FASTNR: An Efficient Fault Simulator for Linear and Nonlinear DC Circuits

J. Soares Augusto and C. F. Beltrán Almeida
INESC/IST, Aptd. 13069, 1000 Lisboa, Portugal.
E-mail: jasa@dalton.inesc.pt and cfb@dalton.inesc.pt

Abstract

In this paper we describe the simulator FASTNR (*FAST Newton-Raphson*) where an efficient methodology for solving the faulty circuit equations, called *FAult RUBber Stamps* (FARUBS), is implemented. Its application to single fault simulation in linear and nonlinear circuits is reported. The efficient fault simulation in nonlinear DC circuits is due both to the development of original linearized Newton-Raphson models for electronic devices and to the simulation of fault values in a "continuation" stream. Fault simulation in linear cascades with up to 5000 nodes has shown an improvement of four orders of magnitude in simulation time, when compared to that of the nominal circuit. In nonlinear circuits, the time efficiency is sometimes better than two orders of magnitude.

Keywords: Efficient Analog Fault Simulation; Solution of Linear and Nonlinear Circuits; Analog Test and Diagnosis.

1. INTRODUCTION

Efficient fault simulation is an important issue when Simulation-Before-Test (SBT) dictionary-based techniques are used in fault diagnosis of electronic circuits. A survey of efficient fault simulation in analog circuits is given in [1, 2, 3]. In this paper, we describe some results of single fault simulation with the FASTNR simulator [4], built around the modified Newton-Raphson (NR) algorithm, that implements a methodology called *fault rubber stamps* for inserting faults in the circuit equations [3]. After a very concise (due to space limitations) survey of the literature on efficient fault simulation, we present the FARUBS methodology applied to linear circuits, and then we extend it to efficient fault simulation in nonlinear circuits. Several examples, including

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35498-9_57
L. M. Silveira et al. (eds.), *VLSI: Systems on a Chip*

large circuits, are given in order to validate the approach. It isn't made any comparison with previous results since the works on the subject previously reported are focused on the methodology instead of on simulation times.

The Householder or Sherman-Morrison formula [5, 6] has been used for simulating efficiently faults in linear circuits by Temes and other authors [1, 7]. This rank one modification formula states that if $\mathbf{M}^{-1}$ is known, then the inverse of $\mathbf{M}+s\mathbf{v}\mathbf{w}^T$, where $\mathbf{v}$ and $\mathbf{w}$ are vectors and s is a scalar, is $(\mathbf{M}+s\,\mathbf{v}\mathbf{w}^T)^{-1} = \mathbf{M}^{-1}(\mathbf{I}-\alpha^{-1}\mathbf{v}\mathbf{w}^T\mathbf{M}^{-1})$, with $\alpha^{-1} = (s^{-1}+\mathbf{w}^T\mathbf{M}^{-1}\mathbf{v})^{-1}$. Since $\mathbf{v}$ and $\mathbf{w}$ are sparse, the amount of operations involved in the application of this formula is rather small when compared with the full inversion of the modified matrix. There are variants of that formula that apply to the $\mathbf{L}$ and $\mathbf{U}$ factors of $\mathbf{M}$ instead of to the inverse (see [8]), and other methods for updating triangular factors are given in [9] and its references.

Efficient fault simulation in nonlinear DC circuits was reported by Lin [10] and by Prasad [11]. In the former, it is used the piecewise linear modeling of the nonlinear devices and then ideal diodes are the only nonlinear elements. These elements load the ports of a multiport comprising the linear elements, and the circuit equations are formulated as a linear complementarity problem. In the latter work, the overall circuit is modeled as a linear multiport terminated by nonlinear elements and the equations of the faulty circuit are written by applying the modification formula to a matrix depending only on linear elements. The solution of the reduced nonlinear circuit equations is not discussed, as well as the potential efficiency that can be achieved.

Recently, other approaches [3] have been tried in this area: the simulation of faults in parallel on several distributed processors (or computers); the hierarchization of circuits in several description levels and the use of simulation tools capable of simulating these levels; and the preliminary development of simulators in the time domain capable of simulating faults in parallel.

In the present work, faults are simulated in linear circuits by *increasing* the nominal triangular LU factors by one line and one column: the nominal factors remain unchanged. In nonlinear circuits this same strategy is applied jointly with specially developed Newton-Raphson companion models for the electronic devices [5], what allows for a special structure in the NR matrix where only a small matrix block is updated in each NR iteration.

2. FAULT RUBBER STAMPS IN LINEAR CIRCUITS

The modified nodal analysis (MNA) equations can be assembled with "rubber stamps" on an element by element basis, by inserting the respective contributions in the circuit matrix [5, 12]. We extended that concept to *rubber stamps in faulty elements* and called this fault insertion and simulation methodology *fault rubber stamps*. This methodology is based on *reusing* the $\mathbf{L}$ and $\mathbf{U}$ fac-

tors of the nominal circuit matrix, obtained with the Crout algorithm [5, 6], to simulate a fault. We described this methodology for linear circuits in [4] and here we extend it to nonlinear circuits.

Consider the MNA equations of a linear circuit

$$\mathbf{Mx} = \mathbf{b} \tag{1.1}$$

where $\mathbf{M}$ is the $n \times n$ circuit matrix, $\mathbf{x}$ is the vector of circuit variables and $\mathbf{b}$ is the vector of independent sources. In circuit simulators, 1.1 is usually solved through the LU factorization of $\mathbf{M}$, that is, by calculating a lower triangular matrix $\mathbf{L}$ and an upper triangular matrix $\mathbf{U}$ with 1's in the diagonal such that $\mathbf{M} = \mathbf{LU}$. The solution of (1.1) is obtained by solving for $\mathbf{z}$ the lower triangular system $\mathbf{Lz} = \mathbf{b}$ and by solving for $\mathbf{x}$ the upper triangular system $\mathbf{Ux} = \mathbf{z}$

The LU factorization with the Crout algorithm calculates the elements l_{ij} and u_{ij} from left to right and from up to down. This means that if an element in $\mathbf{M}$ changes — say m_{ab} —, the l_{ij} and u_{ij} values with $i < a$ or $j < b$ remain unchanged and don't need to be calculated when obtaining the new $\mathbf{L}$ and $\mathbf{U}$ factors of the modified $\mathbf{M}$ matrix. If the modification in the system, due to the existence of a fault, occurs in the lowest rightmost part of the matrix, then most of the nominal $\mathbf{L}$ and $\mathbf{U}$ factors can be reused.

The method developed in order to *push the modifications in the equations due to the fault to the lowest rightmost part of* $\mathbf{M}$, consists in joining to (1.1) one more variable ϕ and one more equation related to the fault, what leads to the following system of equations for the faulty circuit

$$\underbrace{\begin{bmatrix} \mathbf{M} & \mathbf{v}_c \\ \mathbf{v}_\ell^T & -\delta^{-1} \end{bmatrix}}_{\mathbf{M}_a} \begin{bmatrix} \mathbf{y} \\ \phi \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix} \tag{1.2}$$

where $\mathbf{v}_\ell$ and $\mathbf{v}_c$ are $n-$vectors filled with zeros, except in 2 positions at most, and $\mathbf{y}$ is the faulty circuit solution. The factors $\mathbf{L}_a$ and $\mathbf{U}_a$ of the matrix $\mathbf{M}_a$ lead to two triangular systems that are solved in sequence to simulate the fault

$$\underbrace{\begin{bmatrix} \mathbf{L} & \mathbf{0} \\ \mathbf{l}^T & \tau \end{bmatrix}}_{\mathbf{L}_a} \begin{bmatrix} \mathbf{z} \\ \phi \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix}, \qquad \underbrace{\begin{bmatrix} \mathbf{U} & \mathbf{u} \\ \mathbf{0}^T & 1 \end{bmatrix}}_{\mathbf{U}_a} \begin{bmatrix} \mathbf{y} \\ \phi \end{bmatrix} = \begin{bmatrix} \mathbf{z} \\ \phi \end{bmatrix} \tag{1.3}$$

and comparing these triangular factors with $\mathbf{M}_a$ we conclude that

$$\mathbf{M} = \mathbf{LU}, \quad \mathbf{v}_c = \mathbf{Lu}, \quad \mathbf{v}_\ell = \mathbf{U}^T\mathbf{l}, \quad -\delta^{-1} = \mathbf{l}^T\mathbf{u} + \tau. \tag{1.4}$$

$\mathbf{L}$ and $\mathbf{U}$ and are calculated only once when solving the nominal circuit. To calculate the vectors $\mathbf{l}$ and $\mathbf{u}$ it is necessary to solve two $n \times n$ triangular systems of linear equations and to calculate τ the dot product $\mathbf{l}^T\mathbf{u}$ must be carried out.

The intermediate vector $\mathbf{z}$ is the nominal one, already calculated. After doing some algebra with (1.4) and with the lower triangular system in (1.3) we also calculate

$$\phi = \frac{\delta \mathbf{l}^T \mathbf{z}}{1 + \delta \mathbf{l}^T \mathbf{u}} \tag{1.5}$$

and the faulty circuit solution $\mathbf{y}$ is obtained from the system

$$\mathbf{U}\mathbf{y} = \mathbf{z} - \phi\mathbf{u}. \tag{1.6}$$

To exemplify a *fault rubber stamp* we consider an admittance between nodes j and j' with nominal value Y. The MNA stamp of this element in the circuit matrix is [12, 5]

$$\begin{array}{c} \\ j \\ j' \end{array} \begin{array}{c} \begin{array}{cc} v_j & v_{j'} \end{array} \\ \left[\begin{array}{cc} +Y & -Y \\ -Y & +Y \end{array}\right] \end{array}$$

where the left labels j and j' indicate the correspondence with the Kirchhoff's Current Law (KCL) in those nodes.

Let's suppose Y is faulty and its value changes to $Y + \delta$. We introduce a *fault variable* ϕ, the current from node j to node j' in the faulty admittance δ in parallel with Y (see figure 1). The *fault rubber stamp* of Y *plus* δ is

$$\begin{array}{c} \\ j \\ j' \end{array} \begin{array}{c} \begin{array}{ccc} v_j & v_{j'} & \phi \end{array} \\ \left[\begin{array}{cc|c} +Y & -Y & +1 \\ -Y & +Y & -1 \\ \hline +1 & -1 & -\delta^{-1} \end{array}\right] \end{array} \tag{1.7}$$

where the equation $v_j - v_{j'} - \phi/\delta = 0$ of the faulty admittance, called the *fault element equation*, was joined to the nominal rubber stamp increasing by 1 the dimension of the system. We remark that *the fault element equation (variable) is always located in the last row (column) of the faulty circuit matrix* and that the lowest rightmost element is always $-\delta^{-1}$.

We summarize the fault rubber stamps of admittances, impedances and VCCSs. Other stamps of linear elements are given in [3]. The fault variable is ϕ and δ is the *fault value*, that is, the deviation from nominal of the parameter that characterizes the element.

In figure 1 a) is shown a faulty admittance whose stamp was given in (1.7). In figure 1 b) and c) are shown a faulty impedance $Z \quad + \delta$ and a faulty

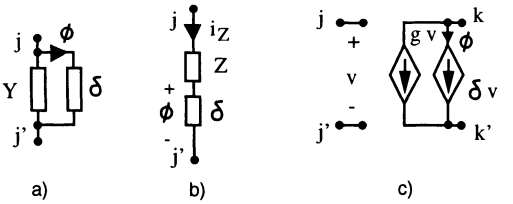

Figure 1 Faulty a) admittance (Y), b) impedance (Z), c) VCCS (g).

transconductance $g + \delta$ whose stamps are, respectively,

$$\begin{array}{c} \\ j \\ j' \\ Z\text{ eq.} \end{array} \begin{array}{c} \begin{array}{cccc} v_j & v_{j'} & i_Z & \phi \end{array} \\ \left[\begin{array}{cc|c|c} & & +1 & \\ & & -1 & \\ \hline +1 & -1 & -Z & -1 \\ \hline & & +1 & -\delta^{-1} \end{array}\right] \end{array} \qquad \begin{array}{c} \\ k \\ k' \\ f \end{array} \begin{array}{c} \begin{array}{ccc} v_j & v_{j'} & \phi \end{array} \\ \left[\begin{array}{cc|c} +g & -g & +1 \\ -g & +g & -1 \\ \hline +1 & -1 & -\delta^{-1} \end{array}\right] \end{array} .$$

Suppose we are dealing with full matrices and vectors. The computational cost of LU factorization is $\sim n^3$ while the computational cost of solving a triangular system is only $\sim n^2$. Thus, *we can simulate a fault with a complexity of* $\sim n^2$ *after the nominal circuit has been solved.* When, as is usually the case in circuit simulation, the matrices and vectors are sparse, the real asymptotes are lower than those above, but there are still substantial savings in fault simulation with FARUBS. The most dramatic efficiency is achieved *when simulating several deviations* δ *for the same fault parameter*: in this case we only need to calculate the new τ, ϕ, and then solve the upper triangular system in (1.3).

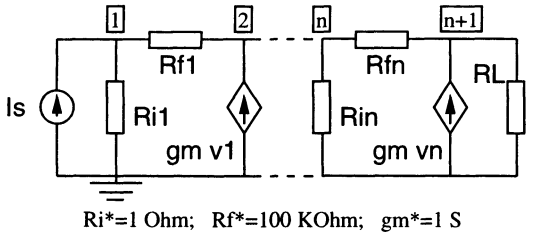

Figure 2 Cascade of n amplifiers.

The performance of the simulator was evaluated in the cascade of n amplifiers shown in figure 2 for values of n between 200 and 5000. The CPU simulation times are collected in figure 3.

This figure displays the experimental evidence of the efficiency of fault simulation in the above cascade of amplifiers. We compare the nominal circuit simulation time (□), the factorization time of the last row and column for each faulty element (+), and the estimated time of fault simulation for each fault value (i. e., the computation of τ and solution of the upper triangular system (1.3) with backward substitutions) (◇). This one *grows linearly with n,* what

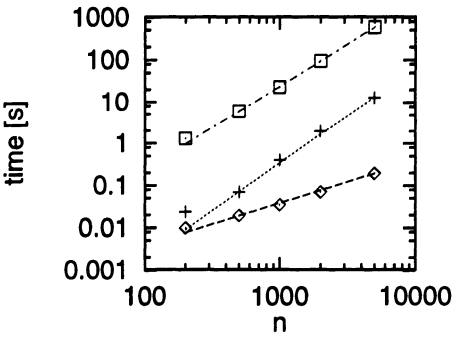

Figure 3 CPU simulation times of faults in the cascade of n amplifiers.

means that the simulation of several values of a given faulty parameter is very cheap, even in large circuits.

The simulation time of the nominal cascade (□) and the factorization time of the last row and column for each faulty element (+) follow approximately an n^2 law. Although both present the same polynomial complexity, *simulating one faulty element is about two orders of magnitude faster than simulating the nominal circuit*, as is seen by comparing the (□) with the (+) points in figure 3. We must say, however, that the nominal matrix is assembled and factorized with dynamic data structures that allow the exploitation of sparsity, while the fault line and column are written in memory and factorized as full vectors what doesn't allow us to take advantage of their sparsity.

3. SIMULATION OF FAULTS IN NONLINEAR CIRCUITS

Consider the simulation of a nonlinear circuit with the Newton-Raphson (NR) algorithm [5]. The electrical characteristics of nonlinear elements are linearized and in each iteration ν it must be solved a linear system of equations (departing from an initial guess $\mathbf{x}^0$)

$$\mathbf{M}^\nu \mathbf{x}^{\nu+1} = \mathbf{b}^\nu \qquad \nu = 0, 1, \ldots \tag{1.8}$$

To calculate $\mathbf{x}^{\nu+1}$ with LU factorization, the factors $\mathbf{L}^\nu$ and $\mathbf{U}^\nu$ such that $\mathbf{M}^\nu = \mathbf{L}^\nu \mathbf{U}^\nu$ must be obtained and two triangular systems must be solved. Since $\mathbf{M}^\nu \equiv \mathbf{M}(\mathbf{x}^\nu)$ changes from iteration to iteration, the triangular factors must be re-calculated in each iteration, and there is no advantage in simulating a fault $p + \delta$ by solving the augmented system

$$\begin{bmatrix} \mathbf{M}^\nu & \mathbf{v}_c \\ \mathbf{v}_\ell^T & -\delta^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{y}^{\nu+1} \\ \phi^{\nu+1} \end{bmatrix} = \begin{bmatrix} \mathbf{b}^\nu \\ 0 \end{bmatrix}. \tag{1.9}$$

This picture changes if we can find a way of bounding the iteration dependent part of $\mathbf{M}^\nu$ to a lower-right block. To simulate efficiently faults in nonlinear circuits we developed new Newton-Raphson models for the electronic devices. We expose the motivation behind the approach using the diode as an example.

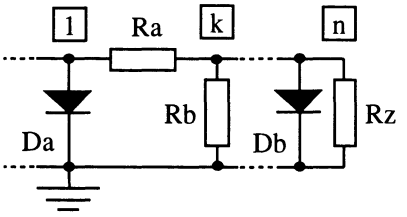

Figure 4 Nonlinear circuit with two nonlinear devices (diodes) between nodes 1 and n and ground.

The circuit in figure 4 has *only two diodes*, D_a between node 1 and ground and D_b between node n and ground. Their $v - i$ characteristic is $i = f(v)$, whose derivative is $g(v) = \frac{df(v)}{dv}$. In each NR iteration the diode current is approximated by a first order Taylor series $i^{\nu+1} \approx i^\nu + g(v^\nu)(v^{\nu+1} - v^\nu)$. Thus, we have the following NR system of equations for this circuit in the presence of a fault, if no reordering of the matrix is performed

$$\left[\begin{array}{ccc|c} g_a^\nu & \cdots & \cdots & \\ \vdots & & & \mathbf{v}_c \\ \vdots & & g_b^\nu & \\ \hline & \mathbf{v}_\ell^T & & -\delta^{-1} \end{array}\right] \left[\begin{array}{c} v_1^{\nu+1} \\ \vdots \\ v_n^{\nu+1} \\ \hline \phi^{\nu+1} \end{array}\right] = \left[\begin{array}{c} b_1^\nu \\ \vdots \\ b_n^\nu \\ \hline 0 \end{array}\right]$$

where $g_a^\nu = g(v_1^\nu)$, $b_1^\nu = g_a^\nu v_1^\nu - i_a^\nu$, $g_b^\nu = g(v_n^\nu)$ and $b_1^\nu = g_b^\nu v_n^\nu - i_b^\nu$. Since both g_a^ν changes in each iteration, all the matrix must be re-factorized and the advantage of the FARUBS methodology is lost.

If, however, D_a didn't exist in the circuit, the only matrix element changing in each NR iteration would be g_b^ν. In this case, the change in $\mathbf{L}^\nu$ and $\mathbf{U}^\nu$ from iteration to iteration would be located on their lower-right 2×2 blocks.

The obvious conclusion is: if the NR iteration-dependent elements of the matrix are pushed to the lowest rightmost block, only this block is re-factorized in each iteration. This allows for achieving some efficiency in simulating faults with the FARUBS methodology, even in nonlinear circuits.

Our approach to implementing the above reasoning consists in *modeling the diode as a current-controlled element* and introducing its current i as a new circuit variable. The diode (for instance D_a) is described by $v = h(i)$, where $h() \equiv f^{-1}()$ is the inverse function of $f()$, with derivative $r(i) = \frac{dh(i)}{di}$. The Taylor series approximation is now $v_1^{\nu+1} \approx v_1^\nu + r(i_a^\nu)(i_a^{\nu+1} - i_a^\nu)$ and we have

the NR iteration

$$\left[\begin{array}{ccc|c} & & & 1 & \\ & & & \vdots & \mathbf{v}_c \\ & & & \vdots & \\ 1 & \cdots & \cdots & -r_a^\nu & 0 \\ \hline & \mathbf{v}_\ell^T & & 0 & -\delta^{-1} \end{array}\right] \left[\begin{array}{c} v_1^{\nu+1} \\ \vdots \\ \vdots \\ i_a^{\nu+1} \\ \hline \phi^{\nu+1} \end{array}\right] = \left[\begin{array}{c} \vdots \\ \vdots \\ \vdots \\ e_a^\nu \\ \hline 0 \end{array}\right]$$

where $r_a^\nu = r(i_a^\nu)$ and $e_a^\nu = v_1^\nu - r_a^\nu i_a^\nu$. It is obvious that in the matrix only r_a^ν changes in each iteration and the only part of the **L** and **U** factors that must be updated corresponds to the lower right 2×2 block of **M**.

Generalizing the approach to any nonlinear circuit, the NR system will be written as

$$\left[\begin{array}{cc|c} \mathbf{M}_{11} & \mathbf{M}_{12} & \mathbf{v}_c \\ \mathbf{M}_{21} & \mathbf{M}_{22}^\nu & \mathbf{0} \\ \hline \mathbf{v}_\ell^T & \mathbf{0}^T & -\delta^{-1} \end{array}\right] \left[\begin{array}{c} \mathbf{y}_1^{\nu+1} \\ \mathbf{y}_2^{\nu+1} \\ \hline \phi^{\nu+1} \end{array}\right] = \left[\begin{array}{c} \mathbf{b}_1 \\ \mathbf{e}_2^\nu \\ \hline 0 \end{array}\right]$$

where the sub-matrix $\mathbf{M}_{22}^\nu$ and the RHS vector $\mathbf{e}_2^\nu$ are the only that change in each iteration, and the **L** and **U** factors of the above matrix will be

$$\left[\begin{array}{cc|c} \mathbf{L}_{11} & \mathbf{0} & \mathbf{0} \\ \mathbf{L}_{21} & \mathbf{L}_{22}^\nu & \mathbf{0} \\ \hline \mathbf{l}^T & \mathbf{p}^{T\nu} & \tau^\nu \end{array}\right] \qquad \left[\begin{array}{cc|c} \mathbf{U}_{11} & \mathbf{U}_{12} & \mathbf{u} \\ \mathbf{0} & \mathbf{U}_{22}^\nu & \mathbf{q}^\nu \\ \hline \mathbf{0}^T & \mathbf{0}^T & 1 \end{array}\right]$$

where is clear that only the lower-right sub-matrix of both triangular factors changes in each iteration.

The smaller the number of nonlinear devices in the circuit, the smaller the dimension of the iteration-dependent blocks of **L** and **U**: thus, *this approach is extremely suitable for circuits with only a few nonlinear devices.*

The aforementioned approach was applied also to *bipolar transistors* described by the Ebers-Moll model, to *MOSFETs* described by the quadratic model and to *OPAMPs* described with a "tanh" equation that models saturation at the output. The models are less precise than those in standard simulation tools, but *its purpose is fault simulation not detailed design verification.* Precise models would increase the dimension of the iteration-dependent block. Those models are detailed in [3].

SIMULATION WITH AN ORDERED FAULT LIST

A further efficiency improvement in FASTNR is achieved by simulating the fault values of each faulty element within a "continuation" philosophy. Suppose

the fault values in a parameter p to be simulated are ordered by increasing values as $p_{f1}, p_{f2}, p_{f3}, \ldots$ The NR system of equations when simulating the first value p_{f1} can be written as:

$$\mathbf{M}(\mathbf{x}^{\nu}, p_{f1})\,\mathbf{x}^{\nu+1} = \mathbf{b}(\mathbf{x}^{\nu}) \tag{1.10}$$

and its solution $\mathbf{x}_{f1}$ strictly satisfies the equality

$$\mathbf{M}(\mathbf{x}_{f1}, p_{f1})\,\mathbf{x}_{f1} \equiv \mathbf{b}(\mathbf{x}_{f1}) \tag{1.11}$$

(that is, $\mathbf{x}^{\nu+1} = \mathbf{x}^{\nu} + \epsilon$ after convergence is achieved, with $\epsilon \approx 0$.) To simulate the next fault value p_{f2}, which is just after p_{f1} in the ordered list, we use for the initial guess $\mathbf{x}^0$ the solution just obtained: we make $\mathbf{x}^0 \leftarrow \mathbf{x}_{f1}$ and begin the NR iteration for fault p_{f2} with the solution of

$$\mathbf{M}(\mathbf{x}_{f1}, p_{f2})\,\mathbf{x}^1 = \mathbf{b}(\mathbf{x}_{f1}) \tag{1.12}$$

after the actualization and factorization of the matrix and RHS vector. The limit of this iteration will be the solution $\mathbf{x}_{f2}$ of the circuit with the parameter p presenting a fault p_{f2}. The reason behind the success of this continuation procedure is that, often, the solution $\mathbf{x}_{f1}$ of the previous fault value lies inside the domain of quadratic convergence to the solution $\mathbf{x}_{f2}$ of the the NR algorithm and, thus, is a good initial guess. When this happens, usually not more than 3 or 4 iterations are needed to get a solution. This depends, of course, on the "distance" between $\mathbf{x}_{f1}$ and $\mathbf{x}_{f2}$: when, for example, one or more devices change their operating region with the new fault value (e.g. a bipolar transistor goes from the active region into saturation) the convergence is not so fast.

When switching from one faulty element to another, the solution of the nominal circuit is reloaded since the first p_f simulated fault value is small (remember that $p_f \equiv \delta$ is the *deviation from the nominal value*) and thus the nominal solution is a good guess for starting the NR iterations.

EXAMPLES WITH NONLINEAR CIRCUITS

We present some results from fault simulations in nonlinear circuits. The video amplifier in figure 5 a) has tw o bipolar transistors and is an example where most of the elements are linear. The dimension of the circuit matrix is $n = 23$, including the extra line and column to handle faults. The sub-matrix $\mathbf{M}_{22}^{\nu}$ is only 4×4, which means that a 5×5 matrix must be updated and factorized in each NR iteration when simulating faults. 40 and 80 fault values were simulated for each of the 9 resistances in the amplifier and the results are tabulated in figure 7. The speed-up achieved is around 18 when considering total time; however, *when estimating the time of simulation of a single fault value, the speed-up is 192.*

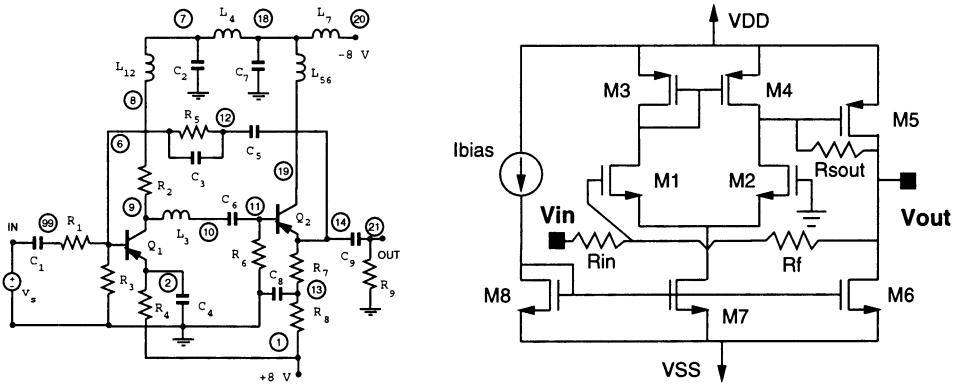

Figure 5 a) Video amplifier. b)Two-stage CMOS OPAMP.

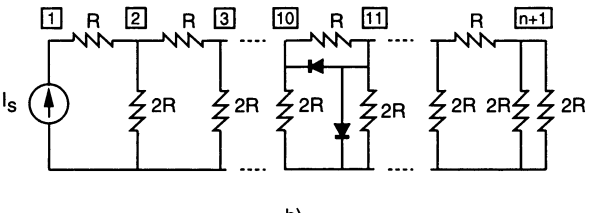

Figure 6 a) Nonlinear ladder with two diodes periodically inserted in the stages.

The circuit in figure 6 is a n-stage nonlinear RR 2 ladder with $2n + 1$ resistors, where $R = 1\ \Omega$, and has, periodically, two diodes in parallel with the resistors in the stage. We simulated 500-stage, 1000-stage and 2000-stage ladders. The *speed-up achieved in the larger ladder is 44.* The simulation results are summarized in the table in figure 7.

We finish this section with the presentation of results from a circuit consisting almost entirely of nonlinear devices, the CMOS two-stage OPAMP in figure 5 b), mounted as an inverting amplifier with voltage gain $G_v = -10$. The fault simulated in this circuit is quite common in CMOS processes: it is a Gate-Oxide-Short (GOS) fault in transistor M5. The simulation command in the circuit file ordered the simulation of 100 values of RSOUT below the nominal ($1\ T\Omega$). This resistance was introduced for fault simulation purposes only. The fault simulation results in this example are also tabulated in figure 7.

The dimension of the NR equations was 29 (including the fault line and column) and the dimension of the iteration dependent block was 16 × 16. Despite being an almost nonlinear circuit, a speed-up of 31 was observed when simulating each value of RSOUT.

It must be stressed that the time overhead resulting from reading the circuit file and assembling the internal data structures only penalizes the simulation of the nominal circuit.

CIRCUIT	*fel*	#f	*Dim*	%fi	its.	spu.	nom./fa.
CMOS	-	-	28	42	40	-	**31**
OPAMP	3	40	29	46	431	**14.7**	-
	3	200	"	"	1266	**25.3**	-
video	-	-	22	25	9	-	**192**
amp.	9	40	23	32	820	**17.5**	-
	9	80	"	"	1591	**18.3**	-
nonl. lad.	-	-	1201	0.8	10	-	**49**
(1000/10)	8	20	1202	1.0	488	**42**	-
	8	40	"	"	1178	**45**	-
nonl. lad.	-	-	2801	0.8	7	-	**44**
(2000/5)	2	10	2802	0.8	101	**9**	-
	8	10	"	"	408	**14**	-

Figure 7 Results of fault simulation in nonlinear circuits. LEGEND: fel is the number of faulty elements, $\#f$ is the number of fault values for each element, *Dim* is the dimension of the system, $\%fi$ is the percentage of fills in the LU factors, *its.* is the total number of NR iterations, *spu.* is the overall speed-up and $nom./fa.$ is the ratio between the simulation times of the nominal circuit and of each *fault value*. The number of faults is $fel \times \#f$.

4. CONCLUSIONS

In this paper we described the FARUBS methodology dedicated to the efficient simulation of faults in linear and in nonlinear circuits, and we also described the FASTNR simulator where it was implemented.

The strategies developed in order to achieve the above goal, consisted in reusing the nominal circuit equations and solution, and in simulating several fault values in the same circuit element in an ordered sequence. This allows for good NR starting points in the NR iteration when simulating those faults.

The observed efficiency reached four orders of magnitude in large linear cascades with 5000 nodes. In nonlinear circuits it was between one and two orders of magnitude. Even in an almost completely nonlinear circuit, a CMOS OPAMP, it was observed a speed-up of 31 when simulating a GOS fault.

It was shown that FARUBS is well suited for efficient fault simulation in linear DC circuits and in nonlinear DC circuits with a small number of nonlinear elements. The simulation of faults in nonlinear devices, as well as fault simulation in the AC domain, are still not implemented.

5. ACKNOWLEDGMENTS

This work has been partially supported by the Portuguese Government funded PRAXIS XXI 2/2.1/TIT/1661/95 project entitled *MIXCHIP - Advanced Techniques for the Design, Simulation and Test of Mixed Analogue-Digital Circuits.*

6. REFERENCES

[1] J. W. Bandler and A. E. Salama, "Fault diagnosis of analog circuits," *Proceedings of the IEEE, vol. 73, 8, Aug.*, 1985.

[2] J. L. Huertas, *Test and Design for Testability of Analog and Mixed-Signal Integrated Circuits: Theoretical Basis and Pragmatical Approaches*, in H. Dedieu (Ed.), *Selected Topics in Circuits and Systems,* Elsevier, EC-CTD'93, Davos, Switzerland, 1993.

[3] J. S. Augusto, *Fault Simulation and Diagnosis in Analog Circuits*, Ph. D. Thesis (in Portuguese), Instituto Superior Técnico, Lisbon, 1999.

[4] J. S. Augusto and C. B. Almeida, Efficient Fault Simulation, Diagnosis and Testability Evaluation in Linear and Nonlinear Circuits with Fault Rubber Stamps. *ETW97 Proc., Cagliari, Italy*, 1997.

[5] J. Vlach and K. Singhal, *Computer Methods for Circuit Analysis and Design.* Van Nostrand Reinhold, second edition, 1994.

[6] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in C.* Cambridge University Press, 1992.

[7] A. Pahwa and R. Rohrer, Band Faults: Efficient Approximations to Fault Bands for the Simulation Before Fault Diagnosis of Linear Circuits. *IEEE Tr. CAS-29, no. 2, pp. 81-8, Feb.* 1982.

[8] I. S. Duff, A. M. Erisman, and J. K. Reid, *Direct Methods for Sparse Matrices.* Oxford, 1986.

[9] I. N. Hajj, Updating method for LU factorization. *Electronic Letters, vol. 8, no. 7*, 1972.

[10] Lin P-M. and Elcherif Y., "Analogue Circuits Fault Dictionary - New Approaches and Implementation," *Cir. Th. and Appl., 12, pp. 149-172,* 1985.

[11] V. Prasad, Equations of Nonlinear Analogue Circuit for Fault Dictionary Generation. *Electronics Letters, vol. 26, no. 1, pp. 24-5, Jan.*, 1990.

[12] P. A. Brennan, C.-W. Ho, A. E. Ruehli and P. A. Brennan, "The modified nodal approach to network analysis," *IEEE Transactions on Circuits and Systems, CAS-22, pp. 504-509*, 1975.