# Optimal Design of IIR Frequency-Response-Masking Filters Using Second-Order Cone Programming

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

***Abstract*—The frequency-response-masking (FRM) technique proposed by Lim has proven effectiveness for the design of very sharp digital filters with reduced implementation complexity compared to other options. In this paper, we propose a constrained optimization method for the design of basic and multistage FRM filters where the prototype filters are of infinite-impulse response (IIR) with prescribed pole radius. The design is accomplished through a sequence of linear updates for the design variables with each update carried out using second-order cone programming. Computer simulations have demonstrated that the class of IIR FRM filters investigated in the paper offers an attractive alternative to its finite-impulse response counterpart in terms of filter performance, system delay, and realization complexity.**

***Index Terms*—Frequency response masking (FRM), infinite-impulse response (IIR) filters, robust stability, second-order cone programming (SOCP).**

## I. Introduction

THE frequency-response-masking (FRM) technique proposed by Lim [1] has proven effectiveness for the design of digital filters with narrow transition bands that can be implemented with reduced complexity compared to other options [2]–[10]. As illustrated in Fig. 1(a), a *basic* FRM *filter* consists of a *prototype filter* $H_a(z)$ with $z$ replaced by $z^M$, a pair of masking filters $\{H_{ma}(z), H_{mc}(z)\}$, and a delay line of $M \cdot D$ delay units with $D$ matching the group delay of the prototype filter. For additional reduction of implementation complexity, the prototype filter itself may be realized with a basic FRM filter, and if necessary, one can repeat this to construct a *multistage* FRM filter. Fig. 1(b) illustrates a two-stage FRM filter, where factor $M$ at the first and second stages become $M_1$ and $M_2$, respectively.

Most of the work on FRM filters to date has been focused on finite-impulse response (FIR) filters [1]–[9], primarily because linear phase response can be readily achieved when all subfilters in an FRM filter are of FIR. Infinite-impulse response (IIR) filters are known to have improved selectivity and implementation efficiency, as well as reduced passband group delay relative to their FIR counterparts. On the other hand, nontrivial IIR filters do not have precise linear phase response and stability is often an issue that makes the design more complicated. In the context of FRM filters, employing an IIR prototype filter appears to be an attractive option for the following reasons.

1) As high selectivity can be readily achieved by low-order IIR filters, FRM filters with low-order IIR prototype filters can offer satisfactory performance as well as further reduction in implementation cost.
2) If an IIR prototype filter, whose passband group delay is $r$ samples less than that of its FIR counterpart, is used in a basic FRM filter, then the passband group delay of the FRM filter is reduced by $M \cdot r$ samples. The reduction in passband group is even greater as the number of filter stages grows. Since very sharp FIR FRM filters always have large group delay, which is undesirable in many applications, the class of FRM filters with IIR prototype filters offers a better alternative.
3) Although in principle, one may consider designing all-IIR FRM filters, the advantages gained by using IIR masking filters are not as great as that of IIR prototype filter, especially in terms of group delay reduction. Moreover, the design of an all-IIR quickly becomes too involved as the number of filter stages increases.

On the other hand, if one adopts the filter structure in Fig. 1, where the prototype filter is of IIR but the masking filters remain linear-phase FIR, then, only *one* IIR filter is involved in the design regardless of the number of filter stages used. With this filter structure, the design becomes more tractable and, as will be presented in the subsequent section, a design methodology applicable to both basic and multistage IIR FRM filters can be developed.

In [10], a method for the design of recursive FRM filters with two allpass filters (called model filters) replacing the prototype filter and the delay line is proposed. The design in [10] is accomplished using a two-stage approach in that a good initial point is obtained by separately optimizing the model filters and the masking filters, and then a second-stage optimization is carried out to finalize the design. Multistage recursive FRM filters were not considered.

There are algorithms that can be used for solving general nonlinear programming problems [24], [25]. However, since these solution methods do not take advantages of special problem structures such as convexity of the objective function and linear or low-order polynomial type constraints, they tend to be less efficient relative to those which consciously utilize as many desirable features as the problem at hand can offer.

In this paper, we propose a new constrained optimization method for the minimax design of recursive basic and multistage FRM filters in Lim's framework as illustrated in Fig. 1 where the prototype filter is the only IIR filter. The proposed design algorithm starts with a trivial initial point, and the coefficients of all

Manuscript received May 9, 2002; revised January 22, 2003. This paper was recommended by Associate Editor W.-P. Zhu.

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 Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima 739-8527, Japan (e-mail: hinamoto@ecl.sys.hiroshima-u.ac.jp).

Digital Object Identifier 10.1109/TCSI.2003.818613

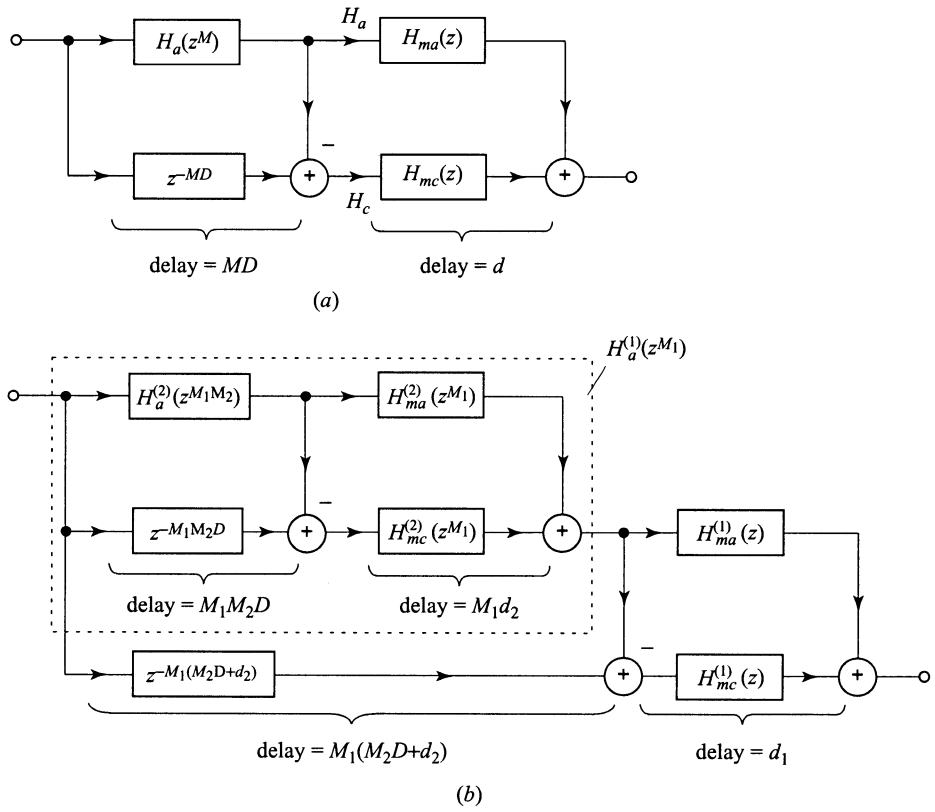



Fig. 1. (a) Basic FRM filter structure. (b) Two-stage FRM filter structure.

subfilters are jointly optimized through a sequence of linear updates with each update carried out using second-order cone programming (SOCP). SOCP is a class of well-structured convex programming problems that can be tackled using efficient interior-point solvers [11]–[17], as such the proposed algorithm can be used as a fast design tool for IIR FRM filters. Other features of our design method include the following.

- It imposes a norm constraint on the parameter update vector to validate a key linear approximation used in the design and eliminate a line search step usually required in nonlinear optimization. The constraint fits nicely into the SOCP formulation.
- By considering factorized denominator of the IIR prototype filter, a sufficient and near necessary condition for robust stability of the prototype filter is converted into a set of linear inequality constraints suitable for the SOCP formulation.
- It provides a framework where the designs of basic and multistage IIR FRM filters can be carried out in a similar manner.

Collectively, these features lead to designs with improved performance relative to their FIR counterparts.

The paper is organized as follows. Section II reviews some basic elements of SOCP and discusses the notion of robust stability triangle that are needed in the rest of the paper. Robust stability constraints for IIR digital filters that are well suited for the proposed SOCP formulation are described in Section III. Section IV presents an SOCP-based design methodology applicable to both basic and multistage IIR FRM filters. Algorithmic details for the design of basic and multistage FRM filters as well as simulation results are presented in Sections V and VI, respectively.

Throughout the paper, boldfaced characters denote matrices and vectors; $\boldsymbol{I}_r$ represents the identify matrix of dimension $r$; $\|\cdot\|$ denotes the standard Euclidean norm; $\omega_p$ and $\omega_a$ denote normalized passband and stopband edges, respectively; and the normalized base frequency band is denoted by $\Omega = \{\omega : -\pi \leq \omega \leq \pi\}$. For the sake of description simplicity, the term "IIR FRM filter" is in this paper referred to as the filter structure in Fig. 1 with an IIR prototype filter and linear-phase FIR masking filters; and the proposed design method will be illustrated in terms of lowpass filters although it is in principle applicable to other types of bandpass filters.

## II. Preliminaries

### A. SOCP

SOCP, which is sometimes called conic quadratic programming [14], [15], is a class of convex programming problems where a linear function is minimized subject to a set of second-order cone constraints [14], [16]

$$\text{minimize} \quad \boldsymbol{f}^T\boldsymbol{x} \tag{1a}$$

$$\text{subject to :} \quad \|\boldsymbol{A}_i\boldsymbol{x}+\boldsymbol{b}_i\| \leq \boldsymbol{c}_i^T\boldsymbol{x}+h_i, \qquad i=1,2,\ldots,N \tag{1b}$$

where $\boldsymbol{x} \in \mathcal{R}^{n\times 1}$, $\boldsymbol{f} \in \mathcal{R}^{n\times 1}$, $\boldsymbol{A}_i \in \mathcal{R}^{(n_i-1)\times n}$, $\boldsymbol{b}_i \in \mathcal{R}^{(n_i-1)\times 1}$, $\boldsymbol{c}_i \in \mathcal{R}^{n\times 1}$, and $h_i \in \mathcal{R}$. The term "cone" here reflects the fact that each constraint in (1b) is equivalent to a conic constraint

$$\begin{bmatrix}\boldsymbol{c}_i^T \\ \boldsymbol{A}_i\end{bmatrix}\boldsymbol{x} + \begin{bmatrix}h_i \\ \boldsymbol{b}_i\end{bmatrix} \in \mathcal{C}_i$$

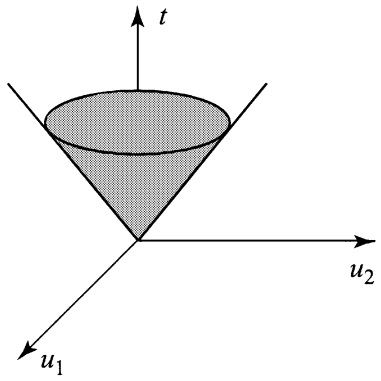



Fig. 2. Second-order cone in $\mathcal{R}^3$.

where $\mathcal{C}_i$ is the second-order cone in $\mathcal{R}^{n_i}$, i.e.,

$$\mathcal{C}_i = \left\{ \begin{bmatrix} t \\ \boldsymbol{u} \end{bmatrix} : \boldsymbol{u} \in \mathcal{R}^{(n_i-1)\times 1},\ t \geq 0,\ \|\boldsymbol{u}\| \leq t \right\}. \quad (2)$$

The second-order cone in (2) is also called ice-cream cone or Lorentz cone, see Fig. 2 which illustrates a second-order cone in $\mathcal{R}^3$.

From (1), it is evident that SOCP includes linear programming and convex quadratic programming as special cases. On the other hand, since each constraint in (1b) can be expressed as

$$\begin{bmatrix} (\boldsymbol{c}_i^T\boldsymbol{x} + h_i)\boldsymbol{I} & \boldsymbol{A}_i\boldsymbol{x} + \boldsymbol{b}_i \\ (\boldsymbol{A}_i\boldsymbol{x} + \boldsymbol{b}_i)^T & \boldsymbol{c}_i^T\boldsymbol{x} + h_i \end{bmatrix} \succeq \boldsymbol{0}$$

where $\boldsymbol{M} \succeq \boldsymbol{0}$ denotes that $\boldsymbol{M}$ is positive semidefinite, SOCP is a subclass of semidefinite programming (SDP) [16], [20]. Commercial and public-domain software based on interior-point optimization algorithms for SOCP and SDP are available [17]–[19]. It is important to stress, however, that in general, the problem in (1) can be solved more efficiently as an SOCP problem than solving it in an equivalent SDP setting [14]. In the subsequent sections, we attempt to formulate the design problems at hand as SOCP problems rather than SDP problems.

### *B. Robust Stability of a Second-Order Discrete-Time System*

Consider the transfer function of a second-order discrete-time system, whose denominator polynomial is given by

$$d(z) = z^2 + d_1 z + d_2. \quad (3)$$

It is well known that the system is stable if and only if coefficients $d_1$ and $d_2$ satisfy [21]

$$d_1 + d_2 + 1 > 0 \quad (4a)$$
$$-d_1 + d_2 + 1 > 0 \quad (4b)$$
$$-d_2 + 1 > 0 \quad (4c)$$

i.e.,

$$\boldsymbol{C}_2\boldsymbol{d} + \hat{\boldsymbol{e}} > \boldsymbol{0} \quad (5a)$$

where

$$\boldsymbol{C}_2 = \begin{bmatrix} 1 & 1 \\ -1 & 1 \\ 0 & -1 \end{bmatrix}, \quad \boldsymbol{d} = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \quad \hat{\boldsymbol{e}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \quad (5b)$$

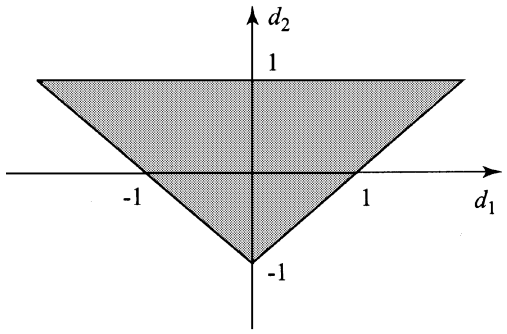



Fig. 3. Stability triangle.

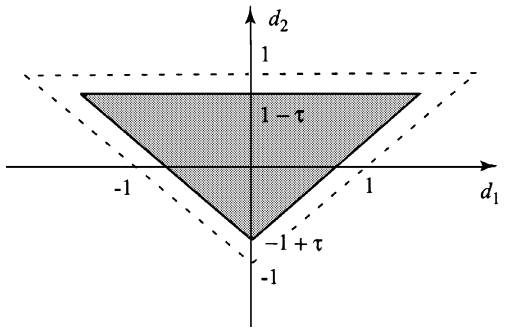



Fig. 4. Internal stability triangle.

Note that the constraints in (4) are *linear* with respect to $d_1$ and $d_2$, and characterize the triangle in the $(d_1, d_2)$-space shown in Fig. 3, which will be referred to as the *stability triangle*.

For the sake of robust stability, we consider a triangle in $(d_1, d_2)$-space that is strictly inside the stability triangle as shown in Fig. 4, where $\tau$ is a small positive scalar. The region enclosed with the internal triangle is characterized by three linear inequalities

$$d_1 + d_2 \geq -1 + \tau$$
$$-d_1 + d_2 \geq -1 + \tau$$
$$-d_2 \geq -1 + \tau$$

i.e.,

$$\boldsymbol{C}_2\boldsymbol{d} + (1-\tau)\hat{\boldsymbol{e}} \geq \boldsymbol{0} \quad (6)$$

where $\boldsymbol{C}_2$, $\boldsymbol{d}$, and $\hat{\boldsymbol{e}}$ are defined in (5b). Using an elementary analysis on the roots of $d(z)$ for $(d_1, d_2)$ going along the boundary of the internal stability triangle, it can be shown that all system poles that are associated with the internal stability triangle in Fig. 4 cover the most part of the disk with radius $\sqrt{1-\tau}$ in the $z$-plane, which is shown as the shaded region in Fig. 5. We shall refer to the internal stability triangle in Fig. 4 as a *robust stability triangle* as any point $(d_1, d_2)$ in the triangle corresponds to a second-order discrete-time system with a pole radius (defined as the maximum magnitude of the poles) no larger than $\sqrt{1-\tau}$. It is noticed that when the value of $\tau$ is small (which is always the case in filter design), the difference between the shaded region in Fig. 5 and the disk with radius $\sqrt{1-\tau}$ becomes insignificant. Therefore, restricting coefficients $(d_1, d_2)$ to within the robust stability triangle is sufficient and *near* necessary for $d(z)$ to have a stability margin $1 - \sqrt{1-\tau}$.

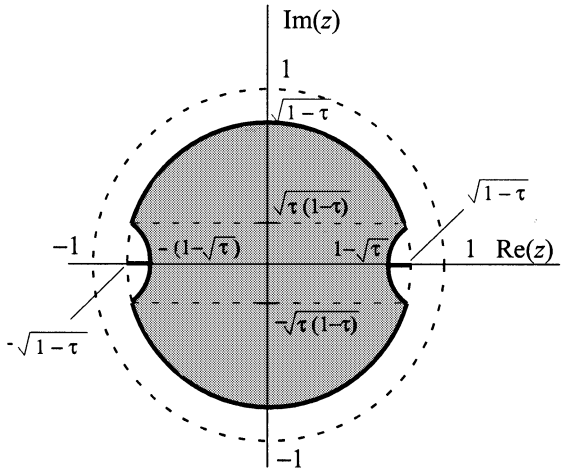



Fig. 5. Shaded region plus two short segments (in solid line) on the real axis represent the pole locations corresponding to the internal stability triangle in Fig. 4.

## III. Constraints for Robust Stability of IIR FRM Filters

Consider a basic IIR FRM filter shown in Fig. 1(a), where the transfer function of the prototype filter assumes the form

$$H_a(z) = \frac{a(z)}{z^{n-r}d(z)} \tag{7a}$$

where

$$a(z) = \sum_{i=0}^{n} a_i z^{n-i} \tag{7b}$$

where $d(z)$ is a polynomial of order $r$ expressed as a product of second-order sections (and a first-order section if $r$ is odd)

$$d(z) = \begin{cases} \prod_{i=1}^{\frac{r}{2}} \left(z^2 + d_{i1}z + d_{i2}\right), & \text{if } r \text{ even} \\ (z+d_o) \prod_{i=1}^{\frac{(r-1)}{2}} \left(z^2 + d_{i1}z + d_{i2}\right), & \text{if } r \text{ odd} \end{cases} \tag{7c}$$

and $r$ is an integer between 0 and $n$. The reason our design formulation uses the above form of denominator, namely $z^{n-r}d(z)$, is that assigning a certain number of poles at the origin was found beneficial for the design of several types of digital filters as observed in [22].

Define vector $\boldsymbol{d}$ as

$$\boldsymbol{d} = \begin{bmatrix} d_0 \\ \boldsymbol{d}_1 \\ \vdots \\ \boldsymbol{d}_L \end{bmatrix} \quad d_i = \begin{bmatrix} d_{i1} \\ d_{i2} \end{bmatrix}, \qquad \text{for } 1 \le i \le L \tag{8a}$$

$$L = \begin{cases} \frac{r}{2}, & \text{if } r \text{ even} \\ \frac{(r-1)}{2}, & \text{if } r \text{ odd} \end{cases} \tag{8b}$$

where $d_0$ presents only if $r$ is odd, and assume polynomial $d(z)$ is robustly stable in the sense that for a given $\tau > 0$ parameter vector $\boldsymbol{d}$ in (8) satisfies

$$-1 + \tau \le d_0 \le 1 - \tau \tag{9a}$$

$$\boldsymbol{C}_2\boldsymbol{d}_i + (1-\tau)\hat{\boldsymbol{e}} \ge \boldsymbol{0} \quad \text{for } 1 \le i \le L \tag{9b}$$

where $\boldsymbol{C}_2$ and $\hat{\boldsymbol{e}}$ are defined in (5b). From Section II-B, it follows that such a $d(z)$ has a stability margin $1-\sqrt{1-\tau}$. Now, suppose vector $\boldsymbol{d}$ is updated to

$$\boldsymbol{d} + \boldsymbol{\delta}_d = \begin{bmatrix} d_0 + \delta_0 \\ \boldsymbol{d}_1 + \boldsymbol{\delta}_1 \\ \vdots \\ \boldsymbol{d}_L + \boldsymbol{\delta}_L \end{bmatrix} \tag{10}$$

and we want the denominator polynomial associated with $\boldsymbol{d}+\boldsymbol{\delta}_d$ to remain robustly stable with the same stability margin, then the constraints in (9) become

$$-1 + \tau \le d_0 + \delta_0 \le 1 - \tau$$
$$\boldsymbol{C}_2(\boldsymbol{d}_i + \boldsymbol{\delta}_i) + (1-\tau)\hat{\boldsymbol{e}} \ge \boldsymbol{0}, \qquad \text{for } 1 \le i \le L$$

i.e.,

$$\hat{\boldsymbol{C}}(\boldsymbol{d} + \boldsymbol{\delta}_d) + (1-\tau)\boldsymbol{e} \ge \boldsymbol{0} \tag{11}$$

where $\boldsymbol{e} = [1 \cdots 1]^T \in \mathcal{R}^{m\times 1}$ with $m = 3L + 2$, and

$$\hat{\boldsymbol{C}} = \begin{bmatrix} \boldsymbol{c}_1 & & \\ & \boldsymbol{C}_2 & \boldsymbol{0} \\ & \ddots & \\ \boldsymbol{0} & & \boldsymbol{C}_2 \end{bmatrix}_{m\times r}$$

with $\boldsymbol{c}_1 = [1\ -1]^T$ (if $r$ is even, then the top-left $\boldsymbol{c}_1$ in $\hat{\boldsymbol{C}}$ does not present and $m = 3L$). The constraint in (11) can be expressed as

$$\hat{\boldsymbol{C}}\boldsymbol{\delta}_d + \boldsymbol{h} \ge \boldsymbol{0} \tag{12}$$

where $\boldsymbol{h} = \hat{\boldsymbol{C}}\boldsymbol{d} + (1-\tau)\boldsymbol{e}$, a linear inequality constraint for updated denominator polynomial to maintain a stability margin $1 - \sqrt{1-\tau}$.

## IV. Design Methodology

This section presents a general design methodology applicable to both basic and multistage IIR FRM filters. The design goal is an FRM filter, whose prototype filter is an IIR filter with prescribed pole radius, that achieves sharp bandpass-type frequency response with reduced passband group delay as well as reduced implementation complexity relative to its FIR counterpart.

Let $H(\omega, \boldsymbol{x})$ be the frequency response of an IIR FRM filter of frequency $\omega$ and parameter vector $\boldsymbol{x} \in \mathcal{R}^{p\times 1}$, and $H_d(\omega)$ be the desired frequency response. We seek to determine a vector $\boldsymbol{x}$ that solves the constrained weighted minimax optimization problem

$$\underset{\boldsymbol{x}}{\text{minimize}} \left\{ \underset{\omega\in\Omega}{\text{minimize}}\ W(\omega)\left|H(\omega,\boldsymbol{x}) - H_d(\omega)\right| \right\} \tag{13a}$$

$$\text{subject to : } H(\omega,\boldsymbol{x}) \text{ stable.} \tag{13b}$$

For a filter structure as illustrated in Fig. 1 with an IIR $H_a(z)$, the optimization problem in (13) is highly nonlinear. In what follows, we present a solution method that converts (13) into a solvable SOCP problem.

If $\eta$ denotes an upper bound of $W(\omega)|H(\omega,\boldsymbol{x}) - H_d(\omega)|$ on $\Omega$, then, the problem in (13) can be converted into

$$\text{minimize } \eta \tag{14a}$$

$$\text{subject to : } W(\omega)|H(\omega,\boldsymbol{x}) - H_d(\omega)| \le \eta, \qquad \text{for } \omega \in \Omega \tag{14b}$$

$$H(\omega,\boldsymbol{x}) \text{ stable.} \tag{14c}$$

Suppose we have a reasonable initial point $\boldsymbol{x}_0$ to start, and we are now in the $k$th iteration. For a smooth $H(\omega,\boldsymbol{x})$ in the vicinity of current point $\boldsymbol{x}_k$, we can write

$$H(\omega,\boldsymbol{x}_k+\boldsymbol{\delta}) \approx H(\omega,\boldsymbol{x}_k)+\boldsymbol{g}_k^T(\omega)\boldsymbol{\delta}$$

provided that

$$||\boldsymbol{\delta}|| \text{ is small} \tag{15a}$$

where $\boldsymbol{g}_k(\omega)$ is the gradient of $H(\omega,\boldsymbol{x})$ with respect to $\boldsymbol{x}$ and evaluated at $\boldsymbol{x}_k$. Thus, for $\boldsymbol{x}=\boldsymbol{x}_k+\boldsymbol{\delta}$ with $\boldsymbol{\delta}$ subject to (15a), we have

$$\begin{aligned} &W(\omega)\,|H(\omega,\boldsymbol{x})-H_d(\omega)| \\ &\quad\approx W(\omega)\left|\boldsymbol{g}_k^T(\omega)\boldsymbol{\delta}+[H(\omega,\boldsymbol{x}_k)-H_d(\omega)]\right|. \end{aligned} \tag{15b}$$

For the filter design at hand, $H(\omega,\boldsymbol{x}_k)$ and $H_d(\omega)$ are complex-valued, and we need to define

$$H(\omega,\boldsymbol{x}) = H_r(\omega,\boldsymbol{x})+jH_i(\omega,\boldsymbol{x}) \tag{16a}$$
$$H_d(\omega) = H_{rd}(\omega)+jH_{id}(\omega) \tag{16b}$$
$$\boldsymbol{g}_k(\omega) = \boldsymbol{g}_{rk}(\omega)+j\boldsymbol{g}_{ik}(\omega). \tag{16c}$$

It follows that

$$\begin{aligned} W(\omega)\,|H(\omega,\boldsymbol{x})-H_d(\omega)| &\approx \left|W(\omega)\left[\boldsymbol{g}_{rk}^T(\omega)\boldsymbol{\delta}+e_{rk}(\omega)\right]\right. \\ &\quad \left.+jW(\omega)\left[\boldsymbol{g}_{ik}^T(\omega)\boldsymbol{\delta}+e_{ik}(\omega)\right]\right| \\ &= ||\boldsymbol{G}_k(\omega)\boldsymbol{\delta}+\boldsymbol{e}_k(\omega)|| \end{aligned} \tag{17}$$

where

$$\begin{aligned} \boldsymbol{G}_k(\omega) &= W(\omega)\begin{bmatrix}\boldsymbol{g}_{rk}^T(\omega)\\ \boldsymbol{g}_{ik}^T(\omega)\end{bmatrix} \\ \boldsymbol{e}_k(\omega) &= W(\omega)\begin{bmatrix}e_{rk}(\omega)\\ e_{ik}(\omega)\end{bmatrix} \\ e_{rk}(\omega) &= H_r(\omega,\boldsymbol{x}_k)-H_{rd}(\omega) \\ e_{ik}(\omega) &= H_i(\omega,\boldsymbol{x}_k)-H_{id}(\omega). \end{aligned}$$

In the light of (14b), (15a), and (17), we see that an approximate solution in the $k$th iteration can be obtained by solving the constrained optimization problem

$$\text{minimize } \eta \tag{18a}$$
$$\text{subject to : } ||\boldsymbol{G}_k(\omega)\boldsymbol{\delta}+\boldsymbol{e}_k(\omega)|| \le \eta, \quad \text{for } \omega\in\Omega_d \tag{18b}$$
$$||\boldsymbol{\delta}|| \text{ small} \tag{18c}$$
$$H(\omega,\boldsymbol{x}_k+\boldsymbol{\delta}) \text{ stable} \tag{18d}$$

where $\Omega_d=\{\omega_i, 1\le i\le N\}\subset\Omega$ is a set of dense grid points placed in the frequency region of interest.

For a $K$-stage IIR FRM filter, parameter vector $\boldsymbol{x}_k$ collects the coefficients of all subfilters in the order

$$\boldsymbol{x}_k = \begin{bmatrix}\boldsymbol{a}\\ \boldsymbol{d}\\ \boldsymbol{a}_{Ka}\\ \boldsymbol{a}_{Kc}\\ \vdots\\ \boldsymbol{a}_{1a}\\ \boldsymbol{a}_{1c}\end{bmatrix} \begin{matrix}\left.\begin{matrix}\\ \\ \\ \\ \end{matrix}\right\}\text{stage } K\\ \vdots \\ \left.\begin{matrix}\\ \\ \end{matrix}\right\}\text{stage } 1\end{matrix} \tag{19}$$

where $\boldsymbol{a}$ and $\boldsymbol{d}$ are the coefficient vectors associated with the numerator and denominator of the IIR prototype filter, respectively, and $\boldsymbol{a}_{ia}$ and $\boldsymbol{a}_{ic}$ are the coefficient vectors associated with the FIR masking filters $\{H_{ma}^{(i)}(z), H_{mc}^{(i)}(z)\}$ in the $i$th stage.

Concerning the constraint in (18c), note that the order of the IIR prototype filter $(n,r)$ is usually considerably lower than that of the masking filters $\{H_{ma}^{(i)}(z), H_{mc}^{(i)}(z)\}$, therefore it is reasonable to control the "smallness" of their coefficients *separately*. To this end, we denote

$$\boldsymbol{\delta} = \begin{bmatrix}\boldsymbol{\delta}_a\\ \boldsymbol{\delta}_d\\ \boldsymbol{\delta}_{aK}\\ \boldsymbol{\delta}_{cK}\\ \vdots\\ \boldsymbol{\delta}_{a1}\\ \boldsymbol{\delta}_{c1}\end{bmatrix} \tag{20}$$

and impose

$$\left\|\begin{bmatrix}\boldsymbol{\delta}_a\\ \boldsymbol{\delta}_d\end{bmatrix}\right\| \le \beta_0, \quad \left\|\begin{bmatrix}\boldsymbol{\delta}_{ai}\\ \boldsymbol{\delta}_{ci}\end{bmatrix}\right\| \le \beta_i, \qquad \text{for } 1\le i\le K \tag{21}$$

where $\beta_i$ for $0\le i\le K$ are prescribed bounds to control the magnitude of $\boldsymbol{\delta}$.

The stability constraint in (18d) can be specified using (12). Furthermore, in order to prevent undersirable overshot in transition band we impose constraints on the magnitude of the FRM filter as

$$|H(\omega,\boldsymbol{x}_k+\boldsymbol{\delta})| \le \beta_t \quad \text{for } \omega\in\Omega_t \tag{22}$$

where $\Omega_t=\{\omega_i, 1\le i\le N_t\}$ is a set of grid points placed in the transition band and $\beta_t$ is a prescribed upper bound to eliminate transition overshot. The constraints in (22) can be approximated by the second-order cone constraints

$$\left\|\hat{\boldsymbol{G}}_k(\omega)\boldsymbol{\delta}+\boldsymbol{H}_k(\omega)\right\| \le \beta_r \quad \text{for } \omega\in\Omega_t \tag{23}$$

where

$$\hat{\boldsymbol{G}}_k(\omega) = \begin{bmatrix}g_{rk}^T(\omega)\\ g_{ik}^T(\omega)\end{bmatrix} \quad \boldsymbol{H}_k(\omega) = \begin{bmatrix}H_r(\omega,\boldsymbol{x}_k)\\ H_i(\omega,\boldsymbol{x}_k)\end{bmatrix}.$$

Replacing the constraints in (18c) and (18d) with that in (21) and (12), respectively, and imposing additional constraints in (23), the $k$th iteration of our design is carried out by solving the SOCP problem

$$\text{minimize } \eta \tag{24a}$$
$$\text{subject to : } ||\boldsymbol{G}_i(\omega)\boldsymbol{\delta}+\boldsymbol{e}_k(\omega)|| \le \eta, \quad \text{for } \omega\in\Omega_d \tag{24b}$$
$$\left\|\begin{bmatrix}\boldsymbol{\delta}_a\\ \boldsymbol{\delta}_d\end{bmatrix}\right\| \le \beta_0, \ \left\|\begin{bmatrix}\boldsymbol{\delta}_{ai}\\ \boldsymbol{\delta}_{ci}\end{bmatrix}\right\| \le \beta_i, \quad \text{for } 1\le i\le K \tag{24c}$$
$$\hat{\boldsymbol{C}}\boldsymbol{\delta}_d+\boldsymbol{h}\ge\boldsymbol{0} \tag{24d}$$
$$\left\|\hat{\boldsymbol{G}}_k(\omega)\boldsymbol{\delta}+\boldsymbol{H}_k(\omega)\right\| \le \beta_t, \quad \text{for } \omega\in\Omega_t. \tag{24e}$$

In (24), there are $N+N_t+K+1$ second-order cone constraints, and $m$ linear constraints (obviously, a linear inequality constraint can be treated as a trivial second-order cone constraint; however, efficient SOCP solvers (e.g., toolbox SeDuMi

[17]) often deal with linear constraints and second-order cone constraints separately).

Several interior-point methods for SOCP have been developed in the last several years, see, for example, [11]–[15], and [18]. Lucid exposition of the subject in terms of what can be expressed via conic quadratic constraints, interior-point polynomial-time methods and complexity analysis can be found in the recent book [16].

The original problem in (13) and, equivalently, the problem in (14) are highly nonlinear and nonconvex optimization problems. As such, the above method, if it converges, only provides a *local* minimizer for the problem; therefore, the design optimality considered in this paper is always in a local sense. Among other things, the performance of such a local solution depends largely on how the initial point is chosen. Fortunately, for FRM filter designs, a technique that generates a reasonably good initial point is available, see Sections V-C and VI-B. Concerning the convergence of the method, although a rigorous proof is presently not available, in our simulations, when the method was applied to design a variety of IIR FRM filters, we had not detected a single failure of convergence. One might attribute the success of the proposed method to three factors: 1) the global convergence of each sub-problem in (24) when an interior-point convex programming algorithm is applied; 2) the use of constraint (24c) that validates the key approximation in (15b); and 3) the use of a good initial point.

Another related issue is the convergence rate or, in a more general term, the computational efficiency. From the above description of the method, it is quite clear that the computational efficiency is determined by how efficient each individual SOCP problem in (24) is solved and how many linear updates are needed to reach a minimizer of (13). For the former, most of the algorithms that are presently available for solving the SOCP problem (24) are so-called polynomial-time algorithms, meaning that the amount of computations required is bounded by a polynomial of the data size [16]. Consequently, the computational complexity for problem (24) is affordable for today's computing devices even for designing relatively high-order IIR FRM filters, and it will increase only moderately when the size of the problem increases. For the latter, with a given set of bounds $\{\beta_i\}$ in constraint (24c), the number of updates needed depends on how far the initial point is from the minimizer.

It should also be pointed out that although problem (24) is merely an *approximation* of (13), as the iteration continues and the local minimizer gets closer, the increment vector $\boldsymbol{\delta}$ obtained by solving (24) gradually shrinks in magnitude and within a limited number of iterations it eventually becomes such a value that the updated solution point is practically the same as the true minimizer.

In summary, we have described a method for minimax optimization of an objective function that is frequently encountered in filter design problems and is allowed to be highly nonlinear. The method proposed here accomplishes the optimization through a sequence of linear updates where each update is solvable in an SOCP setting. The usefulness of this methodology will be demonstrated in the next two sections where IIR FRM filter design problems are addressed.

## V. Design of Basic IIR FRM Filters

### A. Frequency Response and Its Gradient

The reader is referred to Fig. 1(a) as the filter structure considered in this section, where the transfer function $H_a(z)$ is given by (7) and

$$H_{ma}(z) = \sum_{k=0}^{n_a-1} h_k^{(a)} z^{-k} \tag{25a}$$

$$H_{mc}(z) = \sum_{k=0}^{n_c-1} h_k^{(c)} z^{-k}. \tag{25b}$$

Throughout, it is assumed that the masking filters $H_{ma}(z)$ and $H_{mc}(z)$ have linear phase responses; the lengths $n_a$ and $n_c$ are either both even or both odd; and the group delays of $H_{ma}(z)$ and $H_{mc}(z)$ have been equalized to $d = \max\{(n_a-1)/2, (n_c-1)/2\}$. Under these circumstances, the desired passband group delay for the IIR FRM filter is

$$D_s = d + MD \tag{26}$$

where $D$ is the intended passband group delay of the prototype filter, and the frequency response of the FRM filter can be expressed as

$$e^{-jD_s\omega} H(\omega, \boldsymbol{x}) \tag{27}$$

where

$$\begin{aligned}
H(\omega,\boldsymbol{x}) &= \tilde{H}_a(M\omega)\left[\boldsymbol{a}_a^T\boldsymbol{c}_a(\omega) - \boldsymbol{a}_c^T\boldsymbol{c}_c(\omega)\right] + \boldsymbol{a}_c^T\boldsymbol{c}_c(\omega)\\
\tilde{H}_a(\omega) &= e^{jD\omega}\frac{a(\omega)}{d(\omega)} = \frac{\boldsymbol{a}^T\boldsymbol{v}(\omega)}{d(\omega)}\\
\boldsymbol{a} &= [a_0\ a_1\ \cdots\ a_n]^T \quad (see\ (7b))\\
\boldsymbol{v}(\omega) &= \boldsymbol{c}(\omega) - j\boldsymbol{s}(\omega)\\
\boldsymbol{c}(\omega) &= [\cos D\omega\ \cdots\ \cos(D-n)\omega]^T\\
\boldsymbol{s}(\omega) &= [\sin D\omega\ \cdots\ \sin(D-n)\omega]^T\\
v_1(\omega) &= \cos\omega - j\sin\omega\\
\boldsymbol{v}_2(\omega) &= \begin{bmatrix}\cos\omega\\ \cos 2\omega\end{bmatrix} - j\begin{bmatrix}\sin\omega\\ \sin 2\omega\end{bmatrix}
\end{aligned}$$

and

$$d(\omega) = \begin{cases} \prod_{i=1}^{L}\left[1+\boldsymbol{d}_i^T\boldsymbol{v}_2(\omega)\right], & \text{if } r \text{ even}\\ [1+d_0 v_{\mathbf{1}}(\omega)]\prod_{i=1}^{L}\left[1+\boldsymbol{d}_i^T\boldsymbol{v}_2(\omega)\right], & \text{if } r \text{ odd}\end{cases}$$

$$\boldsymbol{a}_a = \begin{cases} \left[h^{(a)}_{\frac{(n_a-1)}{2}}\ 0.5h^{(a)}_{\frac{(n_a+1)}{2}}\ \cdots\ 0.5h^{(a)}_{n_a-1}\right]^T, & \text{if } n_a \text{ odd}\\ 0.5\left[h^{(a)}_{\frac{n_a}{2}}\ \cdots\ h^{(a)}_{n_a-1}\right]^T, & \text{if } n_a \text{ even}\end{cases}$$

$$\boldsymbol{c}_a(\omega) = \begin{cases} \left[1\ \cos\omega\ \cdots\ \cos\left[(n_a-1)\frac{\omega}{2}\right]^T, & \text{if } n_a \text{ odd}\\ \left[\cos\left(\frac{\omega}{2}\right)\ \cdots\ \cos\left[(n_a-1)\frac{\omega}{2}\right]\right]^T, & \text{if } n_a \text{ even}\end{cases}$$

$$\boldsymbol{a}_c = \begin{cases} \left[h^{(c)}_{\frac{(n_c-1)}{2}}\ 0.5h^{(c)}_{\frac{(n_c+1)}{2}}\ \cdots\ 0.5h^{(c)}_{n_c-1}\right]^T, & \text{if } n_c \text{ odd}\\ 0.5\left[h^{(c)}_{\frac{n_c}{2}}\ \cdots\ h^{(c)}_{n_c-1}\right]^T, & \text{if } n_c \text{ even}\end{cases}$$

$$\boldsymbol{c}_c(\omega) = \begin{cases} \left[1\ \cos\omega\ \cdots\ \cos\left[(n_c-1)\frac{\omega}{2}\right]^T, & \text{if } n_c \text{ odd}\\ \left[\cos\left(\frac{\omega}{2}\right)\ \cdots\ \cos\left[(n_c-1)\frac{\omega}{2}\right]\right]^T, & \text{if } n_c \text{ even}\end{cases} \tag{28}$$

and the design variables are put together as parameter vector

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{d} \\ \boldsymbol{a}_a \\ \boldsymbol{a}_c \end{bmatrix} \tag{29}$$

with vector $\boldsymbol{d}$ defined by (8a).

It follows that the gradient of $H(\omega, \boldsymbol{x})$ in (28) is given by

$$\boldsymbol{g}(\omega, \boldsymbol{x}) = \begin{bmatrix} y(\omega)\frac{\partial \tilde{H}_a(M\omega)}{\partial \boldsymbol{a}} \\ y(\omega)\frac{\partial \tilde{H}_a(M\omega)}{\partial \boldsymbol{d}} \\ \tilde{H}_a(M\omega)\boldsymbol{c}_a(\omega) \\ \left[1 - \tilde{H}_a(M\omega)\right]\boldsymbol{c}_c(\omega) \end{bmatrix} \tag{30}$$

where

$$y(\omega) = \boldsymbol{a}_a^T \boldsymbol{c}_a(\omega) - \boldsymbol{a}_c^T \boldsymbol{c}_c(\omega)$$

$$\frac{\partial \tilde{H}_a(\omega)}{\partial \boldsymbol{a}} = \frac{\boldsymbol{v}(\omega)}{d(\omega)}$$

$$\frac{\partial \tilde{H}_a(\omega)}{\partial \boldsymbol{d}} = \begin{bmatrix} \frac{\partial \tilde{H}_a(\omega)}{\partial d_0} \\ \frac{\partial \tilde{H}_a(\omega)}{\partial \boldsymbol{d}_1} \\ \vdots \\ \frac{\partial \tilde{H}_a(\omega)}{\partial \boldsymbol{d}_L} \end{bmatrix}$$

with

$$\frac{\partial \tilde{H}_a(\omega)}{\partial d_0} = -\tilde{H}_a(\omega)\frac{v_1(\omega)}{1 + d_0 v_1(\omega)}$$

$$\frac{\partial \tilde{H}_a(\omega)}{\partial \boldsymbol{d}_i} = -\tilde{H}_a(\omega)\frac{\boldsymbol{v}_2(\omega)}{1 + \boldsymbol{d}_i^T \boldsymbol{v}_2(\omega)}, \qquad \text{for } 1 \le i \le L.$$

In the $k$th iteration, vector $\boldsymbol{g}_k(\omega)$ involved in (24b) can be evaluated using (30) at $\boldsymbol{x} = \boldsymbol{x}_k$.

### B. Desired Frequency Response and Weighting Function

Since the frequency response of the FRM filter is in the form of (27) with the desired phase response factored out, the desired frequency response for $H(\omega, \boldsymbol{x})$ in (27) is a zero-phase lowpass function given by

$$H_d(\omega) = \begin{cases} 1, & \text{for } 0 \le \omega \le \omega_p \\ 0, & \text{for } \omega_a \le \omega \le \pi \end{cases} \tag{31}$$

and the weighting function is typically a piecewise constant function given by

$$W(\omega) = \begin{cases} 1, & \text{for } 0 \le \omega \le \omega_p \\ \omega, & \text{for } \omega_a \le \omega \le \pi \\ 0 & \text{elsewhere} \end{cases} \tag{32}$$

where scalar $w \ge 0$ may assume a value greater or smaller than one to weigh the importance of the stopband relative to the passband.

### C. Initial Design

Given sampling factor $M$, normalized passband and stopband edges $\omega_p$ and $\omega_a$, and filter length $(n, r)$, $n_a$, and $n_c$, a reasonable initial design of subfilters $H_a(z)$, $H_{ma}(z)$, and $H_{mc}(z)$ can be readily prepared as follows.

*1) Prototype Filter $H_a(z)$:* From [1], the passband edge $\theta$ and stopband edge $\phi$ are given by

$$\theta = \omega_p M - 2m\pi \tag{33a}$$

$$\phi = \omega_a M - 2m\pi \tag{33b}$$

$$m = \left\lfloor \frac{\omega_p M}{2\pi} \right\rfloor \tag{33c}$$

where $\lfloor x \rfloor$ denotes the largest integer less than $x$, or by

$$\theta = 2m\pi - \omega_a \pi \tag{34a}$$

$$\phi = 2m\pi - \omega_p \pi \tag{34b}$$

$$m = \left\lceil \frac{\omega_a M}{2\pi} \right\rceil \tag{34c}$$

where $\lceil x \rceil$ denotes the smallest integer larger than $x$, depending on which set of $\{\theta, \phi\}$ satisfies $0 < \theta < \phi < \pi$.

Once $\{\theta, \phi\}$ is determined, an FIR filter of length $n+1$ that approximates the desired lowpass frequency response with passband edge $\theta$, stopband edge $\phi$, and passband group delay $D$ can be readily obtained using an established method (such as a Hamming-window method, see for example [21]). the initial design of $H_a(z)$ is then represented by its parameter vector

$$\begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{0} \end{bmatrix} \tag{35}$$

where $\boldsymbol{a}$ is the impulse response of the FIR filter.

*2) Masking Filters $H_{ma}(z)$ and $H_{mc}(z)$:* If (33) is used to determine the values of $\theta$ and $\phi$, then the passband and stopband edges of $H_{ma}(z)$ are given by $(2m\pi + \theta)/M$ and $[2(m+1)\pi - \phi]/M$, respectively, and the passband and stopband edges of $H_{mc}(z)$ are given by $(2m\pi - \theta)/M$ and $(2m\pi + \phi)/M$, respectively.

If (34) is used to determine the values of $\theta$ and $\phi$, then the passband and stopband edges of $H_{ma}(z)$ are given by $[2(m-1)\pi + \phi]/M$ and $(2m\pi - \theta)/M$, respectively, and the passband and stopband edges of $H_{mc}(z)$ are given by $(2m\pi - \phi)/M$ and $(2m\pi + \theta)/M$, respectively.

Once their passband and stopband edges are determined, the linear-phase FIR masking filters $H_{ma}(z)$ and $H_{mc}(z)$ can be designed using a Hamming-window method [21], and initial parameter vectors $\boldsymbol{a}_a$ and $\boldsymbol{a}_c$ can be obtained using (28). Hence an initial point $\boldsymbol{x}_0$ of form (29) is obtained.

### D. Placement of Grid Points

There are two issues to be addressed here: the number of total grid points in $\Omega_d$, namely the value of $N$, and how we place these grid points. Our design practice has indicated that for satisfactory design results $N$ has an empirical lower bound

$$N \ge 8 \cdot [(n + 1 - r) + 3r + \max(n_a, n_c)] \tag{36}$$

and relatively denser grid points should be placed in the regions near the passband and stopband edges. We recommend that 25% to 50% of the grid points be placed in the 10% of that band nearest to the band edge.

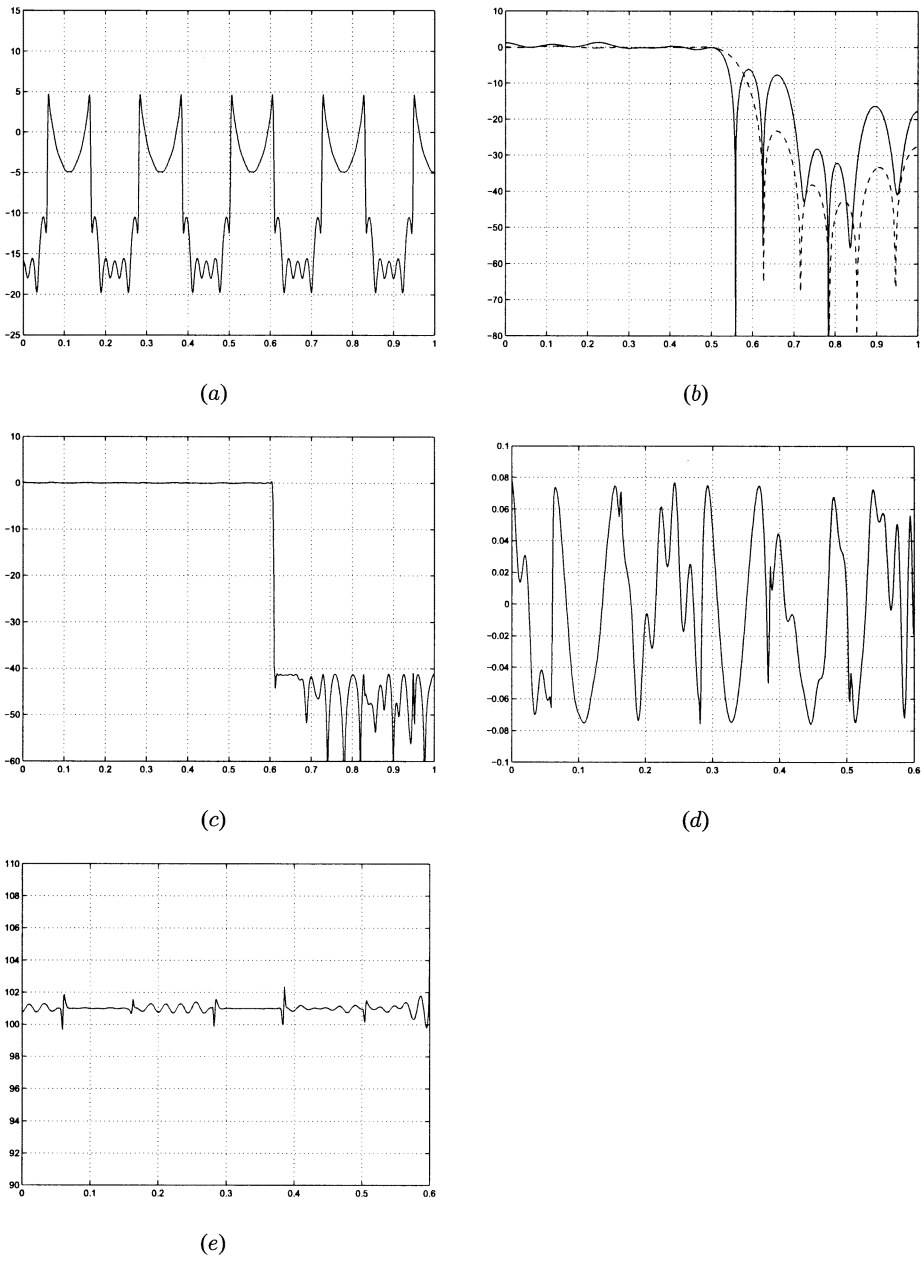



Fig. 6. Amplitude responses of (a) prototype filter $H_a(z^9)$, (b) masking filters $H_{ma}(z)$ (solid line) and $H_{mc}(z)$ (dashed line), (c) FRM filter. (d) Passband ripples of the FRM filter, all in decibels. (e) Passband group delay of the FRM filter.

### *E. A Design Example*

With the preparations made in Section V-A–D, the data required in (24) have been specified and we are now in a position to apply the design method outlined in Section IV to a basic IIR FRM filter. It is a lowpass filter with $n = 14$, $r = 10$, $n_a = 41$, $n_c = 33$, $M = 9$, $D = 9$, $\omega_p = 0.6\pi$, and $\omega_a = 0.61\pi$. Other parameters used in the design are stopband weight $w = 1$, stability parameter $\tau = 0.1$, $\beta_0 = 3 \cdot \sqrt{n+r+1} \cdot 10^{-4}$, $\beta_1 = 3 \cdot \sqrt{n_a + n_c} \cdot 10^{-4}$, $\beta_t = 1$, and $N_t = 4$.

A total of 1100 grid points were used in set $\Omega_d$. The SOCP problem was solved using MATLAB toolbox SeDuMi [17]. With 49 iterations, the algorithm converges to an IIR FRM filter with the amplitude responses of the subfilters and FRM filter shown in Fig. 6(a)–(c), the passband ripple in Fig. 6(d), and the passband group delay in Fig. 6(e). The maximum passband ripple was 0.0775 dB, the minimum stopband attenuation was 40.8921 dB, and the passband group delay was 102 samples with a 3.86% maximum relative deviation. The maximum magnitude of the poles of $H_a(z)$ was 0.9487. An FIR counterpart of the above FRM filter was presented in [1] where $H_a(z)$, $H_{ma}(z)$, and $H_{mc}(z)$ are linear-phase FIR filters of length of 45, 41, and 33, respectively. The values of $M$, $\omega_p$, and $\omega_a$ used in [1] are identical to those in our design. This implies a 218-sample group delay for the FIR FRM filter versus a considerably reduced 102-sample passband group delay [see Fig. 6(e)] for the current IIR FRM filter design. Improved passband ripple (0.0775 dB versus 0.0896 dB) and comparable stopband attenuation (40.8921 dB versus 40.96 dB) over the design in [1] are observed. Concerning the implementation complexity, since the proposed FRM filter has the same struc-

ture as that in [1] with identical masking filters, the difference in implementation is exclusively due to the replacement of linear-phase FIR filter of length $N = 45$ with an IIR filter of length $(n, r) = (14, 10)$ for the prototype filter $H_a(z)$. The basic IIR FRM filter requires a total of 63 multipliers and 99 adders while the FIR FRM filter requires 61 multipliers and 118 adders. Although finite wordlength effect is an issue of concern for IIR filters, we do not intend to address it in detail here but to simply remark that the prototype IIR filter is usually of low order which, in the implementation stage, can be factorized into product of a small number of stable second-order sections with each realized using a simple structure with low roundoff noise [26].

## VI. Design of Multistage IIR FRM Filters

For the sake of notation simplicity, the proposed design algorithm is described for a two-stage IIR FRM filter and the reader is referred to Fig. 1(b) as the filter structure. With straightforward modifications, however, the proposed design algorithm can be applied to IIR FRM filters with arbitrary number of stages.

### A. Frequency Response and Its Gradient

Suppose the prototype IIR filter, $H_a^{(2)}(z)$, assumes the form in (7) and the masking filters are given by

$$H_{ma}^{(i)} = \sum_{k=0}^{n_{ai}-1} h_k^{(a_i)} z^{-k} \tag{37a}$$

$$H_{mc}^{(i)} = \sum_{k=0}^{n_{ci}-1} h_k^{(c_i)} z^{-k} \tag{37b}$$

for $i = 1, 2$. Throughout we assume that all masking filters have linear phase responses; for each $i$ the lengths $n_{ai}$ and $n_{ci}$ are either both even or both odd; and the group delays of $H_{ma}^{(i)}(z)$ and $H_{mc}^{(i)}(z)$ have been equalized to $d_i = \max\{(n_{ai}-1)/2, (n_{ci}-1)/2\}$ for $i = 1, 2$. For simplicity we also assume $M_1 = M_2 = M$. Under these circumstances, the desired passband group delay for the two-stage IIR FRM filter is given by

$$D_{s2} = d_1 + Md_2 + M^2 D \tag{38}$$

and the frequency response of the FRM filter can be expressed as

$$e^{-jD_{s2}\omega} H(\omega, \boldsymbol{x}) \tag{39}$$

where

$$H(\omega, \boldsymbol{x}) = \tilde{H}_a^{(1)}(M\omega) y_1(\omega) + \boldsymbol{a}_{1c}^T \boldsymbol{c}_{1c}(\omega) \tag{40a}$$

$$\tilde{H}_a^{(1)}(\omega) = \tilde{H}_a^{(2)}(M\omega) y_2(\omega) + \boldsymbol{a}_{2c}^T \boldsymbol{c}_{2c}(\omega) \tag{40b}$$

$$\tilde{H}_a^{(2)}(\omega) = e^{jD\omega} \frac{a(\omega)}{d(\omega)} = \frac{\boldsymbol{a}^T \boldsymbol{v}(\omega)}{d(\omega)} \tag{40c}$$

$$y_1(\omega) = \boldsymbol{a}_{1a}^T \boldsymbol{c}_{1a}(\omega) - \boldsymbol{a}_{1c}^T \boldsymbol{c}_{1c}(\omega) \tag{40d}$$

$$y_2(\omega) = \boldsymbol{a}_{2a}^T \boldsymbol{c}_{2a}(\omega) - \boldsymbol{a}_{2c}^T \boldsymbol{c}_{2c}(\omega) \tag{40e}$$

with $\boldsymbol{a}$, $\boldsymbol{v}(\omega)$, and $d(\omega)$ defined in (28), $\boldsymbol{a}_{ia}$, $\boldsymbol{a}_{ic}$, $\boldsymbol{c}_{ia}(\omega)$, and $\boldsymbol{c}_{ic}(\omega)$ for $i = 1, 2$ defined in a way similar to $\boldsymbol{a}_a$, $\boldsymbol{a}_c$, $\boldsymbol{c}_a(\omega)$, and $\boldsymbol{c}_c(\omega)$ in (28), respectively. The design parameters are put together in vector

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{d} \\ \boldsymbol{a}_{2a} \\ \boldsymbol{a}_{2c} \\ \boldsymbol{a}_{1a} \\ \boldsymbol{a}_{1c} \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \\ \end{matrix}\right\} \text{stage 2} \\ \left.\begin{matrix} \\ \end{matrix}\right\} \text{stage 1} \end{matrix} \tag{41}$$

which is a special case of (21) with $K = 2$.

From (40) and (41), if follows that the gradient of $H(\omega, \boldsymbol{x})$ is given by

$$\boldsymbol{g}(\omega, \boldsymbol{x}) = \begin{bmatrix} y_1(\omega) y_2(M\omega) \frac{\partial \tilde{H}_a^{(2)}(M^2\omega)}{\partial \boldsymbol{a}} \\ y_1(\omega) y_2(M\omega) \frac{\partial \tilde{H}_a^{(2)}(M^2\omega)}{\partial \boldsymbol{d}} \\ y_1(\omega) \tilde{H}_a^{(2)}(M^2\omega) \boldsymbol{c}_{2a}(M\omega) \\ y_1(\omega) \left[1 - \tilde{H}_a^{(2)}(M^2\omega)\right] \boldsymbol{c}_{2c}(M\omega) \\ \tilde{H}_a^{(1)}(M\omega) \boldsymbol{c}_{1a}(\omega) \\ \left[1 - \tilde{H}_a^{(1)}(M\omega)\right] \boldsymbol{c}_{1c}(\omega) \end{bmatrix} \tag{42}$$

where

$$y_i(\omega) = \boldsymbol{a}_{ia}^T \boldsymbol{c}_{ia}(\omega) - \boldsymbol{a}_{ic}^T \boldsymbol{c}_{ic}(\omega) \quad \text{for } i = 1, 2$$

$$\frac{\partial \tilde{H}_a^{(2)}(\omega)}{\partial \boldsymbol{a}} = \frac{\boldsymbol{v}(\omega)}{d(\omega)}$$

$$\frac{\partial \tilde{H}_a^{(2)}(\omega)}{\partial \boldsymbol{d}} = \begin{bmatrix} \frac{\partial \tilde{H}_a^{(2)}(\omega)}{\partial \boldsymbol{d}_0} \\ \frac{\partial \tilde{H}_a^{(2)}(\omega)}{\partial \boldsymbol{d}_1} \\ \vdots \\ \frac{\partial \tilde{H}_a^{(2)}(\omega)}{\partial \boldsymbol{d}_L} \end{bmatrix}$$

with

$$\frac{\partial \tilde{H}_a^{(2)}(\omega)}{\partial d_0} = -\tilde{H}_a^{(2)}(\omega) \frac{v_1(\omega)}{1 + d_0 v_1(\omega)}$$

$$\frac{\partial \tilde{H}_a^{(2)}(\omega)}{\partial \boldsymbol{d}_i} = -\tilde{H}_a^{(2)}(\omega) \frac{\boldsymbol{v}_2(\omega)}{1 + \boldsymbol{d}_i^T \boldsymbol{v}_2(\omega)}, \qquad \text{for } 1 \le i \le L.$$

### B. Initial Design

Given filter length $(n, r)$, $n_{ai}$, and $n_{ci}$ for $i = 1, 2$, sampling rate $M$, and band edges $\omega_p$ and $\omega_a$, an initial design for a two-stage IIR FRM filter can be readily obtained as follows.

i) Use parameters $\omega_p$, $\omega_a$, and $M$ to identify the passband and stopband edges $\theta$ and $\phi$ (see Section V-C) for prototype filter $H_a^{(1)}(z)$. Since $H_a^{(1)}(z)$ will be implemented using an FRM filter in the second stage, we do *not* need to prepare an initial $H_a^{(1)}(z)$ as a single filter.

ii) Use the parameters $\theta$ and $\phi$ obtained from Step (i) together with parameters $M$ and $m$ (see (33) and (34)) to prepare initial masking filters $H_{ma}^{(1)}(z)$ and $H_{mc}^{(1)}(z)$ as describe in Section V-C2.

iii) Use $\hat{\omega}_p = \theta$ and $\hat{\omega}_a = \phi$ as the passband and stopband edges, respectively, for the FRM filter in the second

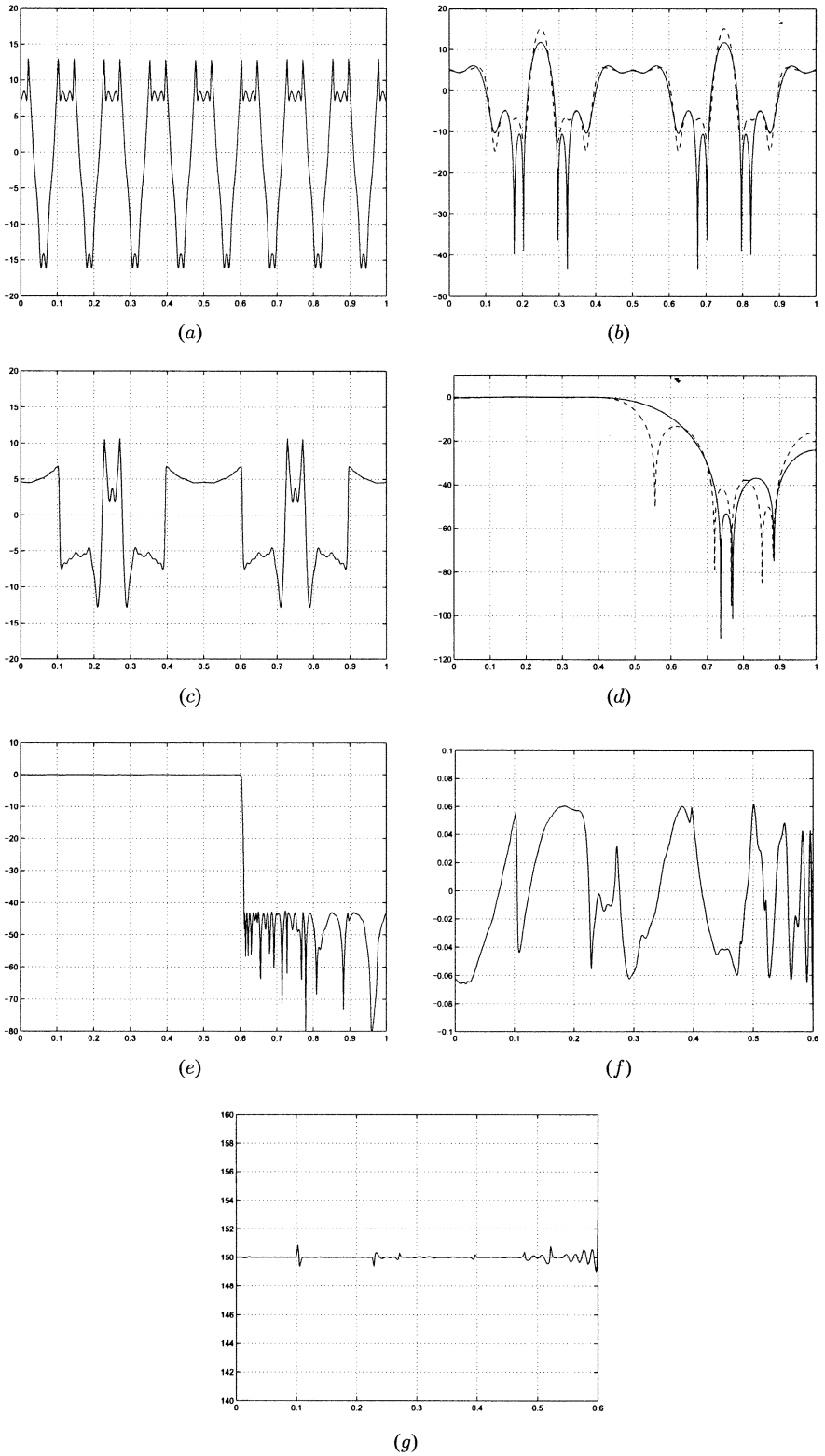



Fig. 7. Amplitude responses of (a) prototype filter $H_a^{(2)}(z^{16})$. (b) Masking filters $H_{ma}^{(2)}(z^4)$ (solid line) and $H_{mc}^{(2)}(z^4)$ (dashed line). (c) Prototype filter $H_a^{(1)}(z^4)$. (d) Masking filters $H_{ma}^{(1)}(z)$ (solid line) and $H_{mc}^{(1)}(z)$ (dashed line). (e) FRM filter. (f) Passband ripples of the FRM filter, all in decibels. (g) Passband group delay of the FRM filter.

stage. This, in conjunction with parameters $(n, r)$, $n_{a2}$, $n_{c2}$, and $M$, can be used to prepare an initial prototype filter $H_a^{(2)}(z)$ and masking filters $\{H_{ma}^{(2)}(z), H_{mc}^{(2)}(z)\}$ using the method described in Section V-C.

Other issues that need to be addressed for the design of multistage IIR FRM filters such as a desired frequency response, a weighting function, and placement of grid points, have been discussed in Sections V-B and Sections V-D.

### C. Design Example

The design presented here is a two-stage lowpass IIR FRM filter with $n = 10$, $r = 6$, $n_{a2} = 13$, $n_{c2} = 23$, $n_{a1} = 15$, $n_{c1} = 21$, $M_1 = M_2 = M = 4$, $D = 6$, $\omega_p = 0.6\pi$, and $\omega_a = 0.61\pi$. Other design parameters used are $w = 1$, $\tau = 0.15$, $\beta_0 = 3 \cdot \sqrt{n+r+1} \cdot 10^{-4}$, $\beta_i = 3 \cdot \sqrt{n_{ai}+n_{ci}} \cdot 10^{-4}$, for $i = 1, 2$, $\beta_t = 1$, $N_t = 4$, and $N = 1,000$.

Using MATLAB toolbox SeDuMi[17], it took the SOCP algorithm 57 iterations to converge. The amplitude responses of the subfilters and FRM filter are shown in Fig. 7(a)–(e), and the passband ripple and passband group delay of the FRM filter are shown in Fig. 7(f) and (g), respectively. The maximum passband ripple was 0.0659 dB, the minimum stopband attenuation was 42.6271 dB, and the passband group delay was $d_1 + Md_2 + M^2D = 150$ samples with 2.11% maximum relative deviation. The maximum magnitude of the poles of $H_a^{(2)}(z)$ was 0.8926.

Compared with the basic IIR FRM filter presented in Section V-E, the current design offers improved performance in terms of passband amplitude ripple and stopband attenuation with reduced implementation complexity in terms of the number of multipliers (55 for the current filter versus 63 for the basic IIR filter) and adders (89 for the current filter versus 99 for the basic IIR filter) used. The cost of the above gains is a 32-sample increase in passband group delay. However, the two-stage IIR FRM filter's 150-sample passband group delay is still considerably less than that of the FIR FRM filter (218 samples).

## VII. Conclusion

We have presented a methodology for the optimal design of basic and multistage FRM filters where the prototype filters are of IIR with prescribed pole radius. It is shown that the design can be accomplished by a sequence of linear updates for the design variables with each update carried out using SOCP. The proposed method begins with a trivial initial point and unifies the algorithms for basic and multistage IIR FRM filters. The design examples presented in the paper have demonstrated that the class of FRM filters with IIR prototype filters of robust stability offers an attractive alternative to its FIR counterpart in terms of filter performance, system delay, and realization complexity.

## References


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

[2] R. Yang, B. Liu, and Y. C. Lim, "A new structure of sharp transition FIR filters using frequency-response masking," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 955–966, Aug. 1988.

[3] G. Rajan, Y. Neuvo, and S. K. Mitra, "On the design of sharp cutoff wide-band FIR filters with reduced arithmetic complexity," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1447–1454, Nov. 1988.

[4] T. Saramaki and A. T. Fam, "Subfilter approach for designing efficient FIR filters," in *Proc. ISCAS'88*, 1988, pp. 2903–2915.

[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] ——, "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. Saramaki, Y. C. Lim, and R. Yang, "The synthesis of half-band filter using frequency-response marking technique," *IEEE Trans. Circuits Syst. II*, vol. 42, pp. 58–60, Jan. 1995.

[8] M. G. Bellanger, "Improved design of long FIR filters using the frequency masking technique," in *Proc. ICASSP'96*, 1996, pp. 1272–1275.

[9] T. Saramaki and H. Johansson, "Optimization of FIR filters using frequency-response masking approach," in *Proc. ISCAS'01*, vol. 2, 2001, pp. 609–612.

[10] H. Johansson and L. Wanhammar, "High-speed recursive digital filters based on the frequency-response masking approach," *IEEE Trans. Circuits Syst. II*, vol. 47, pp. 48–61, Jan. 2000.

[11] Y. E. Nesterov and A. Nemirovski, *Interior-Point Polynomial Methods in Convex Programming*. Philadelphia, PA: SIAM, 1994.

[12] Y. E. Nesterov and M. J. Todd, "Self-scaled barriers and interior-point methods for convex programming," *Math. Oper. Res.*, vol. 22, pp. 1–42, 1997.

[13] J. F. Sturm, *Primal-Dual Interior-Point Approach to Semidefinite Programming*. Amsterdam, The Netherlands: Tinbergen Inst. Res. Series, 1997, vol. 156.

[14] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret, "Applications of second-order cone programming," *Linear Algebr. Applicat.*, vol. 248, pp. 193–228, Nov. 1998.

[15] Z.-Q. Luo, J. F. Sturm, and S. Zhang, "Conic convex programming and self-dual embedding," *Optim. Meth. Softw.*, vol. 14, no. 3, pp. 169–218, 2000.

[16] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization*. Philadelphia, PA: SIAM, 2001.

[17] J. F. Sturm, "Using SeDuMi1.02, a MATLAB toolbox for optimization over symmetric cones," *Optim. Meth. Softw.*, vol. 11–12, pp. 625–653, 1999.

[18] R. H. Tütüncü, K. C. Toh, and M. J. Todd, SDPT3—A MATLAB Software Package for Semidefinite-Quadratic-Linear Programming, Version 3.0, Aug. 2001.

[19] P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali, *Manual of LMI Control Toolbox*. Natick, MA: The MathWorks, Inc,, 1995.

[20] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Rev.*, vol. 38, pp. 49–95, Mar. 1996.

[21] A. Antoniou, *Digital Filters: Analysis and Design*, 2nd ed. New York: McGraw-Hill, 1993.

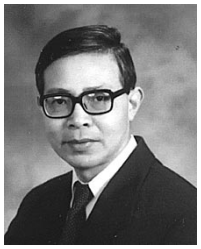
[22] M. C. Lang, "Least-squares design of IIR filters with prescribed magnitude and phase response and a pole radius constraint," *IEEE Trans. Signal Processing*, vol. 48, pp. 3109–3121, Nov. 2000.

[23] J. E. Dennis Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Philadelphia, PA: SIAM, 1996.

[24] D. G. Luenberger, *Linear and Nonlinear Programming*, 2nd ed. Reading, MA: Addison-Wesley, 1984.

[25] R. Fletcher, *Practcal Methods of Optimization*, 2nd ed. New York: Wiley, 1987.

[26] B. W. Bomar, "Computationally efficient low roundoff noise second-order state-space structures," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 35–41, Jan. 1986.


**Wu-Sheng Lu** (S'81–M'SS–SM'90–F'99) received the undergraduate degree in mathematics from Fudan University, Shanghai, China, in 1964, and the M.S. degree in electrical engineering and Ph.D. degree in control science from the University of Minnesota, Minneapolis, in 1983 and 1984, respectively.

He was a Post-Doctoral Fellow at the University of Victoria, Victoria, BC, Canada in 1985, and a Visiting Assistant Professor at the University of Minnesota in 1986. Since 1987, he has been with the University of Victoria, where he is currently a Professor. His teaching and research interests are in the areas of digital signal processing and application of optimization methods. He is the coauthor (with A. Antoniou) of *Two-Dimensional Digital Filters* (New York: Marcel Dekker, 1992). He was an Associate Editor of the *Canadian Journal of Electrical and Computer Engineering* in 1989, and its Editor from 1990 to 1992. He is presently an Associate Editor for the *International Journal of Multidimensional Systems and Signal Processing*.

Dr. Lu served as an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: ANALOG AND SIGNAL DEVICES from 1993 to 1995, and for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: FUNDAMENTAL THEORY AND APPLICATIONS, from 1999 to 2001. He is a Fellow of the Engineering Institute of Canada.

**Takao Hinamoto** (M'77–SM'84–F'01) 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 in the Department of Electrical Engineering, Queen's University, Kingston, ON, Canada, on leave from Kobe University. During 1988–1991, he was a Professor of electronic circuits in the Faculty of Engineering, Tottori University, Tottori, Japan. Since January 1992, he has been a Professor of Electronic Control in the Department of Electrical Engineering, Hiroshima University, Hiroshima, Japan. His research interests include digital signal processing, system theory, and control engineering. He has published more than 290 papers in these areas and is the coeditor and coauthor of *Two-Dimensional Signal and Image Processing* (Tokyo, Japan: SICE, 1996).

Dr. Hinamoto served as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: ANALOG AND SIGNAL DEVICES from 1993 to 1995, and presently serves as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: FUNDAMENTAL THEORY AND APPLICATIONS. He was the Guest Editor of the special section of DSP in the August 1998 issue of the *IEICE Transactions on Fundamentals*. He also served as Chair of the 12th Digital Signal Processing (DSP) Symposium held in Hiroshima in November 1997, sponsored by the DSP Technical Committee of IEICE. Since 1995, he has been a member of the steering committee of the IEEE Midwest Symposium on Circuits and Systems, and since 1998, a member of the Digital Signal Processing Technical Committee in the IEEE Circuits and Systems Society. He served as a member of the Technical Program Committee for ISCAS'99. From 1993 to 2000, he served as a senator or member of the Board of Directors in the Society of Instrument and Control Engineers (SICE), and from 1999 to 2001, he was Chair of the Chugoku Chapter of SICE. He played a leading role in establishing the Hiroshima Section of IEEE, and served as the Interim Chair of the section. Presently, he serves as Chair of the DSP Technical Committee of IEICE and Chair of the Chugoku Chapter of IEICE. He is a recipient of the IEEE Third Millennium Medal.