**SENSITIVITY ANALYSIS OF NONLINEAR RF CIRCUITS USING
PROJECTION BASED TECHNIQUES**

by

Praveen Basty Pai

A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfillment of the requirements
for the degree of
Master of Applied Science

Ottawa-Carleton Institute for Electrical Engineering,
Department of Electronics,
Carleton University

Copyright © 2004 by Praveen Basty Pai

Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-612-97433-2

Our file Notre référence

ISBN: 0-612-97433-2

The author has granted a non-exclusive license allowing the Library and Archives Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Dedication

To my parents

Acknowledgements

I am very grateful to the many people who, through their support, encouragement, and various contributions have made my work on this thesis more enjoyable. First of all I offer my prayers to the Lord. Without his guidance at different stages of my life, I would not have been the person I am today.

I would like to thank my thesis supervisors, Prof. Ram Achar and Prof. Michel Nakhla, for many hours of discussions on harmonic balance. Their insight and technical feedback were invaluable.

I would like to thank Prof. Emad Gad for helping me at every stage of my thesis with helpful suggestions. I also thank him for helping me learn LATEX. I would like to thank Prof. Roni Khazaka for helping me with my work on multi-tone harmonic balance analysis and for providing me with programs related to it.

My colleagues Subendhu Roy, Nigel Rodrigues, Arumika Jerome, Changzhong Chen (Adiu), Wenliang Tseng (Wayne), Dharmendra Saraswat and Gurpreet Singh Shinh made my time in the lab an enjoyable experience. I thank them for the lively discussions we had which helped me relax amidst the long days of work. The staff of the department of electronics were always there to help. Their friendly manner and cheerfulness were greatly appreciated.

I would like to thank all my friends in Ottawa. In particular I would like to thank my friend Harish Gopala for hours of discussions on spirituality. I thank my relatives in Ottawa, the Mallyas, who have become more like family to me.

I leave a special note of appreciation to my parents who, amidst lots of problems, always presented a smiling face and kept me focussed on studies. Their blessings and prayers helped me overcome many hurdles in life to get here.

Contents

List of tables	iv
List of figures	v
1 Introduction	1
1.1 Background and Motivation	1
1.2 Contributions	2
1.3 Organization of the Thesis	3
2 Literature Survey	4
2.1 Formulation of MNA equations for Nonlinear Circuits	5
2.2 Time-Domain Approaches	6
2.2.1 Conventional Transient Analysis	6
2.2.2 Shooting Method	8
2.3 Frequency Domain Approach	10
2.3.1 Harmonic Balance for Nonlinear Circuits	11
2.3.2 Continuation Methods	18
3 Review of Model-Order Reduction	21
3.1 Concept of Model-Order Reduction	22

3.2	Explicit Moment-Matching Techniques	23
3.3	Implicit Moment-Matching Techniques	26
3.3.1	Linear Circuit Reduction	26
3.3.2	Nonlinear Circuit Reduction for Harmonic Balance Equations . . .	29
3.3.3	Multi-Dimension Model-Reduction	37
4	MOR Techniques for Small Scale Sensitivity Analysis	39
4.1	Introduction	39
4.2	Sensitivity Computation using Harmonic Balance	42
4.2.1	Computational Complexity	43
4.3	Proposed Method	45
4.3.1	Computing the Derivatives	47
4.3.2	Computational Complexity	48
4.4	Summary of the Computational Steps	49
4.5	Proof of Preservation of Derivatives	49
4.6	Numerical Examples	53
4.6.1	Example 1	53
4.6.2	Example 2	58
4.7	Conclusion	60
5	MOR Techniques for Large Scale Sensitivity Analysis	61
5.1	Introduction	61
5.2	Proposed Method	63
5.2.1	Obtaining Steady-State Solution of the Original System	63
5.2.2	Tracking Steady-State Solution for Variations in Parameters	66
5.3	Computation of Derivatives	68
5.3.1	Derivatives with respect to Linear Parameters	69

5.3.2	Derivatives with respect to Nonlinear Parameters	72
5.4	Computational Complexity	77
5.5	Proof of Preservation of Derivatives	78
5.5.1	Preservation of Derivatives $\mathbf{A}_\alpha$	79
5.5.2	Preservation of Derivatives $\mathbf{A}_\lambda$	84
5.6	Numerical Examples	87
6	Projection Based Techniques for Multi-Tone Circuits	91
6.1	Introduction	91
6.1.1	Multi-Tone HB equations	92
6.1.2	FFT for Almost Periodic Signals	93
6.2	Numerical Examples	97
6.2.1	Example 1	97
6.2.2	Example 2	102
6.2.3	Example 3	105
7	Summary and Future Work	108
7.1	Future Work	110
	Bibliography	112

List of Tables

3.1	Formulae for the derivatives of some simple functions	33
4.1	Comparing relative sensitivity of first few harmonics w.r.t the junction area of the diode connected to the output node	56
4.2	Computational cost comparison of the proposed method v/s HB	57
4.3	Relative sensitivity of first 3 harmonics w.r.t the junction area of the transistor	58
5.1	$\bar{Y}_1$ when λ is a linear parameter	72
5.2	Error from the proposed method	88
6.1	Amplitude of the harmonics	94
6.2	Mapping of the harmonics	95
6.3	Computational cost comparison of the proposed method v/s HB	101

List of Figures

3.1	Pseudo-code description of block Arnoldi process.	28
3.2	Schematic of Krylov method for nonlinear circuits	29
4.1	Schematic of the proposed sensitivity computation	47
4.2	Summary of the computational steps	50
4.3	A nonlinear circuit	54
4.4	Steady-State response	55
4.5	Third harmonic (which has magnitude = $0.00424V$ at $C = 2nF$) sensitivity variation with respect to the capacitor connected to the output node	55
4.6	A tuned amplifier circuit	59
4.7	Steady-state response at the output node of the tuned amplifier circuit.	59
4.8	Relative error in the sensitivity of the harmonics with respect to the capac- itor C	60
5.1	Summary of the proposed algorithm	68
5.2	A RC coupled amplifier	89
5.3	Tracking third harmonic content for variation in diode saturation current (I_s)	89
5.4	Third harmonic content variation with V_T of the diode	90
5.5	Third harmonic content variation with emitter resistor R	90

6.1	Steps for applying proposed method to multi-tone circuits	96
6.2	A mixer circuit.	98
6.3	Third-order intermodulation variation with transistor junction area	98
6.4	Third-order intermodulation variation with V_T	99
6.5	Third-order intermodulation variation with capacitor C	99
6.6	A LNA circuit.	103
6.7	Third-order intermodulation variation with transistor junction area.	103
6.8	Third-order intermodulation variation with inductor value L_e	104
6.9	A QUAD mixer circuit.	106
6.10	Third-order intermodulation variation with respect to R_E	106
6.11	Third-order intermodulation variation with respect to transistor junction area	107

Notation

$\mathbf{A}$	A real matrix $\mathbf{A}$
$\mathbf{A}^T$	A real matrix $\mathbf{A}$ transposed
$\mathbb{R}^N$	The set of real vectors of size N
$\mathbb{R}^{N \times N}$	The set of real matrices of size $N \times N$
$\mathbb{C}^N$	The set of complex vectors of size N
$\mathbb{C}^{N \times N}$	The set of complex matrices of size $N \times N$

Abbreviations

CAD	Computer Aided Design
CPU	Central Processing Unit
DAE	Differential-Algebraic Equation
DFT	Direct Fourier Transform
EDA	Electronic Design Automation
EMM	Explicit Moment Matching
FFT	Fast Fourier Transform
HB	Harmonic Balance
IDFT	Inverse Direct Fourier Transform
IFFT	Inverse Fast Fourier Transform
IMM	Implicit Moment Matching
LMS	Linear Multi-Step
LTI	Linear Time Invariant
LU	Lower/Upper Triangular Decomposition
LO	Local Oscillator
MMTs	Moment Matching Techniques
MNA	Modified Nodal Admittance
MOR	Model-Order Reduction
NR	Newton Raphson Method
RF	Radio Frequency
RFIC	Radio Frequency Integrated Circuits
RLC	Resistor-Inductor-Capacitor Network
VLSI	Very Large Scale Integrated Circuits

Chapter 1

Introduction

1.1 Background and Motivation

In recent years, computer-aided design (CAD) tools have become an integral part of the design cycle of integrated circuits (ICs) and systems [1]. Deployment of CAD tools in various tasks of IC design has contributed significantly to reduce the time-to-market and cost of developing new systems. However, the rapid development in available technological potentials has resulted in increased complexity of designs, stretching the existing boundaries of modern CAD tools.

One example of increased complexity is in the analog area, where the design of radio frequency integrated circuits (RFIC) has recently gained much prominence. Due to the recent boom in wireless communications, RF circuits such as low noise amplifiers (LNAs), mixers, and power amplifiers, which are typically found in RF front ends, have gained more and more importance. It can be noted that, design/analysis complexity has mainly increased due to three main factors. The first is related to using more complicated circuit topologies, in order to handle more aggressive design criteria such as lower operating voltage, or lower power consumption. Another source of complexity is the drive towards higher integration.

One example of this is, including passive elements such as on chip spiral inductors which warrants complicated models [2], [3]. Finally, another source of increased complexity is due to the active device models. RF circuits typically require accounting for second and higher order effects in device models. Also as device sizes decrease and RF designers explore other technologies such as CMOS [4], complexity of device models is on the rise, thereby adding more internal nodes to the circuit. The increased problem size has pushed the limits of current RF design tools.

Model order reduction techniques offer solutions for ever increasing problem size. Recently, Krylov subspace based reduction techniques [5] were suggested to obtain the steady-state solution of nonlinear circuits. Although algorithms based on model reduction techniques have been developed for fast steady-state analysis of nonlinear circuits, problems associated with sensitivity analysis are still to be addressed.

The objective of this thesis is to address the increasing challenges associated with sensitivity analysis of nonlinear RF circuits, which is an integral part in the optimization of circuit design. The main contributions of this thesis are given below.

1.2 Contributions

1. A new nonlinear circuit reduction algorithm is developed to perform small scale sensitivity analysis of nonlinear RF circuits [6]. Conventionally, sensitivity analysis is performed using steady-state solution and adjoint system solution. For obtaining the steady-state solution, we use Krylov subspace reduction technique [5]. The new algorithm proposes to obtain adjoint solution also, using the Krylov subspace reduction technique. This technique computes sensitivity information without adding any additional significant cost over the cost spent in obtaining steady-state solution using the reduction technique.

2. A new nonlinear circuit reduction algorithm is developed to perform large scale sensitivity analysis of nonlinear RF circuits [7], [8]. Using proposed algorithm, steady-state solution is tracked with respect to the changes in system parameters without repeating the whole procedure of steady-state analysis. Proposed algorithm reduces original system size significantly and hence the computational cost involved for tracking the steady-state solution.

1.3 Organization of the Thesis

The rest of the thesis is organized as follows. Chapter 2 provide a background on the steady-state analysis techniques. Chapter 3 reviews the model order reduction techniques. Chapter 4 and chapter 5, presents the proposed algorithms for small scale sensitivity analysis and large scale sensitivity analysis respectively. Chapter 6 presents the sensitivity analysis of multi-tone circuits using projection based reduction techniques. Finally a summary and a list of future work is given in chapter 7.

Chapter 2

Literature Survey

Steady-state solution of any system is the solution after any start-up transient has died down. It is typically of primary interest to circuit designers since various classic radio frequency (RF) measurements, such as compression point or intermodulation, only make sense in steady state. Hence steady-state analysis becomes crucial requirement in general commercial simulators. The basic idea underlying the concept of steady-state analysis techniques is that the response of a stable circuit due to a forcing periodic input must be periodic [9], [10]. Based on the knowledge that all the waveforms of the circuit variables are periodic, the problem can be formulated as a boundary-value problem [11]. Approaches for finding the steady-state operating point differ in the way this boundary-value problem is formulated. There is considerable literature on the subject of computing the solution to boundary value problem. In general, these approaches are classified based on the domain in which the problem is formulated. Therefore, most of the techniques proposed in the literature may be classified under two categories: time-domain and frequency-domain based approaches. Other approaches that are based on using both domains have been explored more recently with the objective of combining the advantages of both approaches [12], [13], [14], [15].

This chapter begins with formulating modified nodal analysis (MNA) equations for

nonlinear circuits. Section 2.2 describes time-domain approaches. Section 2.3 describes frequency-domain approaches and section 2.3.2 explains the continuation methods.

2.1 Formulation of MNA equations for Nonlinear Circuits

Consider a circuit consisting of linear and nonlinear components. Using a nodal approach, Kirchhoff's current law can be used to describe a given circuit as [16]:

$$\mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{G}\mathbf{x}(t) + \mathbf{f}(\mathbf{x}(t)) = \mathbf{b}(t) \quad (2.1)$$

where

- $\mathbf{x}(t) \in \mathbb{R}^N$ is a vector of node voltage waveforms and currents,
- $\mathbf{C} \in \mathbb{R}^{N \times N}$ and $\mathbf{G} \in \mathbb{R}^{N \times N}$ are matrices describing the linear lumped memory and memoryless elements, respectively,
- $\mathbf{f}(\mathbf{x}(t))$ is a vector of nonlinear functions representing the contribution of nonlinear elements,
- $\mathbf{b}(t)$ is a vector with entries determined by the independent voltage and current sources, and,
- N is the total number of MNA variables

The $\mathbf{G}$ and $\mathbf{C}$ matrices are typically referred to as the MNA matrices. The advantage of using the MNA formulation is that the construction of $\mathbf{G}$ and $\mathbf{C}$ matrices in Equation (2.1) can be easily automated using predefined stamps (or *stencils*) for each component [17].

2.2 Time-Domain Approaches

Time-domain methods tackle the problem of computing steady-state solution as a boundary-value problem in time-domain. The goal of a time-domain based technique, therefore, becomes that of finding the initial condition for the circuit variables such that numerical solution of the MNA differential-algebraic equations (DAE), such as (2.1), at the end of one period of the forcing input matches the initial condition. Such problems where the solution for DAEs satisfy constraints at two or more distinct points in time are referred to as boundary-value problems. Conventional transient analysis and shooting method are discussed as examples of time domain approaches.

2.2.1 Conventional Transient Analysis

Transient analysis is a brute-force way to obtain the steady-state solution by solving the initial value problem over a sufficiently long time interval. In initial-value problems, an initial condition for circuit variables at $t = t_0$ is assumed to be known. The response is obtained by starting with this initial condition and stepping forward in time using numerical integration techniques until all transients have died out [17]. Fast Fourier transform (FFT) [18] is then applied to the signal obtained when the steady-state has been reached in order to obtain the frequency spectrum of the response. Typical integration methods are based on Linear Multi-Step (LMS) methods such as the Trapezoidal and Backward Euler rules [19].

As an example, consider the Backward Euler technique, where the following formula is used to approximate the derivative of circuit variables, $\mathbf{x}(t)$, at $t_1 = t_0 + h$,

$$\left. \frac{d\mathbf{x}(t)}{dt} \right|_{t=t_1} = \frac{\mathbf{x}(t_1) - \mathbf{x}(t_0)}{h} \quad (2.2)$$

Applying (2.2) to the MNA equations of (2.1) yields

$$\left(\mathbf{G} + \frac{1}{h} \mathbf{C} \right) \mathbf{x}(t_1) + \mathbf{f}(\mathbf{x}(t_1)) = \frac{1}{h} \mathbf{C} \mathbf{x}(t_0) + \mathbf{b}(t_1) \quad (2.3)$$

Equation (2.3) can be written in the form

$$\mathbf{A}\mathbf{x}(t_1) + \mathbf{f}(\mathbf{x}(t_1)) = \mathbf{u} \quad (2.4)$$

where

$$\mathbf{A} = \mathbf{G} + \frac{1}{h}\mathbf{C}; \quad \mathbf{u} = \frac{1}{h}\mathbf{C}\mathbf{x}(t_0) + \mathbf{b}(t_1) \quad (2.5)$$

Thus, applying Backward Euler rule to (2.1), converts the set of nonlinear DAEs into a set of nonlinear algebraic equations with $\mathbf{x}(t_1)$ as the set of unknowns. Solving a system of nonlinear equations can be attempted through iterative methods such as Newton-Raphson (NR) method [20]. A typical NR method works iteratively to find the solution of (2.4) as

$$\mathbf{x}^{(i+1)}(t_1) = \mathbf{x}^{(i)}(t_1) - [\mathbf{J}(\mathbf{x}^{(i)}(t_1))]^{-1} \psi(\mathbf{x}^{(i)}(t_1)) \quad (2.6)$$

where $\mathbf{J}(\mathbf{x})$ is the Jacobian matrix of (2.4) and is given by

$$\mathbf{J}(\mathbf{x}) = \mathbf{A} + \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \quad (2.7)$$

and $\psi(\mathbf{x})$ is given by

$$\psi(\mathbf{x}) \equiv \mathbf{A}\mathbf{x} + \mathbf{f}(\mathbf{x}) - \mathbf{u} \quad (2.8)$$

with $\mathbf{x}^{(0)}(t_1)$ as an initial guess solution.

As shown from the above analysis, finding the solution at each time step requires several LU decompositions of the Jacobian matrix in order to solve the linear system of equations in (2.6). This presents a significant CPU cost for large circuits as the size of these matrices is typically in the order of thousands. The CPU cost becomes even more prohibitive in circuits with high quality factor (high Q), where the circuit goes through many cycles before reaching the steady-state.

Among the other disadvantages of this approach, the truncation error (associated with LMS integrations formulae) in each iteration becomes a major source of inaccuracy. Another source of error results from using FFT, where aliasing, interpolation and numerical

noise due to TE degrade the accuracy of the frequency components [21]. Any failure to satisfy the periodicity condition, i.e., $\mathbf{x}(t) = \mathbf{x}(t + T)$, results in some form of noise that corrupts the accuracy of the frequency spectrum [22].

In addition, interpolation errors arise because FFT requires equally spaced time points while simulated data points do not satisfy this property in general. To reduce these FFT errors, tight error tolerance must be imposed on the DAE solver to take smaller time steps which results in a further large number of time steps causing an even higher CPU cost.

2.2.2 Shooting Method

The objective of shooting methods [9], [23], [24] is to accelerate the convergence of the transient time-domain analysis. In other words, shooting methods aim at skipping long transient cycles. The basic idea in the shooting method is that, at steady-state $\mathbf{x}(t) = \mathbf{x}(t + T)$ where T is the period of the forcing input and $\mathbf{x}(t)$ is the vector of waveforms of circuit variables as defined in section 2.1. Thus, if an initial condition vector $\mathbf{x}(0)$ can be found such that

$$\mathbf{x}(T) - \mathbf{x}(0) = \mathbf{0} \quad (2.9)$$

then the periodic solution can be reached in only one cycle of numerical integration. The question now becomes how to find such an initial condition that satisfies Equation (2.9). The condition set in (2.9) is a typical boundary-value problem and can be solved via the state-transition function [25] $\Phi(\mathbf{x}(t), \tau)$, which is defined by

$$\Phi(\mathbf{x}(t), \tau) = \mathbf{x}(t + \tau) \quad (2.10)$$

In other words, a state transition function is a function that produces the state-variables of the system as an output using the value of the state variables at a different time point as an

input and the difference in time between the corresponding two points. Thus,

$$\mathbf{x}(T) = \Phi(\mathbf{x}_0, T) \quad (2.11)$$

where $\mathbf{x}_0 = \mathbf{x}(0)$. Substituting from (2.11) in (2.9), we obtain

$$\Phi(\mathbf{x}_0, T) - \mathbf{x}_0 = \mathbf{0} \quad (2.12)$$

which is a set of nonlinear equations in the initial condition $\mathbf{x}_0$. Thus, solving for the initial conditions ($\mathbf{x}_0$) can be done using iterative methods such as the NR method [20]. This method is typically referred to as Newton-Shooting technique [9], [10]. It is worth mentioning that other approaches based on gradient and extrapolation techniques have been proposed in the literature [24], [23]. This section describes the solution based on NR method as an example. NR starts with an initial guess for $\mathbf{x}_0$, say $\mathbf{x}_0^{(0)}$, and then proceeds with the following iteration scheme

$$\mathbf{x}_0^{(i+1)} = \mathbf{x}_0^{(i)} - [\mathbf{J}_s(\mathbf{x}_0^{(i)})]^{-1} \Psi(\mathbf{x}_0^{(i)}) \quad (2.13)$$

where $\Psi(\mathbf{x}_0^{(i)})$ is the nonlinear function defined by

$$\Psi(\mathbf{x}_0^{(i)}) \equiv \Phi(\mathbf{x}_0^{(i)}, T) - \mathbf{x}_0^{(i)} = \mathbf{0}, \quad (2.14)$$

$\mathbf{J}_s(\mathbf{x}_0^{(i)})$ is the Jacobian matrix of the shooting equations defined by

$$\mathbf{J}_s(\mathbf{x}_0^{(i)}) = \left. \frac{\partial \Psi(\mathbf{x}_0)}{\partial \mathbf{x}_0} \right|_{\mathbf{x}_0 = \mathbf{x}_0^{(i)}} = \left. \frac{\partial \Phi(\mathbf{x}_0, T)}{\partial \mathbf{x}_0} \right|_{\mathbf{x}_0 = \mathbf{x}_0^{(i)}} - \mathbf{U} \quad (2.15)$$

$\mathbf{U}$ is an identity matrix and $\frac{\partial \Phi(\mathbf{x}_0, T)}{\partial \mathbf{x}_0}$ is the sensitivity matrix of the state transition w.r.t. the initial condition computed at $\mathbf{x}_0 = \mathbf{x}_0^{(i)}$.

Computing the sensitivity matrix requires working with the first variational equation which is derived by differentiating the MNA equations in (2.1) w.r.t. $\mathbf{x}_0$ to obtain

$$\mathbf{G} \frac{\partial \mathbf{x}(t)}{\partial \mathbf{x}_0} + \mathbf{C} \frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}_0} + \frac{\partial \mathbf{f}(\mathbf{x}(t))}{\partial \mathbf{x}(t)} \frac{\partial \mathbf{x}(t)}{\partial \mathbf{x}_0} = \mathbf{0} \quad (2.16)$$

or

$$\mathbf{G}\mathbf{z}(t) + \mathbf{C}\dot{\mathbf{z}}(t) + \mathbf{A}(t)\mathbf{z}(t) = 0 \quad (2.17)$$

where

$$\mathbf{z}(t) = \frac{\partial \mathbf{x}(t)}{\partial \mathbf{x}_0}; \quad \mathbf{A}(t) = \frac{\partial \mathbf{f}(\mathbf{x}(t))}{\partial \mathbf{x}(t)} \quad (2.18)$$

Note that $\mathbf{z}(t) \in \mathbb{R}^{N \times N}$ is a matrix with N representing the total number of variables in the MNA formulation. Using (2.11) we get

$$\mathbf{z}(t) = \frac{\partial \mathbf{x}(t)}{\partial \mathbf{x}_0} = \frac{\partial \Phi(\mathbf{x}_0, t)}{\partial \mathbf{x}_0} \quad (2.19)$$

or

$$\frac{\partial \Phi(\mathbf{x}_0, T)}{\partial \mathbf{x}_0} = \mathbf{z}(T) \quad (2.20)$$

and

$$\mathbf{z}(0) = \frac{\partial \mathbf{x}(0)}{\partial \mathbf{x}_0} = \frac{\partial \mathbf{x}(0)}{\partial \mathbf{x}(0)} = \mathbf{U} \quad (2.21)$$

The desired sensitivity matrix can, therefore, be computed by integrating the first variational equation in (2.17) starting from $t = 0$ up to $t = T$. Since $\mathbf{z}(t)$ is a matrix, this process has to be repeated N times for each column of the matrix $\mathbf{z}(t)$.

The major computational cost of the Newton-Shooting method comes from having to solve a system of linear equations at each iteration of the NR method in (2.13), with the matrix $\mathbf{J}_s$ as the coefficients matrix. This matrix is typically full and the CPU cost of solving this problem is $O(N^3)$.

2.3 Frequency Domain Approach

The Harmonic Balance (HB) [26], [27], [28], [29], [30] approach is a technique for finding the steady-state solution in the frequency-domain. Significant advantage of Harmonic Balance approach over the time domain techniques comes from the fact that, HB computes the

steady-state solution of the nonlinear circuits directly in frequency domain avoiding any transients that render the time domain techniques time consuming. Another feature of this technique is that the differential algebraic equation describing the nonlinear circuit as in (2.1) is converted to nonlinear algebraic equations which can be solved directly for steady state solution using Newton Raphson (NR) technique.

2.3.1 Harmonic Balance for Nonlinear Circuits

The main philosophy behind HB formulation is that, waveforms in a periodically excited nonlinear circuit are periodic and, can be represented in the frequency domain as a finite Fourier series. Consider a nonlinear circuit described by the MNA equations as in (2.1). For periodic inputs, $\mathbf{b}(t)$ and $\mathbf{x}(t)$ are periodic in the steady state, and hence can be represented as finite Fourier series, i.e.

$$\mathbf{b}(t) = \sum_{k=-K}^K \mathbf{B}_k e^{jk\omega t}, \quad \mathbf{x}(t) = \sum_{k=-K}^K \mathbf{X}_k e^{jk\omega t} \quad (2.22)$$

where $\mathbf{B}_k$ and $\mathbf{X}_k$ represent harmonics of $\mathbf{b}(t)$ and $\mathbf{x}(t)$, respectively and K is the number of harmonics used in Fourier representation of waveforms.

Consequently, $\mathbf{f}(\mathbf{x}(t))$ is also periodic and can be expressed as

$$\mathbf{f}(\mathbf{x}(t)) = \sum_{k=-K}^K \mathbf{F}_k e^{jk\omega t} \quad (2.23)$$

where $\mathbf{F}_k$ is a vector containing the k^{th} harmonic of each component in the vector $\mathbf{f}(\mathbf{x}(t))$. Substituting from (2.22) and (2.23) into (2.1), and making use of the orthogonality of the Fourier series results in,

$$\bar{\mathbf{G}}\mathbf{X} + j\Omega\bar{\mathbf{C}}\mathbf{X} + \mathbf{F}(\mathbf{X}) = \mathbf{B} \quad (2.24)$$

where

$$\bar{\mathbf{G}} = \begin{bmatrix} \mathbf{G} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \mathbf{G} & \\ & & & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{G} \end{bmatrix}; \quad \bar{\mathbf{C}} = \begin{bmatrix} \mathbf{C} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \mathbf{C} & \\ & & & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{C} \end{bmatrix} \quad (2.25)$$

$$\mathbf{\Omega} = \begin{bmatrix} -K\omega\mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \mathbf{0} & \\ & & & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & K\omega\mathbf{I} \end{bmatrix} \quad (2.26)$$

$$\mathbf{X} = \left[\mathbf{X}_{-K}^T \quad \cdots \quad \mathbf{X}_0^T \quad \cdots \quad \mathbf{X}_K^T \right]^T \quad (2.27)$$

$$\mathbf{B} = \left[\mathbf{B}_{-K}^T \quad \cdots \quad \mathbf{B}_0^T \quad \cdots \quad \mathbf{B}_K^T \right]^T \quad (2.28)$$

Here $\bar{\mathbf{G}}$, $\bar{\mathbf{C}}$, and $\mathbf{\Omega} \in \mathbb{R}^{N_{hb}}$ are the matrices describing the linear part of the network and

$$N_{hb} = NN_h; \quad N_h = 2K + 1 \quad (2.29)$$

The vector $\mathbf{F}(\mathbf{X})$ contains the Fourier coefficients of the nonlinear vector $\mathbf{f}(\mathbf{x}(t))$ defined in (2.1). More precisely, the vector $\mathbf{f}(\mathbf{x}(t))$ is the vector of nonlinear scalar functions

$$\mathbf{f}(\mathbf{x}(t)) = [f_1(\mathbf{x}(t)) \quad f_2(\mathbf{x}(t)) \quad \cdots \quad f_N(\mathbf{x}(t))]^T \quad (2.30)$$

and the vector $\mathbf{F}(\mathbf{X})$ is given by

$$\mathbf{F}(\mathbf{X}) = [F_1(\mathbf{X}) \quad F_2(\mathbf{X}) \quad \cdots \quad F_N(\mathbf{X})]^T \quad (2.31)$$

where $F_1(\mathbf{X})$ is a vector containing the Fourier coefficients of $f_1(\mathbf{x}(t))$, $F_2(\mathbf{X})$ contains the coefficients of $f_2(\mathbf{x}(t))$ etc.

An important observation to be made is that a system of size N represented as nonlinear DAE (2.1) is converted to a system of nonlinear algebraic equations (2.24) of size N_{hb} . Hence the steady-state solution can be obtained by solving the set of nonlinear algebraic equations (2.24) instead of having to deal with the original set of nonlinear DAE system (2.1). This is explained next.

Solving the Harmonic Balance Equations

Being a set of nonlinear equations (containing the harmonics of circuit variables as the unknowns), HB equations can be solved using iterative methods such as NR method. NR method proceeds by starting with an initial guess $\mathbf{X}^{(0)}$ and iterates through

$$\mathbf{X}^{(j+1)} = \mathbf{X}^{(j)} - [\mathbf{J}(\mathbf{X}^{(j)})]^{-1} \Phi_h(\mathbf{X}^{(j)}) \quad (2.32)$$

where

$$\Phi_h(\mathbf{X}) \equiv \bar{\mathbf{G}}\mathbf{X} + j\Omega\bar{\mathbf{C}}\mathbf{X} + \mathbf{F}(\mathbf{X}) - \mathbf{B} \quad (2.33)$$

and $\mathbf{J}(\mathbf{X}) \in \mathbb{C}^{N_{hb} \times N_{hb}}$ is the Jacobian matrix obtained using the following equation

$$\mathbf{J}(\mathbf{X}) \equiv \frac{\partial \Phi_h(\mathbf{X})}{\partial \mathbf{X}} = \bar{\mathbf{G}} + j\Omega\bar{\mathbf{C}} + \frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}} \quad (2.34)$$

The rest of this section presents further details regarding the evaluation of the different terms required in Equations (2.32)-(2.34) to perform the NR iterations.

Direct Fourier Transform

The direct Fourier transform (DFT) is the basic computational component common in computing all the terms needed in Equations (2.32)-(2.34). FFT enables efficient evaluation of DFT when the number of time points in the waveform is a power of 2 [18]. Efficiency of

the FFT method stems from the recursive structures that can be exploited to avoid matrix-vector products.

Although FFT and DFT are mathematically equivalent, the DFT approach has been selected for presentation in this section due to its simplicity. In addition, using DFT provides a mathematical concise formulation for explaining the evaluation of the Jacobian matrix $\mathbf{J}(\mathbf{X})$ and the nonlinear function vector $\mathbf{F}(\mathbf{X})$.

Let $\mathbf{x}(t)$ be a periodic signal with a period $T = \frac{2\pi}{\omega}$, where $\mathbf{x}(t)$ can be expressed as a Fourier series

$$\mathbf{x}(t) = \sum_{k=-K}^K X_k e^{jk\omega t} \quad (2.35)$$

Let $\mathbf{x}(t)$ be sampled at N_h equally spaced time points $[t_0, t_1, \dots, t_{N_h-1}]$ in the interval $[0, T]$ such that

$$t_n = n \frac{T}{N_h}, \quad n = 0, 1, \dots, N_h - 1 \quad (2.36)$$

where N_h is given by (2.29). Writing the Fourier series in (2.35) at each time point yields

$$\mathbf{x}(t_0) = \sum_{k=-K}^K X_k e^{jk\omega t_0} \quad (2.37a)$$

$$\mathbf{x}(t_1) = \sum_{k=-K}^K X_k e^{jk\omega t_1} \quad (2.37b)$$

$$\vdots$$

$$\mathbf{x}(t_{N_h-1}) = \sum_{k=-K}^K X_k e^{jk\omega t_{N_h-1}} \quad (2.37c)$$

Equation (2.37) can therefore be assembled in a matrix form as follows

$$\begin{bmatrix} x_0 \\ \vdots \\ x_{N_h-1} \end{bmatrix} = \begin{bmatrix} e^{-jK\omega t_0} & \dots & 1 & \dots & e^{jK\omega t_0} \\ e^{-jK\omega t_1} & \dots & 1 & \dots & e^{jK\omega t_1} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ e^{-jK\omega t_{N_h-2}} & \dots & 1 & \dots & e^{jK\omega t_{N_h-2}} \\ e^{-jK\omega t_{N_h-1}} & \dots & 1 & \dots & e^{jK\omega t_{N_h-1}} \end{bmatrix} \begin{bmatrix} X_{-K} \\ \vdots \\ X_0 \\ \vdots \\ X_K \end{bmatrix} \quad (2.38)$$

where $x_i = x(t_i)$ represents the time sample of $x(t)$ at time t_i . Equation (2.38) represents the inverse direct Fourier transform (IDFT) and can be put in a more compact representation as

$$\mathbf{X}_t = \Gamma^{-1} \mathbf{X} \quad (2.39)$$

where $\mathbf{X}_t = [x_0 \ x_1 \ \cdots \ x_{N_h-1}]$ represents the time samples, $\mathbf{X} = [X_{-K} \ \cdots \ X_K]$ represents the harmonics, and Γ^{-1} is the IDFT operator.

Note that Γ^{-1} is independent of the frequency ω since the argument of the exponential function is independent of ω as can be shown from

$$k\omega t_i = k\left(\frac{2\pi}{T}\right)\left(i\left(\frac{T}{N_h}\right)\right) = ki\left(\frac{2\pi}{N_h}\right) \quad (2.40)$$

Thus, the only condition for Γ^{-1} to be independent of the frequency is that the time samples $[x_0 \ x_1 \ \cdots \ x_{N_h-1}]$ be equally spaced over the period $[0, T]$. The DFT operator is thus given by

$$\mathbf{X} = \Gamma \mathbf{X}_t \quad (2.41)$$

The main computational complexity of DFT (and IDFT) is equivalent to the cost associated with the classical matrix-vector multiplication, i.e., $O(N_h^2)$. On the other hand, FFT (and IFFT) achieves exactly the same result but with computational complexity of $O(N_h \log_2(N_h))$ via making use of the structure of Γ .

Computation of $F(\mathbf{X})$:

In this section, the evaluation of the nonlinear vector $F(\mathbf{X})$ using direct Fourier transform (DFT) is explained. Without loss of generality, the discussion in this section is limited to one of the scalar functions in (2.30). The same procedure is applied to each scalar element of (2.30) in order to obtain (2.31). Our task is to evaluate the vector $F_1(\mathbf{X})$ containing the Fourier coefficients of $f_1(x(t))$, as a function of $\mathbf{X}$. Without loss of generality, for the

purpose of simplicity, we can assume that $f_1(\mathbf{x}(t))$ is a function of one variable x_1 . The objective is therefore to find $F_1(X_1)$, as a function of X_1 which is the vector containing the Fourier coefficients of $x_1(t)$. Knowing X_1 , the time samples of $x_1(t)$ over one period can be obtained by applying the IDFT in equation (2.39)

$$X_t = \Gamma^{-1} X_1 \quad (2.42)$$

where $X_t = [x_0 \ x_1 \ \cdots \ x_{N_h-1}]$ is a vector containing the time samples of $x_1(t)$. Next, a vector, F_t , is constructed by sampling $f_1(x(t))$ at the time points $\{t_0, t_1, \dots, t_{N_h-1}\}$, such that

$$\mathbf{F}_t = [f(x_0) \ f(x_1) \ \cdots \ f(x_{N_h-1})]^T \quad (2.43)$$

Finally, computing $F_1(X_1)$ can then be carried out using F_t and the DFT operator as follows

$$F_1(X_1) = \Gamma F_t \quad (2.44)$$

Computation of the Matrix $\partial F(\mathbf{X})/\partial \mathbf{X}$:

The evaluation and inversion of the harmonic balance Jacobian represents the bulk of the computational complexity of finding the steady-state response. This section is devoted to explaining the structure of this jacobian and how it is evaluated. The HB Jacobian was defined previously as,

$$\mathbf{J}(\mathbf{X}) \equiv \frac{\partial \Phi_h(\mathbf{X})}{\partial \mathbf{X}} = \bar{\mathbf{G}} + j\Omega \bar{\mathbf{C}} + \frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}} \quad (2.45)$$

The first two terms $\bar{\mathbf{G}}$ and $\bar{\mathbf{C}}$ are defined in equation (2.25). The evaluation of the remaining term $\partial \mathbf{F}(\mathbf{X})/\partial \mathbf{X}$ is the subject of this section. In order to find $\partial \mathbf{F}(\mathbf{X})/\partial \mathbf{X}$, the derivative

of the nonlinear vector $\mathbf{f}$ are first considered in the time domain

$$\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_N}{\partial x_1} & \cdots & \frac{\partial f_N}{\partial x_N} \end{bmatrix} \quad (2.46)$$

where $f_1, \dots, f_N$ are the scalar nonlinear functions that form the vector $\mathbf{f}(\mathbf{x})$, and $x_1, \dots, x_N$ are the scalar variables in the vector $\mathbf{x}(t)$. Now we can write $\partial \mathbf{F}(\mathbf{X})/\partial \mathbf{X}$ as

$$\frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}} = \begin{bmatrix} \frac{\partial F_1}{\partial X_1} & \cdots & \frac{\partial F_1}{\partial X_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_N}{\partial X_1} & \cdots & \frac{\partial F_N}{\partial X_N} \end{bmatrix} \quad (2.47)$$

where F_1 , for example, is a vector containing the Fourier coefficient of f_1 , and X_1 contains the Fourier coefficients of x_1 . Note that the terms in (2.47) ($\frac{\partial F_1}{\partial X_1}$ for example) are matrices. Each block is computed from the corresponding term in (2.46). In order to simplify the notation, we will consider the case of one of the blocks, and the same approach is used to compute the remaining blocks. Consider the function $\frac{\partial f_1}{\partial x_1}$. Let X_t be the vector containing the time-domain samples of $x_1(t)$

$$X_t = \begin{bmatrix} x_1(t_0) \\ x_1(t_1) \\ \vdots \\ x_1(t_{N_h-1}) \end{bmatrix} \quad (2.48)$$

From (2.39), we can write

$$X_t = \Gamma^{-1} X_1 \quad (2.49)$$

The time samples of $f_1(x_1(t))$ are assembled in the vector F_t such that

$$F_t = \begin{bmatrix} f_1(x_1(t_0)) \\ f_1(x_1(t_1)) \\ \vdots \\ f_1(x_1(t_{N_h-1})) \end{bmatrix} \quad (2.50)$$

From (2.44), the following expression can be written

$$F_1 = \Gamma F_t \quad (2.51)$$

Now we can evaluate the derivative of the Fourier coefficients of $f_1(x_1)$ with respect to the Fourier coefficients of x_1

$$\frac{\partial F_1}{\partial X_1} = \Gamma \frac{\partial F_t}{\partial X_1} = \Gamma \frac{\partial F_t}{\partial X_t} \frac{\partial X_t}{\partial X_1} = \Gamma \frac{\partial F_t}{\partial X_t} \Gamma^{-1} \quad (2.52)$$

where $\frac{\partial F_t}{\partial X_t}$ is a diagonal matrix given by

$$\frac{\partial F_t}{\partial X_t} = \begin{bmatrix} \left. \frac{\partial f_1(x_1)}{\partial x_1} \right|_{x_1(t_0)} & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \left. \frac{\partial f_1(x_1)}{\partial x_1} \right|_{x_1(t_{N_h-1})} \end{bmatrix} \quad (2.53)$$

Each block in equation (2.46) is found by using (2.52). Note that multiplication by Γ and Γ^{-1} is not done in practice, and the DFT and IDFT algorithms are used instead. The resulting Jacobian contains many dense blocks and is very expensive to store, and factorize.

2.3.2 Continuation Methods

As explained in previous section, iterative methods such as the Newton iteration are used to solve the harmonic balance equations. Such methods start with an initial guess in order

to start the iteration. A good initial guess would result in a significantly fewer iterations before convergence is achieved. But Newton's methods for example is not globally convergent (i.e. convergence is not guaranteed for all initial guesses). A good initial guess is often necessary to achieve convergence. The difficulty with this problem is that, typically, there are no obvious ways of finding a good initial guess. Continuation methods (also known as *Homotopy* methods) have been introduced to overcome the convergence problems associated with iterative techniques such as the NR method. The basic idea of the continuation methods is to augment the system $\Phi_h(\mathbf{X})$ in (2.33) with some parameter, say α , to obtain another auxiliary system $\Psi_h(\mathbf{X}, \alpha)$. Here α is chosen such that the system given by $\Psi_h(\mathbf{X}, 0)$ has a trivial solution in $\mathbf{X}$ and the system given at $\Psi_h(\mathbf{X}, 1) = \Phi_h(\mathbf{X})$ identically in $\mathbf{X}$. Continuation methods can be broadly classified into two categories based on the technique of embedding the continuation parameter, α , in $\Phi_h(\mathbf{X})$.

External Parameter Embedding

This approach of homotopy is based on using the parameter α externally to $\Phi_h(\mathbf{X})$. An example of this approach is obtained through formulating $\Psi_h(\mathbf{X})$ as

$$\Psi_h(\mathbf{X}) \equiv (1 - \alpha)(\mathbf{X} - \mathbf{X}_a) + \alpha\Phi_h(\mathbf{X}) \quad (2.54)$$

It is clear from (2.54) that at $\alpha = 0$, $\Psi_h(\mathbf{X})$ has a trivial solution given by

$$\mathbf{X} = \mathbf{X}_a \quad (2.55)$$

and at $\alpha = 1$

$$\Psi_h(\mathbf{X}) = \Phi_h(\mathbf{X}) \quad (2.56)$$

Note that in (2.54) the embedding parameter α is external to $\Phi_h(\mathbf{X})$.

Internal Parameter Embedding

This approach is also known as natural embedding. Here, the parameter α is embedded internally in $\Phi_h(\mathbf{X})$, whose solution is required. The choice of the parameter α in the natural embedding approach usually corresponds to some physical quantity in the system. Sweeping α in this case represents deforming the original system into a simple system with a trivial solution at $\alpha = 0$.

For the case of HB nonlinear equations, a natural choice for the parameter α would be the power of the time-varying forcing input. This can be achieved by splitting the harmonics source vector, $\mathbf{B}$, into two vectors and writing down the auxiliary system $\Psi_h(\mathbf{X}, \alpha)$ as

$$\Psi_h(\mathbf{X}, \alpha) = \bar{\mathbf{G}}\mathbf{X} + j\Omega\bar{\mathbf{C}}\mathbf{X} + \mathbf{F}(\mathbf{X}) - \mathbf{B}_{dc} - \alpha\mathbf{B}_{ac} \quad (2.57)$$

Here, $\mathbf{B}_{ac}$ is a vector representing the time-varying forcing input circuit and $\mathbf{B}_{dc}$ represents the biasing dc sources. At $\alpha = 0$, the solution of (2.57) is simply the solution obtained from the usual DC analysis, while at $\alpha = 1$ the system $\Psi_h(\mathbf{X}, 1)$ has the same solution as $\Phi_h(\mathbf{X})$.

A typical continuation method proceeds by taking small increments in α and solving the system $\Psi_h(\mathbf{X}, \alpha)$ for many values of α until $\alpha = 1$.

Difficulties of Continuation Methods

Although continuation methods have provided a major advantage in being able to eliminate convergence problems, they are computationally expensive as they require the solution of many auxiliary systems of equations corresponding to numerous values of $\alpha \in (0, 1)$. This problem is even further aggravated for the case of HB equations since the HB Jacobian matrix is typically huge/coupled and its factorization overwhelms even medium sized computers.

Chapter 3

Review of Model-Order Reduction

The rapid growth in RF silicon integrated circuits (ICs) for mobile communication systems has placed new demands on simulation tools. Ever increasing size of circuits due to large lumped models, from packaging and interconnect structures, has resulted in large matrices requiring huge memory space for storage. Thus, current simulation tools are constantly being challenged for handling large matrices, and has resulted in increased simulation time. During the recent years model-reduction techniques have been proposed as a way to deal with problems occurring due to large circuit sizes. There are various methods available for model-order reduction, in literature. They can be broadly classified into two categories: 1) approaches based on explicitly matching the moments¹ to a reduced-order model and 2) approaches based on implicitly matching the moments.

This chapter begins with explaining the concept of model-reduction in section 3.1. Section 3.2 explains the concept of Padé approximation, as an example for explicit moment-matching technique (EMM). Section 3.3 explains the concept of Krylov subspace technique, as an example for implicit moment-matching technique (IMM). Section 3.3.3 presents the idea of reduction techniques in multi-dimension.

¹Taylor series coefficients of the system transfer function with respect to a system parameter

3.1 Concept of Model-Order Reduction

The basic concept of model-order reduction is that, while the lumped models contain a large number of poles (an infinite number of poles in the case of networks with distributed components), only a few of these poles, the dominant poles², have a significant impact on the circuit response in the frequency range of interest. Model-reduction techniques take advantage of this fact by approximating the original large system with a reduced-order model that captures the essential characteristics, or dominant poles of the original system. A brief mathematical description of the associated concept is given below. Consider a single input/output system. Let $H(s)$ be its transfer function, which can be represented in a rational form as,

$$H(s) = \frac{P(s)}{Q(s)} \quad (3.1)$$

where $P(s)$ and $Q(s)$ are polynomials in s . Equivalently, (3.1) can be written as

$$H(s) = c + \sum_{i=1}^{N_p} \frac{k_i}{s - p_i} \quad (3.2)$$

where p_i and k_i are the i^{th} pole-residue pair, N_p is the total number of system poles, and c is the direct coupling constant. The time-domain impulse response can be computed in a closed form using inverse Laplace transform as

$$h(t) = c\delta t + \sum_{i=1}^{N_p} k_i e^{p_i t} \quad (3.3)$$

For large networks N_p will be of order of thousands. Computing all the N_p poles will be highly CPU intensive even for a small network and completely impractical for large networks. Model-reduction techniques address the above issue by deriving a reduced-order approximation $\hat{H}(s)$ in terms of dominant poles, instead of trying to compute all the poles

²poles that are close to the imaginary axis and significantly influence the time as well as the frequency characteristics of the system

of a system. Assuming that only L dominant poles were extracted, (3.1) can be rewritten to obtain approximate frequency/time responses, as

$$H(s) \approx \hat{H}(s) = \frac{\hat{P}(s)}{\hat{Q}(s)} = \hat{c} + \sum_{i=1}^L \frac{\hat{k}_i}{s - \hat{p}_i} \quad (3.4)$$

$$h(t) \approx \hat{h}(t) = \hat{c}\delta t + \sum_{i=1}^L \hat{k}_i e^{\hat{p}_i t} \quad (3.5)$$

3.2 Explicit Moment-Matching Techniques

Explicit moment-matching techniques [31], [32], [33], [34] work by constructing a direct Padé approximate for the impulse response of the network in the frequency-domain. To illustrate this concept, consider a single input/output linear network with a frequency response $H(s) \in \mathbb{C}$. Using a Taylor's expansion around some point in the frequency domain [32] s_0 , $H(s)$ can be expressed as,

$$H(s) = m_0 + m_1(s - s_0) + m_2(s - s_0)^2 + \dots \quad (3.6)$$

where $m_i, i = \{0, 1, \dots\}$, is the i^{th} moment (or scaled derivative) of $H(s)$ w.r.t. s at $s = s_0$.

EMM techniques use the idea of Padé approximant to construct a reduced-order description for $H(s)$. Padé approximants are basically rational function approximations of the form $q_L(s)/q_M(s)$, where $q_L(s)$ and $q_M(s)$ are two polynomials of order L and M , respectively. To find such an approximation, we write $H(s)$ as

$$\begin{aligned} H(s) &= \frac{a_0 + a_1(s - s_0) + \dots + a_L(s - s_0)^L}{1 + b_1(s - s_0) + \dots + b_M(s - s_0)^M} \\ &= m_0 + m_1(s - s_0) + m_2(s - s_0)^2 + \dots \end{aligned} \quad (3.7)$$

Cross-multiplying (3.7) and equating similar powers of s , coefficients a_i and b_i are obtained

as

$$\begin{bmatrix} m_{L-M+1} & m_{L-M+2} & \cdots & m_L \\ m_{L-M+2} & \ddots & \ddots & m_{L+1} \\ \vdots & \ddots & \ddots & \vdots \\ m_L & m_{L-M+1} & \cdots & m_{L+M-1} \end{bmatrix} \begin{bmatrix} b_M \\ b_{M-1} \\ \vdots \\ b_1 \end{bmatrix} = - \begin{bmatrix} m_{L+1} \\ m_{L+2} \\ \vdots \\ m_{L+M} \end{bmatrix} \quad (3.8)$$

$$\begin{aligned} a_0 &= m_0 \\ a_1 &= m_1 + b_1 m_0 \\ a_2 &= m_2 + b_1 m_1 + b_2 m_0 \\ &\vdots \\ a_L &= m_L + \sum_{i=1}^{\min(L,M)} b_i m_{L-i} \end{aligned} \quad (3.9)$$

Thus, obtaining Padé-approximant is a simple procedure once the moments of the transfer function are obtained. The moments can be computed at a low CPU cost, and it can be shown that it is equivalent to finding the DC operating point of the network [32]. To explain how to obtain the moments, consider a linear multiport network described by the MNA equations as shown below

$$\mathbf{G}\mathbf{x}(t) + \mathbf{C}\dot{\mathbf{x}}(t) = \mathbf{B}\mathbf{u}(t), \quad \mathbf{i}(t) = \mathbf{B}^T \mathbf{x}(t) \quad (3.10)$$

where $\mathbf{u}(t)$ and $\mathbf{i}(t) \in \mathbb{R}_p^N$ represent the input and output of the networks. $\mathbf{G}$ and $\mathbf{C} \in \mathbb{R}^{N \times N}$ represent the memoryless and memory matrices in the network as explained in section 2.1. $\mathbf{B} \in \mathbb{R}^{N \times N_p}$ is a selector matrix that maps the input ports to the rest of the network nodes, N_p is the number of ports and N is the number of MNA variables. Equation (3.10) can be written in the frequency-domain as

$$(\mathbf{I}_N - s\mathbf{A})\mathbf{X}(s) = \mathbf{R}\mathbf{U}(s), \quad \mathbf{I}(s) = \mathbf{B}^T \mathbf{X}(s) \quad (3.11)$$

where

$$\mathbf{A} = -\mathbf{G}^{-1}\mathbf{C} \quad \text{and} \quad \mathbf{R} = \mathbf{G}^{-1}\mathbf{B} \quad (3.12)$$

To obtain the impulse response, we substitute for $U(s)$ with the Laplace-domain function, i.e. $U(s) = 1$ [35]. Hence, the Laplace-domain impulse response can be written as

$$\begin{aligned} H(s) &= (I_N - sA)^{-1}R \\ &= M_0 + M_1s + M_2s^2 + \dots \end{aligned} \quad (3.13)$$

where $M_0, M_1, \dots, M_i, \dots$ are the block moments of $H(s)$ w.r.t. s . From (3.13) it can be shown that

$$\begin{aligned} M_0 &= R \\ M_i &= AM_{i-1} = A^i M_0, \quad i > 0 \end{aligned} \quad (3.14)$$

EMM techniques generally face two main difficulties. The first difficulty is numerical and is directly related to the explicit computation of the moments in (3.14). Generally the matrix in (3.8) becomes ill-conditioned for higher order approximations (typically above 10). This means that practically no further information can be obtained from using higher order moments in the Padé approximant. This problem has been addressed by complex frequency hopping (CFH) technique. The CFH algorithm in [31] makes use of multiple expansion points in order to construct a unified rational approximation that extends the matching range in the frequency-domain beyond what can be obtained through a single expansion point. The second difficulty is passivity. A macromodel should be passive, i.e. a network cannot generate more energy than it absorbs, and no passive termination of the network will cause the system to go unstable. Unfortunately, in EMM techniques, there is no absolute guarantee that the resulting reduced-order model is passive.

3.3 Implicit Moment-Matching Techniques

The Krylov subspace technique was introduced by Nikolai Krylov in 1931, and later studied and developed by many others such as Arnoldi [36] and Lanczos [37]. Recently, these methods were introduced to circuit simulation as a way to perform model-order reduction on high-speed interconnect networks [38], [39], [40]. Krylov subspace methods are based on the computation of the leading eigenvalues (those with largest magnitude). Also, these methods are well behaved as they employ numerically well conditioned algorithms such as Arnoldi process [41]. An important feature of Krylov subspace techniques is that they guarantee the passivity of the macromodel by construction [40], [42]. Krylov subspace techniques for linear and nonlinear circuits are presented next.

3.3.1 Linear Circuit Reduction

Consider the linear lumped multiport network of (3.10) described in the frequency domain

$$GX + sCX = B \quad (3.15)$$

or

$$(I - sA) X(s) = R \quad (3.16)$$

where

$$A = -G^{-1}C \quad R = G^{-1}B \quad (3.17)$$

The main goal of a typical Krylov-subspace procedure is to approximate the matrix A in (3.16) with a reduced-order matrix that preserves its leading eigenvalues, i.e. those eigenvalues with the largest magnitude. Matching the moments of the transfer function comes as a by-product of this approximation step. The reduced-order matrix preserving the

leading eigenvalues can be obtained using simple similarity transformation as follows:

$$\mathbf{A}\mathbf{K} = \mathbf{K}\mathbf{H}_q \quad (3.18)$$

where $\mathbf{H}_q \in \mathbb{R}^{q \times q}$ is the upper-Hessenberg matrix in its companion form, and $\mathbf{K} \in \mathbb{R}^{N \times k}$ is the transformation matrix defined as

$$\mathbf{K} = \begin{bmatrix} \mathbf{R} & \mathbf{A}\mathbf{R} & \mathbf{A}^2\mathbf{R} & \dots & \mathbf{A}^{q-1}\mathbf{R} \end{bmatrix} \quad (3.19)$$

with $k = qN_p$ (N_p is the number of ports).

The space spanned by the columns of the matrix $\mathbf{K}$, i.e. the space spanned by $\mathbf{A}^i\mathbf{R}$, the moments, is the Krylov subspace. Computation of $\mathbf{H}_q$ using the relation (3.18) ($\mathbf{H}_q = \mathbf{K}^{-1}\mathbf{A}\mathbf{K}$) requires the inverse of the matrix $\mathbf{K}$. However, $\mathbf{K}$ is a dense matrix, and hence computation of its inverse will be expensive. Also, $\mathbf{K}$ is likely to be ill-conditioned since the columns of $\mathbf{K}$ are formed based on the sequence $\mathbf{A}^i\mathbf{R}$, which quickly converges to the eigenvector corresponding to the largest eigenvalue. Hence it is difficult to explicitly construct a high order Krylov subspace. A well known solution to this problem is the Arnoldi process [41], which is used to construct an orthonormal basis $\mathbf{Q} \in \mathbb{R}^{N \times k}$ of the Krylov subspace

$$\text{colsp}[\mathbf{Q}] = \text{colsp}[\mathbf{K}] \quad (3.20)$$

The Arnoldi process can construct an orthonormal basis for a very high order Krylov subspace, without explicitly calculating the moments, and without ill-conditioning. Arnoldi algorithm is explained briefly in the pseudo-code procedure of Figure 3.1.

The circuit reduction is done by congruent transformation using the orthonormal basis $\mathbf{Q}$, by the following change of variables

$$\mathbf{X} = \mathbf{Q}\hat{\mathbf{X}} \quad (3.21)$$

Substituting (3.21) in (3.15) and multiplying both sides with $\mathbf{Q}^T$, we obtain the reduced

Algorithm 3.3.1: ARNOLDI METHOD(G, C, R, q)

```

 $[q_0, \mathcal{R}] \leftarrow qr(R)$   comment:  $qr$  factorization of  $R$ 

for  $j \leftarrow 1$  to  $q$ 
     $z \leftarrow G^{-1}Cq_{j-1}$ 
    for  $i \leftarrow 0$  to  $j - 1$ 
        do  $\left\{ \begin{array}{l} h \leftarrow q_i^T z \\ z \leftarrow z - q_i h \end{array} \right.$ 
     $[q_j, \mathcal{R}] \leftarrow qr(z)$   comment:  $qr$  factorization of  $z$ 

 $Q \leftarrow [q_0 \quad q_1 \quad \cdots \quad q_q]$ 
return ( $Q$ )

```

Figure 3.1: Pseudo-code description of block Arnoldi process.

system given as

$$\hat{G}\hat{X} + s\hat{C}\hat{X} = \hat{B}$$

or

$$(\hat{G} + s\hat{C})\hat{X}(s) = \hat{B} \quad (3.22)$$

where $\hat{G} \in \mathbb{R}^{k \times k}$, $\hat{C} \in \mathbb{R}^{k \times k}$, $\hat{B} \in \mathbb{R}^{k \times N_p}$ and $\hat{X} \in \mathbb{R}^k$ are given by

$$\begin{aligned} \hat{G} &= Q^T G Q \quad ; \quad \hat{C} = Q^T C Q \\ \hat{B} &= Q^T B \quad ; \quad \hat{X} = Q^T X \end{aligned} \quad (3.23)$$

Note that the order of the reduced system is much smaller than that of the original system. Model-order reduction using Krylov subspace methods has two main advantages. First, the reduced model can be guaranteed passive by construction [40], [42]. The second advantage is that large number of poles up to a very high frequencies can be captured from a single expansion. However, while Krylov methods can capture all the dominant pole,

the resulting reduced-order models contain many redundant poles making them unnecessarily large compared to multi-point direct expansions such as complex frequency hopping (CFH) [43], [44].

3.3.2 Nonlinear Circuit Reduction for Harmonic Balance Equations

Krylov subspace technique for nonlinear circuits is explained using the algorithm in [5]. An overview of the algorithm is shown in Figure 3.2. This algorithm combines Krylov subspace technique with the continuation method. The algorithm first embeds a continuation parameter α in the HB equations (2.33), $\phi(\mathbf{X}) = 0$, to obtain (2.57). The circuit is then reduced with respect to the continuation parameter α . The reduced system is then solved, sweeping α from 0 to 1, and the solution at $\alpha = 1$ is mapped back to the original system solution. Main advantage of the algorithm is significant reduction in system size without sacrificing the accuracy of the obtained solution. Since analysis is done entirely in the small space, it results in computational cost saving. Details of steps in Figure 3.2 is presented next.

Figure 3.2: Schematic of Krylov method for nonlinear circuits

Moment Computation

In this section, the second step in Figure 3.2, i.e. computing moments, is explained. The Krylov subspace $\mathcal{Q}_q$ is defined as

$$\text{ColumnSpan}\{\mathcal{Q}_q\} = \text{ColumnSpan}\{ \mathbf{A}_0 \ \mathbf{A}_1 \ \cdots \ \mathbf{A}_{q-1} \} \quad (3.24)$$

where $\mathbf{A}_0, \dots, \mathbf{A}_{q-1}$ are the Taylors series coefficients of the system solution with respect to the continuation parameter α , also called the moments of the system with respect to α . In order to obtain $\mathcal{Q}_q$, the moments $\mathbf{A}_0, \dots, \mathbf{A}_{q-1}$ are evaluated, and a standard orthonormalization method [45] is applied in order to obtain an orthonormal basis $\mathcal{Q}_q$, which spans the space of these moments. The focus of this section is on the evaluation of the moments $\mathbf{A}_k$, or the Taylor coefficients of $\mathbf{X}$ with respect to the continuation parameter α , i.e.

$$\mathbf{X}(\alpha) = \sum_{k=0}^{N_q-1} \mathbf{A}_k \alpha^k \quad (3.25)$$

The nonlinear vector $\mathbf{F}(\mathbf{X})$ can also be expanded in a Taylor series as follows

$$\mathbf{F}(\mathbf{X}(\alpha)) = \sum_{k=0}^{N_q-1} \mathbf{D}_k \alpha^k \quad (3.26)$$

Clearly, at $\alpha = 0$, we have $\mathbf{F}(\mathbf{X}(0)) = \mathbf{D}_0 = \mathbf{F}(\mathbf{A}_0)$. Substituting from (3.25) and (3.26) into (2.57), we get,

$$\bar{\mathbf{G}} \sum_{k=0}^{N_q-1} \mathbf{A}_k \alpha^k + j\Omega \bar{\mathbf{C}} \sum_{k=0}^{N_q-1} \mathbf{A}_k \alpha^k + \sum_{k=0}^{N_q-1} \mathbf{D}_k \alpha^k = \mathbf{B}_{dc} + \alpha \mathbf{B}_{ac} \quad (3.27)$$

Setting $\alpha = 0$ we get

$$\bar{\mathbf{G}} \mathbf{A}_0 + j\Omega \bar{\mathbf{C}} \mathbf{A}_0 + \mathbf{F}(\mathbf{A}_0) = \mathbf{B}_{dc} \quad (3.28)$$

Note that $\mathbf{A}_0$ in (3.28) is simply the DC solution. Also, equating the first power of α in (3.27) yields

$$\bar{\mathbf{G}} \mathbf{A}_1 + j\Omega \bar{\mathbf{C}} \mathbf{A}_1 + \mathbf{D}_1 = \mathbf{B}_{ac} \quad (3.29)$$

Now define the matrix $\mathbf{T}(\alpha)$ to be

$$\mathbf{T}(\alpha) = \frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}} \quad (3.30)$$

and let $\mathbf{T}(\alpha)$ have the Taylor series expansion

$$\mathbf{T}(\alpha) = \sum_{k=0} \mathbf{T}_k \alpha^k \quad (3.31)$$

Since $\mathbf{D}_1 = (\partial \mathbf{F} / \partial \alpha)|_{\alpha=0} = (\partial \mathbf{F} / \partial \mathbf{X}) \mathbf{A}_1$, then substituting in (3.29) gives the following relation for $\mathbf{A}_1$

$$(\bar{\mathbf{G}} + j\Omega\bar{\mathbf{C}} + \mathbf{T}_0)\mathbf{A}_1 = \mathbf{B}_{ac} \quad (3.32)$$

The first moment is therefore found by solving the above set of linear equations. Note that the left hand side matrix is the Jacobian of the harmonic balance equations. CPU cost of this operation is one LU decomposition of the Jacobian matrix. In order to compute $\mathbf{A}_2, \dots, \mathbf{A}_q$, the Taylor series expansion of $\mathbf{T}(\alpha)$ is used in the following relation

$$\frac{\partial \mathbf{F}}{\partial \alpha} = \mathbf{T} \frac{\partial \mathbf{X}}{\partial \alpha} \quad (3.33)$$

to get

$$\sum_{i=1} i \mathbf{D}_i \alpha^{i-1} = \sum_{i=0} \mathbf{T}_i \alpha^i \sum_{i=1} i \mathbf{A}_i \alpha^{i-1} \quad (3.34)$$

Taking the n^{th} derivative in (3.34) with respect to α and putting $\alpha = 0$, we get the following recursive relationship

$$\mathbf{D}_n = \mathbf{T}_0 \mathbf{A}_n + \frac{1}{n} \sum_{j=1}^{n-1} (n-j) \mathbf{T}_j \mathbf{A}_{n-j} \quad (3.35)$$

Equating the n^{th} power of α in (3.27) we get

$$\bar{\mathbf{G}} \mathbf{A}_n + j\Omega\bar{\mathbf{C}} \mathbf{A}_n + \mathbf{D}_n = 0 \quad (3.36)$$

Now using equation (3.35), the following recursive relationship for the derivatives $\mathbf{A}_n$ is obtained

$$\mathbf{J}_o \mathbf{A}_n = -\frac{1}{n} \sum_{j=1}^{n-1} (n-j) \mathbf{T}_j \mathbf{A}_{n-j} \quad (3.37)$$

where

$$\mathbf{J}_o = (\bar{\mathbf{G}} + j\Omega\bar{\mathbf{C}} + \mathbf{T}_0) \quad (3.38)$$

It is clear from (3.32) and (3.37) that, computing $\mathbf{A}_1, \dots, \mathbf{A}_q$ requires only one LU decomposition of the Jacobian matrix. Each additional derivative can be obtained by forward/backward substitution. Next we proceed to show how to compute the right hand side of equation (3.37). The term $\mathbf{A}_{n-j}$ represents the $(n-j)^{th}$ derivative of $\mathbf{X}$ with respect to α , with $\mathbf{A}_0$ and $\mathbf{A}_{n-1}$ evaluated using (3.32) and (3.37). The higher order derivatives are found recursively using (3.37). The remaining term required in (3.37) is $\mathbf{T}_j$.

For the sake of simplicity, we consider a circuit with nonlinearity appearing at only one node. Let this nonlinearity be due to a nonlinear resistor, hence we can write $i(x) = f(x)$. In this case $\mathbf{T}$ can be expressed as

$$\mathbf{T} = \Gamma \mathbf{G} \Gamma^{-1}$$

with

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}(\mathbf{x}(t_1, \alpha)) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \\ & & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{g}(\mathbf{x}(t_s, \alpha)) \end{bmatrix}; \quad \text{where } \mathbf{g}(\mathbf{x}) = \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}}$$

where $t_1 t_2 \cdots t_s$ are time points over the fundamental period. We can write the expression for $\mathbf{T}_i$ as

$$\mathbf{T}_i = \Gamma \mathbf{G}_i \Gamma^{-1}$$

where $\mathbf{G}_i$ is a diagonal vector whose elements are the time samples of $\mathbf{g}_i$. Problem of computing $\mathbf{g}_i$ based on the knowledge of $\mathbf{A}_0, \dots, \mathbf{A}_{i-1}$ has been addressed in [46]. As an example for this computation, consider the case where the nonlinear device is represented

Table 3.1: Formulas for the derivatives of some simple functions [46]

Equation	$y = \sum_{i=0} a_i \alpha^i, z = \sum_{i=0} c_i \alpha^i, f = \sum_{i=0} d_i \alpha^i$
$f = \exp(y)$	$d_0 = \exp(a_0)$ $d_n = (1/n) \sum_{i=0}^{n-1} d_i a_{n-i} (n-i)$
$f = \log(y)$	$d_0 = \log(a_0)$ $d_n = (1/a_0) \left(a_n - \sum_{i=1}^{n-1} a_i d_{n-i} \frac{(n-i)}{n} \right)$
$f = y^p$	$d_0 = a_0^p$ $d_n = (1/n a_0) \left(n p d_0 a_n + \sum_{i=1}^{n-1} d_i a_{n-i} (p n - i(p+1)) \right)$
$f = y + z$	$d_n = a_n + c_n$
$f = y - z$	$d_n = a_n - c_n$
$f = yz$	$d_n = \sum_{i=0}^n a_i c_{n-i}$
$f = y/z$	$d_n = (1/c_0) \left(a_n - \sum_{i=0}^{n-1} a_i c_{n-i} \right)$

by a diode:

$$f(x) = I_s(e^{x/V_t} - 1); \quad g(x) = \frac{I_s}{V_t} e^{x/V_t} = \sum_{i=0} g_i \alpha^i \quad (3.39)$$

Let x be represented by the Taylor's series

$$x = \sum_{i=0} a_i \alpha^i \quad (3.40)$$

It can be shown that

$$g_n = \frac{I_s}{n V_t} x = \sum_{i=0}^{n-1} g_i a^{n-i} (n-i) \quad (3.41)$$

Similar expressions can be found for other functions as shown in Table 3.1.

Constructing the reduced system

In this section, the third step in Figure 3.2 (circuit reduction), is explained. Reduction is done by effecting the following change of variables in (2.57),

$$\mathbf{X} \leftarrow \mathbf{Q}_q \hat{\mathbf{X}} \quad (3.42)$$

where $\mathbf{Q}_q \in \mathbb{R}^{N_{hb} \times q}$ is an orthogonal basis for the first q Taylor series coefficients (or scaled derivatives) of $\mathbf{X}$ with respect to α given as in (3.24). With the change of variables in (3.42), the augmented harmonic balance equations in (2.57) become

$$\bar{\mathbf{G}}\mathbf{Q}_q\hat{\mathbf{X}} + j\Omega\bar{\mathbf{C}}\mathbf{Q}_q\hat{\mathbf{X}} + \mathbf{F}(\mathbf{Q}_q\hat{\mathbf{X}}) = \mathbf{B}_{dc} + \alpha\mathbf{B}_{ac} \quad (3.43)$$

Equation (3.43) is then pre-multiplied by $\mathbf{Q}_q^T$, and the following reduced system is obtained

$$\hat{\mathbf{G}}\hat{\mathbf{X}} + \hat{\mathbf{C}}\hat{\mathbf{X}} + \hat{\mathbf{F}}(\hat{\mathbf{X}}) - \hat{\mathbf{B}}_{dc} - \alpha\hat{\mathbf{B}}_{ac} = 0 \quad (3.44)$$

where

$$\begin{aligned} \hat{\mathbf{G}} &= \mathbf{Q}_q^T \bar{\mathbf{G}} \mathbf{Q}_q \\ \hat{\mathbf{C}} &= \mathbf{Q}_q^T j\Omega \bar{\mathbf{C}} \mathbf{Q}_q \\ \hat{\mathbf{F}}(\hat{\mathbf{X}}) &= \mathbf{Q}_q^T \mathbf{F}(\mathbf{Q}_q \hat{\mathbf{X}}) \\ \hat{\mathbf{B}}_{dc} &= \mathbf{Q}_q^T \mathbf{B}_{dc} \end{aligned} \quad (3.45)$$

Basic idea behind this change of variables is that the reduced system obtained by this transformation preserves the first q derivatives of $\mathbf{X}$ with respect to α under the mapping $\mathbf{Q}_q$. More precisely, if $\hat{\mathbf{X}}$ is expanded as the Taylor series around $\alpha = 0$

$$\hat{\mathbf{X}} = \sum_{k=0}^{\infty} \hat{\mathbf{A}}_k \alpha^k \quad (3.46)$$

then

$$\mathbf{A}_k = \mathbf{Q}_q \hat{\mathbf{A}}_k, \quad k = 0, \dots, q-1 \quad (3.47)$$

Proof of conservation of the derivatives can be found in [5]. The solution of the reduced system is therefore expected to accurately approximate that of the original system as α is tracked from 0 to 1.

Solving the reduced system

After the augmented system of equations is reduced (3.44), next step is to solve it (the fourth step in Figure 3.2). Let us denote this system of equations by $\hat{\Psi}(\hat{\mathbf{X}}, \alpha)$, i.e.

$$\hat{\Psi}(\hat{\mathbf{X}}, \alpha) = \hat{\mathbf{G}}\hat{\mathbf{X}} + \hat{\mathbf{C}}\hat{\mathbf{X}} + \hat{\mathbf{F}}(\hat{\mathbf{X}}) - \hat{\mathbf{B}}_{dc} - \alpha\hat{\mathbf{B}}_{ac}$$

In order to simplify the notation, above system will be denoted as $\hat{\Psi}_i(\hat{\mathbf{X}}) = \hat{\Psi}(\hat{\mathbf{X}}, \alpha_i)$ when $\alpha = \alpha_i$. Solution of the reduced system of equations is found by applying the NR iterations to

$$\hat{\Psi}_i(\hat{\mathbf{X}}) = \hat{\mathbf{G}}\hat{\mathbf{X}} + \hat{\mathbf{C}}\hat{\mathbf{X}} + \hat{\mathbf{F}}(\hat{\mathbf{X}}) - \hat{\mathbf{B}}_t \quad (3.48)$$

with

$$\hat{\mathbf{B}}_t = \hat{\mathbf{B}}_{dc} - \alpha_i\hat{\mathbf{B}}_{ac} \quad (3.49)$$

Starting with an initial guess $\hat{\mathbf{X}}_{guess}$, the solution is updated, at each Newton iteration as follows,

$$\hat{\mathbf{X}}_{new} = \hat{\mathbf{X}}_{old} - \hat{\mathbf{J}}(\hat{\mathbf{X}}_{old})^{(-1)}\hat{\Psi}_i(\hat{\mathbf{X}}_{old}) \quad (3.50)$$

where $\hat{\mathbf{X}}_{old}$ is the old guess, $\hat{\mathbf{X}}_{new}$ is the new updated solution, and $\hat{\mathbf{J}}(\hat{\mathbf{X}}_{old})$ is Jacobian of the reduced equations, expressed as

$$\begin{aligned}
 \hat{\mathbf{J}}(\hat{\mathbf{X}}_{old}) &= \left. \frac{\partial \hat{\Psi}_i(\hat{\mathbf{X}})}{\partial \hat{\mathbf{X}}} \right|_{\hat{\mathbf{X}}_{old}} \\
 &= \hat{\mathbf{G}} + \hat{\mathbf{C}} + \left. \frac{\partial \hat{\mathbf{F}}(\hat{\mathbf{X}})}{\partial \hat{\mathbf{X}}} \right|_{\hat{\mathbf{X}}_{old}} \\
 &= \hat{\mathbf{G}} + \hat{\mathbf{C}} + \left. \frac{\partial \left(\mathbf{Q}_q^T \mathbf{F}(\mathbf{Q}_q \hat{\mathbf{X}}) \right)}{\partial \hat{\mathbf{X}}} \right|_{\hat{\mathbf{X}}_{old}} \\
 &= \hat{\mathbf{G}} + \hat{\mathbf{C}} + \mathbf{Q}_q^T \left(\left. \frac{\partial \mathbf{F}(\mathbf{X})}{\partial \mathbf{X}} \right|_{\mathbf{Q}_q \hat{\mathbf{X}}_{old}} \right) \mathbf{Q}_q
 \end{aligned} \tag{3.51}$$

Jacobian of the reduced circuit, $\hat{\mathbf{J}} \in \mathbb{R}^{q \times q}$, is therefore related to the Jacobian of the original system (2.34), by a congruent transformation with the matrix $\mathbf{Q}_q$

$$\hat{\mathbf{J}} = \mathbf{Q}_q \mathbf{J} \mathbf{Q}_q \tag{3.52}$$

and is much smaller in size than the original Jacobian.

The iteration in (3.50) is repeated until the difference $\|\Delta \hat{\mathbf{X}}\|$ between the old solution $\hat{\mathbf{X}}_{old}$ and the updated value is less than the error tolerance ϵ . Analysis is therefore done completely in the domain of the reduced system. Once convergence is achieved the solution is mapped back to the original system space

$$\mathbf{Q}_q \hat{\mathbf{X}} \rightarrow \mathbf{X} \tag{3.53}$$

An estimate of the error can be obtained by substituting the solution $\mathbf{X}$ into the harmonic balance equations in (2.57), and checking if Kirchhoff's current law is satisfied. If the solution is found to be unsatisfactory, order of the reduced system can be increased or an expansion at an intermediate value of α can be performed.

Computational Complexity

In this section computational complexity of the above algorithm is discussed. Computational cost is spent on its two main steps. First, for calculating the moments and second, constructing and solving the reduced system. We note the following observations

- Computational cost in constructing the orthogonal basis comes from factorizing the Jacobian matrix J_o used in (3.37) requiring N_{hb}^β ($\beta \approx 1.2 - 1.5$) CPU operations and q F/B substitutions with different right hand side vectors. It can be shown that the Jacobian matrix J_o is sparse with block diagonal structure, and hence its L/U factors can be obtained by calculating the L/U factors of its each diagonal blocks.
- Significant CPU cost in this step comes from constructing the reduced Jacobian matrix, since Jacobian being a dense matrix its multiplication with Q_q for reduction is expensive. CPU cost involved in solving the reduced system is negligible since its size is $q \ll N_{hb}$.

3.3.3 Multi-Dimension Model-Reduction

In the previous section, the circuit was reduced with respect to one variable α . When the system needs to be reduced with respect to more than one variable, multi-dimension reduction algorithm based on congruent transformation [47] can be used. The algorithm first starts by computing the multi-dimension moments of the system and uses them to form a subspace. If $M_1, \dots, M_n$ are the moments of the system with respect to n different variables, then using multi-dimension reduction technique, a subspace is formed:

$$K = [M_1, \dots, M_n] \quad (3.54)$$

Now K is generally ill-conditioned, hence it is transformed into a new matrix Q through orthonormalization [40], [18]. Once the congruent transformation matrix Q is obtained,

original large circuit is reduced and solved similar to single-dimension reduction technique.

In the subsequent chapters application of model-order reduction techniques for fast sensitivity analysis of RF circuits is presented.

Chapter 4

MOR Techniques for Small Scale Sensitivity Analysis

4.1 Introduction

The term *sensitivity computation*, in the context of circuit simulation, specifically refers to the task of computing derivatives of circuit variables with respect to an arbitrary physical, electrical or manufacturing process parameter. Circuit variables generally correspond to a wide range of performance criteria such as node voltages, branch currents, consumed power, or capacitor charges. It can also imply transient or steady-state time-dependent waveforms of voltages, currents, etc. Sensitivity information is curial for many tasks. For example, sensitivity information provides the gradient vector required in performing automatic optimization of circuit performance. In addition, sensitivity information is needed in many other different situations such as estimating parasitic effects, desensitizing a given design tolerance assignment, fault and worst case analysis.

Sensitivity analysis is performed using the well-known adjoint technique [17]. Also, sensitivity analysis is tightly coupled to steady-state analysis. That is because, steady-state

solution has to be found first and then its sensitivity with respect to design parameters. Once the steady-state is obtained, adjoint technique obtains the corresponding adjoint solution with an incremental cost of one forward/backward substitution. This feature of the adjoint technique has made it very popular for computing sensitivity.

The steady-state solution can be obtained by performing time-domain integration until the transients die out. However, finding the steady-state response for large analog and microwave nonlinear circuits represents an increasing challenge, since these types of circuits exhibit characteristics that make most traditional transient based time-domain approaches perform poorly. For instance, if there are widely separated time constants in the circuit, then the usual transient analysis can take many clock cycles before the steady-state solution is reached. Another traditional approach specifically proposed for obtaining the steady-state response of nonlinear circuits is the harmonic balance (HB) technique [26], [30]. Chapter 2 reviewed these two approaches. In particular, the HB approach has been the focus of increasing interest over the past decade due to its ability to handle nonlinear elements in the frequency-domain. In addition, the HB approach can be extended easily to simulate circuits excited by non-commensurate multi-tone signals.

The main philosophy behind HB is that, waveforms in a periodically excited nonlinear circuit are periodic and thus can be represented in the frequency-domain as a finite Fourier series. This effectively transforms the differential equations describing the circuit into a set of nonlinear algebraic equations that can be solved directly using some iterative techniques such as the Newton Raphson (NR) method [20]. However, the main bottleneck that comes with solving this set of nonlinear equations is the prohibitive CPU cost that is required to store and factorize the Jacobian matrix of the associated nonlinear equations. Furthermore, for large RF integrated circuits with strong nonlinearities this Jacobian matrix becomes large and dense and its factorization can be expensive even for medium sized circuits [48]. In addition, like all other locally convergent methods, NR has convergence problems. Con-

vergence can be achieved only by finding initial values sufficiently close to the solution, which is difficult particularly for the cases of the high excitation levels.

Continuation methods [49] have been proposed to avoid the convergence problems associated with the direct NR method. In the continuation schemes, the original set of nonlinear equations is replaced with an auxiliary system of nonlinear equations of the same size whose solution is trivial. Then using some sweeping parameter, the solution trajectory can be traced back to the original system. However, these methods are computationally expensive as they require the factorization of the Jacobian matrix corresponding to many values of the sweeping parameter.

Model reduction techniques provide solutions for problems occurring due to large circuit size. Chapter 3 reviewed some of the reduction techniques. An algorithm based on Krylov subspace technique was proposed for obtaining steady-state solution in an efficient way. This technique was explained in section 3.3.2. Even though the problems associated with steady-state analysis was solved using Krylov subspace reduction methods, sensitivity analysis using adjoint technique still faced a big problem. The problem is explained in the next paragraph.

Once steady-state solution is computed, adjoint technique requires to solve the adjoint system for sensitivity computation. For solving adjoint system it requires LU factors of the Jacobian matrix, which are already available if the steady-state solution was obtained by solving the original large system of equations. On the other hand, if Krylov techniques are used for obtaining the steady-state solution (which eliminates the necessity of factorizing the Jacobian), adjoint technique has to factorize the Jacobian to solve the adjoint system. This factorization is very CPU expensive, making the sensitivity analysis computationally expensive process.

Proposed method offers an efficient way of computing the sensitivity, and is based on Krylov subspace reduction technique [6]. Steady-state solution is first obtained using

reduction technique explained in section 3.3.2. Proposed method then obtains adjoint solution based on Krylov subspace technique. First, it projects the original adjoint system onto a subspace of much smaller size. The reduced adjoint system is then solved and the solution is mapped back to the original adjoint system. Sensitivity analysis is now performed by substituting the steady-state solution and adjoint solution, both obtained using Krylov subspace reduction techniques, in the adjoint equation. Proposed method significantly reduces the computational cost involved for sensitivity analysis. This chapter primarily focusses on obtaining adjoint solution using reduction techniques.

This chapter is organized as follows. Section 4.2 describes computing the sensitivity using the HB technique. The proposed method is explained in section 4.3. Section 4.4 summarizes the steps involved in performing sensitivity analysis based on the proposed method. Proof for preservation of derivatives in the reduced system is presented in section 4.5. Numerical examples are presented in section 4.6.

4.2 Sensitivity Computation using Harmonic Balance

Nonlinear circuits (2.1) are formulated using HB equations as

$$\bar{G}X + j\Omega\bar{C}X + F(X) = B \quad (4.1)$$

Here we consider the task of computing the sensitivity of the steady-state solution with respect to any arbitrary parameter λ . To this end, we assume that X_c is the steady-state solution of (4.1). Differentiating both sides of (4.1) at $X = X_c$ with respect to λ , we obtain

$$J_c \frac{\partial X}{\partial \lambda} = -\frac{\partial \bar{Y}}{\partial \lambda} X_c - \frac{\partial F}{\partial \lambda} \Big|_{X=X_c} \quad (4.2)$$

where J_c is the Jacobian matrix computed at convergence of (2.32), i.e.,

$$J_c = J(X_c) = \bar{G} + j\Omega\bar{C} + \frac{\partial F}{\partial X} \Big|_{X=X_c} \quad (4.3)$$

and $\bar{\mathbf{Y}} = \bar{\mathbf{G}} + j\Omega\bar{\mathbf{C}}$. Solving the system in (4.2), therefore, results in finding the sensitivity of all Fourier series coefficients (or harmonics for short) for all the circuit variables with respect to the parameter λ . However it is often the case that designers are interested in computing the sensitivity of certain harmonics at few circuit node voltages or currents. Hence, if we denote the sensitivity of the k^{th} harmonic for the m^{th} waveform by $\mathcal{S}_{(m,k)}^\lambda$, then we can write,

$$\mathcal{S}_{(m,k)}^\lambda = \mathbf{d}^T \frac{\partial \mathbf{X}}{\partial \lambda} \quad (4.4)$$

where $\mathbf{d}$ is a selector vector. Substituting from (4.2) into (4.4), we obtain

$$\mathcal{S}_{(m,k)}^\lambda = -\mathbf{X}_a \left[\frac{\partial \bar{\mathbf{Y}}}{\partial \lambda} \mathbf{X}_c + \frac{\partial \mathbf{F}}{\partial \lambda} \Big|_{\mathbf{X}=\mathbf{X}_c} \right] \quad (4.5)$$

where $\mathbf{X}_a$ is the solution of the adjoint system, given by

$$\mathbf{X}_a = \left[(\mathbf{J}_c)^T \right]^{-1} \mathbf{d} \quad (4.6)$$

It is to be noted here that the Jacobian matrix in (4.6) is the same matrix obtained at the convergence of the NR iterations in (2.32). Hence, if LU factorization has been used to solve the linear system in (2.32), then the LU components of $\mathbf{J}_c$ will be available. Therefore, computing $\mathbf{X}_a$ can be obtained by simple Forward/Backward substitution using the vector $\mathbf{d}$, without incurring further significant computational cost. This technique is called the adjoint technique for sensitivity computation.

4.2.1 Computational Complexity

Although HB represented a significant shift in finding the steady-state solution and its sensitivity, it remained a challenging computational problem in real practical circuits for the following reasons

1. Iterative techniques such as NR method are characterized by strong local convergence. This means that the convergence can not be guaranteed, unless the initial guess point $\mathbf{X}^{(0)}$ is chosen sufficiently close to the solution point, $\mathbf{X}_c$.
2. For circuits with strong nonlinearity, the system (4.1) becomes strongly coupled. This fact usually leads to a dense Jacobian matrix whose LU factorization consumes lot of CPU time and requires huge memory storage to keep pace with the fill-ins [25].
3. For large integrated circuits, the size of the system in (4.1) becomes intractably huge when there is more than one non-commensurate tones (tones which are not multiple of each other) in the input signal. For example, for a small circuit consisting of 100 nodes with 16 harmonics for each nodal variable, the circuit size in case of single-tone input is $(2 \times 16 + 1) \times 100 = 3300$ and with two-tones (non-commensurate) the circuit size grows to $[16 \times (16 + 1)] \times 100 = 27200$.

It is needless to mention that the above difficulties in computing the steady-state solution also complicate the task of sensitivity computation. This is due to the fact that computing sensitivity requires obtaining steady-state solution in the very first step as can be observed from the right-hand side in (4.5).

For tackling convergence problem associated with the NR technique, continuation methods can be used. The necessity of factorizing the Jacobian matrix was a main problem until an algorithm based on Krylov subspace technique was proposed for computing steady-state solution. This algorithm obtains the steady-state solution bypassing the necessity of factorizing the large dense Jacobian matrix $\mathbf{J}_c$. The steady-state solution thus obtained can be directly substituted in the right-hand side of the sensitivity expression in (4.5). Unfortunately, we still require one LU decomposition of the matrix $\mathbf{J}_c$ and one Forward/Backward substitution with the vector $\mathbf{d}$ in order to obtain $\mathbf{X}_a$ from (4.6). Proposed algorithm focusses on obtaining $\mathbf{X}_a$ without having to factorize large dense Jacobian matrix $\mathbf{J}_c$, there

by significantly speeding up the sensitivity computation process.

4.3 Proposed Method

As a first step, the proposed method computes steady-state solution using Krylov subspace technique explained in section 3.3.2. Proposed method then proceeds to obtain the adjoint system solution, based on Krylov subspace reduction technique. The method is explained in the remainder of this section.

Note that the required $\mathbf{X}_a$ in computing the sensitivity can be obtained from (4.6), where

$$\mathbf{J}_c^T = \bar{\mathbf{Y}}^T + \left(\frac{\partial \mathbf{F}}{\partial \mathbf{X}} \right)^T \bigg|_{\mathbf{X}=\mathbf{X}_c} \quad (4.7)$$

is the Jacobian matrix obtained upon convergence of the NR iterations (2.32). We also note that $\mathbf{X}_c = \mathbf{X}(\alpha = 1)$, i.e., the solution that would have been obtained if the Krylov subspace technique described in section 3.3.2 has been applied. Noting the dependence of $\mathbf{X}_a$ and $\mathbf{J}$ on the continuation parameter α , (4.6) can be written as:

$$[\mathbf{J}(\alpha)]^T \mathbf{X}_a(\alpha) = \mathbf{d} \quad (4.8)$$

where it is implicitly understood that $\mathbf{X}_a$ required in computing the sensitivity is obtained from

$$\mathbf{X}_a = \mathbf{X}_a(1) \quad (4.9)$$

with $\mathbf{J}(\alpha = 1) = \mathbf{J}_c$. Equation (4.8) explicitly states that the Jacobian matrix, $\mathbf{J}$, is dependent on α if $\mathbf{X}$ is obtained as a solution trajectory in the continuation parameter α . This argument suggests that computing $\mathbf{X}_a$ can be approached as a continuation problem, where the objective then becomes tracing $\mathbf{X}_a(\alpha)$, $\alpha : 0 \rightarrow 1$ assuming that $\mathbf{X}_a(0)$ is easy to obtain. The proposed sensitivity computation algorithm uses Krylov subspace reduction

technique to trace the solution $\mathbf{X}_a(\alpha)$ in (4.8), but in a subspace with much smaller dimension. This idea is presented next.

We first expand $\mathbf{X}_a(\alpha)$ in a Taylor series with respect to α ,

$$\mathbf{X}_a(\alpha) = \sum_{k=0}^{\infty} \mathbf{U}_k \alpha^k \quad (4.10)$$

then construct an orthogonal basis $\mathbf{P}_q$ for the first q Taylor series coefficients,

$$\text{ColumnSpan}\{\mathbf{P}_q\} = \text{ColumnSpan}\{\mathbf{U}_0, \dots, \mathbf{U}_{q-1}\} \quad (4.11)$$

Krylov subspace reduction is performed, by change of variables

$$\mathbf{X}_a(\alpha) \leftarrow \mathbf{P}_q \check{\mathbf{X}}_a(\alpha) \quad (4.12)$$

and a premultiplication by $\mathbf{P}_q^T$ on both sides of (4.8) to obtain the reduced system,

$$\check{\mathbf{J}}(\alpha) \check{\mathbf{X}}(\alpha) = \check{\mathbf{d}} \quad (4.13)$$

where $\check{\mathbf{J}}(\alpha) \in \mathbb{R}^{q \times q} = \mathbf{P}_q^T \mathbf{J}(\alpha) \mathbf{P}_q$ and $\check{\mathbf{d}} = \mathbf{P}_q^T \mathbf{d}$ with $q \ll N_{hb}$.

The reduced system obtained from change of variables (4.12), preserves the first q derivatives of $\mathbf{X}_a$ with respect to α under the mapping $\mathbf{P}_q$. Proof of preservation of the derivatives is given in section 4.5. More precisely, if $\check{\mathbf{X}}_a$ is expanded as the Taylor series

$$\check{\mathbf{X}}_a = \sum_{k=0}^{\infty} \hat{\mathbf{U}}_k \alpha^k \quad (4.14)$$

then

$$\mathbf{U}_k = \mathbf{P}_q \hat{\mathbf{U}}_k, \quad k = 0, \dots, q-1 \quad (4.15)$$

The solution of the reduced system is therefore expected to accurately approximate that of the original system as α is tracked from 0 to 1.

Finally the reduced system (4.13) is solved directly by sweeping α from 0 to 1 and the solution of the reduced system is mapped back to the original system solution $\mathbf{X}_a$, i.e.

$$\mathbf{X}_a \approx \mathbf{P}_q \check{\mathbf{X}}_a(\alpha = 1) \quad (4.16)$$

Figure 4.1 gives schematic of steps involved in the proposed method. Now the adjoint solution obtained can be substituted in the right hand side of (4.5) to calculate the sensitivity, where $\mathbf{X}_c$ is the steady-state solution which is already computed.

Figure 4.1: Schematic of the proposed sensitivity computation

4.3.1 Computing the Derivatives

For computing the Taylor series coefficients $\mathbf{U}_i$ to form an orthogonal basis $\mathbf{P}_q$ (4.11), we start by substituting the Taylor series expansions of $\mathbf{X}_a(\alpha)$ from (4.10) and Taylor series expansions of $\mathbf{J}(\alpha)$ from (3.37), i.e.

$$\mathbf{J}_i = \begin{cases} \mathbf{G} + j\Omega\mathbf{C} + \mathbf{T}_o & \text{for } i = 0 \\ \mathbf{T}_i & \text{for } i > 0 \end{cases} \quad (4.17)$$

in (4.8), to obtain

$$\sum_{i,j=0}^{\infty} \mathbf{J}_i^T \mathbf{U}_j \alpha^{i+j} = \mathbf{d} \quad (4.18)$$

To calculate $\mathbf{u}_i$, we equate similar powers of α on both sides to get the following recursive relations

$$\mathbf{J}_0^T \mathbf{u}_0 = \mathbf{d} \quad (4.19)$$

$$\mathbf{J}_0^T \mathbf{u}_i = - \sum_{j=0}^{i-1} \mathbf{J}_{i-j}^T \mathbf{u}_j, \quad i = 1, \dots, q-1 \quad (4.20)$$

(4.19) and (4.20) show that computing $\mathbf{u}_i$ requires one LU decomposition of the matrix $\mathbf{J}_0$ and q Forward/Backward substitutions with multiple right-hand side vectors.

4.3.2 Computational Complexity

Total computational cost in obtaining the sensitivity information comes from finding the steady-state solution and adjoint solution. Computational cost spent on finding steady-state solution is calculated in section 3.3.2. This section discusses the main computational cost associated with finding the adjoint system solution. The computational cost will be spent in obtaining the moments for reducing the adjoint system, and solving the reduced system. Cost spent on these two steps is shown below

- The matrix $\mathbf{J}_0$ used in (4.19) and (4.20) is the same matrix used in finding moments (3.37) for steady-state analysis. Therefore its LU factors will be available. The Taylor series matrices $\mathbf{J}_i$ are also the same ones that have been used during the steady-state analysis step, and hence can be stored for sensitivity computation in (4.19) and (4.20). Hence the extra computing effort needed to compute Taylor series coefficients $\mathbf{u}_i$ will be spent on performing a sequence of q forward/backward substitutions with q right-hand side vectors. This process scales linearly with required computation time due to the fact that $\mathbf{J}_0$ is block diagonal and also each diagonal block is very sparse. Thus its LU factors can retain the same sparsity pattern.

- Order of the reduced system q is very small, hence the associated solution cost can be neglected.

4.4 Summary of the Computational Steps

In this section steps for performing small scale sensitivity analysis using proposed method are summarized. Figure 4.2 provides the computational steps to be followed.

4.5 Proof of Preservation of Derivatives

In this section the proof of preservation of derivatives with respect to α in the reduced adjoint system is presented. The proof is based on *mathematical induction*. We first prove that the 0^{th} derivative, $\mathcal{U}_0$, is preserved. Then we show that the same relation applies to the k^{th} derivative, $\mathcal{U}_k$, provided $(k-1)^{th}$ derivatives are preserved under the mapping P_q . In other words, for $k = 0, \dots, q-1$, if $\mathcal{U}_k$ is the k^{th} derivative of the adjoint system, and $\check{\mathcal{U}}_k$ is that of the reduced system then we have

$$\mathcal{U}_k = P_q \check{\mathcal{U}}_k \quad (4.21)$$

The reduced adjoint system is given as in (4.13),

$$P_q^T (J(\alpha))^T P_q \check{\mathcal{X}}_a(\alpha) = P_q^T d \quad (4.22)$$

where

$$J(\alpha)^T = \bar{Y}^T + \left(\frac{\partial F(\mathbf{X}(\alpha))}{\partial \mathbf{X}} \right)^T \quad (4.23)$$

Let

$$M(\alpha) = J(\alpha)^T \quad (4.24)$$

Steps for Fast Sensitivity Analysis:

1. Compute steady-state solution of the system $\mathbf{X}_c$:
 - (a) Embed α into (2.24) to obtain (2.57).
 - (b) Find moments w.r.t α and construct orthogonal subspace $\mathbf{Q}_q$ (3.24).
 - (c) Construct reduced system (3.44) using $\mathbf{Q}_q$.
 - (d) Solve the reduced system, updating NR iterations (3.50) for $\alpha = 0 \rightarrow 1$.
 - (e) Map reduced system solution back to the original space at $\alpha = 1$,
i.e. $\mathbf{X}_c = \mathbf{Q}_q \hat{\mathbf{X}}(\alpha = 1)$.
2. Compute adjoint system solution of the system $\mathbf{X}_a$:
 - (a) Embed α into (4.6) to obtain (4.8).
 - (b) Find moments w.r.t α and construct orthogonal subspace $\mathbf{P}_q$ (4.11).
 - (c) Construct reduced system (4.13) using $\mathbf{P}_q$.
 - (d) Solve the reduced system (4.13) directly by sweeping $\alpha \{0 \rightarrow 1\}$.
 - (e) Map reduced system solution back to the original space at $\alpha = 1$,
i.e. $\mathbf{X}_a = \mathbf{P}_q \hat{\mathbf{X}}_a(\alpha = 1)$.
3. Substitute $\mathbf{X}_c$ and $\mathbf{X}_a$ in the right hand side of (4.5) to obtain the sensitivity information.

Figure 4.2: Summary of the computational steps

Hence (4.22) can be written as

$$\mathbf{P}_q^T (\mathbf{M}(\alpha)) \mathbf{P}_q \check{\mathbf{X}}_a(\alpha) = \mathbf{P}_q^T \mathbf{d} \quad (4.25)$$

Expanding $\mathbf{M}$ and $\check{\mathbf{x}}_a$ as Taylor series w.r.t α

$$\begin{aligned}\mathbf{M}(\alpha) &= \sum_{k=0} \mathbf{M}_k \alpha^k \\ \check{\mathbf{x}}_a(\alpha) &= \sum_{k=0}^{\infty} \check{\mathbf{u}}_k \alpha^k\end{aligned}\tag{4.26}$$

and substituting in (4.25) results in

$$\mathbf{P}_q^T \sum_{k=0} \mathbf{M}_k \alpha^k \mathbf{P}_q \sum_{k=0}^{\infty} \check{\mathbf{u}}_k \alpha^k = \mathbf{P}_q^T \mathbf{d}\tag{4.27}$$

Equating similar powers of α in (4.27) we obtain

$$[\mathbf{P}_q^T \mathbf{M}_0 \mathbf{P}_q] \check{\mathbf{u}}_0 = \mathbf{P}_q^T \mathbf{d}\tag{4.28}$$

$$[\mathbf{P}_q^T \mathbf{M}_0 \mathbf{P}_q] \check{\mathbf{u}}_k = - \sum_{j=0}^{k-1} [\mathbf{P}_q^T \mathbf{M}_{k-j} \mathbf{P}_q] \check{\mathbf{u}}_j\tag{4.29}$$

From (4.24), $\mathbf{M}_0 = \mathbf{J}_0^T$. Substituting in (4.28) yields

$$\mathbf{P}_q^T (\mathbf{J}_0^T \mathbf{P}_q \check{\mathbf{u}}_0 - \mathbf{d}) = 0\tag{4.30}$$

Since $\mathbf{P}_q$ is an orthonormal basis for the subspace spanned by the columns of the matrix $[\mathbf{u}_0, \dots \mathbf{u}_{q-1}]$, it can be represented in the form [45]

$$\mathbf{P}_q = [\mathbf{u}_0, \dots \mathbf{u}_{q-1}] \mathbf{R}^{-1}\tag{4.31}$$

where $\mathbf{R}$ is an upper triangular matrix. Substituting from (4.31) in (4.30) results in

$$\mathbf{P}_q^T (\mathbf{J}_0^T [\mathbf{u}_0, \dots \mathbf{u}_{q-1}] \mathbf{R}^{-1} \check{\mathbf{u}}_0 - \mathbf{d}) = 0\tag{4.32}$$

Now we verify that $\mathbf{e}_1 = \mathbf{R}^{-1} \check{\mathbf{u}}_0$ satisfies equation (4.32), where $\mathbf{e}_r$ represents the r^{th} column of the identity matrix. Substituting into (4.32) yields

$$\mathbf{P}_q^T (\mathbf{J}_0^T [\mathbf{u}_0, \dots \mathbf{u}_{q-1}] \mathbf{e}_1 - \mathbf{d}) = 0\tag{4.33}$$

Simplifying equation (4.33) results in

$$\mathbf{P}_q^T (\mathbf{J}_0^T \mathbf{u}_0 - \mathbf{d}) = 0 \quad (4.34)$$

Comparing with (4.19) it is clear that Equation (4.34) is satisfied. Therefore we can conclude that $\mathbf{e}_1 = \mathbf{R}^{-1} \check{\mathbf{u}}_0$ is indeed a solution of equation (4.34). $\check{\mathbf{u}}_0$ can then be written as

$$\check{\mathbf{u}}_0 = \mathbf{R} \mathbf{e}_1 \quad (4.35)$$

Pre-multiplying by $\mathbf{P}_q$ results in

$$\mathbf{P}_q \check{\mathbf{u}}_0 = \mathbf{P}_q \mathbf{R} \mathbf{e}_1 = [\mathbf{u}_0, \dots, \mathbf{u}_{q-1}] \mathbf{R}^{-1} \mathbf{R} \mathbf{e}_1 = [\mathbf{u}_0, \dots, \mathbf{u}_{q-1}] \mathbf{e}_1 = \mathbf{u}_0 \quad (4.36)$$

Therefore the 0^{th} moment $\check{\mathbf{u}}_0$ is preserved under the mapping $\mathbf{P}_q$.

Next to prove for the k^{th} derivative, we assume previous derivatives $\check{\mathbf{u}}_0, \dots, \check{\mathbf{u}}_{k-1}$ are proved to be preserved using *mathematical induction*. From (4.24) we obtain

$$\mathbf{M}_k = \mathbf{J}_k^T \quad (4.37)$$

where $\mathbf{J}_k$ is defined in (4.17). Substituting (4.37) in (4.29) yields

$$\mathbf{P}_q^T \left(\mathbf{J}_0^T \mathbf{P}_q \check{\mathbf{u}}_k + \sum_{j=0}^{k-1} \mathbf{J}_{k-j}^T \mathbf{P}_q \check{\mathbf{u}}_j \right) = 0 \quad (4.38)$$

Substituting for $\mathbf{P}_q$ from (4.31) results in

$$\mathbf{P}_q^T \left(\mathbf{J}_0^T [\mathbf{u}_0, \dots, \mathbf{u}_{q-1}] \mathbf{R}^{-1} \check{\mathbf{u}}_k + \sum_{j=0}^{k-1} \mathbf{J}_{k-j}^T \mathbf{u}_j \right) = 0 \quad (4.39)$$

Denoting $\mathbf{e}_{k+1} = \mathbf{R}^{-1} \check{\mathbf{u}}_k$ as a solution for the above equation results in

$$\mathbf{P}_q^T \left(\mathbf{J}_0^T [\mathbf{u}_0, \dots, \mathbf{u}_{q-1}] \mathbf{e}_{k+1} + \sum_{j=0}^{k-1} \mathbf{J}_{k-j}^T \mathbf{u}_j \right) = 0 \quad (4.40)$$

Simplifying (4.40), we obtain

$$\mathbf{P}_q^T \left(\mathbf{J}_0^T \mathbf{u}_k + \sum_{j=0}^{k-1} \mathbf{J}_{k-j}^T \mathbf{u}_j \right) = 0 \quad (4.41)$$

Comparing with (4.20), it is clear that equation (4.41) is satisfied. The k^{th} moment $\check{\mathbf{U}}_k$ of the reduced system can therefore be written as

$$\check{\mathbf{U}}_k = \mathbf{R} \mathbf{e}_{k+1} \quad (4.42)$$

Pre-multiplying by $\mathbf{P}_q$ results in

$$\mathbf{P}_q \check{\mathbf{U}}_k = \mathbf{P}_q \mathbf{R} \mathbf{e}_{k+1} = [\mathbf{U}_0, \dots, \mathbf{U}_{q-1}] \mathbf{R}^{-1} \mathbf{R} \mathbf{e}_{k+1} = [\mathbf{U}_0, \dots, \mathbf{U}_{q-1}] \mathbf{e}_{k+1} = \mathbf{U}_k \quad (4.43)$$

Therefore the k^{th} moment $\mathbf{U}_k$ is preserved under the mapping $\mathbf{P}_q$ if the previous moments $\mathbf{U}_0, \dots, \mathbf{U}_{q-1}$ are preserved. It can be therefore deduced by induction that the reduced system preserves the first q moments of the adjoint system w.r.t α under the mapping $\mathbf{P}_q$, where q is the number of moments of the adjoint system w.r.t α used in constructing $\mathbf{P}_q$.

4.6 Numerical Examples

4.6.1 Example 1

A nonlinear circuit shown in Figure 4.3 is considered in this example to demonstrate the validity of the proposed algorithm. The circuit is stimulated with $\cos(w_0 t)$, where $w_0 = 2\pi 10^6$, and $|V_{in}| = 1V$. The steady-state solution was computed using the harmonic balance technique which needed 32 harmonics to capture the steady-state, which resulted in solving the system of size 520. The proposed method first obtained the steady-state solution using the Krylov subspace technique. This technique reduced the circuit size from 520 to 48. Figure 4.4 shows the comparison between the solution obtained by solving HB equations and solution from Krylov subspace technique. The proposed method then reduced the adjoint system size from 520 to 48. The adjoint solution was obtained by solving the reduced system of size 48 instead of original system of size 520. Sensitivity of the harmonics at any node with respect to any desired circuit parameter can now be computed. The

variation in relative sensitivity $\left(\frac{\partial \mathbf{X}_i}{\partial \lambda} \times \frac{\lambda}{\mathbf{X}_i} \right)$ of 3rd harmonic at the output node (V_{out}) with respect to the capacitor connected to the output terminal V_{out} is computed. Solution obtained using the adjoint technique based on the HB (as described in section 4.2) and the solution from the proposed method are compared in Figure 4.5. Relative sensitivity of first few harmonics at the output node with respect to junction area of the diode connected to the output node obtained from proposed and adjoint methods is compared in Table 4.1.

Figure 4.3: A nonlinear circuit

Figure 4.4: Steady-State response

Figure 4.5: Third harmonic (which has magnitude = $0.00424V$ at $C = 2nF$) sensitivity variation with respect to the capacitor connected to the output node

Table 4.1: Comparing relative sensitivity of first few harmonics w.r.t the junction area of the diode connected to the output node

Harmonic Number	Harmonic Magnitude	Relative Sensitivity	
		Adjoint Method	Proposed Method
1	0.0206	0.1649	0.1647
2	0.0028	0.4249	0.4259
3	0.0023	0.1653	0.1642
4	0.0020	0.0862	0.0869

The summary of the computational cost involved in performing sensitivity analysis using the proposed and the traditional adjoint technique is given in Table 4.2. The first row shows the computational cost spent by the conventional adjoint method in obtaining the sensitivity information. The conventional method operates on the original system of size 520 by sweeping α from $0 \rightarrow 1$ and updating NR equations at each power step until convergence. The number of incremental steps of α used in this example is 20 and at each step, the number of NR iterations is assumed to be 5. In this case, to obtain the steady-state solution we require 100 LU decompositions and 100 F/B substitutions. The adjoint system solution is obtained at the cost of 1 F/B substitution. Hence total cost for sensitivity computation is 100LU+101F/B of system of size 520. The computational cost spent by the proposed method is explained in the second and third rows. The second row shows the cost spent on computing the moments required for reducing the original system and the adjoint system. From equation (3.37), to compute moments for the original system we require 1 LU decomposition of the Jacobian matrix of the original system at DC and 47 F/B substitutions for first 48 moments. Since the first moment is the DC solution, which is already available from (3.28). From (4.19) and (4.20), to compute moments for the adjoint system we require only 48 F/B substitutions for first 48 moments, since LU factors are available

from (3.37). Hence the total cost for computing the moments is 1LU+95F/B of system of size 520 (original system). The biggest advantage is that the Jacobian matrix is sparse and is block diagonal in structure; hence reducing the computational effort for LU decomposition and F/B substitutions. The cost involved in solving the reduced system is shown in third row. The number of incremental steps of α used for this case is 20 and at each step, assuming the number of NR iterations to be 5, we require 100 LU decompositions and 100 F/B substitutions of the reduced system of size 48 in obtaining the steady-state solution. To obtain the adjoint system solution, we require one LU decomposition of the reduced Jacobian of size 48 and one corresponding F/B substitution. Hence the total cost involving the reduced system (of size 48) is 101LU+101F/B. In summary, computing adjoint sensitivity using the proposed method requires 1LU+95F/B of the large system (of size 520) and 101LU+101F/B of the reduced system (of size 48), for this example.

Table 4.2: Computational cost comparison of the proposed method v/s HB

System Size	Jacobian Size	NR Count for each step	Total num of steps	Total num of LU	Total num of F/B	Total Cost (LU + F/B)
Conventional Adjoint Method						
520	520	5	20	100	100 (for original steady-state)	100 LU + 101 F/B (of large system)
Harmonics = 32					+ 1 (for adjoint solution)	
Proposed Method						
520	–	–	–	1	47 (for moments of original) + 48 (for moments of adjoint)	1 LU + 95 F/B (of large system)
–	48	5	20	100 (for original steady-state) + 1 (for adjoint solution)	100 (for original steady-state) + 1 (for adjoint solution)	101 LU + 101 F/B (of small system)

4.6.2 Example 2

A RC coupled amplifier shown in Figure 4.6 is considered in this example. The steady-state time-domain response was first simulated due to an input of $e(t) = \cos(\omega_0 t)$ with $\omega_0 = 2\pi(2.07 \times 10^7)$. Direct HB technique needed $K = 16$ harmonics to approximate the steady-state response, and therefore required solving a system of size $n(2K + 1) = 13 \times (2 \times 16 + 1) = 429$ simultaneous nonlinear equations. On the other hand, using Krylov subspace technique required solving a system of size 64 nonlinear equations. Figure 4.7 displays a comparison for the steady-state response obtained from the two approaches. After computing the steady-state solution, proposed method considered solving the adjoint system. The proposed method reduced the adjoint system of size 429 to a reduced system of size 48. The reduced adjoint system was solved to obtain the adjoint solution. In this case relative sensitivity $\left(\frac{\partial \mathbf{X}_i}{\partial \lambda} \times \frac{\lambda}{\mathbf{X}_i}\right)$ of first 3 harmonics at the output node with respect to the junction area of the transistor were calculated. To verify the accuracy of the results obtained, relative sensitivity was computed using the *exact* approach by using the solutions obtained from the HB technique and solutions obtained from the adjoint system (4.6). Table 4.3 shows the results obtained from both approaches. Figure 4.8 gives the relative error $\left(\frac{(adjoint_{conventional} - adjoint_{proposed})}{adjoint_{proposed}}\right)$ in the sensitivity of the individual harmonics with respect to the tank capacitor C , obtained from the proposed method.

Table 4.3: Relative sensitivity of first 3 harmonics w.r.t the junction area of the transistor

Harmonic Number	Harmonic Magnitude	Relative Sensitivity	
		Adjoint Method	Proposed Method
1	0.1858	0.0141	0.0141
2	0.0280	0.0186	0.0186
3	0.0174	0.0196	0.0196

Figure 4.6: A tuned amplifier circuit

Figure 4.7: Steady-state response at the output node of the tuned amplifier circuit.

Figure 4.8: Relative error in the sensitivity of the harmonics with respect to the capacitor C

4.7 Conclusion

In this chapter an efficient algorithm for performing small scale sensitivity analysis was proposed. The steady-state solution was first obtained using Krylov subspace reduction technique. Proposed algorithm then obtains adjoint system solution using Krylov subspace reduction technique. Adjoint solution thus obtained and steady-state solution, were used to perform sensitivity analysis at a very low CPU cost. The main advantage of the proposed algorithm is that, sensitivity analysis is performed without incurring any additional significant CPU cost on top of the CPU cost spent in obtaining the steady-state solution based on Krylov subspace technique. This is due to the fact that the Taylor series coefficients of the matrix J_o were already computed while obtaining the steady-state solution. Small scale sensitivity analysis with respect to linear as well as non-linear parameters are performed. The new approach was implemented and tested and shown to be both accurate and efficient.

Chapter 5

MOR Techniques for Large Scale Sensitivity Analysis

5.1 Introduction

Steady-state analysis has always been of primary interest to the circuit designers since system performance is easier to characterize and verify when the system is in steady-state. Also another equally important problem is the problem of repeated steady-state analysis. Repeated analysis is required in obtaining certain figures of merit (FOM) of RF circuits such as IIP3, 1 dB compression etc [50], [51]. For that, the steady-state analysis is repeated by sweeping the input power. Another important area where repeated steady-state analysis is used is in optimization, here the designer tweaks the system parameter value and performs steady-state analysis to verify the effect. In such problems the key idea is to track the steady-state solution with respect to the variation in certain parameters of the circuit over a wide range of values. This problem of repeated steady-state analysis is referred as large scale sensitivity analysis.

The complexity involved in obtaining the steady-state solution is ever increasing with

the advancements in the technology. The hurdles in computing the steady-state solution using the conventional techniques was explained in section 4.1. Recently, as a solution to the problems associated with steady-state analysis an algorithm to compute the steady-state in a efficient way was proposed [5]. This technique was explained in section 2.3.1. Even though the technique offers a significant reduction in CPU cost in finding the system steady-state, for repeated steady-state analysis this method is not feasible. This is because every time when the system is perturbed, i.e., when any circuit component value is changed, a fresh DC solution and a new subspace have to be found for obtaining the steady-state solution. The problem becomes more worse if obtaining the DC solution of the system is not trivial.

In this chapter a new algorithm is presented for performing repeated steady-state analysis in a efficient way [7]. Using the proposed algorithm the steady-state solutions for variations in the system parameters are obtained without repeating the steady-state analysis procedure. The algorithm is based on multi-dimensional reduction technique [47]. The proposed algorithm first constructs an orthogonal subspace. This orthogonal subspace is then used to reduce the original large system of equations to a small parameterized system of equations. Steady-state solution for any variation in the desired parameter is obtained by solving the parameterized reduced system and mapping the solution back to the original system. The size of the parameterized reduced system is very small compared to the original system. The major advantage of the proposed algorithm is that, for obtaining repeated steady-state solutions for any variations in the desired parameters, we have to find a parameterized subspace only once.

This chapter is organised as follows. Section 5.2 explains the proposed algorithm. Section 5.3 illustrates the computation of the subspace used in the circuit reduction. Section 5.4 discusses the computational complexity of the proposed algorithm. Section 6.1 gives a background on multi-tone analysis. Section 5.5 presents the proof of the preservation of

the derivatives in the reduced system. Section 6.2 presents the numerical testing examples.

5.2 Proposed Method

Any circuit with linear and nonlinear components can be formulated using HB equations as in (2.24). Embedding continuation α in (2.24) as explained in section 2.3.2, we obtain

$$\bar{G}\mathbf{X} + j\Omega\bar{C}\mathbf{X} + \mathbf{F}(\mathbf{X}) = \mathbf{B}_{dc} + \alpha\mathbf{B}_{ac} \quad (5.1)$$

The main target of the proposed algorithm is to obtain the steady state solution of (5.1) for any change in circuit component value without repeating the whole steady-state analysis each time. The proposed algorithm has two main steps. First, the steady solution of (5.1) is found by constructing a parameterized reduced system of smaller dimension using multi-dimensional reduction technique [47], and solving it. Next, fresh steady-state solution, for any variation among the desired parameters, is obtained by reducing the new perturbed system using the orthogonal subspace found in step one and solving it. It is important to note that most of the computational cost comes from step one since second step reuses the subspace found in the first step. The details of the algorithm is presented next.

5.2.1 Obtaining Steady-State Solution of the Original System

First, a multi-dimensional orthogonal subspace, $\mathbf{Q}_z$, is constructed. This subspace, $\mathbf{Q}_z \in \mathbb{R}^{N_{hb} \times z}$, is an orthogonal basis for the subspace spanned by first q Taylor series coefficients (or scaled derivatives) of $\mathbf{X}$ with respect to $\alpha, \lambda_1, \dots, \lambda_n$ at $\alpha = 0$. Where λ_i is the parameter with respect to which the steady-state solution is tracked. The orthogonal subspace $\mathbf{Q}_z$ will be reused for tracking steady-state solution with respect to parameter variations in

$\lambda_1, \dots, \lambda_n$. It is given by

$$\text{ColumnSpan}\{Q_z\} = \text{ColumnSpan}\{M_\alpha \ M_{\lambda_1} \ \dots \ M_{\lambda_n}\} \quad (5.2)$$

where $M_\alpha, M_{\lambda_1}, \dots, M_{\lambda_n}$ are matrices containing Taylor series coefficients of X of the system (5.1) with respect to $\alpha, \lambda_1, \dots, \lambda_n$ at $\alpha = 0$ respectively, i.e. if the solution trajectory X of the system (5.1) is expanded as a Taylor series,

$$\begin{aligned} X(\alpha, \lambda) &= \sum_{k=0}^q A_{(\alpha,k)} \alpha^k \\ X(\alpha, \lambda) &= \sum_{k=0}^q A_{(\lambda_1,k)} (\lambda_1 - \lambda_{1,o})^k \\ &\vdots \end{aligned} \quad (5.3)$$

$$X(\alpha, \lambda) = \sum_{k=0}^q A_{(\lambda_n,k)} (\lambda_n - \lambda_{n,o})^k \quad (5.4)$$

then

$$\begin{aligned} M_\alpha &= [A_{(\alpha,0)}, A_{(\alpha,1)}, \dots, A_{(\alpha,q-1)}] \\ M_{\lambda_1} &= [A_{(\lambda_1,0)}, A_{(\lambda_1,1)}, \dots, A_{(\lambda_1,q-1)}] \\ &\vdots \\ M_{\lambda_n} &= [A_{(\lambda_n,0)}, A_{(\lambda_n,1)}, \dots, A_{(\lambda_n,q-1)}] \end{aligned} \quad (5.5)$$

Calculation of Taylors series derivatives is presented in section 5.3. The circuit is reduced by the following change of variables

$$X \rightarrow Q_z \hat{X} \quad (5.6)$$

With the change of variables in (5.6), the augmented harmonic balance equations in (5.1) become

$$\bar{G}Q_z\hat{X} + j\Omega\bar{C}Q_z\hat{X} + F(Q_z\hat{X}) = B_{dc} + \alpha B_{ac} \quad (5.7)$$

Equation (5.7) is then pre-multiplied by Q_z^T , and the following reduced system is obtained

$$\hat{G}\hat{X} + \hat{C}\hat{X} + \hat{F}(\hat{X}) - \hat{B}_{dc} - \alpha\hat{B}_{ac} = 0 \quad (5.8)$$

where

$$\begin{aligned} \hat{G} &= Q_z^T \bar{G} Q_z \\ \hat{C} &= Q_z^T j\Omega \bar{C} Q_z \\ \hat{F}(\hat{X}) &= Q_z^T F(Q_z \hat{X}) \\ \hat{B}_{dc} &= Q_z^T B_{dc} \end{aligned} \quad (5.9)$$

The basic idea behind this change of variables is that the reduced system obtained by this transformation preserves the first q derivatives of $\mathbf{X}$ with respect to $\alpha, \lambda_1, \dots, \lambda_n$ at $\alpha = 0$ under the mapping Q_z . Proof of conservation of the derivatives is given in section 5.5. To solve the reduced system (5.8), Newton iterations are updated with an initial guess $\hat{X}_{guess}$ as follows,

$$\hat{X}_{new} = \hat{X}_{old} - \hat{J}(\hat{X}_{old})^{(-1)} \hat{\Psi}_i(\hat{X}_{old}) \quad (5.10)$$

where $\hat{X}_{old}$ is the old guess, $\hat{X}_{new}$ is the new updated solution, and $\hat{J}(\hat{X}_{old})$ is the Jacobian of the reduced equations, which can be written as

$$\hat{J}(\hat{X}_{old}) = \hat{G} + \hat{C} + Q_z^T \left(\frac{\partial F(\mathbf{X})}{\partial \mathbf{X}} \bigg|_{Q_z \hat{X}_{old}} \right) Q_z \quad (5.11)$$

and

$$\hat{\Psi}_i(\hat{X}) = \hat{G}\hat{X} + \hat{C}\hat{X} + \hat{F}(\hat{X}) - \hat{B}_t \quad (5.12)$$

with

$$\hat{B}_t = \hat{B}_{dc} + \alpha_i \hat{B}_{ac} \quad (5.13)$$

The iteration in (5.10) is repeated until the difference $||\Delta \hat{X}||$ between the old solution $\hat{X}_{old}$ and the updated value is less than the error tolerance ϵ . This process is repeated for $\alpha = \{0 \text{ to } 1\}$ using solution obtained at the convergence of (5.10) as initial guess for next set of iterations. Once the convergence of (5.10) is achieved at $\alpha = 1$, the solution is mapped back to the original system space to obtain steady-state solution of (5.1)

$$Q_z \hat{X} \rightarrow X \quad (5.14)$$

An estimate of the error can be obtained by substituting the solution X into the harmonic balance equations in (5.1), and checking if Kirchhoff's current law is satisfied.

5.2.2 Tracking Steady-State Solution for Variations in Parameters

Once steady-state of the system is obtained, the new steady-state solution for variations in $\lambda_1, \dots, \lambda_n$ is found without incurring any additional significant cost. Consider one of the parameters among $\lambda_1, \dots, \lambda_n$, say λ_p , is varied to obtain a new system, let us call it the perturbed system, given by

$$\bar{G}_p X_p + j\Omega \bar{C}_p X_p + F_p(X_p) = B \quad (5.15)$$

where

$$B = B_{dc} + B_{ac} \quad (5.16)$$

The steady-state solution of this perturbed system is obtained in two simple steps

1. The perturbed system (5.15) is reduced by congruent transform using orthogonal basis Q_z (5.2), to obtain

$$\hat{G}_p \hat{X}_p + \hat{C}_p \hat{X}_p + \hat{F}_p(\hat{X}_p) = \hat{B} \quad (5.17)$$

where

$$\hat{G}_p = Q_z^T \bar{G}_p Q_z; \quad \hat{C}_p = Q_z^T J \Omega \bar{C}_p Q_z; \quad \hat{F}_p(\hat{X}) = Q_z^T F_p(Q_z X_p); \quad \hat{B} = Q_z^T B$$

2. The reduced system is solved by updating Newton iterations using steady-state solution of the original system X (5.14) as initial guess, i.e. using $\hat{X}_{guess} = \hat{X}$ Newton iterations (5.18) is performed till convergence.

$$\hat{X}_{p_{new}} = \hat{X}_{p_{old}} - \left(\hat{J}(\hat{X}_{p_{old}}) \right)^{-1} \hat{\Psi}(\hat{X}_{p_{old}}) \quad (5.18)$$

where

$$\hat{J}(\hat{X}_{p_{old}}) = \hat{G}_p + \hat{C}_p + Q_z^T \left(\frac{\partial F_p(X_p)}{\partial X_p} \bigg|_{Q_z \hat{X}_{p_{old}}} \right) Q_z \quad (5.19)$$

and

$$\hat{\Psi}(\hat{X}_p) = \hat{G}_p \hat{X}_p + \hat{C}_p \hat{X}_p + \hat{F}_p(\hat{X}_p) - \hat{B} \quad (5.20)$$

The steady state solution obtained at the convergence of NR iterations (5.18) is mapped back to the perturbed system solution. i.e.

$$Q_z \hat{X}_p \rightarrow X_p \quad (5.21)$$

Note:

- In step 2 continuation method is omitted since initial guess is pretty close to the required solution, aiding the convergence of NR iterations (5.18).
- If (5.18) does not converge then the number of moments A_α and A_{λ_p} have to be increased while constructing the orthogonal subspace. This is to capture steady-state solution when parameters are varied over relatively larger ranges.

For any variation among the parameters $\lambda_1, \dots, \lambda_n$, the steady state of the new perturbed system is obtained by following the two simple steps shown above. Hence steady-state solution is tracked for any variations among the desired parameters at a very low computational cost. The steps involved in the proposed algorithm is summarized in Figure 5.1.

Figure 5.1: Summary of the proposed algorithm

5.3 Computation of Derivatives

This section addresses the issue of computing the derivatives used in constructing the orthogonal basis Q_z in (5.2). The computation of derivatives with respect to α i.e. $A_{(\alpha,i)}$

is shown in section 3.3.2. In this section derivatives $\mathbf{A}_{(\lambda,i)}$ are computed depending on whether the parameters λ belongs to the linear or nonlinear part of the network. It is important to note that the derivatives ($\mathbf{A}_{(\lambda,i)}$) are computed at the DC operating point of the network, i.e. at $\alpha = 0$. For notational simplicity $\mathbf{A}_{(\lambda,i)}$ is denoted by $\mathbf{P}_i$.

5.3.1 Derivatives with respect to Linear Parameters

We show in this subsection how to compute $\mathbf{P}_i$ when λ belongs to linear part of the network. Denote the contribution of linear part of the perturbed network by $\bar{\mathbf{Y}}$, where $\bar{\mathbf{Y}} = \bar{\mathbf{G}} + j\Omega\bar{\mathbf{C}}$. Since the derivatives ($\mathbf{P}_i$) are computed at $\alpha = 0$, (5.1) can be written as

$$\bar{\mathbf{Y}}(\lambda)\mathbf{X}(\lambda) + \mathbf{F}(\mathbf{X}(\lambda)) = \mathbf{B}_{dc} \quad (5.22)$$

Expand $\bar{\mathbf{Y}}(\lambda)$, $\mathbf{X}(\lambda)$ and $\mathbf{F}(\mathbf{X}(\lambda))$ as a taylor series with respect to λ at λ_o , i.e.

$$\begin{aligned} \bar{\mathbf{Y}}(\lambda) &= \sum_{i=0} \bar{\mathbf{Y}}_i(\lambda - \lambda_o)^i & \mathbf{X}(\lambda) &= \sum_{i=0} \mathbf{P}_i(\lambda - \lambda_o)^i \\ \mathbf{F}(\mathbf{X}(\lambda)) &= \sum_{i=0} \mathbf{V}_i(\lambda - \lambda_o)^i \end{aligned} \quad (5.23)$$

Where λ_o is parameter value of the original system. Substituting from (5.23) into (5.22) we obtain

$$\sum_{i=0} \bar{\mathbf{Y}}_i(\lambda - \lambda_o)^i \sum_{i=0} \mathbf{P}_i(\lambda - \lambda_o)^i + \sum_{i=0} \mathbf{V}_i(\lambda - \lambda_o)^i = \mathbf{B}_{dc} \quad (5.24)$$

We now start equating similar powers of λ on both the sides. From terms independent of λ we obtain

$$\bar{\mathbf{Y}}_o \mathbf{P}_o + \mathbf{V}_o = \mathbf{B}_{dc} \quad (5.25)$$

where $\mathbf{F}(\mathbf{X}(\lambda_o)) = \mathbf{V}_o$. Note that $\mathbf{P}_o$ in (5.25) is nothing but the DC solution of the original system in (5.1). Also equating the first powers of λ yields

$$\bar{\mathbf{Y}}_1 \mathbf{P}_o + \bar{\mathbf{Y}}_o \mathbf{P}_1 + \mathbf{V}_1 = 0 \quad (5.26)$$

Now define $\mathbf{J}(\lambda)$ to be

$$\mathbf{J}(\lambda) = \frac{d\mathbf{F}}{d\mathbf{X}}; \quad \text{with } \mathbf{J}_o = \mathbf{J}(\lambda)|_{\lambda_o} \quad (5.27)$$

Since $\mathbf{V}_1 = \left. \frac{d\mathbf{F}}{d\lambda} \right|_{\lambda_o} = \frac{d\mathbf{F}}{d\mathbf{X}} \mathbf{P}_1$, then substituting in (5.26) gives the following relation for $\mathbf{P}_1$:

$$(\bar{\mathbf{Y}}_o + \mathbf{J}_o) \mathbf{P}_1 = -\bar{\mathbf{Y}}_1 \mathbf{P}_o \quad (5.28)$$

For computing $\mathbf{P}_2, \dots, \mathbf{P}_n$, we expand $\mathbf{J}(\lambda)$ as a Taylor's series in λ at $\lambda = \lambda_o$

$$\mathbf{J}(\lambda) = \sum_{\kappa=0} \mathbf{J}_{\kappa} (\lambda - \lambda_o)^{\kappa} \quad (5.29)$$

Using this expansion to write $\frac{d\mathbf{F}}{d\lambda}$ in the following form:

$$\begin{aligned} \frac{d\mathbf{F}}{d\lambda} &= \mathbf{J} \frac{d\mathbf{X}}{d\lambda} \\ \sum_{i=1} i \mathbf{V}_i (\lambda - \lambda_o)^{i-1} &= \sum_{i=1} \mathbf{J}_i (\lambda - \lambda_o)^i \sum_{i=1} i \mathbf{P}_i (\lambda - \lambda_o)^{i-1} \end{aligned} \quad (5.30)$$

Taking the n^{th} derivative in (5.30) and substituting $\lambda = \lambda_o$, we get the following recursive relationship:

$$\mathbf{V}_n = \mathbf{J}_o \mathbf{P}_n + \frac{1}{n} \left[\sum_{j=1}^{n-1} (n-j) \mathbf{J}_j \mathbf{P}_{n-j} \right] \quad (5.31)$$

Thus taking the n^{th} derivative and substituting $\lambda = \lambda_o$ in (5.24) results in

$$(\bar{\mathbf{Y}}_o + \mathbf{J}_o) \mathbf{P}_n = -\frac{1}{n} \left[\sum_{j=1}^{n-1} (n-j) \mathbf{J}_j \mathbf{P}_{n-j} \right] - \sum_{j=0}^{n-1} \bar{\mathbf{Y}}_{n-j} \mathbf{P}_j \quad (5.32)$$

From (5.25) and (5.32) it can be seen that to compute $\mathbf{P}_o, \dots, \mathbf{P}_n$ requires only one LU decomposition of the Jacobian matrix and each higher derivative can be obtained by one forward/backward substitution. Computing terms required for finding $\mathbf{P}_n$ using (5.32), i.e. $\mathbf{J}_n$ and $\mathbf{Y}_n$ is shown next.

Computing terms J_i s and Y_i s

Computing the matrix J_0 is shown in section 2.3.1. For the sake of simplicity we consider a circuit with nonlinearity appearing at only one node. Let this nonlinearity be due to a nonlinear resistor governed by the relation $i(x) = f(x)$. In this case J can be expressed as

$$J = \Gamma G \Gamma^{-1}$$

with

$$G = \begin{bmatrix} g(x(t_1, \lambda)) & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \\ & & \ddots & 0 \\ 0 & \cdots & 0 & g(x(t_s, \lambda)) \end{bmatrix}; \quad \text{where } g(x) = \frac{\partial f(x)}{\partial x}$$

where $t_1 \ t_2 \ \cdots \ t_s$ are time points over the fundamental period. Noting the Taylor expansions

$$\begin{aligned} x &= \sum_{i=0} p_i \lambda^i \\ f(x) &= \sum_{i=0} v_i \lambda^i \\ g(x) = \frac{\partial f}{\partial x} &= \sum_{i=0} g_i \lambda^i \end{aligned}$$

We can write the expression for J_i as

$$J_i = \Gamma G_i \Gamma^{-1}$$

where G_i is a diagonal vector whose elements are the time samples of g_i . As an example of computing g_i , consider the case where,

$$\begin{aligned} f(x) &= I_s (\exp(x/V_T) - 1) \\ \text{therefore } g(x) &= \frac{I_s}{V_T} (\exp(x/V_T)) \end{aligned} \tag{5.33}$$

Table 5.1: $\bar{Y}_1$ when λ is either R , L or C

$\frac{dY}{dR}$		$\frac{dY}{dL}$		$\frac{dY}{dC}$
$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{R^2} & \frac{1}{R^2} & 0 \\ 0 & \frac{1}{R^2} & -\frac{1}{R^2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$j\Omega$	$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix}$	$j\Omega$	$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

By recursively differentiating (5.33) it can be shown that

$$\mathbf{g}_n = \frac{1}{nV_T} \sum_{i=0}^{n-1} \mathbf{g}_i \mathbf{p}_{n-i} (n-i) \quad \text{with} \quad \mathbf{g}_0 = \frac{I_s}{V_T} (\exp(\mathbf{p}_0/V_T)) \quad (5.34)$$

Similar expansions can be found when nonlinearity appears in different forms, i.e. $\mathbf{f}(x)$ has different relations. Expansions for some functions can be found in Table 3.1.

$\bar{Y}_i$ are computed by differentiating $\bar{Y}$ directly. Table 5.1 shows $\bar{Y}_1$ when λ is either R , L or C . Higher order $\bar{Y}_i$ are obtained by repeated differentiation of $\bar{Y}_1$ from Table 5.1.

5.3.2 Derivatives with respect to Nonlinear Parameters

The procedure for computing derivatives ($\mathbf{P}_i$) when λ belongs to nonlinear part of the network is shown in this section. Denoting $\bar{\mathbf{G}} + j\Omega\bar{\mathbf{C}} = \bar{\mathbf{Y}}$, and substituting $\alpha = 0$ (for analyzing the system at DC operating point), the system of nonlinear equations (5.1) can be written as

$$\bar{\mathbf{Y}}\mathbf{X}(\lambda) + \mathbf{F}(\mathbf{X}(\lambda), \lambda) = \mathbf{B}_{dc} \quad (5.35)$$

where the explicit dependency on λ appears in the nonlinear part only. Expand $\mathbf{X}(\lambda)$ and $\mathbf{F}(\mathbf{X}(\lambda), \lambda)$ as a Taylor series with respect to λ at λ_o (5.23), i.e.

$$\begin{aligned}\mathbf{X}(\lambda) &= \sum_{i=0} \mathbf{P}_i (\lambda - \lambda_o)^i \\ \mathbf{F}(\mathbf{X}(\lambda)) &= \sum_{i=0} \mathbf{V}_i (\lambda - \lambda_o)^i\end{aligned}\quad (5.36)$$

Where λ_o is the parameter value of the original system. Substituting from (5.36) into (5.35), we obtain

$$\bar{\mathbf{Y}} \sum_{i=0} \mathbf{P}_i (\lambda - \lambda_o)^i + \sum_{i=0} \mathbf{V}_i (\lambda - \lambda_o)^i = \mathbf{B}_{dc} \quad (5.37)$$

$\mathbf{P}_i$'s are obtained by equating similar powers of λ on both the sides of (5.37). From terms independent of λ we obtain

$$\bar{\mathbf{Y}} \mathbf{P}_o + \mathbf{V}_o = \mathbf{B}_{dc} \quad (5.38)$$

i.e.

$$\bar{\mathbf{Y}} \mathbf{X}(\lambda_o) + \mathbf{F}(\mathbf{X}(\lambda_o), \lambda_o) = \mathbf{B}_{dc} \quad (5.39)$$

From (5.39) it is clear that $\mathbf{P}_o$ in (5.38) is nothing but the DC solution of the original system in (5.1). Equating first powers of λ on both sides in (5.37), yields

$$\bar{\mathbf{Y}} \mathbf{P}_1 + \mathbf{V}_1 = 0 \quad (5.40)$$

Now define $\mathbf{U}(\lambda)$ and $\mathbf{J}(\lambda)$ to be

$$\begin{aligned}\mathbf{U}(\lambda) &= \frac{\partial \mathbf{F}}{\partial \lambda} & \mathbf{U}_0 &= \left. \frac{\partial \mathbf{F}}{\partial \lambda} \right|_{\lambda_o} \\ \mathbf{J}(\lambda) &= \frac{\partial \mathbf{F}}{\partial \mathbf{X}} & \mathbf{J}_0 &= \left. \frac{\partial \mathbf{F}}{\partial \mathbf{X}} \right|_{\lambda_o}\end{aligned}\quad (5.41)$$

Then $\mathbf{V}_1$ can be written as

$$\begin{aligned}\mathbf{V}_1 &= \left. \frac{d\mathbf{F}}{d\lambda} \right|_{\lambda_o} = \left(\frac{\partial \mathbf{F}}{\partial \lambda} + \frac{\partial \mathbf{F}}{\partial \mathbf{X}} \frac{\partial \mathbf{X}}{\partial \lambda} \right) \Big|_{\lambda_o} \\ &= \mathbf{U}_0 + \mathbf{J}_0 \mathbf{P}_1\end{aligned}\quad (5.42)$$

Using (5.42) in (5.40) we obtain $\mathbf{P}_1$ as

$$(\bar{\mathbf{Y}} + \mathbf{J}_0)\mathbf{P}_1 = -\mathbf{U}_0 \quad (5.43)$$

To compute higher derivatives i.e. $\mathbf{P}_2, \dots, \mathbf{P}_n$ we expand $\mathbf{U}(\lambda)$ and $\mathbf{J}(\lambda)$ as Taylor series in λ , i.e.

$$\begin{aligned} \mathbf{U}(\lambda) &= \sum_{i=0}^{\infty} \mathbf{U}_i(\lambda - \lambda_0)^i \\ \mathbf{J}(\lambda) &= \sum_{i=0}^{\infty} \mathbf{J}_i(\lambda - \lambda_0)^i \end{aligned} \quad (5.44)$$

Using (5.36) and (5.44) $\frac{d\mathbf{F}}{d\lambda}$ can be expressed as:

$$\begin{aligned} \frac{d\mathbf{F}}{d\lambda} &= \mathbf{U} + \mathbf{J} \frac{\partial \mathbf{X}}{\partial \lambda} \\ \sum_{i=0}^{\infty} i \mathbf{V}_i(\lambda - \lambda_0)^{i-1} &= \sum_{i=0}^{\infty} \mathbf{U}_i(\lambda - \lambda_0)^i + \sum_{i=0}^{\infty} \mathbf{J}_i(\lambda - \lambda_0)^i \sum_{i=0}^{\infty} i \mathbf{P}_i(\lambda - \lambda_0)^{i-1} \end{aligned} \quad (5.45)$$

Taking the n^{th} derivative with respect to λ in (5.45) and substituting $\lambda = \lambda_0$ results in

$$\mathbf{V}_n = \mathbf{J}_0 \mathbf{P}_n + \frac{1}{n} \left(\sum_{j=1}^{n-1} (n-j) \mathbf{J}_j \mathbf{P}_{n-j} + \mathbf{U}_{n-1} \right) \quad \text{where } n \geq 1 \quad (5.46)$$

Equating the n^{th} power of λ in (5.37) we obtain

$$\bar{\mathbf{Y}} \mathbf{P}_n + \mathbf{V}_n = 0 \quad (5.47)$$

Substituting (5.46) in (5.47) we obtain recursive relation for computing n^{th} derivative as

$$(\bar{\mathbf{Y}} + \mathbf{J}_0) \mathbf{P}_n = -\frac{1}{n} \left(\sum_{j=1}^{n-1} (n-j) \mathbf{J}_j \mathbf{P}_{n-j} + \mathbf{U}_{n-1} \right) \quad \text{where } n \geq 1 \quad (5.48)$$

From (5.38) and (5.48) it can be seen that $\mathbf{P}_0, \dots, \mathbf{P}_n$ can be computed at the expense of only one LU decomposition of the Jacobian matrix $(\bar{\mathbf{Y}} + \mathbf{J}_0)$, and n forward/backward substitutions. Computation of terms $\mathbf{J}_i$ and $\mathbf{U}_i$ is shown next.

Computing terms J_i and U_i

For the sake of simplicity, we consider nonlinearity appearing at only one node and let the type of nonlinearity be same as in previous section, i.e. $i(x) = f(x)$. $\mathbf{J}$ and $\mathbf{U}$ can be expressed as

$$\mathbf{J} = \Gamma \mathbf{G} \Gamma^{-1}$$

$$\mathbf{U} = \Gamma \mathbf{H}$$

with

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}(\mathbf{x}(t_1), \lambda) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{g}(\mathbf{x}(t_s), \lambda) \end{bmatrix}; \quad \text{where } \mathbf{g}(\mathbf{x}, \lambda) = \frac{\partial \mathbf{f}(\mathbf{x}, \lambda)}{\partial \mathbf{x}} \quad (5.49)$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}(t_1), \lambda) \\ \vdots \\ \mathbf{h}(\mathbf{x}(t_s), \lambda) \end{bmatrix}; \quad \text{where } \mathbf{h}(\mathbf{x}, \lambda) = \frac{\partial \mathbf{f}(\mathbf{x}, \lambda)}{\partial \lambda} \quad (5.50)$$

where $t_1 \ t_2 \ \cdots \ t_s$ are time points over the fundamental period. Noting the Taylor expansions

$$\begin{aligned} \mathbf{x} &= \sum_{i=0} \mathbf{p}_i \lambda^i \\ \mathbf{f}(\mathbf{x}, \lambda) &= \sum_{i=0} \mathbf{v}_i \lambda^i \\ \mathbf{g}(\mathbf{x}, \lambda) &= \frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \sum_{i=0} \mathbf{g}_i \lambda^i \\ \mathbf{h}(\mathbf{x}, \lambda) &= \frac{\partial \mathbf{f}}{\partial \lambda} = \sum_{i=0} \mathbf{h}_i \lambda^i \end{aligned}$$

We can write the expression for $\mathbf{J}_i$ and $\mathbf{U}_i$ as

$$\begin{aligned}\mathbf{J}_i &= \Gamma \mathbf{G}_i \Gamma^{-1} \\ \mathbf{U}_i &= \Gamma \mathbf{H}_i\end{aligned}\tag{5.51}$$

where $\mathbf{G}_i$ is a diagonal vector whose elements are the time samples of $\mathbf{g}_i$, and $\mathbf{H}_i$ is column vector whose elements are the time samples of $\mathbf{h}_i$. For computing $\mathbf{g}_i$ and $\mathbf{h}_i$, consider the example from previous section, i.e.

$$\mathbf{f}(x) = I_s(\exp(\mathbf{x}/V_T) - 1)\tag{5.52}$$

Let λ represent I_s , the diode saturation current. We obtain

$$\begin{aligned}\mathbf{g}(x, \lambda) &= \frac{I_s}{V_T}(\exp(\mathbf{x}/V_T)) \\ \mathbf{h}(x, \lambda) &= \frac{1}{V_T}(\exp(\mathbf{x}/V_T))\end{aligned}\tag{5.53}$$

We can write $g(x, \lambda)$ as

$$g(x, \lambda) = a(\lambda) * b(x(\lambda))\tag{5.54}$$

where

$$a(\lambda) = I_s \quad \text{and} \quad b(x(\lambda)) = \frac{1}{V_T}(\exp(\mathbf{x}/V_T))\tag{5.55}$$

From Table 3.1 it can be shown that

$$\begin{aligned}\mathbf{g}_n &= \sum_{i=0}^n \mathbf{b}_i \mathbf{a}_{n-i} \\ \mathbf{h}_n &= \mathbf{b}_n; \quad \text{since } \mathbf{h}(x, \lambda) = \mathbf{b}(x(\lambda))\end{aligned}\tag{5.56}$$

where a_i and b_i can be obtained from Table 3.1 as

$$\begin{aligned}\mathbf{a}_n &= \begin{cases} I_s & \text{for } n = 0 \\ 0 & \text{for } n > 0 \end{cases} \\ \mathbf{b}_n &= \frac{1}{nV_T} \sum_{i=0}^{n-1} \mathbf{b}_i \mathbf{p}_{n-i}(n-i) \quad \text{with} \quad \mathbf{b}_0 = \frac{1}{V_T}(\exp(\mathbf{p}_0/V_T))\end{aligned}\tag{5.57}$$

5.4 Computational Complexity

This section explains the computational complexity of the proposed algorithm. The computational cost involved in the proposed method comes from its two main steps. Finding system steady state solution and tracking the steady-state solutions for variations among the desired parameters. The cost spent in each step is explained below:

1. The major computational cost of the proposed algorithm is spent on finding steady-state solution of the system. There are two main sources of CPU computational cost involved here. Computational cost due to the first source is spent in the calculating the moments required for constructing the orthogonal subspace, where we require to solve a linear system of equations using $(\bar{\mathbf{Y}} + \mathbf{J}_0)$ as the coefficients matrix. This is achieved through a one-time LU factorization of the matrix $(\bar{\mathbf{Y}} + \mathbf{J}_0)$ cost of which is $O(N_{hb}^\beta)$ ($\beta \approx 1.2 - 1.5$) and forward/backward substitution for each derivative with respect to α and λ . It can be shown that $(\bar{\mathbf{Y}} + \mathbf{J}_0)$ is block diagonal with sparse diagonal blocks. Hence, its LU factorization can be accomplished easily through LU factorization of the diagonal blocks via sparse techniques [17]. The second source comes from the construction and solution of the reduced system where major cost is spent in reducing the Jacobian matrix $\mathbf{J}$. The reduced system is of size z ($z \ll N_{hb}$), hence cost involved in solving it is negligible.
2. Major computational cost in this step comes only from reducing the Jacobian matrix $\mathbf{J}$ while constructing the reduced system. CPU cost spent for solving reduced system is negligible, since its size is z ($z \ll N_{hb}$).

Note: For tracking steady-state with respect to parameters $\lambda_1, \dots, \lambda_n$, CPU cost spent comes only from step 2 each time.

5.5 Proof of Preservation of Derivatives

One of the main features of the reduced order system presented in this chapter is that it shares the first q derivatives with respect to α and λ with the original system. If the reduced system is written as

$$\mathbf{Q}_z^T \bar{\mathbf{Y}} \mathbf{Q}_z \hat{\mathbf{X}} + \mathbf{Q}_z^T \mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}}) - \mathbf{Q}_z^T [\mathbf{B}_{dc} + \alpha \mathbf{B}_{ac}] = \mathbf{0} \quad (5.58)$$

where $\bar{\mathbf{Y}} = \bar{\mathbf{G}} + j\Omega \bar{\mathbf{C}}$, we need to prove that the derivatives of $\hat{\mathbf{X}}$ with respect to α and λ s are preserved under the mapping $\mathbf{Q}_z$, i.e.

$$\begin{aligned} \mathbf{A}_{(\alpha,k)} &= \mathbf{Q}_z \hat{\mathbf{A}}_{(\alpha,k)} \\ \mathbf{A}_{(\lambda_1,k)} &= \mathbf{Q}_z \hat{\mathbf{A}}_{(\lambda_1,k)} \\ &\vdots \\ \mathbf{A}_{(\lambda_n,k)} &= \mathbf{Q}_z \hat{\mathbf{A}}_{(\lambda_n,k)}, \quad \text{where } k = 0, \dots, q-1 \end{aligned} \quad (5.59)$$

where

$$\text{ColumnSpan}\{\mathbf{Q}_z\} = \text{ColumnSpan}\{ \mathbf{M}_\alpha \quad \mathbf{M}_{\lambda_1} \quad \dots \quad \mathbf{M}_{\lambda_n} \} \quad (5.60)$$

with

$$\begin{aligned} \mathbf{M}_\alpha &= [\mathbf{A}_{(\alpha,0)}, \mathbf{A}_{(\alpha,1)}, \dots, \mathbf{A}_{(\alpha,q-1)}] \\ \mathbf{M}_{\lambda_1} &= [\mathbf{A}_{(\lambda_1,0)}, \mathbf{A}_{(\lambda_1,1)}, \dots, \mathbf{A}_{(\lambda_1,q-1)}] \\ &\vdots \\ \mathbf{M}_{\lambda_n} &= [\mathbf{A}_{(\lambda_n,0)}, \mathbf{A}_{(\lambda_n,1)}, \dots, \mathbf{A}_{(\lambda_n,q-1)}] \end{aligned} \quad (5.61)$$

5.5.1 Preservation of Derivatives A_α

In this section, proof of derivative preservation of A_α in the reduced system is given. For notational simplicity, let $A_{\alpha,i} = A_i$. The proof is given by *mathematical induction*. We begin by proving that zeroth derivative A_0 is preserved. Then we show that the first derivative $\hat{A}_1$ of $\hat{X}$ with respect to α as obtained from the reduced equations in (5.8) is equivalent to the one obtained from the original HB equations in (5.1) through mapping by Q_z . Next we proceed to show that the same relation applies to the k^{th} derivative of $\hat{X}$ provided that the previous $(k - 1)$ derivatives are preserved under the mapping Q_z . In other words, for $k = 0, \dots, q - 1$, if A_k is the k^{th} derivative of the original system, and $\hat{A}_k$ is that of the reduced system then we have

$$A_k = Q_z \hat{A}_k \quad (5.62)$$

Expanding $\hat{X}$ and $F(Q_z \hat{X})$ in (5.58) as Taylor series in the form

$$\hat{X}(\alpha) = \sum_{k=0} \hat{A}_k \alpha^k \quad (5.63a)$$

$$F(Q_z \hat{X}) = \sum_{k=0} \hat{D}_k \alpha^k \quad (5.63b)$$

Substituting from (5.63) into (5.58) results in the following

$$Q_z^T (\bar{G} + j\Omega \bar{C}) Q_z \sum_{k=0} \hat{A}_k \alpha^k + Q_z^T \sum_{k=0} \hat{D}_k \alpha^k - Q_z^T B_{dc} - \alpha Q_z^T B_{ac} = 0 \quad (5.64)$$

Equating similar powers of α in equation (5.64) results in

$$Q_z^T (\bar{G} + j\Omega \bar{C}) Q_z \hat{A}_0 + Q_z^T \hat{D}_0 = Q_z^T B_{dc} \quad (5.65a)$$

$$Q_z^T (\bar{G} + j\Omega \bar{C}) Q_z \hat{A}_1 + Q_z^T \hat{D}_1 = Q_z^T B_{ac} \quad (5.65b)$$

$$Q_z^T (\bar{G} + j\Omega \bar{C}) Q_z \hat{A}_k + Q_z^T \hat{D}_k = 0 \quad (5.65c)$$

Proof of Preservation of A_0 (Zeroth Moment)

Noting that

$$\hat{D}_0 = F(Q_z \hat{A}_0) \quad (5.66)$$

Equation (5.65a) can be written as

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) Q_z \hat{A}_0 + F(Q_z \hat{A}_0) - B_{dc} \right) = 0 \quad (5.67)$$

Since Q_z is an orthonormal basis for the subspace spanned by the columns of the matrix $[M_\alpha \ M_{\lambda_1} \ \dots \ M_{\lambda_n}]$, it can be represented in the form [45]

$$Q_z = [M_\alpha \ M_{\lambda_1} \ \dots \ M_{\lambda_n}] R^{-1} \quad (5.68)$$

where R is an upper triangular matrix. Substituting from (5.68) into (5.67) results in

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) [M_\alpha \ M_{\lambda_1} \ \dots \ M_{\lambda_n}] R^{-1} \hat{A}_0 + F \left([M_\alpha \ M_{\lambda_1} \ \dots \ M_{\lambda_n}] R^{-1} \hat{A}_0 \right) - B_{dc} \right) = 0 \quad (5.69)$$

Now we verify that $e_1 = R^{-1} \hat{A}_0$ satisfies equation (5.69), where e_r represents the r^{th} column of the identity matrix. Substituting into (5.69) yields

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) [M_\alpha \ M_{\lambda_1} \ \dots \ M_{\lambda_n}] e_1 + F([M_\alpha \ M_{\lambda_1} \ \dots \ M_{\lambda_n}] e_1) - B_{dc} \right) = 0 \quad (5.70)$$

Simplifying equation (5.70) results in

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) A_0 + F(A_0) - B_{dc} \right) = 0 \quad (5.71)$$

Comparing with (3.28) it is clear that Equation (5.71) is satisfied. Therefore we can conclude that $e_1 = R^{-1} \hat{A}_0$ is indeed a solution of equation (5.69). $\hat{A}_0$ can then be written as

$$\hat{A}_0 = R e_1 \quad (5.72)$$

Pre-multiplying by Q_z results in

$$Q_z \hat{A}_0 = Q_z R e_1 = [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] R^{-1} R e_1 = [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] e_1 = A_0 \quad (5.73)$$

Therefore the zeroth moment A_0 is preserved under the mapping Q_z .

Proof of Preservation of A_1

From equation (5.63b) $\hat{D}_1$ can be written as

$$\hat{D}_1 = \left. \frac{\partial F(Q_z \hat{X})}{\partial \alpha} \right|_{\alpha=0} = \left. \frac{\partial F(Q_z \hat{X})}{\partial (Q_z \hat{X})} \frac{\partial Q_z \hat{X}}{\partial \hat{X}} \frac{\partial \hat{X}}{\partial \alpha} \right|_{\alpha=0} \quad (5.74)$$

Using the definition in (3.30), (5.74) reduces to

$$\hat{D}_1 = T_0 Q_z \hat{A}_1 \quad (5.75)$$

Substituting (5.75) into (5.65b), we get

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) Q_z \hat{A}_1 + T_0 Q_z \hat{A}_1 - B_{ac} \right) = 0 \quad (5.76)$$

Next we substitute the value for Q_z from (5.68) to get

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] R^{-1} \hat{A}_1 + T_0 [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] R^{-1} \hat{A}_1 - B_{ac} \right) = 0 \quad (5.77)$$

Now we verify that $e_2 = R^{-1} \hat{A}_1$ satisfies equation (5.69), where e_r represents the r^{th} column of the identity matrix. Substituting into (5.77) results in

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] e_2 + T_0 [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] e_2 - B_{ac} \right) = 0 \quad (5.78)$$

Simplifying (5.78) we obtain the following

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) A_1 + T_0 A_1 - B_{ac} \right) = 0 \quad (5.79)$$

Comparing with (3.32), it is clear that equation (5.79) is satisfied. The first moment $\hat{\mathbf{A}}_1$ of the reduced system can therefore be written as

$$\hat{\mathbf{A}}_1 = \mathbf{R}\mathbf{e}_2 \quad (5.80)$$

Pre-multiplying by $\mathbf{Q}_z$ results in

$$\mathbf{Q}_z \hat{\mathbf{A}}_1 = \mathbf{Q}_z \mathbf{R}\mathbf{e}_2 = [\mathbf{M}_\alpha \ \mathbf{M}_{\lambda_1} \ \cdots \ \mathbf{M}_{\lambda_n}] \mathbf{R}^{-1} \mathbf{R}\mathbf{e}_2 = [\mathbf{M}_\alpha \ \mathbf{M}_{\lambda_1} \ \cdots \ \mathbf{M}_{\lambda_n}] \mathbf{e}_2 = \mathbf{A}_1 \quad (5.81)$$

Therefore the zeroth moment $\mathbf{A}_1$ is preserved under the mapping $\mathbf{Q}_z$.

Proof of Preservation of $\mathbf{A}_n$

In this section, a proof that the n^{th} moment $\mathbf{A}_n$ is preserved in the reduced system if the previous moments $\mathbf{A}_0, \dots, \mathbf{A}_{n-1}$ are also preserved, is given. First the matrix $\hat{\mathbf{T}}$ is defined as

$$\hat{\mathbf{T}}(\alpha) = \frac{\partial \mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}})}{\partial \hat{\mathbf{X}}} \quad (5.82)$$

The Taylor expansion of $\hat{\mathbf{T}}$ with respect to α is

$$\hat{\mathbf{T}}(\alpha) = \sum_{k=0} \hat{\mathbf{T}}_k \alpha^k \quad (5.83)$$

Recalling that $\hat{\mathbf{A}}_k$ and $\hat{\mathbf{D}}_k$ are the Taylor coefficients of $\hat{\mathbf{X}}$ and $\mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}})$ respectively as defined in equations (5.63a) and (5.63b), and using the chain rule results in

$$\frac{\partial \mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}})}{\partial \alpha} = \frac{\partial \mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}})}{\partial \hat{\mathbf{X}}} \frac{\partial \hat{\mathbf{X}}}{\partial \alpha} = \hat{\mathbf{T}}(\alpha) \frac{\partial \hat{\mathbf{X}}}{\partial \alpha} \quad (5.84)$$

or

$$\sum_{k=1} k \hat{\mathbf{D}}_k \alpha^{k-1} = \sum_{k=0} \hat{\mathbf{T}}_k \alpha^k \sum_{k=1} k \hat{\mathbf{A}}_k \alpha^{k-1} \quad (5.85)$$

Equating the coefficients of equal powers of α in (5.84) results in the following recursive relationship

$$\hat{\mathbf{D}}_k = \hat{\mathbf{T}}_0 \hat{\mathbf{A}}_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) \hat{\mathbf{T}}_j \hat{\mathbf{A}}_{k-j} \quad (5.86)$$

Using the chain rule, $\hat{T}$ can be related to the matrix T which was defined in (3.30)

$$\hat{T} = \frac{\partial F(Q_z \hat{X})}{\partial \hat{X}} = \frac{\partial F(Q_z \hat{X})}{\partial (Q_z \hat{X})} \frac{\partial (Q_z \hat{X})}{\partial \hat{X}} = T Q_z = \sum_{k=0} T_k Q_z \alpha^k \quad (5.87)$$

Substituting (5.87) into (5.86) results in

$$\hat{D}_k = T_0 Q_z \hat{A}_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) T_j Q_z \hat{A}_{k-j} \quad (5.88)$$

The recursive equation in (5.65c) for finding the n^{th} moment of the reduced system can therefore be written as

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) Q_z \hat{A}_k + T_0 Q_z \hat{A}_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) T_j Q_z \hat{A}_{k-j} \right) = 0 \quad (5.89)$$

Substituting for Q_z from (5.68) results in

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) [M_\alpha M_{\lambda_1} \cdots M_{\lambda_n}] R^{-1} \hat{A}_k + T_0 [M_\alpha M_{\lambda_1} \cdots M_{\lambda_n}] R^{-1} \hat{A}_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) T_j Q_z \hat{A}_{k-j} \right) = 0 \quad (5.90)$$

Substituting $e_{k+1} = R^{-1} \hat{A}_k$ as a solution for the above equation results in

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) [M_\alpha M_{\lambda_1} \cdots M_{\lambda_n}] e_{k+1} + T_0 [M_\alpha M_{\lambda_1} \cdots M_{\lambda_n}] e_{k+1} + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) T_j Q_z \hat{A}_{k-j} \right) = 0 \quad (5.91)$$

Assuming that the previous moments $A_0, \dots, A_{k-1}$ are preserved under the mapping Q_z or more precisely

$$A_i = Q_z \hat{A}_i; \quad i = 0, \dots, k-1 \quad (5.92)$$

Equation (5.91) can be written as

$$Q_z^T \left((\bar{G} + j\Omega \bar{C}) A_k + T_0 A_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) T_j A_{k-j} \right) = 0 \quad (5.93)$$

Comparing with (3.37), it is clear that equation n (5.93) is satisfied. The k^{th} moment $\hat{A}_k$ of the reduced system can therefore be written as

$$\hat{A}_k = R e_{k+1} \quad (5.94)$$

Pre-multiplying by Q_z results in

$$Q_z \hat{A}_k = Q_z R e_{k+1} = [M_\alpha M_{\lambda_1} \cdots M_{\lambda_n}] R^{-1} R e_{k+1} = [M_\alpha M_{\lambda_1} \cdots M_{\lambda_n}] e_{k+1} = A_k \quad (5.95)$$

Therefore the k^{th} moment A_k is preserved under the mapping Q_z if the previous moments $A_0, \dots, A_{k-1}$ are preserved. It can be therefore deduced by induction that the reduced system preserves the first q moments of the original system with respect to α under the mapping Q_z , where q is the number of moments of the original system with respect to α used in constructing Q_z .

5.5.2 Preservation of Derivatives A_λ

In this section, proof of preservation of A_λ in the reduced system is presented. The proof is based on *mathematical induction* as in previous section. Consider an arbitrary circuit parameter λ_p . Let this parameter belong to the linear part of the network. For sake of notational simplicity let $A_{\lambda_p, i} = P_i$. Now expanding Y , $\hat{X}$ and $F(Q_z \hat{X})$ in (5.58) as Taylor series with respect to λ_p at $\alpha = 0$

$$\bar{Y}(\lambda) = \sum_{k=0} \bar{Y}_k (\lambda - \lambda_{p_o})^k \quad (5.96a)$$

$$\hat{X}(\lambda) = \sum_{k=0} \hat{P}_k (\lambda - \lambda_{p_o})^k \quad (5.96b)$$

$$F(Q_z \hat{X}) = \sum_{k=0} \hat{D}_k (\lambda - \lambda_{p_o})^k \quad (5.96c)$$

Substituting from (5.96) into (5.58) results in the following

$$\mathbf{Q}_z^T \sum_{k=0} \bar{\mathbf{Y}}_k (\lambda_p - \lambda_{p_o})^k \mathbf{Q}_z \sum_{k=0} \hat{\mathbf{P}}_k (\lambda_p - \lambda_{p_o})^k + \mathbf{Q}_z^T \sum_{k=0} \hat{\mathbf{D}}_k (\lambda_p - \lambda_{p_o})^k - \mathbf{Q}_z^T \mathbf{B}_{dc} = 0 \quad (5.97)$$

Equating similar powers of λ_p in equation (5.97) results in

$$\mathbf{Q}_z^T \bar{\mathbf{Y}}_0 \mathbf{Q}_z \hat{\mathbf{P}}_0 + \mathbf{Q}_z^T \hat{\mathbf{D}}_0 = \mathbf{Q}_z^T \mathbf{B}_{dc} \quad (5.98a)$$

$$\mathbf{Q}_z^T \bar{\mathbf{Y}}_0 \mathbf{Q}_z \hat{\mathbf{P}}_k + \mathbf{Q}_z^T \hat{\mathbf{D}}_k = -\mathbf{Q}_z^T \sum_{n=0}^{k-1} \bar{\mathbf{Y}}_{k-n} \mathbf{Q}_z \hat{\mathbf{P}}_n \quad (5.98b)$$

In this section it shall be proven that the n^{th} moment $\mathbf{P}_n$ is preserved in the reduced system assuming that the previous moments $\mathbf{P}_0, \dots, \mathbf{P}_{n-1}$ are proved to be preserved using mathematical induction. Now, for $k = 0, \dots, q-1$, if $\mathbf{P}_k$ is the k^{th} derivative of the original system, and $\hat{\mathbf{P}}_k$ is that of the reduced system then we have

$$\mathbf{P}_k = \mathbf{Q}_z \hat{\mathbf{P}}_k \quad (5.99)$$

To proceed with the proof, let the matrix $\hat{\mathbf{J}}$ be defined as

$$\hat{\mathbf{J}}(\lambda_p) = \frac{\partial \mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}})}{\partial \hat{\mathbf{X}}} \quad (5.100)$$

The Taylor expansion of $\hat{\mathbf{J}}$ with respect to λ_p is

$$\hat{\mathbf{J}}(\lambda_p) = \sum_{k=0} \hat{\mathbf{J}}_k (\lambda_p - \lambda_{p_o})^k \quad (5.101)$$

Recalling that $\hat{\mathbf{P}}_k$ and $\hat{\mathbf{D}}_k$ are the Taylor coefficients of $\hat{\mathbf{X}}$ and $\mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}})$ respectively as defined in equations (5.96b) and (5.96c), and using the chain rule results in

$$\frac{\partial \mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}})}{\partial \lambda_p} = \frac{\partial \mathbf{F}(\mathbf{Q}_z \hat{\mathbf{X}})}{\partial \hat{\mathbf{X}}} \frac{\partial \hat{\mathbf{X}}}{\partial \lambda_p} = \hat{\mathbf{J}}(\lambda_p) \frac{\partial \hat{\mathbf{X}}}{\partial \lambda_p} \quad (5.102)$$

or

$$\sum_{k=1} k \hat{\mathbf{D}}_k (\lambda_p - \lambda_{p_o})^{k-1} = \sum_{k=0} \hat{\mathbf{J}}_k (\lambda_p - \lambda_{p_o})^k \sum_{k=1} k \hat{\mathbf{P}}_k (\lambda_p - \lambda_{p_o})^{k-1} \quad (5.103)$$

Equating the coefficients of equal powers of λ_p in (5.103) results in the following recursive relationship

$$\hat{D}_k = \hat{J}_0 \hat{P}_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) \hat{J}_j \hat{P}_{k-j} \quad (5.104)$$

Using the chain rule, $\hat{J}$ can be related to the matrix J which was defined in (3.30)

$$\hat{J} = \frac{\partial F(Q_z \hat{X})}{\partial \hat{X}} = \frac{\partial F(Q_z \hat{X})}{\partial (Q_z \hat{X})} \frac{\partial (Q_z \hat{X})}{\partial \hat{X}} = J Q_z = \sum_{k=0} J_k Q_z (\lambda_p - \lambda_{p_o})^k \quad (5.105)$$

Substituting (5.105) into (5.104) results in

$$\hat{D}_k = J_0 Q_z \hat{P}_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) J_j Q_z \hat{P}_{k-j} \quad (5.106)$$

The recursive equation in (5.98b) for finding the k^{th} moment of the reduced system can therefore be written as

$$Q_z^T \left(\bar{Y}_o Q_z \hat{P}_k + \sum_{n=0}^{k-1} \bar{Y}_{k-n} Q_z \hat{P}_n + J_0 Q_z \hat{P}_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) J_j Q_z \hat{P}_{k-j} \right) = 0 \quad (5.107)$$

Substituting for Q_z from (5.60) results in

$$Q_z^T \left(\bar{Y}_o [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] R^{-1} \hat{P}_k + \sum_{n=0}^{k-1} \bar{Y}_{k-n} Q_z \hat{P}_n \right. \\ \left. J_0 [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] R^{-1} \hat{P}_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) J_j Q_z \hat{P}_{k-j} \right) = 0 \quad (5.108)$$

Substituting $e_s = R^{-1} \hat{P}_k$ (where s is the index corresponding to P_k , given as $s = p \times q + k$) as a solution for the above equation results in

$$Q_z^T \left(\bar{Y}_o [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] e_s + \sum_{n=0}^{k-1} \bar{Y}_{k-n} Q_z \hat{P}_n \right. \\ \left. J_0 [M_\alpha \ M_{\lambda_1} \ \cdots \ M_{\lambda_n}] e_s + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) J_j Q_z \hat{P}_{k-j} \right) = 0 \quad (5.109)$$

Assuming that the previous moments $P_0, \dots, P_{k-1}$ are preserved under the mapping Q_z or more precisely

$$P_i = Q_z \hat{P}_i; \quad i = 0, \dots, k-1 \quad (5.110)$$

Equation (5.109) can be written as

$$Q_z^T \left(\bar{Y}_o P_k + \sum_{n=0}^{k-1} \bar{Y}_{k-n} P_n + J_0 P_k + \frac{1}{k} \sum_{j=1}^{k-1} (k-j) J_j P_{k-j} \right) = 0 \quad (5.111)$$

Comparing with (5.32), it is clear that equation (5.111) is satisfied. The k^{th} moment $\hat{P}_k$ of the reduced system can therefore be written as

$$\hat{P}_k = R e_s \quad (5.112)$$

Pre-multiplying by Q_z results in

$$Q_z \hat{P}_k = Q_z R e_s = [M_\alpha M_{\lambda_1} \cdots M_{\lambda_n}] R^{-1} R e_s = [M_\alpha M_{\lambda_1} \cdots M_{\lambda_n}] e_s = P_k \quad (5.113)$$

Therefore the k^{th} moment P_k is preserved under the mapping Q_z if the previous moments $P_0, \dots, P_{k-1}$ are preserved. It can be therefore deduced by induction that the reduced system preserves the first q moments of the original system with respect to λ_p under the mapping Q_z , where q is the number of moments of the original system with respect to λ_p used in constructing Q_z . Similar proof for preservation of derivatives A_λ in the reduced system when λ belongs to the nonlinear part of the network can be deduced.

5.6 Numerical Examples

For the validation of the proposed algorithm, a RC coupled amplifier (Figure 5.2) is considered. The input to the amplifier is a sinusoidal signal at $f = 20.7\text{MHz}$. To perform steady-state analysis using harmonic balance approach it required 16 harmonics for each

nodal variable, resulting in system of size 429. The proposed method reduces this problem size to 109. An orthogonal subspace is first formed using the Taylor series coefficients of the solution with respect to α (64 in number, at $\alpha = 0$), 15 coefficients with respect to the diode¹ saturation current I_s (at $I_s = 1 \times 10^{-15} A$), 15 coefficients with respect to V_T (Thermal voltage) of the diode (at $V_T = 25mV$) and 15 coefficients with respect to the resistor R at the emitter (at $R = 1K\Omega$). This orthogonal subspace is used to construct a parameterized reduced system of size 109 from the original system of size 429. The reduced system of size 109 is solved to obtain the original system steady-state. Now for any variations in V_T , I_s , and R , the new steady-state solution is obtained by solving the reduced system using the steady-state solution of the original system as initial guess. In Figure 5.3 the third harmonic content at the output node, for variation in the saturation current I_s from $1 \times 10^{-15} A$ to $100 \times 10^{-15} A$ is shown. The figure also compares the solutions obtained from the reduced and the original system. Figure 5.4 and Figure 5.5 shows the 3rd harmonic content variation with respect to V_T of the diode and the resistor R obtained from the original and reduced systems. The corresponding maximum relative errors (percentage) between the solutions from the proposed and conventional HB method is given in Table 5.2.

Table 5.2: Error from the proposed method

Parameter (λ)	Maximum Relative Error (%)
I_s (Diode Saturation Current)	2.0042×10^{-7}
V_T (Diode Thermal Voltage)	1.2899×10^{-7}
R (Resistor)	1.0668×10^{-9}

¹emitter-base junction diode and collector-base junction diode

Figure 5.2: A RC coupled amplifier

Figure 5.3: Tracking third harmonic content for variation in diode saturation current (I_s)

Figure 5.4: Third harmonic content variation with V_T of the diode

Figure 5.5: Third harmonic content variation with emitter resistor R

Chapter 6

Projection Based Techniques for Multi-Tone Circuits

6.1 Introduction

Steady-state analysis of large nonlinear circuits is becoming a key simulation requirement due to the rapid growth of RF silicon integrated circuits. These circuits are stimulated with either single input or multi-inputs such as in the case of mixer circuits. Also sometimes the driving signal itself is a multi-tone signal. For example, in the case of IP3 analysis, two input tones are used in order to measure the third order intermodulation product. When more than one input tones are used, the tones are in general non-commensurate (i.e. not multiples of each other) and the input cannot be written as a Fourier series, and is referred to as quasi-periodic. In this chapter, the circuit reduction approach for multi-tone circuits is described [8].

6.1.1 Multi-Tone HB equations

For the case of non-commensurate multi-tone inputs, the steady-state response consists of frequencies at the harmonics of the input tones, as well as at their intermodulation products. For example, if two tones are ω_1 and ω_2 with $\omega_1 > \omega_2$, the steady-state response will have tones at dc, ω_1 , $2\omega_1$, $\dots$, $2\omega_2$, $\dots$, $2\omega_1 - \omega_2$, $2\omega_2 - \omega_1$, $\dots$ etc. In general, the steady-state response can therefore be represented as in (2.22), where the frequencies ω_k represent the harmonics and intermodulation products of the input tones. Substituting (2.22) into MNA equations of the circuit in (2.1) and equating the coefficients of the sines and cosines at the same frequencies results in the HB equations of the form

$$\bar{\mathbf{G}}\bar{\mathbf{X}} + j\Omega\bar{\mathbf{C}}\bar{\mathbf{X}} + \mathbf{F}(\bar{\mathbf{X}}) = \bar{\mathbf{B}}_{dc} + \bar{\mathbf{B}}_{h1} + \bar{\mathbf{B}}_{h2} + \dots \quad (6.1)$$

where $\mathbf{X} \in \mathbb{R}^{N_{hb}}$ is a vector containing the unknown coefficients of the sines and cosines representing $\mathbf{X}(t)$, $\mathbf{B}_{dc} \in \mathbb{R}^{N_{hb}}$ is a vector of the Fourier coefficients of the dc input sources, $\mathbf{B}_{h1} \in \mathbb{R}^{N_{hb}}$, $\mathbf{B}_{h2} \in \mathbb{R}^{N_{hb}} \dots$ are vectors representing the various input tones, $\bar{\mathbf{G}}$, $\bar{\mathbf{C}}$, Ω and $\mathbf{F}$ are defined as in (2.24), except that in this case, the frequencies corresponding to the coefficients in $\mathbf{X}$, $\mathbf{F}(\mathbf{X})$ and Ω are non-commensurate. This causes difficulties when attempting to use FFT algorithm to evaluate the Fourier coefficients of the nonlinear vector $\mathbf{F}(\mathbf{X})$, as was done for the case of single tone inputs. The same difficulty also applies to the calculation of the Jacobian. To tackle this problem, little modification on FFT is done [25] that would make (6.1) as if they represent a periodic signal, as opposed to a quasi-periodic signal from the perspective of the FFT algorithm. This technique is briefly discussed next.

6.1.2 FFT for Almost Periodic Signals

In the case of single tone, applying the FFT algorithm is a straight forward procedure. However, the input of RF circuits is in general composed of multi-tone signals with non-commensurate frequencies. For example, one can have inputs at 900MHz and 910MHz. In such cases the period of the combined signal is very large compared to that of the input tones. Such signals are referred to as “almost periodic”. In such cases applying the FFT algorithm in a brute force way is both inaccurate and inefficient (the signal would have to be sampled over long combined period). There are number of methods in the literature that can be used to solve this problem [52], [53]. The approach used here is one presented in [25], and is simple, efficient, and suitable for harmonic balance analysis.

The approach used for handling circuits with almost periodic inputs relies on two main properties of the harmonic balance problem. The first property is that the nonlinear vector $\mathbf{f}(x(t))$ in the MNA equation in (2.1) is memoryless. The second property is that, when evaluating $\mathbf{F}(x)$ in the harmonic balance algorithm, the time domain waveforms are of no importance, and only the frequency domain Fourier coefficients are of interest. This approach is best explained using an example. Consider the memoryless nonlinear function

$$\mathbf{f}(x(t)) = \mathbf{x}(t)^2 \quad (6.2)$$

For this example, we assume

$$\mathbf{x}(t) = a_1 \cos(\omega_1 t) + a_2 \cos(\omega_2 t) \quad (6.3)$$

With $\omega_1 > \omega_2$. This simple problem is identical to what is encountered in harmonic balance algorithm. The Fourier coefficients of $\mathbf{x}(t)$ are known and the task is to find the Fourier coefficients of $\mathbf{f}(x)$. In this case, the solution can be found analytically using the trigonometric identities.

Table 6.1: Amplitude of the harmonics

Frequency	Amplitude
DC	$\frac{a_1}{2} + \frac{a_2}{2}$
$\omega_1 - \omega_2$	0
ω_2	0
$2\omega_1$	$\frac{a_1}{2}$
$\omega_1 + \omega_2$	$\frac{a_1 a_2}{2}$
$2\omega_2$	$\frac{a_2}{2}$

$$\begin{aligned}
f(x)(t) &= (a_1 \cos(\omega_1 t) + a_2 \cos(\omega_2 t))^2 \\
&= a_1^2 \cos^2(\omega_1 t) + a_2^2 \cos^2(\omega_2 t) + a_1 a_2 \cos(\omega_1 t) \cos(\omega_2 t) \\
&= \frac{a_1}{2} + \frac{a_1}{2} \cos(2\omega_1 t) + \frac{a_2}{2} + \frac{a_2}{2} \cos(2\omega_2 t) \\
&+ \frac{a_1 a_2}{2} \cos((\omega_1 + \omega_2)t) + \frac{a_1 a_2}{2} \cos((\omega_1 - \omega_2)t)
\end{aligned} \tag{6.4}$$

By examining the equation, we can determine that $f(x)$ has 5 different harmonics as listed in the Table 6.1 . The key conclusion to be drawn from this exercise is that the amplitude of the harmonics is independent of the frequencies ω_1 and ω_2 . Of course these frequencies will affect the time domain waveforms, but if our interest is simply to determine Fourier coefficients we can choose any frequency and still obtain the same result. Let us consider an example where the values of are given as

$$\omega_1 = 900 \text{ MHz}; \quad \omega_2 = 910 \text{ MHz} \tag{6.5}$$

These two frequencies are not commensurate and the resulting waveform is quasi periodic. However as we have seen earlier, modifying values of the frequencies does not affect the

Table 6.2: Mapping of the harmonics

Frequency	Actual Frequency	Mapped Frequency	Amplitude
DC	DC	DC	$\frac{a_1}{2} + \frac{a_2}{2}$
$\omega_1 - \omega_2$	10MHz	300MHz	$\frac{a_1 a_2}{2}$
ω_1	900MHz	600MHz	0
ω_2	910MHz	900MHz	0
$2\omega_1$	1800MHz	1200MHz	$\frac{a_1}{2}$
$\omega_1 + \omega_2$	1810MHz	1500MHz	$\frac{a_1 a_2}{2}$
$2\omega_2$	1820MHz	1800MHz	$\frac{a_2}{2}$

Fourier coefficients of the result. So let us consider the following mapping

$$\omega_1 \leftarrow 600 \text{ MHz}; \quad \omega_2 \leftarrow 900 \text{ MHz} \quad (6.6)$$

The harmonics of $f(x)$ obtained from the original definition of ω_1 and ω_2 in (6.5) and from the mapping in (6.6) are shown in Table 6.2. As can be seen from table using the mapped frequencies $f(x)$ becomes periodic signal with fundamental frequency 300 MHz. The FFT algorithm can therefore be applied to the mapped frequencies and the resulting amplitudes are then mapped back to the original frequencies. Practically this approach simply results in a reordering of harmonics and requires no additional CPU expense. A systematic way of finding the appropriate frequency mapping is given in [25].

Using this approach multi-tone circuits can be handled without any significant modification to the harmonic balance analysis. The proposed circuit reduction method described in section 5.2 is therefore applicable to the case of multi-tone inputs. The steps are summarized in Figure 6.1. Next, numerical examples are presented to validate the efficiency and accuracy of the proposed method.

1. Map the system with non-commensurate tones (multi-tone system) to a system with commensurate tones (single-tone system) as explained earlier.
2. Solve the single-tone system using the proposed method: let the resulting single-tone system be represented by (2.24).
 - (a) Embed α into (2.24) to obtain (2.57).
 - (b) Find multi-dimensional moments with respect to parameters of interest and construct orthogonal subspace $\mathbf{Q}_z$ (5.2).
 - (c) Construct parameterized reduced system (5.8) using $\mathbf{Q}_z$.
 - (d) Solve the reduced system using NR iterations (5.10) for $\alpha = 0 \rightarrow 1$.
 - (e) Map the solution back to the original space, i.e. $\mathbf{X} = \mathbf{Q}_z \hat{\mathbf{X}}$.
3. Map the solution obtained from the single-tone system ($\mathbf{X}$) back to the multi-tone system.

Figure 6.1: Steps for applying proposed method to multi-tone circuits

6.2 Numerical Examples

6.2.1 Example 1

In this example, the mixer circuit shown in Figure 6.2 is considered. It converts an input signal of 250MHz to 50MHz output frequency, by mixing it with a signal of 300MHz. This circuit is excited with two tones, at $f_1 = 250\text{MHz}$ and $f_2 = 300\text{MHz}$. The number of harmonics needed for obtaining steady-state solution using harmonic balance is $H = 42$, and therefore requires solving a system with $(20\text{variables} \times (2 \times 42 + 1) = 1700)$ nonlinear equations. An orthogonal subspace is formed using the Taylor series coefficients of solution of the system with respect to α (100 in number, at $\alpha = 0$), 15 coefficients with respect to transistor junction area (at area = 1mm), 15 coefficients with respect to capacitor C (at $C = 5\text{pF}$) and 15 coefficients with respect to V_T of the BJT (at $V_T = 25\text{mV}$). Using the orthogonal subspace on the original system of size 1700, a reduced system of size 145 is constructed. The system steady-state was obtained by solving the reduced system instead of original system. Now the steady-state variation for changes in V_T , capacitor and junction area is tracked by solving the reduced system. The transistor junction area is varied from 1mm to 50mm , and the third order intermodulation term ($2f_1 - f_2$) at the output node obtained by solving system of size 145 and original system of size 1700 are compared in Figure 6.3. V_T of the transistor is varied from 23mV to 27mV , and the third order intermodulation term ($2f_1 - f_2$) from reduced and original is compared in Figure 6.4. Figure 6.5 plots the third order intermodulation term obtained from both the systems when the capacitor value is varied from 5pF to 100pF .

Figure 6.2: A mixer circuit.

Figure 6.3: Third-order intermodulation variation with transistor junction area

Figure 6.4: Third-order intermodulation variation with V_T

Figure 6.5: Third-order intermodulation variation with capacitor C

An estimate of the computational cost spent by the proposed and conventional HB method is given in Table 6.3. The first row shows the cost spent by the conventional HB method for obtaining the steady-state solution. The HB method operates on the original system of size 1700 by sweeping α from $0 \rightarrow 1$ and updating NR equations at each power step until convergence. The number of incremental steps of α used in this example is 20 and at each step, the number of NR iterations is assumed to be 5. In this case, to obtain the steady-state solution we require 100 LU decompositions and 100 F/B substitutions of system of size 1700. The second row shows the cost spent by the proposed method for obtaining the steady-state solution. The proposed method first computes the moments required for reducing the original system at a cost of 1LU decomposition, 99 F/B substitutions (from (3.37), for obtaining 100 moments w.r.t α , note that the first moment is the DC solution, which is already available from (3.28)) and 15×3 F/Bs for moments w.r.t λ s (from (5.32) and (5.48)), resulting in a total cost of 1LU+144F/B of system of size 1700 (original system). The advantage is that the matrix is sparse and is block diagonal in structure; hence reducing the computational effort for LU decomposition and F/B substitutions. The proposed method then solves the reduced system. The number of incremental steps of α used in this example is 20 and at each step, assuming the number of NR iterations to be 5, the steady-state solution is obtained at the cost of 100LU+100F/B of system of size 145. For tracking the steady-state when any parameter is varied, the HB repeats the steady-state analysis procedure incurring the same cost as in first row, i.e. 100LU+100F/B of system of size 1700, shown in the third row. On the other hand, the proposed method tracks the steady-state solution without repeating the whole steady-state analysis procedure. The cost is spent on updating NR equations till convergence, incurring a cost of 5LU+5F/B on the reduced system (of size 145). This is listed in the last row. In summary, the proposed method incurs the cost of 1LU+144F/B of the large system (of size 1700) and 100LU+100F/B of the reduced system (of size 145) for the steady-state analysis, and the steady-state is tracked

each time for any variations in the parameter value at the cost of $5\text{LU}+5\text{F/B}$ of the reduced system (of size 145), for this example.

Table 6.3: Computational cost comparison of the proposed method v/s HB

	Jacobian Size	NR iteration for each step	Total num of steps	Total num of LU	Total num of F/B	Total Cost (LU + F/B)
Steady-State Analysis						
HB	1700	5	20	100	100	100 LU + 100 F/B (of the original)
Proposed	1700	–	–	1	99 (for moments w.r.t α) + 15 + 15 + 15 (for moments w.r.t λ_s)	1 LU + 144 F/B (of the original)
	145	5	20	100	100	100 LU + 100 F/B (of the reduced)
Tracking Steady-State Once (w.r.t λ_i)						
HB	1700	5	20	100	100	100 LU + 100 F/B (of the original)
Proposed	145	5	–	5	5	5 LU + 5 F/B (of the reduced)

6.2.2 Example 2

As a second example, an LNA (Figure 6.6) is considered. The LNA was excited with two tones at 5.1GHz and 5.2GHz. To solve for the steady-state using harmonic balance, 42 harmonics were needed to get good approximation. This resulted in a circuit with 2210 nonlinear equations. An orthogonal subspace is formed using the Taylor series coefficients of the solution of the network with respect to α (150 in number, at $\alpha = 0$), 15 coefficients with respect to transistor junction area (at area = $10\mu m$) and 15 coefficients with respect to inductor L_e (at $L_e = 300pH$). A parameterized system of size 180 is constructed from the original system of size 2210 using the orthogonal subspace. The system steady-state was obtained by solving the reduced system of size 180. For any variation in junction area and L_e , the reduced system is solved for obtaining the steady-state solution. In Figure 6.7, the transistor junction area was changed from $10\mu m$ to $90\mu m$ and the third order intermodulation term ($2f_1 - f_2$) obtained from the reduced system is compared with the solution from the original system. Figure 6.8 compares the third order intermodulation term ($2f_1 - f_2$) for variations in the inductor value L_e obtained from both the systems.

Figure 6.6: A LNA circuit.

Figure 6.7: Third-order intermodulation variation with transistor junction area.

Figure 6.8: Third-order intermodulation variation with inductor value L_e .

6.2.3 Example 3

In this example, a QUAD-mixer circuit of Figure 6.9 was considered. This circuit has two tones, radio frequency (RF) at 2GHz and local oscillator (LO) at 2.1GHz. The output of the mixer is a signal of difference between the two tones i.e. 100MHz. Harmonic balance resulted in system of size 3995, with 42 harmonics for each node. An orthogonal subspace is formed using the Taylors series coefficients of the solution of the system with respect to α (total of 400, at $\alpha = 0$), 15 coefficients with respect to transistor junction area (at area = $1\mu m$) and 15 coefficients with respect to resistor R_E (at $R_E = 100\Omega$) resulting in system of size 430. This orthogonal subspace was used to create a parameterized reduced system of size 430 from the original system of size 3995. The system steady-state was obtained by solving the reduced system of size 430 instead of the original system of size 3995. Now for any variations in R_E and junction area, the reduced system is solved for obtaining the steady-state solutions. Figure 6.10 shows variation in the third order intermodulation term ($2f_1 - f_2$) of the output node voltage when R_E is varied from 75Ω to 250Ω . Also, the figure provides a comparison between the solutions obtained from the reduced system and original system. Figure 6.11 compares ($2f_1 - f_2$) term obtained from the reduced and original system for variation in junction area from $1\mu m$ to $100\mu m$.

Figure 6.9: A QUAD mixer circuit.

Figure 6.10: Third-order intermodulation variation with respect to R_E

Figure 6.11: Third-order intermodulation variation with respect to transistor junction area

Chapter 7

Summary and Future Work

This thesis presents efficient algorithms for performing small scale sensitivity analysis and large change sensitivity analysis. The algorithms are based on Krylov subspace technique for circuit reduction. The main idea underlying the algorithm is to project a large nonlinear system of equations onto a smaller subspace. The reduced system is then solved and solution is mapped back to the original system. Since the analysis is done in smaller subspace it results in significant reduction the CPU cost.

For performing small scale sensitivity analysis we require steady state solution of the system and the adjoint system solution. The projection based reduction technique (section 3.3.2) computes the steady-state of the nonlinear circuits at very low computational cost. Proposed algorithm (chapter 4) provided an efficient way of obtaining the adjoint system solution taking advantage of the projection technique for obtaining the steady-state of the system. The idea underlying the concept is projecting the adjoint system of equations onto a subspace of smaller size using the orthogonal subspace constructed by the Taylors series coefficient of the adjoint system with respect to the continuation parameter α at $\alpha = 0$. The reduced system and the original adjoint system share the first few Taylors series coefficients with respect to α . The adjoint system solution is found by obtaining the reduced system

solution and then mapping it back to the original adjoint system. Using the steady-state solution obtained from the projection technique and the adjoint solution from the proposed method the small scale sensitivity analysis is performed at very low computational cost. Accuracy of the sensitivity of the output node voltage with respect to linear as well as nonlinear parameters of the circuit obtained from the proposed method was comparable to the solution obtained by solving the original set of HB equations.

For performing large change sensitivity analysis the steady-state solution is tracked for variations in the circuit parameters. The projection based reduction technique, section 3.3.2, provided a significant reduction in computational cost in obtaining the steady-state solution. But for tracking the steady-state with the change in parameter value we need to repeat the whole procedure every time. Proposed algorithm avoids this need for performing repeated steady-state analysis for every change in the parameter value. In the new algorithm, first step is constructing an orthogonal subspace using the Taylors series coefficients of the system with respect to continuation parameter α and circuit parameters of interest (parameters which are varied). This orthogonal subspace is used to construct a parameterized reduced system from the original large HB system. The reduced system size is very small compared to the original one and also these two systems share in common first few Taylors series coefficients with respect to α and parameters of interest. Once the parameterized reduced system is formed, the steady-state solution of the original system is found by solving the reduced system using the continuation method i.e. tracing α from 0 to 1 in the reduced space and mapping the solution of the reduced system back to the original system. Next, for any variations in parameters of interest the new steady-state solution is obtained by updating the NR equations using the steady-state solution of the unperturbed system as initial guess. The significant advantage of the proposed algorithm comes from the fact that for any variation in the parameters of interest, the new steady-state solution is found directly updating the NR equations without having to go through the continuation

process. It is due to the fact that the initial guess used for updating the NR equations, i.e. the steady-state solution of the unperturbed system, is very close to the perturbed system solution guaranteeing convergence of NR equations.

Also it was shown that proposed algorithm can be applied to a system with a set of non-commensurate frequencies (multi-tone systems). For this, the set of non-commensurate tones was mapped to a set of commensurate tones which is solved using the proposed algorithm. This solution was then mapped back to the multi-tone system. Proposed algorithm was tested on circuits with single and multi-tone inputs, and steady-state solution was tracked for variations in linear as well as nonlinear parameters.

7.1 Future Work

1. The proposed model reduction technique can be extended to autonomous circuits such as oscillators. In such circuits, the frequency of operation is also an unknown variable. One way of applying the proposed technique is to add the frequency of operation of the circuit as unknown variable along with the circuit node voltages and currents. Proposed method can then be applied to the system.
2. The proposed reduction technique can be applied to solve digital systems. Envelop transient technique is used for solving digital systems. The high frequency component of the solution is obtained by solving the system in frequency domain (using HB method) and low frequency envelop is obtained by solving the system in time domain (using conventional LMS approaches). Proposed method can be applied to the envelop technique to obtain high frequency component of the solution.
3. The proposed model reduction method uses continuation methods as a vehicle to enable the reduction (RF input power was used as the continuation parameter). In-

vestigation shows that for some types of circuits it may be beneficial to use other parameters [54] in order to do reduction. Further research into choosing the continuation parameters used for reduction would provide a significant contribution to this area.

4. Circuit optimization is an important step in design process. Circuits have to be solved repeatedly by varying circuit parameters so that the circuit response meets the specifications applied on it. An important application of the proposed method is the problem of circuit optimization.

Bibliography

- [1] R. Achar and M. Nakhla, "Simulation of high-speed interconnects," *Proceedings IEEE*, vol. 89, no. 5, pp. 693–728, May 2001.
- [2] J. R. Long and M. A. Copeland, "A 1.9 GHz low-voltage silicon bipolar receiver front-end for wireless personal communications systems," *IEEE J. Solid-State Circuits*, vol. 30, no. 12, pp. 1438–1448, Dec. 1995.
- [3] J. W. M. Rogers, J. Macedo, and C. Plett, "A completely integrated 1.9 GHz receiver front-end with monolithic image-reject filter and VCO," *IEEE Trans. Microwave Theory Tech.*, vol. 50, no. 1, pp. 210–336, Jan. 2002.
- [4] B. Razavi, "CMOS technology characterization for analog and RF design," *IEEE J. Solid-State Circuits*, vol. 34, no. 3, pp. 268–276, Mar. 1999.
- [5] E. Gad, R. Khazaka, M. Nakhla, and R. Griffith, "A circuit reduction technique for finding the steady-state solution of nonlinear circuits," *IEEE Trans. Microwave Theory Tech.*, vol. 48, no. 12, pp. 2389–2396, Dec. 2000.
- [6] P. Pai, E. Gad, R. Achar, R. Khazaka, and M. Nakhla, "A projection-based reduction approach to computing sensitivity of steady-state response of nonlinear circuits," *INFORMS Journal on Computing*, 2004.

- [7] P. Pai, E. Gad, R. Achar, R. Khazaka, and M. Nakhla, "Computing large-change sensitivity of periodic responses of nonlinear circuits using reduction techniques," *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. V333–V335, May 2004.
- [8] P. Pai, E. Gad, R. Achar, R. Khazaka, and M. Nakhla, "Parametric circuit reduction for steady-state simulation with multi-tone inputs," *IEEE International Microwave Symposium (IMS)*, pp. 937–940, June 2004.
- [9] T. J. Aprille, Jr. and T. N. Trick, "A computer algorithm to determine the steady-state response of nonlinear oscillators," *IEEE Trans. Circuit Theory*, vol. 19, no. 4, pp. 354–360, July 1972.
- [10] T. J. Aprille, Jr. and T. N. Trick, "Steady-state analysis of nonlinear circuits with periodic inputs," *Proceedings IEEE*, vol. 60, no. 1, pp. 108–114, Jan. 1972.
- [11] S. M. Roberts and J. S. Shipman, *Two-Point Boundary Value Problems*, Elsevier, New York, 1972.
- [12] K. S. Kundert, J. White, and A. Sangiovanni-Vincentelli, "A mixed frequency-time approach for distortion analysis of switching filter circuits," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 443–451, Apr. 1989.
- [13] J. Roychowdhury, "Analyzing circuits with widely separated time scales using numerical pde methods," *IEEE Trans. Circuits Syst. I*, vol. 48, pp. 578–594, May 2001.
- [14] D. Feng, J. Phillips, K. Nabors, K. Kundert, and J. White, "Efficient computation of quasi-periodic circuit operating conditions via a mixed frequency/time approach," in *Proc. IEEE/ACM Design Automation Conf.*, Jun. 1999, pp. 635–640.

- [15] J. Roychowdhury, "Efficient methods for simulating highly nonlinear multi-rate circuits," in *Proc. IEEE/ACM Design Automation Conf.*, 1997, pp. 269–274.
- [16] C. W. Ho, A. E. Ruehli, and P. A. Brennan, "The modified nodal approach to network analysis," *IEEE Trans. Circuits Syst.*, vol. 22, no. 6, pp. 504–509, June 1975.
- [17] J. Vlach and K. Singhal, *Computer Methods for Circuit Analysis and Design*, Van Norstrand Reinolds, New York, 1983.
- [18] J. W. Demmel, *Applied Numerical Linear Algebra*, SIAM Publishers, Philadelphia, PA, 1997.
- [19] J. D. Lambert, *Computational Methods in Ordinary Differential Equations*, Introductory mathematics for scientists and engineers. London Wiley, 1973.
- [20] J. M. Ortega, *Iterative Solution of Nonlinear Equations in Several Variables*, Computer science and applied mathematics. New York: Academic Press, 1970.
- [21] K. Mayaram, D. C. Lee, S. Moinian, D. Rich, and J. Roychowdhury, "Overview of computer-aided analysis tools for RFIC simulation: Algorithms, features, and limitations," in *Proc. IEEE Custom Integrated Circuits Conf.*, 1997, pp. 505–512.
- [22] K. S. Kundert, "Accurate fourier analysis for circuit simulators," in *Proc. IEEE Custom Integrated Circuits Conf.*, May 1994, pp. 25–28.
- [23] S. Skelboe, "Conditions for quadratic convergence of quick periodic steady-state methods," *IEEE Trans. Circuits Syst.*, vol. 29, no. 4, pp. 234–239, Apr. 1982.
- [24] S. Skelboe, "Computation of the periodic steady-state response of nonlinear networks by extrapolation methods," *IEEE Trans. Circuits Syst.*, vol. 27, no. 3, pp. 161–175, Mar. 1980.

- [25] K. S. Kundert, J. K. White, and A. Sangiovanni-Vincentelli, *Steady-State Methods for Simulating Analog and Microwave Circuits*, Kluwer Academic, Boston, 1990.
- [26] M. S. Nakhla and J. Vlach, "A piecewise harmonic-balance technique for determination of periodic response of nonlinear systems," *IEEE Trans. Circuits Syst.*, vol. 23, no. 2, pp. 85–91, Feb. 1976.
- [27] K. Gopal, M. S. Nakhla, K. Singhal, and J. Vlach, "Distortion analysis of transistor networks," *IEEE Trans. Circuits Syst.*, vol. 25, no. 2, pp. 99–106, Feb. 1978.
- [28] R. J. Gilmore and F. J. Rosenbaum, "Modeling of nonlinear distortion in GaAs MES-FETs," in *IEEE MTT-S Intl. Microwave Symp. Dig.*, May 1984, pp. 430–431.
- [29] V. Rizzoli and A. Neri, "State of the art and present trends in nonlinear microwave CAD techniques," *IEEE Trans. Microwave Theory Tech.*, vol. 36, no. 2, pp. 343–365, Feb. 1988.
- [30] K. Kundert and A. Sangiovanni-Vincentelli, "Simulation of nonlinear circuits in the frequency domain," *IEEE Trans. Computer-Aided Design*, vol. 5, no. 4, pp. 521–535, Oct. 1986.
- [31] E. Chiprout and M. Nakhla, "Analysis of interconnect networks using complex frequency hopping (CFH)," *IEEE Trans. Computer-Aided Design*, vol. 14, pp. 186–200, Feb. 1995.
- [32] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design*, vol. 9, pp. 352–366, Apr. 1990.
- [33] T. K. Tang, M. S. Nakhla, and R. Griffith, "Analysis of lossy multiconductor transmission lines using the asymptotic waveform evaluation technique," *IEEE Trans. Microwave Theory Tech.*, vol. 39, no. 12, pp. 2107–2116, Jul. 1991.

- [34] T. Tang and M. Nakhla, "Analysis of high-speed VLSI interconnects using asymptotic waveform evaluation," *IEEE Trans. Computer-Aided Design*, pp. 2107–2116, Mar. 1992.
- [35] N. W. McLachlan, *Laplace Transforms and Their Applications to Differential Equations*, New York, Dover, 1962.
- [36] W. E. Arnoldi, "The principle of minimized iteration in the solution of the matrix eigenvalue problem," *Quart. Appl. Math.*, vol. 9, pp. 17–29, 1951.
- [37] C. Lanczos, "Solution of systems of linear equations by minimized iterations," *J. Research Nat'l Bureau of Standards*, vol. 49, pp. 33–53, 1952.
- [38] A. Dounavis, E. Gad, and M. Nakhla, "Passive model-reduction of multiport distributed networks including frequency dependent parameters," *IEEE Trans. Microwave Theory Tech.*, pp. 2325–2334, Dec. 2000.
- [39] R. W. Freund and P. Feldmann, "Small-signal circuit analysis and sensitivity computations with the PVL algorithm," *IEEE Trans. Circuits Syst.*, vol. 43, no. 8, pp. 577–585, Aug. 1996.
- [40] A. Odabasioglu, M. Celik, and L. T. Pileggi, "PRIMA: passive reduced-order interconnect macromodeling algorithm," *IEEE Trans. Computer-Aided Design*, vol. 17, no. 8, pp. 645–654, Aug. 1998.
- [41] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWD Publishing Company, Boston, 1996.
- [42] K. J. Kerns and A. T. Yang, "Preservation of passivity during RLC network network reduction via split congruence transformations," *IEEE Trans. Computer-Aided Design*, vol. 17, pp. 582–591, July 1998.

- [43] M. Kamon, F. Wang, and J. White, "Generating nearly optimally compact models from Krylov subspace based reduced-order models," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 4, pp. 239–248, Apr. 2000.
- [44] R. Achar, P. Gunupudi, M. Nakhla, and E. Chiprout, "Passive interconnect reduction algorithm for distributed/measured networks," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 4, pp. 287–301, Apr. 2000.
- [45] G. H. Golub and C. Van Loan, *Matrix Computations*, John Hopkins Univ. Press, Baltimore, MD, 1989.
- [46] R. Griffith and M. Nakhla, "A new high order absolutely stable explicit numerical integration algorithm for the time domain simulation of nonlinear circuits," in *Proc. IEEE/ACM Intl. Conf. Computer-Aided Design*, 1997, pp. 504–509.
- [47] P. Gunupudi, *Circuit-Reduction Techniques and Application to High-Speed Systems and Interconnects*, Ph.D. thesis, Carleton University, 2002.
- [48] D. Long, R. Melville, K. Ashby, and B. Horton, "Full-chip harmonic balance," in *Proc. IEEE Custom Integrated Circuits Conf.*, 1997, pp. 379–382.
- [49] E. L. Allgower and K. Georg, *Numerical Continuation Methods: An Introduction*, Springer series in computational mathematics; 13. Springer-Verlag,, 1990.
- [50] J. Rogers and C. Plett, *Radio Frequency Integrated Circuit Design*, Artech House, Massachusetts, 2003.
- [51] B. Razavi, *RF Microelectronics*, Prentice-Hall, New Jersey, 1998.
- [52] A. Ushida and L. O. Chua, "Frequency-domain analysis of nonlinear circuits driven by multi-tone signals," *IEEE Trans. Circuits Syst.*, vol. 31, no. 9, pp. 766–778, Sept. 1984.

- [53] K. S. Kundert, G. B. Sorkin, and A. Sangiovanni-Vincentelli, "Applying harmonic balance to almost-periodic circuits," *IEEE Trans. Microwave Theory Tech.*, vol. 36, no. 2, pp. 366–378, Feb. 1988.
- [54] R. C. Melville, L. Trajković, S. Fang, and L. T. Watson, "Artificial parameter Homotopy methods for the DC operating point problem," *IEEE Trans. Computer-Aided Design*, vol. 12, no. 6, pp. 861–877, June 1993.