Efficient Multi-Tone Harmonic Balance Simulation of Semiconductor Devices in the Presence of Linear High-Q Circuitry

Boris Troyanovsky

EEsof Division,
Hewlett-Packard Co.,
1400 Fountaingrove Pkwy,
Santa Rosa, CA 95403
(formerly with Stanford Univ.)

Francis Rotella
Zhiping Yu
Robert W. Dutton

Center for Integrated Systems,
Stanford University, Stanford, CA 94305

Torkel Arnborg

Ericsson Components AB,
Microelectronics Research Center,
S-16481 Kista, Sweden

Abstract -- The harmonic balance method is a frequency domain analysis technique for simulating the large signal steady state behavior of nonlinear systems. The harmonic balance analysis capability has been used for some time now in the PISCES-2H device simulator to simulate harmonic and intermodulation distortion at the physical level. In this paper, we present extensions to the harmonic balance algorithm which greatly reduce the computational requirements of applying it to large scale multi-tone simulations of single-device amplifiers and mixers, particularly in the cases where high-Q circuitry is present around the device.

I. INTRODUCTION

Large-signal distortion effects are critical to the proper operation of nonlinear analog circuits, particularly in such application areas as RF/microwave communication systems. In some applications, such as highly linear amplifier design, the goal is to minimize any distortion that appears in the amplified output waveform. In other designs, such as mixers or frequency-doublers, nonlinearities are used in conjunction with linear filtering networks to introduce desired harmonics into the output signal, while at the same time suppressing undesired spurious distortion products.

Our work has focused on modeling the semiconductor device nonlinearities at the physical level, by solving the time-dependent drift-diffusion equations on a two-dimensional grid [1][2]. In [1], Krylov subspace techniques were used to solve the extremely large systems of harmonic balance equations arising from two-dimensional physics-based device simulation. It was demonstrated that standard block-diagonal preconditioners were too memory-intensive for use in multi-tone device distortion problems, and a so-called *sectioned block-diagonal preconditioner* was introduced to remedy the problem. Although highly effective for wide-band amplifier applications, the aforementioned preconditioner can run into difficulties in the important case where mixers or tuned amplifiers employ high-Q circuitry to filter out undesired distortion products. In this paper, we present an improved preconditioner to successfully handle the high-Q circuitry problem in the multi-tone case at essentially the same cost as the sectioned block-diagonal preconditioner of [1]. In addition, we introduce other algorithmic improvements to drastically lower the memory usage required by the simulator. Application examples and benchmarks are provided to validate the algorithmic improvements and highlight their areas of applicability.

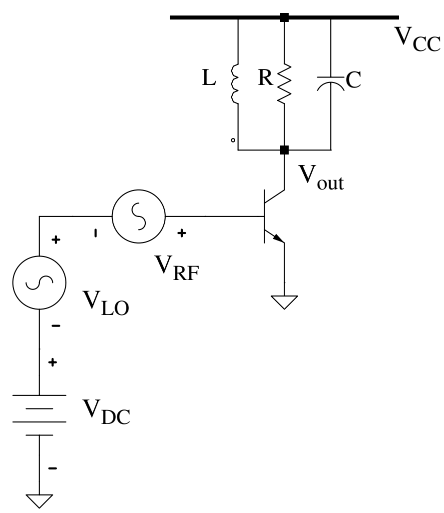

Figure 1. A single-device BJT mixer. The resonant circuit at the output is tuned to either the sum or the difference frequency of the LO and RF, depending on the application.

II. FREQUENCY DOMAIN ANALYSIS -- MOTIVATION

Before proceeding with a description of the algorithms, we briefly review the motivation for using harmonic balance through a simple but illustrative example. Consider the single-BJT mixer of Figure 1 (ref. [9], pp. 439-448). Given a local oscillator frequency f_{LO} and an input RF frequency f_{RF}, the mixer design seeks to downconvert the RF signal to a frequency $f_{IF} = f_{RF} - f_{LO}$. (As an example, we could take f_{RF} as 500 MHz + 100 kHz, f_{LO} as 500 MHz, and f_{IF} as 100 kHz). A resonant RLC filter is used to filter out the undesired image term $f_{IF} = f_{RF} + f_{LO}$ along with the local oscillator feedthrough component f_{LO}, and other distortion products. For relatively tightly spaced LO and RF signals, the filter employed must be a narrow bandpass filter with a high ''quality factor'' Q. The simulation of such a mixer circuit poses considerable difficulties for standard time domain transient approaches, as the narrow bandpass filtering circuitry introduces extremely long time constants into the circuit. Because the designer is primarily interested in the circuit's steady-state behavior, a transient analysis would need to be run over many periods of the stimulus, until the transient response decayed completely. A potentially even more severe problem occurs due to the fact that the desired output frequency $f_{IF} = f_{RF} - f_{LO}$ will be many orders of magnitude smaller than f_{RF} and f_{LO}. A transient simulator must place its time points finely enough to resolve the high-frequency signals f_{RF}/f_{LO}, while covering many periods of the relatively low frequency f_{IF}. The number of time points necessary for such an analysis is prohibitively large.

To address the aforementioned shortcomings of time domain approaches, a harmonic balance capability was implemented on top of the PISCES-2H device simulator [1][2]. Harmonic balance (HB) is a nonlinear frequency domain analysis technique that offers several advantages over standard time domain transient approaches. Perhaps chief among these is its ability to handle input signals with very widely or very tightly spaced incommensurate frequencies, as well as its effectiveness in directly capturing the steady state of systems with widely-varying time constants. In addition, harmonic balance exhibits excellent dynamic range for resolving low-level distortion products, and is robust enough to handle the levels of nonlinearity typically seen in the majority of RF/microwave applications.

III. Applying Harmonic Balance to the Semiconductor Equations

A. Fundamentals

The fundamental algorithms and physical models used for harmonic balance device simulation have been presented in an earlier paper [1]. Here, we once again review the fundamentals. The PISCES-2H device simulator solves the time-dependent drift-diffusion equations

$$\nabla \bullet (-\varepsilon \nabla \psi) = q(p - n + N_D^+ - N_A^-) \qquad (1)$$
$$\frac{\partial n}{\partial t} = \frac{1}{q} \nabla \bullet J_n - U$$
$$\frac{\partial p}{\partial t} = -\frac{1}{q} \nabla \bullet J_p - U$$

where

$$J_n = q D_n \nabla n - q \mu_n n \nabla \psi \qquad (2)$$
$$J_p = -q D_p \nabla p - q \mu_p p \nabla \psi$$

In addition, arbitrary linear circuit networks can be included outside the semiconductor device structure, and handled via constitutive equations based on Kirchoff's Current Law (KCL). For the configuration of Figure 1, the stimulus is a two-tone sinusoidal voltage source having the form

$$V(t) = V_{DC} + V_{RF}\cos(\Omega_{RF} t + \varphi_{RF}) + V_{LO}\cos(\Omega_{LO} t + \varphi_{LO}) \quad . \qquad (3)$$

Henceforth, we will refer to the fundamental stimulus frequencies as Ω_1 and Ω_2 for two-tone problems.

The method of harmonic balance solves the system of equations (1) in the frequency domain. The partial differential equations are discretized spatially, while the time-dependent portion of the solution is expanded in sines and cosines. For H harmonics, we have

$$x_n(t) = X_{n0}^R + \sum_{h=1}^{H} (X_{nh}^R \cos(\omega_h t) - X_{nh}^I \sin(\omega_h t)) \qquad (4)$$

where $X_{nh} = X_{nh}^R + jX_{nh}^I$. In the device simulation problem, the state variables at each internal node k are (ψ_k, n_k, p_k), representing electrostatic potential, electron concentration, and hole concentration (in that order). In addition, there are state variables v_c representing voltages at each external node connecting the device to its surrounding linear network. Assuming a total of K internal nodes and C terminals attached to linear network ports, the time-domain state vector of length $N = 3K + C$ takes the form

$$\boldsymbol{x} = [\psi_1, n_1, p_1, \ldots, \psi_K, n_K, p_K, v_1, \ldots, v_C] \, . \qquad (5)$$

The corresponding harmonic balance state vector $\boldsymbol{X}$ thus has a length of $(3K + C)(2H + 1)$, as it requires $2H + 1$ values to represent each state variable waveform $x_n(t)$ in (4).

Assuming a two-tone stimulus, the set of frequencies present in the waveforms $x_n(t)$ is

$$\omega_{k_1 k_2} = k_1 \Omega_1 + k_2 \Omega_2 \qquad (6)$$

for integer combinations of k_1 and k_2. In principle, there is an infinite number of such frequencies. In practice, only a finite number of these spectral components contain significant energy. Thus, we employ the so-called diamond truncation scheme [7] to restrict the frequency components to

$$\omega_{k_1 k_2} = k_1 \Omega_1 + k_2 \Omega_2 \text{ s.t. } |k_1| + |k_2| \le P \, , \qquad (7)$$

where P is the truncation order. In general, this set of frequencies is not harmonically related to a single fundamental, and thus the standard Discrete Fourier Transform (DFT) cannot be applied directly. To circumvent this problem, and hence enable use of the efficient FFT algorithm, we use the ''frequency remapping'' technique of Hente and Jansen [8]. The remapping technique is applicable to arbitrary numbers of independent input tones, altough the resulting packing is no longer dense for problems with three or more independent fundamentals.

In order to determine the set of harmonic balance equations, we first note that, after discretization, (1) can be written in the time domain as

$$\delta_{cont}\frac{dx_n}{dt} + g_n(\boldsymbol{x}(t)) - w_n(t) = 0 \ , \quad 1 \le n \le 3K. \quad (8)$$

In the equation above, δ_{cont} is zero for the Poisson equation, and unity for the continuity equations. The term $g_n(\boldsymbol{x}(t))$ represents the *n*th discretized semiconductor equation, not counting the continuity equation derivative terms, and $w_n(t)$ is a driving function that represents external sinusoidal excitation. Each of these time domain equations is sampled and then Fourier-transformed into $H+1$ frequency domain relations of the form

$$j\omega_h\delta_{cont}X_{nh} + G_{nh}(\boldsymbol{X}) - W_{nh} = 0 \, , \, 0 \le h \le H \, , \quad (9)$$

where $G_{nh}(\boldsymbol{X})$ is the *h*th harmonic of $g_n(\boldsymbol{x}(t))$, and W_{nh} is the *h*th harmonic of $w_n(t)$. This is the set of harmonic balance equations which is to be solved for $\boldsymbol{X}$.

B. Solving the Harmonic Balance Equations

The nonlinear set of harmonic balance equations are most effectively solved by means of the Newton-Raphson method. When combined with source-stepping or arclength continuation algorithms, Newton-Raphson is an extremely robust technique that exhibits locally quadratic convergence near the solution.The *l*th iteration of Newton-Raphson takes the form

$$\boldsymbol{X}^{(l)} = \boldsymbol{X}^{(l-1)} - \left[\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{X}}(\boldsymbol{X}^{(l-1)})\right]^{-1}\boldsymbol{X}^{(l-1)} \quad , \quad (10)$$

where $\boldsymbol{F}$ is defined as

$$F_{nh} = j\omega_h\delta_{cont}X_{nh} + G_{nh}(\boldsymbol{X}) - W_{nh} \, , \quad (11)$$

and $\boldsymbol{X}^{(l)}$ is the state vector at iteration *l*, with $\boldsymbol{X}^{(0)}$ being the initial guess.

The harmonic balance Jacobian matrix

$$J = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{X}} \quad (12)$$

has dimensions $N(2H+1) \times N(2H+1)$. To each structurally non-zero entry $\partial g_n / \partial x_m$ of the time domain Jacobian corresponds a dense $(2H+1) \times (2H+1)$ block $\partial \boldsymbol{G}_n / \partial \boldsymbol{X}_m$. Each such block consists of derivatives of the harmonics of the *n*th residual with respect to harmonics of the *m*th state variable. As discussed earlier, factorization of the harmonic balance Jacobian by direct methods is prohibitively expensive. Preconditioned Krylov subspace methods have been shown to be extremely effective when applied to solving large-scale HB systems of circuit [4] and device [1] equations. In our work, we use the restarted GMRES algorithm [6] as the Krylov subspace solution method of choice. The reader is referred to [1] for a more thorough discussion of the topic.

III. ALGORITHMIC ADVANCES

A. Sectioned Preconditioners -- A Review

Linear iterative solvers based on Krylov subspace methods are not in general effective when applied directly to the harmonic balance Jacobian $J = \partial \boldsymbol{F} / \partial \boldsymbol{X}$. To accelerate the rate of convergence, a preconditioning matrix *M* must be employed. This matrix should ideally have the property that it is ‘‘close’’ to *J* in the sense that $M^{-1}J \approx I$, while also being much easier to factor than *J* itself. Once such a matrix has been found and factored, the (left-)preconditioned linear system

$$M^{-1}J\boldsymbol{x} = M^{-1}\boldsymbol{b} \quad (13)$$

is solved in lieu of the original system $J\boldsymbol{x} = \boldsymbol{b}$.

In circuit simulation, it is common to precondition the harmonic balance system by $H+1$ AC Jacobians -- i.e., one AC Jacobian for each harmonic balance analysis frequency [4]. While this approach can also be effective in the device simulation arena, the large size of typical semiconductor device Jacobians precludes the storage and factorization of such a large number of matrices in multi-tone problems (recall that the number of harmonics *H* rises as $O(P^2)$ for truncation order *P* in two-tone HB problems, and can easily exceed *H*=100). To deal with this problem, the sectioned block-diagonal preconditioner was introduced in [1]. In the rest of this section, we review this preconditioning method, expose its shortcomings when high-Q linear circuitry is present, and introduce a new preconditioner to remedy the problems.

For notational convenience, let $J_{AC}(\omega_h)$ denote the AC matrix at frequency ω_h, $0 \le h \le H$. To avoid storing and factoring all $H+1$ of these matrices, we note that most two-tone problems have ‘‘bands’’ of harmonics that are very tightly spaced in frequency. For instance, consider an input of the form (3), where $\Delta\Omega = \Omega_2 - \Omega_1$ is small (take $\Omega_1 = \Omega_{LO}$ and $\Omega_2 = \Omega_{RF}$ for our example of Figure 1). By simple algebraic manipulation, the set of harmonic balance frequencies (6) can be written as

$$\omega_{k_1k_2} = (k_1 + k_2)\Omega_1 + k_2\Delta\Omega \ . \quad (14)$$

We proceed to define the $P+1$ section frequencies $p\Omega_1$, $0 \le p \le P$, and, for each HB frequency ω_h, pick the closest section frequency[1] $p_h\Omega_1$. Each of the AC Jacobians is then approximated by a section Jacobian

$$J_{AC}(\omega_h) \approx J_{AC}(p_h\Omega_1) \ . \quad (15)$$

In this manner, only $P+1$ AC Jacobians need to be stored and factored.

B. Sectioned Preconditioners in the Presence Of High-Q Linear Circuitry

The sectioned preconditioner of the preceding paragraph performs extremely well so long as the semiconductor de-

1. Alternately, the section frequencies can be chosen to be the centers of the frequency bands (dashed lines in Figure 2).

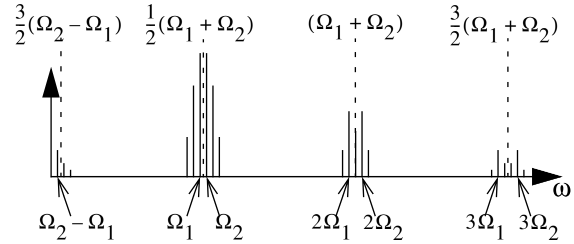

Figure 2. An illustration of tightly spaced frequency bands that occur when $\Delta\Omega = \Omega_2 - \Omega_1$ is small in a two-tone stimulus.

vice is not surrounded by linear circuitry with sharply varying impedances. This is due to the fact that the drift-diffusion Jacobian varies slowly with AC frequency [1], and is thus approximately constant across the narrow frequency bands that occur in closely-spaced two-tone problems. In mixer simulations involving high-Q filters, however, the assumption of an approximately constant AC Jacobian across a given frequency band no longer holds true. Indeed, as discussed in the introduction, the whole aim of such a high-Q filter is to discriminate between closely spaced frequency components within a narrow frequency band. As may be expected, the standard sectioned diagonal preconditioner rapidly loses its effectiveness in such situations.

To remedy the problem, we note that the AC Jacobian for the aforementioned system may be written as a block matrix of the form

$$J_{AC}(\omega) = \begin{bmatrix} J_{DD}(\omega) & J_{12}(\omega) \\ J_{21}(\omega) & Y(\omega) \end{bmatrix} , \tag{16}$$

where $J_{DD}(\omega)$ is a large $3K \times 3K$ matrix representing the drift-diffusion portion of the system, while $Y(\omega)$ is a small $C \times C$ matrix representing the linear circuitry around the device. In typical semiconductor device problems, the dimensionality of J_{DD} is in the thousands, while the size of Y is typically 3×3 or less. The large drift-diffusion block J_{DD} continues to vary slowly with frequency, while the matrix Y may vary sharply when high-Q linear circuitry is present. To efficiently factor $J_{AC}(\omega_h)$, we can approximate it as

$$J_{AC}(\omega_h) \approx \begin{bmatrix} J_{DD}(p_h\Omega_1) & J_{12}(p_h\Omega_1) \\ J_{21}(p_h\Omega_1) & Y(\omega_h) \end{bmatrix} \tag{17}$$

The large drift-diffusion portion of the Jacobian is stored only at the $P+1$ section frequencies, while the extremely small linear portion is evaluated at all $H+1$ frequencies of interest. Block LU-factorization algorithms may then be applied to (17). The effectiveness of this preconditioner is illustrated in Section IV.

C. Efficient Storage of the Jacobian Time Samples

Recall from [1] that matrix-vector products of the form

$$\frac{\partial \boldsymbol{G}_n}{\partial \boldsymbol{X}_m}\boldsymbol{Z}_m = \Gamma \begin{bmatrix} g_{nm}(t_0) & \dots & 0 \\ | & \backslash & | \\ 0 & \dots & g_{nm}(t_{2H+1}) \end{bmatrix} \Gamma^{-1}\boldsymbol{Z}_m , \tag{18}$$

are required by the GMRES solution algorithm, where Γ represents the Fast Fourier Transform matrix, $\boldsymbol{Z}_m$ is an arbitrary complex vector, and the time-varying Jacobian entries are written as $g_{nm}(t) = \partial g_n(t)/\partial x_m$ for notational convenience. To carry out such matrix-vector products efficiently, the samples of $\partial g_n/\partial x_m$ can be computed once and stored for the duration of the iterative linear solve.

Due to the large-size of two-dimensional device simulation problems, however, storing all $2H+2$ time samples can become prohibitively expensive when large numbers of harmonics are required. To reduce the memory requirements associated with storing the Jacobian time samples, we transform them into the frequency domain, and observe that for most Jacobian entries the resulting spectrum is quite sparse. The spectral components below a certain user-defined threshold are discarded, and the compact frequency domain representation is then stored. Although in theory the resulting Jacobian becomes only approximately correct, experience has shown that memory requirements can be significantly reduced without a significant impact on Newton-Raphson's quadratic rate of convergence.

IV. Results

Several examples of harmonic balance based device simulation have been presented previously [1][2][3]. In this section, we present a simple mixer circuit simulation where a high-Q filter is employed. The simple example is presented for illustrative purposes only; more realistic circuit configurations will be presented at the conference. [Note to reviewers: More realistic examples will also be present in the final version of this paper]. All simulation results reported in this section have been obtained on an HP-J210 workstation with a 120MHz CPU and 250MB of RAM. The memory usage below includes an additional 30MB of PISCES static arrays which are not used by the harmonic balance engine. In principle, the memory usage presented below could be reduced by 30MB with the same algorithms applied to a modified PISCES code.

A single-BJT mixer circuit is taken directly from [9], p. 445. The configuration is shown in Figure 1 of this paper. The goal of the mixer circuit is to downconvert a 500.1 MHz RF signal down to a 100 kHz IF, using a 500 MHz LO. A resonant RLC circuit having Q=$2\pi 100$ is used (R = 15 kΩ, C = 66.667 nF, L = 37.995 μH). The resonant frequency is 100 kHz, and the high Q effectively filters out the IF-band distortion products. The voltage source parameters used are

$V_{CC} = 10V$, $V_{DC} = 0.7V$ $V_{LO} = 0.15V$ at 500 MHz, and $V_{RF} = 0.05V$ at 500.1 MHz. A silicon BJT from [2] was employed, resulting in a total (DC) system size of $N = 2114$. A harmonic balance analysis at $H = 110$ frequencies was carried out, for a total harmonic balance problem size of $(2H+1)N = 467, 194$. A simulation using the improved sectioned preconditioner required 53 min. 57 sec. and 191 MB RAM (only 153 MB of which was resident at any one time). In contrast, simulation using the old variant of the sectioned preconditioner did not converge during the GMRES linear solves due to the presence of the resonant circuit. Simulation using a full (non-sectioned) block diagonal preconditioner at all $H+1$ frequencies was impossible due to the excessive amount of memory required.

The results of the simulation are presented in Figure 3 and Figure 4. Only the relevant baseband portions of the spectrum are shown in the figures, due to the very large differential in the frequencies present. The large levels of distortion present in the BJT's collector current (Figure 4) are effectively filtered out using the RLC resonator. Consequently, the 200 kHz distortion component is over 70 dB down relative to the desired IF signal at 100 kHz (Figure 4). The distortion level could be further reduced by using a resonator with a higher Q, or by reducing the relatively high RF drive level.

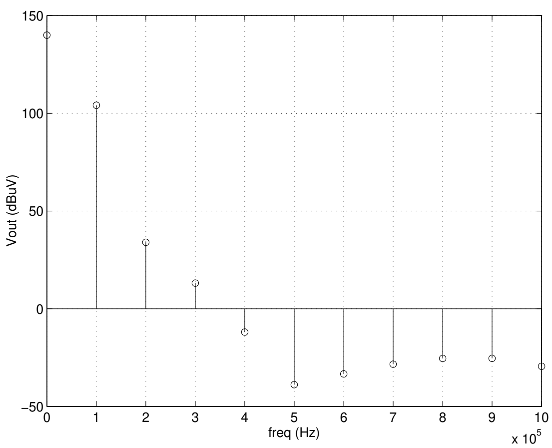

Figure 3. Baseband spectrum of the output voltage. Note the suppression of distortion components by the resonant circuit.

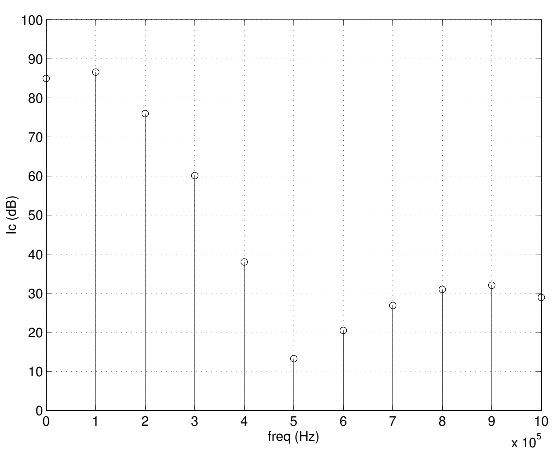

Figure 4. Baseband collector current spectrum of the BJT. The significant distortion components indicate that a filter is needed to remove them from Vout.

V. Conclusion

This paper presented algorithmic improvements and extensions to previous work in harmonic balance device simulation [1]. In particular, algorithms were introduced to allow dramatic reductions in memory usage for multi-tone HB simulations involving high-Q circuit networks around the semiconductor device. A new sectioned preconditioner was introduced which is applicable to arbitrary frequency characteristics in the surrounding linear network, and improved algorithms for storing the harmonic balance Jacobian were described.

Acknowledgements

The authors wish to thank the Center for Integrated Systems at Stanford (CIS) for support of this work and the collaborations between Ericsson and Stanford. The PISCES program has been developed with partial support from SRC. The support from HP EEsof in collaborations with Stanford in creating the PISCES-HB software prototype is gratefully acknowledged. Boris Troyanovsky would like to thank Jeffrey W. Meyer and Niranjan Kanaglekar, his managers at HP EEsof, for their support. He would also like to thank David D. Sharrit of HP EEsof for many valuable technical discussions.

References

[1] B. Troyanovsky, F. Rotella, Z. Yu, R. W. Dutton, and J. Sato-Iwanaga, "Large Signal Analysis of RF/Microwave Devices with Parasitics Using Harmonic Balance Device Simulation," Proc. SASIMI '96, Nov. 1996.

[2] B. Troyanovsky, Z. Yu, L. So, and R. W. Dutton, "Relaxation-based harmonic balance technique for semiconductor device simulation," *Proc. IEEE/ACM International Conf. on Computer-Aided Design*, pp. 700-703, Nov. 1995.

[3] J. Sato-Iwanaga, K. Fujimoto, H. Masato, Y. Ota, K. Inoue, B. Troyanovsky, Z. Yu, and R. W. Dutton, "Distortion Analysis of GaAs MESFETs Based on Physical Model Using PISCES-HB," Proc. IEDM, pp. 163-166, Dec. 1996.

[4] R. Melville, P. Feldmann, and J. Roychowdhury, "Efficient multi-tone distortion analysis of analog integrated circuits," *Proc. of the IEEE Custom Integrated Circuits Conference*, pp. 241-244, May 1995.

[5] P. Feldmann, R. Melville, and D. Long, "Efficient frequency domain analysis of large nonlinear analog circuits," *Proc. of the IEEE Custom Integrated Circuits Conference*, pp. 461-464, May 1996.

[6] Y. Saad and M.H. Schultz, "GMRES: a generalized minimal residual method for solving nonsymmetric linear systems," SIAM J. Sci. Stat. Comput., Vol. 7, pp. 856-869, July 1986.

[7] K.S. Kundert, J.K. White, and A. Sangiovanni-Vincentelli, *Steady-State Methods for Simulating Analog and Microwave Circuits*, Kluwer Academic Publishers, 1990.

[8] D. Hente and R.H. Jansen, "Frequency domain continuation method for the analysis and stability investigation of nonlinear microwave circuits," IEE Proceedings, part H, vol. 133, no.5, pp. 351-362, Oct. 1986.

[9] D. O. Pederson and K Mayaram, *Analog Integrated Circuits for Communication*, Kluwer Academic Publishers, 1991.

[10] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Baltimore: The Johns Hopkins University Press, 1996.