Convexity-based Algorithms for Design Centering

Sachin S. Sapatnekar† Pravin M. Vaidya‡ Steve M. Kang§

†Dept. of EE & CprE
Iowa State University
Ames, IA 50011

‡Department of Computer Science
University of Illinois
Urbana, IL 61801

§CSL & Department of ECE
University of Illinois
Urbana, IL 61801

Abstract

Two new geometrical algorithms for design centering are presented. In each case, the feasible region is first approximated by a convex polytope. The first algorithm inscribes the largest Hessian ellipsoid within the approximating polytope. The second assumes Gaussian parameter distributions, formulates the problem as a convex programming problem, and uses an efficient algorithm to perform the optimization.

1 Introduction

While manufacturing a circuit, it is inevitable that process variations will cause design parameters, such as component values, to vary from their nominal values. As a result, the manufactured circuit may not meet some behavioral specifications, such as requirements on the delay, gain and bandwidth, that it has been designed to satisfy. The procedure of design centering attempts to select the nominal values of design parameters so as to ensure that the behavior of the circuit remains within specifications, with the greatest probability. Previous approaches to solving this problem have traditionally taken two routes : the *statistical* approach, which primarily involves Monte Carlo techniques, and the *geometrical* approach. This work presents a new geometrical approach.

Geometric techniques frequently assume that the feasible region is convex and bounded. These methods approximate the feasible region by a known geometric body, such as a polytope or an ellipsoid, whose center is taken to be the design center. In the case of ellipsoidal approximation, as in [1], the inherently symmetric shape of ellipsoids prohibits them from accurately approximating convex bodies that are "less symmetric." A polytope can provide a better approximation to a convex body than an ellipsoid, since any convex body can be thought of as a polytope with an infinite number of faces. However, finding the exact center of a polytope is computationally difficult [2].

In this work, the feasible region is first approximated by a convex polytope using the procedure described in [3]. In the next phase, the design center may be found using one of two approaches.

The first approach, called the largest Hessian ellipsoid (LHE) method, inscribes the largest Hessian ellipsoid, defined in Section 3, inside the polytope. The shape of the Hessian ellipsoid is well-known to be representative of the shape of the polytope [2]. This method presents an improvement over [4], in which the center of either a hypersphere, or a crudely chosen ellipsoid, is taken to be the design center.

The second approach, the convex programming (CP) approach, proceeds by formulating the design centering problem as a convex programming problem, assuming that the variations in the design parameters are modeled by Gaussian probability distributions. A convex programming algorithm [2], whose efficacy has been illustrated, for example, in [5], is used to find the solution to this problem. As shown by an example in Section 4, the design center is sensitive to the parameter probability distributions. This method recognizes this, and explicitly uses the available probability distribution information. In doing so, it differs from other geometrical methods, such as [1] and [4].

2 Feasible Region Approximation

The feasible region $\mathcal{F} \subset \mathbf{R}^n$, where n is the number of design parameters, is defined as the set of points in the design parameter space for which the circuit satisfies all specifications on its behavior. We persevere with the common assumption made by geometrical design centering algorithms is that $\mathcal{F}$ is a bounded convex body, and use some properties of convex sets to approximate $\mathcal{F}$.

The details of the algorithm are described in [3]. It is based on the supporting hyperplane theorem of convex sets, and approximates a convex feasible region, $\mathcal{F} \subset \mathbf{R}^n$ by a polytope

$$\mathcal{P} = \{\mathbf{x} \mid A\mathbf{z} \geq \mathbf{b}\}, A \in \mathbf{R}^{m \times n}, \mathbf{b} \in \mathbf{R}^m, \qquad (1)$$

formed by the intersection of m half-spaces in $\mathbf{R}^n$. It requires an initial feasible point to be specified, and thereafter requires simulations to check whether a point is feasible or not, and constraint function gradient evaluations.

3 Finding the Design Center

The random variations in the values of the design parameters are modeled by a probability density function, $\Phi(\mathbf{z}) : \mathbf{R}^n \rightarrow [0, 1]$, with a mean corresponding to the nominal value of the design parameters. The yield of the circuit, Y, as a function of the mean, $\mathbf{x}$, is given by

$$\mathrm{Y}(\mathbf{x}) = \int_{\mathcal{F}} \Phi_{\mathbf{X}}(\mathbf{z}) d\mathbf{z} \qquad (2)$$

where $\mathcal{F}$ represents the feasible region, where the design parameters are such that the circuit satisfies its behavioral requirements. The *design center* is the point $\mathbf{x}$ at which the yield, $\mathrm{Y}(\mathbf{x})$, is maximized.

The simplicial approximation method [4] does not attempt to evaluate the above integral. Instead, it approximates the feasible region by a polytope and attempts to inscribe the largest hypersphere within the approximating polytope, taking its center as the design center. For elongated regions of acceptability, such as rectangles, a more realistic center would be obtained by inscribing the largest ellipsoid inside the polytope. In [4], a very approximate method is used, in which only a limited set of ellipsoids (whose major axes are along the coordinate directions) is considered. Therefore, a better method of determining the shape of the largest inscribed ellipsoid is required; this is provided by our LHE method.

3.1 Largest Hessian Ellipsoid Algorithm

Consider a polytope P defined by (1), and let $\mathbf{a_i}^T$ be the i^{th} row of matrix $A \in \mathbf{R}^{m \times n}$, and b_i be the i^{th} element of $\mathbf{b} \in \mathbf{R}^m$. The approximate polytope center $\mathbf{x}_c$, is taken to be the point that minimizes the *log-barrier function*

$$F(\mathbf{x}) = -\sum_{i=1}^{m} \log_e(\mathbf{a_i}^T \mathbf{x} - b_i). \tag{3}$$

Note that near the boundary of the polytope, $F(\mathbf{x})$ tends to infinity and its value decreases as one moves *deeper* into the interior of the polytope. The value of $F(\mathbf{x})$ is undefined outside the boundary of the polytope. F is a convex function of $\mathbf{x} \in P$, with an $n \times n$ Hessian matrix (for further details, see [5,8])

$$\mathcal{H}(\mathbf{x}) = \nabla^2 F(\mathbf{x}) = \sum_{i=1}^{m} \frac{\mathbf{a_i}\mathbf{a_i}^T}{(\mathbf{a_i}^T \mathbf{x} - b_i)^2}. \tag{4}$$

Let E($\mathbf{x}$,$\mathcal{B}$,r) denote the ellipsoid

$$\left\{ \mathbf{y} \mid (\mathbf{y} - \mathbf{x})^T \mathcal{B} (\mathbf{y} - \mathbf{x}) \leq \mathrm{r}^2 \right\}. \tag{5}$$

The Hessian ellipsoid at a point $\mathbf{x}$ in the polytope P, namely the ellipsoid E($\mathbf{x}$,$\mathcal{H}(\mathbf{x})$,r), where $\mathcal{H}(\mathbf{x})$ is the Hessian of the log-barrier function (Equation (3)) is a good approximation to the polytope locally around $\mathbf{x}$ [2]. Hence, the goal is to find the largest ellipsoid in the class E($\mathbf{x}$, $\mathcal{H}(\mathbf{x})$,r) that can be inscribed in the polytope, and its center $\mathbf{x}_c$. The point $\mathbf{x}_c$ will be taken to be the computed design center.

The minimum of the log-barrier function gives a good approximation to $\mathbf{x}_c$, and is used as an initial guess for the iterative process, described below:

distance $= \infty$
$\mathbf{x}_c$ = *Minimizer of the log barrier function (Eq. (3))*
while (distance $> \epsilon$*)* {
  $\mathbf{x}_{old} = \mathbf{x}_c$
  $\mathcal{H}$ = *Hessian at* $\mathbf{x}_c$
  Inscribe the largest ellipsoid of the type E($\mathbf{x}$, $\mathcal{H}(\mathbf{x}_c)$, r*)*
  (Note that the ellipsoid shape, determined by $\mathcal{H}(\mathbf{x}_c)$*, is fixed, and that* $\mathbf{x}$ *and* r *are allowed to vary)*
  Set $\mathbf{x}_c$ = *center of this ellipsoid*
  distance $= \|\mathbf{x}_c - \mathbf{x}_{old}\|$
}

In each iteration, the shape of the ellipsoid is fixed by the Hessian, $\mathcal{H}$, computed at the current value of $\mathbf{x}_c$; $\mathbf{x}$ and r are allowed to vary. Since $\mathcal{H}$ is positive definite and can be Cholesky-decomposed, the process of inscribing an ellipsoid of shape $\mathcal{H}$ can be mapped through a linear transformation to the problem of inscribing a hypersphere in a polytope, as in [4].

3.2 Convex Programming Approach

The LHE algorithm computes a single point that is the design center, regardless of the probability distributions; this ignores the fact that the design center is, in reality, sensitive to probability distributions. The convex programming approach assumes that the probability density functions that represent variations in the design parameters are Gaussian in nature, and poses the design centering problem as a convex programming problem.

The joint Gaussian probability density function of n independent random variables $\mathbf{z} = (z_1, \cdots, z_n)$, where z_i has mean x_i and variance σ_i, is given by

$$\Phi_{\mathbf{x}}(\mathbf{z}) = \frac{1}{(2\pi)^{\frac{n}{2}} \sigma_1 \cdots \sigma_n} \exp\left[\sum_{i=1}^{i=n} -\frac{(z_i - x_i)^2}{2\sigma_i^2} \right] \tag{6}$$

where $\mathbf{x} = (x_1, \cdots, x_n)$. This is a log-concave function of $\mathbf{x}$ and $\mathbf{z}$. (Note that arbitrary (symmetric) covariance matrices can be handled, since a symmetric matrix may be converted into a diagonal form by a simple linear transformation.)

The design centering problem is now formulated as

$$\text{maximize} \qquad \mathrm{Y}(\mathbf{x}) = \textstyle\int_{\mathcal{P}} \Phi_{\mathbf{x}}(\mathbf{z}) d\mathbf{z}. \tag{7}$$

where $\mathcal{P}$ is the polytope approximation to the feasible region $\mathcal{F}$. It is a known fact that the integral of a log-concave function over a convex region is also a log-concave function [6]. Thus, the yield function $\mathrm{Y}(\mathbf{x})$ is log-concave, and the above problem reduces to a problem of maximizing a log-concave function over a convex set. Hence, this can be transformed into a convex programming problem, with the corresponding property that any local minimum of the problem is a global minimum. It is worth noting here that the yield function remains convex as long as $\phi_{\mathbf{x}}(\mathbf{z})$ is any log-concave function of $\mathbf{x}$ and $\mathbf{z}$.

Applying the Convex Programming Algorithm

The convex programming algorithm proposed in [2] provides an efficient technique for solving a convex programming problem, such as (7). If $\mathcal{P}$ is the polytope that approximates the feasible region, we define the *feasible set*

$$S = \{\mathbf{z} \in \mathbf{R}^n \mid \mathbf{z} \in \mathcal{P}\} \tag{8}$$

and let $\mathbf{x}_c$ be the solution to (7). Initially, we set $\mathcal{Q} = \mathcal{P}$; the invariant here is that $\mathcal{Q}$ always contains the design center, $\mathbf{x}_c$. The polytope $\mathcal{Q}$ is given by

$$\mathcal{Q} = \{\mathbf{z} \mid \hat{A}\mathbf{z} \geq \hat{\mathbf{b}}\}, \hat{A} \in \mathbf{R}^{p \times n}, \hat{b} \in \mathbf{R}^p. \tag{9}$$

The algorithm proceeds iteratively as follows. First, a *center* $\mathbf{z}_c$, deep in the interior of the current polytope $\mathcal{Q}$ is found, by minimizing the log-barrier function

$$F(\mathbf{z}) = -\sum_{i=1}^{p} \log_e(\hat{\mathbf{a}}_i^T \mathbf{z} - \hat{b}_i) \tag{10}$$

where $\hat{\mathbf{a}}_i^T$ is the i^{th} row of matrix $\hat{A}$ and $\hat{b}_i$ is the i^{th} element of $\hat{\mathbf{b}}$. The minimization procedure is as in [5].

There exists a hyperplane that divides the polytope into two parts, such that $\mathbf{x}_c$ is contained in one of them, satisfying the constraint

$$\mathbf{c}^T \mathbf{z} \geq \mathbf{c}^T \mathbf{z}_c \qquad \text{with} \quad \mathbf{c} = -[\nabla \mathrm{Y}(\mathbf{x})]^T \tag{11}$$

being the negative of the gradient of the yield (objective) function.

Since the yield function is not available in an explicit form, its gradient is estimated using the yield gradient approximation method presented in [7]. This yield estimator works with the polytope approximation of the feasible region. A point is considered to be feasible if it lies within the approximating polytope; this leads to a substantial savings in computation, since it is much cheaper to find out whether a point lies within a polytope than to simulate a circuit for many sets of parameter values.

In practice, the yield gradient is approximate, and possibly erroneous, as it is based on a Monte Carlo simulation. To offset this problem, the new hyperplane is taken as $\mathbf{c}^T\mathbf{z} \geq \mathbf{c}^T\mathbf{z}_c - \delta$ where δ is small, representing the fact that the plane is moved away by a certain fraction towards the boundary of the current polytope. Qualitatively speaking, the above hyperplane shaves off less from the polytope given by Equation (11), thereby reducing the possibility of errors due to incorrect gradient estimations.

The new constraint is added to the current polytope to give a new polytope, $\mathcal{Q}$, that has roughly half the original volume. The process is repeated until the polytope is sufficiently small, and the final center $\mathbf{z}_c$ is taken to be the computed design center.

4 Experimental Results

We first present an example that illustrates the deficiencies of ellipsoidal-based approaches, such as [1], [4] and our LHE approach. This example provides an exposition of the need for using information on the probability distributions of the design variables. The feasible region, $\mathcal{F}$, illustrated in Fig. 1, is a triangle in $\mathbf{R}^2$, described by the planes $x_1 + x_2 \leq 1, x_1 - x_2 \geq -1$, and $x_2 \geq 0$. Any approach that attempts to inscribe the largest circle or ellipsoid in this region would compute the center of the incircle of the triangle, $[0, 0.414]^T$ as the design center.

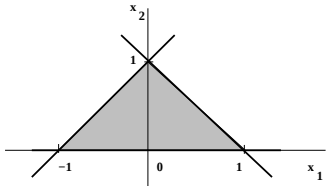

Figure 1: Triangular Feasible Region

The actual design center, however, is a function of the parameter variances, $\sigma_{\mathbf{X}_1}$ and $\sigma_{\mathbf{X}_2}$. Due to symmetry, the $\mathbf{x}_1$ coordinate of the design center is 0 and is insensitive to the variance. However, changes in the variance of the random variables do affect the $\mathbf{x}_2$ coordinate of the design center. As $\sigma_{\mathbf{X}_1}/\sigma_{\mathbf{X}_2} \to 0$, the design center $\to (0, 0.5)$. As $\sigma_{\mathbf{X}_1}/\sigma_{\mathbf{X}_2}$ increases, the design center starts moving downward along the $\mathbf{x}_2$ axis. When $\sigma_{\mathbf{X}_1}/\sigma_{\mathbf{X}_2} = 1$, the design center is the centroid. As $\sigma_{\mathbf{X}_1}/\sigma_{\mathbf{X}_2} \to \infty$, the design center $\to (0, 0)$. For actual yield figures for this example, see [8].

These algorithms has been tested on a wide variety of circuits [8], including several filter circuits that are often used as circuit examples for design centering algorithms. For all of these circuits, the feasible region is nonconvex. These two methods provided comparable or better results than [1].Due to space limitations, we only present two circuit examples here, that of a CMOS op amp circuit shown in Fig. 2 [9], and a high-pass filter [1], shown in Fig. 3.

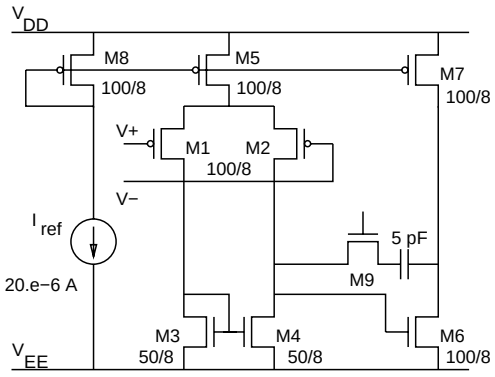

Figure 2: A CMOS operational amplifier.

In Fig. 2, the transistor pairs M1-M2 and M3-M4 are matched. The designable parameters are the widths of M1, M5, and M6. The constraints that define the feasible region for this problem are (i) Gain $\geq$ 98 dB, (ii) Area = sum of widths of M1, M5 and M6 $\leq$ 308, (iii) Bandwidth $\geq 17 \times 10^6$ rad/sec, (iv) Power dissipation $\leq$ 0.65 mW.

TABLE 1 : CMOS OPERATIONAL AMPLIFIER

Variances $[\sigma_{w_1}, \sigma_{w_5}, \sigma_{w_6}]$	CP Design Center in $[\mu m, \mu m, \mu m]$	Yield (500 max)	
		CP	LHE
[3.0, 3.0, 3.0]	[106.7,87.8,108.8]	452	446
[7.0, 7.0, 7.0]	[105.5,93.9,98.7]	252	219
[5.0, 8.0, 1.0]	[108.4,94.7,94.7]	302	281
[5.0, 7.0, 4.0]	[103.5 93.7 101.3]	300	288
[3.0, 4.0, 9.0]	[108.4,94.7,94.7]	288	289
[4.0, 1.0, 4.0]	[106.6,84.9,107.2]	425	411
[3.0, 2.0, 4.0]	[106.5,88.6,102.9]	440	437
[4.0, 2.0, 3.0]	[108.1,88.2,102.0]	445	421

The nominal values of all transistor widths, corresponding to an initial feasible solution, are shown in the figure. We performed the LHE and the CP design centering procedures in the transformed domain, to obtain the results shown in Table 1. The LHE design center was found to be $[101.0\mu m, 94.1\mu m, 103.4\mu m]$. The yield estimates are based on a Monte Carlo simulation of 500 points with the design center corresponding to the mean value. The variances are in terms of percentages of a nominal value that corresponds to the design center found by the LHE. algorithm. For this example, the CP method is almost always better than the LHE method. However, this is not true for all circuits.

The next circuit example is a high-pass filter [1], whose circuit diagram and specifications are shown in Fig. 3. The frequencies of interest are $\{170, 350, 440, 630, 680, 990, 1800\}$ Hz, which correspond to seven constraints.

For this example, the reference frequency, ω_0, is 990 Hz. The design parameters are $[C_1, C_3, C_4, C_5]$. The initial feasible point was taken as [11.1nF, 12.9nF, 34.3nF, 97.3nF], as in Abdel-Malek *et al.*'s ellipsoidal method (A-ME) [1], where the solution was found to be [10.37nF, 13.28nF, 34.63nF, 87.84 nF]. The solution found by the LHE method is the point [9.614nF, 14.148nF, 33.642nF, 99.301nF]. The

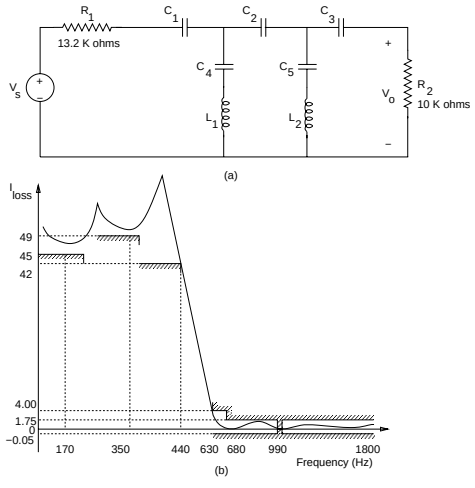

Figure 3: A high-pass filter.

CP method was applied and, as before, the design center was found to change, depending on the variances associated with the p.d.f.'s.

The yield figures for the design centers found by the A-ME method, the LHE method and the CP method are displayed in Table 2. As before, the figures are based on a Monte Carlo simulation of 500 points, with the mean at the calculated center. It can be seen from this table that both the LHE and the CP methods frequently performed better than the A-ME method, and none of these three methods is consistently better than another. However, here the overall performance of LHE is the best of the three.

TABLE 2 : HIGH PASS FILTER (4 PARAMETERS)

Parameter Variations $[\sigma_{C_1}, \sigma_{C_3}, \sigma_{C_4}, \sigma_{C_5}]$	Yield (500 max)		
	LHE	A-ME	CP
[9.0, 9.0, 9.0, 9.0]	439	432	434
[10.0, 10.0, 10.0, 10.0]	412	407	415
[10.0, 15.0, 20.0, 5.0]	262	258	252
[15.0, 20.0, 5.0, 10.0]	432	403	422
[5.0, 8.0, 10.0, 12.0]	423	409	417
[5.0, 8.0, 10.0, 15.0]	411	400	404
[5.0, 10.0, 10.0, 15.0]	403	392	402
[5.0, 10.0, 12.0, 15.0]	365	365	364
[8.0, 12.0, 10.0, 12.0]	409	390	403
[8.0, 12.0, 10.0, 10.0]	411	399	416
[15.0, 12.0, 10.0, 10.0]	398	387	395
[8.0, 12.0, 12.0, 10.0]	373	360	369

The second experiment took $[C_1, C_2, C_3, C_4, C_5, L_1, L_2]$ as the design parameters. The initial feasible point was taken, as in [1], as [11.65 nF,10.47 nF,13.99 nF,39.93 nF,99.4 nF,3.988 H,2.685 H]. A-ME's design center was found to be [12.76 nF,10.37 nF,11.88 nF,40.26 nF,117.37 nF,3.609 H,2.504 H]. The solution obtained by the LHE method was [12.68 nF,8.775 nF,12.68 nF,30.93 nF,93.17 nF,4.623 H,2.748 H]. A comparison of results for this experiment from the LHE, CP and A.M.-E. methods, for various values of component variances (as percentages of the LHE design center), is shown in Table 3. Both the CP and LHE methods provide better solutions than A-ME for almost all cases here.

TABLE 3 : HIGH PASS FILTER (7 PARAMETERS)

Parameter Variations $[\sigma_{C_1}, \cdots, \sigma_{C_5}, \sigma_{L_1}, \sigma_{L_2}]$	Yield (500 max)		
	LHE	A-ME	CP
[10, 10, 10, 10, 5, 5, 5]	243	244	251
[10, 10, 10, 10, 10, 10, 10]	159	149	155
[8, 8, 8, 8, 8, 8, 8]	237	220	231
[5, 5, 5, 5, 5, 5, 5]	420	363	410
[8, 8, 8, 8, 5, 5, 5]	307	303	306
[5, 10, 5, 10, 5, 10, 5]	243	205	230
[8, 12, 10, 12, 5, 8, 8]	175	162	164
[4, 4, 8, 8, 8, 4, 4]	361	359	363
[5, 5, 5, 5, 10, 10, 10]	257	228	246
[9, 10, 8, 10, 5, 4, 6]	256	253	260

5 Conclusion

While the approaches presented here assume that the feasible region is convex, real feasible regions are almost always nonconvex. In the practical examples that we tested, however, it was seen that both the LHE and CP approaches provided good results. This could be attributed to the fact that real feasible regions are often "nearly convex," and that the polytope approximation algorithm shifts the hyperplane away from the feasible region boundary to compensate for slight nonconvexities [3].

For convex feasible regions, it would be expected that the CP method would give better results than the LHE method. However, this is not always so in practice, since real feasible regions are typically nonconvex.

REFERENCES

[1] H. L. Abdel-Malek and A.-K. S. O. Hassan, "The ellipsoidal technique for design centering and region approximation," *IEEE Trans. CAD*, pp. 1006–1014, Aug. 1991.

[2] P. M. Vaidya, "A new algorithm for minimizing convex functions over convex sets," *Proc. IEEE Foundations of Computer Science*, pp. 332–337, Oct. 1989.

[3] S. S. Sapatnekar, P. M. Vaidya, and S. M. Kang, "Feasible region approximation using convex polytopes," *Proc. ISCAS-93*, pp. 1786-1789, May 1993.

[4] S. W. Director and G. D. Hachtel, "The simplicial approximation approach to design centering," *IEEE Trans. CAS*, pp. 363–372, 1977.

[5] S. S. Sapatnekar, P. M. Vaidya, and V. B. Rao, "A convex programming approach to transistor sizing for CMOS circuits," *Proc. ICCAD-91*, pp. 482–485, Nov. 1991.

[6] A. Prekopa, "Logarithmic concave measures and related topics," in *Stochastic Programming* (M. Dempster, ed.), pp. 63–82, Academic Press, London, 1980.

[7] K. Singhal and J. F. Pinel, "Statistical design centering and tolerancing using parametric sampling," *IEEE Trans. CAS*, pp. 692–702, July 1981.

[8] S. S. Sapatnekar, P. M. Vaidya, and S. M. Kang, "Convexity-based algorithms for design centering," Tech. Rep. ISU-VLSI-93-SS01, Iowa State University, 1993.

[9] P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*. John Wiley and Sons, 1984.