Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 3, No. 1, pp. 73 – 86, January 2007

DESIGN OF FIR FILTER WITH DISCRETE COEFFICIENTS BASED ON SEMI-INFINITE LINEAR PROGRAMMING METHOD

Rika Ito and Ryuichi Hirabayashi

This paper is dedicated to Professor Masakazu Kojima on the occasion of his 60th birthday.

Abstract: In this paper, we propose a new design method of FIR filters with Signed Power of Two (SP2) coefficients. In the proposed method, the design problem of FIR filters is formulated as a discrete semi-infinite linear programming problem (DSILP), and the DSILP is solved using a branch and bound technique. We guarantee the optimality of the obtained solution. It is confirmed that the optimal coefficients of linear phase FIR filter with the SP2 coefficients could be designed fast with enough precisions by the computational experiments.

Key words: *semi-infinite pogramming problem (SILP), FIR filters, optimization, 3 phase method, signed power of two (SP2)*

Mathematics Subject Classification: *90C34*

1 Introduction

In recent years tremendous advances have been achieved in computer hardware as well as in digital technology in general. Significant reductions in the cost, size, and power consumption of digital hardware have led to increasingly widespread application. Digital systems are finding their way into our lives in computers and communications. For many diverse applications, information is now most conveniently recorded, transmitted, and stored in digital form. As a result, digital signal processing (DSP) has become an exceptionally important modern tool.

Digital signal processing deals with the representation of signals as ordered sequencers of numbers and the processing of those sequences. Typical reasons for signal processing include: estimation of characteristic signal parameters, elimination or reduction of unwanted interference, and transformation of a signal into a form that is in some sense more informative.

For a signal to be completely representable and storable in a digital computer memory, it must be sampled in time and quantized in value. That is, it must be a practical digital signal with both finite duration and a finite number of quantized values. Very long sequences can be processed much at a time. To quantize the value, a rounding or quantization procedure must be used. However once sampled and converted to a fixed bit-length binary form, the signal data are extremely convenient. These data can be stored on hard disks or diskettes, on magnetic tape, or in semiconductor memory chips. All the advantages of digital processing are now available to handle them as digital data on the computer.. Unfortunately, these signal data usually contain also noise data. To eliminate the noise data, we use the so-called filter. There are at least two types of filters, that have finite impulse responses (FIR filter) and infinite impulse responses (IIR filter). Both filters are studied very deeply. In this paper, we deal with the FIR filter.

There are two methods for the realization of FIR filter, one is a software realization method and another is a hardware realization by using digital circuits. In hardware implementation of FIR filters, the filter coefficients corresponding to multiplier coefficients are presented as values composed of the finite word length numbers. When the coefficients are simply rounded to the nearest discrete number, precision of filters are degraded from the one with the optimal real coefficients. Therefore, design methods of FIR filters with discrete coefficients have been widely researched [14], [24].

There are no design methods of designing filters that could be easily adapted to special design specifications. So each filter has to be designed, in principle, by a complete mathematical design procedure. It is the aim of all design methods to approximate a desired frequency response as close as possible by a finite number of FIR filter coefficients. The starting point of all these methods is the assumption of idealized frequency responses or tolerance specifications in the passband and stopband. Low variation of the magnitude (ripple) in the passband, high attenuation in the stopband and sharp cut-off are competing design parameters in this context. Some of error measures are generally used in FIR filter design. One is the average of the squared error in the frequency-response approximation. The second is the maximum of the error over specified regions of the frequency response and so on. The method based on the first error measure is called a least squared (LS) approximation, the second a Chebyshev approximation or equi-ripple approximation. And equi-ripple approximation is much important since the characteristic of the response function is much better than the one obtained by the LS approximation.

Recently, many studies on a design method for linear phase FIR filters with discrete coefficients have been published [19], [23], in which, a numerical representation by a sum of signed power of two (SP2) has been used in several methods [16], [19], [20], [27]. It is a reason that a small number of non-zero digits is often required for a representation of the coefficients in a VLSI implementation of the filters. There exist a lot of studies to obtain an approximated solution for this design problem. See, for example, Ito et. al [12], W.-S. Lu [23]. They proposed to use a semidefinite programming (SDP) relaxation method for the design problem. However, if we do not have the optimal solution for the design problem, we cannot mention the performance of the approximation method precisely.

Since the design problem is formulated as a discrete semi-infinite linear programming problem, the most practical method to solve the problem is to use the branch and bound (B & B) method. And, there are some methods using B & B method for the FIR design problem, for example, based on LP, Remez algorithm, and so on. Cho et. al [1] proposed a B & B method based on LP focusing only on the active constraints to decrease the computational time. However, they did not assure the optimality of the solution obtained by the algorithm.

In this paper, we propose a new design method of linear phase FIR filters with SP2 coefficients which guarantees the optimality of the obtained solution. In the proposed method, the design problem is formulated as a discrete semi-infinite linear programming problem (DSILP) and solved by B & B method. In the B & B method, a branching tree is generated and, on each node, it is necessary to solve semi-infinite linear programming problem (SILP) [6].

It is shown by the results of some computational experiments for the filter designing problem, the developed algorithm is rather practical.

2 Problem Formulation

In this section, we introduce the design method of digital FIR filters with SP2 coefficients. A design problem of FIR digital filters is considered to minimize the maximal error, i.e., minimize the following function:

$$e = \max_{\omega \in \Omega} |H(e^{j\omega}) - H_d(\omega)| \tag{1}$$

where $H_d(\omega)$ is a desired frequency response function and $\Omega = [0, \omega_p] \cup [\omega_s, \pi]$ is an approximation band. Here, $[0, \omega_p]$ denotes a passband and $[\omega_s, \pi]$ denotes a stopband. ω_p is a passband cutoff frequency and ω_s is a stopband cutoff frequency.

In the first, we consider the continuous coefficient case. Then the design function of the FIR filter is:

$$H(e^{j\omega}) = \sum_{k=0}^{N-1} h_k e^{-jk\omega} \tag{2}$$

for $h_k \in \mathbb{R}, k = 0, \ldots, N-1$ where $\mathbb{R}$ is the set of real numbers.

An FIR filter is easily realized to produce a linear phase response. Then, the corresponding coefficients of the FIR filter have even/odd symmetrical property with respect to their midpoint, that means, $h_k = h_{N-1-k}$, $k = 0, \ldots, \lfloor (N-1)/2 \rfloor$ or $h_k = -h_{N-1-k}$, $k = 0, \ldots, \lfloor (N-1)/2 \rfloor$. Where $\lfloor a \rfloor$ denotes the maximum integer that does not exceed a. Linear phase FIR filter has an important property that the group delay is constant. The implication of constant group delay is that all frequency components of an input sequence are similarly delayed in the output sequence. The shapes of impulse response of FIR filter are classified into four types by filter length N and even or odd symmetry characteristic. These four cases are illustrated in Figure 1. In our proposed method, we consider FIR filter of type1 because it makes possible to design all types of filters (high-pass, low-pass and band-pass filters).

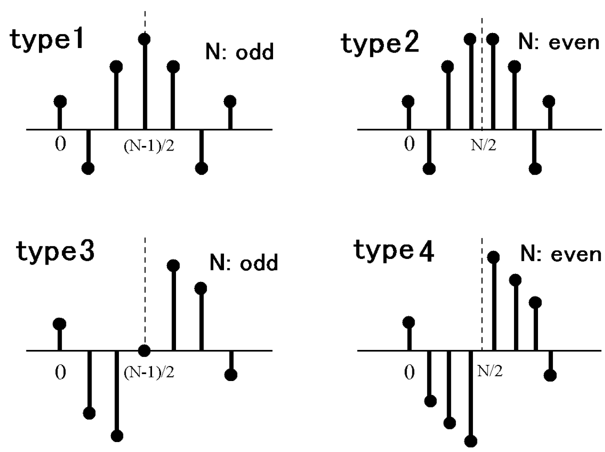


Figure 1: Four types of FIR filter

By the symmetrical property, the frequency response for type 1 is expressed as follows.

$$\begin{aligned} H(e^{j\omega}) &= \sum_{k=0}^{N-1} h_k e^{-jk\omega} && (3)\\ &= h_0 + h_1 e^{-j\omega} + h_2 e^{-2j\omega} \dots h_{N-1} e^{-(N-1)j\omega} && (4)\\ &= e^{-\frac{N-1}{2}j\omega} \sum_{k=0}^{(N-1)/2} h_k \cos k\omega && (5) \end{aligned}$$

Omitting the linear phase factor $e^{-\frac{N-1}{2}j\omega}$ in the equation (5), the frequency response of a symmetrical impulse response filter with N odd is given by

$$H(\omega) = \sum_{k=0}^{K} h_k \cos k\omega. \tag{6}$$

Here $K = (N-1)/2$ and this equation is called a magnitude response. Then the number of filter coefficients we consider is $K+1$.

Given a budget of total number of power-of-two terms M, a certain number of SP2 terms, m_k, is allocated to the k-th target discrete coefficient d_k. Then we denote the frequency response $H(\omega)$ as follows.

$$H(\omega) = \sum_{k=0}^{K} d_k \cos k\omega \tag{7}$$

The method to allocate SP2 terms is proposed, for example, by Lu [22], Ito et. al [10], [11].

We assume that the absolute value of each SP2 $d_k,\ k = 0, \dots, N-1$ is in the interval $[2^{-U}, 2^0]$ where U is a natural number. Then, with a given term allocation m_k, the discrete coefficients d_k in the equation (3) can be expressed as

$$d_k = \sum_{i=0}^{m_k} b_i^{(k)} 2^{-q_i^{(k)}}. \tag{8}$$

Since each SP2 d_k is consisted of m_k non-zero digits, the relation of $m_0, \dots, m_k$ and M is represented as the following equation:

$$\sum_{k=0}^{K} m_k = M. \tag{9}$$

Here, we have $b_i^{(k)} \in \{-1, 1\}$ and $1 \le q_i^{(k)} \le U,\ (0 \le i \le m_k,\ 0 \le k \le N-1)$.

Suppose a desired response $H_d(\omega)$ is given as follows

$$H_d(\omega) = \begin{cases} S, & \omega \in [0, \omega_p], \\ 0, & \omega \in [\omega_s, \pi]. \end{cases} \tag{10}$$

Then, the optimal problem to approximate $H(\omega)$ to $H_d(\omega)$ in a min-max sense can be written as

$$\min_{d_0,\dots,d_K} \max_{\omega\in\Omega} \left| H(\omega) - H_d(\omega) \right|. \tag{11}$$

If we introduce a new variable δ that corresponds to the L_∞-approximation error, it is easy to convert the above min-max problem to the following minimization problem, that is a semi-infinite programming problem with SP2 (DSILP):

$$\text{(DSILP)}\ \left|\begin{array}{lrcll} \text{min} & \delta & & & \\ \text{sub.to} & H(\omega)+\delta & \geq & H_d(\omega), & \omega\in\Omega, \\ & -H(\omega)+\delta & \geq & -H_d(\omega), & \omega\in\Omega. \end{array}\right. \tag{12}$$

3 3 Phase Method for Solving SILP

Now, we describe briefly how to solve the standard SILP

$$\text{(SILP)}\ \left|\begin{array}{ll} \text{min} & \boldsymbol{c}^T\boldsymbol{x} \\ \text{sub.to} & \boldsymbol{a}(t)^T\boldsymbol{x}\geq b(t), \quad t\in T \end{array}\right. \tag{13}$$

by means of 3 phase method. Here $\boldsymbol{c},\boldsymbol{a}(t)\in\mathbb{R}^n\ (t\in T), b(t)\in\mathbb{R}\ (t\in T),\ T\subseteq\mathbb{R}$ and $\boldsymbol{c}^T$ denotes the transposition of the vector $\boldsymbol{c}$. Also, $\mathbb{R}^n$ denotes the n-dimensional Euclidean space. $\boldsymbol{x}\in\mathbb{R}^n$ is a variable vector. The dual problem for (SILP) is, of course,

$$\text{(dualSILP)}\ \left|\begin{array}{lrcl} \text{min} & \sum_{t\in T} b(t)y(t) & & \\ \text{sub.to} & \sum_{t\in T}\boldsymbol{a}(t)y(t) & = & \boldsymbol{c} \\ & y(t) & \geq & 0, \quad t\in T, \end{array}\right. \tag{14}$$

Here, $y:T\to\mathbb{R}$. If there exists a $\boldsymbol{x}\in\mathbb{R}^n$ that satisfies $\boldsymbol{a}(t)^T\boldsymbol{x}>b(t)$ ($\forall t\in T$), then (SILP) is said to satisfy Slater's condition. When (SILP) satisfies (1) T is compact, (2) (dualSILP) is feasible and (3) (SILP) satisfies Slater's condition, then each (SILP) and (dualSILP) has an optimal solution and there exists no duality gap between them, see for example [4]. It is easily verified that (12) and its dual satisfy these three conditions when we assume $d_k\in\mathbb{R},\ k=0,\dots,N-1$. Hence, in the following we consider only the case.

By use of Carathéodory's theorem, we can show that there is an optimal solution for the dual problem (14) that has at most n positive dual variables [4]. Now we set

$$f(t):=\boldsymbol{a}(t)^T\boldsymbol{x}-b(t), \quad t\in T. \tag{15}$$

Then, $\boldsymbol{x}$ is an optimal solution for (SILP) and $y_1:=y(t_1)>0,\dots,y_p:=y(t_p)>0,\ (t_1,\dots,t_p\in T,\ p\leq n),\ y(t)=0\ (\forall t\in T,\ t\neq t_1,\dots.t_p)$ is an optimal solution for (dualSILP) if and only if the following equations hold.

$$\boldsymbol{a}(t_i)^T\boldsymbol{x}=b(t_i), \quad i=1,\dots,p, \tag{16}$$

$$\sum_{i=1}^{p}\boldsymbol{a}(t_i)y_i=\boldsymbol{c} \tag{17}$$

$$f(t)\ \text{ attains the minimum value at }\ t=t_1,\dots,t_p. \tag{18}$$

See [4]. If $\boldsymbol{a}(t)$ and $b(t)$ are differentiable and $t_1,\dots,t_p\in\operatorname{int}T$ then the last condition in the above is equivalent to

$$\frac{\partial}{\partial t}f(t_i)=0, \quad j=1,\dots,p, \tag{19}$$

where int T denotes the interior of T. Then we can solve (SILP) and (dualSILP) simultaneously by solving equations (16), (17) and (19) for $\boldsymbol{x},y_1,\dots,y_p,t_1,\dots,t_p$ if we know p. When some t_i is in the boundary of T we have to replace (19) by the Karush-Kuhn-Tucker condition for the i. Since this modification is a standard theory in mathematical programming, we will omit to describe it in detail.

If we know the value of p and know good initial feasible solutions for (SILP) and (dualSILP), we can solve equations (16), (17) and (19) by Newton/quasi-Newton method. To find p and good initial feasible solutions, we discretize T and replace (SILP)/(dualSILP) by a discretized primal/dual linear programming problem:

$$\text{(PLP)} \left| \begin{array}{ll} \min & \boldsymbol{c}^T\boldsymbol{x} \\ \text{sub.to} & \boldsymbol{a}(t_i)^T\boldsymbol{x} \geq b(t_i), \quad t = t_1, \ldots, t_q, \end{array} \right. \tag{20}$$

$$\text{(DLP)} \left| \begin{array}{llll} \min & \sum_{i=1}^{q} b(t_i)y_i & & \\ \text{sub.to} & \sum_{i=1}^{q} \boldsymbol{a}(t_i)y_i & = & \boldsymbol{c} \\ & y_i & \geq & 0, \quad i = 1, \ldots, q, \end{array} \right. \tag{21}$$

where $t_1, \ldots, t_q \in T$, and $q\,(\geq n)$ is an appropriate integer that (PLP)/(DLP) approximates (SILP)/(dualSILP). We call q as a discretizing parameter. In the 3 phase method, we first solve (DLP) and obtain its optimal dual solution $y_1^*, \ldots, y_q^*$ and an optimal primal solution $\boldsymbol{x}^*$ (phase 1).

We assume that $y^*(t_{i_1}), \ldots, y^*(t_{i_\ell})$ $(\ell \leq n)$ are positive. If (DLP) is not degenerate, then we have $\ell = n$ and this is the usual case. However, often is the case that $p < n$ for (dualSILP). This means that there exist several t_{i_j} for positive y_i such that t_{i_j} are close to t_i and $y_{i_j}^*$ are positive. Hence, we have to gather together those t_{i_j}'s to a single $t_{i'_j}$ and $y_{i_j}^*$'s to a single $y^*(t'_{i_j})$ respectively. By this way, we can obtain the value p and the approximate solution for (SILP) and (dulaSILP) (phase 2) and we can use Newton/quasi-Newton method to solve the equations (16), (17) and (19) (phase 3), see [4], [6].

4 An Algorithm for Solving (DSILP).

Our aim is to solve (DSILP). However, it is impossible to solve (DSILP) directly, because (DSILP) contains discrete SP2 variables d_k's. Hence, we replace d_k's by the continuous variables h_k's and we obtain a semi-infinite linear programming problem (we also call this problem as SILP for abbreviation) and we can use 3 phase method to solve it. Since SILP is a continuous optimization problem, an obtained optimal solution does not always satisfy the condition that each coefficient is an SP2. Hence, we have to combine SILP and a branch and bound (B & B) method.

When we solve SILP by using 3 phase method, some $\bar{h}_i$ may not be SP2 in an optimal solution. If we find such $\bar{h}_i$ that is not SP2, then we select one of them. And we generate two subproblems, which one has an additional constraint $h_j \leq \lfloor \bar{h}_j \rfloor$ and the other has an additional constraints $h_j \geq \lceil \bar{h}_j \rceil$. Here $\lfloor \bar{h}_j \rfloor$ is the maximum SP2 that is less than or equal to $\bar{h}_j$ instead of the maximum integer that is less than or equal to $\bar{h}_j$ and $\lceil \bar{h}_j \rceil$ is the minimum SP2 that is greater than or equal to $\bar{h}_j$. Here, we notice that these two subproblems are also SILP and can be solved by 3 phase method.

Now, we describe the 3 phase algorithm shortly in the following. Here, we set the discretized linear programming problem with discretizing parameter q is:

$$
(\mathrm{PLP})_q \left| \begin{array}{lrcll} \min & \delta & & & \\ \text{sub.to} & H(\omega_i)+\delta & \geq & H_d(\omega_i), & \omega_i \in \Omega, i=0,\dots,q' \\ & -H(\omega_i)+\delta & \geq & -H_d(\omega_i), & \omega_i \in \Omega, i=q'+1,\dots,q-1, \end{array} \right. \tag{22}
$$

where $H(\omega_i) = \sum_{k=0}^{K} h_k \cos k\omega_i,\ \ i = 0,\dots,q-1$ and $q' = q/2$ if q is even and $q' = (q-1)/2$ if q is odd. And

$$
(\mathrm{RSILP}) \left| \begin{array}{lrcll} \min & \delta & & & \\ \text{sub.to} & H(\omega)+\delta & \geq & H_d(\omega), & \omega \in \Omega, \\ & -H(\omega)+\delta & \geq & -H_d(\omega), & \omega \in \Omega. \end{array} \right. \tag{23}
$$

is a standard semi-infinite linear programming problem if $H(\omega) = \sum_{k=0}^{K} h_k \cos k\omega$. We note here, $H(\omega) = \sum_{k=0}^{K} d_k \cos k\omega$ in (DSILP) and (RSILP) is a continuous relaxation of (DSILP).

An algorithm for solving SILP

INPUT: N, ω_p, ω_s
OUTPUT: $\bar{\boldsymbol{h}} = (\bar{h}_0, \dots, \bar{h}_K), \bar{\delta}$
Phase 1:
Generate the discretized linear programming problem $(\mathrm{PLP})_q$ with discretizing parameter q.
Solve $(\mathrm{PLP})_q$ and obtain $\bar{\boldsymbol{h}} \in \mathbb{R}^{K+1}, \bar{\delta}, \bar{\boldsymbol{y}} \in \mathbb{R}^q$, $\omega_0, \dots, \omega_{q-1}$ where $\bar{\boldsymbol{y}}$ is an optimal dual variable vector for the dual problem of $(\mathrm{PLP})_q$ and $\omega_0, \dots, \omega_{q-1}$ are the frequencies that correspond to the constraints in $(\mathrm{PLP})_q$. Also, $\bar{\boldsymbol{h}}$ and $\bar{\delta}$ are an optimal primal variable vector and an optimal variable for $(\mathrm{PLP})_q$ respectively.
Phase 2:
while
there exist two positive $\bar{y}_i, \bar{y}_j$ which ω_i and ω_j corresponding to $\bar{y}_i$ and $\bar{y}_j$ are very close
do
$\bar{y}(\omega_i) \leftarrow \bar{y}(\omega_i) + \bar{y}(\omega_j)$,
$\bar{y}(\omega_j) \leftarrow 0$,
$\omega_i \leftarrow (\omega_i + \omega_j)/2$.
end of while
Reconstruct $\bar{\boldsymbol{y}}$ $(\in \mathbb{R}^p)$ by positive elements of $\bar{\boldsymbol{y}} \in \mathbb{R}^q$ and set $\omega_{i_1}, \dots, \omega_{i_p}$ for the reconstructed $\bar{\boldsymbol{y}}$.
Phase 3:
Solve (RSILP) by Newton/quasi-Newton method with using $(\bar{\boldsymbol{h}}, \bar{\delta}, \bar{\boldsymbol{y}}, \omega_{i_1}, \dots, \omega_{i_p})$ as the initial solution.
Output the solution of the Newton/quasi-Newton method.

Now, we describe the B & B method for solving DSILP in the following:

B & B procedure for DSILP:

INPUT: $N, \omega_p, \omega_s, S, m_0, \ldots, m_K$
OUTPUT: $h_0, \ldots, h_K, \delta,$
$k \leftarrow 0,$
$\bar{z} \leftarrow$ high value.
Generate DSILP (12), and set SILP $P(0)$ by relaxing the condition to be SP2 numbers.
$\mathcal{P} \leftarrow \{P(0)\}.$
while $\mathcal{P} \neq \emptyset$ **do**
  Select $P \in \mathcal{P}$.
  $\mathcal{P} \leftarrow \mathcal{P} \backslash \{P\}$.
  Solve SILP P by 3 Phase method.
  if $\delta < \bar{z}$
  then
    if the optimal solution $(\bar{\boldsymbol{h}}, \bar{\delta})$ of P is a solution with SP2 coefficients
    then
      $\bar{z} \leftarrow \bar{\delta},$
      $\boldsymbol{h}^* \leftarrow \bar{\boldsymbol{h}},$
    else
      select j that $\bar{h}_j$ is not an SP2, and generate $P(k+1)$ by adding a constraint
      $h_j \geq \lceil \bar{h}_j \rceil$ to P,
      generate $P(k+2)$ by adding a constraint
      $h_j \leq \lfloor \bar{h}_j \rfloor$ to P,
      $\mathcal{P} \leftarrow \mathcal{P} \cup \{P(k+1), P(k+2)\},$
      $k \leftarrow k+2.$
    end if
  end if
end while
Output $h_0^*, \ldots, h_K^*, \bar{z}$.

5 Numerical Experiments

We executed some computational experiments to certify the performance of the proposed filter design method.

We consider a low pass filter with the odd length and the symmetric characteristic with $S = 1$.

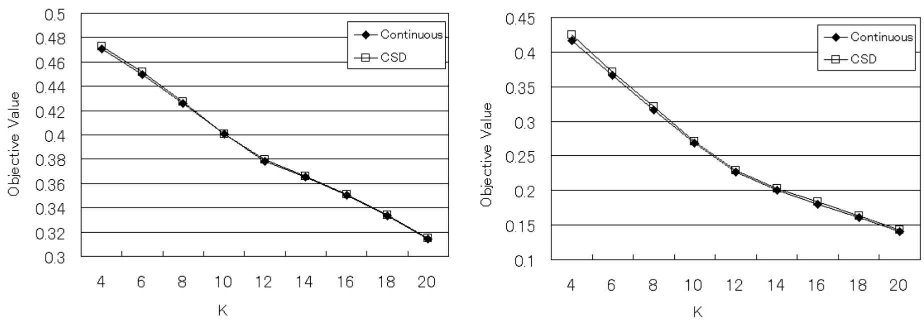
$$\Omega = [0, \omega_p] \cup [\omega_s, \pi], \qquad H_d(\omega) = \begin{cases} 1, & 0 \leq \omega \leq \omega_p, \\ 0, & \omega_s \leq \omega \leq \pi. \end{cases} \tag{24}$$

The approximation errors from the proposed scheme are calculated for the following three sets of parameters, (A), (B), (C). $N = 9, \ldots, 41$. Discretizing parameter q to generate the discretized linear programming problem is $4(K+1)$.

	M	ω_p	ω_s	U
(A)	$2(K+1)$	0.4π	0.41π	16,
(B)	$2(K+1)$	0.4π	0.43π	16,
(C)	$2(K+1)$	0.3π	0.35π	16,

We set each $m_k = 2$. The CPU is mobile Pentium III 650 MHz, and memory is 192 M bytes. We use glpk (Ver.4.4) [5] to obtain continuous solutions and to solve subproblems in Branch and Bound. The CPU time contains the execution time from the beginning to the end of obtaining the solution by our method.

In Figures 2 and 3, we show the objective value of our method and of continuous solutions



(1) approximation errors of case (A) (2) approximation errors of case (B)

Figure 2: Comparison of approximation errors

for $K = 4, 6, \ldots, 20$. The expression "Continuous" in Figures 2 and 3 means the optimal continuous solution and "CSD" means the CSD solution of our method, where CSD (canonical signed digit) is a well-known representation form for "canonical" binary numbers with ternary digit set $\{-1, 0, 1\}$ in the sense that any non-zero digit is always followed by 0.

In these figures, it was confirmed the objective values of our method are close to that of optimal solutions. In general, it is known that the transferband gets narrow, it is difficult to design FIR filter, but in case of (1), the objective value by our method is almost optimal in spite that transferband is narrow.

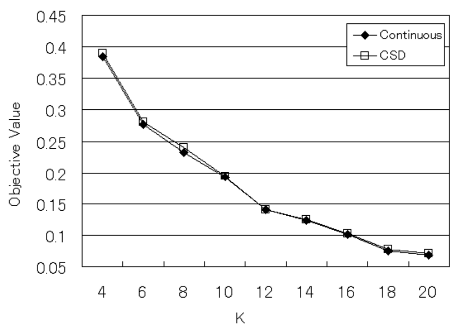
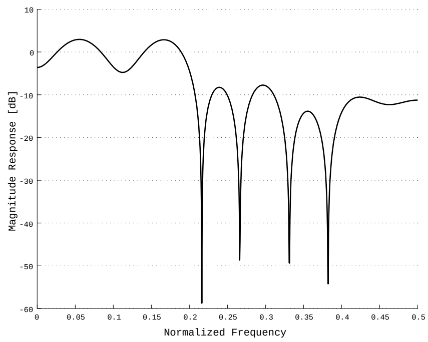
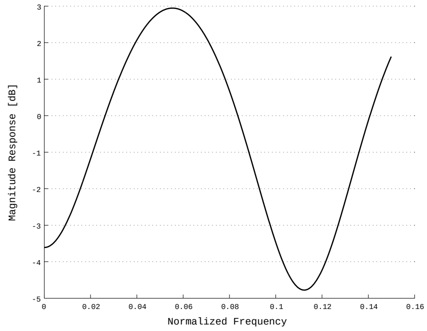


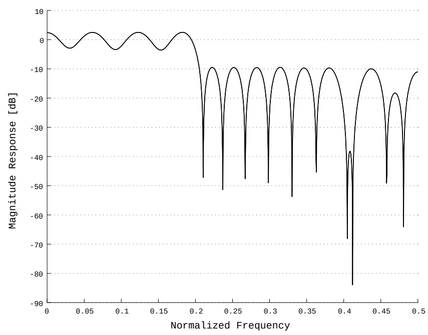
Figure 3: (3) approximation errors of case (C)



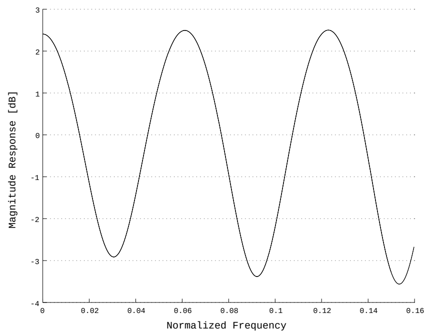
(1) case (A) of $K = 14$



(2) case (A) of $K = 14$



(3) case (A) of $K = 18$



(4) case (A) of $K = 18$

Figure 4: Magnitude response

In Figure 4, the magnitude responses (6) are shown for $\omega_p = 0.4\pi$, $\omega_s = 0.41\pi$ and Figure 4 shows the magnitude responses for $\omega_p = 0.3\pi$, $\omega_s = 0.35\pi$ and $\omega_p = 0.4\pi$, $\omega_s = 0.43\pi$. In Figure 4, it is observed that almost equi-ripple characteristic are obtained in both of two cases $K = 14$, and $K = 18$. Especially, in case of $K = 18$, it is shown that the magnitude response is almost equi-ripple.

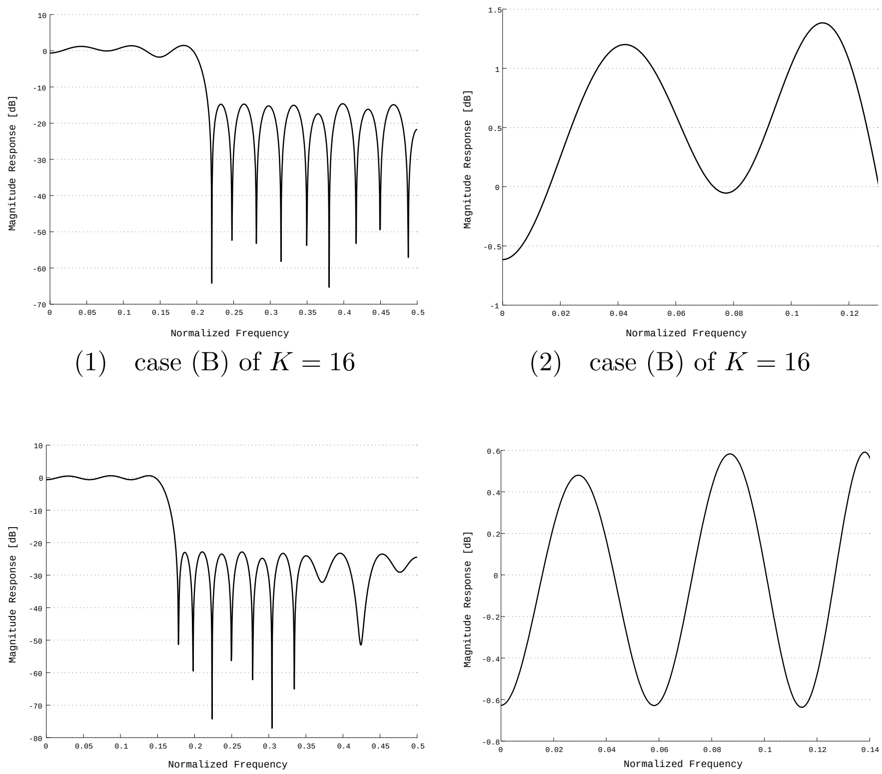
In Figure 5, these magnitude responses show that our method is efficient in not only stopband but also passband. In case of $K = 20$, it is shown that the magnitude response in passband is small and in case of $K = 16$, the magnitude response in stopband is almost equi-ripple.

In these results, it is shown that our method to design FIR filter is effective on obtaining of equi-ripple magnitude responses.

In Table 1, the comparison of the computational time is shown.
As much as a K becomes big, calculation time grows large. However it is observed that the computation time of $K = 20$ of (C) is about one hour,

6 Remarks

According to these experiments, though we changed q from $4(K+1)$, $5(K+1), \ldots, 10(K+1)$ on conditions $K = 4$ of (A), $K = 5$ of (B), $K = 6$ of (C), the objective values did not change. When discretizing parameter q is $4(K+1)$, the computational time



(1) case (B) of $K = 16$ (2) case (B) of $K = 16$

(3) case (C) of $K = 20$ (4) case (C) of $K = 20$

Figure 5: Magnitude response

is much faster than that of $q = 10(K + 1)$. It is said that for the equi-ripple FIR filter design problem, it is enough to set the discretizing parameter q to be $4(K + 1)$, and solve the discretized linear programming problem [17]. In the conventional technique, only the LP-solution for the discretized linear programming problem is used to solve B & B method, however, we cannot obtain the optimal solution for the discretizing parameter $q = 4(K+1)$. Hence, we recommend to use the SILP approach to solve DSILP since the Newton iterations occur only a few times for each $P(k)$ and the computational time is negligible for these iterations.

7 Conclusion

In this paper, we propose a new design method of FIR filters with SP2 coefficients. In this method, it is possible to obtain the optimal discrete coefficients. It is confirmed that the optimal coefficients of linear phase FIR filter with the SP2 coefficients could be designed fast with enough precisions through the computational experiments.

Table 1: Computational time (second)

K	A	B	C
4	1	9	5
6	2	23	7
8	27	44	13
10	82	219	122
12	290	309	155
14	656	495	442
16	2892	2804	1924
18	9340	7799	4190
20	14164	25456	3654

Acknowledgement

The authors would like to thank the referees for their many valuable comments and helpful advice.

References

[1] N.I. Cho and S.U. Lee, Optimal design of finite precision FIR filters using linear programming with reduced constraints, *IEEE Trans. Signal Process.* 46 (1998) 195–199.

[2] L. Faybusovich, Jordan algebras, symmetric cones and interior point methods, in *Technical Report, Department of Mathematics, University of Notre Dame*, Notre Dame, USA, 1995.

[3] T. Fujie and M. Kojima, Semidefinite relaxation for nonconvex programs, *Global Optim.* 10 (1997) 367–380.

[4] K. Glashoff and A-Å. Gustafson, *Linear Optimization and Approximation*, Appl. Math. Sci. 45, Springer-Verlag, New York, 1983.

[5] http://www.gnu.org/software/glpk/glpk.html

[6] M.A. Goberna and M. A. Lopéz, *Linear Semi-infinite Optimization*, Wiley Series Mathematical Methods in 2, John Wiley & Sons, Chichester, 1998.

[7] M.X. Goemans and D. Williamson, Improved approximation algorithms for maximum cut and satisfiability problem using semidefinite programming, *J. ACM.* 42 (1995) 1115–1145.

[8] F. Grenez, Design of FIR linear phase digital filters to minimize the statistical wordlength of the coefficients, *IEE J. Electron. Circuits Syst.* 1 (1977) 181–185.

[9] M. Grötschel, L.Lováz and A. Schrijver, *Geomatric Algorithms and Combinatorial Optimization*, Springer, New York, 1988.

[10] R. Ito and K. Suyama, A heuristic qpproach to sp2 term allocation for FIR filter based on least means square criteria, *Int. J. Innovative Computing, Information and Control* 1 (2005) 65–71.

[11] R Ito, T. Fujie, K. Suyama and R. Hirabayashi, New design methods of FIR filters with signed power of two coefficients based on a new linear programming relaxation with triangle inequalities, in *Proc. Int. Symp. Circuits Syst.*, IEEE, Phoenix, 2002, pp.I-813–816.

[12] R. Ito, T. Fujie, K. Suyama and R. Hirabayashi, A new heuristic signed-power of two term allocation approach for desgning of FIR filters, in *Proc. Int. Symp. Circuits Syst.*, IEEE, Bangkok, 2003, Vol. IV pp.285–288.

[13] S. Kim and M. Kojima, Second order cone programming relaxation of nonconvex quadratic optimization problems, *Optim. Methods Softw.* 15: 3-4 (2001) 201–224.

[14] Y.W. Kim, Y.M. Yang, J.T.Yoo and S.W. Kim, Low-power digital Filtering using approximate processing with variable canonic signed digit coefficients, in *Proc. Int. Symp. Circuits Syst.*, IEEE, 2000, pp. 337–340.

[15] D.M. Kodek, Design of optimal finite wordlength FIR digital filters using integer programming techniques, *IEEE Trans. Acoustics, Speech, Signal Processing*, ASSP-28 (1980) 304–308.

[16] D.M. Kodek and K. Steiglitz, Finite-length word-length tradeoffs in FIR digital filter design, *IEEE Trans. Acoustics, Speech, Signal Processing*, ASSP-28 (1980) 739–744.

[17] Y.C. Lim, Design of discrete-coefficient-value linear phase FIR filters with optimum normalized peak ripple magnitude, *IEEE Trans. Circuits Syst.* 37 (1990)1480–1486.

[18] Y.C. Lim and A.G. Constantinides, Linear phase FIR digital filter without multipliers, in *Proc. Int. Symp. Circuits Syst.*, Tokyo, Japan, July 1983, pp. 185-188.

[19] Y.C. Lim and S.R. Parker, FIR filter design over a discrete power-of-two coefficient space, in *Proc. IEEE Int. Symp. Circuits Syst.* Vol. II, 1983, pp. 583–591.

[20] Y.C. Lim and S.R. Parker, Discrete coefficient FIR digital filter design based upon lms criteria, *IEEE Trans. Circuits Syst.* CAS-30 (1983) 723–739.

[21] Y.C. Lim, J. Lee, C.K. Chen and R. Yang, A weighted least squares algorithm for quasi-equiripple FIR and IIR digital filter design, *IEEE Trans. Signal Process* 40 (1992) 551–558.

[22] Y.C. Lim, R. Yang, D. Li and J. Song, Signed power-of-two (SPT) term allocation scheme for the design of digital filters, *IEEE Trans. Circuits Syst II.* 46 (1999) 577–584.

[23] W.-S. Lu, Design of FIR Filters with discrete coefficients: A semidefinite programming relaxation approach, in *Proc. IEEE Int. Symp. Circuits Syst.* Vol. II, 2001, pp. 297–300.

[24] T.W. Parks and J.H. McClellan, Chebyshev approximation for nonrecursive digital filters with linear phase, *IEEE Trans. Circuit Theory* CT-19 (1972) 89–194.

[25] E. Polak, *Optimization: Algorithms and Consistent Approximations*, Appl. Math. Sci. 124, Springer-Verlag, New York, 1997.

[26] P. Siohan and C. Roche, Optimal design of 1-D and 2-D FIR multiplierless filters, in *Proc. Int. Conference on Acoustics, Speech, and Signal Processing, ICASSP '91*, 1991, pp. 2877–2880.

[27] M. Suk and S.K. Mitra, Computer-aided design of digital filters with finite wordlength, *IEEE Trans. Audio Electroacoust.* AU-20 (1972) 356–366.

[28] L. Vandenberghe and S. Boyd, Semidefinite programming, *SIAM Rev.* 38 (1996) 49–95.

[29] Y. Ye, Approximating quadratic programming with bound and quadratic constraints, *Math. Program.* 84 (1999) 219–226.

[30] J.T. Yli-Kaakinen and T.A. Saramaki, An Algorithm for the design of multiplierless approximately linear-phase lattice wave digital filters, in *Proc. IEEE Int. Symp. Circuits Syst.* Vol. II, 2000, pp. 77–80

[31] X. Zhang and H. Iwakura, Design of FIR linear phase filters with discrete coefficients using Hopfield neural networks, *IEE Electronics Letters* 13 (1994) 1039–1040.

[32] Q. Zhao and Y. Tadokoro, A simple design of FIR filters with power-of-two coefficients, *IEEE Trans. Circuits Syst.* 35 (1988) 566–570.

Manuscript received 28 December 2005
revised 20 October 2006
accepted for publication 23 October 2006

Rika Ito
Department of Information Science, Toho University, 2-2-1, Miyama, Funabashi-shi, Chiba, Japan
E-mail address: `Ritonet@aol.com`

Ryuichi Hirabayashi
Faculty of Management, Mejiro University, 4-31-1, Nakaochiai, Shinjuku-ku, Tokyo, Japan
E-mail address: `hira@mejiro.ac.jp`