Time-Domain Performance Bound Analysis for Analog and Interconnect Circuits Considering Process Variations

Tan Yu*, Sheldon X.-D. Tan*, Yici Cai‡ and Puying Tang §
* Dept. Electrical Engineering, University of California, Riverside, CA 92521
‡ Department of Computer Science and Technology, Tsinghua University, Beijing, China 10084
§ School of Optoelectronic Information, University of Electronic Science and Technology of China, Chengdu, China 610054

***Abstract*—Time-Domain worst case or performance bound estimation for analog integrated circuits and interconnect circuits are crucial for both analog and digital circuit design and optimization in the presence of process variations. In this paper, we present a novel non-Monte-Carlo (MC) performance bound analysis technique in time domain. The new method consists of several steps. First the symbolic transient modified nodal analysis (MNA) formulation of the circuit matrices of (linearized) analog and interconnect circuits at a time step is formed. Then the closed-form expressions of the interested performance in terms of variational parameters of the circuit matrices of (linearized) analog and interconnect circuits are derived via a graph-based symbolic analysis method. Then time-domain performance response bound of current time step are obtained by a nonlinear constrained optimization process subject to the parameter variations and variational circuit state bounds computed from the previous time step. We study the bounds computed by the proposed against the different sigma bounds by the standard MC method, which shows that the proposed method is more efficient for computing high sigma bounds than the MC method. Experimental results show that the new method can deliver order of magnitudes speedup over the standard Monte Carlo simulation on some typical analog circuits and interconnect circuits with high accuracy.**

1. Introduction

At the nano-scale, circuit parameters are no longer truly deterministic and most of the quantities of practical interests present themselves as probability distributions. Circuit designers must now contend with these variations and uncertainties to ensure the robustness of their circuit designs. Traditional corner-based verification can't meet the accuracy requirements. For statistical analysis of digital and analog integrated circuits under process variations, Monte-Carlo (MC) based statistical simulation are the most popular methods due to their advantage of generality and high accuracy [1], [2]. However, MC method is expensive and slow especially for rare events (high sigma estimations) as more samplings are required, which will lead to the bottleneck of analog circuit optimization. Many fast Monte Carlo methods have been proposed to improve the efficiency of classical Monte Carlo methods. Existing approaches include importance sampling [3], Latin hypercube sampling based method [4], [5], and quasi Monte Carlo based method [2], [6]. However, the importance sampling method is circuit specific, Latin hypercube sampling does not work for all the circuits, and quasi Monte Carlo method suffers the high-dimensional problems [5].

This research was supported in part by NSF grants under No. CCF-1116882, No. CCF-1017090, No. OISE-1130402, No. OISE-0929699 and UC MEXUS-CONACYT CN-11-575.

At the same time, some alternative non Monte Carlo methods have been proposed for statistical analysis. Among them, performance bound analysis methods emerged as attractive techniques for statistical analysis and yield estimation. Those techniques hold the promises that they are more scalable for high sigma and high dimensional statisitical analysis problems compared to existing Monte-Carlo method approaches. Bounding or worse case analysis of analog circuits under parameter variations has been studied in the past for fault-driven testing and tolerance analysis of analog circuits [7]–[9]. Recently, some frequency domain performance bound methods were proposed in [10]–[13] to compute the lower and upper bounds of transfer function's magnitude and phase. The work in [10] applied a control-based method [14] to obtain the performance bounds in frequency domain, and [11] applied an optimization based method to compute the bounds in time domain, in which the whole circuit equations are treated as constraints, which can be very expensive to enforce the constraints. This method has been improved by [12] where symbolic analysis approach was applied to derive exact transfer functions and affine interval method was used to compute variational transfer functions. However, the affine interval method can lead to over conservative results. This work has been extended to compute the time-domain performance bounds based on the frequency domain bounds [13]. Recently analog performance bound analysis in frequency domain based on the optimization method and symbolic analysis techniuqe was proposed in [15].

In this paper, we present a novel non-Monte-Carlo performance bound analysis technique in time domain. The new contributions of the paper lies in the following aspects:

- First, we develop a new general time-domain performance analysis method, which consists of several steps: First the time-domain symbolic modified nodal analysiss (MNA) formulation of (linearized) analog and interconnect circuits at a time step is formed. Then the closed-form expressions of the interested performance in terms of variational parameters of the circuit matrices of (linearized) analog and interconnect circuits are dervied via a graph-based symbolic analysis method. Then time-domain performance response bounds of current time step are obtained by finding the max/min values via a nonlinear constrained optimization process subject to the parameter variations and variational circuit state bounds computed from the previous time step.
- Second, we further study the bounds computed by the

proposed method againt the different sigma bounds by the standard MC method, which shows that the proposed method is more efficient for computing high sigma bounds than the MC method, which will increase rapidly (almost exponentially) with increasing sigma. In contrast, the run time of the proposed method will remain the almost the same as it only deals with different parameter bounds with the same number of parameters.

Experimental results show that the new method can deliver one or two order of magnitudes speedup over standard Monte Carlo simulation on some typical analog circuits and interconnect circuits with very high accuracy.

2. New time-domain performance bound analysis technique

We first present the whole algorithm flow of the proposed new performance bound analysis algorithm in Alg. 1. Basically the proposed method consists of three major computing steps. The first step is to set up the symbolic circuit matrices in the time domain based on the companion models of the dynamic elements (Step 2). The second step is to compute the variational closed form expressions of interesting states from the variational circuit parameters, which will be done via DDD-based symbolic analysis method (Step 3). Third, we compute the time-domain response bounds via a constrained nonlinear optimization process in each time step (Step 6-7). We will present the computing steps in the following subsections.

Algorithm 1 New time-domain performance bound analysis

Input: Circuit netlist, bounds of selected parameters.
Output: Conservative performance bound of interests

1: Convert the circuit C and L elements into companion models
2: Generate symbolic expression of closed form expressions for interesting nodes
3: **for** each time step **do** *// Perform transient analysis*
4: Set bounds on process variational parameters.
5: Set bounds on the voltage or current states from results of optimization of last time step.
6: Run nonlinear constrained optimization (9) which uses closed form function as the objective. To find upper bound and lower bound.
7: Save bound information for the optimization of next time step.
8: **end for**
9: Output the bound of voltage or current on every time step.

A. Symbolic transient analysis for analog circuits

In this subsection, we review a graph-based transient symbolic analysis for obtaining the exact symbolic closed form expressions of analog circuits. Graph-based symbolic technique is a viable tool for calculating the behavior or characteristic of analog circuits [16]. The introduction of determinant decision diagrams based symbolic analysis technique (DDD) allows exact symbolic analysis of much larger analog circuits than all the other existing approaches [17], [18]. Furthermore, with hierarchical symbolic representations [19], exact symbolic analysis via DDD graphs essentially allows the analysis of arbitrary large analog circuits.

Existing symbolic analysis was mainly formed in the frequency domain to build the symbolic transfer functions [20]. Symbolic analysis in time domain is less investigated and will be explored in this paper. To better illustrate the proposed method, we would like to walk through one simple example. Fig. 1 shows a simple RC ladder circuit. To perform the transient analysis, we first convert capacitance into its companion models (using the Back-Euler method) as shown in Fig. 2.

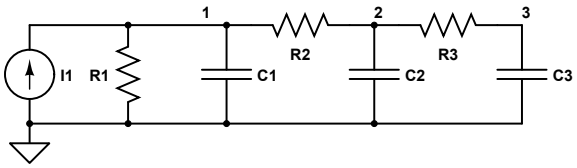


Fig. 1: RC ladder circuit

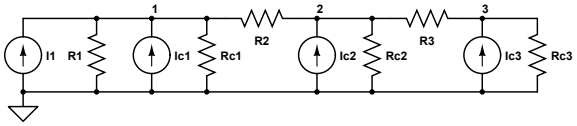


Fig. 2: RC ladder with companion models for capacitances

The corresponding modified nodal analysis (MNA) formulation of the circuit in time-domain at time step $n+1$ can be written as:

$$Y\vec{v}(n+1) = \vec{i}(n+1) \tag{1}$$

where Y is the MNA matrix given by

$$\begin{bmatrix} \frac{1}{R_1}+\frac{1}{R_{c1}} & -\frac{1}{R_2} & 0 \\ -\frac{1}{R_2} & \frac{1}{R_2}+\frac{1}{R_{c2}}+\frac{1}{R_3} & -\frac{1}{R_3} \\ 0 & -\frac{1}{R_3} & \frac{1}{R_3}+\frac{1}{R_{c3}} \end{bmatrix} \tag{2}$$

and

$$\vec{v}(n+1) = \begin{bmatrix} v_1(n+1) \\ v_2(n+1) \\ v_3(n+1) \end{bmatrix} \tag{3}$$

and

$$\vec{i}(n+1) = \begin{bmatrix} i_1(n+1)+i_{c1}(n) \\ i_{c2}(n) \\ i_{c3}(n) \end{bmatrix} \tag{4}$$

where $R_{c1} = \frac{C_1}{\Delta t}$, $R_{c1} = \frac{C_2}{\Delta t}$ $R_{c1} = \frac{C_3}{\Delta t}$, $i_{c1}(n) = \frac{v_1(n)*C_1}{\Delta t}$, $i_{c2}(n) = \frac{v_2(n)*C_2}{\Delta t}$, $i_{c3}(n) = \frac{v_3(n)*C_3}{\Delta t}$ and Δt is the time step size.

Then the unknown nodal voltage are solved using Cramer's rules.

$$v_i(n+1) = \frac{det(Y_i(n+1))}{det(Y)} \tag{5}$$

where $Y_i(n+1)$ is the matrix formed by replacing the ith column of Y by vector $\vec{i}(n+1)$.

DDD is a very powerful tool to compute the symbolic determinant. Once the characteristics of circuits are presented by

DDDs, evaluation of DDDs, whose CPU time is proportional to the size of DDDs, will give exact numerical values.

We view each entry in the circuit matrix as one distinct symbol, and rewrite its system determinant in the left-hand side of Fig. 3. Then its DDD representation is shown in the right-hand side.

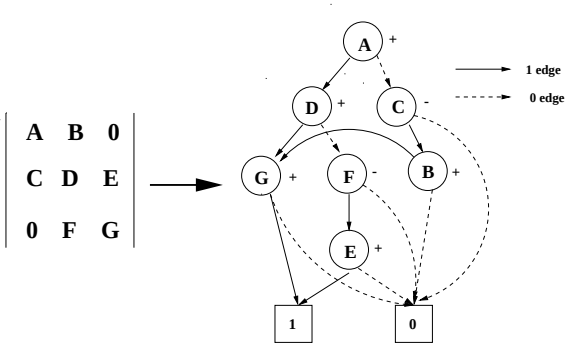


Fig. 3: A matrix determinant and its DDD representation.

Once a DDD has been constructed, the numerical values of the determinant it represents can be computed by performing the depth-first type search of the graph and performing one multiplication and addition at each node, whose time complexity is linear function of the size of the graphs (its number of nodes). The computing step is called *Evaluate(D)* where D is a DDD root. With proper node ordering and hierarchical approaches, DDD can be very efficient to compute transfer functions of large analog circuits [17], [19].

B. Variational symbolic closed-form expressions for transient states

To find the performance bounds of specific transient state variable, say $v_i(n+1)$ at time step $n+1$, DDD graphs are built for $det(Y_i(n+1))$ and $det(Y)$, we will obtain the following closed form symbolic expression for $v_i(n+1)$,

$$v_i(n+1) = f_i(p_1, \ldots, p_m, v_1(n), \ldots, v_k(n)) = \frac{f_{n,i}(p_1, \ldots, p_m, v_1(n), \ldots, v_k(n))}{f_{d,i}(p_1, \ldots, p_m)} \quad (6)$$

where functions $f_{n,i}(p_1, \ldots, p_m, v_1(n), \ldots, v_k(n))$ and $f_{d,i}(p_1, \ldots, p_m)$ are represented by DDD graphs and $p_1, \ldots, p_m$ are m circuit variables and $v_1(n), \ldots, v_k(n)$ are the state variables computed from previous time step n. Notice that $v_i(n+1) = f_i(p_1, \ldots, p_m, v_1(n), \ldots, v_k(n))$ describes nonlinear functions in terms of $p_1, \ldots, p_m$, $v_1(n), \ldots, v_k(n)$. All the variables at current time step $n+1$ have variational bounds:

$$p_{il} \leq p_i \leq p_{iu} \quad (7)$$

$$v_{il}(n) \leq v_i(n) \leq v_{iu}(n) \quad (8)$$

Note that the variational bounds of state variable $v_i(n)$ are obtained form the previous time step n. In our presentation, we assume that the external voltage or current sources do not have variations to simplify our presentation. But this is not the limitation of the proposed method and we can trivially add this into our method.

To compute the numerical value of $v_i(n+1)$ for given specific values of $v_i(n+1) = f_i(p_1, \ldots, p_m, v_1(n), \ldots, v_k(n))$, this can be done by DDD *Evaluation* operation, which traverses the DDD in a depth-first style and performs one multiplication and one addition at each node.

Get back to the illustrative example, for voltage at node i at time step $n+1$, $v_i(n+1)$, we have

$$v_i(n+1) = f_i(C_1, C_2, C_3, R_1, R_2, R_3, v_1(n), v_2(n), v_3(n))$$

C. Variational bound analysis in time domain

To find the performance bounds subject to the parameter variations at time step $n+1$, we formulate the bound computing problem into a nonlinear constrained optimization problem. We use the lower bound of the voltage of node i on time step $n+1$ for an example. The symbolic expression of the voltage of node1, which has been obtained by DDD symbolic analysis, is used as the nonlinear objective function to be minimized:

$$\begin{array}{ll} \text{minimize} & v_i(n+1)(\mathbf{x}) = f_i(\mathbf{x}) \\ \text{subject to} & \mathbf{x}_{\text{lower}} \leq \mathbf{x} \leq \mathbf{x}_{\text{upper}}, \end{array} \quad (9)$$

where $\mathbf{x} = [\mathbf{p}, \mathbf{v}]$, in which, $\mathbf{p} = [p_1, \ldots, p_m]$ represents the circuit parameter variable vector, which is subjected to the optimization constraints $[\mathbf{p}_{\text{lower}}, \mathbf{p}_{\text{upper}}]$. In circuit design, foundries and cell library vendors supply these constraints. On the other hand, $\mathbf{v} = [v_1(n), \ldots, v_k(n)]$ represents the nodal voltage on the last time step, which are determined by the results of optimization of the last time step. Hence, after (9) is solved by an optimization engine, the lower bound of the v_1 on $(n+1)$th time step is returned and then serves as constrained condition for the optimization of voltage on $(n+2)$th time step.

The nonlinear optimization problem with simple upper and lower bounds given in (9) can be efficiently solved by several methods such as active-set, interior-point, and trust-region algorithms [21]–[23]. All those methods are iterative approaches starting with an initial feasible solution. In this work, we use the active-set method [23], as it turns to be the most robust nonlinear optimization method for our application. Active-set method is a two-phase iterative method that provides an estimate of the active set (which is the set of constraints that are satisfied with equality) at the solution. In the first phase, the objective is ignored while a feasible point is found for the constraints. In the second phase, the objective is minimized while feasibility is maintained. In this phase, starting from the feasible initial point $\mathbf{x}_0$, the method computes a sequence of feasible iterates $\{\mathbf{x}_k\}$ such that $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ and $f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k)$ via methods like quadratic programming, where $\mathbf{d}_k$ is a nonzero search direction and α_k is a nonnegative step length.

Since the responses at two neighboring time step are usually close to each other, the starting point $\mathbf{x}$ for nth time step can be set using the solution on $(n-1)$th time step. This strategy tends to reduce the time required by the optimization to search its minimal or maximal point in the whole variable space, and thus speedup the calculation time of the bound analysis.

We remark that the active-set method is still a local optimization method, which finds the local optimal solutions. But find the true bound may come with more or much higher

computing costs by performing many tries. In our approach, we still perform one optimization. Our experimental results show that the proposed method gives conservative bounds for given sigma values compared with Monte Carlo methods for the examples used.

3. Numerical results and discussions

In this section, we show experimental results of the proposed method on interconnect circuits and analog circuits. The DDD symbolic tool generates the exact transfer function expressions first [17], and all the follow-up optimization based bound calculation and yield estimation are done in MATLAB. The nonlinear constrained optimizations are solved by the *fmincon* function in MATLAB's Optimization Toolbox [24]. All running time are sampled from a Linux server with a 2.4 GHz Intel Xeon Quad-Core CPU, and 36 GB memory.

We compare proposed method with standard Monte Carlo analysis in terms of running time and accuracy using two examples. In all the examples, we assume that variational parameter has Gaussian distributions with the standard deviation σ. Their variational bound (3-sigma bound) will be $[-3\sigma+\mu, 3\sigma+\mu]$ where μ is the mean of the random process.

A. An interconnect RC tree circuit example

The first example is an interconnect RC tree example, which is driven by a voltage source as shown in Fig. 4.

The variational parameters are $R_i = 0.1\Omega$, $i = 1, 2, 3$, $C_j = 0.1pf$, $j = 1, 2, 3$. All parameters have 10% variations, which means that, for proposed method, the constrained condition is $(1 - 5\%) * p_{std} \leq p \leq (1 + 5\%) * p_{std}$, for Monte Carlo analysis, $\sigma = 1/6 * 10\% * p_{std}$ (3-sigma bound), in which, p represents the value of a certain varational parameter, and p_{std} is the standard value of the parameter.

Fig. 5 shows the transient step response 3-sigma bound of voltage of node 8 with that from the proposed method and simulation result from 5000 MC runs. This figure shows that the bounds from proposed method could safetly cover the curves from the Monte Carlo simulation. Fig. 6 shows 3-sigma bounds from 2000 MC runs, 5000 MC runs and the proposed method at 0.5 ns.

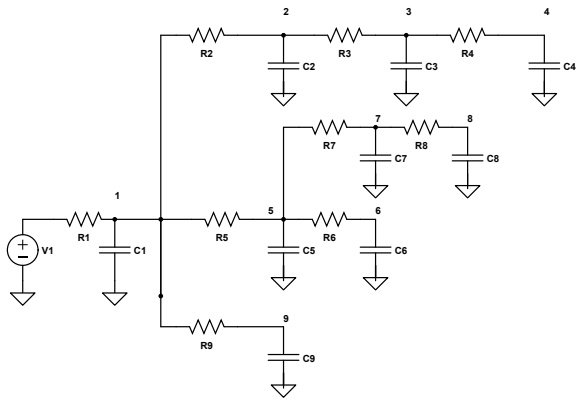


Fig. 4: A RC tree circuit

We have several observations: First, the bounds given by the proposed method matches with that given by the MC method

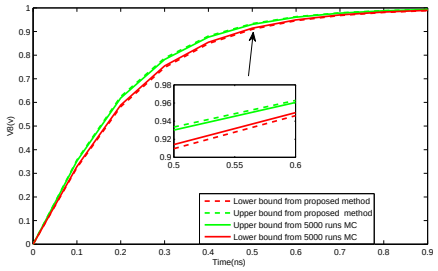


Fig. 5: The bounds of V8 obtained from 5000 MC runs and the proposed method on the RC tree circuit

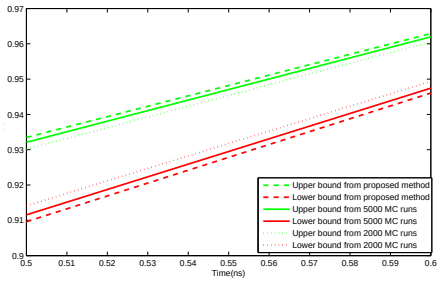


Fig. 6: Comparison of bounds of V8 from 2000 MC runs, 5000 MC runs and the proposed method on the RC tree circuit

very well. Since all of parameters take 3-sigma bounds, the bounds computed by the proposed method should be close to 3-sigma bounds as well. If the output bounds are Gaussian, then 3-sigma will cover 99.730% area under the probability density function (pdf) of Gaussian distribution, which means we need to take at least 370 MC runs to have event to reach the bound.

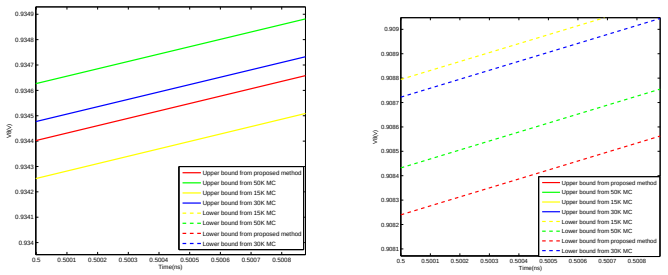
Table I compares the runtime, voltage values of the proposed method and that of the Monte Carlo method and also shows the error ratio of 2000 MC runs, 5000 MC runs. The table also shows that, our proposed method has 8.3x speedup over 5000 MC run simulation.

TABLE I: Comparison between the methods on lower bounds of V8 at t=0.5ns for the RC tree circuit

Method	Samplings#	CPU(sec)	Voltage(V)	% Error
Monte Carlo	2000	573.383	0.915	0.55
Monte Carlo	5000	1490.798	0.912	0.22
Proposed Method	1	180.432	0.910	N/A

To further study the bounds computed by the proposed method, we compared 3-sigma bounds given by 15K, 30K, 50K MC runs. Fig. 7a shows the 3-sigma upper bound of V8 from the 15K, 30K, 50K runs of MC simulation and that from our proposed method around 0.5ns. In this figure, we observe that 3-sigma bounds given by 30K and 50K now go outside the bound of the proposed method.

Fig. 7b shows the 3-sigma lower bound of V8 from 15K, 30K, 50K runs of MC simulation and the bound give by the proposed method. In this case, we observe that the the bound by the proposed method contain ALL the bounds by different MC runs. This is a very interesting observation.



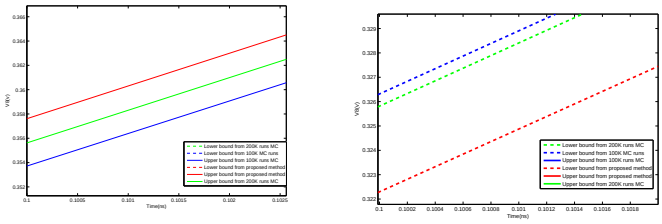
(a) Upper bounds (b) Low bounds

Fig. 7: Comparison of 3-sigma upper and lower bounds of V8 from 15K MC runs, 30K MC runs, 50K MC runs and the proposed method on the RC tree circuit at 0.5ns

One possible explanation for the Fig. 7a and Fig. 7b is that the performance function may not be monotonic function for the some of parameter variables (called non-monotonic parameters here). In other words, the min/max values of performance function may not be reached at the edges of the bounds of those parameters. So the 3-sigma bound already includes the values to reach the min/max values of the functions. This can explain Fig. 7a, in which the maximum value is reached when most of variational parameter values found are not at edges of the bound. As a result, new approach will find more conservative bounds as those non-monotonic parameters reach their min/max values already. For Fig. 7b, on the other hand, the minimum value is reached when many variational parameters are at edges of the bound. However, it is almost impossible for all variational parameters are at edge of bounds at the same time considering the Gaussian distribution especially when the number of variational parameter is large. Therefore, the bound computed by new approach close to the true bound and more MC run can only get closer to the bound, but can't go beyond the true bound. As a result, we can see the new method tend to find the true bound more efficient than the MC method, especially for performance functions which achives min/max values when many parameters are at edge of bounds, as it requires quite a great amount of samplings to possibly get the maxmum or minimum.

To further study the behavior of the proposed method, we perform 4-sigma bound analysis in which the bounds of each parameters will be $[-4\sigma + \mu, 4\sigma + \mu]$. Fig. 8a shows the 4 sigma upper bounds from 100K MC runs, 200K MC runs and the proposed method at 0.1ns. Fig. 8b shows the 4-sigma lower bounds from 100K MC runs, 200K MC runs and that from proposed method at 0.1ns. From the two figures we can see, that in this case, the proposed method contain both upper bounds and lower bounds from the MC runs (even with 200K MC). It means that 4-sigma upper bound computed by our method is large than 4-sigma bounds of MC simulation. On the other hand, for the lower bounds, we observe the same results as 3-sigma bound results: the proposed method is always lowest bound among all the methods.

As a result, it can be seen that the proposed method is more efficient to find the high sigma bounds, as it takes almost the same computational costs as computing low sigma bounds, than the standard MC methods, whose computational costs go up almost exponentially with high sigma bounds.



(a) Upper bounds (b) Low bounds

Fig. 8: Comparison of 4-sigma upper and lower bounds of V8 from 100K MC runs, 200K MC runs and the proposed method on the RC tree circuit at 0.1ns

TABLE II: Comparison between the methods on lower bounds of Vout at t=1ms for the amplifier circuit

Method	Samplings#	CPU(sec)	Voltage(V)	% Error
Monte Carlo	2000	412.071	-0.942	3.7
Monte Carlo	5000	1112.597	-0.906	0.79
Proposed method	1	105.460	-0.899	N/A

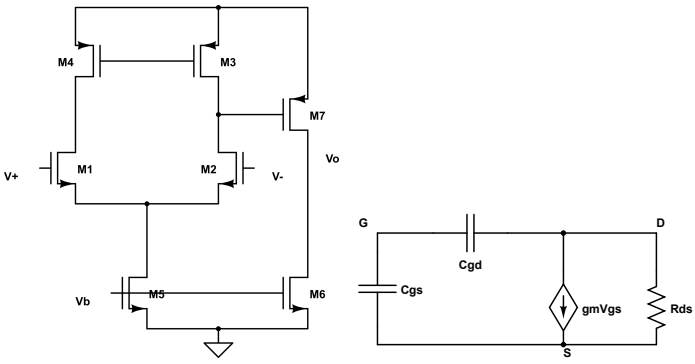
B. An opamp circuit example

The second example is an opamp circuit with 7 MOSFETs as shown in Fig. 9a. To perform the bound analysis, we use a linearized and simplified device models for the MOSFETs as shown in Fig. 9b. The variable parameters are $M_1.g_m = 1.5 * 10^{-5}$, $M_1.C_{gd} = 0.5fF$, $M_1.C_{gs} = 5fF$, $M_2.g_m = 1.5*10^{-5}\Omega^{-1}$, $M_2.C_{gd} = 0.5fF$, $M_2.C_{gs} = 5Ff$, $M_5.r_{ds} = 5*10^7\Omega$, $M_6.r_{ds} = 5*10^7\ \Omega$. Again all parameters have 10% variations.

Fig. 10 shows the transient response 3-sigma bound of Vout with sinusoidal wave input obtained from proposed method and simulation result from 5000 MC runs. Fig. 11 shows the 3-sigma bounds from 2000 MC runs, 5000 MC runs, and the proposed method at 1ms. In this case, we observe that the bounds from proposed method is still conservative such that it sill contain the bounds from all the MC runs. The possible reasons have been explained before.

Table II compares the runtime, voltage values of the proposed method and that of Monte Carlo method. It also shows the error ratio of 2000 MC runs and 5000 MC runs. It can been seen that the errors are quite small and get smaller as we take more MC runs, which is the consistent with the MC method.

The same table also shows that, our proposed method has 10.6x speedup over 5000 samplings MC simulation. We remark that, if high sigma (larger than 3 sigma) bounds, the standard MC runs will increase rapidly (almost exponentially), while the run time of the proposed method will remain the almost the same as it only deal with different parameter bounds with the same number of parameters. As a result, the proposed method indeed overcome the high sigma issues with the standard MC based method, which is the major advantage of the proposed method over MC based methods.



(a) An opamp circuit (b) The simplifed MOSFET model

Fig. 9: The opamp circuit and its MOSFET model

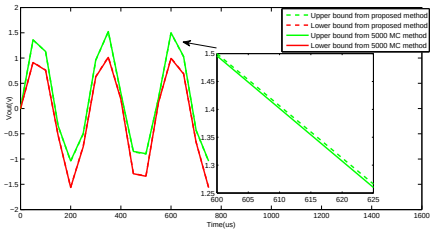


Fig. 10: The bounds from 5000 MC runs and the proposed method on the amplifier circuit

4. Conclusion

In this paper, we have presented a new non-Monte-Carlo performance bound analysis technique in time domain. The new method is based on the constrained non-linear optimization and advanced symbolic analysis techniques. We have shown that the proposed method is more efficient for computing high sigma bounds than the standard MC method. Experimental results have shown that the new method can deliver order of magnitudes speedup over standard Monte Carlo simulation on some typical analog circuits and interconnect circuits with very high accuracy.

References

[1] G. F. Fishman, *Monte Carlo, concepts, algorithms, and Applications*. Springer, 1996.

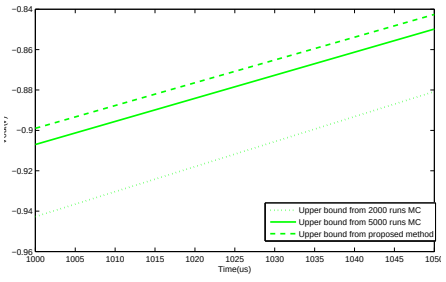


Fig. 11: Comparison of upper bounds of Vout from 2000 MC runs, 5000 MC runs and the proposed method on the the amplifier circuit

[2] A. Singhee and R. A. Rutenbar, "Why quasi-monte carlo is better than monte carlo or latin hypercube sampling for statistical circuit analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 11, pp. 1763–1776, 2010.

[3] T. S. Doorn, E. J. W. ter Maten, J. A. Croon, A. Di Bucchianico, and O. Wittich, "Important sampling monte carlo simulations for accurate estimation of sram yield," in *ESSCIRC 2008 - 34th European Solid-State Circuits Conference*, pp. 230–233, IEEE, 2008.

[4] J. F. Swidzinski, M. Keramat, and K. Chang, "A novel approach to efficient yield estimation for microwave integrated circuits," in *42nd Midwest Symposium on Circuits and Systems*, pp. 367–370, IEEE, 1999.

[5] A. B. Owen, "Latin supercube sampling for very high-dimensional simulations," *ACM Transactions on Modeling and Computer Simulation*, vol. 8, pp. 71–102, Jan. 1998.

[6] B. Liu, J. Messaoudi, and G. Gielen, "A fast analog circuit yield estimation method for medium and high dimensional problems," in *Proc. Design, Automation, and Test in Europe (DATE)*, pp. 751–756, 2012.

[7] L. Kolev, V. Mladenov, and S. Vladov, "Interval mathematics algorithms for tolerance analysis," *IEEE Trans. on Circuits and Systems*, vol. 35, pp. 967 –975, Aug. 1988.

[8] W. Tian, X.-T. Ling, and R.-W. Liu, "Novel methods for circuit worst-case tolerance analysis," *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications*, vol. 43, pp. 272 –278, Apr. 1996.

[9] C.-J. R. Shi and M. W. Tian, "Simulation and sensitivity of linear analog circuits under parameter variations by robust interval analysis," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 4, pp. 280–312, July 1999.

[10] L. Qian, D. Zhou, S. Wang, and X. Zeng, "Worst case analysis of linear analog circuit performance based on kharitonov's rectangle," in *Proc. IEEE Int. Conf. on Solid-State and Integrated Circuit Technology (ICSICT)*, Nov. 2010.

[11] S. Saibua, L. Qian, and D. Zhou, "Worst case analysis for evaluating VLSI circuit performance bounds using an optimization method," in *IEEE/IFIP 19th International Conference on VLSI and System-on-Chip*, pp. 102–105, 2011.

[12] Z. Hao, R. Shen, S. X.-D. Tan, and G. Shi, "Performance bound analysis of analog circuits considering process variations," in *Proc. Design Automation Conf. (DAC)*, pp. 310–315, July 2011.

[13] X.-X. Liu, S. X.-D. Tan, Z. Hao, and G. Shi, "Time-domain performance bound analysis of analog circuits considering process variations," in *Proc. Asia South Pacific Design Automation Conf. (ASPDAC)*, Jan. 2012.

[14] V. L. Kharitonov, "Asymptotic stability of an equilibrium position of a family of systems of linear differential equations," *Differential. Uravnen.*, vol. 14, pp. 2086 –2088, 1978.

[15] X. Liu, A. Palma-Rodriguez, S. Rodriguez-Chavez, S. X.-D. T. E. Tlelo-Cuautle, and Y. Cai, "Performance bound and yield analysis for analog circuits under process variations," in *Proc. Asia South Pacific Design Automation Conf. (ASPDAC)*, pp. 761–766, January 2013.

[16] G. Gielen, P. Wambacq, and W. Sansen, "Symbolic analysis methods and applications for analog circuits: A tutorial overview," *Proc. of IEEE*, vol. 82, pp. 287–304, Feb. 1994.

[17] C.-J. Shi and X.-D. Tan, "Canonical symbolic analysis of large analog circuits with determinant decision diagrams," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, pp. 1–18, Jan. 2000.

[18] C.-J. Shi and X.-D. Tan, "Compact representation and efficient generation of s-expanded symbolic network functions for computer-aided analog circuit design," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, pp. 813–827, April 2001.

[19] S. X.-D. Tan, W. Guo, and Z. Qi, "Hierarchical approach to exact symbolic analysis of large analog circuits," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, pp. 1241–1250, August 2005.

[20] Z. Qin, S. X.-D. Tan, and C. Cheng, *Symbolic Analysis and Reduction of VLSI Circuits*. Boston, MA: Kluwer Academic Publishers, 2005.

[21] R. H. Byrd, R. B. Schnabel, and G. A. Shultz, "A trust region algorithm for nonlinearly constrained optimization," *SIAM Journal on Numerical Analysis*, vol. 24, no. 5, pp. pp. 1152–1170, 1987.

[22] P. E. Gill, W. Murray, Michael, and M. A. Saunders, "Snopt: An sqp algorithm for large-scale constrained optimization," *SIAM Journal on Optimization*, vol. 12, pp. 979–1006, 1997.

[23] C. A. Floudas, *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications (Topics in Chemical Engineering)*. Oxford University Press, 1995.

[24] The Mathworks Inc., *MATLAB Optimization Toolbox*. http://www.mathworks.com/help/toolbox/optim/, 2012.