# Design of FIR Filters with Discrete Coefficients via Sphere Relaxation

Wu-Sheng Lu
Dept. of Electrical and Computer Engineering
University of Victoria
Victoria, BC, Canada V8W 3P6
Email: wslu@ece.uvic.ca

***Abstract*— A method for designing FIR digital filters, with each coefficient a sum of signed power-of-2 terms, by sphere relaxation is proposed. To justify the design methodology, we present an analysis showing that the globally optimal design always lies in a vicinity of the optimal FIR filter with continuous coefficients. The design problem at hand is then addressed using a new relaxation method in which the constraints characterizing the binary nature of the design variables are relaxed to a single sphere type constraint on the corresponding continuous variables. This yields a simple nonconvex continuous optimization problem whose solution can be calculated considerably faster than previously reported relaxation methods. Design examples are presented to demonstrate that the proposed algorithm offers near-optimal designs with small fraction of design complexity relative to that required by the existing methods.**

## I. Introduction

Design of digital filters with discrete coefficients, especially with sum-of-power-of-2 coefficients, has been a subject of study for more than two decades because these filters admit fast implementation, see for example [1]–[15] and the references cited therein. The major technical difficulty encountered here is the design complexity because the design problem is essentially an integer programming problem that is known to have exponential complexity as the filter length grows. A recent research trend in this area seems to be the use of relaxation techniques that reformulate the problem as an optimization problem of polynomial-time complexity, at the cost of moderate performance degradation. It is in this sense the term "relaxation" is used to emphasize a reformulation process generating a problem which can be solved efficiently and realiably and whose solution offers satisfactory suboptimal design. Among other things, the semidefinite programming relaxation (SDPR) [12] and sequential quadratic programming relaxation (SQPR) [15] can yield good designs, but SQPR turns out to be considerably faster than SDPR.

The objectives of this paper are twofold. First, we present an analysis which shows that the globally optimal design of the problem at hand always lies in a vicinity of the optimal FIR filter with continuous coefficients. This justifies the design approach to be taken in this paper. Second, the design problem is addressed using a new relaxation method in which the constraints characterizing the binary nature of the design variables are relaxed to a *single* sphere type constraint on the corresponding continuous variables. This yields a simple nonconvex continuous optimization problem whose solution can be calculated considerably faster than previously reported relaxation methods. Design examples are presented to demonstrate that the proposed algorithm offers near-optimal designs with small fraction of design complexity relative to that required by the existing methods.

## II. Where is the Global Solution?

### A. An Analysis

To simplify the notation, we only consider linear phase FIR filters of odd length $n_1$ whose amplitude response can be expressed as

$$A(\omega) = \sum_{k=0}^{n} a_k \cos k\omega, \qquad n = (n_1 - 1)/2 \tag{1}$$

The results obtained here can be extended to nonlinear phase FIR filters of arbitrary length in a straightforward manner. Following [7][8], we seek to find a linear phase FIR filter of odd length $n_1$ with each coefficient $a_k$ a sum of signed power of 2 (SP2) terms such that $A(\omega)$ in (1) best approximates a desired amplitude response $A_d(\omega)$ in the least-squares sense. It has been argued that, with a fixed total number of signed SP2 terms, significant advantage may be gained if the number of SP2 terms for each $a_k$ is individully allocated and a strategy for doing so has been proposed in [8]. In the rest of the paper, it is assumed that the bit budget is fixed and the exercise of term allocation is done in accordance with [8].

Now let $\{\hat{a}_k,\ 0 \le k \le n\}$ be the set of *continuous* coefficients of the optimal FIR filter and let

$$\hat{A}(\omega) = \sum_{k=0}^{n} \hat{a}_k \cos k\omega \tag{2a}$$

that minizes the $L_2$ error

$$\hat{\varepsilon} = \left\{ \int_{-\pi}^{\pi} [\hat{A}(\omega) - A_d(\omega)]^2 d\omega \right\}^{1/2} \tag{2b}$$

We define

$$\varepsilon = \left\{ \int_{-\pi}^{\pi} [A(\omega) - A_d(\omega)]^2 d\omega \right\}^{1/2} \tag{3}$$

and compute

$$\int_{-\pi}^{\pi}[A(\omega)-\hat{A}(\omega)]\cos k\omega d\omega = \begin{cases} 2\pi(a_0-\hat{a}_0) & \text{for } k=0 \\ \pi(a_k-\hat{a}_k) & \text{for } k\neq 0 \end{cases}$$

Let $c_k = 2\pi$ for $k = 0$ and $c_k = \pi$ for $k \neq 0$, and apply the Cauchy-Schwartz inequality, we can write

$$\begin{aligned} |a_k-\hat{a}_k| &= \frac{1}{c_k}\left|\int_{-\pi}^{\pi}[A(\omega)-\hat{A}(\omega)]\cos k\omega d\omega\right| \\ &\le \frac{1}{c_k}\left\{\left|\int_{-\pi}^{\pi}[A(\omega)-A_d(\omega)]\cos k\omega d\omega\right| \right. \\ &\quad \left. +\left|\int_{-\pi}^{\pi}[\hat{A}(\omega)-A_d(\omega)]\cos k\omega d\omega\right|\right\} \\ &\le \frac{1}{\sqrt{\pi}}(\varepsilon+\hat{\varepsilon}) \end{aligned} \tag{4}$$

Suppose the coefficient set $\{a_k^*,\ 0 \le k \le n\}$ corresponds to the global optimal FIR filter with SP2 coefficients, then (4) implies that

$$|a_k^*-\hat{a}_k| \le \frac{1}{\sqrt{\pi}}(\varepsilon^*+\hat{\varepsilon}) \qquad \text{for } 0 \le k \le n \tag{5}$$

where

$$\varepsilon^* = \left\{\int_{-\pi}^{\pi}[\sum_{k=0}^{n} a_k^*\cos k\omega - A_d(\omega)]^2 d\omega\right\}^{1/2} \tag{6}$$

Eq. (5) indicates that all the optimal discrete coefficients reside within a strip centered along the trajectory passing through the optimal continuous coefficients. In conjunction with (2b) and (6), Eq. (5) also indicates that, for a reasonable filter length and bit budget, this strip shall be narrow because both $L_2$-errors $\varepsilon^*$ and $\hat{\varepsilon}$ are small in such a case. An illustration of this situation is given in Fig. 1. Based on the above analysis, we formulate the design problem at hand as a $\{-1,\ 1\}$ quadratic programming problem in the next section.

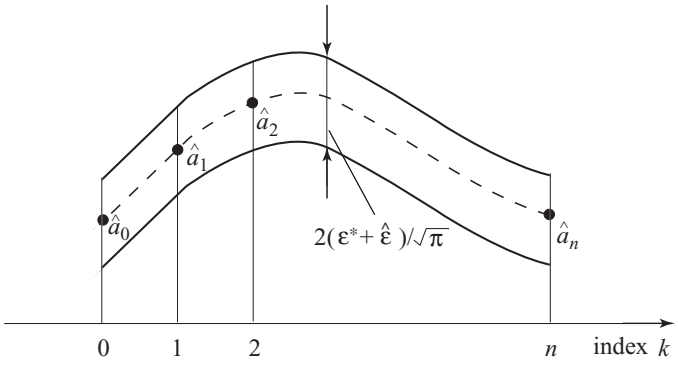


Fig. 1. The optimal discrete coefficients shall reside within a narrow strip centered along the optimal continuous coefficients.

### *B. Problem Formulation*

Given each optimal continuous coefficient $\hat{a}_k$ and the number (say $\alpha_k$) of SP2 terms allocated to the $k$th discrete coefficient, we use $\underline{a}_k$ and $\bar{a}_k$ to denote the largest lower bound and smallest upper bound of $\hat{a}_k$, respectively, where $\underline{a}_k$ and $\bar{a}_k$ each has $\alpha_k$ SP2 terms and are inside the strip shown in Fig. 1. In general the strip may well include more than one layer of lower bounds $\underline{a}_k$, $\underline{\underline{a}}_k$, etc., and upper bounds $\bar{a}_k$, $\bar{\bar{a}}_k$, etc. On the other hand, since the strip is usually narrow, the number of bound layers on each side of $\{\hat{a}_k\}$ is supposed to be small. Fig. 2 illustrates a possible situation for coefficient $\hat{a}_k$.

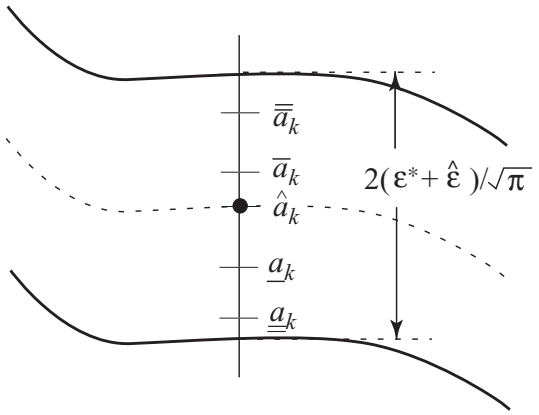


Fig. 2. First and second largest lower SP2 bounds and first and second smallest upper SP2 bounds of $\hat{a}_k$ happen to be inside the strip.

For the sake of notation simplicity, in the rest of the paper we only consider the one-layer case. The approach taken here can nevertheless be generalized to cover multi-layer cases in a straightforward manner. Note that the candidate discrete coefficients can now be expressed as

$$a_k = a_{mk} + x_k\delta_k \tag{7}$$

for $k = 0,\ 1,\ \ldots,\ n$, where

$$\begin{aligned} a_{mk} &= (\bar{a}_k+\underline{a}_k)/2 \\ \delta_k &= (\bar{a}_k-\underline{a}_k)/2 \\ x_k &\in \{-1,\ 1\}. \end{aligned}$$

Consequently, the filter's frequency response can be written in terms of variable $\boldsymbol{x} = [x_0 \ \ x_1 \ \ \ldots \ \ x_n]^T$ as

$$\begin{aligned} A(\omega) &= A_m(\omega) + \boldsymbol{x}^T\boldsymbol{c}(\omega) \\ A_m(\omega) &= \sum_{k=0}^{n} a_{mk}\cos k\omega \\ \boldsymbol{c}(\omega) &= [\delta_0 \ \ \delta_1\cos\omega \ \ \ldots \ \ \delta_n\cos n\omega]^T \end{aligned} \tag{8}$$

and the least-squares design problem can be formulated as the constrained optimization problem

$$\begin{aligned} \text{minimize} \quad & \int_{-\pi}^{\pi} W(\omega)[A_m(\omega)+\boldsymbol{x}^T\boldsymbol{c}(\omega)-A_d(\omega)]^2 d\omega \\ \text{subject to:} \quad & x_k \in \{-1,1\} \quad k=0,\ 1,\ \ldots,\ n \end{aligned}$$

or, more explicitly,

$$\text{minimize} \quad f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} \tag{9a}$$

$$\text{subject to:} \quad x_k \in \{-1,\ 1\} \quad k=0,\ 1,\ \ldots,\ n \tag{9b}$$

where

$$\boldsymbol{Q} = \int_{-\pi}^{\pi} W(\omega)\boldsymbol{c}(\omega)\boldsymbol{c}^T(\omega)d\omega$$

$$\boldsymbol{q} = 2\int_{-\pi}^{\pi} W(\omega)[A_m(\omega) - A_d(\omega)]\boldsymbol{c}(\omega)d\omega$$

## III. A Suboptimal Solution via Sphere Relaxation

### A. A Basic Solution Formula

The problem in (9) is an integer programming (IP) problem whose complexity grows exponentially with its size. The problem was relaxed to an SDP problem in [12] and a sequential QP problem in [15], both providing satisfactory suboptimal solutions but still with considerable complexity. In what follows, we propose a new and more efficient relaxation of problem (9) and describe an algorithm to solve the relaxed problem

The basis of the proposed relaxation is the following chain of implications:

$$x_k \in \{-1,\ 1\} \Longleftrightarrow x_k^2 = 1 \Longrightarrow \sum_{k=0}^{n} x_k^2 = n+1$$

This suggests the relaxed problem

$$\text{minimize} \quad f(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} \tag{10a}$$

$$\text{subject to:} \quad \|\boldsymbol{x}\|_2^2 = n+1 \tag{10b}$$

The critical differences between problems (9) and (10) are (i) the feasible region characterized by (10b) includes the feasible region described by (9b). Consequently, if $\boldsymbol{x}^{**}$ and $\boldsymbol{x}^*$ denote the solutions of (9) and (10), respectively, then $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}^{**})$, but $\boldsymbol{x}^*$ does not necessarily satisfy (9b); (ii) while (9) is an IP problem, (10) is a constrained problem with *continuous variables* with a *single* equality constraint that $\boldsymbol{x}^*$ must lie on the sphere centered at the origin with radius $\sqrt{n+1}$ in space $R^{n+1}$. For these reasons, the problem (10) is referred to as a *sphere relaxation* (SR) of problem (9).

The feasible region defined by (10b) is obviously not convex. The problem in (10) can however be solved efficiently using a Lagrange-multiplier approach as described below.

The Lagrangian of problem (10) is given by

$$L(\boldsymbol{x}, \lambda) = \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q}^T\boldsymbol{x} + \lambda(\boldsymbol{x}^T\boldsymbol{x} - n - 1)$$

hence the first-order necessary conditions for $\boldsymbol{x}$ to be a minimizer of (10) are given by

$$2\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{q} + 2\lambda\boldsymbol{x} = \boldsymbol{0} \tag{11a}$$

$$\boldsymbol{x}^T\boldsymbol{x} = n+1 \tag{11b}$$

From (11a) we have

$$\boldsymbol{x} = -\frac{1}{2}(\lambda\boldsymbol{I} + \boldsymbol{Q})^{-1}\boldsymbol{q} \tag{12}$$

Since $\boldsymbol{Q}$ is positive definite and symmetric, we can write $\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}^T$ with $\boldsymbol{U}$ orthogonal and $\boldsymbol{\Sigma} = \text{diag}\{\sigma_1,\ \sigma_2,\ \ldots,\ \sigma_{n+1}\}$, so (12) becomes

$$\boldsymbol{x} = -\frac{1}{2}\boldsymbol{U}(\lambda\boldsymbol{I} + \boldsymbol{\Sigma})^{-1}\tilde{\boldsymbol{q}},\ \tilde{\boldsymbol{q}} = \boldsymbol{U}^T\boldsymbol{q} \tag{13}$$

which in turn leads (11b) to

$$g(\lambda) \equiv \sum_{k=1}^{n+1} \frac{\tilde{q}_k^2}{(\lambda + \sigma_k)^2} = 4(n+1) \tag{14}$$

where $\tilde{q}_k$ is the $k$th component of $\tilde{\boldsymbol{q}}$. The function $g(\lambda)$ on the left-hand side of (14) monotonically decreases with $\lambda$ and is well defined on interval $\left(-\underline{\sigma},\ \dfrac{\|\tilde{\boldsymbol{q}}\|}{1.5\sqrt{n+1}} - \underline{\sigma}\right)$ where $\underline{\sigma}$ is the smallest eigenvalue of $\boldsymbol{Q}$, say $\sigma_l$, such that $\tilde{q}_l \neq 0$. It can be readily verified that on the above interval the value of $g(\lambda)$ decreases from larger than $4(n+1)$ to less than $4(n+1)$, hence a unique $\lambda^*$ that satisfies (14) exists on the interval and can be readily identified using for example a bisection method. Having obtained $\lambda^*$, the solutin of (10) can be found using (12) as

$$\boldsymbol{x}^* = -\frac{1}{2}\boldsymbol{U}(\lambda^*\boldsymbol{I} + \boldsymbol{\Sigma})^{-1}\tilde{\boldsymbol{q}} \tag{15}$$

Notice that the vector $(\lambda^*\boldsymbol{I} + \boldsymbol{\Sigma})^{-1}\tilde{\boldsymbol{q}}$ in (15) can be computed without performing a matrix inverse. As a matter of fact, its $k$th component is equal to $\tilde{\boldsymbol{q}}_k/(\lambda^* + \sigma_k)$.

### B. A Sequential Solution Process

A straightforward treatment of the solution $\boldsymbol{x}^*$ in (15) is to use its signs, i.e., $\boldsymbol{x}_1 = \text{sign}(\boldsymbol{x}^*)$ as a suboptimal solution of (9).

There are several ways to generate improved solutions. When problem (10) is solved, one may verify the solution components and round those, whose magnitudes exceed a prescribed threshold, to 1 or $-1$ depending on their signs. The identified components are then substituted into (10) which yields a similar constrained problem with reduced size. The process continues until all components are identified. An alternative way is to round the component of the solution of (10) with the largest magnitude to 1 or $-1$ depending on its sign, and this component is then fixed in problem (10) to generate a similar constrained problem whose size is reduced by one. The process continues until all components are identified. The solution obtained using one of these approaches shall be denoted as $\boldsymbol{x}_b^*$. Below the second solution method will be referred to as the sequential SR (SSR) algorithm.

### C. Solution Enhancement by Low-Bit Descent Search

Having obtained $\boldsymbol{x}_b^*$, one may carry out a low-bit descent search for a further improved solution. Here, the one-bit descent search refers to a process of evaluation, comparison, and determination of the optimal one-component sign switch among all possible such switches of $\boldsymbol{x}_b^*$. By simple algebra, it can be readily verified that the one-bit descent search can be accomplished by evaluating $\boldsymbol{\gamma} = \boldsymbol{x}_b^* \odot \boldsymbol{v}$ ($\odot$ denotes component-wise multiplication), where $\boldsymbol{v} = \hat{\boldsymbol{Q}}\boldsymbol{x}_b^* + \boldsymbol{q}$ with $\hat{\boldsymbol{Q}}$ being the modified $\boldsymbol{Q}$ by setting its diagonal to zero, then identifying index $i^*$ with which the associated component of $\boldsymbol{\gamma}$ assumes the largest value. An improved solution is obtained by switching the sign of the $i^*$-th component of $\boldsymbol{x}_b^*$.

Similarly, the two-bit descent search finds two components of $\boldsymbol{x}_b^*$ with indices $i^*$ and $j^*$ such that by switching the signs

of these two components, $f(\boldsymbol{x}_b^*)$ is reduced most among all possible two-component sign switches. The two-bit descent search can be carried out as follows: (i) compute matrix $\boldsymbol{P} = \boldsymbol{\gamma}\boldsymbol{e}^T + \boldsymbol{e}\boldsymbol{\gamma}^T - 2\boldsymbol{Q} \odot (\boldsymbol{x}_b^* \boldsymbol{x}_b^{*T})$ where $\boldsymbol{e}$ is the all-one vector; (ii) identify index pair $(i^*, j^*)$ where the component of $\boldsymbol{P}$ reaches the maximum value, and (iii) switch the signs of the $i^*$-th and $j^*$-th components of $\boldsymbol{x}_b^*$.

It is obvious that the two-bit search does not necessarily perform better than the one-bit search and vice versa. Since both search methods are of low complexity even for large filter length, it is often feasible to perform both searches to obtain the best solution.

## IV. Design Examples

The SSR-based algorithm enhanced with both one-bit and two-bit descent search was applied to design a variety of linear phase FIR filters with SP2 coefficients. In what follows we evaluate a total of eight designs of lowpass FIR filters with normalized passband edge $\omega_p = 0.4$ and stopband edge $\omega_a = 0.5$ of length $n_1 = 19 + 8i$ for $i = 0,\ 1,\ \ldots,\ 7$. The weighting function was set to $W(\omega) \equiv 1$ for both passband and stopband, and $W(\omega) \equiv 0$ elsewhere. In all designs, the filter performance was evaluated by the $L_2$-error $\varepsilon$ (see Eq. (3)) and the design complexity was evaluated by the CPU time consumed by a MATLAB implementation of the algorithm on a 3.4 GHz Pentium 4 PC. The designs were compared with the suboptimal solutions obtained using sequential QP-relaxation (SQPR), the optimal solution (with SP2 coefficients) using IP, and the optimal solution with continuous coefficients (WLS). The results obtained are summarized in Table 1, where $n_1$ denotes filter length, $M$ denotes the total number of SP2 terms for the filter's first $(n_1 + 1)/2$ coefficients, and $-L$ is the most negative power of 2 allowed. For SQPR-, SSR-, and IP-based designs, for each $(n_1,\ M,\ L)$ the two numerical values given in the table are the $L_2$-error (in the upper row) and the CPU time (in the lower row, in seconds), respectively. In the case of IP-based designs, no results were obtained for filter length $n_1 = 67$ and 75 due to the exceedingly high complexity required. In the case of WLS designs, the CPU time are not included because the amount of CPU time was found neglegiable relative to the other design methods.

From the design results in Table 1, it is observed that both the SQPR- and SSR-based algorithms are able to offer near-optimal designs with much less computational complexity relative to that required by the IP-based algorithm. It is also observed that the proposed SSR-based algorithm is considerably more efficient than the SQPR-based algorithm needing only a small fraction of the CPU time required by the SQPR method.

TABLE I

Performance and efficiency comparisons of various design methods

| $n_1$ | $M/L$ | SQPR | SSR | IP | WLS |
|---|---|---|---|---|---|
| 19 | 22/5 | 0.0040 | 0.0040 | 0.0040 | 0.0027 |
| | | 0.110 | 0.002 | 0.050 | |
| 27 | 30/5 | 0.0021 | 0.0021 | 0.0021 | 0.0005 |
| | | 0.170 | 0.004 | 7.594 | |
| 35 | 38/8 | 1.39e−4 | 1.39e−4 | 1.39e−4 | 0.90e−4 |
| | | 0.220 | 0.007 | 11.016 | |
| 43 | 46/8 | 8.69e−5 | 5.46ee−5 | 5.46e−5 | 2.07e−5 |
| | | 0.270 | 0.010 | 54.406 | |
| 51 | 54/8 | 4.73e−5 | 4.73e−5 | 4.31e−5 | 0.61e−5 |
| | | 0.300 | 0.012 | 751.609 | |
| 59 | 62/10 | 3.89e−5 | 3.89e−5 | 2.92e−5 | 0.17e−5 |
| | | 0.340 | 0.014 | 15525 | |
| 67 | 74/10 | 1.08e−5 | 1.08ee−5 | — | 0.40e−6 |
| | | 0.400 | 0.016 | — | |
| 75 | 84/10 | 5.67e−6 | 5.67e−6 | — | 0.81e−7 |
| | | 0.440 | 0.023 | — | |

## Acknowledgement

The author is grateful to the Natural Sciences and Engineering Research Council of Canada for supporting this work.

## References

[1] Y. C. Lim and S. R. Parker, "FIR filter design over a disrete power-of-two coefficient space," *IEEE Trans. ASSP*, vol. 31, pp. 588–591, June 1983.

[2] P. P. Vaidyanathan, "Efficient and multiplierless design of FIR filters with very sharp cutoff via maximally flat building blocks," *IEEE Trans. CAS*, vol. 32, pp. 236–244, March 1985.

[3] S. C. Pei and S. B. Jaw, "Efficient design of 2-D multiplierless FIR filters by transformation," *IEEE Trans. CAS*, vol. 34, pp. 436–438, April 1987.

[4] P. Siohan and A. Benslimane, "Finite precision design of optimal linear phase 2-D FIR digital filters," *IEEE Trans. CAS*, vol. 36, pp. 11–22, Jan. 1989.

[5] M. Benvenuto and M. Marchesi, "Digital filters design by simulated annealing," *IEEE Trans. CAS*, vol. 36, pp. 459–460,March 1989.

[6] Y. C. Lim, "Design of discrete-coefficient-value linear phase FIR filters with optimum normalized peak ripple magnitude," *IEEE Trans. CAS*, vol. 37, pp. 1480-1486, Dec. 1990.

[7] D. Li, J. Song and Y. C. Lim, "A polynomial-time algorithm for designing digital filters with power-of-two coefficients," *Proc. ISCAS'93*, pp. 84-87, May 1993.

[8] Y. C. Lim, R. Yang, D. Li, and J. Song, "Signed power-of-two (SPT) term allocation scheme for the design of digital filters," *Proc. ISCAS*, Monterey, CA., June 1998.

[9] F. Ashrafzadeh, B. Nowrouzian, and A. T. G. Fuller, "A novel modified branch-and-bound technique for discrete optimization over canonical signed-digit number space," *Proc. ISCAS 1998*, vol. V, pp. 391–394, Monterey, CA., June 1998.

[10] J. T. Yli-Kaakinen and T. A. Saramäki, "An algorithm for the design of multiplierless approximately linear-phase lattice wave digital filters," *Proc. ISCAS*, vol. II, pp. 77-80, Geneva, June 2000.

[11] Y. C. Lim and Y. J. Yu, "A successive reoptimization approach for the design of discrete coefficient perfect reconstruction lattice filter bank", *Proc. ISCAS*, vol. II, pp. 69–72, Geneva, June 2000.

[12] W.-S. Lu, "Design of FIR filters with discrete coefficients: A semidefinite programming relaxation approach," *Proc. ISCAS*, vol. 2, pp. 297–300, Sydney, Australia, May 2001.

[13] T. Çiloğlu, "An efficient local search method guided by gradient information for discrete coefficient FIR filter design," *Signal Processing*, vol. 82, pp. 1337–1350, 2002.

[14] L. Cen and Y. Lian, "A hybrid GA for the design multiplication-free frequency response masking filters," *Proc. ISCAS 2005*, pp. 520–523, Kobe, May 2005.

[15] W.-S. Lu, "Design of FIR digital filters with discrete coefficients via convex relaxation," *Proc. ISCAS 2005*, pp. 1831–1834, Kobe, May 2005.