THE EXPLOITATION OF SPARSE-MATRIX TECHNIQUES IN CONJUNCTION WITH THE PIECEWISE HARMONIC-BALANCE METHOD FOR NONLINEAR MICROWAVE CIRCUIT ANALYSIS

Vittorio RIZZOLI
Dipartimento di Elettronica, Informatica e Sistemistica, University of Bologna ([1])
Franco MASTRI
Istituto di Elettrotecnica, University of Bologna ([2])
Fiorella SGALLARI
Dipartimento di Matematica, University of Bologna ([3])
Valeria FRONTINI
CINECA - Centro Interuniversitario di Calcolo dell' Italia Nordorientale ([4])

ABSTRACT

The paper describes a novel sparse-matrix approach to the simulation of nonlinear microwave circuits by the Newton-iteration-based piecewise harmonic-balance technique. Selected elements of the Jacobian matrix are set to zero according to some physical criterion, which leads to a sparse Jacobian with a preselected the pattern of nonzeros. This allows the use of specialized sparse-matrix solvers, with an effective optimization of both memory storage and CPU time. The code has been developed in both a scalar and a vectorized version optimized for the Cray Y-MP supercomputer.

INTRODUCTION

Within the frame of harmonic-balance (HB) methods, two basic approaches to nonlinear circuit analysis are commonly used in computer-aided microwave engineering practices. One of them, usually referred to as the "piecewise" HB technique {1}, is based on the decomposition of the given circuit into a linear and a nonlinear multiport subnetwork interconnected through a number of common ports (which will be conventionally named "device ports"). With this method the number of state variables (SV) is equal to the number of device ports, and an algebraic solving system is obtained by requiring that the HB errors at all device ports and all frequencies of the assumed (discrete) spectrum vanish. The other approach is based on nodal analysis {2}. In this case, the state variables coincide with the node voltages, and the circuit equations are obtained in a straightforward way from the application of Kirchhoff's current law to each circuit node. In both cases the problem unknowns are represented by the complex phasors of the SV harmonics, and the nonlinear system is solved by a Newton iteration.

The advantage of the piecewise method is that CPU time is virtually independent of the linear-subnetwork complexity. When a highly accurate modeling is required for the linear part of the circuit (i.e., all relevant parasitics and discontinuities are taken into account), the number of circuit nodes may far exceed the number of device ports, typically by a factor ranging from 5 to 10. This means that for a given circuit description, the nodal approach usually works with a much larger number of unknowns than the piecewise technique. Thus the use of sparse-matrix techniques is mandatory with this method, and is made possible by the the sparsity of the node-admittance matrix and of the Jacobian matrix generated thereof. On the other hand, the piecewise HB technique works with a much smaller, dense Jacobian, and can be successfully implemented by conventional factorization.

When the number of nonlinear devices in the circuit and/or the number of frequencies become large, simulation costs are dominated by the Jacobian factorization time, and increase very quickly. This has represented until now a limiting factor in view of the application of the piecewise HB method to very large problems (e.g., more than 20 devices and 25 frequencies). In this paper we introduce a new algorithm allowing the advantages of sparse-matrix solvers to be exploited in conjunction with this method. The analysis is based on a generalized parametric description of the nonlinear devices allowing the exact computation of the error gradients for any kind of multitone excitation. A predetermined spectrum is used for both the signal waveforms and the derivatives, so that the pattern of nonzero elements of the Jacobian is exactly known *a priori*. This allows specific, purposefully developed sparse-matrix solvers to be implemented for specific classes of circuit problems, rather than using a general-purpose solver for all cases. In this way the potential of sparse-matrix methods is exploited to the highest possible extent. The result is a considerable saving of both memory occupation and CPU time, and an excellent tradeoff between speed and dynamic range. For illustrative purposes the new technique is applied to a mixer problem typical of a state-of-the-art MMIC environment {3}.

COMPUTATION OF THE JACOBIAN

Following {4}, we describe the nonlinear subnetwork by a set of generalized parametric equations of the form

$$\mathbf{v}(t) = \mathbf{u}\left[\mathbf{x}(t)\ ,\ \frac{d\mathbf{x}}{dt}\ ,\ \ldots,\ \frac{d^n\mathbf{x}}{dt^n}\ ,\ \mathbf{x}_D(t)\right]$$
$$\mathbf{i}(t) = \mathbf{w}\left[\mathbf{x}(t)\ ,\ \frac{d\mathbf{x}}{dt}\ ,\ \ldots,\ \frac{d^n\mathbf{x}}{dt^n}\ ,\ \mathbf{x}_D(t)\right] \qquad (1)$$

In eqn. (1), $\mathbf{x}(t)$ is a vector of state variables, $\mathbf{v}(t)$, $\mathbf{i}(t)$ are vectors of voltages and currents at the device ports, and $\mathbf{x}_D(t)$ is a vector of time-delayed state variables, i.e., $x_{Di}(t)=x_i(t-\tau_i)$ where the τ_i's are time constants. The vector-valued functions $\mathbf{u}$, $\mathbf{w}$ are assumed to be nonlinear and memoryless. All vectors have the same size n_D, equal to the number of device ports.

Making use of the piecewise HB technique {1}, the nonlinear solving system may be written in the form $\mathbf{E}(\mathbf{X})=\mathbf{0}$, where $\mathbf{E}$, $\mathbf{X}$ are vectors of complex phasors of the HB errors at the device ports, and of the state-variable harmonics, respectively. In the general case of multitone excitation, there are F independent sinusoidal sources acting in the circuit, with angular frequencies $\omega_1, \omega_2, \ldots, \omega_F$. A generic frequency of the

([1]) Villa Griffone, 40044 Pontecchio Marconi, Bologna - ITALY
([2]) Via Risorgimento 2, 40136 Bologna - ITALY
([3]) Via Saragozza 8, 40123 Bologna - ITALY
([4]) Via Magnanelli 6/3, 40033 Casalecchio di Reno, Bologna - ITALY

CH2848-0/90/0000-1295$01.00 © 1990 IEEE

spectrum is thus given by $k_1\omega_1+k_2\omega_2+...+k_F\omega_F$ (where the k_i's are integer numbers) and can be identified by a vector $\mathbf{k}\equiv[k_1, k_2,..., k_F]^T$. This naturally leads to partitioning the vectors $\mathbf{E}, \mathbf{X}$ into subvectors $\mathbf{E_k}, \mathbf{X_k}$, where $\mathbf{E_k}$ contains the HB errors at the **k**-th frequency, and $\mathbf{X_k}$ contains the **k**-th harmonics of all state variables. The vectors **U**, **W** of the voltage and current harmonics at the device ports are partitioned in a similar way. According to the piecewise HB method we have

$$\mathbf{E_k} = \mathbf{Y}(\omega_k)\,\mathbf{U_k} + \mathbf{N_k} + \mathbf{W_k} \tag{2}$$

$\mathbf{Y}(\omega_k)$ being the linear-subnetwork admittance matrix computed at the **k**-th angular frequency, and $\mathbf{N_k}$ the vector of Norton equivalent current sources of the free generators of frequency ω_k.

In practice, in order to avoid the use of negative frequencies, the circuit equations are formulated in terms of real and imaginary parts of both the HB errors and the SV harmonics. Thus $\mathrm{Re}[\mathbf{E_k}]$ and $\mathrm{Im}[\mathbf{E_k}]$ (for all **k**'s) are stacked to form a real error vector, and so are $\mathrm{Re}[\mathbf{X_k}]$ and $\mathrm{Im}[\mathbf{X_k}]$ to form a vector of real unknowns. Accordingly, the Jacobian matrix used to solve the nonlinear system by the Newton-Raphson method is also partitioned into submatrices $\mathbf{J_{k,s}}$, containing the real and imaginary parts of the partial derivatives of the HB errors at the **k**-th frequency with respect to the real and imaginary parts of the **s**-th harmonics of all state variables. We have

$$\mathbf{J_{k,s}} = \begin{bmatrix} \dfrac{\partial \mathrm{Re}[\mathbf{E_k}]}{\partial \mathrm{Re}[\mathbf{X_s}]} & \dfrac{\partial \mathrm{Re}[\mathbf{E_k}]}{\partial \mathrm{Im}[\mathbf{X_s}]} \\ \dfrac{\partial \mathrm{Im}[\mathbf{E_k}]}{\partial \mathrm{Re}[\mathbf{X_s}]} & \dfrac{\partial \mathrm{Im}[\mathbf{E_k}]}{\partial \mathrm{Im}[\mathbf{X_s}]} \end{bmatrix} \tag{3}$$

Since the DC components of both the errors and the state variables are real, in (3) the second column is suppressed for **s=0**, the second row is suppressed for **k=0**, and only the first term is retained for **k=s=0**. Thus, if the total number of frequencies (not including DC) is denoted by n_H, the size of the Jacobian matrix is $N=n_D(2n_H+1)$. From (2) we obtain directly

$$\frac{\partial \mathrm{Re}[\mathbf{E_k}]}{\partial \mathrm{Re}[\mathbf{X_s}]} = \mathrm{Re}\left\{\mathbf{Y}(\omega_k)\,\frac{\partial \mathbf{U_k}}{\partial \mathrm{Re}[\mathbf{X_s}]}\right\} + \frac{\partial \mathrm{Re}[\mathbf{W_k}]}{\partial \mathrm{Re}[\mathbf{X_s}]} \tag{4}$$

and the like. The partial derivatives appearing on the right-hand side of (4) can be exactly computed in the following way {4}. For any given vector **X** (i.e., at any step of the Newton iteration), the derivatives of **u**, **w** with respect to their arguments are expressed by multiple Fourier expansions. For the voltages (e.g.) the **p**-th coefficients of such expansions are denoted by $\mathbf{C_{m,p}}$ (for the derivative of **u** wrt. $d^m\mathbf{x}/dt^m$), and by $\mathbf{C_p^D}$ (for the derivative wrt. $\mathbf{x_D}$). Then we have {4}

$$\frac{\partial \mathbf{U_k}}{\partial \mathrm{Re}[\mathbf{X_s}]} = \sum_{m=0}^{n} (j\omega_s)^m \left[\Gamma_{m,k-s} + (-1)^m \Gamma_{m,k+s}\right]$$

$$\frac{\partial \mathbf{U_k}}{\partial \mathrm{Im}[\mathbf{X_s}]} = \sum_{m=0}^{n} j(j\omega_s)^m \left[\Gamma_{m,k-s} - (-1)^m \Gamma_{m,k+s}\right] \tag{5}$$

where

$$\Gamma_{m,k\pm s} = \mathbf{C}_{m,k\pm s} \qquad (m > 0)$$

$$\Gamma_{0,k\pm s} = \mathbf{C}_{0,k\pm s} + \mathbf{C}^{D}_{k\pm s}\exp(\pm j\omega_s\tau) \tag{6}$$

and τ is the diagonal matrix of the time delays τ_i.

SPARSE-MATRIX TECHNIQUE

In any harmonic-balance simulation, the discrete spectrum **S** of all time-dependent physical quantities must be defined *a priori*. For a strictly periodic regime this is easy to accomplish, since all harmonics of a given fundamental up to a specified maximum order are usually of interest. On the other hand, in the case of multitone excitation the spectrum is considerably more complicated, and a direct selection of the desired lines at the data entry level might become a cumbersome job. For this reason, our simulator makes available to the user a number of predetermined spectral patterns, which may be selected by providing a very limited number of input parameters. As an example, for the case of two-tone excitation two spectra are available: a general two-tone intermodulation (IM) pattern, and a mixer-like pattern. The latter is useful when the two intermodulating tones are close in frequency and have very different power levels. For both spectra, the frequencies are given by $\omega_k=k_1\omega_1+k_2\omega_2$, with the following bounds:

1) for the general intermodulation analysis

$$0 \le |k_1| + |k_2| \le M \tag{7}$$

2) for the mixer case

$$0 \le |k_1 + k_2| \le N_O$$
$$0 \le |k_2| \le N_B \tag{8}$$

Note that (7) defines a conventional intermodulation spectrum with all IM products up to the M-th order taken into account. On the other hand, (8) defines a set of of N_B baseband (IF) lines plus N_O clusters of $2N_B+1$ lines each, centered around the harmonics of ω_1 (N_O being the number of significant LO harmonics). ω_1 plays the role of local-oscillator frequency.

The key idea of our sparse-matrix approach is to set to zero selected entries of the Jacobian matrix on the basis of some predetermined physical criterion. In the first application of this concept, we describe by predetermined spectra not only the waveforms to be balanced, but the derivatives (4), (5) as well. For the two-tone case, the spectrum of the derivatives is still defined by (7) or (8), but with independent bounds M_d, N_{Od}, N_{Bd}, which are empirically optimized for a given class of problems. Let the assumed spectrum of the derivatives for a given simulation be denoted by $\mathbf{S_d}$. Then the terms appearing on the right-hand side of (5) are retained whenever $\omega_{k\pm s}$ belongs to $\mathbf{S_d}$, and are set to zero otherwise. The only exception to this rule is that all couplings with the DC components are considered non-negligible (so that the first n_D rows and columns of the Jacobian are always dense), since this was found to enhance considerably the dynamic range of the sparse-matrix analysis. In this way the Jacobian matrix becomes sparse, with a pattern of nonzero submatrices of the form (3) which is uniquely determined (for a given **S**) by the

choice of S_d. On the other hand, the submatrix (3) usually contains a very limited number of structural zeros, and is thus treated as dense in the factorization.

Generally speaking, this technique is very powerful, for several reasons. Since the sparsity pattern is known *a priori*, one can avoid the use of general-purpose sparse-matrix solvers, and implement instead a family of specialized solvers, each individually optimized for a specific matrix structure, and making use of specific rules for addressing the nonzeros. This leads to an effective optimization of both memory storage and CPU time, and to the slowest possible increase of both with increasing problem size. Also, as shown below, in many practical cases the sparse Jacobians have very simple and repetitive structures. This makes for an accurate control of the generation of fill-ins, and allows the specialized solvers to be extremely efficient. Finally, it is worth noting that the pattern of nonzero submatrices in the Jacobian is *topology independent*, in the sense that it is only established by the choices of $\mathbf{S}$, $\mathbf{S}_d$. This means that a specific sparse-matrix solver can be applied to *all* the nonlinear circuits which can be analysed in terms of the associated spectra $\mathbf{S}$, $\mathbf{S}_d$, irrespective of their physical configuration.

As a representative example, we consider a mixer problem defined by (8), and make use of a very simplified spectrum $\mathbf{S}_d$ defined by $N_{Od}=1$, $N_{Bd}=0$. If the spectral lines are suitably ordered, and the LO harmonics are considered first, the Jacobian takes the doubly-bordered tridiagonal structure illustrated in fig. 1. In this figure, the crosses and dots are representative of nonzero and zero submatrices of the form (3), respectively, and the typical case $N_O=8$, $N_B=1$ has been considered for reference.

The structure shown in fig. 1 is very interesting, since a specialized routine for the factorization of this matrix can be easily written. Memory occupation is virtually limited to the storage of the nonzero entries of the Jacobian, and is thus reduced by a factor of 5.2 with respect to the full matrix storage for $n_D=16$, $n_H=25$. With the same dimensions, the speedup factor with respect to a standard dense-matrix solver is about 40 on a VAX 8800

AN EXAMPLE OF APPLICATION

The potential of the new technique for the simulation of large-size nonlinear circuits by the piecewise HB method is presently being investigated in depth. As a preliminary example of application, we report here on the analysis of a distributed DGFET mixer of the kind described in {3}, whose simplified topology is schematically shown in fig. 2. Taking discontinuities and parasitics into account, this circuit is simulated by 8 single-gate FET's and 79 circuit components, for a total of 16 device ports and 77 nodes. The spectrum is defined by (8) with $\omega_1/2\pi=3$ GHz, $\omega_2/2\pi=3.7$ GHz, and $N_O=8$, $N_B=1$, corresponding to 25 frequencies overall (plus DC). The analysis is carried out at an LO power level of +10 dBm (for which the small-signal conversion gain is about -1.3 dB), with swept RF power from -30 dBm up to +5 dBm (corresponding to a gain compression of about 2.2 dB) in 1 dB steps.

Making use of the HB simulator described in {4} with exact derivatives as previously discussed, the analysis takes an average of about 124 CPU seconds per point on a VAX 8800. The storage of the Jacobian alone requires approximately 5.3 MB of memory, and its factorization takes about 36 CPU seconds per point.

The same analysis is then repeated by the sparse-matrix technique with $N_{Od}=1$, $N_{Bd}=0$, i.e., with a Jacobian matrix having the structure shown in fig. 1. As in the previous case, no convergence problems are encountered in the power range considered, and the numerical results are identical to those obtained from the dense-matrix calculations. Thus even in this very simple implementation, the sparse-matrix approach can cover the entire range of normal mixer operation, and produce significant gain-compression information as well.

By the specialized solver the system-solving time is reduced by a factor of 40, and the average analysis cost drops to about 42 CPU seconds per point. The memory required to store the Jacobian and the related working areas is only 1 MB. Note that in this case the percentage of nonzero entries of the Jacobian matrix is approximately 14.8%, which is still far too large for an effective use of general-purpose sparse-matrix solvers, e.g., those available in the Harwell mathematical library {5}. Also, it is worth noting that the performance of the new algorithm becomes more and more impressive as the size of the numerical problem is increased. In particular, the memory and CPU time required by the sparse-matrix solver are *linear* functions of the number of spectral components to be balanced. This is shown in fig. 3 where the factorization time for the Jacobian of fig. 1 is plotted against the number of spectral lines (VAX 8800). Thus the new method is expected to provide dramatic performance improvements in those applications requiring a very large number of harmonics, such as intermodulation analysis in mixers{4}. This problem will be tackled in a future work.

The sparse-matrix analysis algorithm was also developed in a fully vectorized version, which was carefully optimized for exploiting the vector processing capabilities of a Cray Y-MP computer system. The performance of this code is definitely impressive: on a Y-MP/832 the swept-power mixer analysis described above takes an average of 0.60 CPU seconds per point, of which only 0.018 are due to system solving. This result is likely to have a considerable impact on microwave circuit CAD capabilities, since for the first time the direct numerical optimization of complex multioctave MMIC components such as distributed mixers might become feasible with acceptable costs and wall-clock times.

ACKNOWLEDGMENT

This work was partly sponsored by the Italian National Research Council (CNR).

REFERENCES

{1} M. S. Nakhla and J. Vlach, "A piecewise harmonic-balance technique for determination of periodic response of nonlinear systems", *IEEE Trans. Circuits Systems*, Vol. CAS-23, Feb. 1976, pp. 85-91.

{2} K. S. Kundert and A. Sangiovanni-Vincentelli, "Simulation of nonlinear circuits in the frequency domain", *IEEE Trans. Computer-Aided Design*, Vol. CAD-5, Oct. 1986, pp. 521-535.

{3} T. S. Howard and A. M. Pavio, "A distributed monolithic 2-18 GHz dual-gate MESFET mixer", *IEEE 1987 Microwave and mm-Wave Monolithic Circuits Symp. Dig.* (Las Vegas), June 1987, pp. 27-30.

{4} V. Rizzoli, C. Cecchetti, A. Lipparini, and F. Mastri, "General-purpose harmonic balance analysis of nonlinear microwave circuits under multitone excitation", *IEEE Trans Microwave Theory Tech.*, Vol. 36, Dec. 1988, pp. 1650-1660.

{5} I. S. Duff, "A survey of sparse-matrix research", *Proc. IEEE*, Vol. 65, Apr. 1977, pp. 500-535.

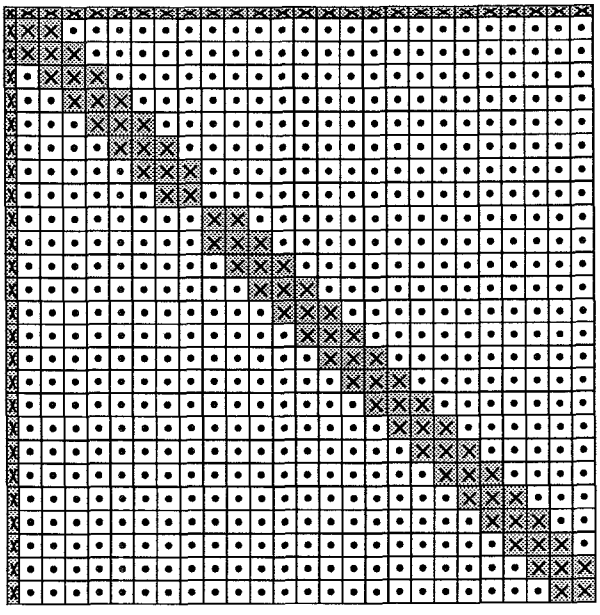

Fig. 1 - Pattern of zero and nonzero submatrices of the Jacobian matrix for $n_H = 25$, $N_{Od} = 1$, $N_{Bd} = 0$.

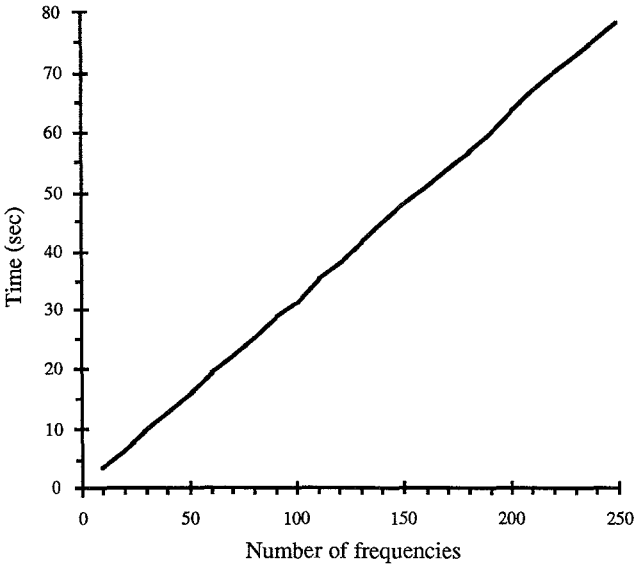

Fig. 3 - Factorization time of the Jacobian matrix shown in fig. 1 (VAX 8800).

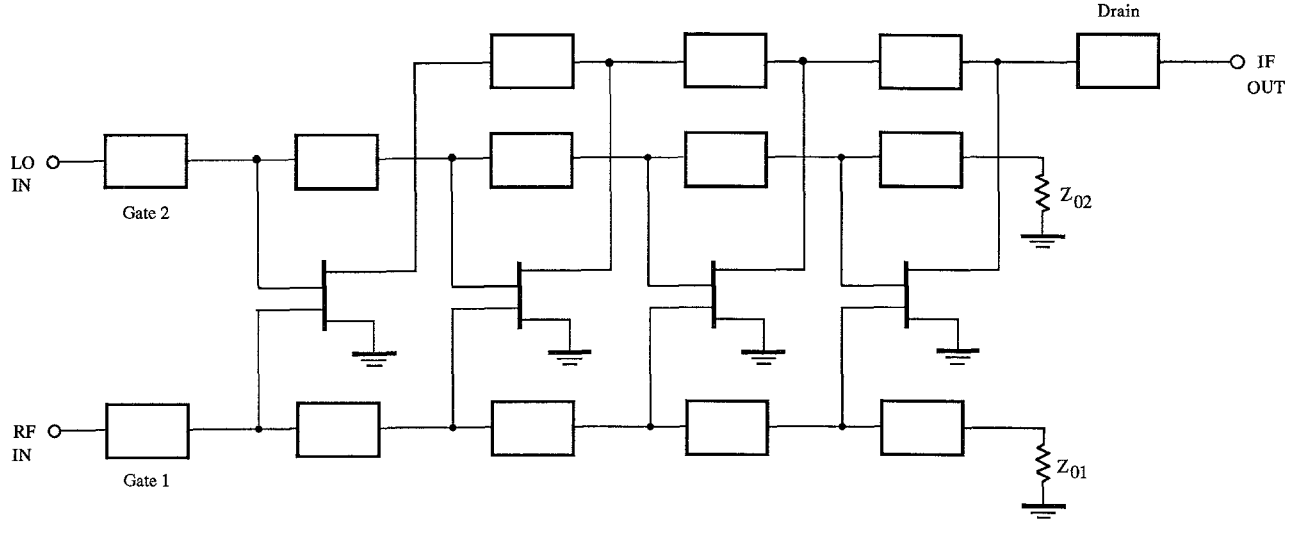

Fig. 2 - Schematic topology of a monolithic distributed mixer.