AN ALGORITHM FOR ONE-SIDED $\mathbb{1}_1$ OPTIMIZATION WITH APPLICATION TO CIRCUIT DESIGN CENTERING

J.W. Bandler*, S.H. Chen* and K. Madsen**

*Optimization Systems Associates Inc.
163 Watson's Lane
Dundas, Ontario, Canada L9H 6L1

**Inst. for Numerical Analysis
Technical University of Denmark
DK-2800 Lyngby, Denmark

Abstract

A new and highly efficient algorithm for one sided nonlinear $\mathbf{1}_1$ optimization is presented. The new method is used as an integral part of an approach to design centering and yield enhancement.

Introduction

Gradient-based optimization techniques have become powerful tools serving practicing engineers in today's computer-aided design. The recent approach due to Hald and Madsen [1-3] has proved highly successful in solving minimax and $\mathbf{1}_1$ problems. Following the Hald and Madsen approach, we have developed a nonlinear one-sided $\mathbf{1}_1$ algorithm which combines a trust region Gauss-Newton method and a quasi-Newton method.

The one-sided $\mathbf{1}_1$ optimization problem can be stated as

$$\underset{\mathbf{x}}{\text{minimize}}\ U(\mathbf{x}) = \sum_{j \in J(\mathbf{x})} f_j(\mathbf{x}) , \qquad (1)$$

where $\mathbf{x} = [x_1\ x_2\ \ldots\ x_n]^T$ is a set of variables, $\mathbf{f} = [f_1\ f_2\ \ldots\ f_m]^T$ is a set of nonlinear functions, and

$$J(\mathbf{x}) = \{\, j \mid f_j(\mathbf{x}) > 0 \,\}$$

identifies the set of positive functions. In circuit design $\mathbf{f}$ may represent error functions arising from upper and lower specifications (e.g., Bandler et al. [4] have considered multiplexer design by the one-sided $\mathbf{1}_1$ optimization). We present an approach to design centering and yield enhancement of which the one-sided $\mathbf{1}_1$ optimization constitutes an integral part.

The new algorithm

The method we propose for solving (1) is a hybrid method combining a first order method with an appropriate second order method. The user must supply first order derivatives of the functions involved (besides the function values) whereas the second order derivatives are approximated automatically by the algorithm.

The first order method is denoted *Method 1* and the second order method is called *Method 2*.

Method 1 is the following iteration: At the k'th iterate $\mathbf{x}_k$ a local bound Λ_k is given. In order to find a better estimate of a solution the following linearized problem is solved,

$$\underset{\mathbf{h}}{\text{minimize}}\ \bar{U}(\mathbf{x}_k, \mathbf{h}) \equiv \sum_{j \in J_k} \{ f_j(\mathbf{x}_k) + \mathbf{f}_j'(\mathbf{x}_k)^T \mathbf{h} \}$$

subject to $\|\mathbf{h}\|_\infty \le \Lambda_k$ where

$J_k = J_k(\mathbf{x}_k + \mathbf{h}) = \{\, j \mid f_j(\mathbf{x}_k) + \mathbf{f}_j(\mathbf{x}_k)^T \mathbf{h} > 0 \,\}$.

This is equivalent with the following problem,

$$\underset{\mathbf{h}, \mathbf{y}}{\text{minimize}} \sum_{j=1}^{m} y_j \qquad (2a)$$

subject to

$$y_j \ge f_j(\mathbf{x}_k) + \mathbf{f}_j'(\mathbf{x}_k)^T \mathbf{h}, \quad j = 1, 2, \ldots, m, \qquad (2b)$$

$$y_j \geq 0, \qquad j = 1,2,\ldots, m, \quad (2c)$$

$$\Lambda_k \geq h_i, \ \Lambda_k \geq -h_i, \qquad i = 1,2,\ldots, n, \quad (2d)$$

where $\mathbf{f}_j'$ denotes the gradient vector of f_j w.r.t. $\mathbf{x}$. This subproblem can be solved by a standard linear programming routine. The constraints (2b) and (2c) define a piecewise linearized model for each f_j, as $y_j = \max \{ 0 , f_j + (\mathbf{f}_j')^T\mathbf{h} \}$. The index set $J(\mathbf{x}_k+\mathbf{h})$ is approximated by $J_k(\mathbf{x}_k+\mathbf{h})$ which is updated at each step of solving the linear program.

In contrast, a more conventional approach to the one-sided problem is to define $f_j^+ = \max\{0,f_j\}$ and minimize the $\mathbf{l}_1$ norm of f_j^+ using a conventional (two-sided) algorithm. This approach assumes

either $y_j \equiv f_j + (\mathbf{f}_j')^T\mathbf{h}$

or $y_j \equiv 0$

throughout an iteration of solving one subproblem. In other words, $\mathbf{J}(\mathbf{x})$ is approximated by

$$\mathbf{J}(\mathbf{x}_k) = \{ j \mid \mathbf{f}_j(\mathbf{x}_k) > 0 \}$$

which will not be updated for an entire iteration. In our new algorithm, by allowing the index set $\mathbf{J}$ to vary within an iteration, the discontinuity at $y_j = 0$ is taken into account in solving the subproblem.

The set of constraints (2d) defines a trust region in which the linearized model is considered to be a good approximation to the nonlinear functions. The local bound Λ_k is adjusted after each iteration based on the goodness of the linearized model, using criteria similar to those described in [4].

Method 2 applies a quasi-Newton method (BFGS) to solving a set of optimality equations given by

$$\sum_{j\in J} \mathbf{f}_j'(\mathbf{x}) + \sum_{j\in Z} \delta_j \mathbf{f}_j'(\mathbf{x}) = 0,$$

$$\mathbf{f}_j(\mathbf{x}) = 0, \quad j \in Z, \qquad (3)$$

where Z identifies the set of functions that are zero at the optimum. The multipliers δ_j, $j\in Z$, must satisfy $1\geq\delta_j\geq 0$. These optimality equations result from applying the Kuhn-Tucker conditions to the one-sided $\mathbf{l}_1$ problem. They are different from, but similar to the optimality equations for the (two-sided) $\mathbf{l}_1$ problem [4].

The Hybrid Method. Based on the theory of Hald & Madsen [1-3], our algorithm combines the trust region Gauss-Newton method (Method 1) with the quasi-Newton iteration (Method 2). Method 1 is intended to be used away from a solution, i.e. it should provide the global convergence, and Method 2 is used when a solution is approached to obtain a fast local convergence.

Initially, Method 1 is used, and the set Z identifying the functions which are zero at the solution is estimated. When a local minimum seems to be approached the nonlinear system (3) is set up using the current estimate of Z, and a switch to Method 2 is made. If the Method 2 iteration is unsuccessful (e.g. if the estimate of Z is wrong), then the safe method, Method 1, is used again. Several switches between the two methods may take place.

It should be mentioned that linear equality and inequality constraints can be readily incorporated into the algorithm (similarly to [4]).

Design centering and yield enhancement

One important application of the one-sided $\mathbf{l}_1$ algorithm is found in circuit design centering and yield enhancement [5].

Given a set of circuit parameters ϕ and a set of performance specifications, we can calculate a set of error functions $\mathbf{e}(\phi)$ and a generalized $\mathbf{l}_p$ function $v(\mathbf{e}(\phi))$ [5,6]. The sign of v signifies the acceptability of ϕ. A nonpositive v

indicates that all the specifications are satisfied, whereas a negative v indicates that some specifications are violated.

Given a nominal design ϕ^0, we can generate some Monte Carlo points, denoted by ϕ^k, k = 1, 2,..., K, according to the statistical distribution of the toleranced circuit parameters. Let the total number of points (ϕ^k) which violate the specifications be K_{fail}, given by the total number of nonpositive $v(\phi^k)$. Then a discrete estimate of the yield is given by $(K - K_{fail})/K$. It is a matter of great significance to circuit engineers to find a centered design ϕ^0 which minimizes K_{fail}. However, a direct minimization of K_{fail}, which is a discrete number, using gradient-based techniques is not practical.

Consider the one-sided $\mathbf{l}_1$ sum defined as

$$U(\phi^0) = \sum_{k \in J} \alpha_k v_k \quad , \qquad (4)$$

where $v_k = v(\phi^k)$ and $J = \{ k \mid v_k > 0 \}$. Note that the variables to be optimized here are the nominal point ϕ^0. In (4) we define a set of multipliers α_k which are calculated at the starting point as $\alpha_k = 1/v_k$ and kept constant during optimization. The one-sided $\mathbf{l}_1$ objective function $U(\phi^0)$ as defined in (4) becomes precisely K_{fail} (the number of Monte Carlo points that fail to meet the specifications) at the starting point. By minimizing $U(\phi^0)$ which is used as a smooth and convex interpolating function for K_{fail}, we wish to achieve a centered design and an enhanced yield. The one-sided $\mathbf{l}_1$ algorithm described in this paper serves as a powerful tool.

Consider as an example a Chebyshev lowpass filter which has 11 parameters [7]. We assume a 1.5% relative tolerance with a uniform distribution for each circuit parameter. The nominal design standard synthesis was used as a starting point. It has a yield of 49%. The centered solution found by our algorithm improves the yield to 84%. The solution, as shown in Table 1, was achieved by a sequence of three design cycles, with a total CPU time of 66 seconds on the VAX 8600.

TABLE 1
One sided $\mathbf{l}_1$ centering of Singhal and Pinel's filter

Component	Nominal Values			
	Case 1	Case 2	Case 3	Case 4
x_1	0.2251	0.21954	0.21705	0.21530
x_2	0.2494	0.25157	0.24677	0.23838
x_3	0.2523	0.25529	0.24784	0.24120
x_4	0.2494	0.24807	0.24019	0.23687
x_5	0.2251	0.22042	0.21753	0.21335
x_6	0.2149	0.22628	0.23565	0.23093
x_7	0.3636	0.36739	0.37212	0.38224
x_8	0.3761	0.36929	0.38012	0.39023
x_9	0.3761	0.37341	0.38370	0.39378
x_{10}	0.3636	0.36732	0.37716	0.38248
x_{11}	0.2149	0.22575	0.22127	0.23129
Yield	49%	78%	80%	84%
Number of multiple circuits used		50	100	100
Starting point		Case 1	Case 2	Case 3
Number of iterations		16	18	13
CPU time (VAX 8600)		10 sec.	30 sec.	26 sec.

A uniformly distributed 1.5% relative tolerance is assumed for each component. The yield in this table was estimated by Monte Carlo analyses with 300 samples. The parameter values in Case 1 were obtained by standard filter synthesis [7].

REFERENCES

[1] J. Hald and K. Madsen, "Combined LP and quasi-Newton methods for minimax optimization", **Math. Programming**, vol. 20, 1981, pp. 49-62.

[2] J. Hald and K. Madsen, "Combined LP and quasi-Newton methods for nonlinear $\mathbf{l}_1$ optimization", **SIAM J. Numerical Analysis**, vol. 22, 1985, pp. 68-80.

[3] K. Madsen, "Minimization of non-linear approximation functions". Doctor technices thesis, Institute for Numerical Analysis, Tech. Univ. Denmark, DK-2800 Lyngby, Denmark, 1985.

[4] J.W. Bandler, W. Kellermann and K. Madsen, "A nonlinear $\mathbf{l}_1$ optimization algorithm for design, modelling and diagnosis of networks", **IEEE Trans. Circuits and Systems**, vol. CAS-34, 1987, pp. 174-181.

[5] J.W. Bandler and S.H. Chen, "Circuit optimization: the state-of-the-art", **IEEE Trans. Microwave Theory Techn.**, vol. MTT-36, 1988.

[6] J.W. Bandler and C. Charalambous, "Theory of generalized least pth approximation", **IEEE Trans. Circuit Theory**, vol. CT-19, 1972, pp. 287-289.

[7] K. Singhal and J.F. Pinel, "Statistical design centering and tolerancing using parametric sampling", **IEEE Trans. Circuits and Systems**, vol. CAS-28, 1981, pp. 692-701.