Transactions Briefs

Discrete Optimization of Digital Filters Using Gaussian Adaptation and Quadratic Function Minimization

G. KJELLSTRÖM, L. TAXÉN, AND P. O. LINDBERG

Abstract —In this paper, a new strategy for discrete optimization of a function $F(x)$ is presented. Let A be the region in the n-dimensional parameter space, where $F(x)$ is less than some constant. First, A is located and characterized by a Gaussian search process, called *Gaussian adaptation*. This makes it possible to approximate the behavior of $F(x)$ over A by a quadratic function $Q(x)$. $Q(x)$ is then optimized for the N best discrete solutions using a branch and bound technique. Finally, these points are evaluated for the best $F(x)$ points.

By various digital filter examples it will be demonstrated that the new method is more capable of finding good solutions than methods presented so far.

I. Introduction

Discrete optimization appears in many different areas, such as the design of digital filters, the design of switched capacitor filters, and the design of analog filters, where only certain standard component values are allowed. A discrete variable may also be the number of wagons in a train or the number of ships used for transportation. Some authors even think of discrete optimization as a model for the evolution of biological systems [5].

In this paper, however, we will restrict ourselves to the problem of designing digital filters with quantized coefficients. Several methods [1]–[4], [6], [7], [9]–[15] for solving this problem exist. One way is to do a preoptimization of an associated continuous optimization problem before the assault on the discrete problem is made. In doing so, any well-established method for continuous optimization of a function $F(x)$ of designable parameters x can be used. (For a proper definition of $F(x)$, see Section II.)

One such approach is to utilize the eigenvectors of the Hessian (a quadratic function approximation of the local behavior of $F(x)$ at the minimum) to generate linear search directions for the optimal discrete points [2]. In this case, $F(x)$ should also be a differentiable function. Another very similar method is to generate the linear search directions at random from the Gaussian distribution adapted with maximum hitting probability to the set of acceptability [1]. A third method is the procedure based on the pattern search of Hooke and Jeeves [3].

Another interesting approach has been given by Jain [7]. In this case, the total number of bits different from zero in the coefficients of the digital filter (according to the CSD code) is taken as the criterion, or cost function $C(x)$. This criterion is derived from four important cost factors, which have to be considered by designers of digital filters:

- chip area,
- power dissipation,
- signal processing time,
- design time.

In this case, $F(x)$ may be defined as being smaller than some constant value when all requirements on the digital filter are fulfilled. All points in A are then "acceptable," i.e., they correspond to a filter meeting all requirements. However, their corresponding cost function $C(x)$ has a very irregular structure. There may even exist many points having the same optimum cost value and all spread out like isolated islands in A. Obviously, classical optimization algorithms do not work satisfactorily on such a function.

One way to attack this problem is to construct an algorithm that can select potentially interesting points from either

- the set of acceptable points A or
- the set C of all points having a cost smaller than some constant.

In [7], A is approximated by a special procedure, and then a strategy for selecting points from C is recommended.

In this paper, we will show that better results may be obtained by

- a different approximation of A,
- a more systematic way of selecting discrete points from A in the case of performance optimization of $F(x)$,
- a different way of selecting points from A in the case of cost optimization of $C(x)$.

In Section II, definitions are given. The problems are then formulated in Section III. Section IV contains a summary of the Gaussian adaptation search process, and Section V describes the discrete optimization procedure for quadratic functions. An outline of the new method is presented in Section VI, and, finally, some examples are given in Section VII.

II. Definitions

Designable Parameters

$x \equiv n$-dimensional vector of designable parameters

$x \in \mathbf{R}^n$ for continuous optimization problems

$x \in \mathbf{D}^n$ for discrete optimization problems.

$\mathbf{D}$ is some numerable set of numbers, e.g. integers.

Performance Function $F(x)$

A function F: $\mathbf{R}^n \to \mathbf{R}^r \to \mathbf{R}^1$ mapping the input space to the filter response space and further to the performance space using the filter specifications.

Example:

f	frequency
$H(f,x)$	transfer function
$A(f,x)$	$-20*\log_{10}(\lvert H(f,x)\rvert)$, attenuation
$U(f)$	upper requirement
$L(f)$	lower requirement
$F(x)$	$\max_f(A(f,x)-U(f), L(f)-A(f,x))$ or
$F(x)$	$\Sigma_f(A(f,x)-U(f))^2$, where we assume $U(f)=L(f)$.

The Set of Acceptability

$A \equiv \{x \mid F(x) < a\}$, $a =$ some constant.

Manuscript received January 21, 1986; revised August 5, 1986.
G. Kjellström and L. Taxén are with Ericsson Telecom, S-126 25 Stockholm, Sweden.
P. O. Lindberg is with the Royal Institute of Technology, Stockholm, Sweden.
IEEE Log Number 8715939.

0098-4094/87/1000-1238$01.00 ©1987 IEEE

Cost Function $C(x)$

According to Jain [7], $C(x)$ is defined as follows: let

$$x_j = \sum_{i=0}^{m} b_{ji} 2^{-i}$$

be the jth component of x, $b_{ji} \equiv \{-1,0,1\}$ the ith bit of x_j, and m the number of bits to represent x_j. The cost of integration $C(x)$ is then given by

$$C(x) = \sum_{j=1}^{n} C(x_j)$$

where $C(x_j)$ is the cost of an individual coefficient as follows:

$$C(x_j) = \sum_{i=1}^{m} |b_{ji}|$$

which means that the bit corresponding to 2^0 is not included.

The Set of Low Cost

$\boldsymbol{C} \equiv \{x | C(x) < a\}$, $a =$ some constant.

The Gaussian p.d.f.

$G(x) = (2\pi)^{-n/2}(\det M)^{-1/2} \exp Q$

$Q(x) = -(x-\mu)^T M^{-1}(x-\mu)/2$

where μ is the average of Gaussian p.d.f., and M is the moment matrix of Gaussian p.d.f.

III. Problem Formulation

Minimize the nonlinear continuous function $F(x)$ wrt x subject to

$$x \in \boldsymbol{D}^n.$$

This problem, which will be referred to as (P1) is the classical discrete optimization problem in digital filter design. In practical cases, it is more relevant to minimize the cost function $C(x)$. We then have the following problem.

Minimize the discrete nondifferentiable function $C(x)$ subject to

$$x \in \boldsymbol{A}$$

and

$$x \in \boldsymbol{D}^n.$$

This problem will be referred to as (P2). Both these problems will be considered in the following. In addition, the following subproblem has to be solved.

Minimize the quadratic function

$$Q(x) = (x-\mu)^T M^{-1}(x-\mu)$$

subject to

$$x \in \boldsymbol{D}^n.$$

This problem will be referred to as (Q).

IV. Gaussian Adaptation

According to [1], a Gaussian distribution can, in a fairly straightforward way, be adapted to the set $\boldsymbol{A}$ to give maximum hitting probability on $\boldsymbol{A}$. This gives us maximum (in the maximum entropy sense) information about the location and orientation of $\boldsymbol{A}$ in the parameter space.

Since the concept of Gaussian adaptation (GA) has been reported elsewhere [1], we only give a short resume of its main features here.

A. Dispersion

The first point is the choice of the Gaussian prior to other distributions as a sampling distribution for this kind of optimization procedure. The reason is that a Gaussian variable has the largest *dispersion* ("ability to cover $\boldsymbol{R}^n$") among variables having equal moment matrices, where the dispersion is defined by

$$d = \exp\left(-\int V(x) \log V(x)\, dx\right).$$

The expression in parentheses is usually referred to as the *entropy* [8] of the variable X having the p.d.f. $V(x)$. While the entropy is usually not uniquely defined, the dispersion is always unique and is exactly equal to the volume of a set $\boldsymbol{A}$ if $V(x)$ is uniformly distributed over $\boldsymbol{A}$.

B. Maximizing Conditions

The next point is that it is possible to derive necessary conditions for $G(x)$ to maximize the hitting probability. Keeping the determinant of M constant, these conditions are

$$\mu = \mu^*$$
$$M = cM^*, \qquad c = \text{constant}$$

where μ and M are the average and the moment matrix of $G(x)$, and μ^* and M^* the corresponding moments restricted to $\boldsymbol{A}$. This means that $G(x)$ will adapt to the location and extension of $\boldsymbol{A}$.

V. Discrete Optimization of Quadratic Functions

A *quadratic function minimization* (QFM) routine for the problem (Q) above when the matrix M^{-1} is positive semidefinite has been constructed. The same ideas as in branch and bound techniques for linear integer programming are used. That such techniques may be applied also to nonlinear problems has been observed by Dakin [6].

The general ideas behind the routine are simple. First, problem (Q) is relaxed by solving the associated continuous problem, which will be referred to as (QC):

$$\text{minimize } (x-\mu)^T M^{-1}(x-\mu)$$
$$x \in \boldsymbol{R}^n.$$

Now an optimal solution x to (QC) is computed. If this solution is integer (discrete), the optimum is found.

Otherwise, if x is noninteger, (Q) is split into two subproblems, say (Q1) and (Q2), by adjoining the constraints $x_i \leqslant [x_i]$, $x_i \geqslant [x_i]+1$, respectively ($[a]$ denoting the integer part of a). Either of these conditions is obviously fulfilled by any integer solution to (Q).

Next, the problems (Q1) and (Q2) are addressed in the same way; i.e., we relax the integrality constraints and compute the optimal continuous solutions. Then we proceed with the problems with noninteger solutions, etc. In this way, a sequence of subproblems is generated.

For a given subproblem (Qi), splitting does not take place if either of the following conditions is fulfilled (in which case one says (Qi) is fathomed):

(i) the optimal continuous solution is integer,
(ii) the problem is infeasible,
(iii) the optimal continuous function value is worse than that of the best discrete solution to (Q) found so far.

The process continues until all generated subproblems are fathomed. Then the optimal solution to (Q) is the best found discrete solution.

When one wants to find, e.g., the N best solutions to (Q), (iii) above is changed to

(iii′) the optimal value is worse than that of the Nth best discrete solution so far.

Due to the fact that we are minimizing a fixed quadratic function under varying bounds on the variables, efficient routines may be designed for optimization of the problems (Qi). The number of parameters that can be handled in practice is of course limited by the available computer resources. On a VAX 780, this limit is about 15 parameters.

It must be further noted that one must find the global optimum to each subproblem (Qi) in order to guarantee the finding of the global optimum to (Q). This seems not to have been noted in Charalambous and Best [4], who are also using branch and bound techniques. They claim that they have found the optimal solution to (P1) in example 1 below, in spite of the fact that they do not demonstrate that they solve their subproblems (P1i) optimally.

In our case, the problems (Qi) are solved optimally, since M^{-1} is a positive semidefinite matrix (and hence Q is convex); furthermore, the restrictions are linear. Thus, the global optimim of Q is always found.

VI. The Method

Compared with the classical optimization problems, where a differentiable function has to be minimized by some continuous gradient search method, the new discrete problems are much more irregular.

If we consider (P1), we have a continuous function $F(x)$ to minimize, but since x is restricted to a set of discrete points, the discrete optimum may be very far from the continuous one. The search for small $F(x)$ in the vicinity of the continuous optimum is therefore not sufficient.

In the case of (P2), things become even more difficult because now $C(x)$ is not a continuous function. It exists only at the discrete x points, and in general, no inherent structure of $C(x)$ can be utilized in the optimization. This means for example that the classical way of searching, where neighboring points to x are examined for smaller $F(x)$, will in general not lead to the optimal point.

Thus, a quite different approach is needed. We must somehow confine the search for the optimal point to "hot" regions in the parameter space. In the method to be presented, we first use GA to approximate the set A. This is motivated by the fact that A contains points with low values of $F(x)$, and thus is a region where the optimal solutions are found. From Section IV we know that GA is an adaptive stochastic search process which strives to "cover" A with a Gaussian distribution $G(\mu, M)$. The moments of the distribution are adapted in such a way as to give maximum "hitting" probability of samples drawn from G to fall in A. In a certain sense, this give us maximal information about A.

The moment matrix M of G defines the quadratic function $Q(x)$, which can be considered as an approximation of the "global" behavior of $F(x)$ over A. By global approximation we mean that $F(x)$ is approximated over a large region, contrary to the local approximation given by, e.g., the Hessian of a gradient method. This helps us to locate optimal discrete solutions far away from the optimal continuous solution.

As pointed out, it would have been possible to use gradient search to find a continuous minimum to $F(x)$ and a corresponding quadratic approximation of $F(x)$ at that minimum. But such an approximation depends on the local properties of $F(x)$ and is thus not necessarily a good approximation to A (Fig. 1). Consequently, a method based on this does not work well.

$Q(x)$ can then be minimized using QFM to give the N discrete points having the smallest Q values. Since G is adapted for maximum hitting probability over A, the chances of finding many points having small F values among the N points are very good. This is demonstrated by Examples 1 and 2 in Section VII.

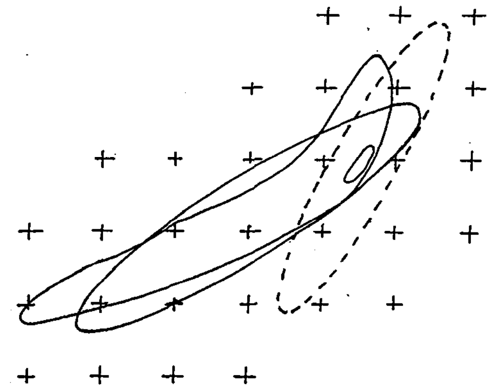

Fig. 1. Two different quadratic approximations of the same banana-shaped set. ——— GA approximation. ----- Approximation using the Hessian at the minimum point of $F(x)$.

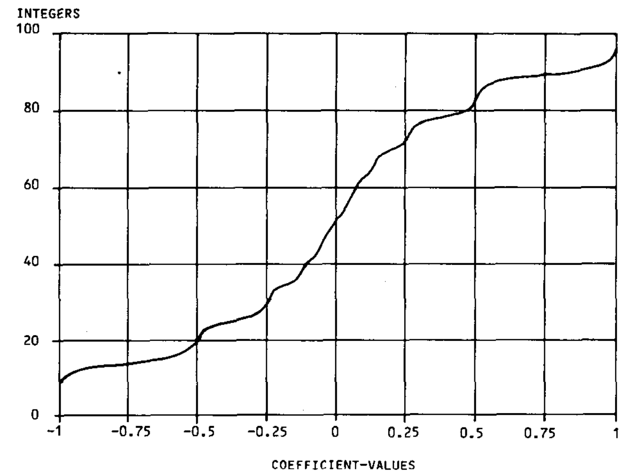

Fig. 2. Function mapping filter coefficients with $\leqslant 2$ bits different from zero to consecutive integers. Seven fractional bits are used in the determination of this function. Nondiscrete values are linearly interpolated.

There may exist cases where GA is less useful. For instance, if A is very much like an n-dimensional sphere containing a tremendous number of discrete points, then the N best points will be close to the center of A. Therefore, if the desired solution is close to the surface of A, it will probably not be found.

However, in many filter design examples, the set A tends to form long thin "cigars" or thin "disks" that may also be curved. The advantage of GA is that the most voluminous part of such a region can be located and approximated by a Gaussian distribution. Therefore, we have high probability of finding acceptable discrete points. Discrete points far away from the continuous minimum of $Q(x)$ or $F(x)$ may in principle be found in this way (see Fig. 1). But the optimal solution might still not be found. It may be close to the surface of the cigar or it may exist in some other less voluminous part of A.

To solve (P2), we proceed in much the same way. Instead of finding the point in A with the best performance value of $F(x)$, we now want to find one or more points which have the lowest possible cost $C(x)$ and still are feasible, i.e., belong to A. Since each point in the discrete x space has a cost associated with it (the number of nonzero bits), we can now optimize $Q(x)$ using a discrete grid of only low-cost points. For example, if we have five coefficients and use at most two nonzero bits for each coefficient, the maximum cost will be 10. The optimization of $Q(x)$ is now merely a way to select those low-cost points that have a high possibility of belonging to A.

If no feasible points are found, the maximum allowable cost for each coefficient is raised, and the optimization of $Q(x)$ is redone. After each optimization, the N best discrete points are found and checked for feasibility.

In order to concentrate the search on the low-cost points, the nonequidistant grid is mapped to an equidistant grid using a linear interpolation function (shown in Fig. 2) before GA is carried out. In our experience this gives a higher probability of finding low-cost solutions. This has been confirmed by running several examples of type (P2) (Example 3 below).

VII. Examples

Three examples have been chosen to demonstrate the capability of the method.

Example 1: The first one—given by [2]—is a wide-band differentiator. The filter configuration is a single second-order section having a transfer function

$$H(f) = g(1 + az^{-1} + bz^{-2})/(1 + cz^{-1} + dz^{-2})$$
$$z = \exp(j\pi f)$$

and 8-bit representation (sign bit, one integer bit, and six fractional bits) is to be used for a, b, c, d, and g. An integer constrained parameter vector is defined by multiplying each of the five variables by 64.

The criterion function is defined as

$$F(x) = 10000 \sum_{i=1}^{21} \left(|H(f_i)| - f_i\right)^2$$
$$f = .0, .05, .1, .15, \cdots, .95, 1.$$
$$x = a, b, c, d, g.$$

The optimization run was carried out in two steps, as follows. In the first step, a Gaussian distribution was adapted to a set of points defined by $F(x) < 5.2796$, which was the second best solution found by the algorithm used in [2]. The number of function evaluations (F) needed was $= 1000$.

In the second step, the 100 best discrete points of the corresponding quadratic function are searched for and finally their F values are calculated. The table below shows the 10 best function values sorted in ascending order.

Values of $F(x)$ in ascending order:

5.2230	5.2796	5.2865	5.3833	5.5742
5.7560	5.8822	5.8874	5.8986	6.3872

The best discrete solution is

$$-19 \quad -47 \quad 59 \quad 8 \quad 23.$$

Example 2: In this example, given by Rader and Gould and studied by Steiglitz [3], we consider the following transfer function:

$$H(f) = g \prod_{i=1}^{4} \left\{ \left(1 + a_i z^{-1} + z^{-2}\right) / \left(1 + b_i z^{-1} + c_i z^{-2}\right) \right\}$$
$$z = \exp(j\pi f).$$

Some restrictions are necessary for the proper position of poles and zeros:

$$|a_i| < 2, \qquad c_i < 1, \qquad 1 + c_i + b_i > 0, \qquad 1 + c_i - b_i > 0$$
$$128 * a_i,\ 128 * b_i,\ 128 * c_i \text{ are integers.}$$

The criterion function is defined by

$$F(x) = \max\left(\left||H(f)| - r\right|\right)/h$$
$$r = 1, h = .03 \qquad \text{for } 0 < f < .4$$
$$r = 0, h = .001 \qquad \text{for } .44 < f < 1$$
$$x = a_1, b_1, c_1$$
$$a_2, b_2, c_2$$
$$\cdots$$
$$a_4, b_4, c_4.$$

The gain factor g is adjusted to minimize F when x is given.

The discrete solution given by Steiglitz [3] has $F(x) = 1.018$. Since the continuous solution has $F(x) = .87$, a natural question to ask is whether there exists some discrete points with, say, $F(x) < 1$. Starting from the Steiglitz continuous solution

$128 * x =$

207.1686	−150.8402	54.2163
45.8626	−115.4443	84.7414
−24.2721	−87.5567	109.9035
−45.7889	−76.7622	123.3013

we first adapted a Gaussian distribution to the set A defined by $F(x) < 1$ for all x in A. For this purpose, we used 4000 evaluations of $F(x)$. After this, the 100 best points for the corresponding quadratic function were searched. Finally, their F values were calculated. By this method, nine points having $F(x) < 1$ were found. The ten best F values were

0.9616	0.9707	0.9782	0.9803	0.9889
0.9911	0.9925	0.9953	0.9991	1.0003.

The best discrete solution found ($F(x) = .9616$) was

$128 * x =$

207	−150	54
44	−114	85
−25	−86	110
−46	−76	123.

A similar run was performed with the set A of all x having $F(x) < .9616$, since this was the best solution thus far. It is, however, interesting to note that not even a point x having $F(x) < 1$ was found in this way. This is an indication that the continuous solution is very eccentric and that the corresponding ellipsoid does not include many good points. Thus, a method using the local behavior of F at x_0 would certainly be less efficient in this case.

Example 3: This example is given by Jain [7]. It is a fifth-order cascade voice-band filter.

CCITT specifications on the relative gain for the PCM transmit voice-band filter are given as upper ($U(f)$) and lower ($L(f)$) limits for the gain in the table below. Linear interpolation is assumed between the break points. The sample rate is 8 kHz and the reference frequency is 800 Hz.

f	$U(f)$	$L(f)$
0	0.125	−0.125
3000	0.125	−0.125
3400	0.7	0
4000		14
4599		14
4600		32
8000		32

The system performance function is now defined as follows:

$$z = \exp(j\pi f/8000)$$

$$H(f) = \left(1 + x_1 z^{-1} + x_2 z^{-2}\right)/\left(1 - x_3 z^{-1} - x_4 z^{-2}\right)$$
$$* \left(1 + x_5 z^{-1} + x_6 z^{-2}\right)/\left(1 - x_7 z^{-1} - x_8 z^{-2}\right)$$
$$* \left(1 + x_9 z^{-1}\right)/\left(1 - x_{10} z^{-1}\right)$$

$$A(f) = -20 * \log_{10}(|H(f)|) - A(800)$$

$$F(x) = \max_f \left(A(f) - U(f), L(f) - A(f)\right).$$

As in the preceding examples, we will use GA and QFM in order to select points from A. In contrast to the preceding examples, the problem now is to find points of low cost rather than points with a small value of $F(x)$. We will do this by simply using a grid of low-cost points during the discrete optimization and trying to find at least one such point inside A.

For instance, we may try coefficients having at most 2 bits different from zero in each word (1 bit is perhaps too optimistic). Seven fractional bits have been used in determining the function mapping coefficient values into integers, as shown in Fig. 2. As in the preceding example, we used 4000 calls for $F(x)$ in order to adapt the Gaussian. After this, the 100 best points for the quadratic function were selected and sorted in ascending order according to their F values. In this way, 38 points $x \in A$ were found. We calculated $C(x)$ for these points and we found 14 points with $C = 12$, 19 points with $C = 11$, and five points with $C = 10$. The point having $C = 10$ and at the same time the smallest value of F is

$x =$

0.265625	1.0	0.25	−0.75	1.0
1.0	0.1875	−0.3125	1.0	0.140625.

This can be compared with Jain's solution, which has a cost of 14.

Efficiency and Quality

In Example 1, we have used three times as many evaluations of $F(x)$ as Smith to find the global solution. But the GA algorithm is designed for sets A that have a very complicated structure, which means that A may contain several local minima of $F(x)$ or it may be constrained in different ways. It therefore seems reasonable that the GA algorithm consumes relatively more evaluations on very simple examples.

In Example 2, the exact number of evaluations was not given by Steiglitz, so it is difficult to compare the efficiency of the algorithms. But we got a higher quality in the final result and we think that this is more important in most cases. Besides, we can never tell how many extra evaluations Steiglitz would need to reach our result.

In the last example, efficiencies are also difficult to compare because we end up with a 40-percent better result with five times more effort. Nevertheless, a 40-percent cost reduction of the PCM voice-band filter means saving a lot of money as compared to some extra 3000 evaluations of $F(x)$. In similar situations, therefore, we think such extra evaluations should always be done if possible.

VIII. Conclusions

We have used GA in combination with QFM in order to solve certain discrete optimization problems in the area of digital filter design. The GA method was originally developed for centering and tolerancing in the design of analog filters, and it turns out that this method is, also very well suited for discrete optimization of digital filters.

The reasons for choosing the GA algorithm prior to gradient search are as follows.

- Min–max criteria usually used in filter design applications are not differentiable and therefore principally not suitable for gradient algorithms.
- Many functions that appear in real design problems are complicated and "ill-behaved" (e.g. corrupted by noise). This also makes it difficult to use ordinary optimization methods.
- In most filter design applications, the main concern is to find discrete points inside the set of acceptability. The Hessian, however, and the corresponding quadratic from are determined from the very local behavior of $F(x)$ in the vicinity of the continuous solution, and are not necessarily a good approximation to the set of acceptability. The GA algorithm, on the other hand, can adapt a Gaussian for maximum hitting probability to a set of any shape and is therefore a better springboard for the discrete search.

Acknowledgment

The authors are very grateful to K. Boestad of Ericsson for her contribution concerning discrete quadratic function minimization and for her preparation of the corresponding software modules.

References

[1] G. Kjellström and L. Taxén, "Stochastic optimization in system design," *IEEE Trans. Circuits Syst.*, vol. CAS-28, pp. 702–715, July 1981.

[2] N. I. Smith, "A random-search method for designing finite-wordlength recursive digital filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 1, pp. 40–46, Feb. 1979.

[3] K. Steiglitz, "Designing short-word recursive digital filters," in *Proc. 9th Annu. Allerton Conf. Circuit Syst. Theory*, Oct. 1971, pp. 778–788.

[4] C. Charalambous and M. J. Best, "Optimization of recursive digital filters with finite word lengths," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-22, pp. 424–431, Dec. 1974.

[5] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press, 1975.

[6] R. J. Dakin, "A tree-search algorithm for mixed integer programming problems," *Comput. J.*, vol. 8, pp. 250–255, 1966.

[7] R. Jain, "Computer-aided discrete-coefficient optimization for custom integrated digital filters," Katholieke Universitet Leuven, 1985.

[8] D. Middleton, *An Introduction to Statistical Communication Theory*. New York: McGraw-Hill, 1960.

[9] E. Avenhaus, "On the design of digital filters with coefficients of limited word length," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 206–212, Aug. 1972.

[10] F. Brglez, "Digital filter design with short wordlength coefficients," *IEEE Trans. Circuits Syst.*, vol. CAS-25, pp. 1044–1050, Dec. 1978.

[11] R. E. Crochiere, "A new statistical approach to the coefficient word length problem for digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 190–196, Mar. 1975.

[12] M. Suk and S. K. Mitra, "Computer-aided design of digital filters with finite word length," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 356–363, Dec. 1972.

[13] H.-K. Kwan, "On the problem of designing IIR digital filters with short coefficient word length," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 6, pp. 620–624, Dec. 1979.

[14] A. Antoniou, "New improved method for the design of weighted-Chebyshev, nonrecursive, digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-30, pp. 740–750, Oct. 1983.

[15] K. Boestad, G. Kjellström, L. Taxén, and P. O. Lindberg, "Discrete optimization using Gaussian adaptation and quadratic function minimization," in *Proc. China Int. Conf. Circuits Syst.* (Beijing), 1985, pp. 526–529.