Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

OPTIMIZATION OF ANALOG CIRCUITS VIA SIMULATION AND A LAGRANGIAN-TYPE GRADIENT-BASED METHOD

Eunji Lim

School of Management and Marketing
Kean University, NJ, USA

Youngmin Kim

Department of Computer Engineering
Kwangwoon University, Seoul, Republic of Korea

Jaehyouk Choi

School of Electrical and Computer Engineering
UNIST, Ulsan, Republic of Korea

ABSTRACT

We propose a new method for determining the physical sizes of components in an electrical circuit that maximize some primary performance measure while satisfying some conditions on the secondary performance measures. The proposed method is based on the observation that the performance measures are unimodal and smooth. Thus, it focuses on a local search and applies a Lagrangian method to search for a local optimum. The proposed method has advantages over existing methods because it does not rely on approximate formulas for the performance measures, like other equation-based methods do, and finds the "exact" optimal solution by calling an electrical circuit simulator, such as SPICE, at each iteration to evaluate the performance measures and to compute their gradients. The proposed method also enjoys fast convergence because it focuses on a local search rather than global searches. Numerical experiments illustrate the effectiveness of the proposed method in a one-stage operational amplifier.

1 INTRODUCTION

One of the challenges we face when designing analog circuits in semiconductor manufacturing companies is determining the physical sizes of the transistors and other components that are included in the circuits. The performance measures of the circuits are directly related to the physical sizes of their components. For example, the widths of the transistors in an operational amplifier (op-amp) affect the performance measures of the circuit such as the gain, the phase margin, and the 3–dB bandwidth. In such a case, it is important to determine the widths of the transistors so that we can achieve the best possible performance on the primary measure while maintaining satisfactory levels of the secondary performance measures. The problem of determining the physical sizes of the components in a circuit can be formulated as the following optimization problem:

$$\begin{aligned} &\text{maximize} && f(x) \\ &\text{subject to} && g_i(x) \geq 0, \quad i = 1, \ldots, r, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^d$ is a vector of decision variables (such as the widths of the transistors), $f : \mathbb{R}^d \to \mathbb{R}$ is the primary performance measure we wish to maximize (such as the gain), and the equations $g_i \geq 0$ are the conditions on the secondary performance measures (such as the conditions on the phase margin and the 3–dB bandwidth) with $g_i : \mathbb{R}^d \to \mathbb{R}$ for $1 \leq i \leq r$.

In this paper, we propose a new method for solving (1). The main difficulty in solving (1) is that f and the g_is cannot be easily described with mathematical equations. For a complex circuit, it is not obvious

how to find closed-form formulas for f and the g_is. Even when the circuit is relatively simple, we need to use several approximations to express f and the g_is in mathematical equations. These approximations often lead to an inaccurate solution to (1). The main feature of our proposed method is that we do not try to find equations for f and the g_is. Instead of using formulas for f and the g_is, the proposed method calls an electric circuit simulator, such as SPICE, to accurately evaluate the performance measures.

This idea was inspired when we observed that numerous performance measures arising in the circuit sizing problems are unimodal and smooth as functions of the decision variables. For example, Fig. 1 shows the logarithm of the gain and the 3-dB bandwidth of the one-stage op-amp shown in Fig. 2 as functions of $\log x_1$ and $\log x_2$, where x_1 represents the widths of transistors 1 and 2, and x_2 represents the widths of transistors 3 and 4. These functions appear to be unimodal and smooth, and these properties were observed

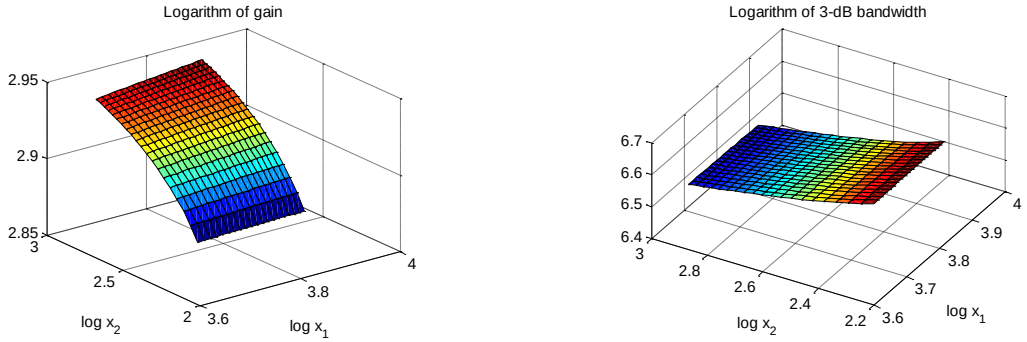

Figure 1: The graph on the left side is the logarithm of the gain (dB). The graph on the right side is the logarithm of the 3-dB bandwidth (MHz). In all graphs, the horizontal axes are $\log x_1$ and $\log x_2$, where x_1 represents the widths of transistors 1 and 2, measured in μm, and x_2 represents the width of transistors 3 and 4, measured in μm.

in a majority of the circuit design problems. Based on this observation, we propose a local search algorithm that can be best suited for unimodal and smooth functions. The proposed method rewrites problem (1) using the Lagrangian function $L(x,\lambda) = f(x) + \lambda_1 g_1(x) + \cdots + \lambda_r g_r(x)$ with $\lambda_i \in \mathbb{R}$ for $1 \leq i \leq r$, and iteratively updates the decision variable x and the dual variable $\lambda = (\lambda_1, \ldots, \lambda_r)$ using a gradient-based method. The proposed method resembles the Arrow-Hurwicz-Uzawa algorithm (Arrow, Hurwicz, and Uzawa 1958) in the sense that it updates the decision variable and the dual variable using a gradient-based method. However, the proposed method is applied to the case where f and the g_is are not available as functional equations, and thus computer simulation is used to estimate the gradients. More precisely, when we update the decision variable x and the dual variable λ, we evaluate f and the g_is using a circuit simulation program such as SPICE at various values of x and use the finite differences to estimate the gradients of f and the g_is. The use of a circuit simulation program enables us to accurately evaluate f and the g_is and to estimate their gradients without any functional formulas for f and the g_is.

As the integrated circuit (IC) technology continues to progress, it is increasingly important to design circuits more accurately. For example, op-amps are one of the basic building blocks of many analog or mixed-signal ICs, so it is critical to design them so that they can achieve the best possible performance. If we approximate f and the g_is using mathematical equations and solve (1) with these equations, it typically takes a few seconds to reach a "crude" solution. However, it is more desirable to find the "exact" global optimum even if it takes a longer period of time. In this regard, the proposed method has a significant advantage because it reaches the "exact" optimal solution in a reasonable amount of time. Numerical experiments in Section 3 show that the optimal solution is found within 4 minutes for a one-stage op-amp.

The idea of using optimization techniques for the circuit design problems has received a great deal of attention within the past few decades. Previously developed methods can be divided into three categories: 1) Knowledge-based methods, 2) Equation-based methods, and 3) Simulation-based optimization methods.

In a knowledge-based method, an expert sets a list of rules obtained from previous experiences and executes them during the sizing process (Degrauwe 1987, El-Turky and Perry 1989, Harjani, Rutenbar, and Carley 1989, Beenker, Conway, Schrooten, and Slenter 1993, Antoa and Brodersen 1995, Jangkrajarng, Bhattacharya, Hartono, and Shi 2003). One drawback of the knowledge-based method is that it cannot be automated because an experienced specialist needs to monitor the process constantly. Furthermore, it often fails to produce the optimal solution because the search is not exhaustive and is rather heuristic.

In an equation-based method, the performance measures are approximated by mathematical equations and the equations are used during the sizing process (Koh, Sequin, and Gray 1990, Horta, Franca, and Leme 1991, Maulik and Carley 1991, Harvey, Elmasry, and Leung 1992, Gielen, Debyser, Lampaert, Leyn, Swings, Plas, Sansen, Leenaerts, Veselinovic, and Bokhoven 1995, Medeiro, Perez-Verdu, Rodriguez -Vazquez, and Huertas 1995, Horta and Franca 1996, Ochotta, Rutenbar, and Carley 1996, Fernández-Fernandéz, Rodríguez-Vázquez, Huertas, and Gielen. 1998, Hershenson, Boyd, and Lee 1998, Chen, Chu, and Wong 1999, Hershenson, Boyd, and Lee 2001, Boyd, Kim, Patil, and Horowitz 2005). This approach has the advantage of computing the solution quickly, but it requires finding the mathematical equations for all performance measures. This requirement can be quite restrictive because there are a number of cases where we need to use several approximations to express the performance measures in mathematical equations, or there are no appropriate equations for the performance measures. As an example of an equation-based method, Hershenson, Boyd, and Lee (2001) propose a method that uses geometric programming (GP). They require all performance measures to be expressed in a particular functional form, namely a posynomial function. Posynomial functions are then transformed, in their method, to convex functions. Several approximations are needed to express the performance measures in posynomial functions. For example, the phase margin of the two-stage op-amp in Fig. 1 in Hershenson, Boyd, and Lee (2001) can be expressed as a posynomial function only when we approximate $\arctan(x)$ by x (Hershenson, Boyd, and Lee 2001). This approximation does not take into account the secondary effects of the transistors, so it often leads to inaccurate solutions; see, for example, Section IV. Fig. 3(a) of Maji and Mandal (2013) provides another example in which a GP-based method cannot be applied. The circuit topology is so complex that the performance measures are not likely to be posynomial. As evidence for this speculation, Maji and Mandal (2013) report significant errors when the performance measures are approximated using posynomial functions. On the other hand, there are cases where no apparent posynomial-type equations are available for the performance measures. For example, the low–frequency positive power supply rejection ratio in Equation (42) of Hershenson, Boyd, and Lee (2001) cannot be expressed as a posynomial function. In such a case, one can attempt to fit posynomial functions after generating data points on the performance measures using SPICE simulation. However, one often finds difficulties in fitting the performance measures into posynomial functions when the data set shows non-posynomiality. One of the motivations of this paper was due to an inquiry from an engineer in the semiconductor industry who had difficulties fitting performance measures into posynomial functions.

Our proposed method uses a circuit simulation program rather than an equation in order to evaluate the performance measures; thus, it falls into the third category. Many algorithms developed within this category fail to use the smoothness and the unimodality of the performance measures. Instead, they use global search methods such as simulated annealing, a generic algorithm, and a pattern search (Medeiro, Fernandez, Dominguez-Castro, and Rodriguez-Vazquez 1994, Torralba, Chavez, and L.G.Franquelo 1996, Krasnicki, Phelps, Rutenbar, and Carley 1999, Kruiskamp and Leenaerts 1995, Phelps, Krasnicki, Rutenbar, Carley, and Hellums 2000, Cohn, Garrod, Rutenbar, and Carley 1991, Zhang and Kleine 2004, Koza, Bennett, Andre, Keane, and Dunlap 1997). These global search methods often show prohibitively slow convergence. Nye, Riley, Sangiovanni-Vincentelli, and Tits (1988) focus on a local search and proposes a gradient-type algorithm, but they employ a line search method as a subroutine of their procedure, which can significantly slow down the speed of the algorithm.

To our knowledge, this is the first paper that suggests a Lagrangian-type gradient-based method that uses electrical circuit simulator. The proposed method produces more accurate solutions than equation-based

methods and enjoys faster convergence than other simulation-based methods. Its performance is illustrated through experiments in Section 3.

This paper is organized as follows. Section 2 describes our proposed method in greater detail. We apply the proposed method to a one-stage op-amp and compare the proposed method to a GP-based method in Section 3.

2 THE PROPOSED METHOD

In this section, we describe the details of the proposed method. The proposed method considers the Lagrangian function $L : \mathbb{R}^d \times \mathbb{R}^r_+ \to \mathbb{R}$ defined by

$$L(x,\lambda) \quad = \quad f(x) + \lambda_1 g_1(x) + \cdots + \lambda_r g_r(x)$$

for $x \in \mathbb{R}^d$ and $\lambda = (\lambda_1, \ldots, \lambda_r) \in \mathbb{R}^r_+$. We observe that the solution to the following maximin problem:

$$\max_{\lambda \in \mathbb{R}^r_+} \min_{x \in \mathbb{R}^d} -L(x,\lambda) \tag{2}$$

is always a solution to (1); see Theorem 2.18 on page 48 of (Zangwill 1969) for details. Inspired by this, we search for the solution to (2) by using the following iterative procedure. We start from an initial solution $(x^1, \lambda^1) \in \mathbb{R}^d \times \mathbb{R}^r_+$ and recursively generate (x^n, λ^n) for $n \geq 1$ using the equations

$$x^{n+1} \quad = \quad x^n + cL_x(x^n, \lambda^n) \tag{3}$$

$$\lambda^{n+1} \quad = \quad \max(\theta, \lambda^n - cL_\lambda(x^n, \lambda^n)) \tag{4}$$

for some positive real number c, where $\theta = (0, \ldots, 0) \in \mathbb{R}^r$, $\max(a,b) = (\max(a_1,b_1), \ldots, \max(a_r,b_r))$ for $a = (a_1, \ldots, a_r)$ and $b = (b_1, \ldots, b_r) \in \mathbb{R}^r$, $L_x(x,\lambda) = \nabla f(x) + \lambda_1 \nabla g_1(x) + \cdots + \lambda_r \nabla g_r(x)$ is the gradient of L with respect to x and $L_\lambda = (g_1(x), \ldots, g_r(x))$ is the gradient of L with respect to λ. Equations (3) and (4) can be rewritten as follows:

$$x^{n+1} \quad = \quad x^n + c\,(\nabla f(x^n) + \lambda_1^n \nabla g_1(x^n) + \cdots + \lambda_r^n \nabla g_r(x^n)) \tag{5}$$

$$\lambda_i^{n+1} \quad = \quad \max(0, \lambda_i^n - c g_i(x^n)) \text{ for } 1 \leq i \leq r, \tag{6}$$

where $\lambda^n = (\lambda_1^n, \ldots, \lambda_r^n)$. Because there are no closed-form formulas for f and the g_is, we can only evaluate them through a circuit simulation program, such as SPICE. Thus, we estimate ∇f and ∇g_i numerically using finite differences, as follows. We call a circuit simulation program and evaluate f and g_i at x^n and $x^n + \delta^n e_i$ for $1 \leq i \leq d$, where $(\delta^n : n \geq 1)$ is a sequence of positive real numbers converging to zero as $n \to \infty$ and $e_i \in \mathbb{R}^d$ is a vector of zeros except for 1 in the ith entry for $1 \leq i \leq d$. The ith entry of $\nabla f(x^n)$ is estimated by

$$\frac{f(x^n + \delta^n e_i) - f(x^n)}{\delta^n}$$

for $1 \leq i \leq d$. For $1 \leq k \leq r$, the ith entry of $\nabla g_k(x^n)$ is estimated by

$$\frac{g_k(x^n + \delta^n e_i) - g_k(x^n)}{\delta^n}$$

for $1 \leq i \leq d$. Thus, the following recursions are used

$$x_i^{n+1} = x_i^n + c\left(\frac{f(x^n + \delta^n e_i) - f(x^n)}{\delta^n} + \lambda_1^n \frac{g_1(x^n + \delta^n e_i) - g_1(x^n)}{\delta^n} + \cdots + \lambda_r^n \frac{g_r(x^n + \delta^n e_i) - g_r(x^n)}{\delta^n}\right) \tag{7}$$

$$\lambda_i^{n+1} = \max(0, \lambda_i^n - c g_i(x^n)) \tag{8}$$

for $1 \leq i \leq d$ instead of equations (5) and (6).

The proposed method can be summarized as follows:

Proposed Algorithm
Step 1. Set $n = 1$ and select an initial solution (x^1, λ^1) from $\mathbb{R}^d \times \mathbb{R}^r_+$.
Step 2. Call a circuit simulation program, such as SPICE, and evaluate f and g_i at x^n and at $x^n + \delta^n e_i$ for $1 \le i \le d$.
Step 3. Update (x^{n+1}, λ^{n+1}) using equations (7) and (8).
Step 4. Increase n by 1 and repeat Steps 2 and 3 until some stopping criterion is satisfied.

3 NUMERICAL EXPERIMENTS WITH A ONE-STAGE OP-AMP

In this section, we investigate the performance of the proposed method in a one-stage op-amp. To compare the performance of the proposed method to other algorithms, we implement a GP-based method and compare the performance of the proposed method with that of the GP-based method.

The proposed algorithm is implemented in version 5.8.8 of the Perl programming language (Perl 5.8.8, available at http://www.perl.org/) with the help of HSPICE (HSPICE ver H-2013.03-SP1, available at http://www.synopsys.com/), a commercial circuit and device-level simulator. New values of the decision variables are computed from the algorithm in the perl script and a circuit netlist is generated based on the new values. HSPICE is then called to simulate the circuit, it returns the measured results, and the algorithm uses the results to update the decision variables for the next iteration in the optimization procedure. All of the simulations are conducted on a 64-bit Linux machine with an Intel XEON 2.66 GHz, Qua-Core system, and a 12 GB main memory.

In the GP-based method, equations are derived for f and g_i in the Appendix. The equations for f and the g_is are used to solve (1) with CVX, a package for specifying and solving convex programs (Grant and Boyd 2008, Grant and Boyd 2014).

We consider the one-stage op–amp in Fig. 2. Our goal is to determine the width of transistors one and two (M1 and M2) and the width of transistors three and four (M3 and M4) that maximize the open-loop gain while achieving a satisfactory level of the 3-dB bandwidth and an acceptable range of the phase margin. We set the decision variables as follows:

$$\begin{aligned} x_1 &= \text{width of transistors one and two,} \\ x_2 &= \text{width of transistors three and four,} \end{aligned}$$

and denote the open-loop gain, the 3-dB bandwidth, and the phase margin by $g(x_1,x_2)$, $b(x_1,x_2)$, and $m(x_1,x_2)$, respectively. The problem is then formulated as follows:

$$\begin{aligned} \text{maximize} \quad & g(x_1,x_2) \\ \text{subject to} \quad & b(x_1,x_2) \ge b_l \\ & m(x_1,x_2) \ge m_l \end{aligned} \tag{9}$$

for some lower limit b_l on the 3–dB bandwidth and a lower limit m_l on the phase margin. In the op-amps, the 3-dB bandwidth or the cutoff frequency is defined as the frequency range over which the gain is above 70.7 % or -3 dB of its maximum value. The phase margin is a widely used parameter to validate the stability of feedback systems. A phase margin greater than or equal to 45° is required for stable responses.

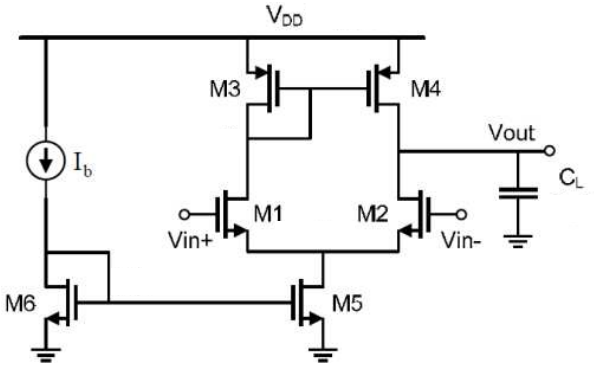

Figure 2: One-stage op-amp

Proposed Approach By changing variables through $v_1 = \log x_1$ and $v_2 = \log x_2$, we obtain the following formulation equivalent to (9):

$$\begin{aligned} \max_{v_1, v_2 \in \mathbb{R}} \quad & \log g(e^{v_1}, e^{v_2}) \\ \text{subject to} \quad & \log b(e^{v_1}, e^{v_2}) \geq \log b_l \\ & \log m(e^{v_1}, e^{v_2}) \geq \log m_l. \end{aligned} \tag{10}$$

We apply the proposed method to solve (10) as follows: We start from an initial solution $(v_1^1, v_2^1, \lambda_1^1, \lambda_2^1)$ and recursively generate $(v_1^n, v_2^n, \lambda_1^n, \lambda_2^n)$ for $n \geq 1$ using the equations

$$\begin{aligned} v_1^{n+1} &= v_1^n + c\left(\frac{\log g(e^{v_1^n+\delta^n}, e^{v_2^n}) - \log g(e^{v_1^n}, e^{v_2^n})}{\delta^n} + \lambda_1^n \frac{\log b(e^{v_1^n+\delta^n}, e^{v_2^n}) - \log b(e^{v_1^n}, e^{v_2^n})}{\delta^n}\right. \\ &\quad \left. + \lambda_2^n \frac{\log m(e^{v_1^n+\delta^n}, e^{v_2^n}) - \log m(e^{v_1^n}, e^{v_2^n})}{\delta^n}\right) \\ v_2^{n+1} &= v_2^n + c\left(\frac{\log g(e^{v_1^n}, e^{v_2^n+\delta^n}) - \log g(e^{v_1^n}, e^{v_2^n})}{\delta^n} + \lambda_1^n \frac{\log b(e^{v_1^n}, e^{v_2^n+\delta^n}) - \log b(e^{v_1^n}, e^{v_2^n})}{\delta^n}\right. \\ &\quad \left. + \lambda_2^n \frac{\log m(e^{v_1^n}, e^{v_2^n+\delta^n}) - \log m(e^{v_1^n}, e^{v_2^n})}{\delta^n}\right) \\ \lambda_1^{n+1} &= \max(0, \lambda_1^n - c(\log b(e^{v_1^n}, e^{v_2^n}) - \log b_l)) \\ \lambda_2^{n+1} &= \max(0, \lambda_2^n - c(\log m(e^{v_1^n}, e^{v_2^n}) - \log m_l)) \end{aligned}$$

for some positive constant c and a sequence of positive real numbers $(\delta^n : n \geq 1)$, where the values of g, b, and m are obtained from HSPICE using a commercial 65-nm technology and open-loop AC simulation.

Fig. 3 shows x_1^n, x_2^n, $g(x_1^n, x_2^n)$, $b(x_1^n, x_2^n)$, and $m(x_1^n, x_2^n)$ for $1 \leq n \leq 300$. The parameters we used in this experiment are: $V_{DD} = 1.2V$, $I_b = 80\,\mu A$, $C_L = 0.1\,pF$, the lengths of transistors 1, 2, 3, and 4 are $0.1\,\mu m$, the lengths of transistors 5 and 6 (M5 and M6) are $1\,\mu m$, the width of transistor 5 is $150\,\mu m$, and the width of transistor 6 is $15\,\mu m$. The parameters we used in the algorithm are: $c = 0.4$, $\delta^n = 0.01/\lceil n/100 \rceil$ ($\lceil x \rceil$ is the smallest integer greater than or equal to $x \in \mathbb{R}$), $b_l = 700\,\text{MHz}$, and $m_l = 45^\circ$. The initial values are $x_1^1 = 10\,\mu m$, $x_2^1 = 10\,\mu m$, $\lambda_1^1 = 1$, and $\lambda_2^1 = 1$. The algorithm shows convergence to the optimal point of (9) by increasing the open-loop gain while forcing the constraints into satisfaction as the iteration progresses. The optimal solution is estimated by averaging the last 100 iterates that the proposed method produces. The optimal solution estimated from the first 1,000 iterations is $(x_1^*, x_2^*) = (52.5\,\mu m, 13.3\,\mu m)$. At this solution,

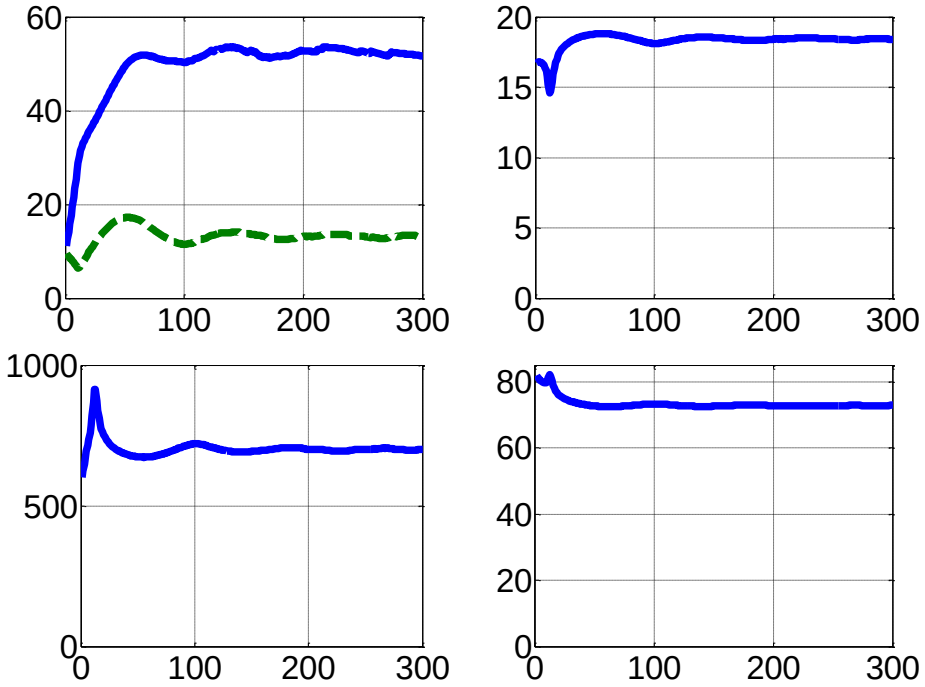

Figure 3: In the first graph, the solid line is x_1^n (μm) and the dashed line is x_2^n (μm). In the second graph, the solid line is the gain (dB). In the third graph, the solid line is the 3-dB bandwidth (MHz). In the fourth graph, the solid line is the phase margin (°). In all graphs, the horizontal axis is the number of iterations.

the gain, the 3–dB bandwidth, the phase margin, and the power consumption are 18.4 dB, 700 MHz, 72.7°, and 965 μW, respectively. The results are summarized in Table 1. The time required to run 200 iterations is 4.2 minutes.

Comparison to GP Following the detailed derivation given in the Appendix, problem (9) can be approximately formulated as the following GP:

$$\begin{aligned} \max_{x_1,x_2} \quad & 3.1x_1^{0.5} \\ \text{subject to} \quad & 790\,\text{MHz} \geq 700\,\text{MHz} \\ & \pi - \pi/2 \\ & -0.00553x_1^{0.5}x_2^{-0.5}\left(0.41x_1 + 2.13x_2\right. \\ & +(1+1/(3.1x_1^{0.5}))0.072x_1 \\ & \left.+(1+2.7x_2^{0.5})0.08x_2\right) \\ & \geq \pi/4. \end{aligned}$$

We solve the above GP using CVX in the geometric programming mode, and compute the following solution: $(x_1^*, x_2^*) = (60.2\,\mu m, 8.4\,\mu m)$. At this solution, the open-loop gain, the 3–dB bandwidth, the phase margin, and the power consumption computed from HSPICE are 16.6 dB, 818 MHz, 74.9°, and 957 μW, respectively. The results are summarized in Table 1.

Table 1 summarizes the results obtained from the proposed method and the GP method. The proposed method achieves a 1.8 dB higher open-loop gain. While the GP method cannot not fully spend the margin of the 3–dB bandwidth to obtain a higher open-loop gain, the proposed method efficiently maximizes the open-loop gain by exactly meeting the boundary condition of the 3–dB bandwidth, 700 MHz. Also, the

proposed method achieves an approximately 5% higher gain-bandwidth product (GBP), a major figure of merit of op-amps, than the GP method.

Table 1: Optimal values obtained from the proposed method and GP for the one-stage op-amp in Fig. 2

Description	Proposed Method	GP
Open-loop gain (dB)	18.4	16.6
3–dB bandwidth (MHz)	700	818
Phase margin (°)	72.7	74.9
Power consumption (μW)	965	957
Width of transistors 1 and 2 (μm)	52.5	60.2
Width of transistors 3 and 4 (μm)	13.3	8.4

4 CONCLUSIONS

In this paper, we proposed to use a Lagrangian-type method to determine the sizes of components in analog circuits. We applied the proposed method to a one-stage op-amp and observed that our method outperforms a GP method. Future work includes exploring the performance of the proposed method in a variety of analog circuits.

A DERIVATION OF A GP FORMULATION FOR ONE-STAGE OP-AMP

In this section, we derive a GP formulation for (9) in Section IV, A. We need to express the gain, the 3-dB bandwidth, and the phase margin in terms of x_1 and x_2. We use the following parameters in Appendices A and B:

$$\mu_n \text{ (electron mobility)} = 0.022\, m^2/(V \cdot s),$$
$$\mu_p \text{ (hole mobility)} = 0.018\, m^2/(V \cdot s),$$
$$C_{ox} \text{ (oxide capacitance per unit area)}$$
$$= \begin{cases} C_{oxn} = 13.3\, fF/(\mu m)^2 \text{ for NMOS,} \\ C_{oxp} = 12.3\, fF/(\mu m)^2 \text{ for PMOS,} \end{cases}$$
$$\lambda_n \text{ (NMOS channel-length modulation parameter)}$$
$$= 0.76 V^{-1},$$
$$\lambda_p \text{ (PMOS channel-length modulation parameter)}$$
$$= 0.48 V^{-1},$$
$$|V_{GS} - V_{TH}| \text{ (overdrive voltage)}$$
$$= 0.13\, V \text{ for both NMOS and PMOS,}$$
$$L_D \text{ (source/drain lateral diffusion length)}$$
$$= \begin{cases} L_{Dn} = 5.4\, nm \text{ for NMOS,} \\ L_{Dp} = 6.55\, nm \text{ for PMOS,} \end{cases}$$
$$C_{db} = \begin{cases} C_{dbn} = 0.41\, fF/\mu m \text{ for NMOS,} \\ C_{dbp} = 0.31\, fF/\mu m \text{ for PMOS.} \end{cases}$$

We use W_i and L_i to denote the width and the length of transistor i, respectively, for $1 \leq i \leq 6$. The following quantities are fixed for the rest of the computation:

$$L_1 = L_2 = L_3 = L_4 = 0.1\,\mu m,$$
$$L_5 = L_6 = 1\,\mu m,\ W_5 = 150\,\mu m,\ W_6 = 15\,\mu m,\ I_b = 80\,\mu A,\ C_L = 100\,fF.$$

Also, I_2 can be computed as follows.

$$I_1 = I_2 = \frac{W_5 L_6}{2L_5 W_6} I_b = \frac{(150)(1)}{(2)(1)(15)} 80\,\mu A = 400\,\mu A.$$

The open-loop voltage gain is given by

$$A_v = \frac{1}{(\lambda_n + \lambda_p)} \sqrt{\frac{2\mu_n C_{oxn} W_2}{L_2 I_2}} \approx 3.1 x_1^{0.5}. \tag{11}$$

The 3–dB bandwidth is given by

$$f_{3dB} \quad = \quad \frac{g_{m1}}{2\pi A_v C_L}, \tag{12}$$

where

$$g_{m1} \quad = \quad \sqrt{2\mu_n C_{oxn} \frac{W_1}{L_1} I_1}. \tag{13}$$

From (11), (12), and (13), the 3–dB bandwidth can be computed as

$$\frac{\sqrt{I_1 I_2}}{2\pi C_L} (\lambda_n + \lambda_p) \sqrt{\frac{W_1 L_2}{L_1 W_2}} = 790 \text{ MHz}. \tag{14}$$

On the other hand, the phase margin is given by

$$\begin{aligned} &\pi + \arctan\left(\frac{C_M}{2C_L}\sqrt{\frac{\mu_n L_3 W_1}{\mu_p L_1 W_3}}\right) \\ &\quad - \arctan\left(\frac{1}{(\lambda_n + \lambda_p)}\sqrt{\frac{2\mu_n C_{oxn} W_2}{L_2 I_2}}\right) \\ &\quad - \arctan\left(\frac{C_M}{C_L}\sqrt{\frac{\mu_n L_3 W_1}{\mu_p L_1 W_3}}\right), \end{aligned} \tag{15}$$

where

$$\begin{aligned} C_M &= C_{db1} + C_{db3} + C_{gs3} + C_{gs4} + \\ &\quad (1 + 1/Av_1) C_{gd1} + (1 + Av_2) C_{gd4}, \end{aligned} \tag{16}$$

$$\begin{aligned} C_{gsi} &= (2/3) W_i L_i C_{ox} + W_i L_D C_{ox} \\ &\quad \text{for transistor } i, 1 \leq i \leq 6, \end{aligned} \tag{17}$$

$$C_{gdi} = W_i L_D C_{ox} \text{ for transistor } i, 1 \leq i \leq 6, \tag{18}$$

$$C_{dbi} = C_{db} W_i \text{ for transistor } i, 1 \leq i \leq 6, \tag{19}$$

$$Av_1 = \frac{1}{(\lambda_n + \lambda_p)} \sqrt{\frac{2\mu_n C_{oxn} W_2}{L_2 I_2}}, \tag{20}$$

$$Av_2 = \frac{1}{(\lambda_n + \lambda_p)} \sqrt{\frac{2\mu_p C_{oxp} W_4}{L_4 I_2}}. \tag{21}$$

In equations (17), (18), and (19), we use the following values:

$$C_{ox} = \begin{cases} C_{oxn} \text{ if transistor } i \text{ is NMOS,} \\ C_{oxp} \text{ if transistor } i \text{ is PMOS,} \end{cases}$$
$$L_D = \begin{cases} L_{Dn} \text{ if transistor } i \text{ is NMOS,} \\ L_{Dp} \text{ if transistor } i \text{ is PMOS,} \end{cases}$$
$$C_{db} = \begin{cases} C_{dbn} \text{ if transistor } i \text{ is NMOS,} \\ C_{dbp} \text{ if transistor } i \text{ is PMOS} \end{cases}$$

for $1 \leq i \leq 6$. Since

$$\arctan\left(\frac{1}{(\lambda_n+\lambda_p)}\sqrt{\frac{2\mu_n C_{oxn} W_2}{L_2 I_2}}\right) = \arctan(A_v),$$

when the open-loop voltage gain A_v is high, we can use the following approximation:

$$\begin{aligned} &\arctan\left(\frac{1}{(\lambda_n+\lambda_p)}\sqrt{\frac{2\mu_n C_{oxn} W_2}{L_2 I_2}}\right) \\ &= \quad \arctan(A_v) \approx \pi/2. \end{aligned} \tag{22}$$

Approximating $\arctan(x)$ by x and using equations (15) through (22), the phase margin can be approximated by

$$\begin{aligned} &\pi - \pi/2 - 0.00553 x_1^{0.5} x_2^{-0.5} \left(0.41 x_1 + 2.13 x_2 \right. \\ &\left. +(1+1/(3.1x_1^{0.5}))0.072 x_1 + (1+2.7x_2^{0.5})0.08 x_2\right). \end{aligned} \tag{23}$$

From equations (11), (14), and (23), problem (9) can be approximately formulated as the following GP:

$$\begin{aligned} \max_{x_1,x_2} \quad & 3.1 x_1^{0.5} \\ \text{subject to} \quad & 790\,\text{MHz} \geq 700\,\text{MHz} \\ & \pi - \pi/2 \\ & -0.00553 x_1^{0.5} x_2^{-0.5} \left(0.41 x_1 + 2.13 x_2 \right. \\ & +(1+1/(3.1x_1^{0.5}))0.072 x_1 \\ & \left. +(1+2.7x_2^{0.5})0.08 x_2\right) \\ & \geq \pi/4. \end{aligned}$$

REFERENCES

Antoa, B. A. A., and A. J. Brodersen. 1995. "ARCHGEN: Automated synthesis of analog systems". *IEEE Transactions on VLSI Systems* 3 (2): 231–244.

Arrow, K. J., L. Hurwicz, and H. Uzawa. 1958. *Studies in linear and non-linear programming*. Stanford, CA: Stanford University Press.

Beenker, G., J. Conway, G. Schrooten, and A. Slenter. 1993. *Analog CAD for consumer ICs*. Norwell, MA: Kluwer.

Boyd, S. P., S.-J. Kim, D. D. Patil, and M. A. Horowitz. 2005. "Digital Circuit Optimization via Geometric Programming". *Operations Research* 53 (6): 899–932.

Chen, C.-P., C. C. N. Chu, and D. F. Wong. 1999. "Fast and Exact Simultaneous Gate and Wire Sizing by Lagrangian Relaxation". *IEEE Transactions on Computer-Aided Design of Intergated Circuits and Systems* 18 (7): 1014–1025.

Cohn, J., D. Garrod, R. Rutenbar, and L. R. Carley. 1991. "KOAN/ANAGRAMII: New tools for device-level analog placement and routing". *IEEE Journal of Solid-State Circuits* 26:330–342.

Degrauwe, G. R. 1987. "IDAC: An interactive design tool for analog CMOS circuits". *IEEE Journal of Solid-State Circuits* 22:1106–1116.

El-Turky, F., and E. Perry. 1989. "BLADES: An artificial intelligence approach to analog circuit design". *IEEE Transactions on Computer-Aided Design* 8:680–692.

Fernández-Fernandéz, F., A. Rodríguez-Vázquez, J. L. Huertas, and G. Gielen.. 1998. *Symbolic analysis techniques: Applications to Analog Design Automation.* IEEE Press.

Gielen, G., G. Debyser, K. Lampaert, F. Leyn, K. Swings, G. d. V. Plas, W. Sansen, D. Leenaerts, P. Veselinovic, and W. v. Bokhoven. 1995. "An analog module generator for mixed analog/digital ASIC design". *International Journal of Circuit Theory and Applications* 23:269–283.

Grant, M., and S. Boyd. 2008. "Graph implementations for nonsmooth convex programs". In *Recent Advances in Learning and Control*, edited by V. Blondel, S. Boyd, and H. Kimura, Lecture Notes in Control and Information Sciences, 95–110. Springer-Verlag Limited. http://stanford.edu/$\sim$boyd.

M. Grant and S. Boyd 2014, March. "CVX: Matlab Software for Disciplined Convex Programming, version 2.1". http://cvxr.com/cvx.

Harjani, R., R. Rutenbar, and L. R. Carley. 1989. "OASYS: A framework for analog circuit synthesis". *IEEE Transactions on Computer-Aided Design* 8:1247–1265.

Harvey, J. P., M. I. Elmasry, and B. Leung. 1992. "STAIC: An interactive framework for synthesizing CMOS and BICMOS analog circuits". *IEEE Transactions on Computer-Aided Design* 11 (11): 1402–1417.

Hershenson, M., S. Boyd, and T. Lee. 1998. "GPCAD: A tool for CMOS op-amp synthesis". In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 296–303.

Hershenson, M. d. M., S. P. Boyd, and T. H. Lee. 2001. "Optimal Design of a CMOS op-amp via Geometric Programming". *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 20 (1): 1–21.

Horta, N. C., and J. E. Franca. 1996. "High-level data conversion synthesis by symbolic methods". In *Proceedings of the IEEE International Symposium on Circuits and Systems*, 802–805.

Horta, N. C., J. E. Franca, and C. A. Leme. 1991. "Framework for architecture synthesis of data conversion systems employing binary-weighted capacitor arrays". In *Proceedings of the IEEE International Symposium on Circuits and Systems*, 1789–1792.

Jangkrajarng, N., S. Bhattacharya, R. Hartono, and C.-J. R. Shi. 2003. "IPRAIL: Intellectual property reuse based analog IC layout automation". *Integration, the VLSI Journal* 36 (4): 237–262.

Koh, H. Y., C. H. Sequin, and P. R. Gray. 1990. "OPASYN: A compiler for CMOS operational amplifiers". *IEEE Transactions on Computer-Aided Design* 9 (2): 113–125.

Koza, J. R., F. H. Bennett, D. Andre, M. A. Keane, and F. Dunlap. 1997. "Automated synthesis of analog electrical circuits by means of genetic programming". *IEEE Transactions on Evolutionary Computation* 1 (2): 109–128.

Krasnicki, M., R. Phelps, R. Rutenbar, and L. R. Carley. 1999. "MAELSTROM: Efficient simulation-based synthesis for custom analog cells". In *Proceedings of the ACM/IEEE Design Automation Conference (DAC)*, 945–950.

Kruiskamp, W., and D. Leenaerts. 1995. "DARWIN: CMOS opamp synthesis by means of a genetic algorithm". In *Proceedings of the ACM/IEEE Design Automation Conference (DAC)*, 550–553.

Maji, S., and P. Mandal. 2013. "Effcient approaches to overcome non-convexity issues in analog design automation". In *Proceedings of the 13th International Symposium on Quality Electronic Design*, 566–571.

Maulik, P. C., and L. R. Carley. 1991. "Automating analog circuit design using constrained optimization techniques". In *Proceedings of the IEEE International Conference on Computer-Aided Design*, 390–393.

Medeiro, F., F. V. Fernandez, R. Dominguez-Castro, and A. Rodriguez-Vazquez. 1994. "A statistical optimization-based approach for automated sizing of analog cells". In *Proceedings of the ACM/IEEE International Conference on Computer-Aided Design (ICCAD)*, 594–597.

Medeiro, F., B. Perez-Verdu, A. Rodriguez -Vazquez, and J. L. Huertas. 1995. "A vertically integrated tool for automated design of modulators". *IEEE Journal of Solid-State Circuits* 30 (7): 762–772.

Nye, W., D. C. Riley, A. Sangiovanni-Vincentelli, and A. L. Tits. 1988. "Delight.spice: An optimization-based system for the design of integrated circuits". *IEEE Transactions on Computer-Aided Design* 7 (4): 501–519.

Ochotta, E. S., R. A. Rutenbar, and L. R. Carley. 1996. "Synthesis of high-performance analog circuits in ASTRX/OBLX". *IEEE Transactions on Computer-Aided Design* 15 (3): 273–294.

Phelps, R., M. Krasnicki, R. Rutenbar, L. R. Carley, and J. Hellums. 2000. "ANACONDA: Simulation-based synthesis of analog circuits via stochastic pattern search". *IEEE Transactions on Computer-Aided Design* 19 (6): 703–717.

Torralba, A., J. Chavez, and L.G.Franquelo. 1996. "FASY: A fuzzy-logic based tool for analog synthesis". *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 15 (7): 705–715.

Zangwill, W. I. 1969. *Nonlinear Programming: A Unified Approach*. NJ: Prentice-Hall, Inc.

Zhang, L., and U. Kleine. 2004. "A novel analog layout synthesis tool". In *Proceedings of the International Symposium on Circuits and Systems*, 101–104.

AUTHOR BIOGRAPHIES

EUNJI LIM is an Assistant Professor in the School of Management and Marketing at Kean University. Her research interest includes simulation-based optimization and stochastic models. Her email address is elim@kean.edu.

YOUNGMIN KIM received the B.S. degree in electrical engineering from the Yonsei University, Seoul, Korea, in 1999, and the M.S. and Ph.D. degrees in electrical engineering from the university of Michigan, Ann Arbor, in 2003 and 2007, respectively. He has held a senior engineer position in the Qualcomm, San Diego, CA. He is currently an Assistant Professor in Kwangwoon University, Seoul, South Korea. Prior to joining Kwangwoon University, he was with the school of electrical engineering and computer engineering at the Ulsan National Institute of Science and Technology (UNIST), Ulsan, South Korea. His research interests include variability-aware design methodologies, design for manufacturability, and design and technology co-optimization methodologies, and low-power and 3D IC designs.

JAEHYOUK CHOI received the B.S. degree (summa cum laude) in electrical engineering from Seoul National University, Seoul, Korea, in 2003, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2008 and 2010, respectively. From 2010 to 2011, he was with Qualcomm, Inc., San Diego, CA, where he was involved in designing multi-standard cellular transceivers. In 2012, he joined Ulsan National Institute of Science and Technology (UNIST), Ulsan, Korea, where he is currently an Assistant Professor of electrical and computer engineering. His research interests include low power and high performance analog, mixed signal, and RF integrated circuits for emerging wireless/wired standards.