

PAPER

Minimax Design of Sparse IIR Filters Using Sparse Linear Programming

Masayoshi NAKAMOTO^{†a)}, *Member* and Naoyuki AIKAWA^{††b)}, *Senior Member*

SUMMARY Recent trends in designing filters involve development of sparse filters with coefficients that not only have real but also zero values. These sparse filters can achieve a high performance through optimizing the selection of the zero coefficients and computing the real (non-zero) coefficients. Designing an infinite impulse response (IIR) sparse filter is more challenging than designing a finite impulse response (FIR) sparse filter. Therefore, studies on the design of IIR sparse filters have been rare. In this study, we consider IIR filters whose coefficients involve zero value, called sparse IIR filter. First, we formulate the design problem as a linear programming problem without imposing any stability condition. Subsequently, we reformulate the design problem by altering the error function and prepare several possible denominator polynomials with stable poles. Finally, by incorporating these methods into successive thinning algorithms, we develop a new design algorithm for the filters. To demonstrate the effectiveness of the proposed method, its performance is compared with that of other existing methods.

key words: infinite impulse response (IIR) filters, sparse filter, zero coefficients, sparse linear programming, minimax design

1. Introduction

Digital filters are essential tools in the field of signal processing [1]–[4]. The Finite impulse response (FIR) [5]–[13] and infinite impulse response (IIR) filters [14]–[20] are some of the most common filters. Although the design of IIR filters is generally more complicated than that of FIR filters, IIR filters can achieve a higher performance than that of FIR filters. In a digital circuit, these filters comprise multipliers (non-zero filter coefficients), adders, and delay elements. Further, the performance of a filter strongly depends on the number of multipliers (or non-zero filter coefficients).

We assume $\mathbf{x}$ is the vector of the filter coefficients, and K presents the number of elements in $\mathbf{x}$, i.e., the number of multipliers of the filter is K . Further, to obtain a filter with a higher performance, a large K (more multipliers) is required. These filters are designed such that the filter coefficients are all real, i.e., $\mathbf{x} \in \mathbb{R}^K$, are usually approached. Meanwhile, the recent trend is to design sparse filters whose coefficients not only have real but also zero values. That is,

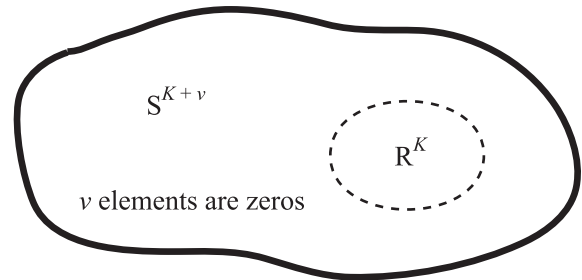


Fig. 1 Inclusive relationship between $\mathbb{S}^{K+v}$ and $\mathbb{R}^K$.

the strategy of the sparse filter design is to determine the element of $\mathbf{x}$ with zero coefficients. When zeros are included among the elements of $\mathbf{x}$ for a sparse filter, the number of non-zero elements in $\mathbf{x}$ can be increased to be identical to that of the corresponding vector for a non-sparse equivalent filter. In addition, we assume v is a non-negative integer, and $\mathbb{S}^{K+v}$ is a set of $K + v$ real values, where v elements are zeros. That is, for sparse filter design, we consider $\mathbf{x} \in \mathbb{S}^{K+v}$. When the number of multipliers is K , the search space $\mathbb{S}^{K+v}$ is always larger than $\mathbb{R}^K$ and includes $\mathbb{R}^K$, where $\mathbb{S}^{K+v} = \mathbb{R}^K$ if and only if $v = 0$. Figure 1 illustrates an inclusive relationship between $\mathbb{S}^{K+v}$ and $\mathbb{R}^K$. Consequently, it is expected that the solution from the space $\mathbb{S}^{K+v}$ is better than that of the space $\mathbb{R}^K$.

With respect to the design of sparse filters, researchers have focused on the design of FIR filters [21]–[32], the main reason being the highly complicated design of IIR filters compared with that of FIR filters, which is because of the non-linearity of the design problem caused by the presence of a denominator polynomial. Hence, there have been limited studies on the design of sparse IIR filters [33], [34]. For instance, a recent study [34] employed second-order cone programming (SOCP) [1] to design an IIR filter. Because of the stability constraint, sparse optimization of denominator coefficients is difficult. Therefore, sparse optimization of the numerator coefficients was introduced in [34]. Additionally, the design examples are limited to the case of second-order denominator polynomial.

In this study, we consider the minimax design of sparse IIR filters based on linear programming problems. This consists of multiple steps and includes sparse optimization (sparse linear programming). Since the denominator coefficients affect the stability constraint, the proposed method considers the sparse optimization of denominator coefficients, similar to [34]. Further, a method that can design

Manuscript received August 13, 2020.

Manuscript revised December 9, 2020.

Manuscript publicized February 15, 2021.

[†]The author is with the Graduate School of Advanced Science and Engineering, Hiroshima University, Higashi-Hiroshima-shi, 739-8527 Japan.

^{††}The author is with the Department of Applied Electronics, Faculty of Industrial Science and Technology, Tokyo University of Science, Tokyo, 125-8585 Japan.

a) E-mail: msy@hiroshima-u.ac.jp

b) E-mail: ain@te.noda.tus.ac.jp

DOI: 10.1587/transfun.2020EAP1096

denominator coefficients of any order is required. Our proposed method can design denominator coefficients of an arbitrary order. First, we formulated the design problem as a function of linear programming without imposing the stability condition. By solving the problem, we obtain an inconclusive denominator coefficient. Next, if there are unstable poles of the inconclusive denominator, we rearrange the unstable poles. According to the procedure, we compute several possible denominator polynomials. Thereafter, we modify the error function and re-formulate the design problem for each denominator polynomial. Finally, we search the numerator coefficients that contain zero value elements based on the sparse linear programming by changing the denominator coefficients.

The remainder of this paper is organized as follows. Section 2 presents a formulation for the minimax design of IIR filters using the linearized error function. In Sect. 3, we propose a new design algorithm for IIR filters based on linear programming, where the error function is switched before computing the numerator coefficients. Subsequently, we present sparse linear programming for the numerator coefficients while varying the denominator coefficients. Moreover, to demonstrate the effectiveness of the proposed method, we compare its performance with that of the corresponding non-sparse and sparse IIR filters, which is designed using another method in Sect. 4. Section 5 concludes this paper.

2. Problem Formulation

Consider the desired response expressed as

$$H_d(\omega) = G(\omega)e^{-j\tau(\omega)}, \quad \forall \omega \in \Omega \quad (1)$$

where $G(\omega)$ is a gain response, $\tau(\omega)$ is a phase response, and $j^2 = -1$. Further, Ω is a frequency region of interest and a closed subset of $[0, \pi]$.

In this work, we try to solve the problem to approximate $H_d(\omega)$ with the rational transfer function $H(\omega)$, which is given as

$$H(\omega; \mathbf{a}, \mathbf{b}) = \frac{B(\omega; \mathbf{b})}{A(\omega; \mathbf{a})} \quad (2)$$

where $A(\omega; \mathbf{a})$ for $\mathbf{a} \in \mathbb{R}^{m+1}$ is the denominator polynomial of order m , and $B(\omega; \mathbf{b})$ for $\mathbf{b} \in \mathbb{R}^{n+1}$ is the numerical polynomial of order n . The vectors $\mathbf{a}$ and $\mathbf{b}$ are the denominator and numerator coefficient vectors, respectively, defined as:

$$\mathbf{a} = [a_0, a_1, \dots, a_m], \quad a_0 = 1 \quad (3)$$

$$\mathbf{b} = [b_0, b_1, \dots, b_n]. \quad (4)$$

In (2), $A(\omega; \mathbf{a})$ and $B(\omega; \mathbf{b})$ are respectively written as

$$A(\omega; \mathbf{a}) = \mathbf{a} \cdot [1, e^{-j\omega}, \dots, e^{-jm\omega}]^T \quad (5)$$

$$B(\omega; \mathbf{b}) = \mathbf{b} \cdot [1, e^{-j\omega}, \dots, e^{-jn\omega}]^T \quad (6)$$

where the superscript T indicates the transposition of the

matrix (vector).

Now, define the weighted error function as

$$E(\omega) = W(\omega) |H_d(\omega) - H(\omega; \mathbf{a}, \mathbf{b})|, \quad \forall \omega \in \Omega \quad (7)$$

where $W(\omega)$ is the weighting function and $W(\omega) \geq 0$. We set $W(\omega) = 0$ in the transition band. Assuming that δ is a positive real number such that $E(\omega) \leq \delta$, we wish to determine $\mathbf{a}$ and $\mathbf{b}$ that minimize δ . However, since the minimization of δ is a difficult task, we focus on the linearized error function defined as

$$\varepsilon(\omega) = W(\omega) |H_d(\omega)A(\omega; \mathbf{a}) - B(\omega; \mathbf{b})|, \quad \forall \omega \in \Omega. \quad (8)$$

Instead of $E(\omega) \leq \delta$, we solve the problem to determine $\mathbf{a}$ and $\mathbf{b}$ that minimize δ_0 with

$$\varepsilon(\omega) \leq \delta_0. \quad (9)$$

Substituting (1)–(6) into (8), we have

$$\begin{aligned} \varepsilon(\omega) &= W(\omega) \left| G(\omega) \sum_{k=0}^m a_k \cos[k\omega + \tau(\omega)] \right. \\ &\quad \left. - \sum_{l=0}^n b_l \cos(l\omega) + j \left\{ -G(\omega) \sum_{k=0}^m a_k \right. \right. \\ &\quad \left. \cdot \sin[k\omega + \tau(\omega)] + \sum_{l=0}^n b_l \sin(l\omega) \right\} \Big| \\ &\leq W(\omega) \left| G(\omega) \sum_{k=0}^m a_k \cos[k\omega + \tau(\omega)] \right. \\ &\quad \left. - \sum_{l=0}^n b_l \cos(l\omega) \right| + W(\omega) \left| G(\omega) \sum_{k=0}^m a_k \right. \\ &\quad \left. \cdot \sin[k\omega + \tau(\omega)] - \sum_{l=0}^n b_l \sin(l\omega) \right|. \end{aligned} \quad (10)$$

Now, we impose the constraints

$$\begin{aligned} &W(\omega) \left| G(\omega) \sum_{k=0}^m a_k \cos[k\omega + \tau(\omega)] - \sum_{l=0}^n b_l \cos(l\omega) \right| \\ &\leq \frac{\delta_0}{2} \end{aligned} \quad (11)$$

$$\begin{aligned} &W(\omega) \left| G(\omega) \sum_{k=0}^m a_k \sin[k\omega + \tau(\omega)] - \sum_{l=0}^n b_l \sin(l\omega) \right| \\ &\leq \frac{\delta_0}{2} \end{aligned} \quad (12)$$

to achieve (9). The aim of this work is to find $\mathbf{a}$ and $\mathbf{b}$ that minimizes δ_0 . Hence, δ_0 is the evaluation value of the solution (or filter).

Chottera and Jullien [14] established that the linear programming that minimizes δ_0 with the constraints (11) and (12) as well as the positive realness constraint is embedded in a linear program to achieve stability.

3. Design Scheme

3.1 Computation of Possible Denominator Coefficients

We aim at solving the problem of minimizing δ_0 under a stability constraint and with zero values of the numerator coefficients. We utilize the approach established in [20]. The design procedure of [20] consisted of three steps. First, a temporary denominator was designed with a numerator of order n without imposing the stability condition. Second, the poles were moved such that the maximum pole radius was within or on a circle with a specified radius. Third, the numerator coefficients of the filter were modified to cancel the degradation due to the pole movement.

However, the numerator coefficients in the final step are modified again because of the sparse optimization of the numerator coefficients. Hence, n in (11) and (12) may not necessarily be equal to the order of the numerator coefficients at the final step. Hence, we modify (11) and (12) to

$$W(\omega) \left| G(\omega) \sum_{k=0}^m a_k \cos[k\omega + \tau(\omega)] - \sum_{l=0}^{\mu} b_l \cos(l\omega) \right| \leq \frac{\delta_0}{2} \quad (13)$$

$$W(\omega) \left| G(\omega) \sum_{k=0}^m a_k \sin[k\omega + \tau(\omega)] - \sum_{l=0}^{\mu} b_l \sin(l\omega) \right| \leq \frac{\delta_0}{2} \quad (14)$$

where μ is a positive integer. Furthermore, μ is a *temporary* order, not the definitive order, of the numerator polynomial. The following procedure is implemented by the sequential assignment of μ .

The number of grid points over $[0, \pi]$ is denoted by L . Subsequently, ω is taken from a finite and dense frequency grid $\Omega_d = \{\omega_1, \omega_2, \dots, \omega_L\}$. According to (13) and (14) the problem can be formulated as

$$\min_{\mathbf{a}, \mathbf{b}} \quad \delta_0 \quad (15a)$$

$$\text{subject to} \quad \mathbf{a}\mathbf{P} + \mathbf{b}\mathbf{Q}^{(\mu)} - \frac{\delta_0}{2}\mathbf{u} \leq \mathbf{0} \quad (15b)$$

$$\mathbf{a} \cdot \mathbf{p}_0 = 1 \quad (15c)$$

where $\mathbf{P}$ is an $(m+1) \times 4L$ matrix, $\mathbf{Q}^{(\mu)}$ is a $(\mu+1) \times 4L$ matrix, $\mathbf{u}$ is a $4L$ -dimensional row vector $\mathbf{u} = [1, 1, \dots, 1]$, and $\mathbf{p}_0$ is an $(m+1)$ -dimensional column vector $\mathbf{p}_0 = [1, 0, 0, \dots, 0]^T$. $\mathbf{P}$ is expressed as

$$\mathbf{P} = \begin{bmatrix} \mathbf{p}_1^{(0)} & -\mathbf{p}_1^{(0)} & \mathbf{p}_2^{(0)} & -\mathbf{p}_2^{(0)} \\ \mathbf{p}_1^{(1)} & -\mathbf{p}_1^{(1)} & \mathbf{p}_2^{(1)} & -\mathbf{p}_2^{(1)} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{p}_1^{(m)} & -\mathbf{p}_1^{(m)} & \mathbf{p}_2^{(m)} & -\mathbf{p}_2^{(m)} \end{bmatrix} \quad (16)$$

where

$$\mathbf{p}_1^{(k)} = \begin{bmatrix} W(\omega_1)G(\omega_1) \cos[k\omega_1 + \tau(\omega_1)] \\ W(\omega_2)G(\omega_2) \cos[k\omega_2 + \tau(\omega_2)] \\ \vdots \\ W(\omega_L)G(\omega_L) \cos[k\omega_L + \tau(\omega_L)] \end{bmatrix}^T$$

$$\mathbf{p}_2^{(k)} = \begin{bmatrix} W(\omega_1)G(\omega_1) \sin[k\omega_1 + \tau(\omega_1)] \\ W(\omega_2)G(\omega_2) \sin[k\omega_2 + \tau(\omega_2)] \\ \vdots \\ W(\omega_L)G(\omega_L) \sin[k\omega_L + \tau(\omega_L)] \end{bmatrix}^T$$

for $k = 0, 1, \dots, m$. Further, $\mathbf{Q}^{(\mu)}$ is represented by

$$\mathbf{Q}^{(\mu)} = \begin{bmatrix} \mathbf{q}_1^{(0)} & -\mathbf{q}_1^{(0)} & \mathbf{q}_2^{(0)} & -\mathbf{q}_2^{(0)} \\ \mathbf{q}_1^{(1)} & -\mathbf{q}_1^{(1)} & \mathbf{q}_2^{(1)} & -\mathbf{q}_2^{(1)} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{q}_1^{(\mu)} & -\mathbf{q}_1^{(\mu)} & \mathbf{q}_2^{(\mu)} & -\mathbf{q}_2^{(\mu)} \end{bmatrix} \quad (17)$$

where

$$\mathbf{q}_1^{(l)} = - \begin{bmatrix} W(\omega_1) \cos(l\omega_1) \\ W(\omega_2) \cos(l\omega_2) \\ \vdots \\ W(\omega_L) \cos(l\omega_L) \end{bmatrix}^T$$

$$\mathbf{q}_2^{(l)} = - \begin{bmatrix} W(\omega_1) \sin(l\omega_1) \\ W(\omega_2) \sin(l\omega_2) \\ \vdots \\ W(\omega_L) \sin(l\omega_L) \end{bmatrix}^T$$

for $l = 0, 1, \dots, \mu$.

By solving the problem (15), we obtain $\mathbf{a}$ and $\mathbf{b}$. In addition, the elements of $\mathbf{a}$ represent the *temporary* coefficients of the denominator polynomial. Notably, in the next step, $\mathbf{a}$ is changed, and $\mathbf{b}$ is ignored.

Therefore, we design the denominator polynomial with a specified maximum pole radius r_c based on the elements of $\mathbf{a}$. The poles $\rho_k^{(\mu)} = |\rho_k^{(\mu)}|e^{j\theta_k^{(\mu)}}$ are obtained by solving the problem (15), and $\tilde{\rho}_k^{(\mu)}$ is the new pole for $k = 1, 2, \dots, m$. Based on [20], the influence due to the movement of poles is minimized by setting new poles as

$$\tilde{\rho}_k^{(\mu)} = \begin{cases} r_c \frac{\rho_k^{(\mu)}}{|\rho_k^{(\mu)}|}, & \text{if } |\rho_k^{(\mu)}| > r_c \\ \rho_k^{(\mu)}, & \text{otherwise} \end{cases} \quad (18)$$

for $k = 1, 2, \dots, m$. With the new poles obtained by (18), we have

$$\prod_{k=1}^m (1 - \tilde{\rho}_k^{(\mu)} e^{-jk\omega}) = \sum_{k=0}^m \tilde{a}_k^{(\mu)} e^{-jk\omega}, \quad \tilde{a}_0^{(\mu)} = 1 \quad (19)$$

where $\tilde{a}_0^{(\mu)}, \tilde{a}_1^{(\mu)}, \dots, \tilde{a}_m^{(\mu)}$ are new denominator coefficients. Now, define

$$\tilde{\mathbf{a}}^{(\mu)} = [\tilde{a}_0^{(\mu)}, \tilde{a}_1^{(\mu)}, \tilde{a}_2^{(\mu)}, \dots, \tilde{a}_m^{(\mu)}]. \quad (20)$$

The elements of (20) correspond to the possible values of the denominator coefficients. The maximum pole radius of $\tilde{\mathbf{a}}^{(\mu)}$ is r_c . We observe that specifying the maximum pole radius as in [16] is beneficial to the design of an IIR filter because robust stability can be achieved.

3.2 Modification of Error Function

The numerator coefficients are optimized by fixing the denominator coefficients. Substituting (20) into (5), a denominator polynomial is obtained $\tilde{A}(\omega; \tilde{\mathbf{a}}^{(\mu)})$. In order to simplify the notation, we write

$$\tilde{A}(\omega) = \sum_{k=0}^m \tilde{a}_k e^{-jk\omega}, \quad \tilde{a}_0 = 1 \quad (21)$$

as a substitute of $A(\omega; \tilde{\mathbf{a}}^{(\mu)})$. Furthermore, there are several different values of $\tilde{A}(\omega)$ that correspond to the index μ .

Subsequently, we redefine the error function after computing (20) as

$$\tilde{E}(\omega) = \frac{W(\omega)}{|\tilde{A}(\omega)|} |H_d(\omega) \tilde{A}(\omega) - B(\omega; \mathbf{b})|, \quad \forall \omega \in \Omega \quad (22)$$

with

$$|\tilde{A}(\omega)| = \left[\left\{ \sum_{k=0}^m \tilde{a}_k \cos(k\omega) \right\}^2 + \left\{ \sum_{k=0}^m \tilde{a}_k \sin(k\omega) \right\}^2 \right]^{\frac{1}{2}}. \quad (23)$$

$\tilde{E}(\omega)$ is a re-weighted version of $\varepsilon(\omega)$. Furthermore, the numerator coefficients are determined to minimize $\tilde{\delta}$ such that $\tilde{E}(\omega) \leq \tilde{\delta}$. Consequently, the error function is changed as $\varepsilon(\omega) \rightarrow \tilde{E}(\omega)$. In particular, $\varepsilon(\omega)$ is the linearized error function. Moreover, $\tilde{E}(\omega)$ is a *true* error function that is not linearized.

3.3 Reformulation of the Design Problem

Since the error function was changed, we re-formulate the design problem to determine $\mathbf{b}$ for minimizing $\tilde{\delta}$.

Recalling (21) and (22), we have

$$\begin{aligned} \tilde{E}(\omega) = & \frac{W(\omega)}{|\tilde{A}(\omega)|} \left| G(\omega) \sum_{k=0}^m \tilde{a}_k \cos[k\omega + \tau(\omega)] \right. \\ & - \sum_{l=0}^n b_l \cos(l\omega) + j \left\{ -G(\omega) \sum_{k=0}^m \tilde{a}_k \right. \\ & \left. \left. \cdot \sin[k\omega + \tau(\omega)] + \sum_{l=0}^n b_l \sin(l\omega) \right\} \right| \end{aligned}$$

$$\begin{aligned} \leq & \frac{W(\omega)}{|\tilde{A}(\omega)|} \left| G(\omega) \sum_{k=0}^m \tilde{a}_k \cos[k\omega + \tau(\omega)] \right. \\ & - \sum_{l=0}^n b_l \cos(l\omega) \left| + \frac{W(\omega)}{|\tilde{A}(\omega)|} \left| G(\omega) \sum_{k=0}^m \tilde{a}_k \right. \right. \\ & \left. \left. \cdot \sin[k\omega + \tau(\omega)] - \sum_{l=0}^n b_l \sin(l\omega) \right| \right|. \quad (24) \end{aligned}$$

Similar to the previous stage, we let

$$\begin{aligned} W(\omega) \left| G(\omega) \sum_{k=0}^m \tilde{a}_k \cos[k\omega + \tau(\omega)] - \sum_{l=0}^n b_l \cos(l\omega) \right| \\ \leq \frac{\tilde{\delta}}{2} |\tilde{A}(\omega)| \quad (25) \end{aligned}$$

$$\begin{aligned} W(\omega) \left| G(\omega) \sum_{k=0}^m \tilde{a}_k \sin[k\omega + \tau(\omega)] - \sum_{l=0}^n b_l \sin(l\omega) \right| \\ \leq \frac{\tilde{\delta}}{2} |\tilde{A}(\omega)| \quad (26) \end{aligned}$$

such that $\tilde{E}(\omega) \leq \tilde{\delta}$.

By substituting (25) and (26) into (24), and discretizing ω as in the case of problem (15), the design problem can be expressed as

$$\min_{\mathbf{b}} \quad \tilde{\delta} \quad (27a)$$

$$\text{subject to} \quad \mathbf{b} \mathbf{Q}^{(n)} - \frac{\tilde{\delta}}{2} \tilde{\mathbf{u}} \leq -\tilde{\mathbf{p}} \quad (27b)$$

where $\tilde{\delta}$ is the evaluation value of $\mathbf{b}$. Here, $\tilde{\mathbf{u}}$ is $4L$ -dimensional row vector as follows.

$$\begin{aligned} \tilde{\mathbf{u}} = & [|\tilde{A}(\omega_1)|, |\tilde{A}(\omega_2)|, \dots, |\tilde{A}(\omega_L)|, \\ & |\tilde{A}(\omega_1)|, |\tilde{A}(\omega_2)|, \dots, |\tilde{A}(\omega_L)|, \\ & |\tilde{A}(\omega_1)|, |\tilde{A}(\omega_2)|, \dots, |\tilde{A}(\omega_L)|, \\ & |\tilde{A}(\omega_1)|, |\tilde{A}(\omega_2)|, \dots, |\tilde{A}(\omega_L)|], \quad (28) \end{aligned}$$

and $\tilde{\mathbf{p}}$ is $4L$ -dimensional row vector

$$\tilde{\mathbf{p}} = [\tilde{\mathbf{p}}_1 \quad -\tilde{\mathbf{p}}_1 \quad \tilde{\mathbf{p}}_2 \quad -\tilde{\mathbf{p}}_2] \quad (29)$$

where

$$\tilde{\mathbf{p}}_1 = \begin{bmatrix} \sum_{k=0}^m W(\omega_1) G(\omega_1) \tilde{a}_k \cos[k\omega_1 + \tau(\omega_1)] \\ \sum_{k=0}^m W(\omega_2) G(\omega_2) \tilde{a}_k \cos[k\omega_2 + \tau(\omega_2)] \\ \vdots \\ \sum_{k=0}^m W(\omega_L) G(\omega_L) \tilde{a}_k \cos[k\omega_L + \tau(\omega_L)] \end{bmatrix}^T$$

$$\tilde{\mathbf{p}}_2 = \begin{bmatrix} \sum_{k=0}^m W(\omega_1)G(\omega_1)\tilde{a}_k \sin[k\omega_1 + \tau(\omega_1)] \\ \sum_{k=0}^m W(\omega_2)G(\omega_2)\tilde{a}_k \sin[k\omega_2 + \tau(\omega_2)] \\ \vdots \\ \sum_{k=0}^m W(\omega_L)G(\omega_L)\tilde{a}_k \sin[k\omega_L + \tau(\omega_L)] \end{bmatrix}^T.$$

The denominator coefficients are fixed in the problem (27). By solving the problem (27), the numerator coefficients can be computed.

3.4 Sparse Linear Programming

The number of multipliers of the non-sparse filter is $m+n+1$ because of $\mathbf{a} \in \mathbb{R}^{m+1}$, $\mathbf{b} \in \mathbb{R}^{n+1}$, and $a_0 = 1$. At this stage, we assume that the order of the numerator coefficients is

$$n_+ = n + \nu \quad (30)$$

where ν is a positive integer. In this work, we consider ν a redundant order. If the coefficients of the sparse filter contain ν zero coefficients, the number of multipliers will be $m+n+1$, which is the same as that of the non-sparse filter. Hence, we attempt to obtain the coefficients $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{b}}$, where $\tilde{\mathbf{a}} \in \mathbb{R}^{m+1}$ and $\tilde{\mathbf{b}} \in \mathbb{R}^{n+\nu+1}$ are subject to $\tilde{\mathbf{b}}$ containing ν zero coefficients.

In this section, we utilize sparse linear programming for the design of the numerator coefficients with zero coefficients with possible denominator coefficients (20).

We write a vector of the numerator coefficients with ν redundant order as

$$\tilde{\mathbf{b}} = [\tilde{b}_0, \tilde{b}_1, \dots, \tilde{b}_n, \tilde{b}_{n+1}, \tilde{b}_{n+2}, \dots, \tilde{b}_{n+\nu}]. \quad (31)$$

Now, let i be the index of the thinning operator. The thinning operator changes the real coefficient to zero value step by step. Therefore, the thinning operator with $i = 0$ indicates that all coefficients are non-zero. At the first step ($i = 0$), problem (27) can be rewritten as

$$\min_{\tilde{\mathbf{b}}} \quad \tilde{\delta} \quad (32a)$$

$$\text{subject to} \quad \tilde{\mathbf{b}}\mathbf{Q}^{(n+\nu)} - \frac{\tilde{\delta}}{2}\tilde{\mathbf{u}} \leq -\tilde{\mathbf{p}} \quad (32b)$$

by replacing $\mathbf{b} \rightarrow \tilde{\mathbf{b}}$ and $n \rightarrow n + \nu$.

Hereafter, we assume $i = 1, 2, \dots, \nu$. Let $\tilde{L}$ be the number of grid points over $[0, \pi]$ to evaluate the performance of the filter. In addition, we choose $\tilde{L}$ such that $\tilde{L} \gg L$. Then, ω is taken from a finite and dense frequency grid $\tilde{\Omega}_d = \{\omega_1, \omega_2, \dots, \omega_{\tilde{L}}\}$. We write a complex error as

$$e(\omega) = H_d(\omega) - H(\omega; \tilde{\mathbf{a}}, \tilde{\mathbf{b}}), \quad \forall \omega \in \tilde{\Omega}_d \quad (33)$$

where we define $e(\omega) \equiv 0$ in the transition band. We use

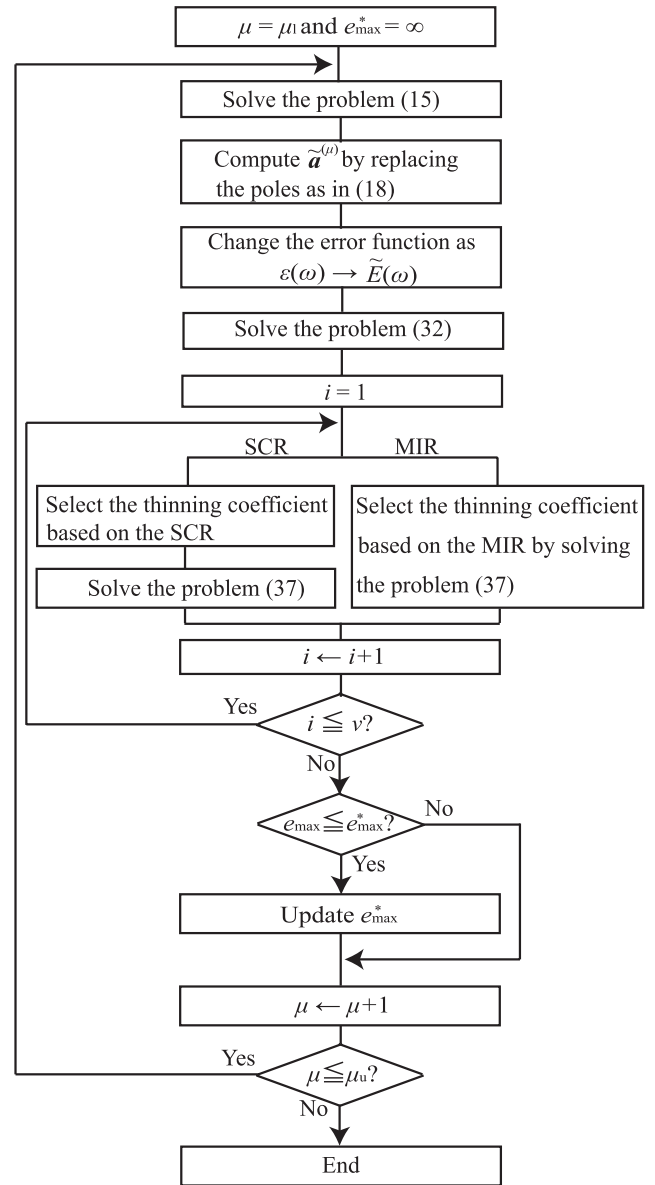


Fig. 2 Flowchart of the overall design algorithm.

$$e_{\max} = \max_{\omega \in \tilde{\Omega}_d} |e(\omega)| \quad (34)$$

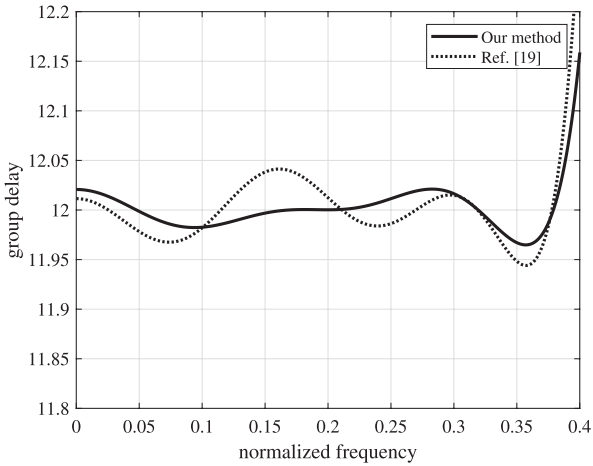
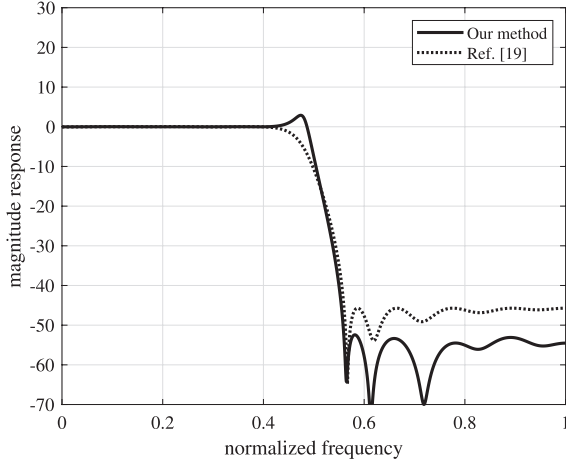
as an evaluation value of the following algorithm. Hence, we attempt to obtain the combination of ν zero coefficients that minimized (34). Afterwards, we employ the successive thinning algorithm presented in [24] for sparse optimization of the numerator coefficients. It contains two rules: the smallest coefficients rule (SCR) and the minimum-increase rule (MIR).

Moreover, we demonstrate how to compute the numerator coefficients with several zero coefficients. Let I_S be a set of index of the zero coefficients, and I_R be a set of index of the non-zero coefficients. Hence, $I_S \cup I_R = \{0, 1, \dots, n + \nu\}$ and $I_S \cap I_R = \{\}$, where the sets are initially $I_R = \{0, 1, \dots, n + \nu\}$ and $I_S = \{\}$.

At this point, let $k^{(i)}$ be the index of the zero coefficients

Table 1 Relation between r_c , $e_{\max}^*$ and μ of $e_{\max}^*$ in Example 1.

Rule of Search	r_c	$e_{\max}^*$	μ of $e_{\max}^*$
SCR	0.92	0.004380	17
	0.94	0.003254	17
	0.96	0.002574	18
MIR	0.92	0.004434	17
	0.94	0.003015	17
	0.96	0.002388	17

**Fig. 3** Magnitude (dB, the upper graph) and group delays in the pass-band (the lower graph) of the responses of the sparse IIR filter designed using our method and the corresponding non-sparse IIR filter designed using the method introduced in [19] in Example 1.

selected at the i th step. Moreover, assume that $\mathbf{s}^{(i)}$ is an $(n + \nu + 1)$ -dimensional column vector. $s_k^{(i)}$ is the $(k + 1)$ th element of $\mathbf{s}^{(i)}$ and

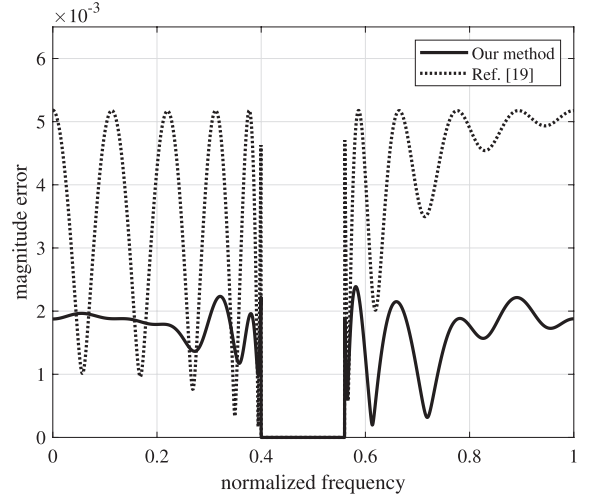
$$s_k^{(i)} = \begin{cases} 1, & \text{if } k = k^{(i)} \\ 0, & \text{otherwise} \end{cases} \quad (35)$$

for $k = 0, 1, \dots, n + \nu$. At the i th step, an $(n + \nu + 1) \times i$ matrix $\mathbf{S}^{(i)}$ is recursively computed as

$$\mathbf{S}^{(i)} = [\mathbf{S}^{(i-1)} \quad \mathbf{s}^{(i)}] \quad (36)$$

where $\mathbf{S}^{(0)} = \mathbf{s}^{(0)}$.

The sparse optimization problem at the i th step is presented as

**Fig. 4** Magnitude of the complex error of the sparse IIR filter designed using our method and the corresponding non-sparse IIR filter designed using the method introduced in [19] in Example 1.**Table 2** Filter coefficients of our filter in Example 1.

$\tilde{a}_0 = 1.0000\text{e}+0$	$\tilde{a}_1 = -1.5986\text{e}-1$	$\tilde{a}_2 = 1.2328\text{e}+0$
$\tilde{a}_3 = -7.5519\text{e}-2$	$\tilde{a}_4 = 2.8731\text{e}-1$	
$\tilde{b}_0 = -1.6376\text{e}-3$	$\tilde{b}_1 = 0.0000\text{e}+0$	$\tilde{b}_2 = 0.0000\text{e}+0$
$\tilde{b}_3 = 2.8597\text{e}-3$	$\tilde{b}_4 = -4.1613\text{e}-3$	$\tilde{b}_5 = -5.9249\text{e}-3$
$\tilde{b}_6 = 3.2946\text{e}-3$	$\tilde{b}_7 = 1.2178\text{e}-2$	$\tilde{b}_8 = -7.0507\text{e}-3$
$\tilde{b}_9 = -3.5737\text{e}-2$	$\tilde{b}_{10} = 1.3397\text{e}-2$	$\tilde{b}_{11} = 2.0119\text{e}-1$
$\tilde{b}_{12} = 4.5608\text{e}-1$	$\tilde{b}_{13} = 6.0490\text{e}-1$	$\tilde{b}_{14} = 5.4322\text{e}-1$
$\tilde{b}_{15} = 3.3602\text{e}-1$	$\tilde{b}_{16} = 1.3451\text{e}-1$	$\tilde{b}_{17} = 2.7346\text{e}-2$
$\tilde{b}_{18} = 0.0000\text{e}+0$	$\tilde{b}_{19} = 0.0000\text{e}+0$	$\tilde{b}_{20} = 0.0000\text{e}+0$

Table 3 Relation between r_c , $e_{\max}^*$ and μ of $e_{\max}^*$ in Example 2.

Rule of Search	r_c	$e_{\max}^*$	μ of $e_{\max}^*$
SCR	0.92	0.04225	19
	0.94	0.04151	20
	0.96	0.03482	19
MIR	0.92	0.03863	21
	0.94	0.03323	21
	0.96	0.02824	21

$$\min_{\tilde{\mathbf{b}}} \quad \tilde{\delta} \quad (37a)$$

$$\text{subject to} \quad \tilde{\mathbf{b}}\mathbf{Q}^{(n+\nu)} - \frac{\tilde{\delta}}{2}\tilde{\mathbf{u}} \leq -\tilde{\mathbf{p}} \quad (37b)$$

$$\tilde{\mathbf{b}}\mathbf{S}^{(i)} = \mathbf{0} \quad (37c)$$

where $\mathbf{0}$ is the zero vector of an appropriate size.

In the SCR, the thinning coefficient is selected as

$$k^{(i)} = \arg \min_{k \in \mathbb{I}_R} |\tilde{b}_k|. \quad (38)$$

On the other hand, if we use the MIR, the thinning coefficient is selected as

$$k^{(i)} = \arg \min_{k \in \mathbb{I}_R} e_{\max}^*. \quad (39)$$

The computation order with the SCR is $O(\nu)$, and that with

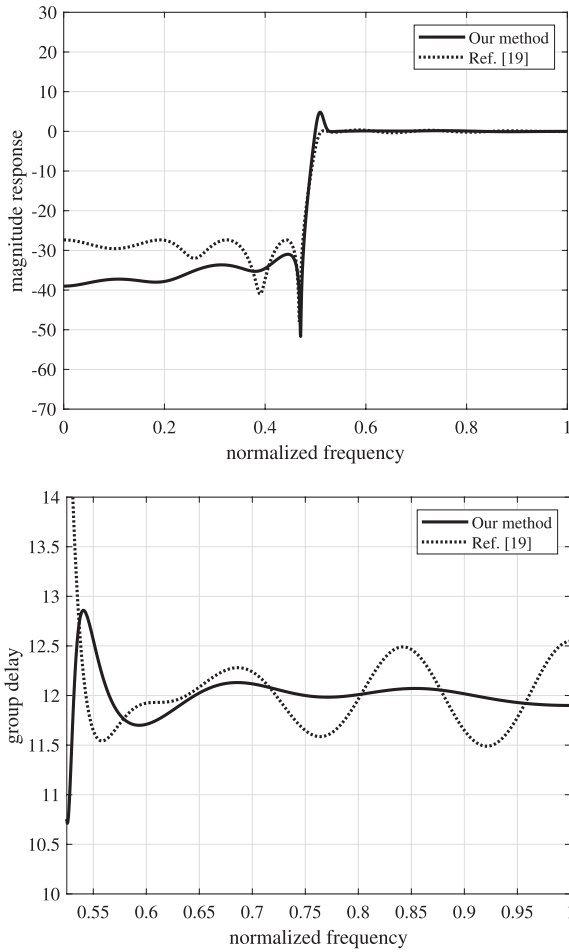


Fig. 5 Magnitude (dB, the upper graph) and group delays in the pass-band (the lower graph) of the responses of the sparse IIR filter designed using our method and the corresponding non-sparse IIR filter designed using the method introduced in [19] in Example 2.

the MIR is $O(n\nu)$ for the ν step procedure.

Next, the following procedure is performed:

$$I_S + \{k^{(i)}\} \rightarrow I_S \quad (40)$$

$$I_R - \{k^{(i)}\} \rightarrow I_R. \quad (41)$$

Finally, the overall procedure of the proposed method is shown in Fig. 2, wherein $e_{\max}^*$ is the minimum value of $e_{\max}$. In addition, the best value of μ is not obvious. Therefore, we obtain it via trial and error by changing as $\mu = \mu_l, \mu_l + 1, \dots, \mu_u$, where μ_l and μ_u are the values of the lower and upper bounds of μ , respectively. In Fig. 2, μ_l and μ_u are values decided by the user. That is, the sparse optimization of the numerator coefficients is performed for each of the denominator coefficients of Eq. (20).

4. Design Examples

To evaluate the performance of our proposed method, we present design examples in this section. The design algo-

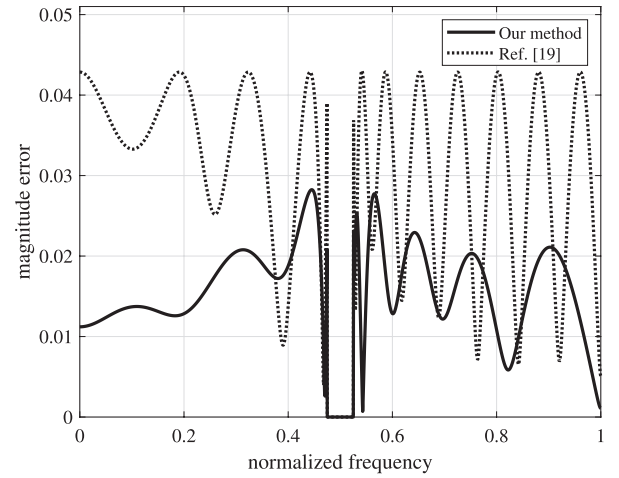


Fig. 6 Magnitude of the complex error of the sparse IIR filter designed using our method and the corresponding non-sparse IIR filter designed using the method introduced in [19] in Example 2.

Table 4 Filter coefficients of our filter in Example 2.

$\tilde{a}_0 = 1.0000\text{e}+0$	$\tilde{a}_1 = 2.1764\text{e}+0$	$\tilde{a}_2 = 5.7815\text{e}+0$
$\tilde{a}_3 = 8.7422\text{e}+0$	$\tilde{a}_4 = 1.2615\text{e}+1$	$\tilde{a}_5 = 1.3896\text{e}+1$
$\tilde{a}_6 = 1.3582\text{e}+1$	$\tilde{a}_7 = 1.1014\text{e}+1$	$\tilde{a}_8 = 7.6683\text{e}+0$
$\tilde{a}_9 = 4.4735\text{e}+0$	$\tilde{a}_{10} = 2.1630\text{e}+0$	$\tilde{a}_{11} = 8.4034\text{e}-1$
$\tilde{a}_{12} = 2.4907\text{e}-1$	$\tilde{a}_{13} = 5.0808\text{e}-2$	$\tilde{a}_{14} = 5.4597\text{e}-3$
$\tilde{b}_0 = 0.0000\text{e}+0$	$\tilde{b}_1 = 6.0020\text{e}-3$	$\tilde{b}_2 = 1.3309\text{e}-2$
$\tilde{b}_3 = 1.7701\text{e}-2$	$\tilde{b}_4 = 1.8327\text{e}-2$	$\tilde{b}_5 = 9.0104\text{e}-3$
$\tilde{b}_6 = 1.0007\text{e}-2$	$\tilde{b}_7 = 0.0000\text{e}+0$	$\tilde{b}_8 = 0.0000\text{e}+0$
$\tilde{b}_9 = 0.0000\text{e}+0$	$\tilde{b}_{10} = 0.0000\text{e}+0$	$\tilde{b}_{11} = -1.7912\text{e}-1$
$\tilde{b}_{12} = 9.1054\text{e}-2$	$\tilde{b}_{13} = -4.8588\text{e}-1$	$\tilde{b}_{14} = 1.8072\text{e}-1$
$\tilde{b}_{15} = -5.0145\text{e}-1$	$\tilde{b}_{16} = 1.2153\text{e}-1$	$\tilde{b}_{17} = -2.3225\text{e}-1$
$\tilde{b}_{18} = 2.7260\text{e}-2$	$\tilde{b}_{19} = -4.0710\text{e}-2$	

rithm is implemented using the MATLAB[†] software package. The weight function is always set to $W(\omega) = 1$ except in the transition band. In addition, the number of grid points over $[0, \pi]$ are always equal to $L = 200$ and $\bar{L} = 2001$. To investigate a suitable range of μ , we set the range of μ to be wide enough as $\mu_l = 1$ and $\mu_u = 50$.

To conduct the first comparison, we need to select an equivalent non-sparse filter. As mentioned earlier, several design methods are proposed, similar to that in [14]–[20]. We select the method of [19] as this is the latest minimax approach for designing a non-sparse IIR filter. In addition, its effectiveness is demonstrated using various design examples. Therefore, the method of [19] and its design examples provide a decent benchmark to evaluate the performance of the sparse filters. Subsequently, our design method is compared with the existing design method of a sparse IIR filter introduced in [34].

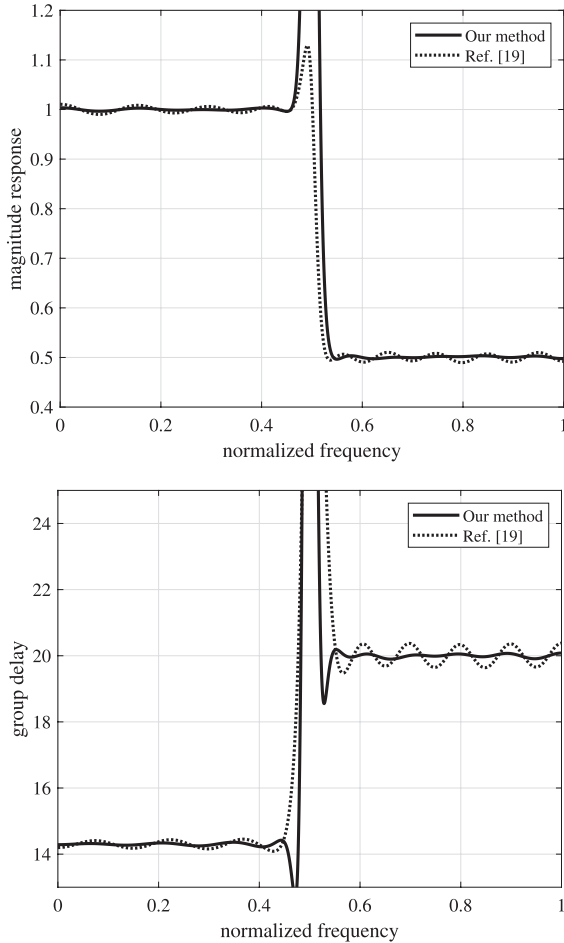
4.1 Comparison with Equivalent Non-Sparse IIR Filters [19]

Example 1: We consider a low-pass filter whose desired

[†]MATLAB is a trademark of *The Math Works Inc.*

Table 5 Relation between r_c , $e_{\max}^*$ and μ of $e_{\max}^*$ in Example 3.

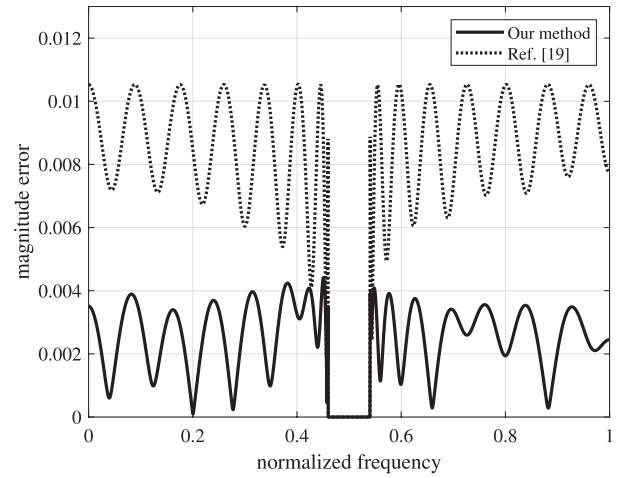
Rule of Search	r_c	$e_{\max}^*$	μ of $e_{\max}^*$
SCR	0.92	0.006660	50
	0.94	0.005662	46
	0.96	0.004955	41
MIR	0.92	0.006083	50
	0.94	0.005477	45
	0.96	0.004419	41

**Fig. 7** Magnitude (the upper graph) and group delays in the pass-band (the lower graph) of the responses of the sparse IIR filter designed using our method and the corresponding non-sparse IIR filter designed using the method introduced in [19] in Example 3.

response is given by

$$H_d(\omega) = \begin{cases} e^{-j12\omega}, & 0 \leq \omega \leq 0.4\pi \\ 0, & 0.56\pi \leq \omega < \pi. \end{cases}$$

The orders of the sparse IIR filter are $m = 4$, $n_+ = 20$, and $\nu = 5$. We design the IIR filters with $r_c = 0.92, 0.94$, and 0.96 . Table 1 summarizes $e_{\max}^*$ and its μ . In that case, the optimal solution is obtained with $r_c = 0.96$ and $\mu = 17$ using the MIR. The magnitude (dB) and the group delays of the responses of the optimal solution are shown in Fig. 3. Moreover, Fig. 4 indicates the magnitude of the complex error of our filter (solid line) and that of the equivalent non-sparse IIR filter with $m = 4$, and $n = 15$ (denoted by the dotted

**Fig. 8** Magnitude of the complex error of the sparse IIR filter designed using our method and the corresponding non-sparse IIR filter designed using the method introduced in [19] in Example 3.**Table 6** Filter coefficients of our filter in Example 3.

$\tilde{a}_0 = 1.0000\text{e}+0$	$\tilde{a}_1 = -2.0954\text{e}-2$	$\tilde{a}_2 = 2.5927\text{e}+0$
$\tilde{a}_3 = -3.6053\text{e}-2$	$\tilde{a}_4 = 2.2306\text{e}+0$	$\tilde{a}_5 = -1.5429\text{e}-2$
$\tilde{a}_6 = 6.3643\text{e}-1$		
$\tilde{b}_0 = 0.0000\text{e}+0$	$\tilde{b}_1 = 3.5432\text{e}-3$	$\tilde{b}_2 = 9.8895\text{e}-4$
$\tilde{b}_3 = 4.0409\text{e}-3$	$\tilde{b}_4 = 0.0000\text{e}+0$	$\tilde{b}_5 = 5.0196\text{e}-3$
$\tilde{b}_6 = 0.0000\text{e}+0$	$\tilde{b}_7 = 0.0000\text{e}+0$	$\tilde{b}_8 = -2.3884\text{e}-3$
$\tilde{b}_9 = 6.1194\text{e}-3$	$\tilde{b}_{10} = 3.0845\text{e}-3$	$\tilde{b}_{11} = -1.5119\text{e}-2$
$\tilde{b}_{12} = -1.5897\text{e}-2$	$\tilde{b}_{13} = 9.1572\text{e}-2$	$\tilde{b}_{14} = 3.7624\text{e}-1$
$\tilde{b}_{15} = 8.0152\text{e}-1$	$\tilde{b}_{16} = 1.2002\text{e}+0$	$\tilde{b}_{17} = 1.3688\text{e}+0$
$\tilde{b}_{18} = 1.1998\text{e}+0$	$\tilde{b}_{19} = 7.4314\text{e}-1$	$\tilde{b}_{20} = 6.6899\text{e}-1$
$\tilde{b}_{21} = -3.4232\text{e}-1$	$\tilde{b}_{22} = 6.7210\text{e}-1$	$\tilde{b}_{23} = -7.0085\text{e}-1$
$\tilde{b}_{24} = 5.6452\text{e}-1$	$\tilde{b}_{25} = -3.5319\text{e}-1$	$\tilde{b}_{26} = 1.6222\text{e}-1$
$\tilde{b}_{27} = -4.0294\text{e}-2$	$\tilde{b}_{28} = 0.0000\text{e}+0$	$\tilde{b}_{29} = 7.8202\text{e}-3$

line) [19].

Table 2 presents the filter coefficients of our filter. The number of non-zero coefficients of the non-sparse IIR filter is 20, which is identical to that of the sparse filter.

Example 2: We design a half-band high-pass filter. The desired response is given by

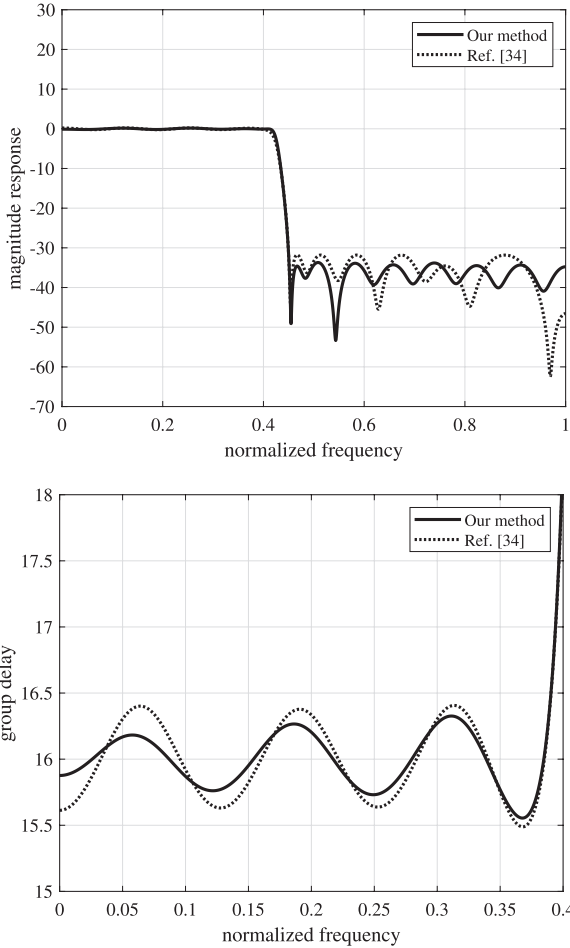
$$H_d(\omega) = \begin{cases} 0, & 0 \leq \omega \leq 0.475\pi \\ e^{-j12\omega}, & 0.525\pi \leq \omega < \pi. \end{cases}$$

The orders of the sparse IIR filters are set as $m = 14$, $n_+ = 19$, and $\nu = 5$. Similar to Example 1, we design the IIR filters with $r_c = 0.92, 0.94$, and 0.96 . Table 3 shows the relationship between $e_{\max}^*$ and its μ . In that case, the optimal solution is computed with $r_c = 0.96$ and $\mu = 21$ using the MIR.

The magnitude (dB) and the group delays of the responses of the optimal case are shown in Fig. 5. Furthermore, Fig. 6 shows the comparison of the magnitude of the complex error of the sparse IIR filter designed by our method (denoted by the solid line), and the equivalent non-sparse IIR filter with $m = 14$ and $n = 14$ (denoted by the dotted line) [19]. The filter coefficients of our filter are summarized in Table 4.

Table 7 Relation between r_c , $e_{\max}^*$ and μ of $e_{\max}^*$ in Example 4 (a).

Rule of Search	r_c	$e_{\max}^*$	μ of $e_{\max}^*$
SCR	0.92	0.047254	28
	0.94	0.034563	35
	0.96	0.024335	28
MIR	0.92	0.04459	23
	0.94	0.03176	28
	0.96	0.02246	42

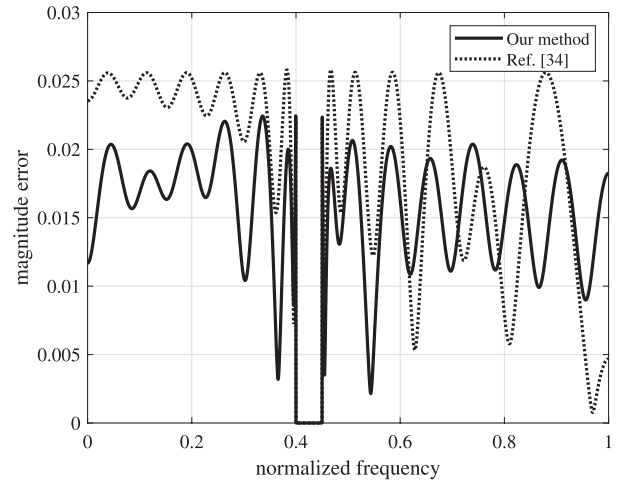
**Fig. 9** Magnitude (dB, the upper graph) and group delays in the pass-band (the lower graph) of the responses of the sparse IIR filter designed using our method and the method introduced in [34] in Example 4 (a).

Example 3: Next, we consider a two-band filter whose desired response is as follows.

$$H_d(\omega) = \begin{cases} e^{-j14.3\omega}, & 0 \leq \omega \leq 0.46\pi \\ 0.5e^{-j20\omega}, & 0.54\pi \leq \omega < \pi. \end{cases}$$

The orders of the sparse IIR filter are $m = 6$, $n_+ = 29$, and $\nu = 5$. This is a decent benchmark problem as the gains and the group delays are different in each band. Table 5 summarizes $e_{\max}^*$ and its μ with $r_c = 0.92, 0.94$, and 0.96 . The optimal solution is achieved using the MIR when $r_c = 0.96$ and $\mu = 41$.

The magnitude and group delays of the responses in the best case are illustrated in Fig. 7. Moreover, Fig. 8 shows the

**Fig. 10** Magnitude of the complex error of the sparse IIR filter designed using our method and the method introduced in [34] in Example 4 (a).**Table 8** Filter coefficients of our filter in Example 4 (a).

$\tilde{a}_0 = 1.0000\text{e}+0$	$\tilde{a}_1 = -4.4981\text{e}-1$	$\tilde{a}_2 = 9.2160\text{e}-1$
$\tilde{b}_0 = 6.046\text{e}-3$	$\tilde{b}_1 = 1.2950\text{e}-2$	$\tilde{b}_2 = -3.1172\text{e}-3$
$\tilde{b}_3 = 0.0000\text{e}+0$	$\tilde{b}_4 = 0.0000\text{e}+0$	$\tilde{b}_5 = 0.0000\text{e}+0$
$\tilde{b}_6 = 0.0000\text{e}+0$	$\tilde{b}_7 = -5.1709\text{e}-3$	$\tilde{b}_8 = -1.2388\text{e}-2$
$\tilde{b}_9 = 2.3958\text{e}-3$	$\tilde{b}_{10} = 1.4842\text{e}-2$	$\tilde{b}_{11} = 0.0000\text{e}+0$
$\tilde{b}_{12} = -2.6532\text{e}-2$	$\tilde{b}_{13} = -2.9997\text{e}-2$	$\tilde{b}_{14} = 4.5997\text{e}-2$
$\tilde{b}_{15} = 2.0150\text{e}-1$	$\tilde{b}_{16} = 3.5007\text{e}-1$	$\tilde{b}_{17} = 4.0323\text{e}-1$
$\tilde{b}_{18} = 3.2565\text{e}-1$	$\tilde{b}_{19} = 1.6740\text{e}-1$	$\tilde{b}_{20} = 3.8576\text{e}-2$
$\tilde{b}_{21} = -2.0830\text{e}-2$	$\tilde{b}_{22} = -2.2070\text{e}-2$	$\tilde{b}_{23} = 7.5102\text{e}-3$
$\tilde{b}_{24} = 0.0000\text{e}+0$	$\tilde{b}_{25} = 9.9574\text{e}-3$	$\tilde{b}_{26} = -1.1429\text{e}-2$

Table 9 Relation between r_c , $e_{\max}^*$ and μ of $e_{\max}^*$ in Example 4 (b).

Rule of Search	r_c	$e_{\max}^*$	μ of $e_{\max}^*$
SCR	0.92	0.047329	26
	0.94	0.034878	27
	0.96	0.025001	28
MIR	0.92	0.044482	25
	0.94	0.031681	26
	0.96	0.023706	50

comparison between the sparse IIR filter designed using the proposed method (denoted by the solid line) and the equivalent non-sparse IIR filter with $m = 6$ and $n = 24$ (denoted by the dotted line) [19]. Table 6 indicates the filter coefficients of our designed filter.

4.2 Comparison with the Sparse IIR Filters [34]

Example 4: We consider the sparse IIR filter presented in [34]. The desired response is given by

$$H_d(\omega) = \begin{cases} e^{-j16\omega}, & 0 \leq \omega \leq 0.4\pi \\ 0, & 0.45\pi \leq \omega < \pi. \end{cases}$$

The orders of the filter are $m = 2$ and $n_+ = 26$ with cases of (a) $\nu = 6$ and (b) $\nu = 8$ such that the orders and the numbers of zero coefficients of both filters are equal.

Similar to previous examples, we design the filter with

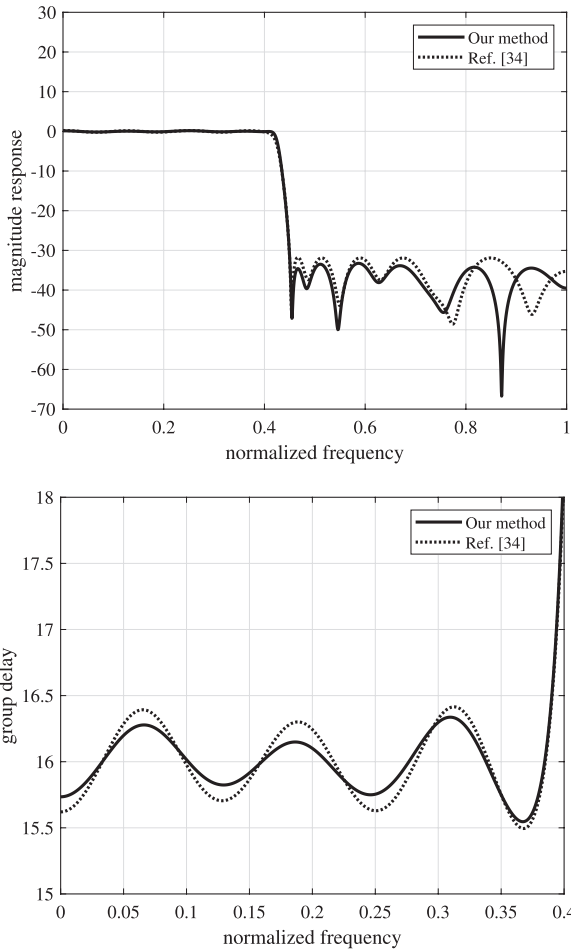


Fig. 11 Magnitude (dB, the upper graph) and group delays in the pass-band (the lower graph) of the responses of the sparse IIR filter designed using our method and the method introduced in [34] in Example 4 (b).

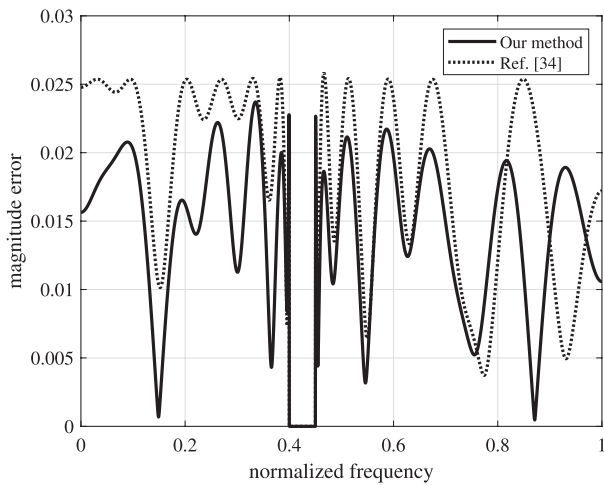


Fig. 12 Magnitude of the complex error of the sparse IIR filter designed using our method and the method introduced in [34] in Example 4 (b).

$r_c = 0.92, 0.94$, and 0.96 in the case of (a). $e_{\max}^*$ of the optimal solution and its μ are summarized in Table 7. In that case, the optimal solution is obtained with $r_c = 0.96$

Table 10 Filter coefficients of our filter in Example 4 (b).

$\bar{a}_0 = 1.0000\text{e}+0$	$\bar{a}_1 = -4.4887\text{e}-1$	$\bar{a}_2 = 9.2160\text{e}-1$
$\bar{b}_0 = 6.8797\text{e}-3$	$\bar{b}_1 = 1.0780\text{e}-2$	$\bar{b}_2 = 0.0000\text{e}+0$
$\bar{b}_3 = 0.0000\text{e}+0$	$\bar{b}_4 = 0.0000\text{e}+0$	$\bar{b}_5 = 8.4060\text{e}-3$
$\bar{b}_6 = 0.0000\text{e}+0$	$\bar{b}_7 = 0.0000\text{e}+0$	$\bar{b}_8 = -4.1003\text{e}-3$
$\bar{b}_9 = 0.0000\text{e}+0$	$\bar{b}_{10} = 2.1260\text{e}-2$	$\bar{b}_{11} = 0.0000\text{e}+0$
$\bar{b}_{12} = -2.5776\text{e}-2$	$\bar{b}_{13} = -3.1625\text{e}-2$	$\bar{b}_{14} = 4.9402\text{e}-2$
$\bar{b}_{15} = 1.9643\text{e}-1$	$\bar{b}_{16} = 3.5427\text{e}-1$	$\bar{b}_{17} = 4.0246\text{e}-1$
$\bar{b}_{18} = 3.2660\text{e}-1$	$\bar{b}_{19} = 1.6847\text{e}-1$	$\bar{b}_{20} = 4.3032\text{e}-2$
$\bar{b}_{21} = -2.5501\text{e}-2$	$\bar{b}_{22} = -1.1525\text{e}-2$	$\bar{b}_{23} = 0.0000\text{e}+0$
$\bar{b}_{24} = 9.0602\text{e}-3$	$\bar{b}_{25} = 5.9189\text{e}-3$	$\bar{b}_{26} = -8.6887\text{e}-3$

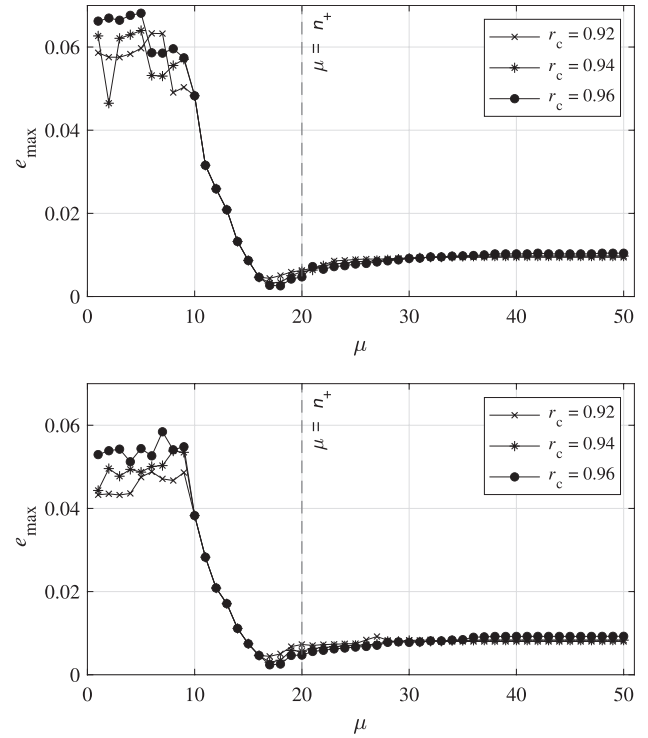


Fig. 13 Relation between μ and $e_{\max}$ when using the SCR (the upper graph) and MIR (the lower graph) in Example 1.

and $\mu = 42$ using the MIR.

The magnitude (dB) and group delays of the responses of our filter are shown in Fig. 9. Furthermore, Fig. 10 demonstrates the comparison of the magnitude of the complex error in the case of (a) $\nu = 6$ between our filter (denoted by the solid line) and the sparse IIR filter designed in [34] (denoted by the dotted line). The filter coefficients of our filter are presented in Table 8.

In the case of (b), $e_{\max}^*$ and its μ are summarized in Table 9. The optimal solution is computed with $r_c = 0.96$ and $\mu = 50$ using the MIR. The magnitude in dB and group delays of the responses of our filter are shown in Fig. 11. Similar to the case (a), Fig. 12 shows the comparison of the magnitude of the complex error of the filters in the case (b) $\nu = 8$. The filter coefficients of our filter are summarized in Table 10.

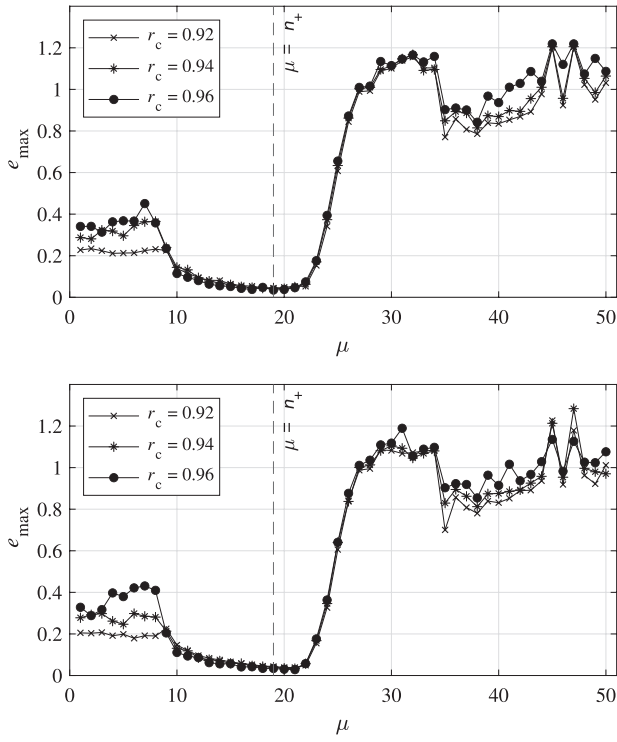


Fig. 14 Relation between μ and $e_{\max}$ when using the SCR (the upper graph) and MIR (the lower graph) in Example 2.

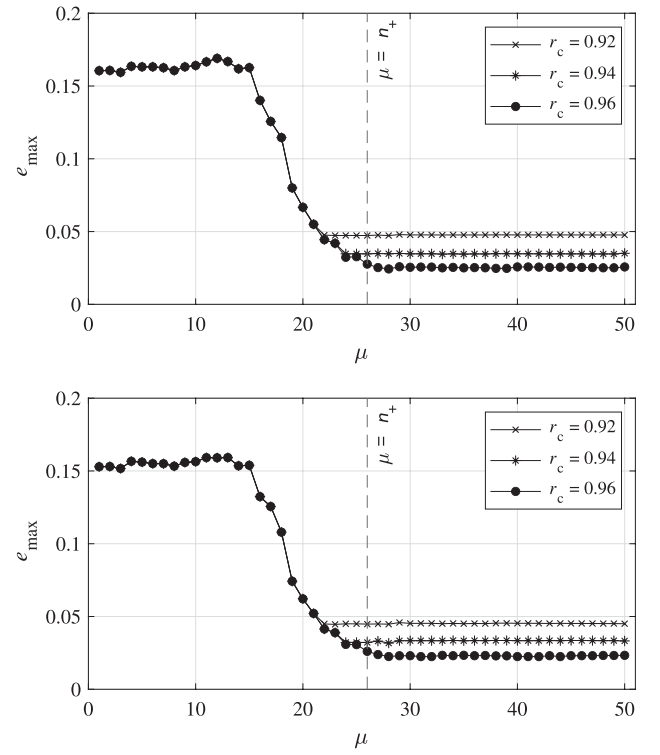


Fig. 16 Relation between μ and $e_{\max}$ when using the SCR (the upper graph) and MIR (the lower graph) in Example 4 (a).

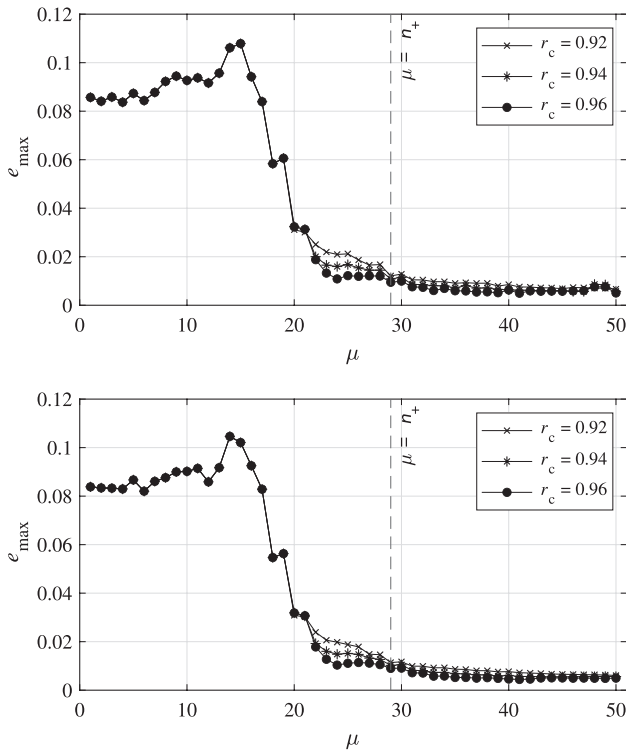


Fig. 15 Relation between μ and $e_{\max}$ when using the SCR (the upper graph) and MIR (the lower graph) in Example 3.

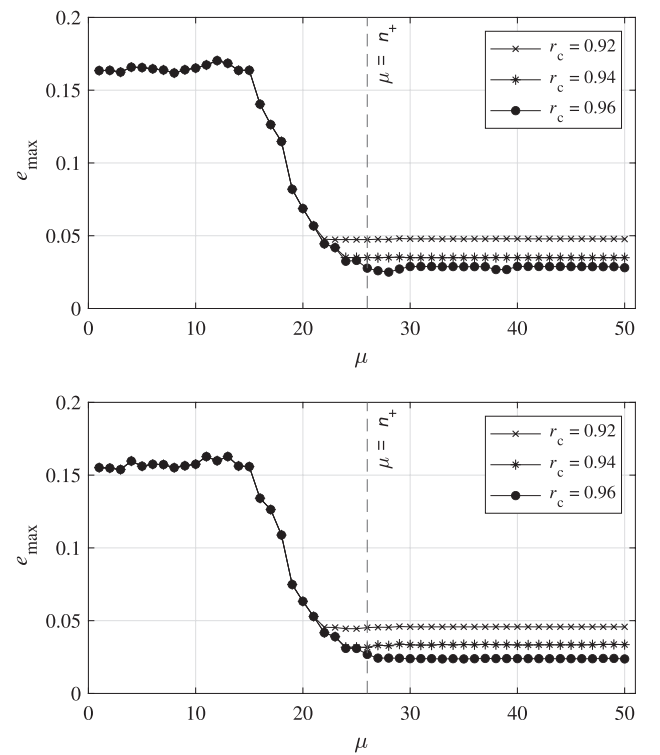


Fig. 17 Relation between μ and $e_{\max}$ when using the SCR (the upper graph) and MIR (the lower graph) in Example 4 (b).

Table 11 Relation between ν and $e_{\max}^*$ with $r_c = 0.96$ in Example 1.

ν	$e_{\max}^*$ when using the SCR	$e_{\max}^*$ when using the MIR
1	0.004642	0.004642
2	0.002574	0.002388
3	0.002574	0.002388
4	0.002574	0.002388
5	0.002574	0.002388
6	0.002574	0.002388
7	0.002574	0.002388
8	0.002574	0.002388
9	0.002574	0.002388
10	0.002574	0.002388

Table 12 Relation between ν and $e_{\max}^*$ with $r_c = 0.96$ in Example 2.

ν	$e_{\max}^*$ when using the SCR	$e_{\max}^*$ when using the MIR
1	0.04282	0.04195
2	0.03870	0.03796
3	0.03482	0.03405
4	0.03563	0.02917
5	0.03482	0.02824
6	0.03563	0.02330
7	0.03425	0.02202
8	0.03106	0.02521
9	0.03106	0.02510
10	0.03106	0.02510

4.3 Discussion

The results of all the examples indicate that the peak error of our proposed method is smaller than that of the existing methods. Hence, from the results of Examples 1–3, we demonstrated the superiority of the sparse IIR filter against the non-sparse filter. Moreover, we confirmed the effectiveness of our design algorithm compared with the IIR sparse filter introduced in [34]. In addition, according to Tables 1, 3, 5, 7, 9, the peak error, except for the transition region, increases as r_c increases. However, the peak in the transition region will increase as r_c increases.

Furthermore, μ is used to obtain a better performance of the filter. Figures 13–17 show the relationship between μ and $e_{\max}$ with a straight line $\mu = n_+$ in Examples 1–4, where $r_c = 0.92, 0.94$, and 0.96 . In Fig. 13, the upper graph is for the SCR and the lower graph is for the MIR. It is natural to set $\mu = n_+$ because the conclusive order of the numerator polynomial is n_+ . However, the results show that $\mu = n_+$ is not optimal. Moreover, from the results of Examples 1–4, it is evident that the performance is not acceptable when μ is sufficiently lower than n_+ . Further, when μ is sufficiently higher than n_+ , the performance is not acceptable or $e_{\max}$ is converged. Hence, it is advisable to search for μ around n_+ so as to obtain a decent value of μ . Consequently, we may achieve a better solution of μ if we search within a wider range. However, a dramatic improvement is not gained as it can be time consuming.

Further, we investigate how $e_{\max}^*$ is changed when ν increases. Tables 11–13 summarize $e_{\max}^*$ for $\nu = 1, 2, \dots, 10$ with $r_c = 0.96$ in Examples 1–3. Tables 11–13 and Tables 1, 3, 5, 7, 9 indicate that using the sparse linear programming

Table 13 Relation between ν and $e_{\max}^*$ with $r_c = 0.96$ in Example 3.

ν	$e_{\max}^*$ when using the SCR	$e_{\max}^*$ when using the MIR
1	0.01136	0.01046
2	0.01086	0.01027
3	0.006473	0.006146
4	0.006473	0.006265
5	0.004955	0.004419
6	0.004955	0.004419
7	0.004136	0.003578
8	0.004136	0.003578
9	0.004136	0.003693
10	0.004136	0.003722

with the MIR is usually better than the SCR. However, the computation cost with the MIR is approximately n_+ times of that of the SCR. Moreover, better solutions are generated when ν is increased. However, in Table 13 (when using the MIR), the solution with $\nu = 10$ is worse than that of $\nu = 7$ because the algorithm cannot find a sound solution. That is, when ν is large, the search space will be extremely large. Accordingly, the algorithm cannot search the solution in every part of the space.

5. Conclusions

We proposed the minimax design for the IIR filter with zero coefficients (known as sparse IIR filter). To confirm the effectiveness of the proposed method, we compared our sparse IIR filter with not only corresponding non-sparse IIR filters but also sparse IIR filters designed using the existing methods. In conclusion, we have demonstrated the utility of the proposed algorithm for the minimax design of IIR filters. Thus, the proposed method can be used to design high-performance IIR filters.

Acknowledgments

The authors would like to express sincere thanks to the anonymous reviewers for their valuable comments.

References

- [1] A. Antoniou and W.-S. Lu, *Practical Optimization: Algorithms and Engineering Applications*, Springer, 2007.
- [2] A.V. Oppenheim and R.W. Schaffer, *Discrete-Time Signal Processing*, Prentice Hall, 2009.
- [3] S. Koshita and K. Suyama, “Research activities and future tasks on digital filters — To obtain “Good” digital filters and bring them to widespread use —,” *IEICE ESS Fundamentals Review*, vol.9, no.1, pp.24–36, 2015 (in Japanese).
- [4] T. Hinamoto and W.-S. Lu, *Digital Filter Design and Realization*, River Publishers, June 2017.
- [5] T. Parks and J. McClellan, “Chebyshev approximation for nonrecursive digital filters with linear phase,” *IEEE Trans. Circuit Theory*, vol.19, no.2, pp.189–194, March 1972.
- [6] J. McClellan, T. Parks, and L. Rabiner, “A computer program for designing optimum FIR linear phase digital filters,” *IEEE Trans. Audio and Electroacoust.*, vol.21, no.6, pp.506–526, Dec. 1973.
- [7] V.R. Algazi, M. Suk, and C.S. Rim, “Design of almost minimax FIR filters in one and two dimensions by WLS techniques,” *IEEE Trans. Circuits Syst.*, vol.CAS-33, no.6, pp.590–596, June 1986.

- [8] P. Vaidyanathan and T. Nguyen, "Eigenfilters: A new approach to least-squares FIR filter design and applications including Nyquist filters," *IEEE Trans. Circuits Syst.*, vol.34, no.1, pp.11–23, Jan. 1987.
- [9] J.W. Adams, "FIR digital filters with least-squares stopbands subject to peak-gain constraints," *IEEE Trans. Circuits Syst.*, vol.38, no.4, pp.376–388, April 1991.
- [10] C.S. Burrus, J.A. Barreto, and I.W. Selesnick, "Iterative reweighted least-squares design of FIR filters," *IEEE Trans. Signal Process.*, vol.42, no.11, pp.2926–2936, Nov. 1994.
- [11] L.J. Karam and J.H. McClellan, "Complex Chebyshev approximation for FIR filter design," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol.42, no.3, pp.207–216, March 1995.
- [12] D. Burnside and T.W. Parks, "Optimal design of FIR filters with the complex Chebyshev error criteria," *IEEE Trans. Signal Process.*, vol.43, no.3, pp.605–616, March 1995.
- [13] M. Okuda, M. Ikehara, and S. Takahashi, "Fast and stable least-squares approach for the design of linear phase FIR filters," *IEEE Trans. Signal Process.*, vol.46, no.6, pp.1485–1493, June 1998.
- [14] A.T. Chottera and G.A. Jullien, "A linear programming approach to recursive digital filter design with linear phase," *IEEE Trans. Circuits Syst.*, vol.CAS-29, no.3, pp.139–149, March 1982.
- [15] W.-S. Lu, S.-C. Pei, and C.-C. Tseng, "A weighted least-squares method for the design of stable 1-D and 2-D IIR digital filters," *IEEE Trans. Signal Process.*, vol.46, no.1, pp.1–10, Jan. 1998.
- [16] M.C. Lang, "Least-squares design of IIR filters with prescribed magnitude and phase responses and a pole radius constraint," *IEEE Trans. Signal Process.*, vol.48, no.11, pp.3109–3121, Nov. 2000.
- [17] A. Tarczynski, G.D. Cain, E. Hermanowicz, and M. Rojewski, "A WISE method for designing IIR filters," *IEEE Trans. Signal Process.*, vol.49, no.7, pp.1421–1432, July 2001.
- [18] W.-S. Lu and T. Hinamoto, "Optimal design of IIR digital filters with robust stability using conic quadratic-programming updates," *IEEE Trans. Signal Process.*, vol.51, no.6, pp.1581–1592, June 2003.
- [19] A. Jiang and H.K. Kwan, "Minimax design of IIR digital filters using iterative SOCP," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol.57, no.6, pp.1326–1337, June 2010.
- [20] M. Nakamoto and S. Ohno, "Design of multi-band digital filters and full-band digital differentiators without frequency sampling and iterative optimization," *IEEE Trans. Ind. Electron.*, vol.61, no.9, pp.4857–4866, Sept. 2014.
- [21] M. Smith and D. Farden, "Thinning the impulse response of FIR digital filters," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, (ICASSP), pp.240–242, 1981.
- [22] J.L.H. Webb and D.C. Munson, "Chebyshev optimization of sparse FIR filters using linear programming with an application to beam-forming," *IEEE Trans. Signal Process.*, vol.44, pp.1912–1922, Aug. 1996.
- [23] D. Mattered, F. Palmieri, and S. Haykin, "Efficient sparse FIR digital filters," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, (ICASSP), pp.1537–1540, May 2002.
- [24] T. Baran, D. Wei, and A.V. Oppenheim, "Linear programming algorithms for sparse filter design," *IEEE Trans. Signal Process.*, vol.58, no.3, pp.1605–1617, March 2010.
- [25] A. Jiang, H.K. Kwan, and Y. Zhu, "Peak-error-constrained sparse FIR filter design using iterative SOCP," *IEEE Trans. Signal Process.*, vol.60, no.8, pp.4035–4044, Aug. 2012.
- [26] D. Wei, C.K. Sestok, and A.V. Oppenheim, "Sparse filter design under a quadratic constraint: Low-complexity algorithms," *IEEE Trans. Signal Process.*, vol.61, no.4, pp.857–870, Feb. 2013.
- [27] D. Wei and A.V. Oppenheim, "A branch-and-bound algorithm for quadratically-constrained sparse filter design," *IEEE Trans. Signal Process.*, vol.61, no.4, pp.1006–1018, Feb. 2013.
- [28] A. Jiang and H.K. Kwan, "WLS design of sparse FIR digital filters," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol.60, no.1, pp.125–135, Jan. 2013.
- [29] A. Jiang, H.K. Kwan, Y. Zhu, X. Liu, N. Xu, and Y. Tang, "Design of sparse FIR filters with joint optimization of sparsity and filter order," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol.62, no.1, pp.195–204, Jan. 2015.
- [30] W.-S. Lu and T. Hinamoto, "A unified approach to the design of interpolated and frequency-response-masking FIR filters," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol.63, no.12, pp.2257–2266, Dec. 2016.
- [31] W. Chen, M. Huang, and X. Lou, "Design of sparse FIR filters with reduced effective length," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol.66, no.4, pp.1496–1506, April 2019.
- [32] W. Chen, M. Huang, W. Ye, and X. Lou, "Cascaded form sparse FIR filter design," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol.67, no.5, pp.1692–1703, May 2020.
- [33] W.-S. Lu and T. Hinamoto, "Minimax design of stable IIR filters with sparse coefficients," *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, pp.398–401, May 2011.
- [34] W.-S. Lu and T. Hinamoto, "New algorithm for minimax design of sparse IIR filters," *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, pp.2920–2923, May 2013.



Masayoshi Nakamoto received the B.E. and M.E. degrees from Okayama University of Science, Okayama, Japan, in 1997 and 1999, respectively, and the Dr.Eng. degree from Hiroshima University, Higashi-Hiroshima, Japan, in 2002. From April 2002 to March 2005, he was JSPS Research fellow. He is currently a research associate of Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima, Japan. His research interests are in the areas of stochastic process, digital signal processing and optimization. He is a member of IEEEJ, SICE and IEEE.



Naoyuki Aikawa received the B.E. degree from Yamanashi University, Yamanashi, Japan, in 1985 and the M.S. degree in electrical engineering from Tokyo Metropolitan University, Tokyo, Japan, in 1987, and the Dr. degree in electrical engineering from Tokyo Metropolitan University, Tokyo, Japan, in 1992, all in electrical engineering. Currently, he is a Professor in the Department of Applied Electronics, Tokyo University of Science. His teaching and research interests are in the areas of digital and analog signal processing and education. He is a member of IEEEJ, SICE and IEEE.