

A Methodology for Analog Circuit Macromodeling

Rohan Batra, Peng Li[†] and Lawrence T. Pileggi

Department of Electrical and Computer Engineering
Carnegie Mellon University, Pittsburgh, PA 15213
{rbatra, pli, pileggi}@ece.cmu.edu

Yu-Tsun Chien

SoC Technology Center
Industrial Tech. Research Institute, Taiwan
mozart@itri.org.tw

Abstract — This paper describes a complete framework for generation of compact analog circuit macromodels which significantly reduce the model complexity while still capturing the dominant linear and nonlinear response of the circuit. The technique is applicable to a broad class of circuits that exhibit weakly nonlinear behavior such as mixers, RF power amplifiers and switched-capacitor circuits. The Volterra-based circuit models are first characterized using a combination of SPICE simulation and efficient numerical fitting techniques. The complexity of the extracted circuit models is further reduced by model order reduction techniques while maintaining a high degree of accuracy. The efficacy of our macromodeling methodology is verified by comparison with SPICE simulations. The efficiency of our macromodels makes them suitable for whole-system verification and high-level design analysis.

1. INTRODUCTION

Verifying a complete analog system via transistor-level simulation is an extremely difficult process and can often become infeasible due to the limitation of simulation capacity. A similar difficulty is encountered when high-level design analysis is performed for the whole system. For these reasons, compact macromodels of analog blocks are desired which can be substituted in place of the actual transistor-level netlist to speedup the simulation without sacrificing any of the required accuracy.

The NORM algorithm that was recently proposed in [1] utilizes the Volterra-Series to represent nonlinear transfer functions and employs projection-based techniques to significantly reduce the size of the nonlinear system equations, thereby generating compact representations for analog and RF circuits. This algorithm can be applied to time-invariant as well as time-varying weakly nonlinear circuits. Some extensions have been described in [7][8]. To generate analog macromodels that can be used in commercial simulation environments, the circuits under consideration must be characterized and then modeled based on industry-standard device models in these environments.

In this paper, we develop a complete methodology which extends the NORM algorithm to generate nonlinear reduced-order macromodels directly from transistor-level netlists. The reduced nonlinear macromodels will capture

[†] As of August 2004, the author has joined the Dept. of Electrical Engineering, Texas A&M University, College Station, TX 77843

the nonlinear characteristics of corresponding circuit blocks, such as IIP3, THD and gain compression, in a compact form while maintaining an accuracy comparable to commercial simulators such as SpectreRF and HSPICE. The purpose of developing compact nonlinear analog macromodels is two-fold. Firstly, macromodels can facilitate efficient system-level design exploration by allowing designers to effectively “re-use” the macromodels from their previous designs to predict the system-level behavior. High-level decisions and tradeoff analyses can be made efficiently by evaluating system specifications through the use of a library of “reduced-order” macromodels corresponding to a variety of circuit topologies and configurations. Secondly, compact component macromodels also facilitate the whole-system verification which is otherwise intractable.

2. BACKGROUND

Volterra Series provides an elegant way to characterize weakly nonlinear systems in terms of nonlinear transfer functions. For a circuit with input $u(t)$ the response $x(t)$ can be expressed as the sum of responses at different orders:

$$x(t) = \sum_{n=1}^{\infty} x_n(t) \quad (1)$$

where, $x_n(t)$ is the n th order response. More generally, we can use Volterra Kernels to capture both nonlinearities and dynamics by convolution [3]:

$$x(t) = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} h_n(\tau_1, \dots, \tau_n) u(t-\tau_1) \dots u(t-\tau_n) d\tau_1 \dots d\tau_n \quad (2)$$

where, $h_n(\tau_1, \dots, \tau_n)$ is the n th order Volterra kernel. The frequency domain transform of the n th order Volterra kernel denoted by $H_n(s_1, \dots, s_n)$ is generally referred to as the n th order nonlinear transfer function. These nonlinear transfer functions are independent of the input and fully describe the weakly nonlinear behavior of the circuit. In order to the Volterra nonlinear transfer function for a SIMO weakly nonlinear system we can consider its standard MNA formulation:

$$f(x(t)) + \frac{d}{dt}(q(x(t))) = bu(t), \quad y(t) = d^T x(t) \quad (3)$$

For a circuit with a time-invariant operating condition given by $x = x_0$, the first order linear transfer function is given by:

$$(G_1 + sC_1)H_1(s) = b \quad (4)$$

The symmetrized second order nonlinear transfer function is determined by [1]:

$$[G_1 + \dot{s}C_1]H_2(s_1, s_2) = -[G_2 + \dot{s}C_2] \overline{H_1(s_1) \otimes H_1(s_2)}, \quad (5)$$

where $\dot{s} = s_1 + s_2$ and for equation (4) and (5):

$$G_i = \left. \frac{1}{i!} \frac{\partial^i}{\partial x^i} (f) \right|_{x=x_0}, \quad C_i = \left. \frac{1}{i!} \frac{\partial^i}{\partial x^i} (q) \right|_{x=x_0},$$

$$\overline{H_1(s_1) \otimes H_1(s_2)} = \frac{1}{2} (H_1(s_1) \otimes H_1(s_2) + H_1(s_2) \otimes H_1(s_1))$$

When a nonlinear circuit has a large input excitation, however, it causes the operating points of the active devices to change with time. For example, for a mixer circuit, the operating condition with respect to RF signal is determined by a large LO signal rather than a fixed operating point. This requires the analysis of a small-signal excitation over a large periodic operating condition. Therefore, we can apply a time-varying formulation of the Volterra transfer functions $H_n(t, s_1, s_2, \dots, s_n)$ which can be formulated similar to the time-invariant case [6][7][8].

Volterra based nonlinear descriptions, however, often increase dramatically with problem size, thereby making them ineffective when used directly. Therefore, we instead apply the projection based nonlinear reduced order method (NORM) proposed in [1] to reduce the model size. The algorithm computes a projection matrix by explicitly considering moment-matching of nonlinear transfer functions. For example, if we expand the first-order transfer function $H_1(s)$ at the origin:

$$H_1(s) = \sum_{k=0}^{\infty} s^k M_{1,k}, \quad (6)$$

where $M_{1,k}$ is a kth order moment for the first-order transfer function. Now, expanding the second-order nonlinear transfer function $H_2(s_1, s_2)$ at the origin (0,0):

$$H_2(s_1, s_2) = \sum_{k=0}^{\infty} \sum_{l=0}^k s_1^l s_2^{k-l} M_{2,k,l}, \quad (7)$$

where, $M_{2,k,l}$ is a kth order moment of the second-order transfer function. The actual expression for $M_{2,k,l}$ can be obtained by first substituting (6) into (5) and expanding w.r.t $\dot{s} = s_1 + s_2$. This procedure can be also applied to obtain the moments of the third order transfer functions. In NORM, a projection matrix is built such that the reduced

order model will match certain number of transfer function moments. It has also been demonstrated that multi-point expansion based approach produces much more compact models than the single-point expansion.

3. OVERALL MACROMODELING FLOW

We outline the complete flow for the generation of reduced-order models from transistor-level netlists in Fig. 1. First, we simulate the transistor-level netlist in a commercial simulator such as SPICE to determine a proper operating condition for the circuit. In the case of a time-invariant circuit, a fixed DC operating point will be computed. Otherwise, a large-signal time-varying operating point will be computed for a time-varying circuit such as a mixer. We model the nonlinearities for each transistor in the circuit as a third-order polynomial. We simulate each transistor in the circuit multiple times, varying the bias-voltage for its terminals to generate accurate data-points for fitting the polynomial. We then construct the full Volterra-based model of the circuit and generate the reduced-order model of the circuit using NORM[1].

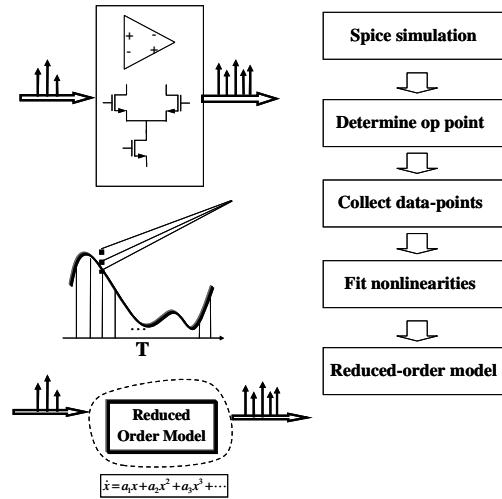


Figure. 1. Extraction of reduced-order model

4. EXTRACTION OF VOLTERRA PARAMETERS

The nonlinear modeling techniques outlined in Section 2 depend on extracting the parameters of the Volterra model accurately. In Volterra series, a nonlinearity is represented as a power series expansion around a bias point. To illustrate, let us consider a nonlinear device characteristic $f(x)$ expanded about a bias point x_0 :

$$f(x) = f(x_0) + a_1(x - x_0) + a_2(x - x_0)^2 + \dots \quad (8)$$

where,

$$a_i = \frac{1}{i!} \left. \frac{\partial^i f(x)}{\partial x^i} \right|_{x=x_0}$$

Many different small-signal models for MOS transistors exist, and most sophisticated models include substrate coupling effects and transcapacitances[2][4]. Spice models like BSIM3 not only represent physical effects but also include many numerical parameters which further increase the complexity of the model equations. It is infeasible to find the coefficients of the equation given in (8) by finding the higher-order derivatives from the model equations in BSIM3 and other models. Instead, we employ least-mean-square error (LMSE) fitting techniques to find the coefficients [2].

We will show how the nonlinear parameters for the drain current of a MOS transistor are extracted. We model the drain current I_{ds} as a third-degree polynomial with respect to the drain, source and gate voltages. For simplicity, we have used the body terminal as the reference voltage although other possibilities can be easily accommodated. The equation includes individual voltage terms as well as cross-terms. Compared to ordinary hand-analysis equations we model not only the first-order nonlinearities but also the second and third-order nonlinearities as small-signal quantities around the bias point:

$$I_{ds} = I_{ds0} + g_d v_d + g_s v_s + g_g v_g + g_{ds} v_d v_s + \dots \quad (9)$$

$$g_{ss} v_s^2 + \dots g_{dds} v_d^2 v_s + \dots g_{sss} v_s^3$$

where,

- I_{ds0} = bias current value at operating point
- v_x = small-signal voltage at terminal $x = \{d, g, s\}$
- g_x = first-order coefficient for voltage at terminal x
- g_{xy} = second-order coefficient for cross-product of voltages at terminals x and y
- g_{xyz} = third-order coefficient for cross-product of voltages at terminals x , y and z

Therefore, there are 3 first-order terms, 6 second-order terms and 10 third-order terms in the equation.

It is not possible to get the second and third-order terms directly from transistor-level simulation so we have formulated an efficient way to get these terms and model the nonlinearity accurately. We extract the first-order model parameters from Hspice simulation[5]. For a time-invariant circuit, we perform a DC operating point analysis to obtain the bias current value and the first-order coefficients. For a time-varying circuit, we perform a single-tone transient analysis for a sufficient settling time and then sample a single time-period of the settled response to obtain time-varying operating points for the

circuit. We then perform a DC operating-point analysis at each of these points to get the first order coefficients g_d , g_s and g_g . We can express these coefficients in terms of the more commonly used small-signal coefficients G_m , G_{ds} and G_{mbs} :

$$g_g = G_m, \quad g_d = G_{ds} \quad \text{and} \quad g_s = -(G_{ds} + G_m + G_{mbs}) \quad (11)$$

For both the time-invariant and time-varying cases, the bias voltages for each transistor are perturbed by small-amounts to obtain data-points for fitting the second and third-order coefficients in the appropriate fitting range represented by the bounding box shown in Fig. 2. The figure shows the drain current as a function of v_{ds} and v_{gs} . It is possible to measure the current I_{ds} by perturbing the v_{ds} and v_{gs} slightly around each bias-point to obtain many different points. From (9):

$$(I_{dsi} - I_{ds0}) = g_d v_{di} + g_s v_{si} + g_g v_{gi} + g_{ds} v_{di} v_{si} + \dots \quad (10)$$

where, the subscript i denotes the i -th data-point and $(I_{dsi} - I_{ds0})$ is called the ‘‘residue’’. To solve the coefficients of the RHS in equation (10) we write the powers and cross-terms of v_d , v_g and v_s for n sampling points into matrix Y , the corresponding coefficients into the vector p and the residue $(I_{ds} - I_{ds0})$ into matrix R :

$$Y = \begin{bmatrix} v_{d1} & v_{s1} & v_{g1} & v_{d1}^2 & v_{d1} v_{g1} & \dots & v_{s1}^3 \\ v_{d2} & v_{s2} & v_{g2} & v_{d2}^2 & v_{d2} v_{g2} & \dots & v_{s2}^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{dn} & v_{sn} & v_{gn} & v_{dn}^2 & v_{dn} v_{gn} & \dots & v_{sn}^3 \end{bmatrix}$$

$$p = [g_d \quad g_s \quad g_g \quad g_{dg} \quad \dots \quad g_{sss}]$$

$$\text{and} \quad R = [I_1 \quad I_2 \quad \dots \quad I_n]^T \quad I_n = I_{dsn} - I_{ds0} \quad (11)$$

We have to fit the coefficients of (10) such that the error for each of the n data-points around the operating point is minimized. The aggregate error for the i th data-point is denoted by \mathcal{E}_i . We have to minimize the error $e = [\mathcal{E}_1 \quad \mathcal{E}_2 \quad \dots \quad \mathcal{E}_n]^T$:

$$Yp - R = e \quad (12)$$

The LMSE algorithm estimates p by minimizing the sum of squared errors:

$$F = e^T e = (Yp - R)^T (Yp - R) \quad (13)$$

This leads to the optimal solution:

$$p = (Y^T Y)^{-1} \cdot (Y^T R) \quad (14)$$

In order to guarantee a good fit for the nonlinearities we ensure that the fitting range for the data is correct [2] (Fig. 2). The range must be large enough to fit the nonlinearities

accurately but it should not attempt to cover the effects outside the signal-swing range. For example, if the gate of a transistor has an expected signal swing of $\pm 10\text{mV}$, the fitting range for v_g of this transistor should be limited by the signal swing. It is also imperative to select enough data-points to fit the nonlinear parameters accurately.

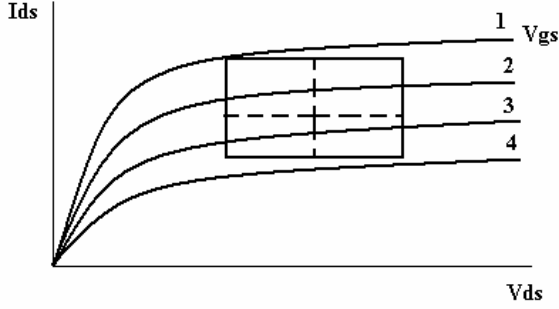


Figure. 2. Effective fitting range for Volterra parameters

In some cases, the fitted results might still cause large relative errors for certain points in the data. The fit may be improved by using a weighted-least squares method instead of the conventional method. In this case, it is important to select individual weights for each equation:

$$w_i(I_{dsi} - I_{ds0}) = g_d v_{di} w_i + g_s v_{si} w_i + \dots + g_{ds} v_{di} v_{si} w_i + \dots \quad (15)$$

such that the effective residue $w_i(I_{dsi} - I_{ds0})$ for each data-point is in the same range. This weighting scheme gives each individual data-point the same importance as far as the fitting process is concerned. This aids in reducing the error for each point and gives a better fit for the entire range of data-points. First, we can perform a LMSE fit on the data to get an initial estimate of the points where the error \mathcal{E}_i may be large. We select the appropriate weight w_i for each data-point and scale the residue accordingly. After performing the weighted least-squares fit using (16) and (17) we can look at the error \mathcal{E}_i again and modify the weights if we are still not satisfied with the results. This is done for a few iterations till we can no longer improve the results. For the weighted least-squares method we introduce another matrix W , which is the diagonal matrix of the individual weights. We have to minimize the weighted least squares error function:

$$F = (Yp - R)^T W (Yp - R) \quad (16)$$

where Y, p and R have been defined earlier. This can be solved to get the nonlinear parameters in (9):

$$p = (Y^T W Y)^{-1} \cdot (Y^T W R) \quad (17)$$

4. RESULTS

The methodology presented in the previous sections has been demonstrated on a double-balanced mixer and an opamp. The macromodels generated using this approach are compared with detailed transistor-level simulation of the circuits with HSPICE.

4.1 A Double-Balanced Mixer

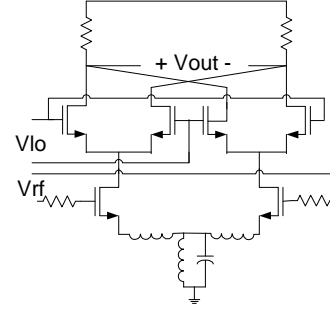


Figure. 3. A Double-Balanced Mixer

A double-balanced mixer (Fig. 3) is modeled as a time-varying weakly nonlinear system with respect to the RF input. The LO frequency is set at 1GHz, and we calculate the time-varying operating point of the circuit by setting the RF input voltage to zero and using transient analysis in HSPICE to sample a single time-period of the settled response. The third order nonlinearities are modeled around this time-varying operating point using numerical fitting techniques outlined in Section 4. The fitted second and third order coefficients are used to generate a Volterra-based full model for the circuit.

A single-tone RF input is used to verify the model results with the transient simulation results. The third-order harmonic of the RF input frequency down-converted with respect to the LO frequency is compared between the model and the simulation results. The second order nonlinearities should ideally be zero except for numerical noise, by design. We performed transient analysis for the circuit in Hspice followed by an accurate Fourier Transform of the output time-domain waveform to verify the results. The RF input frequency is varied from 300Mhz to 1200Mhz. The maximum error in the full model as compared to Hspice simulation for the first-order results is less than 2% for all frequencies. The maximum error in the third-order results is less than 10% for third-order for all frequencies. The results in Fig. 4 have been normalized with respect to the RF input amplitude.

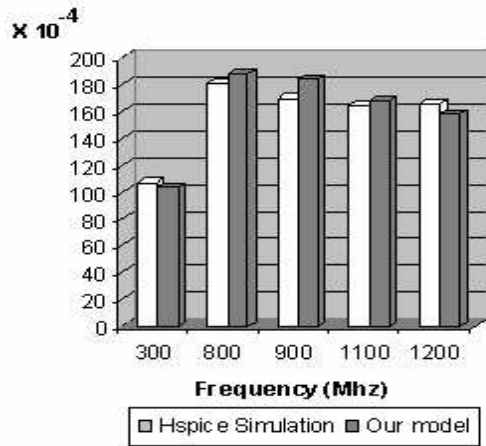


Figure. 4. Third-order harmonic (down-converted) for different input frequencies

Once we have the Volterra Series based full model, it is possible to measure the third-order response at more useful harmonics also. For example, Fig. 5 shows the plot for the third-order transfer function $H_3(t, j2f_1, j2f_2, j2f_3)$ where, $300\text{Mhz} \leq f_1, f_2 \leq 1200\text{Mhz}$ and $f_3 = -900\text{Mhz}$. The full model has 1350 time-sampled circuit unknowns which is reduced to approximately 14 circuit variables using the NORM method, while still capturing the dominant response of the circuit. The relative modeling error between the full and reduced-order model for the first-order results is less than 0.01%. Fig. 6 shows that the relative percentage error between the full-model and reduced-order model for the third-order results is less than 6% for all cases.

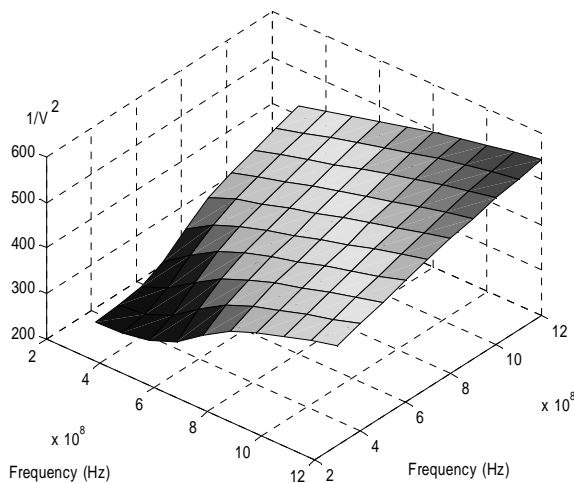


Figure. 5. Third-order transfer function for mixer(full)

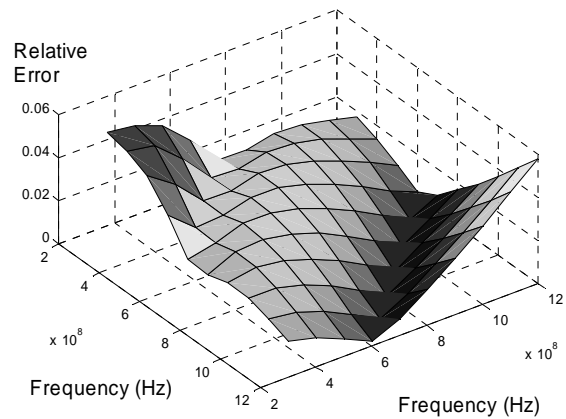


Figure. 6. Relative modeling error for third-order transfer function

4.2 An Operational Amplifier

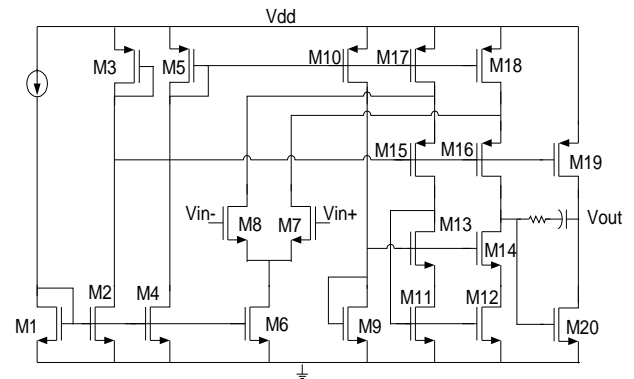


Figure. 7. A two-stage opamp

A two-stage Operational Amplifier topology is shown in Fig. 7. The closed-loop AC response of this circuit is shown in Fig. 8. Using the extraction method described in Section 4, it is possible to match the AC (first order response) of the circuit accurately to about 99-100% compared with Hspice simulation. For this circuit, second order nonlinearities are more important than the third-order nonlinearities since they are much higher in magnitude. The opamp is modeled as a time-invariant system and is linearized at the DC bias point to fit the second and third order coefficients for each transistor in the circuit.

The second-order distortion for a single-tone input is shown in Fig. 9. We compared the Hspice simulation results for input frequencies ranging from 1Mhz to 100Mhz with our model results. The relative error between the full-model and the simulation results is less than 10% for all input frequencies. The number of state-variables for the circuit reduced from 22 in the full-model

to about 5 in the reduced order model. The comparison of the full and reduced order model results shows that there is less than 0.01% error for both first and second-order responses.

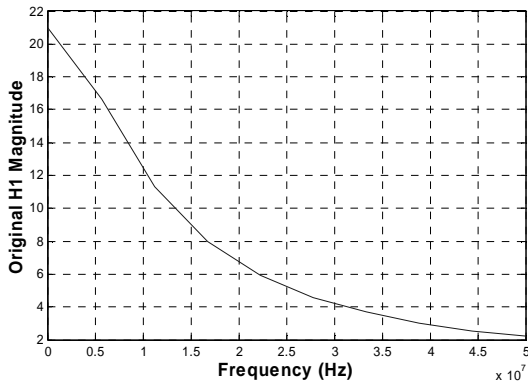


Figure 8. Closed-loop AC response of the opamp

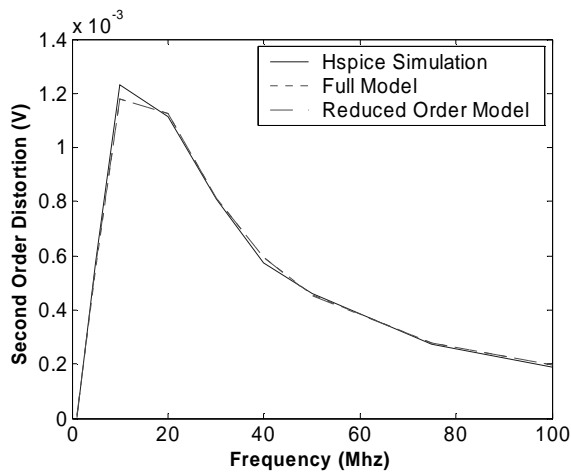


Figure 9. Second order distortion as a function of frequency

5. CONCLUSIONS

In this paper, we have presented a methodology for generating analog circuit macromodels from the transistor-level netlists. This methodology can be applied to a broad range of time-invariant and time-varying weakly nonlinear circuits. The macromodels generated using this methodology are characterized using efficient numerical fitting of simulation data and model order reduction techniques. Our experimental results have shown that the macromodels offer significant decrease in model size with comparable accuracy to full transistor-level simulation in Hspice. We would like to further explore the possibility of

adopting these compact macromodels in behavioral languages such as Verilog-A.

6. REFERENCES

- [1] P. Li and L. Pileggi, "NORM: compact model order reduction of weakly nonlinear systems," in *Proceedings of ACM/IEEE DAC*, 2003.
- [2] J. Vuolevi and T. Rahkonen, "Distortion in RF Power Amplifiers", Artech House, 2003.
- [3] W. Sansen and P. Wambacq, "Distortion Analysis of Analog Integrated Circuits", Kluwer Academic Publishers, 1998.
- [4] Y. Tsividis, "Operation and Modeling of the MOS Transistor", McGraw-Hill, 1999.
- [5] Hspice User's Manual, Pg. 3-24, Version I, 1996
- [6] J. Roychowdhury, "Reduced-order modeling of time-varying systems", *IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 10, Oct. 1999.
- [7] P. Li and L. Pileggi, "Modeling Nonlinear Communication IC's using a Multivariate Formulation" in *BMAS workshop 2003*.
- [8] P. Li, X. Li, Y. Xu and L. Pileggi, "A hybrid approach to nonlinear macromodel generation for time-varying analog circuits," in *Proceedings of ACM/ICCAD*, 2003.