q}_0 \boldsymbol{a}_{fc}) - \boldsymbol{a}_c^T \boldsymbol{t}_c(\omega) - \delta_w + H_d(\omega) \tag{26d}$$

with

$$\boldsymbol{a}_{fa} = \begin{bmatrix} \boldsymbol{a}_f \\ \boldsymbol{a}_a \end{bmatrix} \quad \text{and} \quad \boldsymbol{a}_{fc} = \begin{bmatrix} \boldsymbol{a}_f \\ \boldsymbol{a}_c \end{bmatrix}$$

are convex with respect to $\boldsymbol{x}$ and δ because their Hessian matrices are characterized by positive semidefinite blocks $\boldsymbol{P}_i = \boldsymbol{p}_i \boldsymbol{p}_i^T$ and $\boldsymbol{Q}_i = \boldsymbol{q}_i \boldsymbol{q}_i^T$ with

$$\boldsymbol{p}_i = \begin{bmatrix} \boldsymbol{t}_f(L\omega) \\ (-1)^i \boldsymbol{t}_a(\omega) \end{bmatrix} \text{ and } \boldsymbol{q}_i = \begin{bmatrix} \boldsymbol{t}_f(L\omega) \\ (-1)^i \boldsymbol{t}_c(\omega) \end{bmatrix} \text{ for } i = 0, 1.$$

Consequently, the constraints in (26a) and (26b) fit into the form of (3b). By applying CCP to (26), we obtain a pair of convex constraints

$$u_1(\boldsymbol{x}, \omega) \leq \tilde{v}(\boldsymbol{x}, \boldsymbol{x}_k, \omega)$$

$$u_2(\boldsymbol{x}, \omega) \leq \tilde{v}(\boldsymbol{x}, \boldsymbol{x}_k, \omega)$$

where

$$\tilde{v}(\boldsymbol{x}, \boldsymbol{x}_k, \omega) = v(\boldsymbol{x}_k, \omega) + \nabla^T v(\boldsymbol{x}_k, \omega)(\boldsymbol{x} - \boldsymbol{x}_k)$$

with

$$\nabla v(\boldsymbol{x},\omega) = \begin{bmatrix} 2(\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega))\boldsymbol{t}_f(L\omega) \\ (\boldsymbol{a}_a^T \boldsymbol{t}_a(\omega))\boldsymbol{t}_a(\omega) \\ (\boldsymbol{a}_c^T \boldsymbol{t}_c(\omega))\boldsymbol{t}_c(\omega) \end{bmatrix}$$

where for practical purposes ω is taken from a finite and dense frequency grids $\Omega_d = \{\omega_j, j = 1,\ 2,\ \ldots,\ K\} \subset \Omega$. In summary, the convex problem to be solved in the kth iteration of CCP is given by

$$\begin{aligned} &\text{minimize} \ \ \delta && \text{(27a)} \\ &\text{s.t.} \ u_1(\boldsymbol{x},\omega_j) \le \tilde{v}(\boldsymbol{x},\boldsymbol{x}_k,\omega_j),\ 1 \le j \le K && \text{(27b)} \\ &\quad u_2(\boldsymbol{x},\omega_j) \le \tilde{v}(\boldsymbol{x},\boldsymbol{x}_k,\omega_j),\ 1 \le j \le K && \text{(27c)} \end{aligned}$$

Like the case of IFIR filter design, this problem of minimizing a linear function subject to convex quadratic constraints can be formulated as an SDP or SOCP problem [24], which can be solved efficiently [25]-[27].

We remark that the Hessian of functions $u_1(\boldsymbol{x},\omega)$ and $u_2(\boldsymbol{x},\omega)$ in (27) are of rank-two and positive semidefinite, but not positive definite as can be seen from (26c) and (26d). Therefore, an argument similar to that in Sec. III.C can be applied to conclude that constraints (27b) and (27c) define largest feasible region relative to other CCP-compatible options as discussed in Sec. III.C. A step-by-step summary of the algorithm is given below.

Algorithm 2 for FRM filters

Step 1: input $\boldsymbol{x}_0, (N, N_a, N_c), H_d(\omega)$ for $\omega \in \Omega_d$, $w(\omega)$, and K_i.

Step 2: for $k = 0,\ 1,\ \ldots,\ K_i - 1$
solve (27) for $\boldsymbol{x}_{k+1}$
end

Step 3: output $\boldsymbol{x}^* = \boldsymbol{x}_{K_i}$

V. FRM FILTERS WITH REDUCED COMPLEXITY

FRM filters are widely considered computationally efficient [3], [4], [9]. Variants of FRM filters with further reduced complexity have also been proposed in the literature. Unlike these variants, in this section we propose a CCP algorithm which is actually a modified version of that developed in Sec. IV for the design of FRM filters that simultaneously promotes sparsity of the impulse responses of the subfilters involved so as to reduce implementation complexity. Here an FIR filter $H_1(z)$ is said to more sparse than filter $H_2(z)$ of same length, if the impulse response of $H_1(z)$ contains more zero entries than that of $H_2(z)$. The algorithm proposed below consists of two phases:

A. Design Phase 1

The aim of phase 1 is to identify the locations (i.e. indices) of filter coefficients that may be set to zero without substantially affecting filter's performance. Motivated by the fact that for most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution [28], [29], we propose to promote the sparsity of an FRM filter by solving a modified version of (27) where the objective function combines upper bound with a weighted l_1-norm of design variable $\boldsymbol{x}$, namely,

$$\begin{aligned} &\text{minimize} \ \ \delta + \mu\|\boldsymbol{x}\|_1 && \text{(28a)} \\ &\text{s.t.} \ u_1(\boldsymbol{x},\omega_j) \le \tilde{v}(\boldsymbol{x},\boldsymbol{x}_k,\omega_j),\ 1 \le j \le K && \text{(28b)} \\ &\quad u_2(\boldsymbol{x},\omega_j) \le \tilde{v}(\boldsymbol{x},\boldsymbol{x}_k,\omega_j),\ 1 \le j \le K && \text{(28c)} \end{aligned}$$

where weight $\mu > 0$ controls the trade-off between error bound δ and sparsity of filter coefficients. Once the solution $\boldsymbol{x}_s = [\boldsymbol{a}_f^T \quad \boldsymbol{a}_a^T \quad \boldsymbol{a}_c^T]^T$ of problem (28) is obtained, a prescribed threshold $\varepsilon > 0$ is used to identify an index set $I_o = \{I_{af},\ I_{aa},\ I_{ac}\}$ as follows:

$$\begin{aligned} &I_{af} = \{i : |\boldsymbol{a}_f(i)| \le \varepsilon\},\ I_{aa} = \{i : |\boldsymbol{a}_a(i)| \le \varepsilon\}, \\ &I_{ac} = \{i : |\boldsymbol{a}_c(i)| \le \varepsilon\} \end{aligned} \qquad (29)$$

Concerning the selection of parameters μ and ε, in principle they should be small: μ should be small because otherwise the solution of (28) becomes less relevant to the design objective which is to minimize the error bound δ; and ε should be small because the use of a large ε would set some intrinsically nonzero coefficients to zero, hence inevitably do harm to the filter performance. We stress that the goal of phase 1 is merely to identify an index set to nullify the associated filter coefficients, and it is a collective action of μ and ε that identifies an appropriate index set: the introduction of term $\mu\|\boldsymbol{x}\|_1$ promotes the sparsity in $\boldsymbol{x}$, and the size of the index set is controlled by threshold ε.

B. Design Phase 2

The design now proceeds with a second phase in that the remaining nonzero entries of the impulse responses are optimally tuned by minimizing the same error bound as in (27a), but subject to additional constraints that the filter coefficients with indices in set $\{I_{af},\ I_{aa},\ I_{ac}\}$ are equal to zero. Namely, we solve

$$\begin{aligned} &\text{minimize} \ \ \delta && \text{(30a)} \\ &\text{s.t.} \ u_1(\boldsymbol{x},\omega_i) \le \tilde{v}(\boldsymbol{x},\boldsymbol{x}_k,\omega_i),\ \text{for } i = 1,\ldots,K && \text{(30b)} \\ &\quad u_2(\boldsymbol{x},\omega_i) \le \tilde{v}(\boldsymbol{x},\boldsymbol{x}_k,\omega_i),\ \text{for } i = 1,\ldots,K && \text{(30c)} \\ &\quad \boldsymbol{a}_f(i) = 0 \ \text{for } i \in I_{af} && \text{(30d)} \\ &\quad \boldsymbol{a}_a(i) = 0 \ \text{for } i \in I_{aa} && \text{(30e)} \\ &\quad \boldsymbol{a}_c(i) = 0 \ \text{for } i \in I_{ac} && \text{(30f)} \end{aligned}$$

Since the additional constraints are all linear equalities, problem (30) remains convex and can be solved efficiently. The algorithm proposed above is outlined below.

Algorithm 3 for sparse FRM filters

Step 1: input $\boldsymbol{x}_0, (N, N_a, N_c), \Omega_d, H_d(\omega)$ for $\omega \in \Omega_d$, $w(\omega), \mu, \varepsilon$, and K_i.

Step 2: for $k = 0,\ 1,\ \ldots,\ K_i - 1$
solve (28) to obtain $\boldsymbol{x} = [\boldsymbol{a}_f^T\ \boldsymbol{a}_a^T\ \boldsymbol{a}_c^T]^T$
apply (29) to $\boldsymbol{x}$ to obtain I_{af}, I_{aa}, and I_{ac}
solve (30) to obtain $\boldsymbol{x}_{k+1}$
end

Step 3: output $\boldsymbol{x}^* = \boldsymbol{x}_{K_i}$

VI. Design Examples

We now present design examples to illustrate the design methods proposed above.

A. Design and Evaluation Settings

A common goal of the designs presented below is to construct a linear-phase FIR digital system with narrow transition band, whose implementation efficiency is achieved via an IFIR or FRM structure. We follow the convention to evaluate the performance of each design in terms of peak-to-peak passband ripple A_p (dB) and minimum stopband attenuation A_a (dB) [4] which are defined by

$$A_p = 20\log_{10}\left(\frac{p_{max}}{p_{min}}\right)$$

and

$$A_a = -20\log_{10}(a_{max})$$

where p_{max} and p_{min} denote the maximum and minimum $|H(e^{j\omega})|$ over passband, respectively, and a_{max} denotes the maximum $|H(e^{j\omega})|$ over stopband.

As a part of performance evaluation, each design is compared with one or more designs made by existing algorithms from the literature. In addition, these designs are compared with their counterparts obtained using conventional FIR structure, see part D below.

B. Design of IFIR Filters

Example 1

Algorithm 1 was applied to design a lowpass IFIR filter with normalized passband edge $\omega_p = 0.15\pi$, stopband edge $\omega_a = 0.2\pi$. The sparsity factor was set to $L = 4$, and orders of $F(z)$ and $M(z)$ were set to 31 and 17, respectively. The frequency weight $w(\omega)$ was set to $w(\omega) \equiv 1$ for ω in the passband and $w(\omega) \equiv 2$ for ω in the stopband. An initial $\boldsymbol{x}_0 = [\boldsymbol{a}_{f0}^T \;\; \boldsymbol{a}_{m0}^T]^T$ was generated by the standard technique proposed in [3]. A total of $K = 1400$ frequency grids were uniformly placed in $[0,\ \omega_p]\bigcup[\omega_a,\ \pi]$ to form the discrete set Ω_d for problem (15). It took the algorithm 91 iterations to converge to an IFIR filter with $A_p = 0.0317$ dB and $A_a = 60.84$ dB. The same design problem was addressed as Example 10.29 in [4] using the method described in [30]. The method was implemented as function `ifir` in the Signal Processing Toolbox of MATLAB. With `[F,M] = ifir(4,low, [0.15, 0.2], [0.002, 0.001],advanced)`, the function returns with optimized impulses of filter $F(z)$ of order 31 and $M(z)$ of order 17 (in Example 10.29 of [4], the order of $M(z)$ was said to be 16, however the order of $M(z)$ produced by the above MATLAB code was actually 17), with $A_p = 0.0340$ dB and $A_a = 60.18$ dB. The passband and stopband amplitude responses of the two designs are depicted as solid and dashed lines in Fig. 3a and 3b, respectively.

Example 2

Reference [9] presents an example (Example 10.2) where a highpass IFIR filter with sampling frequency $\omega_s = 16000$Hz, stopband edge $\omega_a = 6600$Hz and passband edge $\omega_p = 7200$Hz is designed. In the design, the interpolator is fixed to $M(z) = (1 - z^{-1})^4$ and sparsity factor is set to $L = 2$ while a linear-phase subfilter $F(z)$ of order 20 is optimized in minimax sense. The performance of the IFIR filter is given by $A_p = 0.9061$ dB and $A_a = 40.78$ dB.

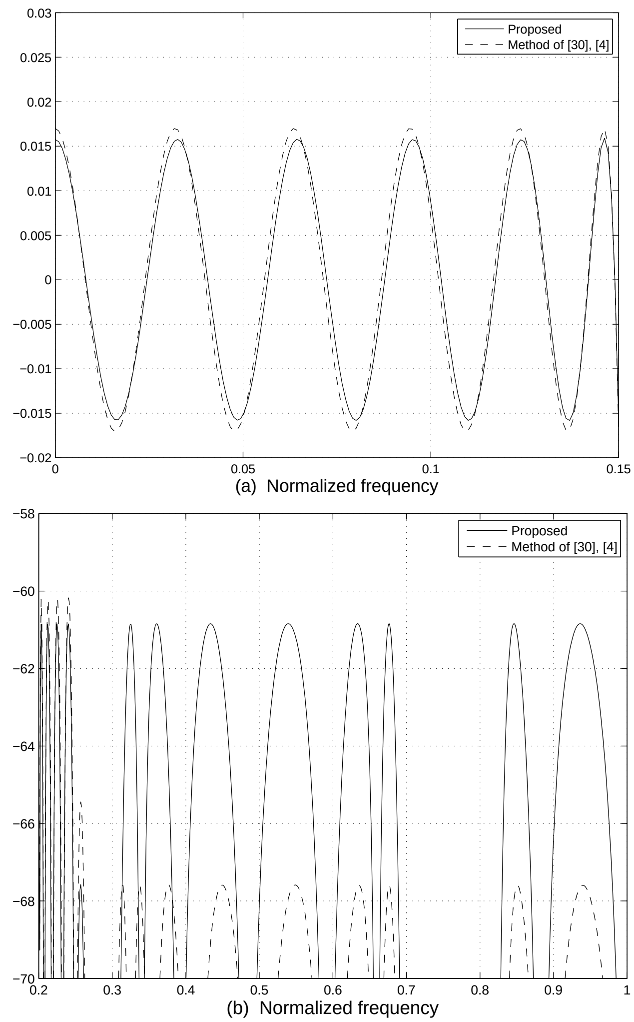

Fig. 3. Amplitude response (in dB) of the IFIR filters for Example 1 by the proposed algorithm (solid line) and the method of [30] and [4] (dashed line) in (a) passband and (b) stopband.

As a follow-up of the above example, Algorithm 1 was applied to jointly optimize an $F(z)$ of order 20 and an $M(z)$ of order 4 for a highpass IFIR filter $H(z) = F(z^L)M(z)$ with the same design specifications as Example 10.2 of [9]. The frequency weight $w(\omega)$ was set to $w(\omega) \equiv 1$ for ω in the passband and $w(\omega) \equiv 4.5$ for ω in the stopband. An initial $\boldsymbol{x}_0 = [\boldsymbol{a}_{f0}^T \; \boldsymbol{a}_{m0}^T]^T$ was generated by the standard technique proposed in [3]. A total of $K = 1400$ frequency grids were uniformly placed in [0, 6600Hz] $\bigcup$[7200Hz, 8000Hz] to form the discrete set Ω_d for problem (15). It took the algorithm 14 iterations to converge to an IFIR filter with $A_p = 0.6914$ dB and $A_a = 41.07$ dB. The amplitude responses of the two designs in the entire baseband and over its passband are depicted respectively as solid and dashed lines in Fig. 4a and 4b.

For the sake of a more efficient implementation, $M(z)$ is normalized by dividing it with the coefficient of its constant term, m_0, so that both coefficients of the constant term and the 4th-order term become unity. The transfer function $F(z^L)$ is rescaled to $m_0F(z^L)$ so that the overall transfer function $H(z)$ remains unaltered. The first eleven coefficients of $F(z)$ and first three coefficients of $M(z)$ after the normalization are shown in Table I. We stress that implementing the interpolator $M(z)$ in Example 10.2 of [9] requires no multiplications, therefore the performance gain of the proposed design over that of [9] was achieved at a cost of two more multiplications per output.

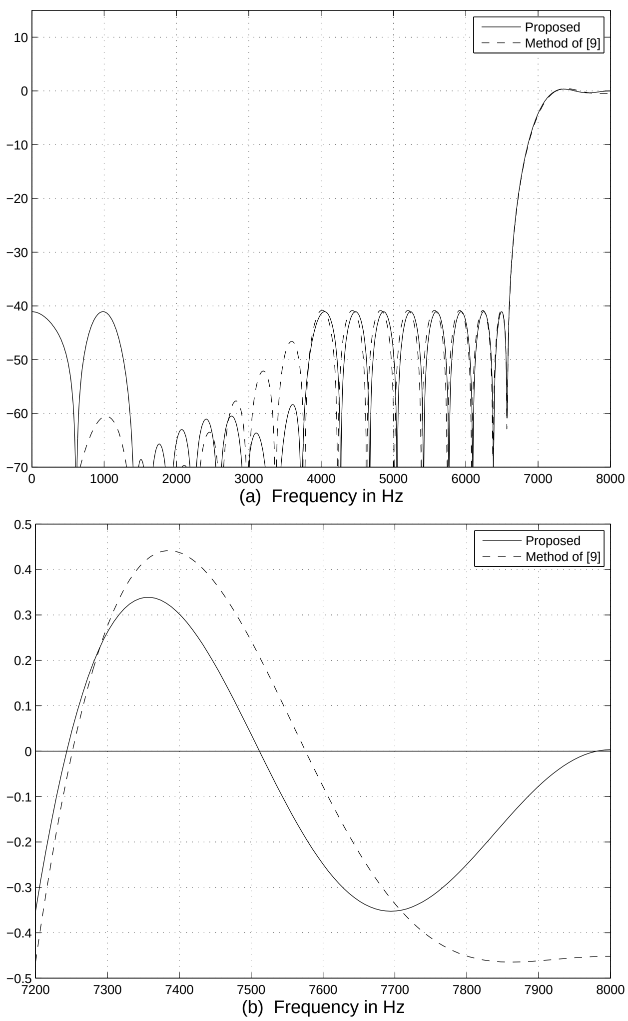

Fig. 4. Amplitude response (in dB) of the IFIR filters in Example 2 by the proposed algorithm (solid line) and the method of [9] (dashed line) in (a) entire baseband and (b) passband.

C. Design of FRM Filters

Example 3

Algorithm 2 was applied to design a lowpass FRM filter with the same design specifications as in the first example in [3] and [10]. The normalized passband and stopband edges were $\omega_p = 0.6\pi$ and $\omega_a = 0.61\pi$. The sparsity factor was set to $L = 9$, and orders of $F(z)$, $M_a(z)$, and $M_c(z)$ were 44, 40, and 32, respectively. A trivial weight $w(\omega) \equiv 1$ was utilized. With $K = 1100$, it took the algorithm 87 iterations to converge to an FRM filter with $A_p = 0.1320$ dB and $A_a = 42.49$ dB, which are favorably compared with those achieved in [3] ($A_p = 0.1792$ dB and $A_a = 40.96$ dB), which has been a benchmark for FRM filters, and those reported in [11] ($A_p = 0.1348$ dB and $A_a = 42.25$ dB). The amplitude response of the FRM filter in the entire baseband and passband is shown in Fig. 5.

TABLE I
COEFFICIENTS OF $F(z)$ AND $M(z)$ FOR EXAMPLE 2

f_i for $i = 0, 1, \dots 10$	m_i for $i = 0, 1, 2$
0.002287418334499	1
0.001299308145379	−2.439354581245607
0.001489820681472	2.965711835392111
−0.001928211344492	
−0.004121111010416	
−0.005691224464356	
−0.002031403630543	
0.005260711732820	
0.015907596775075	
0.024282544840202	
0.028106104367025	

Example 4

Algorithm 2 was applied to design a lowpass FRM filter with the same passband, stopband, and sparsity factor as in Example 3, i.e., $\omega_a = 0.6\pi$, $\omega_p = 0.61\pi$, and $L = 9$. However, the orders of $F(z)$, $M_a(z)$, and $M_c(z)$ were reduced to 42, 36, and 28, respectively. With $w(\omega) \equiv 2.1$ and $K = 1000$, it took the algorithm 600 iterations to converge to an FRM filter with $A_p = 0.2600$ dB and $A_a = 43.01$ dB. The amplitude response of the FRM filter in the entire baseband and passband is shown in Fig. 6. The above design specifications coincide with a design presented in [16] based on a neural network approach. From the numerical results proveded in Tables 1–3 of [16], it was found that $A_p = 0.2652$ dB (instead of 0.0672 dB as reported in [16] as there was a deep notch at passband edge 0.6π) and $A_a = 42.42$ dB.

Example 5

Algorithm 2 was also applied to design a lowpass FRM filter with orders of $F(z)$, $M_a(z)$, and $M_c(z)$ being (56, 30, 24), $L = 7$, and normalized passband and stopband edges being $\omega_p = 0.65\pi$ and $\omega_a = 0.66\pi$, respectively. With weight $w(\omega) \equiv 1$ and $K = 2400$, it took the algorithm 94 iterations to converge to an FRM filter with $A_p = 0.1510$ dB and $A_a = 41.22$ dB. The amplitude response of the FRM filter in the entire baseband and passband is shown in Fig. 7. For comparison, in [10] lowpass FRM filters with the same passband and stopband edges and interpolation factor L are designed using a weighted least-squares Chebyshev approach. With orders of $F(z)$, $M_a(z)$, and $M_c(z)$ being (56, 32, 26), the performance of the FRM filters are reported in terms of $A_p = 0.1960$ dB and $A_a = 40.11$ dB for a filter named Filter 4 and $A_p = 0.1920$dB and $A_a = 40.44$dB for a filter named Filter 5.

Example 6

Algorithm 3 was applied to re-design the FRM filter pre-

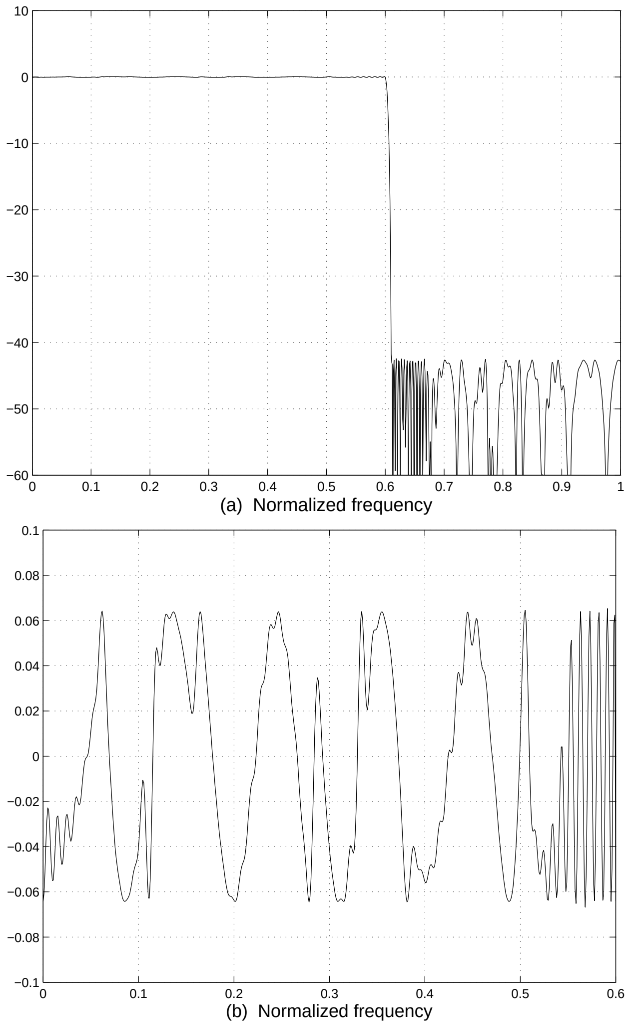

Fig. 5. Amplitude response (in dB) of the FRM filter in Example 3 in (a) entire baseband and (b) passband.

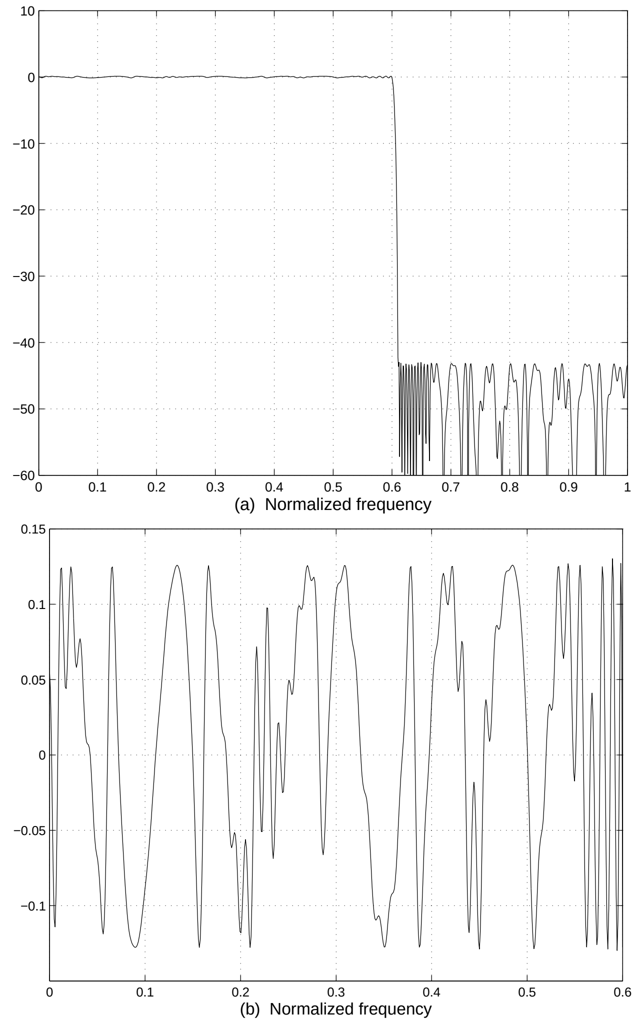

Fig. 6. Amplitude response (in dB) of the FRM filter in Example 4 in (a) entire baseband and (b) passband.

sented in Example 5 so as to reduce its complexity while maintain a comparable performance. With weight $w(\omega) \equiv 1$, $K = 1000$, and $\mu = 0.05$, phase 1 of the design was carried out using 30 iterations of (28) to yield a coefficient vector $\boldsymbol{x}_s = [\boldsymbol{a}_f^T \quad \boldsymbol{a}_a^T \quad \boldsymbol{a}_c^T]^T$. With a threshold $\varepsilon = 0.0006$, the three index sets defined in (29) were identified as $I_{af} = \{13,\ 18,\ 25\}$, $I_{aa} = \{11,\ 15,\ 16\}$, $I_{ac} = \{11,\ 13\}$, indicating there are a total of eight coefficients that may be set to zero values without substantially degrading the filter's performance provided that the remaining coefficients are optimized in the second phase of the design. Proceeding with design phase 2 with weight $w(\omega) \equiv 1$ and $K = 2400$, it took 10 iterations of (30) to converge to an FRM filter with $A_p = 0.1629$dB and $A_a = 40.56$dB, in that eight coefficients whose indices are identified above have been set to zero. The amplitude response of the FRM filter in the entire baseband and passband is shown in Fig. 8.

D. Comparisons with Conventional FIR Filters

The six designs presented above are compared with their conventional counterparts that are linear-phase equiripple FIR filters designed using the Parks-McClellan (P-M) algorithm [4] with practically the same performance as those in Examples 1–6 in terms of respective A_p and A_a. The comparisons are made in terms of computational complexity measured by the number of multiplications required per output sample and the overall group delay of the filter, see Table II.

TABLE II
COMPARISONS WITH CONVENTIONAL FIR FILTERS

Design	IFIR/FRM filters		P-M FIR filters [4]	
	Multiplications	Group delay	Multiplications	Group delay
1	25	70.5	65	64
2	13	22	22	21
3	60	218	210	209
4	56	207	193	192
5	58	211	202	201
6	50	211	198	197

It is observed from Table II that in all design instances the IFIR and FRM filters designed by the proposed algorithms

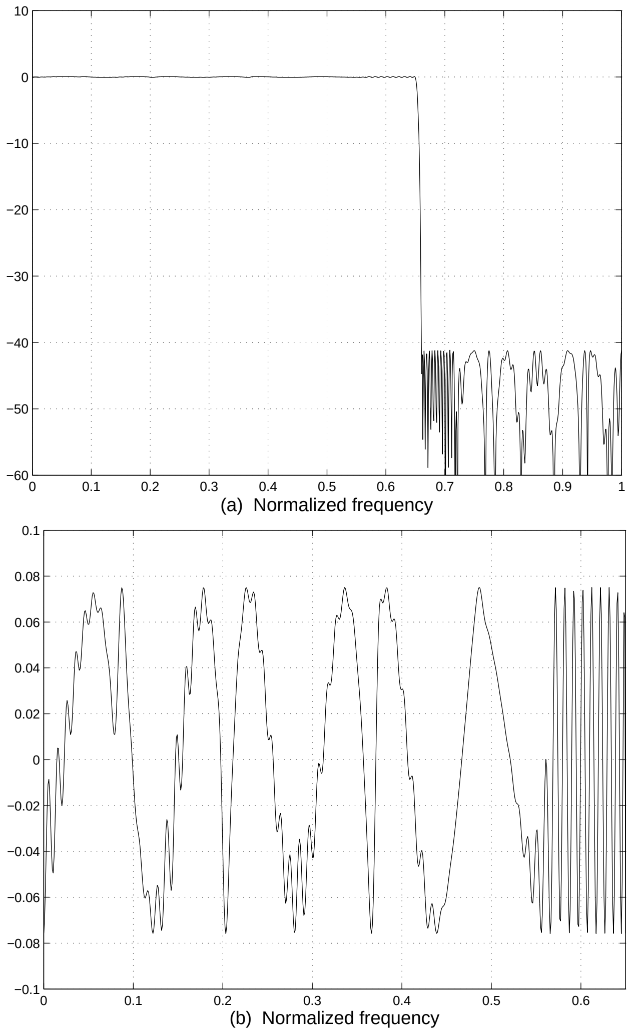

Fig. 7. Amplitude response (in dB) of the FRM filter in Example 5 in (a) entire baseband and (b) passband.

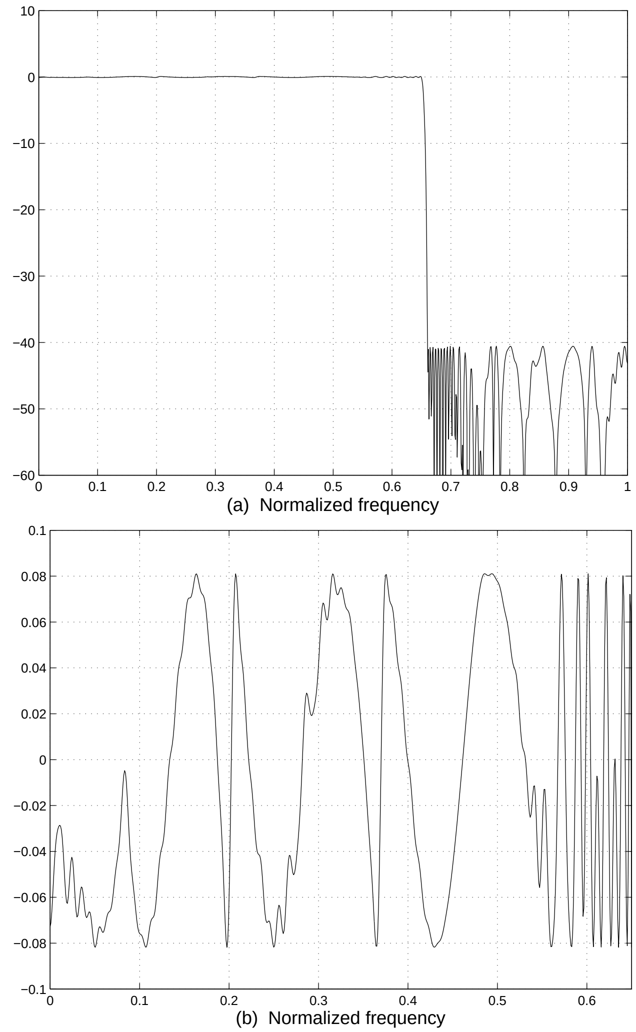

Fig. 8. Amplitude response (in dB) of the FRM filter in Example 6 in (a) entire baseband and (b) passband.

offer considerably improved computational efficiency at the cost of slightly increased group delay.

E. Discussion

In all six cases, the designs presented above have demonstrated consistently improved performance relative to their counterparts obtained by several well-established existing algorithms, with varying degree of improvement. The performance gain appears to be more pronounced over those that are not entirely optimization-based (Example 2) or not jointly optimized (Example 5), while becomes less so over those that are also jointly optimized (Examples 1, 3, 4). In the latter case, we attribute the ability of the proposed method to push the envelope to the fact that the CCP approach is especially tailored to the design of both IFIR and FRM filters as explained in Sec. III.C and Sec. IV.B.

VII. CONCLUSION

We have proposed a unified approach based on CCP to the design of minimax IFIR and FRM filters. The design method is conceptually simple and produces designs that are shown to provide satisfactory performance relative to those available from the literature. In addition, the proposed design method is shown to allow extension to the design of FRM filters that promotes sparsity of filter coefficients to reduce implementation complexity without substantially degrading filter's performance. Design examples have been presented to illustrate the proposed design algorithms that compare favorably with several known design techniques. Extensions of CCP-based designs to multistage FRM filters is possible, but developing a CCP type of convexification to maintain largest feasible region turns out to be challenging.

REFERENCES

[1] Y. Neuvo, C. Y. Dong, and S. K. Mitra, "Interpolated finite impulse response filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, pp. 563-570, June 1984.

[2] T. Saramäki, Y. Neuvo, and S. K. Mitra, "Design of computationally efficient interpolated FIR filters," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 70-88, Jan. 1988.

[3] Y. C. Lim, "Frequency-response masking approach for the synthesis of sharp linear phase digital filters," *IEEE Trans. Circuits Syst.*, vol. 33, pp. 357-364, Apr. 1986.

[4] S. K. Mitra, *Digital Signal Processing — A Computer-Based Approach*, 3rd ed., McGraw Hill, 2006.

[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*, vol. 40, 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*, vol. 41, pp. 518-525, Aug. 1994.

[7] T. Saramäki, Y. C. Lim, and R. Yang, "The synthesis of half-band filter using frequency-response masking thchnique," *IEEE Trans. Circuits Syst. II*, vol. 42, pp. 58-60, Jan. 1995.

[8] T. Saramäki and H. Johansson, "Optimization of FIR filters using frequency-response masking approach," in *Proc. ISCAS 2001*, vol. 2, pp. 177-180.

[9] P. S. R. Diniz, E. A. B. da Silva, and S. L. Netto, *Digital Signal Processing*, Cambridge University Press, 2002.

[10] L. C. R. de Barcellos, S. L. Netto, and P. S. R. Diniz, "Optimization of FRM filters using the WLS-Chebyshev approach," *Circuits, Syst. and Signal Process.*, vol. 22, no. 2, pp. 99-113, Mar. 2003.

[11] W.-S. Lu and T. Hinamoto, "Optimal design of frequency-response-masking filters using semidedinite programming," *IEEE Trans. Circuits Syst. I*, vol. 50, pp. 557-568, Apr. 2003.

[12] T. Saramäki, J. Yli-Kaakinen, and H. Johansson, "Optimization of frequency-response masking based FIR filters," *J. Circuits, Syst., Comput.*, vol. 12, pp. 563-589, May 2003.

[13] W.-S. Lu and T. Hinamoto, "Optimal design of frequency-response-masking filters using second-order cone programming," in *Proc. ISCAS 2004*, vol. 3, pp. 878-881.

[14] Y. Liu and Z. Lin, "Optimal design of frequency-response masking filters with reduced group delays," *IEEE Trans. Circuits Syst. I*, vol. 55, no. 6, pp. 1560-1570, July 2008.

[15] Y. Wei and D. Liu, "Improved design of frequency-response masking filters using band-edge shaping filter with non-periodical frequency response," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3269-3278, July 2013.

[16] X.-H. Wang and Y.-G. He, "A neural network approach to FIR filter design using frequency-response masking technique," *Signal Process.*, vol. 88, pp. 2917-2926, 2008.

[17] J. Yli-Kaakinen and T. Saramäki, "An efficient alorithm for the optimization of FIR filters synthesized using the multistage frequency-response masking approach," *Circuits, Syst. and Signal Process.*, vol. 30, no. 1, pp. 157-183, 2011.

[18] Y. Wei, S. Huang, and X. Ma, "A novel approach to design low-cost two-stage frequency-response masking filters," *IEEE Trans. Circuits Syst. II*, vol. 62, no. 10, pp. 982-986, Oct. 2015.

[19] A. L. Yuille and A Rangarajan, "The concave-convex procedure," *Neural Computation*, vol. 15, no. 4, pp. 915-936, 2003.

[20] T. Lipp and S. Boyd, "Variations and extensions of the convex-concave procedure," *Research Report*, Stanford University, Aug. 2014.

[21] B. K. Sriperumbudur and G. R. Lanckriet, "On the convergence of the concave-convex procedure," in *Advances in Neural Information Process. Systems*, pp. 1759-1767, 2009.

[22] W.-S. Lu and T. Hinamoto, "A unified approach to the design of interpolated and frequency-response-masking FIR filters," in *Proc. ISCAS 2016*, pp. 2174-2177.

[23] P. Hartman, "On functions representable as a difference of convex functions," *Pacific J. of Math.*, vol. 9, no. 3, pp. 707-713, 1959.

[24] A. Antoniou and W.-S. Lu, *Practical Optimazation: Algorithms and Engineering Applications*, Springer, 2007.

[25] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11-12, pp. 625-633, 1999.

[26] R. H. Tütüncü, K. C. Toh, and M. J. Todd, "Solving semidefinite-quadratic-linear programs using SDPT3," *Mathematical Programming, Series B*, vol. 9, pp. 189-217, 2003.

[27] M. Grant, S. Boyd, and Y. Ye, "Disciplined convex programming," in *Global Optimization: from Theory to Implementation, Nonconvex Optimization and Its Applications*, L. Liberti and N. Maculan, eds., Springer, 2006.

[28] D. L. Donoho, "For most largest underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution," *Comm. Pure Applied Math.*, vol. 59, no. 6, pp. 797-829, June 2006.

[29] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Magazine*, vol. 25, no. 2, pp. 21-30, Mar. 2008.

[30] T. Saramäki, "Finite impulse response filter design," in *Handbook for Digital Signal Processing*, S. K. Mitra and J. F. Kaiser eds., Wiley-Interscience, New York NY, 1993.

Wu-Sheng Lu (S'81-M'85-SM'90-F'99-LF'12) received his undergraduate 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 University of Minnesota, Minneapolis, USA, in 1983 and 1984, respectively. He was a post-doctoral fellow at University of Victoria, Victoria, B.C., Canada, in 1985, and a visiting assistant professor at University of Minnesota from January 1986 to April 1987. Since May 1987, he has been with the Electrical and Computer Engineering Department, University of Victoria where he is a professor. His current research interests include analysis and design of digital filters, signal and image processing, and methods of convex optimization. He is the co-author with A. Antoniou of Two-Dimensional Digital Filters (Marcel Dekker, 1992) and Practical Optimization: Algorithms and Engineering Applications (Springer, 2007). Dr. Lu served as editor for the Canadian Journal of Electrical and Computer Engineering and associate editor for several journals including IEEE Transactions on Circuits and Systems I and II, International Journal of Multidimensional Systems and Signal Processing, and Journal of Circuits, Systems, and Signal Processing. Dr. Lu 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, Kobe, Japan. From 1979 to 1981, he was a Visiting Member of Staff in the Department of Electrical Engineering, Queen's University, Kingston, ON, Canada, on leave from Kobe University. During 1988-1991, he was Professor of Electronic Circuits in the Faculty of Engineering, Tottori University, Tottori, Japan. During 1992-2009, he was Professor of Electronic Control in 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 at Hiroshima Institute of Technology, Hiroshima, Japan. His research interests include digital signal processing, system theory, and control engineering. He has published over 430 papers in these areas. He is co-editor and co-author of Two-Dimensional Signal and Image Processing (Tokyo, Japan: SICE, 1996). Dr. Hinamoto is a Fellow of the Institue of Electronics, Information and Communication Engineers (IEICE) and a Fellow of the Society of Instrument and Control Engineers (SICE).