

Parameterized Macromodeling for Analog System-Level Design Exploration

Jian Wang, Xin Li and Lawrence T. Pileggi

Department of ECE, Carnegie Mellon University
5000 Forbes Ave, Pittsburgh, PA 15213, USA

{jianw, xinli, pileggi}@ece.cmu.edu

ABSTRACT

In this paper we propose a novel parameterized macromodeling technique for analog circuits. Unlike traditional macromodels that are only extracted for a small variation space, our proposed approach captures a significantly larger analog design space to facilitate system-level design exploration. Combining a novel piece-wise approximation algorithm and a new multi-point model-order-reduction approach, the proposed method generates compact macromodels covering the entire feasible design space. Our experiments demonstrate that using such models can achieve more than 60× speed-up while incurring less than 4% overall error when varying design parameters by an order of magnitude.

Categories and Subject Descriptors:

B.7.2 [Integrated Circuits]: Design Aids – *simulation*

General Terms: Algorithms, Design

Keywords: Analog macromodeling, parameterized macromodel

1. INTRODUCTION

As the complexity of on-chip analog systems continues to increase, designing and optimizing these large-size systems become increasingly challenging. One major difficulty of analog system optimization stems from the expensive performance evaluation, which requires SPICE simulation of large transistor-level netlists and is impractical to be included into an optimization loop.

In order to make analog optimization computationally feasible, various modeling techniques have been proposed. One well-known approach is to approximate the block-level performance (e.g., amplifier gain) as a function of design variables [1–3]. However, a complete analog system (e.g., ADC, PLL) consists of dozens, or even hundreds of building blocks. Directly building the performance model for the entire system is too expensive, since it requires thousands or even millions of simulation samples [3]. Conversely, using block-level performance models to evaluate system-level performance is not easy, since the relation between block-level and system-level performances is typically unknown.

In this paper, we propose to systematically create block-level parameterized *macromodels* that are suitable for analog system-level optimization. Our proposed macromodel attempts to

approximate the input-output relationship of an analog block by a set of simplified differential algebraic equations. These block-level macromodels can be interconnected to facilitate fast system-level simulation. In addition, unlike traditional analog macromodels that have been mostly applied for bottom-up verification [1], the proposed macromodel is parameterized as a function of design variables such that it can facilitate top-down design exploration.

To create such parameterized macromodels, the challenging problem is how to accurately capture a *large* design space where the design variables can vary by orders of magnitude. The author of [4] demonstrated that for most analog circuits, a number of implicit sizing rules must be enforced to guarantee circuit functionality. Therefore macromodels should be created over this constrained design space, or *feasible region* only. As we will demonstrate in this paper, a low-dimensional projection space can be found for parameterized order reduction, if and only if the analog circuit stays within its feasible region.

The proposed analog macromodeling algorithm consists of two steps. Firstly, a novel algorithm is applied to automatically and recursively partition the large design space into small portions. The partitioning is formulated as a convex programming problem such that it can be solved both efficiently and robustly. Thereby a unified global macromodel is constructed as an accurate piece-wise approximation over the entire feasible region. Such a piece-wise approximation is necessary in our application, since most analog circuit equations are strongly nonlinear in feasible region and cannot be easily approximated by low-order polynomials.

Next, parameterized order reduction is applied to create a simple, yet accurate macromodel. We extend the single-point multi-parameter moment matching proposed in [5–6] to multiple-point cases such that creating a unified macromodel over the large design space becomes feasible.

2. BACKGROUND

2.1 Feasible Region

For analog design, after the circuit topology is decided, the designer will try to improve the performance of interest by optimizing several *design variables*, such as transistor sizes. In this paper, we assume that there are in total k design variables $p = [p_1, p_2, \dots, p_k]^T \in R^{k \times 1}$.

The space spanned by the design variables is referred to as the *design space*. In practice, a set of constraints must be applied to the design variables, which can be derived from physical requirements such as the minimal transistor size, or from functional requirements such as the bias condition for keeping a transistor in saturation [4]. These constraints generally can be approximated by posynomial functions in the form of [2]:

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 2007, June 4–8, 2007, San Diego, California, USA
Copyright 2007 ACM 978-1-59593-627-1/07/0006...\$5.00

$$f(p) = \sum_i c_i p_i^{\alpha_{i1}} \cdots p_k^{\alpha_{ik}} \leq 1 \quad (1)$$

where c_i is real non-negative and $\alpha_i = [\alpha_{i1}, \alpha_{i2}, \dots, \alpha_{ik}]^T \in R^{k \times 1}$. A design satisfying all the constraints is considered functional. The design space defined by those constraints is referred to as the *feasible region* [4], denoted as $P = \{p \mid f_j(p) \leq 1, j = 1 \cdots K\}$.

2.2 Parameterized Macromodel

For simplicity we limit our discussion in this paper to linear time-invariant (LTI) macromodeling. However, it should be noted that the proposed methodology can be extended to weakly nonlinear and/or time-variant macromodeling with minor modification.

In general, the LTI behavior of an analog circuit can be modeled by a set of linearized equations at the DC bias point:

$$\begin{aligned} (sC + G)x &= Bu \\ y &= L^T x \end{aligned} \quad (2)$$

where $u \in R^{m \times 1}$, $y \in R^{l \times 1}$, $x \in R^{N \times 1}$; C , G , B and L^T are matrices with appropriate dimensions. For a particular design, equation (2) can be obtained by using the small-signal device models and the modified nodal analysis (MNA) method. The resulting system matrices have deterministic values.

In design exploration, the values of the matrices C and G are defined as functions of the design variables (i.e., $C(p)$ and $G(p)$) and they should be optimized to improve circuit performance. The matrices B and L^T are uniquely determined by the circuit topology and therefore remain constant.

One common method to build parameterized macromodel is to approximate matrices C and G by polynomials [5–7]. For example, the first-order Taylor expansion of both matrices yields:

$$C(p) = C_0 + \sum_{i=1}^k C_i p_i, \quad G(p) = G_0 + \sum_{i=1}^k G_i p_i \quad (3)$$

where C_0 , C_i , G_0 and G_i are coefficient matrices.

2.3 Parameterized Order Reduction

To speed-up the simulation of large-scale circuits, model order reduction (MOR) is widely employed. The purpose of MOR is to generate a simplified system that approximates the original system, such that the evaluation cost can be reduced. Among various order reduction techniques, projection based methods are commonly used, such as PRIMA [8] or PMTBR [9]. In such methods, the original system (2) is projected onto a low-dimensional subspace to form the reduced system:

$$\begin{aligned} (s\hat{C} + \hat{G})\hat{x} &= \hat{B}u \\ y &= \hat{L}^T \hat{x} \end{aligned} \quad (4)$$

where $\hat{x} \in R^{n \times 1}$ ($n < N$).

Similarly, several order-reduction techniques have been developed recently to model large-scale parameterized systems [5–7]. For example, the CORE algorithm [5] uses a two-step scheme to match the multi-parameter moments. In the first step, an augmented system is formulated to explicitly match the moments of the parameters. Then in the second step, projection based method is applied to construct the reduced model.

3. PARAMETERIZED MACROMODEL GENERATION

Our proposed macromodeling technique is facilitated by two novel

techniques: 1) a recursive partitioning algorithm for piece-wise linear (or polynomial) approximation, and 2) a multi-point moment matching algorithm for parameterized model order reduction. In the following subsections we describe the details of these algorithms.

3.1 Design Space Partitioning and Model Fitting

The polynomial models (3) for matrices C and G were originally developed for interconnect analysis under process variations, where the parameters normally vary by 10–20% [5]. The situation, however, is quite different in analog macromodeling, where design variables can vary by 10× or more within the feasible region. In this case, the system response may exhibit strong nonlinearity with respect to the design variables, which cannot be approximated as a low-order polynomial. High order polynomial can be employed to improve the accuracy; however, it can be extremely expensive and computationally infeasible. To resolve this issue, we propose a novel piece-wise approximation technique to adaptively partition the large design space into multiple small spaces where the system response is weakly nonlinear and can be effectively approximated by the model template (3).

The partition process consists of three steps. Firstly we approximate the posynomial constraints in (1) by a set of linear constraints, resulting in a polytope in the design space. This enables us to formulate the partitioning problem as a convex optimization which can be solved efficiently and robustly. Such a convex partition is applied recursively until the predefined error tolerance is satisfied. Finally, when the partitioning is completed, we use a piecewise template to fit a global, parameterized model.

1) Constraint approximation

Assuming that all the design variables are positive (if not, simple shifting can be applied), we can convert the posynomial constraint (1) into a logarithmic space as:

$$f_L(p_L) = \log\left(\sum_i \exp(\alpha_i^T p_L + \beta_i)\right) \leq 0 \quad (5)$$

where $p_L = \log(p)$ and $\beta_i = \log(c_i)$ (both are element-wise). The author of [10] shows an algorithm to approximate an inequality in the form of (5) by a set of linear inequalities with controllable error bound. By such method, we can convert the constraints defining the feasible region into a set of linear constraints as:

$$\tilde{\alpha}_j^T \tilde{p}_L \leq 1 - \beta_j, \text{ for } j = 1 \cdots \tilde{K} \quad (6)$$

which actually define a polytope. The tilde superscripts are used since some auxiliary variables are introduced in the approximation.

2) Adaptive partitioning

Intuitively, the partition process can be described as the following steps. (a) Firstly the inscribed ellipsoid of the polytope is found, which roughly approximates the shape of the polytope. (b) Next, we choose a hyperplane passing the center of the ellipsoid and divide the polytope into two pieces of similar volumes. (c) This procedure is recursively applied to each piece until the predefined error tolerance is satisfied. The process is illustrated in Fig. 1

Assume that the ellipsoid is represented as $\Phi = \{Ev + d \mid \|v\|_2 \leq 1\}$

where d is its center. For the polytope defined in (6), the problem of finding the inscribed ellipsoid is equivalent to maximizing the volume of Φ while constraining it inside the polytope, which can be formulated as the following optimization problem:

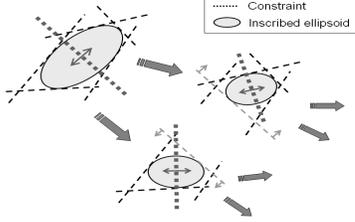


Figure 1 Design space partitioning.

$$\begin{aligned} \max \quad & \log(\det(E)) \\ \text{s.t.} \quad & \sup_{\|v\|_2 \leq 1} \tilde{\alpha}_j^T (Ev + d) \leq 1 - \beta_j, \text{ for } j = 1 \dots \tilde{K} \end{aligned} \quad (7)$$

This is a convex optimization problem and can be effectively solved by the interior-point or other algorithms [11].

When the inscribed ellipsoid is available, we can decide a direction τ and divide the ellipsoid (and the polytope) by the hyperplane orthogonal to τ . The direction should be chosen as in which the system response exhibit large nonlinearity. We use the fitting error to decide such a direction. First, we choose the $2\tilde{k}$ end-points of the axes and the center of current ellipsoid as sampling points. Based on these $2\tilde{k} + 1$ samples, a simple linear model is fitted using the template in (3). Then, this local model is evaluated at the sampling points; and we pick the axis with the largest error ε_{\max} as the partition direction.

The hyperplane is then defined as $\tau^T \tilde{p}_L = 0$. We can obtain two new polytopes by inserting constraints $\tau^T \tilde{p}_L \leq 0$ and $\tau^T \tilde{p}_L \geq 0$ into the current constraint set respectively. Then the above procedures are recursively applied to each new polytope until the predefined error tolerance is satisfied.

3) Global model fitting

We assume that we have in total q small partitions. The center of the ellipsoid associated with each local space will be used as one expansion point for fitting the matrices C and G . We replace the template in (3) with the following multi-point version:

$$C(p_L) = \sum_{i=1}^q w^{(i)}(p_L) \left[C_0^{(i)} + \sum_{i=1}^k C_i^{(i)} (p_{L_i} - p_{L_i}^{(i)}) \right] \quad (8)$$

where $w^{(i)}$ is the weighting function. (G -related equation is defined in the same pattern as C and is henceforth ignored.) Note that the bracketed superscript or subscript indicates the corresponding expansion point. The weighting function is defined as [12]:

$$w^{(i)}(p_L) = \frac{\exp(-\lambda \cdot \|p_L - p_{L_i}^{(i)}\|_2)}{\sum_{r=1}^q \exp(-\lambda \cdot \|p_L - p_{L_r}^{(r)}\|_2)} \quad (9)$$

where λ is a constant for shape-tuning. By this weighting function, the global model is represented as the weighted sum of all the local models. It is obvious that when evaluating the model at a certain point, the local models at nearby spaces will be assigned a large weight. This is based on the understanding that the matrices will share high similarities for systems in close range.

In the fitting process, we can reuse the sampling points used for local fitting and evaluation in the partition process. This will save the total cost for building the macromodel.

3.2 Multi-point Order Reduction

Next, we will demonstrate our proposed macromodeling technique

using a modified CORE algorithm. However, it should be noted that other parameterized MOR algorithms (e.g., variational PMTBR [6]) can also be utilized in our proposed flow.

Similar to the system matrices, state variable x is now expanded as:

$$x(p_L, s) = \sum_{i=1}^q w^{(i)}(p_L) \left[x_0^{(i)}(s) + \sum_{i=1}^k x_i^{(i)}(s) (p_{L_i} - p_{L_i}^{(i)}) \right] \quad (10)$$

In the step of explicit moment matching, the augmented system is constructed to match the moments of design variables at every expansion points, i.e., $x_0^{(i)}$ and $x_i^{(i)}$ for every t and i . The augmented system is formulated as:

$$\begin{pmatrix} C^{(1)} & \dots & C^{(q)} \\ s & \ddots & \vdots \\ C^{(1)} & \dots & C^{(q)} \end{pmatrix} + \begin{pmatrix} G^{(1)} & \dots & G^{(q)} \\ \vdots & \ddots & \vdots \\ G^{(1)} & \dots & G^{(q)} \end{pmatrix} \begin{bmatrix} x_A^{(1)} \\ \vdots \\ x_A^{(q)} \end{bmatrix} = \begin{bmatrix} B_A^{(1)} \\ \vdots \\ B_A^{(q)} \end{bmatrix} u \quad (11)$$

$$y = \begin{bmatrix} L_A^{(1)T} & \dots & L_A^{(q)T} \\ \vdots \\ x_A^{(q)} \end{bmatrix}$$

$$\text{where } C_{(r)}^{(i)} = \begin{bmatrix} WC_W & W\Delta p_{L_i} C_W & \dots & W\Delta p_{L_k} C_W \\ M_{C_1} & \Delta p_{L_i} M_{C_1} + WC_W & \dots & \Delta p_{L_k} M_{C_1} \\ \vdots & \vdots & \ddots & \vdots \\ M_{C_k} & \Delta p_{L_i} M_{C_k} & \dots & \Delta p_{L_k} M_{C_k} + WC_W \end{bmatrix},$$

$$B_A^{(i)} = \begin{bmatrix} B \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad L_A^{(i)} = w^{(i)}(p_L) \begin{bmatrix} L \\ (p_{L_1} - p_{L_1}^{(i)})L \\ \vdots \\ (p_{L_k} - p_{L_k}^{(i)})L \end{bmatrix}, \quad x_A^{(i)} = \begin{bmatrix} x_0^{(i)} \\ x_1^{(i)} \\ \vdots \\ x_k^{(i)} \end{bmatrix}; \text{ and the}$$

following notations have been used for simplicity: $W \equiv w^{(i)}(p_L)$,

$C_W \equiv C(p_L^{(i)})$, $M_{C_i} \equiv \partial[WC_W]/\partial p_{L_i}$, and $\Delta p_{L_i} \equiv p_{L_i}^{(i)} - p_{L_i}^{(r)}$.

For the derivative term M_{C_i} , the close-form expression can be obtained from (9) and (10) and is neglected here. Similar to the original version, the first equation of (11) is not parameterized. Therefore we can use any conventional method in the second-step moment matching such as PRIMA [8] or PMTBR [9].

For a large, nonlinear analog design space, we can always divide it into small slices using the adaptive partitioning algorithm such that the error of the global piece-wise model will stay in the same level as in the small local spaces. This feature makes the macromodeling methodology applicable to the large design space in the system-level design exploration. Furthermore, by the multi-point reduction scheme, a low-order reduced system can be constructed to accurately capture the dominant circuit response. The obtained macromodel can greatly reduce the evaluation cost, while simultaneously providing sufficient accuracy.

4. NUMERICAL EXAMPLES

In this section, we demonstrate the efficacy of the proposed macromodeling approach using two examples, a two-stage op-amp and a transconductor, as shown in Fig. 2. Both examples are implemented in the IBM 0.25 μm BiCMOS process.

4.1 Two-stage Op-amp

For this design, we recognize 7 design variables. And the expected ranges of some of these design variables are as much as 10 \times . To identify the feasible region, 21 constraints are included, which

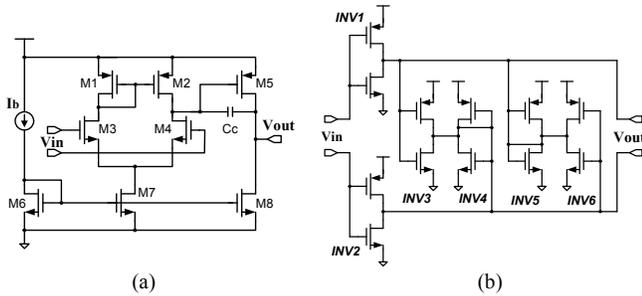


Figure 2 Schematics of (a) a two-stage op-amp, and (b) a transconductor.

guarantee the following requirements: (a) all the transistors are in saturation region; (b) a minimum overdrive voltage is provided for each transistor; (c) systematic offset voltage is minimized.

Macromodels are built by different methodologies. To evaluate each macromodel, a Monte Carlo (MC) simulation with 1000 randomly selected designs is performed. The simulation error is then calculated against HSPICE simulation results.

We first fit the linear model for the entire feasible region. As shown in Table 1, this single-partition (SP) model yields significant errors. Next, we apply the proposed macromodeling flow to the op-amp circuit and generate the multi-partition (MP) model. The feasible region is adaptively divided into 25 partitions and a uniform model is created. The parameterized order reduction method is then applied to reduce the model order from 51 to 16. As summarized in Table 1, the MP fitting template is much more effective in characterizing such a large design space. The proposed macromodeling flow is able to simultaneously provide model accuracy and compactness over the entire feasible design space.

In addition, for the MC simulation with 1000 sample designs, HSPICE requires 1160.0 seconds. Meanwhile, using the MP model needs only 15.3 seconds, which provides over 60 \times improvement in runtime. This indicates that by utilizing the parameterized macromodel in the system-level design exploration, the design process can be significantly accelerated.

Table 1 Macromodeling results

Circuit		op-amp		transconductor	
Model		SP	MP	SP	MP
Fitting	# of partitions	1	25	1	16
	Fitting error	> 50%	0.86%	> 50%	1.37%
	Model order	51	51	73	73
MOR	2nd step	—	PRIMA	—	PMTBR
	Final order	—	16	—	16
	MOR error	—	1.78%	—	3.60%
Overall error		—	1.74%	—	3.86%

4.2 Transconductor

Our second example is the transconductor circuit [13] with 8 design variables and 24 constraints. Again, some design parameters can range more than one order of magnitude.

Similarly, we build the MP model using the proposed approach. A SP model is constructed for comparison. 1000 designs randomly selected in the feasible region are employed to evaluate the model accuracy against the HSPICE transistor-level simulations. The results are summarized in Table 1 as well.

5. CONCLUSIONS

In this paper we proposed a novel technique to systematically build parameterized macromodels for analog circuit blocks. Unlike traditional macromodels that are extracted only for a fixed design for bottom-up verification, our macromodel is parameterized as a function of design variables to facilitate system-level trade-off analysis and optimization. In the macromodeling process, the feasible design space is first identified and recursively partitioned into smaller slices. A unified piece-wise model is then built such that the error will not scale up with the size of design space. More importantly, the high-dimensional design space partition problem is formulated as a convex optimization which can be reliably solved to find the optimal partitioning. After the partition process, a two-step parameterized order reduction technique is applied to effectively compress the obtained model while retaining good accuracy. It follows that the resulting macromodel can provide accurate system-level simulation results at a substantially lower evaluation cost.

6. ACKNOWLEDGEMENTS

This work was funded in part by the FCRP Focus Center for Circuit & System Solutions (C2S2), under contract 2003-CT-888.

7. REFERENCES

- [1] W. Daems, G. Gielen and W. Sansen, "Simulation-based automatic generation of signomial and posynomial performance models for analog integrated circuit sizing," *Proc. of ICCAD '01*, pp. 70-74, 2001.
- [2] M. Hershenson, "Efficient description of the design space of analog circuits," *Proc. of DAC '03*, pp. 970-973, 2003.
- [3] H. Liu, A. Singhee, R. Rutenbar and L. Carley, "Remembrance of circuits past: macromodeling by data mining in large analog design spaces," *Proc. of DAC '02*, pp. 437-442, 2002.
- [4] G. Stehr, M. Pronath, F. Schenkel, H. Graeb and K. Antreich, "Initial sizing of analog integrated circuits by centering within topology-given implicit specifications," *Proc. of ICCAD '03*, pp. 241-246, 2003.
- [5] X. Li, P. Li and L. T. Pileggi, "Parameterized interconnect order reduction with explicit-and-implicit multi-parameter moment matching for inter/intra-die variations," *Proc. of ICCAD '05*, pp. 806-812, 2005.
- [6] J. R. Phillips, "Variational interconnect analysis via PMTBR," *Proc. of ICCAD '04*, pp. 872-879, 2004.
- [7] L. Daniel, O. C. Siong, L. S. Chay, K. H. Lee and J. White, "A multiparameter moment-matching model-reduction approach for generating geometrically parameterized interconnect performance models," *IEEE Trans. on CAD*, vol. 23, no. 5, pp. 678-693, May 2004.
- [8] A. Odabasioglu, M. Celik and L. T. Pileggi, "PRIMA: passive reduced-order interconnect macromodeling algorithm," *IEEE Trans. on CAD*, vol. 17, no.8, pp. 645-654, Aug. 1998.
- [9] J. R. Phillips and L. M. Silveira, "Poor man's TBR: a simple model reduction scheme," *IEEE Trans. on CAD*, vol. 24, no. 1, pp. 43-55, Jan. 2005.
- [10] Y. Xu, K.-L. Hsiung, X. Li, I. Nausieda, S. Boyd and L. Pileggi, "OPERA: optimization with ellipsoidal uncertainty for robust analog IC design," *Proc. of DAC '05*, pp. 632-637, 2005.
- [11] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.
- [12] M. Rewienski and J. White, "A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices," *IEEE Trans. on CAD*, vol. 22, no. 2, pp. 155-170, Feb. 2003.
- [13] B. Nauta, "A CMOS transconductance-C filter technique for very high frequencies," *IEEE JSSC*, vol. 27, no. 2, pp. 142-153, Feb. 1992.