Augmented Lagrangian algorithm for optimizing analog circuit design

EINDHOVEN UNIVERSITY OF TECHNOLOGY
Department of Mathematics and Computing Science

RANA 02-32
December 2002

Augmented Lagrangian algorithm for optimizing analog circuit design

by

T.G.A. Heijmen, C. Lin, E.J.W. ter Maten, and M.F. Sevat

Reports on Applied and Numerical Analysis
Department of Mathematics and Computer Science
Eindhoven University of Technology
P.O. Box 513
5600 MB Eindhoven, The Netherlands
ISSN: 0926-4507

Augmented Lagrangian Algorithm for Optimizing Analog Circuit Design

T.G.A. Heijmen[1], C. Lin[1], E.J.W. ter Maten[1,2], and M.F. Sevat[1]

[1] Philips Research, Eindhoven, The Netherlands, Jan.ter.Maten@philips.com
[2] Eindhoven University of Technology, Eindhoven, The Netherlands

1 Introduction

The analog design assistance tool Adapt [5, 6] has been developed to help analog electronic circuit designers tuning design parameters, such that the functional design specifications are met, given process technology constraints. Tuning is based on an optimization process, in which each iteration of the optimization loop implies the evaluation of the circuit by an analog circuit simulator. Considering the simulator as a black box tool, the choice of the optimization technique is restricted, because the simulator does not automatically supply derivatives of the design metrics and numerical noise is inherently present (for instance due to adaptive time stepping). This excludes optimization algorithms that adopt finite-difference schemes to approximate derivatives.

One of the two optimization algorithms available in Adapt is the Nelder–Mead (NM) method [7]. Adapt includes constraints by adding quadratic penalty terms to the cost function when using NM. The Nelder–Mead algorithm is very robust but has rather poor performance characteristics.

The subject of the current paper is the second available algorithm in Adapt, named *Gridmom*. This algorithm uses an augmented Lagrangian as a merit function [1], which is minimized by a grid-based Trust-Region approach. In this process, the cost function is locally approximated by a (smooth) model function. This model function is minimized within the Trust Region, in which the model is assumed to be a good approximation. The minimum of the model defines the next evaluation point for the simulator. The evaluation results are used to update the approximating function and to adapt the Trust Region (by moving, by shrinking, or by expanding it). Clustering of evaluation points at an early stage is prevented by restricting evaluation to points on a grid [2]. A special feature of Adapt is the concept of ***priority groups*** that allows for a dedicated sequence of optimization problems. We briefly describe how Gridmom can accommodate this.

Finally, we apply Gridmom to a small example optimization problem.

2 Constrained optimization by augmented Lagrangian

The search for the optimal values of the optimization variables (OVs) $\mathbf{x}$ can be formulated as a nonlinear constrained optimization problem in n variables with m constraints,

$$\begin{array}{rl} \text{minimize} & f(\mathbf{x}), \quad \mathbf{x} = (x_1, x_2, \ldots, x_n)^\top, \\ \text{subject to} & c_i(\mathbf{x}) \leq 0, \quad i = 1, \ldots, m, \\ & a_j \leq x_j \leq b_j, \; j = 1, \ldots, n, \end{array} \tag{1}$$

where x_j denotes the j-th OV. The values of the objective function $f(\mathbf{x})$ and the constraining functions $c_i(\mathbf{x})$ are obtained from circuit simulation. The performance and stability of the optimization algorithm is affected by the *scaling* of the OVs, of $f(\mathbf{x})$ and of the $c_i(\mathbf{x})$ [6]. By introducing a slack variable $s_i \geq 0$, each inequality constraint in (1) can be rewritten as an equality: $c_i(\mathbf{x}) + s_i = 0$. The augmented Lagrangian penalty function can then be written as [1]

$$\Phi_{\mathrm{ALAG,s}}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \mathbf{s}) = \underbrace{f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i [c_i(\mathbf{x}) + s_i]}_{\mathcal{L}(\mathbf{x},\boldsymbol{\lambda})} + \sum_{i=1}^{m} \mu_i [c_i(\mathbf{x}) + s_i]^2, \tag{2}$$

in which $\mathcal{L}$ is the standard Lagrangian. The parameters λ_i and μ_i are Lagrange multipliers and penalty factors, respectively. Minimization over the slack variables s_i yields a simplified merit function that is used in the Gridmom algorithm,

$$\Phi_{\mathrm{ALAG}}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \sum_{i=1}^{m} \left\{ \mu_i \max \left[c_i(\mathbf{x}) - \frac{\lambda_i}{2\mu_i}, 0 \right]^2 - \frac{\lambda_i^2}{4\mu_i} \right\}. \tag{3}$$

In constrained minimization one may not have $\nabla_x f(\mathbf{x}^\star) = 0$ at the minimum. However, together with optimal $\boldsymbol{\lambda}^\star$, $(\mathbf{x}^\star, \boldsymbol{\lambda}^\star)$ becomes a stationary point of $\mathcal{L}$ and satisfies the Karush-Kuhn-Tucker (KKT) conditions [1]

$$\begin{array}{c} \nabla_x \mathcal{L}(\mathbf{x}^\star, \boldsymbol{\lambda}^\star) = 0, \\ c_i(\mathbf{x}^\star) \leq 0, \quad \lambda_i^\star \leq 0, \quad \lambda_i^\star c_i(\mathbf{x}^\star) = 0, \quad i = 1, \ldots, m. \end{array} \tag{4}$$

Hence there are $|\mu_i| < \infty$ such that Φ_{ALAG} has a local minimum in $(\mathbf{x}^\star, \boldsymbol{\lambda}^\star)$. Clearly, using the augmented Lagrangian is a better method for constrained minimization than just adding a quadratic penalty term to the objective function $f(\mathbf{x})$, which is done by Adapt when using Nelder-Mead.

3 Method of Multipliers and Trust-Region Minimization

The algorithm uses the *method of multipliers* (MOM) to solve the problem (1), with the augmented Lagrangian (3) as a merit function [5, 6]. The variables x_i and the penalty factors μ_i are initialised to $x_i^{(0)}$ and $\mu_i^{(0)}$, respectively, and the multipliers λ_i are set to $\lambda_i^{(0)} = 0$. Now the values of λ_i and μ_i are fixed and the merit function,

$$\Phi_{\mathrm{ALAG}}^{(k+1)}(\mathbf{x}) = \Phi_{\mathrm{ALAG}}(\mathbf{x}, \boldsymbol{\lambda}^{(k)}, \boldsymbol{\mu}^{(k)}), \tag{5}$$

is minimized, resulting in the argument $\mathbf{x}^{(k+1)} = \arg\min_x \Phi_{\mathrm{ALAG}}^{(k+1)}(\mathbf{x})$. In case of insufficient decrease in the i-th constraint violation, we increase the penalty factor $\mu_i^{(k+1)} := 10 * \mu_i^{(k)}$. Otherwise, we update the multiplier, based on comparing $\nabla_x \Phi_{\mathrm{ALAG}}$ with $\nabla_x \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda})$, by

$$\lambda_i^{(k+1)} = \lambda_i^{(k)} - 2\max[\mu_i^{(k)} c_i(\mathbf{x}^{(k+1)}), \lambda_i^{(k)}]. \tag{6}$$

This is repeated until the termination criteria have been satisfied. A combined set of variable values $\mathbf{x}^*$ and multipliers $\boldsymbol{\lambda}^*$ is regarded as a solution of (1), if the KKT conditions (4) are fulfilled. Each bound-constrained subproblem involves the minimization of a merit function $\Phi_{\mathrm{ALAG}}^{(k)}(\mathbf{x})$. To this purpose, an algorithm is applied that is similar to the one reported in [3]. This algorithm uses a grid and applies a Trust-Region approach. In addition $\Phi_{\mathrm{ALAG}}^{(k)}(\mathbf{x})$ is approximated by a *quadratic model function*,

$$q(\mathbf{x}) = a + \mathbf{g}^T(\mathbf{x} - \mathbf{x}^{\mathrm{ref}}) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^{\mathrm{ref}})^T \mathbf{G}\,(\mathbf{x} - \mathbf{x}^{\mathrm{ref}}), \tag{7}$$

which is minimized within a *Trust Region* $\mathcal{B} = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x} - \mathbf{x}^{\mathrm{ref}}\|_\infty \leq \Delta\}$ with radius Δ centered at the reference point $\mathbf{x}^{\mathrm{ref}}$. In the Trust Region, $q(\mathbf{x})$ is assumed to be a good estimate of the true merit function $\Phi_{\mathrm{ALAG}}^{(k)}(\mathbf{x})$. Since $q(\mathbf{x})$ is smooth, gradient type optimization algorithms are allowed.

Evaluation of the merit function $\Phi_{\mathrm{ALAG}}^{(k)}(\mathbf{x})$ is restricted to points on a grid. Note that the merit function can be re-evaluated cheaply at a gridpoint, after updating λ_i or μ_i, if $f(\mathbf{x})$ and the $c_i(\mathbf{x})$ have been calculated before and have been saved. This speeds up the building of the model function $q(\mathbf{x})$. Which evaluation points are used in building $q(\mathbf{x})$ depends on the distance to $\mathbf{x}^{\mathrm{ref}}$.

An accurate estimate of the gradient $\mathbf{g}$ is more crucial than an accurate approximation of the Hessian $\mathbf{G}$. Therefore, the accuracy of the gradient is improved locally by performing a second least-squares fit, with a fixed Hessian obtained from the first fit, using a small number of evaluation points close to $\mathbf{x}^{\mathrm{ref}}$ [6]. Let $\mathbf{x}^\circ$ be the gridpoint nearest to the optimum point of $q(\mathbf{x})$. If $\mathbf{x}^\circ$ is a new gridpoint, $f(\mathbf{x}^\circ)$ and the $c_i(\mathbf{x}^\circ)$ are evaluated and one sticks to the current grid, otherwise the grid is refined. The application of successively refined grids prevents the clustering of evaluation points in an early stage.

The rules for updating the Trust-Region radius Δ are based on the ratio [2]

$$\rho^{\Delta} = \frac{\Phi^{(k)}_{\mathrm{ALAG}}(\mathbf{x}^{\mathrm{ref}}) - \Phi^{(k)}_{\mathrm{ALAG}}(\mathbf{x}^{\circ})}{q(\mathbf{x}^{\mathrm{ref}}) - q(\mathbf{x}^{\circ})} \tag{8}$$

(which tends to 1 for a good approximation) and are according to

$$\begin{aligned} \rho^{\Delta} \geq \rho^{\max} &\Longrightarrow \Delta := 2\Delta, \\ \rho^{\Delta} < \rho^{\min} &\Longrightarrow \Delta := \frac{1}{2}\Delta, \\ \rho^{\min} \leq \rho^{\Delta} < \rho^{\max} &\Longrightarrow \text{no change.} \end{aligned}$$

Furthermore, $\Delta^{\min} \leq \Delta \leq \Delta^{\max}$.
The criterion for updating $\mathbf{x}^{\mathrm{ref}}$ is determined by

$$\rho^{\mathrm{ref}} = \frac{\Phi^{(k)}_{\mathrm{ALAG}}(\mathbf{x}^{\mathrm{ref}}) - \Phi^{(k)}_{\mathrm{ALAG}}(\mathbf{x}^{\mathrm{best}})}{q(\mathbf{x}^{\mathrm{ref}}) - q(\mathbf{x}^{\circ})}, \tag{9}$$

where $\mathbf{x}^{\mathrm{best}}$ is the best current estimate of the optimum point of $\Phi^{(k)}_{\mathrm{ALAG}}(\mathbf{x}^{\mathrm{ref}})$. Then $\mathbf{x}^{\mathrm{ref}}$ is updated according to

$$\begin{aligned} \rho^{\mathrm{ref}} \geq \rho^{\min} &\Longrightarrow \mathbf{x}^{\mathrm{ref}} := \mathbf{x}^{\mathrm{best}}, \\ \rho^{\mathrm{ref}} < \rho^{\min} &\Longrightarrow \text{no change.} \end{aligned}$$

At initialization, the merit function is evaluated at as many points as are necessary to enable the construction of an approximating function. The Gridmom algorithm starts using the Uniform Design (UD) approach [4], which is based on number theory. Another way is to adopt so-called *Priority Groups* (PG), that allows for splitting a multi-objective optimization problem into a sequence of optimizations, with increasing priority. It reflects the way analog and mixed-signal designers treat the various specifications in traditional design, but also elsewhere the approach is well-known. Target specifications values from previous PGs are treated as constraints in following groups. This clearly may narrow the region for the final $\mathbf{x}^{\star}$. However, also interesting problems arise when stepping to the next PG: (1) The set of optimization variables may change (even by dimension); (2) The objective function f may change; (3) The constraints c_i may change. To reduce costly evaluations one might express f and c_i into subquantities that can be saved allowing cheap re-evaluation of $\Phi_{\mathrm{ALAG}}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$.

4 Example and Conclusion

In Fig. 1 a small Bandpass Filter optimization example is shown. In the specifications, the Group Delay GD $= \frac{\delta \arg(H(\omega))}{\delta\omega}$ is to be kept constant, and the 3dB Bandwidth BW and Lower and Upper sideband suppression SBL, SBU, have to be maximized. The design variables in the schematic are the four inductors L_i and the four capacitors C_i. They are coupled by the fixed notch

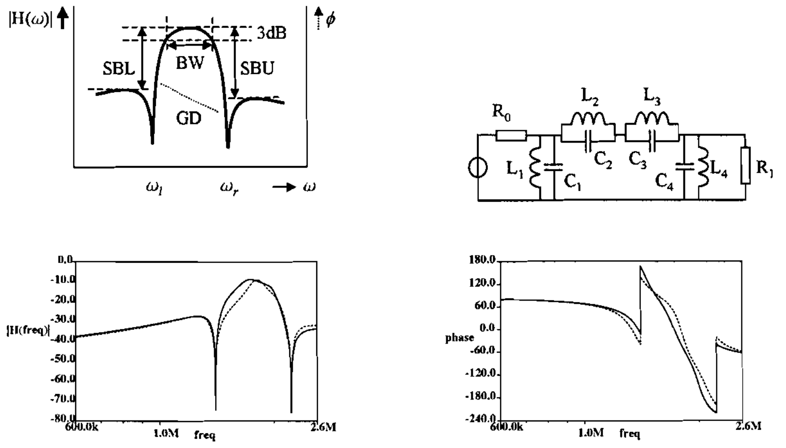

Fig. 1. Bandpass Filter: *Top-Left:* Quantities in specification of frequency transfer $H(\omega)$, *Top-Right:* Schematic, *Bottom-Left:* $|H(2\pi f)|$, *Bottom-Right:* $\arg(H(2\pi f))$

frequencies: $\omega_l = \frac{1}{\sqrt{C_2 L_2}}$, $\omega_r = \frac{1}{\sqrt{C_3 L_3}}$, by which the number of optimization variables reduces to 6. The dotted and straight lines show the initial and final results, respectively, for $|H(2\pi f)|$ and $\arg(H(2\pi f))$.

Our experience is that Gridmom is more efficient than Nelder-Mead within a single Priority Group. However, Nelder-Mead exploits the PG concept better.

References

1. M.S. Bazaraa, H.D. Sherali, C.M. Shetty: *Nonlinear programming: theory and algorithms*, 2nd edition, John Wiley and Sons, New York, 1993.
2. A.R. Conn, N. Gould, Ph.L. Toint: *Global convergence of a class of trust region algorithms for optimization with simple bounds*, SIAM J. Numer. Anal., Vol. 25, pp. 433-460, 1988.
3. C. Elster, A. Neumaier: *A grid algorithm for bound constrained optimization of noisy functions*, IMA J. Numer. Anal., Vol. 15, pp. 585–608, 1995.
4. K.-T. Fang, Y. Wang: *Number-theoretic methods in statistics*, Chapman and Hall, London, 1994.
5. M. Kole, T. Heijmen, T. Kevenaar, H.-J. Pranger, M. Sevat: *Adapt, an interactive tool for analog synthesis*, In: *Proc. SAME-2001 Conf.*, Sophia Antipolis, France, pp. 15–19, 2001.
6. C. Lin, T. Heijmen, J. ter Maten, M. Kole: *ADAPT: Design assistance for iterative analog synthesis*, In: *Analog 2002 - Entwicklung von Analogschaltungen mit CAE-Methoden (13-14 May 2002, Bremen)*, GMM-Fachbericht 38, VDE Verlag GMBH, Berlin, pp. 195–200, 2002.
7. K.I.M. McKinnon: *Convergence of the Nelder-Mead simplex method to a non-stationary point*, SIAM J. Optimization, Vol. 9-1, pp. 148–158, 1998.