A least squares based methodology for the extraction of corner model parameters

Simone Locci, Klaus-Willi Pieper, Elmar Gondro
Infineon Technologies AG
Neubiberg, Germany

***Abstract*— For SPICE level circuit development, corner models address the designer's need for fast simulations which can reproduce specific scenarios of interest. To systematically and automatically determine the model parameters for the corners, we present a methodology based on a constrained least squares optimization. We also show how it performs when applied to some of our devices for best and worst case scenarios.**

Keywords—spice simulation; corner modeling; least squares optimization

I. INTRODUCTION

Nowadays circuit development relies on several tools and simulation options to determine accurately and in detail the behavior of an electronic system. Let us consider spice level circuit simulations. A Monte Carlo analysis, for example, provides the designers with information on statistical deviations with respect to the nominal models, accounting for the fluctuations in the fabrication process parameters.

The designer might also be interested in simulating specific circuit scenarios, like a worst case scenario where the parameters of the compact models describe devices operating with the minimal performances determined by the process variations. In this case, corner simulations are a quick and effective solution [1]. In general, a corner model can be seen as any defined set of model parameters which describe a device in a condition of interest; as they are particularly important for us, we will focus on corners which can reproduce the edges of the devices process windows (i.e. at the limits of the allowed ranges of the process parameters, when the performances of the devices match their upper and lower specifications), but all the considerations that will follow can be easily extended to different corners configurations.

Goal of this paper is to develop a methodology which allows a systematic and automatic extraction of the optimal parameters needed by the compact models to reproduce the corners we use for circuit development.

The paper is structured as follows: in section II, we introduce the definitions and parameters used throughout the paper. Section III has a description of our methodology. In section IV we will show its application to generate the corners used for our technologies. Section V provides a summary of the most important points and the conclusions.

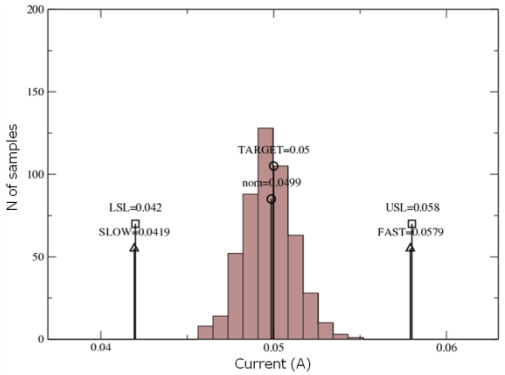

Fig. 1. Saturation current of an n-MOSFET device. The target value is the desired nominal current, while the "nom" value is the actual current reproduced by the model. USL and LSL are the maximum and minimum allowed currents, whereas FAST and SLOW are the currents we would like to get with a corner simulation. A Monte-Carlo simulation is also shown for comparison (total number of samples: 1000).

II. DEFINITIONS

To introduce the parameters that are required throughout the paper, we would like to refer to Fig. 1.

It shows, for an NMOS device, the possible values that the saturation current ISAT can assume at a certain bias point as well as a Monte Carlo simulation of its statistical behaviour. The current is regularly measured on the test structures of the wafers in the fab and it is part of a set of measured parameters which are called PCM (Process Control Monitor) data. We call the highest and lowest values for the current that will be accepted as suitable for production USL (Upper Spec Limit), and LSL (Lower Spec Limit), respectively. Depending on the process, the difference $(USL - LSL) = \lambda\sigma$ will also determine a long-term estimation for its standard deviation (for the sake of simplicity, we assume that a Gaussian distribution can properly fit the long-term estimation). Typically $\lambda = 9$ or 6 .

The target value is the value that the current should ideally have in standard operating conditions, whereas the nominal value is the one which can be reproduced by the compact model under the same conditions and it is also the mean value of the Gaussian distribution. If the latter is correctly centred, the difference between the target and the nominal value is minimal.

TABLE I. CORNER DEFINITION FOR AN N-TYPE MOSFET

	FAST	*SLOW*
VTLIN	LSL	USL
VTSAT	LSL	USL
ISAT	USL	LSL
RON	LSL	USL

For CMOS devices, corners are usually defined for the fastest and slowest switch time cases [1]. Following this definitions, we will define "fast" and "slow" corners as the corners describing the best and worst performances for the devices under investigation. In the example of Fig. 1, the fast corner is the corner with the highest current (USL), whereas the slow corner reproduces the lowest current (LSL). Table 1 shows how USL and LSL can be chosen for the fast and slow corners of an n-type MOSFET device: a fast corner will have, besides a high drain current ISAT, a low threshold voltage (VTLIN in linear regime and VTSAT in saturation regime) as well as a low on-resistance RON. The associated numerical vector will be referred as **p**.

In a compact model, a specific device is in general described by a set of (nominal) model parameters. To generate a corner which reproduces **p** as good as possible, we will change a subset **m** of all the available compact model parameters. Usually, the choice of the parameters in the subset is left to the modelling engineer, but often it will resemble the set of parameters which are varied during a Monte Carlo simulation. In a previous work [2], we showed that from a correlation analysis of all PCM data it is possible to find, for all parameters **m**, the optimal correlations and standard deviations $\boldsymbol{\sigma}(\mathbf{m})$ for Monte Carlo simulations. Although not strictly necessary for our goal, knowing the standard deviations of the parameter models $\boldsymbol{\sigma}(\mathbf{m})$ will be useful later when setting up the problem constraints and the scaling matrix. Therefore, in the following paragraphs, we will assume that the standard deviations of the parameter models $\boldsymbol{\sigma}(\mathbf{m})$ are known.

Formally, if $\mathbf{C}(\mathbf{m})$ is the function which describes the compact model, then we would ideally like to find a vector **m** such that

$$\mathbf{p} = \mathbf{C}(\mathbf{m}) \tag{1}$$

In reality, (1) might not be verified, either because not enough model parameters are provided or because of the limitations of the compact model itself. In the next section we will show how to find a solution for **m**, including also some additional requirements.

III. METHODOLOGY SETUP

In general, the nonlinear system of equations (1), together with the other inequality constraints which will be identified later in this session, constitutes a nonlinear programming problem, for which several algorithms exist [3-4], including Lagrange multipliers (extended to account for inequality constraints) or numerical methods based on the conjugate gradient method.

To reduce the complexity of the problem let us rewrite first (1) so that it becomes a least squares optimization problem in which we find the set of model parameters **m** which best approximate **p**:

$$\min_{\mathbf{m}} \left\| \mathbf{p} - \mathbf{C}(\mathbf{m}) \right\|_2^2 \tag{2}$$

We would like now to simplify the problem linearizing $\mathbf{C}(\mathbf{m})$ near the point $\mathbf{m}_0$, which is the vector of model parameters which reproduces the nominal values of the PCM data. We will call the latter $\mathbf{p}_0$. Therefore, by definition,

$$\mathbf{p}_0 = \mathbf{C}(\mathbf{m}_0) \tag{3}$$

and

$$\mathbf{C}(\mathbf{m}) \approx \mathbf{C}(\mathbf{m}_0) + \mathbf{S} \cdot \Delta\mathbf{m} \tag{4}$$

wherein **S** is the sensitivity matrix of the system defined by

$$S_{ij} = \frac{\partial p_i}{\partial m_j} \tag{5}$$

We can now rewrite (2) as

$$\min_{\mathbf{m}} \left\| \mathbf{p} - \mathbf{C}(\mathbf{m}) \right\|_2^2 = \min_{\mathbf{m}} \left\| \Delta\mathbf{p} - \mathbf{S} \cdot \Delta\mathbf{m} \right\|_2^2 \tag{6}$$

Wherein $\Delta\mathbf{p} = \mathbf{p} - \mathbf{p_0}$. As the elements of the spec limits vector $\Delta\mathbf{p}$ can differ in their order of magnitude, weighted least squares must be considered. We introduce the diagonal normalization matrix $\mathbf{D} = \mathrm{diag}(1/\Delta p_i)$, and solve

$$\min_{\mathbf{m}} \left\| (\Delta\mathbf{p} - \mathbf{S} \cdot \Delta\mathbf{m}) \cdot \mathbf{D} \right\|_2^2 \tag{7}$$

Equation (7) must be solved considering two additional constraints, which are determined by design requirements. Since we do not want our modelled corner parameters to lie inside the process window (otherwise the corners could not be considered for a worst/best case scenario) we must ensure that they stay outside, writing an inequality in the form

$$\Delta\mathbf{p} \le \mathbf{S} \cdot \Delta\mathbf{m} \tag{8}$$

The signs of the elements of $\mathbf{\Delta p}$ should be changed according to the corner we want to model.

Moreover, the model parameters should only span over the ranges allowed by the compact model, therefore

$$\mathbf{m}_{\min} \le \mathbf{\Delta m} \le \mathbf{m}_{\max} \tag{9}$$

If the only corners of interest are those describing the edges of the process window, stricter bounds can be enforced considering the standard deviations $\boldsymbol{\sigma}(\mathbf{m})$ which come from the correlation analysis. If the width of the process window is λ we could rewrite (9) as

$$-\frac{\lambda}{2}\boldsymbol{\sigma}(\mathbf{m}) \le \mathbf{\Delta m} \le +\frac{\lambda}{2}\boldsymbol{\sigma}(\mathbf{m}) \tag{10}$$

Numerical reasons require that the sensitivity matrix, whose elements can greatly differ in their order of magnitude, is properly scaled. Since the standard deviations $\boldsymbol{\sigma}(\mathbf{m})$ are of the same order of magnitude of the solution $\mathbf{\Delta m}$, we can scale **S** defining a matrix **N** as

$$\mathbf{N} = diag\left[1/\sigma_j(m_j)\right] \tag{11}$$

Introducing the scaled sensitivity matrix $\tilde{\mathbf{S}} = \mathbf{S}\cdot\mathbf{N}$ and the scaled solution $\mathbf{\Delta\tilde{m}} = \mathbf{N}^{-1}\mathbf{\Delta m}$ (7) can be rewritten as

$$\min_{\mathbf{\Delta\tilde{m}}} \left\| \left(\mathbf{p} - \tilde{\mathbf{S}}\cdot\mathbf{\Delta\tilde{m}}\right)\cdot\mathbf{D} \right\|_2^2 \tag{12}$$

Together with the previously defined inequalities, the problem is a constrained least squares problem, for which a solution can be found by numerical means [5-6]. However, it is possible that (8) and (9) do not admit a solution: in this case, it is up to the modelling engineer to look for a compromise and properly relax the constraints.

In the next section, we will show how this methodology, applied to our technology, allows us to generate corner models which adhere to the spec limits.

IV. Results

Different devices realized with the same process can share some parameters, for example the oxide thickness or some offsets in the device geometry; it is then important to group together all the devices with these common variables. The vectors **m** and **p** will therefore refer not just to one single device, but to a whole family with the same process characteristics. The device family for the technology we would like to investigate consists of 45 spec limit parameters, and 30 model parameters have to be extracted.

For the sake of simplicity, however, we will limit the analysis to two different PMOS transistors within the technology (named A and B), which have the same oxide thickness (whose values for the corners are extracted from the minimum and maximum gate oxide capacitance) as well as same length- and width-channel offsets. These common offsets (called wint and lint) are extracted together with the other MOSFET parameters, which are unique for each device, namely the threshold voltage parameter vth0, the mobility u0 and the parameter rdson, which is used in the sub-circuits of the models to model the on-resistance. For both devices, upper and lower limits are considered as shown in table 1, i.e. two threshold voltages (in linear and saturation regime), a saturation current and the on-resistance in the linear regime are considered. Table 2 and 3 summarize the corners setup and show mean-free target values as well as simulated values for the corners. The normalized error in both tables is defined as $E = \left(\Delta p_{\text{tar.}} - \Delta p_{\text{sim.}}\right)/p_{\text{tar.}}$ Figures 2 and 3 show how the threshold voltage in linear region and in the saturation current parameters look like for device A, together with a Monte Carlo simulation. All device simulations are carried out using the Virtuoso Spectre simulator by Cadence [7], the sensitivity analysis is performed by the Muneda tool Wicked [8], which integrates with Virtuoso, whereas a set of Python scripts solve the constrained optimization problem.

The error between the spec limits delta and the simulated corners delta is in general within 25%, whereas the error normalized to the target value is below 6%. With one exception: the on-resistance for the device A is lying quite far from the spec limit (but outside the process window). This can be ascribed to two different issues: first, it is possible that the chosen models parameters are not enough to describe the corners, and this would mean that no better solution exists, unless more are considered. The vector **m** has then to be changed and/or expanded. It should also be noted that the system in (12) should not become under-determined, therefore also other meaningful spec limit parameters should be added to **p**.

The second issue, which can lead to wrong corners, appears when the linear approximation introduced in (4) is not valid anymore. When this happens, the corners can lie inside the process window (this explains the small error in the determination of the linear threshold voltage for the device A and the saturation currents) or too far away from it; the modelling engineer must then artificially change the vector **p** to compensate for this intrinsic limitation and re-iterate the procedure.

TABLE II. Mean-free values and errors for a best case scenario (fast corner)) for two PMOS devices (A and B).

	Target* Δp *FAST	***Simulated* Δp *FAST***	***Error (%)***	***Norm. error (%)***
VTLIN_A	0.105V	0.100V	-4.8	-0.5
VTSAT_A	0.090V	0.108V	20	1.8
ISAT_A	-0.03mA	-0.034mA	13.3	3.1
RON_A	-0.9kΩ	-1.59kΩ	76	16.8
VTLIN_B	0.130V	0.130V	0	0
VTSAT_B	0.130V	0.130V	0	0
ISAT_B	-0.25mA	-0.31mA	24	5.2

RON_B	-80Ω	-88Ω	10	2.1

TABLE III. MEAN-FREE VALUES AND ERRORS FOR A WORST CASE SCENARIO (SLOW CORNER)) FOR TWO PMOS DEVICES (A AND B).

	Target* Δp *SLOW	***Simulated* Δp *SLOW***	***Error (%)***	***Norm. error (%)***
VTLIN_A	-0.105V	-0.105V	0	0
VTSAT_A	-0.090V	-0.106V	17.7	1.7
ISAT_A	0.03mA	0.027mA	-10	-2.3
RON_A	0.9kΩ	0.83kΩ	-7	-1.7
VTLIN_B	-0.130V	-0.130V	0	0
VTSAT_B	-0.130V	-0.130V	0	0
ISAT_B	0.25mA	0.246mA	-16	-0.3
RON_B	80Ω	76Ω	-5	-1.1

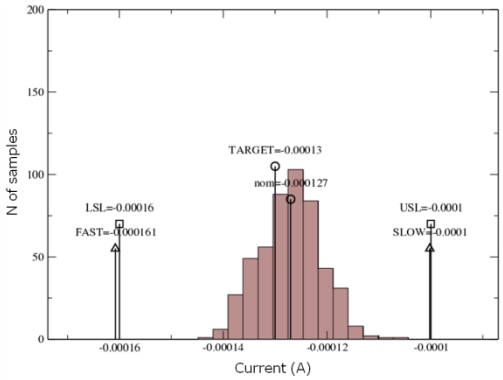

Fig. 2. Saturation current of device A. The y-axis show the number of Monte Carlo iterations (total: 1000). It can be seen that there is a residual difference between the nominal value of the model parameter and the real target value, due to a non-perfect centering.

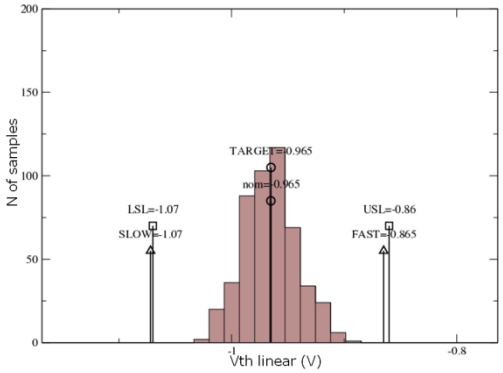

Fig. 3. Threshold voltage in the linear regime of device A. The y-axis show the number of Monte Carlo iterations (total: 1000).

V. CONCLUSION

We have presented a methodology, based on a constrained least squares optimization, which allows a systematic and automatic extraction of the model parameters needed for corners simulation. The results show that simulated and target values are generally in good agreement. User's supervision is anyway required to address the issues which can arise when the linearization approximation is not verified or when the chosen set of parameters is not enough to reproduce the corners.

REFERENCES

[1] Weste, N.H.E.; Harris, D.: CMOS VLSI Design: A Circuits and Systems Perspective, 3rd Ed. Addison-Wesley, 2005.

[2] Pieper, K.-W.; Gondro E.: An effective method for solving the covariance equation for statistical modeling - Semiconductor Conference Dresden (SCD), 2011.

[3] Mordecai, A.: Nonlinear Programming: Analysis and Methods - Dover Publishing, 2003.

[4] Bazaraa, M. S.; Shetty, C. M.: Nonlinear program-ming. Theory and algorithms - John Wiley & Sons, 1979.

[5] Coleman, T.F.; Li, Y.: A Reflective Newton Method for Minimizing a Quadratic Function Subject to Bounds on Some of the Variables - SIAM Journal on Optimization, Vol. 6, Number 4, pp. 1040-1058, 1996.

[6] Gill, P.E.; Murray, W.; Wright, M.H.: Practical Optimization - Academic Press, London, UK, 1981

[7] www.cadence.com

[8] www.muneda.com