Abstract

CHRISTOFFERSEN, CARLOS ENRIQUE. State Variable Harmonic Balance Simulation of a Quasi-Optical Power Combining System. (Under the direction of Michael B. Steer.)

A state variable implementation of harmonic balance circuit analysis is developed. The equations are formulated with the minimum number of unknowns starting from the modified nodal admittance matrix of the linear part of the circuit. The program uses advanced numerical techniques, such as automatic differentiation for calculating the Jacobian of the system. One key feature of the program is that it is easy to add new types of nonlinear elements and several numeric methods can be used to solve the harmonic balance equations. The program is tested by simulating a quasi-optical grid and comparing the results with measurements. Also, different numerical methods to solve the nonlinear equations are compared, and conclusions are made.

STATE VARIABLE HARMONIC BALANCE SIMULATION OF A QUASI-OPTICAL POWER COMBINING SYSTEM

by

CARLOS ENRIQUE CHRISTOFFERSEN

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

ELECTRICAL ENGINEERING

Raleigh

1998

APPROVED BY:

Chair of Advisory Committee

Biographical Summary

Carlos E. Christoffersen was born in Santa Fe, Argentina on April 17, 1968. He received the Electronic Engineer degree at the National University of Rosario, Rosario, Argentina in march 1993. From 1991 to June 1996 he was member of the research staff of the Laboratory of Microelectronics of the National University of Rosario, Argentina. During this time he developed several CAD tools, including a hierarchical circuit extractor program, that converts IC descriptions from CIF to SPICE. From 1993 to 1995 he was research fellow of the National Research Council of Argentina (CONICET).

In May 1996 he was awarded with a Fulbright scholarship to study in the United States. He was admitted into the Masters Program at North Carolina State University in Fall 1996. While working toward his M. S. degree he held a Research Assistantship with the Electronics Research Laboratory in the Department of Electrical and Computer Engineering. His research interest include computer aided simulation of circuits, analog, RF and integrated circuit design. He is a member of the Institute of Electrical and Electronic Engineers.

Acknowledgments

I would like to express my gratitude to my advisor Dr. Michael Steer for his support and guidance during my graduate studies. It was a privilege to be part of his quasi-optical research group.

I would also like to express my sincere appreciation to Dr. Frank Kauffman, and Dr. Griff Bilbro for showing an interest in my research and serving on my M.S. committee.

A very big thanks go to my past and present graduate student colleagues. First to Mr. Mark Summers for helping me understanding Transim. Also to To Mr. Huan-sheng Hwang, who helped me a lot of times. To Dr. Todd Nuteson, Mr. Chris Hicks, Ms. Jaee Patwardhan, Mr. Mete Ozkar and Mr. Ahmed Khalil. And to everyone in the group.

And finally, I wish to thank my wife, my son and my parents for their support and encouragement.

Contents

List of Figures

List of Tables

List of Symbols

$\Re()$ – Real part.
$v(t)$ – Voltage in the time domain.
V_m – m^{th} harmonic component of voltage.
ω – Angular frequency.
t – Time.
ϕ – Phase angle.
$F()$ – Error function.
I – Current vector in the frequency domain.
V – Voltage vector in the frequency domain.
$Q()$ – Charge vector in the frequency domain.
$\mathbf{\Omega}$ – Frequency domain differentiation matrix.
$\mathbf{Y}$ – Nodal admittance matrix.
I_S – Source vector.
$\mathcal{F}$ – Fourier transform operator.
$\mathcal{F}^{-1}$ – Inverse Fourier transform operator.
$\mathbf{q}$ – Charge vector in the time domain.
$\mathbf{i}$ – Current vector in the time domain.
$\mathbf{v}$ – Voltage vector in the time domain.
N – Vector of Norton equivalent current sources.
$\mathbf{J}$ – Jacobian matrix.
Γ – Matrix multiplier representing the DFT.
f – Objective function.
$O(f)$ – "Big o" of f.
$\mathbf{X}$ – State variable vector.
$\mathbf{M}_{fi}$ – Modified nodal admittance matrix at frequency fi.
S_{fi} – MNAM source vector at frequency fi.
$\mathbf{W}$ – Matrix that rearranges the nonlinear currents.
$\mathbf{T}$ – Transformation matrix.

$S_{sv,fi}$ – State variable source vector.
$\mathbf{M}_{sv,fi}$ – State variable impedance matrix.
HB – Harmonic Balance.
QO – Quasi-optical.
FFT – Fast Fourier transform.
DFT – Direct Fourier transform.
KCL – Kirchoff current law.
CAD – Computer aided design.
APFT – Almost periodic Fourier transform.

Chapter 1

Introduction

1.1 Motivations and Objectives of This Study

Steady state (harmonic balance) analysis of microwave circuits continues to be the dominant approach to the simulation of nonlinear RF and microwave circuits. Development efforts are currently being directed at extending the approach to accommodate a very large number of tones, improve robustness, and development of matrix-free methods to handle very large problems without explicit storage of the Jacobian.

One issue that is of particular concern is the simulation of quasi-optical (QO) power combiners with hundreds of active devices with significant thermal effects and distributed over an electrically large region. This exacerbates the problem of robustness and of model development. Consider the grid of Figure 1.1. Let's assume that there are two 3-terminal active devices per unit cell, and that we are interested in the response of the circuit up to the sixth harmonic. Then, the harmonic balance simulation of the resulting circuit using a conventional approach would use $100 \times 3 \times (1 + 2 \times 6) = 3900$ variables. On the other hand, a state variable approach would need only two

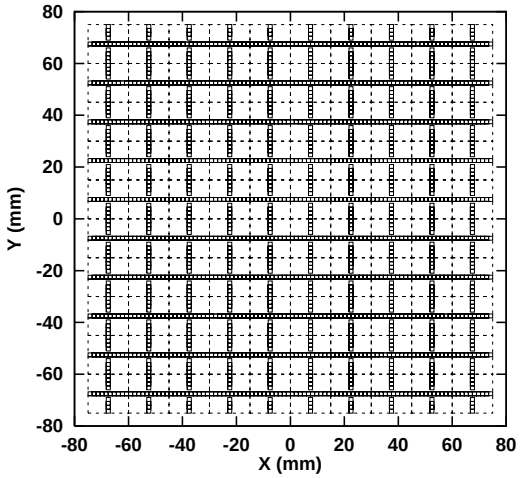

Figure 1.1: A quasi-optical grid.

state variables per element, therefore the number of unknowns is reduced by one third. Besides that, the stability of the nonlinear system can be greatly improved with a good election of the state variables.

In this work we made necessary developments which makes the steady-state simulation of large systems possible. The issue of robustness is addressed by

1. Using state variables which enables the use of parametrized device models including nonlinear behavioral models.

2. Using the minimum number of state variables and error functions and so eliminating the redundancy in the Jacobian.

3. A formulation permitting the use of various nonlinear equation solvers.

4. Handling of local reference nodes.

Another motivation for this work was to allow rapid nonlinear element

model development by eliminating the need for derivative coding. That is achieved using a novel implementation of automatic differentiation technology.

With respect to the memory requirements and running time, our program performs on average as well as commercial tools for small and medium size problems.

1.2 Thesis Overview

There is a large volume of research that has been done in the area of nonlinear circuit analysis. Chapter 2 presents a review of the previously published material on harmonic balance, with emphasis in the latest developments.

Chapter 3 presents the harmonic balance formulation used in Transim first, and then implementation of the algorithm along with an example.

In chapter 4 we see applications of the program in the calculation of QO grids.

In the last chapter we present the conclusions and the direction for future study.

Chapter 2

Literature Review

2.1 Introduction

Harmonic balance differs [17] from traditional transient analysis in two fundamental ways. These differences allow harmonic balance to compute periodic and quasi-periodic solutions directly and in certain circumstances give the method significant advantages in terms of accuracy and efficiency. Transient analysis, which uses standard numeric integration, constructs a solution as a collection of time samples with an implied interpolating polynomial. Typically the interpolating function is a low order polynomial. However, polynomials fit sinusoids poorly, and so many points are needed to approximate sinusoidal solutions accurately.

The first difference between harmonic balance and transient analysis is that harmonic balance uses a linear combination of sinusoids to build the solution. Thus, it approximates naturally the periodic and quasi-periodic signals found in a steady-state response.

Harmonic balance also differs from traditional time domain methods in that time domain simulators represent waveforms as a collection of samples

whereas harmonic balance represents them using the coefficients of the sinusoids.

There is no known way to compute the coefficients of the response directly from the coefficients of the stimulus for an arbitrary nonlinearity, though it is possible if the nonlinearity is described by a polynomial or a power series. It is not neccesary to consider only these special cases, nor to accept the error of using them to approximate arbitrary nonlinearities. Instead, we convert the coefficient representation of the stimulus into a sampled data representation; this is a conversion from the frequency domain to the time domain and is accomplished with the inverse Fourier transform. With this representation the nonlinear devices are easily evaluated. The results are converted back into coefficient form using the forward Fourier transform.

Harmonic balance was given its name because it was viewed as a method for balancing of currents between the linear and nonlinear subcircuits. Furthermore, harmonic balance is usually considered a mixed-domain method, because the nonlinear devices are evaluated in the time domain while the linear devices are evaluated frequency domain. However, evaluating the nonlinear devices in the time domain is not a fundamental part of the algorithm, but rather a convenience that does not affect the essential character of the algorithm. It is the formulation of the circuit equations in the frequency domain that give harmonic balance its essential characteristics. Thus, harmonic balance can be summarized as just being the method where KCL is formulated in the frequency domain.

Harmonic balance methods [8] are variations of Galerkin's method applied to nonlinear circuits. Galerkin's method, first described in 1915, assumes a

solution containing unknown coefficients. The assumed solution is substituted into the governing equations and the unknown coefficients adjusted so that the governing equations are satisfied as accurately as possible.

When the assumed solution is a sum of sinusoids, this procedure has been referred as harmonic balance. The name appears as early as 1937 in the work of the Ukrainian scientists Kryloff and Bogoliuboff (translated into English in 1943). More recently, the method has been developed and applied to nonlinear circuits by Baily and Lindenlaub in the 1960s. The modern version of the harmonic balance method was presented by Nakhla and Vlach in 1976. They reduce the number of variables to be optimized by partitioning the network into smaller subnetworks that are composed of either linear circuit elements or nonlinear elements. The linear subnetworks are solved in the frequency domain. Only the variables associated with the connection of the subnetworks need to be optimized. They called the resulting technique piecewise harmonic balance. In recent years, their method has been adopted and the adjective piecewise is usually dropped.

The harmonic balance method is an efficient method for the simulation of steady-state response because the form of the solution is a priori imposed onto the equations. A linear combination of sinusoids is chosen as the basis and the coefficients of those sinusoids then become the circuit unknowns. This differs from the time domain method, where a set of time samples is chosen and tied to adjacent time samples only through the circuit equations. The advantage of making no a priori assumption about the functional form of the solution is that the waveform can have any spectral content (subject of course to the band-limiting effect of sampling). However, now a set of

(time-invariant) state variables describes the circuit only at a single time instant.

In harmonic balance, imposing the condition that the circuit is already operating in steady-state enables a time-invariant set of variables to identify the circuit at all time instants. The set typically consists of the coefficients of the sinusoids for each circuit variable i.e. their spectral or phasor content, although an alternative representation is a set of time points within a single period, for each variable.

In the last few years, the harmonic-balance method has gained widespread acceptance among microwave engineers as a simulation tool for nonlinear circuits. The main advantages of this approach are its ability to directly address the steady-state circuit operation under single- or multiple-tone excitation, and its full compatibility with the characterization of the linear subnetwork in the frequency domain, which is usually a prerequisite for high-frequency applications.

In exchange for this, the HB method suffers from a number of shortcomings which have traditionally restricted its domain of applicability to selected aspects of the general nonlinear CAD problem. By harmonic balance, a nonlinear circuit analysis is reduced to the solution of a nonlinear algebraic system, which is usually solved by some sort of iterative procedure. In traditional HB simulators as the exciting signal levels are increased, the system becomes more and more ill-conditioned, and the iteration slows down and eventually fails. Thus is taken usually for granted that harmonic balance handles extremely nonlinear behavior only poorly.

The size of the solving system is equal to the number of state variables

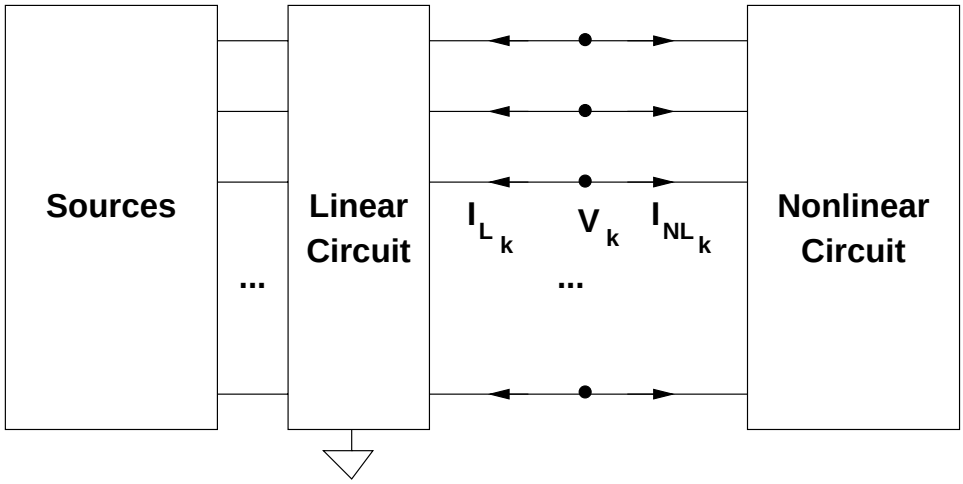

Figure 2.1: Partition representation of the time-invariant harmonic balance method

times the number of spectral lines. For multiple-device circuits excited by multiple tones, this may lead to nonlinear problems with thousands of unknowns, which may be impossible to deal with by conventional techniques. This fatally places an upper bound to the size of the circuit problems that can be solved by the HB technique, from the stand point of both memory occupation and CPU time. One of the main challenges of harmonic-balance simulation is the extension of this bound.

2.2 Formulation of the HB equations

2.2.1 Conventional formulation

The idea of harmonic balance [9] is most easily demonstrated using the approach of Nakhla and Vlach [6]. This approach is based on partitioning of the linear and nonlinear portions of a circuit as shown in Figure 2.1.

A solution at the k^{th} node up to the n^{th} harmonic is assumed of the form

$$v_k(t) = \Re\left\{\sum_{m=0}^{n} V_m \exp(j\omega m t + \phi_m)\right\} \tag{2.1}$$

with the unknown variables being the amplitude and phase of each frequency component at each interfacial node. Note that phasor amplitude and phase at each frequency is fixed with respect to time. The solution is, then, a finite sum of commensurate discrete spectra. Conservation of current and charge in the frequency domain at each of the interfacial nodes of Figure 2.1 gives

$$F(V) = I(V) + \mathbf{\Omega}Q(V) + \mathbf{Y}V + I_S = 0 \tag{2.2}$$

where $\mathbf{Y}$ is the nodal admittance matrix of the linear part and I_S is the source vector. The first two terms on the right side are calculated as:

$$I_{NL}(V) = \mathcal{F}\{\mathbf{i}[\mathcal{F}^{-1}(V)]\} + \mathbf{\Omega}\mathcal{F}\{\mathbf{q}[\mathcal{F}^{-1}(V)]\} \tag{2.3}$$

where $\mathcal{F}$ is the Fourier transform operator and $\mathbf{\Omega}$ is a diagonal matrix of $j\omega m$ terms representing frequency domain differentiation. Solution is carried out in the frequency domain, with nonlinearities transformed to the frequency domain using FFT with precisely spaced time samples.

2.2.2 State-Variable Formulation

Let the nonlinear subnetwork be described by the following generalized parametric equations [3]

$$\mathbf{v}(t) = u[\mathbf{x}(t), \frac{d\mathbf{x}}{dt}, \ldots, \frac{d^n\mathbf{x}}{dt^n}, \mathbf{x}_D(t)] \tag{2.4}$$

$$\mathbf{i}(t) = w[\mathbf{x}(t), \frac{d\mathbf{x}}{dt}, \ldots, \frac{d^n\mathbf{x}}{dt^n}, \mathbf{x}_D(t)] \tag{2.5}$$

where $v(t)$, $i(t)$ are vectors of voltages and currents at the common ports, $\mathbf{x}(t)$ is a vector of state variables and $\mathbf{x}_D(t)$ a vector of time-delayed state variables, i.e., $x_{Di}(t) = x_i(t - \tau_i)$. The time delays τ_i may be functions of the state variables. All vectors in (2.5) have a same size n_d equal to the number of common (device) ports. This kind of representation is very convenient from the physical viewpoint, because it is in fact equivalent to a set of implicit integro-differential equations in the port currents and voltages. This allows an effective minimization of the number of subnetwork ports, and what is more important, results in extreme generality in device modeling capabilities.

The quasi-periodic electrical regime of the nonlinear circuit resulting from a multi-tone excitation is completely defined by a set of time-dependent state variables of the form (2.1), or equivalently by the vector X of the real and imaginary parts of their harmonics.

The linear subnetwork may be represented by the frequency domain equation

$$\mathbf{Y}(\omega)V(\omega) + N(\omega) + I(\omega) = 0 \tag{2.6}$$

where $V(\omega)$, $I(\omega)$ are vectors of voltage and current phasors, $Y(\omega)$ is the linear subnetwork admittance matrix, and $N(\omega)$ is a vector of Norton equivalent current sources. Thus a set of complex harmonic-balance errors at a generic IM product Ω_k has the expression

$$F_k(X) = \mathbf{Y}(\Omega_k)U_k(X) + N(\Omega_k) + W_k(X) \tag{2.7}$$

The nonlinear analysis problem is reduced to the solution of a nonlinear algebraic system by imposing that all the HB errors vanish. In order to avoid the use of negative frequencies, the nonlinear solving system is formulated in

terms of a vector F of real and imaginary parts of the HB errors given by (2.7), and thus written as a systems of N_t real equations in N_t unknowns, namely

$$F(X) = 0 \tag{2.8}$$

2.3 Frequency-Time conversion

Until 1984, harmonic balance was only used to analyze circuits with a periodic response [17]. The reason for this is that with the linear devices evaluated in the frequency domain and the nonlinear devices evaluated in the time domain, a transform is needed to convert signals between the two domains. The FFT was used to perform this operation, however the FFT is only applicable to periodic signals, which excludes a very important class of circuits whose steady state response is not periodic: mixers. The signals present in mixers consist of sinusoids at the sum and difference frequencies of the two or more input frequencies and their harmonics. In general, these input frequencies are not harmonically related, and so the signals found in mixers are not periodic, but rather quasi-periodic.

There are currently five different methods available for transforming signals between time and frequency domains that are suitable for use with quasi-periodic harmonic balance. The first three of these methods are general in nature. Two methods, those of Ushida and Chua [25], and Gilmore and Rosenbaum [27], are considered less efficient than the remaining methods and thus are no longer used. The third method is the almost periodic Fourier transform (APFT) [17]. This transform has a simple operator notation and so is useful for theoretical manipulation.

The last two methods exploit the fact that in harmonic balance one desires the frequency domain response of a nonlinear device to a frequency domain stimulus and the time domain waveforms generated along the way are of no interest. These methods transform spectra into waveforms with distorted time axis back into the spectra, and so as long as the nonlinearities being evaluated in the time domain are algebraic, which was the basic assumption with harmonic balance, then the resulting spectra are correct. Of these two remaining methods, the first is based in the multidimensional DFT [28, 26], and it is restricted to box truncation. The second is based on the one dimensional DFT. It is faster and has fewer restrictions than the multidimensional DFT approach, and is the one presented here [17].

2.3.1 Artificial frequency mapping

This method allows the use of the DFT — and hence the FFT — with harmonic balance even when the desired solution is quasi-periodic. Here are the conditions to use this method:

1. The nonlinearities must be algebraic and time invariant.
2. The signals must be quasi-periodic.
3. The time domain signals must be of no interest.

Consider a nonlinear resistor with the constitutive equation

$$i(v) = v^2$$

Assume that the resistor is being driven with the voltage waveform

$$v(t) = \cos(\alpha t) + \cos(\beta t)$$

The resistor responds with a current waveform of

$$i(v(t)) = 1 + \frac{1}{2}\cos(2\alpha t) + \cos(\alpha t - \beta t) + \cos(\alpha t + \beta t) + \frac{1}{2}\cos(2\beta t)$$

Notice that the coefficients of the cosines (i.e. the spectrum of the response signal) are independent of the frequencies α and β. This is true, whenever the nonlinearities are algebraic. Thus, for the purpose of evaluating the nonlinear devices, the actual fundamental frequencies are of no importance and can be chosen freely. In particular, the fundamentals can be chosen to be multiples of some arbitrary frequency so that the resulting signals will be periodic. Once the fundamentals are chosen in this manner, the DFT can be used. It is important to realize that these artificially chosen fundamental frequencies are not actually used in the harmonic balance equations, such as Y and Ω. Rather, they are used when determining in which order to place the terms in the spectra so that the DFT can be used when evaluating the nonlinear devices.

For simplicity, the way in which the artificial frequencies are chosen is illustrated by examples. The actual artificial frequencies, and the scale factors that convert the original fundamental to the artificial frequencies, are of no interest except in determining the correspondence between the quasi-periodic and periodic harmonic indices.

Consider the following set of frequencies:

$$\Lambda_K = \{\omega : \omega = k_1\lambda_1 + k_2\lambda_2; 0 \le k_1 \le H_1, |k_2| \le H_2, k_1 \ne 0 \text{ if } k_2 < 0\}$$

Let $\alpha_1 = 1$ and $\alpha_2 = \lambda_1/[\lambda_2(2H_2 + 1)]$ be the scaling factors of the two fundamentals. Then the scaled set of frequencies is equally spaced and no

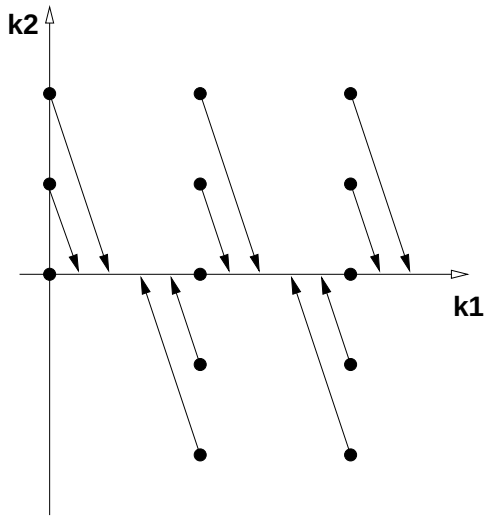

Figure 2.2: The mapping of quasi-periodic frequencies into periodic frequencies for box truncations.

two frequencies overlap. This scaling, which is ideal for box truncations, is illustrated in Figure 2.2. The correspondence between original and artificial frequencies is given by

$$k\lambda_0 = k_1\alpha_1\lambda_1 + k_2\alpha_2\lambda_2$$

where

$$\lambda_0 = \frac{\lambda_1}{2H_2 + 1}$$

and

$$k = (2H_2 + 1)k_1 + k_2$$

This approach can be extended to the case where more than two fundamentals are applied or to other truncation schemes.

2.4 Solution Strategies for Solving the Equations

The solution of the set of equations can be obtained by several methods. One method, known as relaxation uses no derivative information and is relatively simple and fast, but it is not robust. Alternatively, gradient methods can be used to solve either a system of equations (e.g. using Newton-Raphson) or to minimize an objective function using a quasi-Newton or search method.

2.4.1 Relaxation Methods

Relaxation methods [8] use fixed point iteration to solve (2.2). Rewriting (2.2) yields

$$YV^{j+1} = -I(V^j) - \mathbf{\Omega}Q(V^j) - I_s \tag{2.9}$$

where real and imaginary components are now being separated to eliminate complex quantities.

The fixed point relaxation method was first used by Kerr and later by Hicks and Khan in determining the local oscillator waveforms at mixer diodes. It has also been used in the harmonic balance procedure to analyze MESFET amplifiers. The relaxation method, when successful, can result in much faster computation times than gradient methods. This is partially due to decoupling of the frequency components, as their values can be computed separately in a nonlinear function, and they are held constant when applied to the linear network. The iteration of harmonics is associated only with the calculation in the nonlinear network, because the linear network is independent at each harmonic. Its main advantage is that a Jacobian is not required

with consequent reduction in memory and CPU usage.

2.4.2 Optimization and Minimization Methods

This method [8] uses an optimizer to minimize $F(V)F^T(V)$, i.e., it seeks a least squares minimum of an error function representing the residual currents flowing into each node. The optimizer will typically be a global optimizer that uses a quasi-Newton method. The method tends to be inefficient because information about each of the contributors to the error function is lost in the calculation of the sum of the squares. However, the advantage of minimizing an objective function using an optimizer is that any arbitrary circuit parameters (such as power or frequency) can be added into the objective function and so be optimized simultaneously.

2.4.3 Newton Method

These methods [8] have the best convergence properties of all the methods and, provided that the initial solution guess is close, the rate of convergence is quadratic (i.e. at each subsequent iteration the error is the square of the previous error).

With a Newton-Raphson type solution method, eq. 2.2 can be solved iteratively as:

$$V^{j+1} = V^j - \mathbf{J}^{-1}F(V^j) \tag{2.10}$$

where $\mathbf{J}$ is the network Jacobian function.

The Newton method is the most commonly used approach in harmonic balance simulators, and generally the best one to achieve convergence. A major drawback can be the amount of memory required to store and invert

the Jacobian matrix, and the speed with which it can be inverted. The methods fails if the Jacobian is singular. However, the Jacobian itself is very useful as its properties enable it to be used for other purposes as well. In particular, the Jacobian can be used in the determination of stability, in calculating the conversion matrix for mixers, and in the adjoint technique for calculating the sensitivity of nonlinear circuits.

The Jacobian can be formulated either analytically or numerically. Numerical calculation of the Jacobian can be effective when the nonlinearities are not very strong, but for very nonlinear circuits the error introduced in the Jacobian estimation results in inaccurate updates of the unknowns and a large number of iterations and possibly non-convergence. Analytical calculation of the Jacobian involves considerable calculation and transformation, but it is the preferred method and still requires less computation than numerical evaluation of the Jacobian.

The Jacobian matrix is

$$\mathbf{J} = \frac{\partial F}{\partial V} \tag{2.11}$$

and consists of terms $J_{mn}(V)$ such as

$$\frac{\partial F_m(V)}{\partial V_n}, m, n = 1, 2, 3, ..., N \tag{2.12}$$

where m and n are node numbers, and

$$\frac{\partial F_m(V)}{\partial V_n} = \frac{\partial I_m(V)}{\partial V_n} + \Omega_{mn}\frac{\partial Q_m(V)}{\partial V_n} + Y_{mn} \tag{2.13}$$

Now, $\partial I_m/\partial V_n$ is the derivative of the harmonic component of current at the m^{th} node with respect to the harmonic component voltage at the n^{th} node. The derivative can only be formulated analytically in the time domain using

the chain rule:

$$I_m(V) = \Gamma i_m(v) \tag{2.14}$$

where Γ is the matrix multiplier representing the DFT.

$$\frac{\partial I_m(V)}{\partial V_n} = \Gamma \frac{\partial i_m(v)}{\partial v_n} \frac{\partial v_n}{\partial V_n} \tag{2.15}$$

Now $i(v)$ is just the known nonlinear time domain function for the model, and if it is algebraic, its partial derivative can be found analytically. Finally, using the fact that $\Gamma^{-1} V_n = v_n$, we have

$$\frac{\partial I_m(V)}{\partial V_n} = \Gamma \frac{\partial i_m(v)}{\partial v_n} \Gamma^{-1} \tag{2.16}$$

It appears that this approach requires both a forward and an inverse Fourier transform for each component of the Jacobian. However, it can be arranged so that a number of scalar multiplications followed by a single transform are required for the derivative.

2.4.4 Newton-based methods

The major drawback of Newton's method is that it is not globally convergent so, in practice, it fails too often to be a useful general method.

In the case of large circuits, or strongly nonlinear ones, the Jacobian becomes large, and a great contribution to the computation burden is due to the LU decomposition required by each iteration [22]. In the literature, several algorithms have been presented.

In the Newton-Samanskii method [8], the Jacobian is re-used in a 'chord' iteration. There is also the block Newton method, in which terms relating different frequency components are deleted from the Jacobian. Thus the

Jacobian is block diagonal and the inversion of it requires the inversion of the much smaller submatrices [20]. The block Newton iteration scheme can be combined with the chord method for a further reduction in computational complexity. However these methods can only be employed for almost linear circuits, since they present poor convergence properties.

The globally convergent methods [13] are the *line search methods* and the *trust region methods*. These methods have better convergence properties when the initial guess is not close to the solution.

The line search method expands on the strict Newton method. The point generated by equation 2.10 is not necessarily accepted as a new point; it must meet some acceptance criterion. To do this, some measure of merit of the current point is required, i.e., is this point preferred versus the previous ones? A common method is to define an objective function f such that

$$f = \frac{1}{2}F(X)F^T(X) \tag{2.17}$$

where the factor $1/2$ is for convenience only. When f is at its global minimum, 0, a solution has been found. However, local minima of f can exist which are not solutions; this is the main downfall of global Newton-based methods.

The trust region [14] method attempts to combine the better global convergence properties of searches in the steepest descent direction with the local quadratic convergence of the Newton method. It is similar to the well-known Levenberg-Marquardt method for nonlinear least squares.

There exist methods to update the inverse of the Jacobian matrix at each iteration without having to perform neither the derivative calculation nor the matrix decomposition. These are known as *quasi-Newton* update methods

(for example, Broyden's method). These methods can be combined with Newton-based methods to reduce the overall calculation time.

There are reports of using variations of the Newton-Powell hybrid method to solve the harmonic balance equations [22]. This method is one of the so called trust region methods combined with quasi-Newton updates. The convergence of this method is very good.

Also, variations of globally convergent methods have been used to solve the harmonic balance equations [3]. These methods combined with a good election of the state variables are the best solution found until now for the harmonic balance problem.

2.4.5 Solution of very large systems

The only outstanding drawback of HB simulation lies in the huge demand of computer resources when the problem size becomes large. This is due to the fact that the storage of the Jacobian requires N^2 words, and its factorization time using direct methods is $O(N^3)$, where N is the number of scalar unknowns.

One approach to overcome this problem is to set a large percentage of the Jacobian matrix to zero according to a suitable numerical or physical criterion, and a sparse-matrix solver is used for the linear system [3]. Generally speaking, the penalty for the use of such techniques is a considerable limitation on the power-handling capabilities of the HB simulator.

The current proposed approaches [1, 4, 5] are usually aimed at avoiding the storage and factorization of the full Jacobian matrix. These methods are called inexact Newton methods. They use iterative techniques to ap-

proximately solve the Newton equation, thus avoiding the storage and factorization of the Jacobian. High numerical efficiency is obtained by suitably choosing the accuracy requirements for the approximate solution at each step. The iterative techniques to solve the linear system are Krylov subspace methods with GMRES and QMR being the most commonly used.

Chapter 3

Improved Harmonic Balance

3.1 Nonlinear Equation Formulation

In this chapter the harmonic equations are formulated with the minimum number of unknowns starting from a modified nodal admittance matrix of the linear part of the circuit. This approach has the advantage that we keep all the flexibility of the modified nodal admittance matrix, and at the same time has the advantages given by the state variable approach.

The system equation formulation begins with the partitional network of Figure 3.1, with the nonlinear elements replaced by variable voltage or current sources [6]. For each nonlinear element one terminal is taken as reference and the element is replaced by a set of sources connected to the reference terminal. Clearly, the state of the element can be determined considering only the current of the sources, or the voltages across the sources, or some combination of the voltages and currents. Identifying the local element reference eliminates one potential state variable and one error function term that would otherwise be considered if all of the nonlinear element terminals were treated equally. For example, in a conventional voltage-current formu-

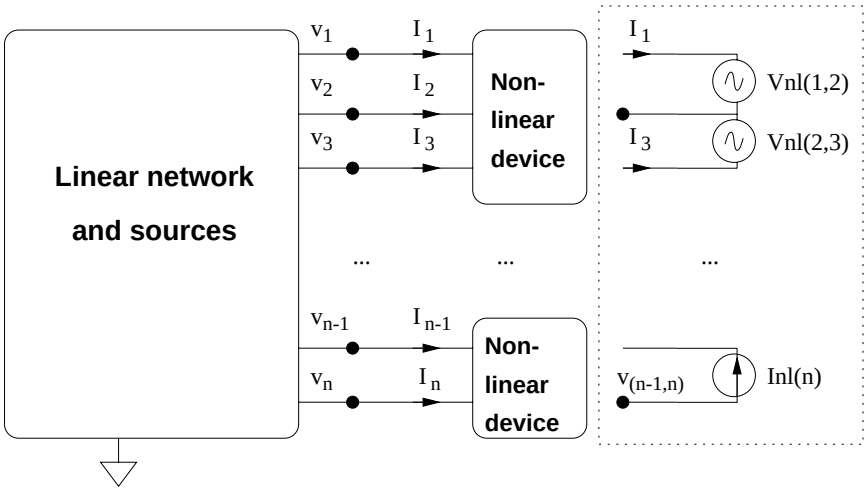

Figure 3.1: Network with nonlinear elements.

lation the voltage and current associated with the reference node need not be considered in the nonlinear formulation. With the three-terminal element in Figure 3.1, terminal 2 is chosen as the reference node and only two voltages, $V_{12} = (V_1 - V_2)$ and $V_{32} = (V_3 - V_2)$, and two currents, I_1 and I_3, are needed.

Using the *state variable* concept, let $\mathbf{X}$ be the *state variable vector*. Each element of $\mathbf{X}$ is a vector with all the frequency components of the same state variable. Therefore, this leads to the following matrix

$$\mathbf{X} = \begin{bmatrix} x_{1,f0} & x_{1,f1} & x_{1,f2} & \cdots & x_{1,fm} \\ x_{2,f0} & x_{2,f1} & x_{2,f2} & \cdots & x_{2,fm} \\ \vdots & & & & \\ x_{n,f0} & x_{n,f1} & x_{n,f2} & \cdots & x_{n,fm} \end{bmatrix} \tag{3.1}$$

We denote X_i as the i^{th} row of the matrix, i.e., all the frequency components of the i^{th} state variable.

Now the equations can be formulated. Unlike [3], we formulate the equations in the frequency domain. The current and voltage at the i^{th} terminal

of each nonlinear device are

$$V_i = V_i(X_k, \ldots, X_l) \tag{3.2}$$

$$I_i = I_i(X_k, \ldots, X_l) \tag{3.3}$$

Where the element under consideration depends only on the k^{th} to l^{th} components of the state variable vector. Since the state variable approach is used both the current and the voltage of the nonlinear elements are needed for developing the error function of the network.

Since we have replaced the nonlinear elements by sources, the modified nodal admittance matrix (MNAM) of the entire circuit can be written for each frequency:

$$\mathbf{M}_{fi} V_{fi} = S_{fi} \tag{3.4}$$

The source vector S_{fi} is composed of the fixed sources present in the circuit and the state variable-dependent sources due to the nonlinear elements. For simplicity, assume that all the nonlinear devices are replaced by current sources. Let $I_{NL,fi}(X)$ be the vector with the currents of all the nonlinear elements at frequency f_i. These currents are 'applied' to the linear circuit. Then, the source vector can be written as:

$$S_{fi} = S_{\text{fixed},fi} + \mathbf{W} I_{NL,fi}(\mathbf{X}) \tag{3.5}$$

Where $\mathbf{W}$ is a matrix that maps the nonlinear currents to the proper place in the source vector. This matrix is composed only of zeros, ones and minus ones, and it is the same for all frequencies, since it only contains topology information.

Provided that the MNAM is not singular, we can calculate the voltages at the linear circuit given the values of the state variables $\mathbf{X}$:

$$V = \mathbf{M}_{fi}^{-1} S_{\text{fixed},fi} + \mathbf{M}_{fi}^{-1} \mathbf{W} I_{NL,fi}(\mathbf{X}) \tag{3.6}$$

We denote $U_{NL,fi}(\mathbf{X})$ the vector with the voltages of all the current sources due to the nonlinear elements at frequency f_i (port voltages of the nonlinear elements). The error function is formulated by comparing the port voltages of the nonlinear elements $U_{NL,fi}(\mathbf{X})$ with the port voltages at the linear circuit $U_{L,fi}(\mathbf{X})$. In order to obtain $U_{L,fi}(\mathbf{X})$, we need to perform differences between the nodal voltages of the current source's terminals. This can be accomplished by multiplying the nodal voltage vector V by a matrix $\mathbf{T}$ such that the result are the desired port voltages. It can be shown that $\mathbf{W}$ is the transpose of $\mathbf{T}$ so that:

$$U_{L,fi}(\mathbf{X}) = \mathbf{T}\mathbf{M}_{fi}^{-1} S_{\text{fixed},fi} + \mathbf{T}\mathbf{M}_{fi}^{-1} \mathbf{T}^T I_{NL,fi}(\mathbf{X}) \tag{3.7}$$

The first term in this sum is a constant vector, and the second is a constant matrix multiplied by a function of $\mathbf{X}$. Then,

$$S_{sv,fi} = \mathbf{T}\mathbf{M}_{fi}^{-1} S_{\text{fixed},fi} \tag{3.8}$$

$$\mathbf{M}_{sv,fi} = \mathbf{T}\mathbf{M}_{fi}^{-1} \mathbf{T}^T \tag{3.9}$$

$S_{sv,fi}$ is the state variable source vector, and the matrix $\mathbf{M}_{sv,fi}$ is the state variable impedance matrix.

The solution of the problem is achieved by finding the zero of each error function

$$F_{fi}(\mathbf{X}) = U_{L,fi}(\mathbf{X}) - U_{NL,fi}(\mathbf{X}) = 0 \tag{3.10}$$

The complete system of equations is the following,

$$F(\mathbf{X}) = \begin{bmatrix} S_{sv,f0} + \mathbf{M}_{sv,f0} I_{NL,f0}(\mathbf{X}) - U_{NL,f0}(\mathbf{X}) \\ S_{sv,f1} + \mathbf{M}_{sv,f1} I_{NL,f1}(\mathbf{X}) - U_{NL,f1}(\mathbf{X}) \\ \vdots \\ S_{sv,fm} + \mathbf{M}_{sv,fm} I_{NL,fm}(\mathbf{X}) - U_{NL,fm}(\mathbf{X}) \end{bmatrix} = \mathbf{0} \tag{3.11}$$

The resulting system of equations is composed of $m + 1$ systems each with n equations. If f_0 is denoted as being DC, then the dimension of the system (and the number of unknowns) is $(2m + 1)n$, since for each state variable a real component is needed for DC, and a real and imaginary component for each alternating frequency. This is the minimum number of unknowns that can be achieved, and it only depends on the nonlinear elements and the frequencies considered.

After solving (3.11), the value of the state variable vector is known, so finding the voltages (and the current of the ideal voltage sources) for the entire network is straightforward using Equation (3.6).

Note that the entire analysis was performed in the frequency domain. All the frequency mixing is performed in the functions of Equation (3.2), i.e., inside the nonlinear devices. The particular method used to calculate the currents and voltages from the state variables in the frequency domain is not relevant here.

3.2 Implementation in Transim

The goal in designing the program was to allow speed in development, to use "off the shelf" advanced numerical techniques, and to allow easy expansion and testing of new models and numerical methods.

The frame for the harmonic balance routine is the program *Transim*. It provides netlist parsing, a library with many linear microwave elements, and output routines to display voltages, currents, state variables and operations among them.

Figure 3.2 shows the flow diagram of the program. Note that the nonlinear equation solver knows nothing about harmonic balance. This fact allows us to integrate any general solver into the program. The integration is done by writing a file containing interface routines.

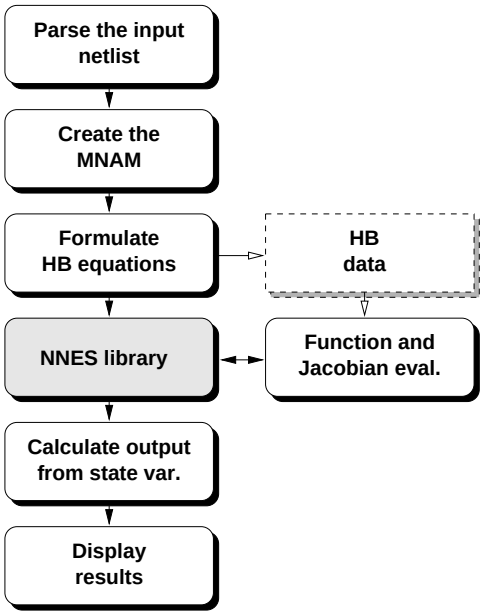

Figure 3.2: General flow diagram of the program.

Several free software packages have been used. They are listed in Table 3.1.

Sparse	Sparse matrix library
NNES	Nonlinear equation systems solver
Adol-c	Automatic differentiation package
Gnuplot	Plotting utility

Table 3.1: Software packages used in Transim

3.2.1 Modified nodal admittance matrix representation

For the MNAMs, the package *Sparse 1.3* [16] is used. It is a flexible package of subroutines written in C that quickly and accurately solve large sparse systems of linear equations. It also provides utilities such as MNAM reordering, etc. There is one sparse matrix per frequency. The linear system solving capability is used to calculate the matrices of Equations (3.8) and (3.9).

3.2.2 Frequency-time conversion

Each nonlinear element has a function that given the value of its state variables in the frequency domain, returns the phasors of the currents and voltages at its ports. The actual calculation is performed in the time domain, using the FFT to convert between time and frequency domain [23]. Multi-tone analysis is supported by using artificial frequency mapping described in 2. This approach is more efficient than the multidimensional Fourier transform, and it is simpler to implement [17]. One of the advantages is that the analysis remains essentially the same as with single-tone excitation.

3.2.3 Solution of the nonlinear system

The nonlinear system is solved using the library *NNES* [13]. It is written in Fortran and it provides Newton and quasi-Newton methods with many options, such as the use of analytic Jacobian or forward, backwards or central differences to approximate it, different quasi-Newton Jacobian updates, two globally convergent methods, etc.

After formulating all the matrices needed for the error function ($S_{sv,fi}$ and $\mathbf{M}_{sv,fi}$ for each frequency), the NNES routine is called using an interface function. When function Jacobian evaluation is needed, NNES calls Transim function evaluation routines. All the harmonic balance information is kept in a global structure, so that the only information really needed by the Transim routines is the value of the state variable vector. In this way, almost any nonlinear solver can be integrated. Many options used by NNES are set in the input netlist, most of them have default values. The default method is a combination of line search and Newton. The factored Jacobian is updated using the Broyden scheme (other alternatives are available). This approach increases the speed of the analysis due to the saving in the time to invert the Jacobian.

3.2.4 Automatic differentiation

The analytic Jacobian is calculated in the routine using *Adol-C* [15]. This is a software package written in C and C++ that performs automatic differentiation. The numerical values of derivative vectors are obtained free of truncation errors at a small multiple of the run time and randomly accessed memory of the given function evaluation program.

The implementation of the automatic differentiation in Transim is shown in Figure 3.3. The element routines are called at the first evaluation of the error function. In that call the operations to calculate the currents and voltages of each element are recorded by adol-c in a *tape*, that is actually a internal buffer. After that, each time that the values or the derivatives of the nonlinear elements are required, a adol-c function is called and the values are calculated using the tapes. In this way, the functions are taped only once. This implementation is efficient because the taping process is done only once at the beginning of the analysis. This almost doubles the speed of the calculation comparing with the case where the functions are taped each time. When the Jacobian is needed, the corresponding adol-c function is called using the same tapes. We have tested the program with large circuits with many tones, and the function or Jacobian evaluation times are always very small compared with the time required to invert the Jacobian. We can conclude that there is no detriment in the performance of the program by using automatic differentiation, and the advantages from the development view point are evident.

3.2.5 Display of results

To display graphs, the program *Gnuplot* is used. Transim call this program at the end of the analysis and display the graphs requested in the input netlist. This feature is available to all analysis types in Transim.

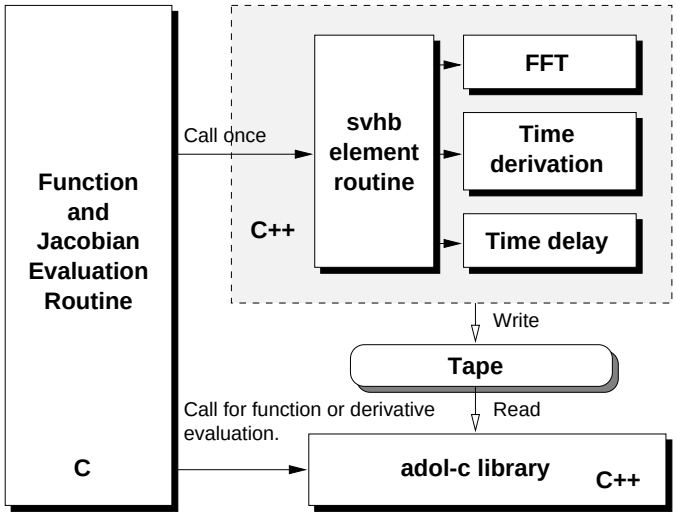

Figure 3.3: Implementation of automatic differentiation.

3.3 Adding nonlinear elements

It is relatively simple to add a new element or model to the program. That is because each element is self-contained in its own files, so there is no need to modify the source files of the rest of the program. Basically, to add an element consists of writing one routine that calculates the currents and voltages at the terminals of the element as a function of the element state variables in the frequency domain. There is a library of functions provided to convert vectors from frequency domain to time domain and *vice-versa*, and functions to calculate derivatives with respect to time and time-delay in the frequency domain. No derivation with respect to the state variables is needed.

As the input and output of the routine is entirely in the frequency domain, the Fourier transform is provided as a tool to evaluate the nonlinear function, but the model could be written entirely in the frequency domain if this is

desired.

In this way, it is easy to add and test new models. Besides that, the user may try several numerical methods to solve the system equations.

3.3.1 Example: Diode Element

To illustrate the creation of a nonlinear element, we will show here the routine for a microwave diode using Rizzoli's parametric model [3]. The actual code in Transim includes modeling of capacitances and nonlinear resistance, but we show here a simple case for didactic purposes. The conventional current equation for the diode is

$$i(t) = I_s(\exp(\alpha v(t)) - 1) \tag{3.12}$$

The parametric model is the following

$$v(t) = \begin{cases} x(t) & \text{if } x(t) \leq V_1 \\ V_1 + \frac{1}{\alpha}\ln(1 + \alpha(x(t) - V_1)) & \text{if } x(t) > V_1 \end{cases} \tag{3.13}$$

$$i(t) = \begin{cases} I_s(\exp(\alpha x(t)) - 1) & \text{if } x(t) \leq V_1 \\ I_s \exp(\alpha V_1)(1 + \alpha(x(t) - V_1)) - I_s & \text{if } x(t) > V_1 \end{cases} \tag{3.14}$$

where V_1 is some threshold value. The model requires the current, voltage and derivatives to be continuous at $x = V_1$. This implies that

$$V_1 = \frac{\ln(G_1/\alpha I_s)}{\alpha} \tag{3.15}$$

where G is the slope $\partial i/\partial v$. The suggested value for this slope in [3] is 1. Nevertheless, we have achieved much better convergence using the value 0.01.

The flow diagram of the diode routine (or any normal element routine) can be seen in Figure 3.4. The C++ code for this example follows. Note

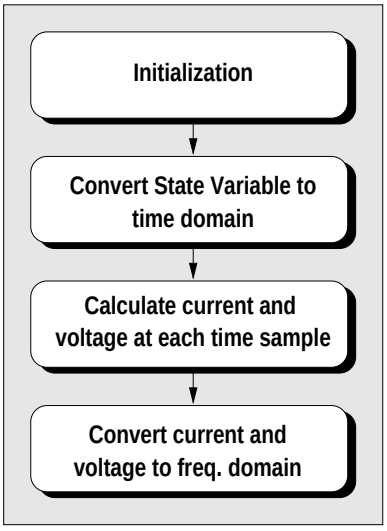

Figure 3.4: Flow diagram of the diode routine.

that we are required to substitute all branches in the routine flow by conditional assignments. For this purpose, Adol-C provides a special function called **condassign(a,b,c,d)**. Its calling sequence sequence corresponds to the syntax of the conditional assignment $a = (b > 0)?c : d$, which C++ inherited from C. In the routine elements, this function should be accessed as **cassgn(a,b,c,d)**.

```
/****************************************************************
 *
 * Contains the state variable harmonic balance routine
 * with automatic differenciation code for a simple diode element
 *
 ****************************************************************/

extern "C" {
#include <math.h>
#include "cap.h"
#include "par.h"
#include "svHB.h"
#include "el_diode.h"
}
```

```
#include "adol-c/adouble.h"
#include "svHB_timefreq.h"

/*************************************************************
 * Diode_svHB:
 *
 * Calculate the voltage and current of the diode given the values
 * of the corresponding state variable phasors in the frequency
 * domain.
 *************************************************************/
int Diode_svHB(IN gr_Id_t nid, IN doublem_t x,
       OUT doublem_t u_NL, OUT doublem_t i_NL)

{
  const double one = 1.;

  // v1: threshold voltage (for state variable)
  double v1;

  double k3;
  int idx;
  DiodeBlock_Pt diode_P;

  // Get a diode element pointer.
  diode_P = (DiodeBlock_Pt) Gr_NodePhyData_P(nid);

  // calculate some constants.
  // G1 = di/dv at v=v1, G1 = .01
  v1 = log(.01 / (diode_P->alpha * diode_P->Is)) / diode_P->alpha;
  k3 = exp(diode_P->alpha * v1);

  // Begin recording of operations.
  traceon(nid.val);

  // Define the variables used by adol-c.
  adoublev xs(svHBdata->NoSamples);
  adoublev ud(svHBdata->NoSamples);
  adoublev id(svHBdata->NoSamples);

  // assign independent variables.
  xs <<= x[0];

  // Convert to the time-domain
  Time2Freq(xs, -1);
```

```
  Time2Freq(dxs_dt, -1);

  // Loop through all the time samples. Calculate the current
  // and voltage of the diode.

  for (idx=0; idx < svHBdata->NoSamples ; idx++) {

    // Calculate the voltage
    cassgn(ud[idx], v1 - xs[idx],
   xs[idx],
   v1 + log(one + diode_P->alpha*(xs[idx]-v1))/diode_P->alpha);

    // Calculate the current
    cassgn(id[idx], v1 - xs[idx],
   diode_P->Is * (exp(diode_P->alpha * xs[idx]) - one),
   diode_P->Is * k3 * (one + diode_P->alpha * (xs[idx] - v1)));

  }

  // Convert the time samples to freq.
  Time2Freq(ud, 1);
  Time2Freq(id, 1);

  // Assign dependent variables.
  ud >>= u_NL[0];
  id >>= i_NL[0];

  // End recording of operations.
  traceoff();

  return(0);
}
```

Chapter 4

Simulations and Analysis

4.1 Introduction

In this chapter we first describe a quasi-optical system. Then we show the results of simulation and measurements on that system. At last, we compare the performance of the harmonic balance simulation using different numerical techniques to solve the equations.

Quasi-optical systems combine the power of numerous solid state devices in free space. A typical quasi-optical system is the grid system shown in Fig. 4.1. A large number of active devices are distributed on the grid surface.

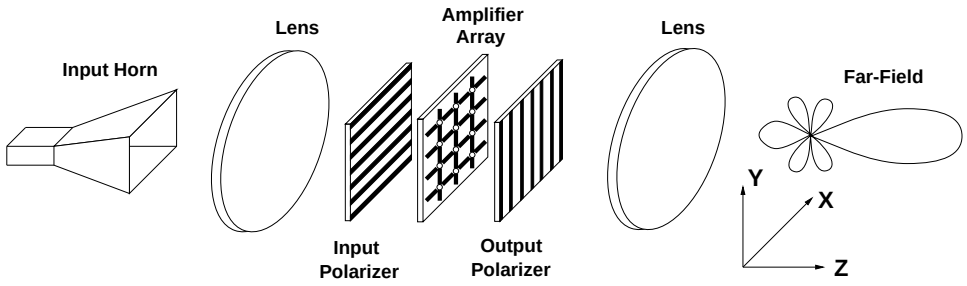

Figure 4.1: Quasi-optical lens system configuration with a grid amplifier array and polarizers

The grid is excited by a horn and lens system which concentrates the incident field on the grid. Polarizers are used to isolate the input and output.

Electromagnetic models are required to simulate these spatially distributed systems. The electromagnetic simulator produces the multi-port admittance matrix of the passive grid structure for inclusion in a microwave circuit simulation program. The admittance matrix produced is port based due to the lack of a global reference node in the spatially distributed structure. The multiport admittance matrix, along with a nonlinear active device model are used in a circuit simulator to model the performance of the spatial power combining system.

4.2 Simulation and Measurement Results

The layout for the system we consider in this chapter is shown in Figure 4.2. We will concentrate in the harmonic balance part of the simulation. For a detailed description of the whole simulation and measurements, see [21]. For the amplifiers, a behavioral model was used, so each amplifier is one nonlinear element for harmonic balance purposes. The quasi-optical grid was modeled by a multi-port admittance matrix produced by the program *Yomoma*, an electromagnetic simulator. The simulated and measured output power is shown in Fig. 4.3. The simulation follows the basic pattern of the measured power output of the grid amplifier.

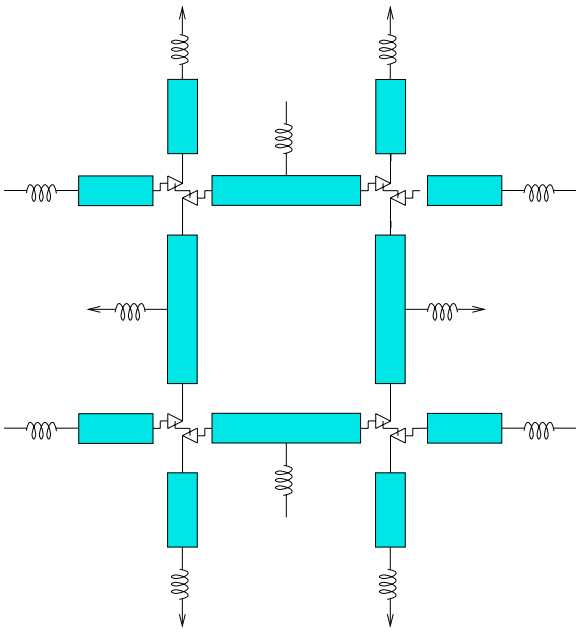

Figure 4.2: Layout for 2x2 quasi-optical grid amplifier with bias inductors shown.

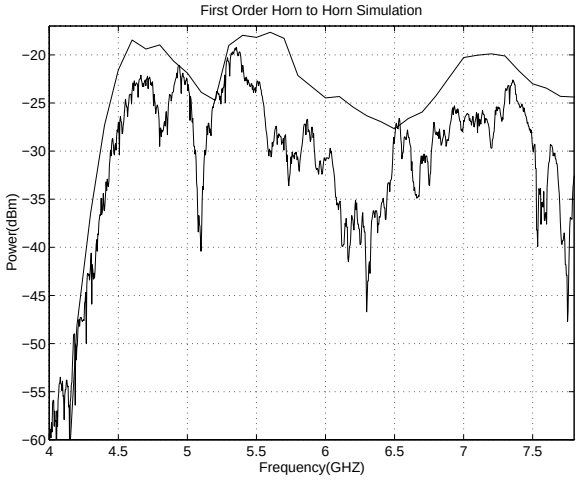

Figure 4.3: Power output for 2x2 grid amplifier system: solid line, measurement; dashed line, simulation.

Number of frequencies	4
Oversample	2
Number of state variables	32
System dimension	224
Set-up time	.84 s
Function evaluation time	.10 / .04 s
Jacobian evaluation time	.72 s
Maximum Jacobian Density	36 %

Table 4.1: Paramenters common to all simulations.

Numerical Method	Total Simulation Time (s)	Function Evaluations	Jacobian Evaluations	Convergence (yes/no)
Newton	39	6	5	yes
Abs. Newton	39	6	5	yes
Line Search	39	6	5	yes
Trust Region	40	6	5	yes
L.S. w/Broyden	28	13	1	yes
T.R. w/Broyden	26	13	1	yes

Table 4.2: Comparation of numerical methods

4.3 Analysis of the Running Time with Different Methods

Using the netlist of the described quasi-optical system, we will analyze the running time and stability of the harmonic balance simulator. Table 4.1 shows values common to all simulations. Table 4.2 shows the comparisons of some parameters using different numerical methods to solve the equations. For the function evaluation time, we have two values. The first value is when the tapes are generated, this happens only once during one simulation. The other figure is the normal evaluation time. We can observe that any method achieve convergence. This is so because the circuit is not strongly

nonlinear. There is some improvement in the total time when using Broyden quasi Newton Jacobian updates. The reason for that is the smaller number of Jacobian calculations and inversions. This is done at a expense of more function evaluations, because the convergence is somewhat slower, but the overall time is decreased. Note also that there is almost no time penalty for using globally convergent methods.

Chapter 5

Conclusions and Future Research

5.1 Conclusions

An state variable-based harmonic balance analysis was developed for the use of general and quasi-optical circuits. The simulator is very robust and flexible. The following are the main conclusion items:

- The equation formulation used allows the simulation of quasi-optical systems transparently, without ground problems and redundancies presented by other formulations.
- Globally convergent methods are the best choice for robustness and they do not slow down the analysis.
- Normally, the use of quasi-Newton Jacobian updates speed up the simulation.
- The Jacobian density increases with the nonlinearity in the circuit, therefore sparse-matrix techniques become inefficient for strongly non-

linear circuits. The dominant calculation for this kind of circuit is the Jacobian inversion.

- The use of well-chosen state variables makes easier to write the nonlinear element equations and improves the robustness of the analysis.
- The use of automatic differentiation makes the writing of nonlinear element models much easier.
- The convergence rate is greater using automatic differentiation than using differences.
- The use of automatic differentiation do not constitute a significant increase in the running time with respect to calculating the derivatives analytically.
- Multi-tone analysis can be handled very easily using artificial frequency mapping.

5.2 Future Research

There are several improvements that can be made to the current implementation of the harmonic balance in Transim.

The most important of them would be the use of an inexact Newton method to solve the equations when the system becomes very large (see section 2.4.5). This would virtually remove the limit on the size of circuit that can be simulated.

Also, currently there is a limit of two tones for the excitation. We can add multi-tone analysis without too much effort.

The simulation of a circuit is as good as the models used to represent the elements. Therefore, another area for further studies would be the inclusion of additional effects in the circuit elements such as temperature. Another issue concerning the element models is whether to use FFT or to calculate the response directly in the frequency domain, using techniques such as the arithmetic operator method [18, 19].

Another important improvement would be a re-writing of the element and network representation. It has grown so much that now not only it is cumbersome to add a new element or a new analysis type, but it also has some problems difficult to fix, if not impossible. These problems could be overcome using an object-oriented approach.

Bibliography

[1] V. Rizzoli, F. Mastri, F. Sgallari, G. Spaletta, "Harmonic-Balance Simulation of Strongly Nonlinear very Large-Size Microwave Circuits by Inexact Newton Methods", IEEE MTT-S Digest, 1996.

[2] V. Rizzoli, A. Costanzo, and A. Lipparini, "An Electrothermal Functional Model of the Microwave FET Suitable for Nonlinear Simulation" *International Journal of Microwave and Millimeter-Wave Computer-Aided Engineering*, Vol. 5, No. 2, 104-121 (1995).

[3] V. Rizzoli, A. Lipparini, A. Costanzo, F. Mastri, C. Ceccetti, A. Neri and D. Masotti, "State-of-the-Art Harmonic-Balance Simulation of Forced Nonlinear Microwave Circuits by the Piecewise Technique", IEEE Trans. on Microwave Theory and Techniques, Vol. 40, No. 1, Jan 1992.

[4] M. M. Gourary, S. G. Rusakov, S. L. Ulyanov, M. M. Zharov, K. K. Gullapalli, and B. J. Mulvaney, "Iterative Solution of Linear Systems in Harmonic Balance Analysis", IEEE MTT-S Digest, 1997.

[5] I. Moret, "On the Convergence of Inexact Quasi-Newton Methods", *International J. of Computer Math.*, Vol. 28, pp. 117-137, 1987.

[6] M. S. Nakhla, J. Vlach, "A Piecewise Harmonic Balance Technique for Determination of Periodic Response of Nonlinear Systems", *IEEE Trans. on Circuits and Systems*, Vol CAS-23, No. 2, Feb 1976.

[7] A. Materka and T. Kacprzak, "Computer Calculation of Large-Signal GaAs FET Amplifier Characteristics", *IEEE Trans. on Microwave Theory and Techniques*, Vol MTT-33, No. 2, Feb 1985.

[8] M. B. Steer, "Transient and Steady-State Analysis of Nonlinear RF and Microwave Circuits", ECE603 class notes, August 15, 1996.

[9] J. F. Sevic, M. B. Steer, and A. M. Pavio, "Nonlinear Analysis Methods for the Simulation of Digital Wireless Communication Systems", *International Journal of Microwave and Millimiter-Wave Computer-Aided Engineering*, Vol. 6, No. 3, 197-216, 1996.

[10] J. Kunisch and I. Wolff, "Steady-State Analysis of Nonlinear Forced and Autonomous Microwave Circuits Using the Compression Approach", *International Journal of Microwave and Millimeter-Wave Computer-Aided Engineering*, Vol. 5, No. 4, 241-255 (1995).

[11] E. Ngoya, A. Suárez Raphaël Sommet and R. Quéré, "Steady State Analysis of Free or Forced Oscillators by Harmonic Balance and Stability Investigation of Periodic and Quasi-Periodic Regimes", *International Journal of Microwave and Millimeter-Wave Computer-Aided Engineering*, Vol. 5, No. 3, 210-223 (1995).

[12] Compact Software, "Microwave Harmonica Elements Library", (1994).

[13] R. S. Bain, "NNES User's Manual", (1993).

[14] M. J. D. Powell, "A hybrid method for nonlinear equations", Numerical Methods for Nonlinear Algebraic Equations, P. Rabinowitz, Editor, Gordon and Breach, 1988.

[15] A. Griewank, D. Juedes, J. Utke, "Adol-C: A Package for the Automatic Differenciation of Algorithms Written in C/C++", Version 1.7, Sep 1996.

[16] K. S. Kundert, A. Songiovanni-Vincentelli, "Sparse User's Guide - A Sparse Linear Equation Solver", Department of Engineering and Computer Sciences, University of California, Berkeley, Calif. 94720 Version 1.3a, Apr 1988.

[17] K. S. Kundert, J. K. White and A. Sangiovanni-Vincentelli, "Steady-state methods for simulating analog and microwave circuits", Boston, Dordrecht, Kluwer Academic Publishers, 1990.

[18] M. B. Steer, C. Chang and G. W. Rhyne, "Computer-Aided Analysis of Nonlinear Microwave Circuits Using Frequency-Domain Nonlinear Analysis Techniques: The State of the Art", *International Journal of Microwave and Millimeter-Wave Computer-Aided Engineering*, Vol. 1, No. 2, 181-200, 1991.

[19] R. J. Gilmore and M. B. Steer, "Nonlinear Circuit Analysis Using the Method of Harmonic Balance—A Review of the Art. II. Advanced Concepts", *International Journal of Microwave and Millimeter-Wave Computer-Aided Engineering*, Vol. 1, No. 2, 159-180, 1991.

[20] C. R. Chang, "Computer-Aided Analysis of Nonlinear Microwave Analog Circuits Using Frequency-Domain Spectral Balance", Ph.D. Thesis, Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, 1990.

[21] M. A. Summers, "Simulation of a quasi-optical grid amplifier", M.S. Thesis, Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC, 1997.

[22] D. D'Amore, P. Maffezzoni and M. Pillan, "A Newton-Powell Modification Algorithm for Harmonic Balance-Based Circuit Analysis", *IEEE Transactions on Circuits and Systems—I: Fundamental Theory and Applications*, Vol. 41, No. 2, February 1994.

[23] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, "Numerical Recipes in C", Cambridge University Press, 1992.

[24] Y. Thodesen, K. Kundert, "Parametric harmonic balance", IEEE MTT S. International Microwave Symposium Digest, Vol 3, 1996, IEEE, Piscataway, NJ, USA, pp. 1361-1364.

[25] A. Ushida and L. O. Chua. "Frequency-domain analysis of nonlinear circuits driven by multi-tone signals", *IEEE Transactions on Circuits and Systems*, Vol. CAS-31, No. 9, September 1984, pp. 766-778.

[26] A. Ushida, L. O. Chua and T. Sugawara. "A substitution algorithm for solving nonlinear circuits with multi-frequency components", *International Journal on Circuit Theory and Application*, Vol. 15, 1987, pp. 327-355.

[27] R. J. Gilmore and F. J. Rosenbaum, "Modelling of nonlinear distortion in GaAs MESFETs", *1984 IEEE MTT-S International Microwave Symposium Digest*, May 1984, pp. 430-431.

[28] G. P. Bava, S. Benedetto, E. Biglieri, F. Filicori, V. A. Monaco, C. Naldi, U. Pisani and V. Pozzolo, "Modelling and perfomance simulation Techniques of GaAs MESFETs for microwave power amplifiers", *ESA-ESTEC Report*, Noordwijk, Holland, March 1982.

[29] H. Makino and H. Asai, "Relaxation-based circuit simulation techniques in the frequency domain", IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol E76-A, No. 4 Apr 1993, p 626-630.

[30] A. Brambilla, D. D'Amor, M. Pillan, "Convergence improvements of the harmonic balance method", Proceedings IEEE International Symposium on Circuits and Systems, Vol. 4 1993, Publ. by IEEE, IEEE Service Center, Piscataway, NJ, USA. p 2482-2485.

[31] V. Rizzoli, A. Costanzo, P. R. Ghigi, F. Mastri, D. Masotti, C. Cecchetti, "Recent advances in harmonic-balance techniques for nonlinear microwave circuit simulation", AEU Arch Elektron Uebertrag Electron Commun, Vol. 46, No. 4 Jul 1992, p 286-297.

[32] M. Celik, A. Atalar, M. A. Tan, "New method for the steady-state analysis of periodically excited nonlinear circuits", IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, Vol. 43, No. 12 Dec 1996, p 964-972.

[33] H. G. Brachtendorf, G. Welsch, R. Laur, "Fast simulation of the steady-state of circuits by the harmonic balance technique", Proceedings IEEE International Symposium on Circuits and Systems, Vol. 2 1995, IEEE, Piscataway, NJ, USA, p 1388-1391.

[34] H. G. Brachtendorf, G. Welsch, R. Laur, "Simulation tool for the analysis and verification of the steady state of circuit designs", International Journal of Circuit Theory and Applications, Vol. 23, No 4 Jul-Aug 1995, p 311-323.

[35] I. Barbancho Perez, I. Molina Fernandez, "Predictor strategies for continuation methods applied to nonlinear circuit analysis", Industrial Applications in Power Systems, Computer Science and Telecommunications Proceedings of the Mediterranean Electrotechnical Conference MELECON, Vol. 3 1996, IEEE, Piscataway, NJ, USA, p 1419-1422.