

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

MODEL ORDER REDUCTION USING RATIONAL TRANSFER FUNCTION
FITTING AND EIGENMODE ANALYSIS

A thesis submitted in partial fulfillment of the

requirements for the degree of

MASTER OF SCIENCE

in

ELECTRICAL ENGINEERING

by

Arjun Surya Mullaguru

2008

To: Interim Dean Amir Mirmiran
College of Engineering and Computing

This thesis, written by Arjun Surya Mullaguru, and entitled Model Order Reduction using Rational Transfer Function Fitting and Eigenmode Analysis, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this thesis and recommend that it be approved.

Grover L. Larkins

Jean H. Andrian

Jeffrey Fan, Major Professor

Date of Defense: July 17, 2008

The thesis of Arjun Surya Mullaguru is approved.

Interim Dean Amir Mirmiran
College of Engineering and Computing

Dean George Walker
University Graduate School

Florida International University, 2008

DEDICATION

I dedicate this thesis to my mother M. Indira and my father Dr. M. Raghupathi. Without their love, patience, understanding, and support I would have not been able to complete this work.

ACKNOWLEDGMENTS

I first would like to thank my family, who have supported me throughout my entire life and I'll forever be grateful. Special mention goes to all my friends, VLSI lab members, who have helped me greatly. I would also wish to thank the members of my committee for the support, patience and understanding.

I especially would like to express my appreciation to my major professor, Dr. Jeffrey Fan for his continuous support and contribution on the development of this work. His belief in me and his tremendous strength for life inspired me and will continue to inspire me throughout.

ABSTRACT OF THE THESIS
MODEL ORDER REDUCTION USING RATIONAL TRANSFER FUNCTION
FITTING AND EIGENMODE ANALYSIS

by

Arjun Surya Mullaguru

Florida International University, 2008

Miami, Florida

Professor Jeffrey Fan, Major Professor

The purpose of this work was to propose a novel model order reduction technique using rational transfer function fitting and eigenmode analysis considering residues. A constant was defined as the key in the sorting algorithm as one of correlations, in order to sort the order of eigenvalues. It was demonstrated that the accuracy via eigenmode analysis considering residues was improved. The proposed algorithm is a general method to match pole values with frequency domain poles for linear RC and RLC systems. Calculation of pole eigenvalues and eigenvectors can be done with more sophisticated analysis with the same level or smaller cost in the proposed algorithm in comparison to Passive Reduced – Order Interconnect Macromodeling Algorithm (PRIMA). The experimental results simulated in Matlab R2007a show that the proposed algorithm reduced up to 90% errors compared to the existing model order reduction algorithm, such as PRIMA, in wide frequency environment with the same number of poles in comparison. But, at the cost of runtime which was increased up to 3X times.

TABLE OF CONTENTS

CHAPTER	PAGE
CHAPTER 1	1
INTRODUCTION	1
1.1 MODEL ORDER REDUCTION.....	1
1.1.1 WHY MOR?	3
1.2 MODIFIED NODAL ANALYSIS	5
1.2.1 RC CIRCUIT FORMULATION	6
1.2.2 RLC CIRCUIT FORMULATION.....	8
1.2.2.1 STATE SPACE REPRESENTATION.....	10
1.3 PROPOSED SYSTEM DESCRIPTION AND OBJECTIVE	12
1.3.1 PROBLEM STATEMENT	12
1.3.2 SCOPE AND OBJECTIVES	13
1.4 CHAPTER SUMMARY.....	14
CHAPTER 2	15
PROJECTION BASED MODEL ORDER REDUCTION	15
2.1 EIGEN DECOMPOSITION.....	15
2.2 CONCEPT OF MOMENTS	16
2.3 POINT MATCHING	18
2.4 MOMENT MATCHING	21
2.5 FRAME WORK OF PROJECTION BASED MODEL ORDER REDUCTION ..	25
2.5.1 PROJECTION BASED METHOD OF POLE COMPUTATION	26
2.5.2 KRYLOV SUBSPACES	27
2.6 PRIMA (PASSIVE REDUCED ORDER INTERCONNECT MACRO MODELLING ALGORITHM)	28
2.7 DISADVANTAGES OF PRIMA.....	30
2.8 CHAPTER SUMMARY.....	31
CHAPTER 3	32
STATISTICAL SPECTRUM MODEL ORDER REDUCTION	32
3.1 PROBLEM FORMULATION	32
3.2 SPATIAL CORRELATION.....	34
3.2.1 CONCEPT OF PRINCIPAL COMPONENT ANALYSIS	34
3.3 FRAMEWORK FOR STATISTICAL MODEL ORDER REDUCTION	36
3.3.1 MODIFIED KRYLOV SUBSPACE MODEL ORDER REDUCTION	36
3.3.2 STATISTICAL MODEL ORDER REDUCTION FLOW	38
3.3.3 STATISTICAL MOMENT COMPUTATION	39
3.3.4 SECOND ORDER EXPANSION OF TWO RANDOM VARIABLES.....	40
3.3.5 FIRST ORDER EXPANSION ON MULTIPLE RANDOM VARIABLES...	41
3.4 CHAPTER SUMMARY.....	43

CHAPTER 4	44
PROPOSED ALGORITHM	44
4.1 ORIGINAL DYNAMIC SYSTEM	44
4.2 EIGEN ANALYSIS.....	45
4.3 BUBBLE SORT ALGORITHM	49
4.4 IMPROVEMENT TO THE PROPOSED ALGORITHM.....	51
4.5 CHAPTER SUMMARY.....	53
 CHAPTER 5	 54
PROPOSED ALGORITHM TESTING AND VALIDATION	54
5.1 SIMULATION RESULTS	54
5.2 CHAPTER SUMMARY.....	57
 CHAPTER 6	 58
CONCLUSIONS AND FUTURE WORK.....	58
 LIST OF REFERENCES	 61

LIST OF FIGURES

FIGURE	PAGE
Figure 1. Basic concept of MOR (model order reduction) [1]	2
Figure 2. Example of MOR (model order reduction) transmission line with n passive elements	4
Figure 3. Example of MOR (model order reduction) equivalent transmission line	4
Figure 4. A RC tree demonstrating nodal analysis formulation [3].....	7
Figure 5. A RLC tree demonstrating modified nodal analysis formulation [3].....	9
Figure 6. Point Matching at $2q$ points [9].....	20
Figure 7. Rational transfer function fitting by Moment matching [9].....	23
Figure 8. Flow chart of Statistical spectrum model order reduction.....	39
Figure 9. Proposed algorithm vs. PRIMA with RC benchmark of 4800 nodes in wide frequency band.....	55
Figure 10. Proposed algorithm vs. PRIMA with benchmark of RLC circuits.....	57

LIST OF ACRONYMS

CMOS:	Complimentary Metal Oxide Field Effect Transistor
C:	Capacitor
L:	Inductor
MIMO:	Multi Input Multi Output
MOSFET:	Metal Oxide Field Effect Transistor
MOR:	Model Order Reduction
MNA:	Modified Nodal Analysis
ODE:	Ordinary Differential Equation
PRIMA:	Passive Reduced –Order Interconnect Macromodeling Algorithm
PCA:	Principal Component Analysis
R:	Resistor
SSMOR:	Statistical Spectrum Model Order Reduction
IC:	Integrated Circuit
I/O:	Input/Output
VLSI:	Very Large Scale Integration

CHAPTER 1

INTRODUCTION

The idea for this project grew out from a general interest in the broad field of Model Order Reduction (MOR). As VLSI technology advances into the sub-100 nm regime with increased operating frequency and decreased feature sizes, the nature of the VLSI design has changed significantly. One fundamental paradigm change is that parasitic interconnect effects dominate both the chip's performance and the design complexity growth [1]. As feature sizes become smaller, their electromagnetic couplings become more significant. Signal Integrity, crosstalk, skin effects, substrate loss, and digital and analog substrate couplings are now adding severe complications to design methodologies already stressed by increasing device counts. The unchecked parasitics from on-chip interconnects and off-chip packaging would de-tune the performance of high-speed circuits in terms of slew rate, phase margin, and bandwidth. Reduction of design complexity, especially for the high order RLC networks, is crucial for reducing the explosive design productivity gap in the nanometer VLSI regime and verification.

1.1 MODEL ORDER REDUCTION

Model Order Reduction (MOR) is a branch of system and control theory, which studies properties of dynamical systems in application for reducing their complexity, while preserving (to the possible extent) their input-output behavior [9]. The goal of model order reduction was to replace a large scale model of a physical system by a model of lower dimension, which exhibits similar behavior, typically measured in terms of its

frequency or time response characteristics. Such techniques were commonly used for analysis, approximation, and simulation of models arising from electromagnetic formulation of physical structures. The need to accurately account for all relevant physical effects imply that the mathematical formulation was used to describe such structures often results in very large models. Reducing the order or dimension of these models is crucial to enable the simulation and verification of such systems.

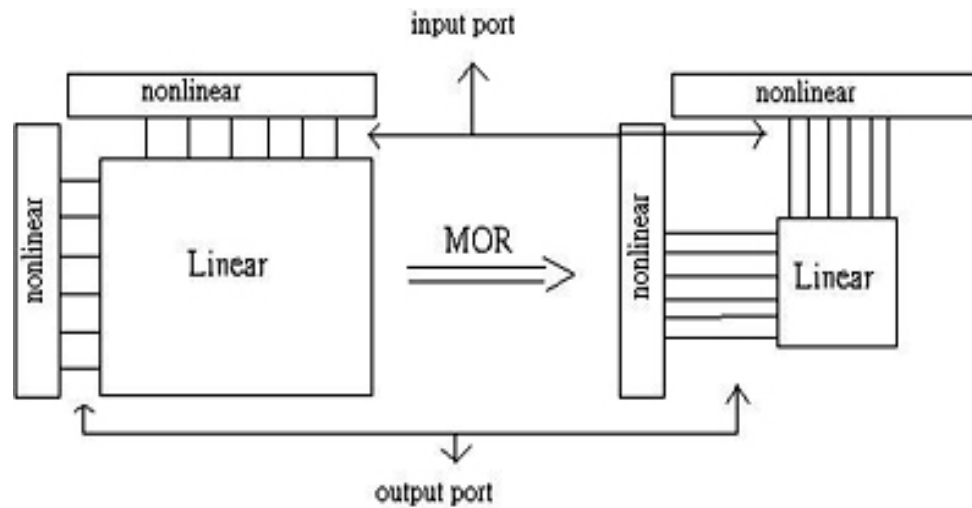


Figure 1. Basic concept of MOR (model order reduction) [1]

After entering deep sub-micron era, integrated circuits and systems are composed of a great number of linear (RLC) and non-linear, such as MOSFET or diode elements. The power grid (P/G) network connects millions to tens-of-millions of transistors together. Therefore, it may cost a huge amount of computation time to analyze the whole network. In the past few years, MOR, as shown in Figure-1 has become a promising technology to accelerate this kind of simulation time for linear components [12].

Model reduction is undoubtedly one of the most useful aspects of system theory for simulation, because of its immediate relevance to model simplification. It combines mathematical modelling problems with computational complexity issues [6]. However, mathematical models usually have some properties, which are very important from the physical point of view, such as conservativeness, dissipation, etc [4]. As integrated circuits and systems continue to be designed with smaller size and faster operation, RLC interconnect effects have a more dominant impact on signal propagation than ever before. In addition, parasitic coupling effects and reduced power supply voltage levels make the interconnect modelling more important than ever before.

1.1.1 WHY MOR?

The necessity for the MOR can be explained with the following example.

Consider a transmission line in a digital circuit. Its dynamical model is obtained by discretizing its length and representing each small piece as a small resistor, inductor plus capacitor to the ground. Then a description of the network was created using nodal voltage analysis. By solving this system for any given input, the voltage distribution was known at any given point of the line. The keen interest here is to know, how the signal is transmitted through the line, but not in knowing the exact distribution of a voltage along the line. The dependence of voltage and current at the one end of the line on the voltage and current at another end of the line should be known. In order to simulate this line efficiently (especially if this line is part of some complex circuit!), a "simplified" representation of this line was needed. Model order reduction process will produce this small approximation.

In Figure-2 a transmission line is represented in the form a lumped RLC circuit containing N number of resistor, capacitor and inductor elements. It is very difficult to analyze the output of the following complex lumped circuit. As said above, the keen interest here was to know the output voltage and current values for a given voltage and current at the input.

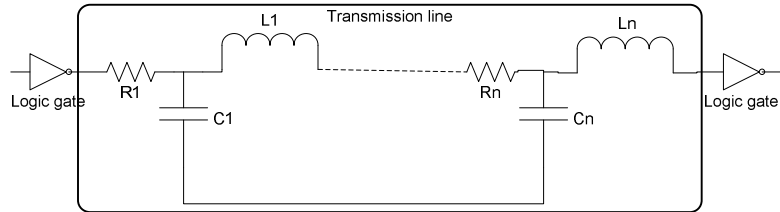


Figure 2. Example of MOR (model order reduction) transmission line with n passive elements

This lumped circuit containing N number of resistor, capacitor and inductor elements can be written as a circuit with equivalent resistor, capacitor and inductor element values as shown in Figure-3. Now, that the output can be obtained more easily than the previous case with this simplified transmission line. The equivalent resistor, inductor and capacitor elements provide a similar response equivalent to that of the transmission line containing all the elements. This simplified model can be used to analyze complex system.

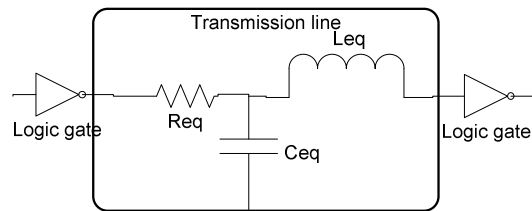


Figure 3. Example of MOR (model order reduction) equivalent transmission line

1.2 MODIFIED NODAL ANALYSIS

As the size of the arbitrary circuit grows larger nodal analysis was chosen over the loop analysis. In nodal analysis the number of nodal equations was usually smaller than the number of loop equations [2]. Also, formulating the nodal equations for computer solutions was easier than loop analysis. This assertion follows because in a manual analysis the loops are easily identified by inspection, whereas in an automated formulation of loop equations some algorithm must construct a set of independent loop equations. On the other hand, writing nodal equations was particularly easy if the circuit contains only resistors, independent current sources and voltage controlled current sources [3]. For such a network, one simply applies KCL to every node (except the reference node) and obtains a set of node equations directly without any finesse.

The MNA method is the most commonly used method in present-day computer aided circuit analysis programs. It retains the simplicity of the nodal method, while removing its limitations. In MNA method, the unknowns were the usual nodal voltages, plus some naturally occurring auxiliary currents. These unknown auxiliary currents include the following:

- Currents through independent voltage sources
- Currents through dependent voltage sources
- Currents through short-circuit elements
- The controlling currents of appropriate dependent sources
- Currents declared as output quantities

Once, the auxiliary currents in a network N are identified. Then the MNA equations for the network N are written step by step as follows:

Step 1. For every element x whose current I_x has been chosen as an auxiliary current, the element is replaced temporarily with an independent current source having the value I_x . Also, every current-controlled current source βI_x is replaced by an independent current source having value βI_x

Step 2. For every element x whose current I_x has been chosen as an auxiliary current, equation is written describing its constitutive relationship in the original network N.

Step 3. Combine the equations of step 1 and 2 and write in the form of a single linear matrix equation:

$$[\text{Coefficient matrix}] \times [\text{Unknown vector}] = [\text{Known vector}] \quad [2]$$

1.2.1 RC CIRCUIT FORMULATION

RC interconnect circuits can be formulated using nodal analysis formulation. Nodal analysis is a classical circuit analysis method based on Kirchhoff's current law (KCL) and branch constitutive equations. For a given RC linear circuit with $n + 1$ nodes, nodal analysis formulates the problem in the following two steps:

Step 1. Choose a ground or reference node, which usually is taken to be at a potential of zero volt. All other node voltages constitute n unknowns.

Step 2. Establish KCL equations for all the n nodes by representing branch currents in terms of node voltages using branch constitutive equations.

Consider a RC tree shown in the Figure-4. In nodal analysis formulation of the circuit, the unknowns are determined first.

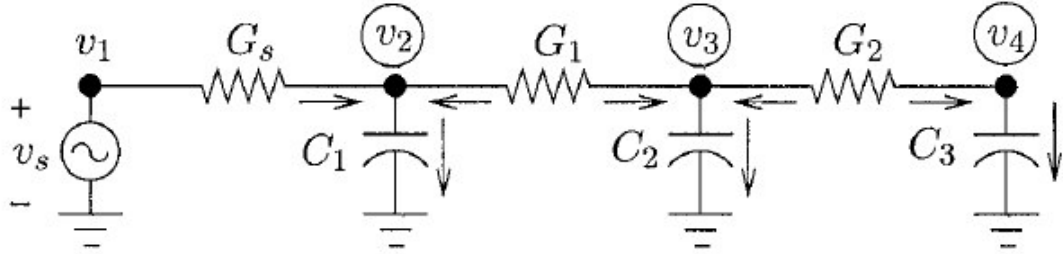


Figure 4. A RC tree demonstrating nodal analysis formulation [3]

Since v_1 is equal to v_s , which was the given input, the node voltages v_2 , v_3 and v_4 were used as unknown variables to write the three KCL equations.

$$C_1 \dot{v}_2 = G_s (v_s - v_2) + G_1 (v_3 - v_2) \quad (1.1)$$

$$C_2 \dot{v}_3 = G_1 (v_2 - v_3) + G_2 (v_4 - v_3) \quad (1.2)$$

$$C_3 \dot{v}_4 = G_2 (v_3 - v_4), \quad (1.3)$$

The matrix form of the above three simultaneous equations would be

$$\begin{bmatrix} C_1 & & \\ & C_2 & \\ & & C_3 \end{bmatrix} \begin{bmatrix} \dot{v}_2 \\ \dot{v}_3 \\ \dot{v}_4 \end{bmatrix} = - \begin{bmatrix} G_s + G_1 & -G_1 & 0 \\ -G_1 & G_1 + G_2 & -G_2 \\ 0 & -G_2 & G_2 \end{bmatrix} \begin{bmatrix} v_2 \\ v_3 \\ v_4 \end{bmatrix} + \begin{bmatrix} G_s v_s \\ 0 \\ 0 \end{bmatrix}. \quad (1.4)$$

Note that we have put a minus sign outside the square matrix. Without loss of generality, it was assumed that the voltage drop and the current from v_2 to v_3 were two output variables that are interested in, i.e., i_x and v_x were circuit outputs,

$$\begin{bmatrix} i_x \\ v_x \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ G_1 & -G_1 & 0 \end{bmatrix} \begin{bmatrix} v_2 \\ v_3 \\ v_4 \end{bmatrix}. \quad (1.5)$$

In general, RC circuit formulation can be expressed as

$$\begin{aligned} C\dot{v}(t) &= -Gv(t) + bu(t) \\ y(t) &= Lx(t) \end{aligned} \quad (1.6)$$

Where v denotes the n unknown nodal voltages in RC circuits the matrices $G \in \mathfrak{R}^{n \times n}$ and $C \in \mathfrak{R}^{n \times n}$ represent the conductance and capacitance elements, respectively. It must be noted that matrix C may be singular, i.e., some rows in C may be zero. It happens when the corresponding node does not connect to any capacitor.

1.2.2 RLC CIRCUIT FORMULATION

Modified nodal analysis is yet another classical circuit formulation method, which improves nodal analysis method by adding currents in inductors as unknown variables. The introduction of the inductance current variables would help keep modified nodal analysis formulation in the differential form.

The branch consecutive equation of an inductor is

$$v_L = L \frac{di(t)}{dt} \quad (1.7)$$

Where L is the inductance value. In the KCL equations involving the inductor, i_L can be represented in terms of v_L as,

$$i_L = \frac{1}{L} \int_{t_0}^t v_L dt + i(t_0) \quad (1.8)$$

For a given RLC linear circuit with $n + 1$ nodes, modified nodal analysis formulates the problem in the following two steps:

Step 1. Choose a ground or reference node, which usually was taken to be at a potential of zero volt. All other node voltages constitute n unknowns.

Step 2. Create a current variable for each inductor with certain direction defined.

Step 3. Establish KCL equations for all the n nodes by representing branch currents of RC elements in terms of node voltages and current variables pre-defined in step 2.

Step 4. Establish the branch constitutive equation of inductance in differential form using pre-defined current and nodal voltage variables.

Consider a RLC circuit as shown in the Figure-5 with a mutual inductance M between the inductors L_1 and L_2 . Let i_1 and i_2 are two current variables in addition to the four nodal voltages.

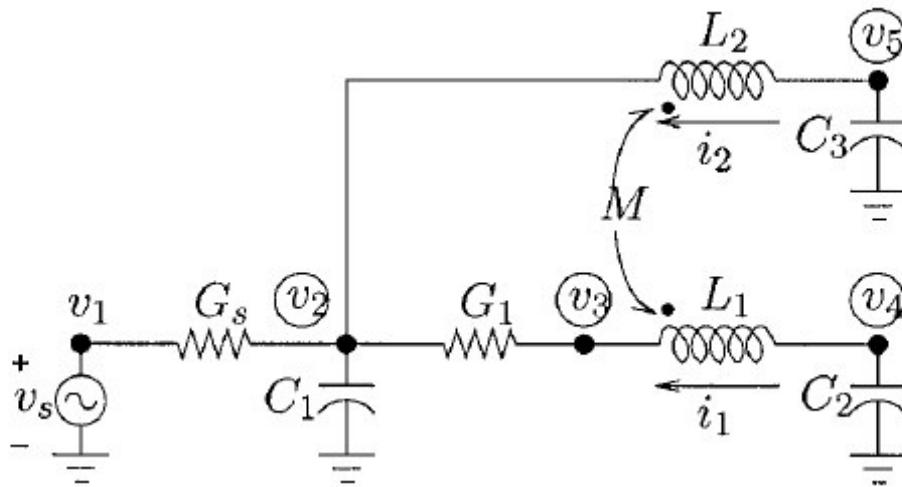


Figure 5. A RLC tree demonstrating modified nodal analysis formulation [3]

The modified nodal analysis formulation of the circuit is given by,

$$\begin{bmatrix} C_1 & & & & 0 & 0 \\ & 0 & & & 0 & 0 \\ & & C_2 & & 0 & 0 \\ & & & C_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & L_1 & M \\ 0 & 0 & 0 & 0 & M & L_2 \end{bmatrix} \begin{bmatrix} \dot{v}_2 \\ \dot{v}_3 \\ \dot{v}_4 \\ \dot{v}_5 \\ i_1 \\ i_2 \end{bmatrix} = \begin{bmatrix} G_s + G_1 & -G_1 & 0 & 0 & 0 & -1 \\ G_1 & G_1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_2 \\ v_3 \\ v_4 \\ v_5 \\ i_1 \\ i_2 \end{bmatrix} + \begin{bmatrix} G_s v_s \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (1.9)$$

The first four rows were derived from KCL for the five circled nodes. The last two rows are branch constitutive equations of the two inductors and the mutual one, which were added because of the two extra variables, i_1 and i_2 .

Let v_6 be the output voltage then,

$$[v_6] = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ i_1 \\ i_2 \end{bmatrix}. \quad (1.10)$$

1.2.2.1 STATE SPACE REPRESENTATION

A state space representation is a mathematical model of a physical system as a set of input, output and state variables related by first-order differential equations. To

abstract from the number of inputs, outputs and states, the variables were expressed as vectors and the differential and algebraic equations were written in matrix form (the last one can be done when the dynamical system was linear and time invariant)[1]. The state space representation (also known as the "time-domain approach") provides a convenient and compact way to model and analyze systems with multiple inputs and outputs. With p inputs and q outputs, it would otherwise have to be written down as $p \times q$ Laplace transforms to encode all the information about a system [4]. Unlike the frequency domain approach, the use of the state space representation was not limited to systems with linear components and zero initial conditions. "State space" refers to the space whose axes were the state variables. The state of the system can be represented as a vector within that space.

The internal state variables are the smallest possible subset of system variables that can represent the entire state of the system at any given time. State variables must be linearly independent; a state variable cannot be a linear combination of other state variables. The minimum number of state variables required to represent a given system, n , is usually equal to the order of the system's defining differential equation. In electric circuits, the number of state variables was often, though not always, the same as the number of energy storage elements in the circuit such as capacitors and inductors.

The most general state space representation of a linear system with p inputs, q outputs and n state variables is written in the following form:

$$\begin{aligned}\frac{dx(t)}{dt} &= Ax(t) + Bu(t) , \\ y(t) &= Cx(t) .\end{aligned}\tag{1.11}$$

Where $x(t)$ is called the "state vector", $y(t)$ is called the "output vector", $u(t)$ is called the "input (or control) vector", A is the "state matrix", B is the "input matrix", C is the "output matrix",

For a general RLC linear network, the modified nodal analysis is applied to formulate into the state space equations,

$$M \frac{dx(t)}{dt} = -Gx(t) + Bu(t),$$

$$y(t) = L^T x(t). \quad (1.12)$$

Where G and M are $n \times n$ conductance and susceptance matrices, B and L are the $n \times N$ input and output positions matrices, and, typically, $B = L$ or $B = -L$; N is the number of input or output ports.

Where,

$$x(t) = \begin{bmatrix} v(t) \\ i(t) \end{bmatrix} \quad M = \begin{bmatrix} C & 0 \\ 0 & L \end{bmatrix}$$

The C and L are generally capacitance and inductance matrices of the circuit.

State variables x can be nodal voltage or branch currents of the linear circuits.

1.3 PROPOSED SYSTEM DESCRIPTION AND OBJECTIVE

1.3.1 PROBLEM STATEMENT

The problem addressed in this thesis was to propose a new method of model order reduction using eigenmode analysis and rational transfer function fitting. The first goal was to obtain the response of the dynamic linear time variant system with less number of states than the actual number in the wide frequency band ($0-5 \times 10^{10}$ Hz).

Secondly, the existing model order reduction algorithm PRIMA [13] using subspace projection matrix was the most popular model order reduction technique in modern circuit simulation. However, PRIMA [13] will cause severe accuracy problem in wide frequency environments. But most of the modern day circuits are operated in the wide frequency band ($0-5 \times 10^{10}$ Hz). The proposed method is compared with existing method of PRIMA [13], at different nodes in wide frequency band ($0-5 \times 10^{10}$ Hz). The results would exhibit that response obtained using the proposed method should closely be matched with the response of the actual system, obtained through PRIMA [13].

1.3.2 SCOPE AND OBJECTIVES

The objective of this thesis is to obtain the response of the system with less number of states than the actual number in the wide frequency band ($0-5 \times 10^{10}$ Hz). The values of the actual system response at $2q$ states should match with the $2q$ states of the reduced linear RC & RLC systems in the frequency domain with frequency band ($0-5 \times 10^{10}$ Hz).

Firstly, RLC system was formulated as state space representation using MNA analysis. Then at the steady state condition, eigendecomposition was applied on the ordinary differential equation. The state matrix was then represented in the form of eigenmodes. Some of these modes will be eliminated based on assumptions. But, the response obtained here may not be accurate. It may not work well at higher order frequencies and large state matrices.

Secondly, by introducing the concept of Laplace transforms the RLC system was represented in the form, output function is equal to the transfer function times the input

function. This transfer function is then broken down into pole-residue form. Here the dominant $2q$ eigenvalues representing dominant poles were correlated with a constant key defined and then resultant eigenvalues were sorted out using sorting algorithm. Thus obtained $2q$ poles are matched with the original poles of the system in the wide frequency band of $(0-5 \times 10^{10} \text{ Hz})$.

In order to facilitate the progress, this project will be broken down into simplified points without going to much into details at the present moment.

- Obtain the dynamic linear time invariant system from MNA equations
- Writing Matlab code to implement the proposed method
- Simulate the code for different number of nodes for both RC and RLC meshes
- Compare the results obtained with the results of PRIMA

1.4 CHAPTER SUMMARY

In this chapter, first basic introduction of model order reduction was discussed. This is supported by the example of a transmission line consisting of RLC elements. Secondly, the concept of modified nodal analysis was discussed too. Also, the formulation of the equations for the RC and RLC meshes was learnt. Thirdly, the state space representation of a system was discussed. Here a linear time variant system is written in the form of an ordinary differential equation. Finally, the problem statement of the proposed new model order reduction algorithm was discussed along with its scope and objectives.

CHAPTER 2

PROJECTION BASED MODEL ORDER REDUCTION

2.1 EIGEN DECOMPOSITION

Eigenvectors and eigenvalues are numbers and vectors associated to square matrices, and together they provide the eigendecomposition of a matrix which analyzes the structure of this matrix. Even though the eigendecomposition does not exist for all square matrices, it has a particularly simple expression for a class of matrices often used in multivariate analysis such as correlation, covariance, or cross-product matrices [2].

Given a square matrix A , matrix eigenanalysis seeks to decompose A as a product $A = PDP^{-1}$ with P invertible and D diagonal.

$$A = PDP^{-1} \quad (2.1)$$

The equation $A = PDP^{-1}$ is equivalent to $AP = PD$. Let $D = \text{diag}(\lambda_1, \dots, \lambda_n)$ denote the columns of P by P_1, \dots, P_n . The columns of AP are the vectors AP_1, \dots, AP_n . The columns of PD are $\lambda_1 P_1, \dots, \lambda_n P_n$. Therefore, the decomposition $A = PAP^{-1}$ was found from the system of vector equations

$$AP_1 = \lambda_1 P_1,$$

$$AP_2 = \lambda_2 P_2,$$

...

$$AP_n = \lambda_n P_n.$$

A pair (x, λ) , where x is a vector and λ is a complex number, is called an eigenpair of the $n \times n$ matrix A , provided $Ax = \lambda x$ ($x \neq 0$ required). The nonzero

requirement is the result of seeking directions for a coordinate system the zero vector is not a direction. Any vector $x \neq 0$ that satisfies was called an eigenvector for λ and the value λ was called an eigenvalue of the square matrix A.

The decomposition $A = PDP^{-1}$, which is equivalent to solving system, requires finding all eigenpairs of the $n \times n$ matrix A. If A has N eigenpairs and N independent eigenvectors, then it is solved by constructing D as the diagonal matrix of eigenvalues and P as the matrix of corresponding eigenvectors

2.2 CONCEPT OF MOMENTS

In the s domain, the transfer function of a linear network $H(s)$ is defined as the ratio of the output to the input under zero initial conditions:

$$H(s) = \frac{Y(s)}{X(s)} \quad (2.2)$$

If the input was an impulse function $\delta(t)$, its Laplace transformation is 1. So the transfer function was also the impulse response at the port. If $H(s)$ was expanded around $s=0$ by the Taylor series expansion,

$$H(s) = \sum_{k=0}^{\infty} m_k s^k, \quad (2.3)$$

Where

$$m_k = \frac{1}{k!} \times \left. \frac{d^k H(s)}{ds^k} \right|_{s=0} \quad (2.4)$$

Where the k^{th} coefficient $H(s)$, m_k , is called the k^{th} moment

After the moments were generated, a general multi-input multi-output (MIMO) [6] transfer function $H(s)$ was represented as a Taylor series expansion form or the block moment form

$$H(s) = m_0 + m_1s + m_2s^2 + \dots, \quad (2.5)$$

Where, m_i is the i^{th} block moment of the circuit, and

$$m_i = L^T x_i \quad (2.6)$$

Where x_i is the i^{th} order state block moment vector. Once the moment expansion is available

The rational transfer function $H(s)$ is written as

$$H(s) = \frac{b_1 + b_2s + \dots b_{n-1}s^{n-1}}{1 + a_2s + \dots a_n s^n} \quad (2.7)$$

To compute the transient response waveforms, the partial fraction form was needed to be derived from the rational function form.

Partial fraction decomposition of a transfer function $H(s)$ is to represent it in the following form,

$$\begin{aligned} H(s) &= \frac{k_0}{s - p_0} + \frac{k_1}{s - p_1} + \dots + \frac{k_p}{s - p_q} \\ &= \sum_{i=0}^q \frac{k_i}{s - p_i} \end{aligned} \quad (2.8)$$

Where q is the order of the transfer function. Here p_i and k_i are the poles and residues of the transfer function.

2.3 POINT MATCHING

Consider an ordinary differential equation of a general dynamic linear time-invariant state-space model with only one input and one output

$$\begin{aligned}\frac{dx(t)}{dt} &= Ax(t) + bu(t) , \\ y(t) &= C^T x(t) .\end{aligned}\tag{2.9}$$

Where u is the input variable, y is the output variable, A is a $n \times n$ matrix and b is a $n \times 1$ matrix. x is a $n \times 1$ vector of state variables.

Consider some Laplace transform properties like, the bilateral Laplace transform which is given as,

$$X(s) = \int_{-\infty}^{\infty} x(t)e^{-st} dt\tag{2.10}$$

And the key transform property, given as

$$sX(s) = \int_{-\infty}^{\infty} \frac{dx(t)}{dt} e^{-st} dt\tag{2.11}$$

Considering the above two properties, the ordinary differential equation referred above can be rewritten as

$$Y(s) = C^T (sI - A)^{-1} bU(s)\tag{2.12}$$

Then the transfer function from $Y(s)$ to $U(s)$ can be written as

$$H(s) = C^T (sI - A)^{-1} b\tag{2.13}$$

Applying Eigen decomposition on 2.13, it is rewritten as

$$H(s) = C^T E (sI - A)^{-1} E^{-1} b\tag{2.14}$$

And replace $\tilde{c}^T = C^T E$ and $\tilde{b} = E^{-1} b$, we get

$$= \tilde{c}^T \begin{bmatrix} \frac{1}{s - \lambda_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{s - \lambda_n} \end{bmatrix} \tilde{b}$$

Hence the transfers function $H(s) = \sum_{i=1}^N \frac{\tilde{c}_i \tilde{b}_i}{s - \lambda_i}$. (2.15)

This is represented in a pole and residue form where $\tilde{c}_i \tilde{b}_i$ are called poles and $s - \lambda_i$ are called the residues. But, it still has N states that are started of with. i.e.,

$$H(s) = \frac{\tilde{c}_1 \tilde{b}_1}{s - \lambda_1} + \dots + \frac{\tilde{c}_n \tilde{b}_n}{s - \lambda_n} \quad (2.16)$$

When all the terms are cross multiplied, it results to the following representation of a $N - 1$ th order polynomial in the numerator and a N th order polynomial in the denominator and is represented as

$$H(s) = \frac{b_0 + b_1 s + \dots + b_{N-1} s^{N-1}}{1 + a_1 s + \dots + a_N s^N} \left. \vphantom{\frac{b_0 + b_1 s + \dots + b_{N-1} s^{N-1}}{1 + a_1 s + \dots + a_N s^N}} \right\} \text{Rational_Function} \quad (2.17)$$

Some of the terms of \tilde{b}_i are uncontrollable. These are the terms that can not be controlled during the inputs. Some terms of \tilde{c}_i are unobservable. In some cases product of $\tilde{b}_i \tilde{c}_i$ are very small and they are neglected. Considering all the above conditions the transfer function $H(s)$ with order N is reduced to an order $H_r(s)$ with order q where $q \ll N$. So, the transfer function $H(s)$ is written as.

$$H_r(s) = \frac{b_0^r + b_1^r s + \dots + b_{N-1}^r s^{N-1}}{1 + a_1^r s + \dots + a_n^r s^N} \quad (2.18)$$

In point matching the reduced transfer function $H_r(s)$ was matched with the original transfer function at particular $2q$ points and s is written as

for $i = 1$ to $2q$

$$H_r(s_i) = \frac{b_0^r + b_1^r s + \dots + b_q^r s^q}{1 + a_1^r s + \dots + a_q^r s^q} \quad (2.19)$$

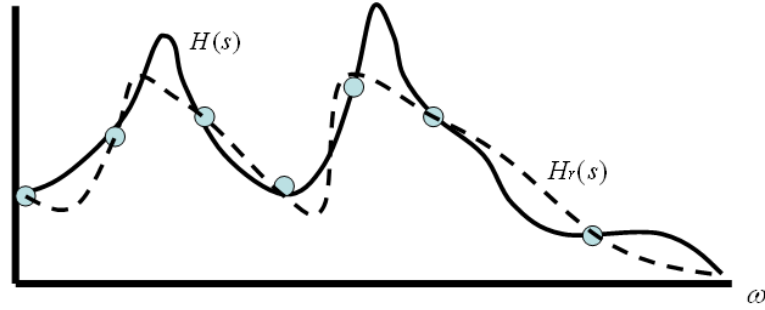


Figure 6. Point Matching at $2q$ points [9].

The denominator polynomial was cross multiplied with $H(s_i)$ and is equated to zero.

for $i = 1$ to $2q$

$$(1 + a_1^r s + \dots + a_q^r s^q)H_r(s_i) - (b_0^r + b_1^r s + \dots + b_q^r s^q) = 0 \quad (2.20)$$

This equation is represented in the matrix form below

$$\begin{bmatrix} s_1 H_r(s_1) & s_1^2 H_r(s_1) & \dots & -s_1^{q-1} \\ \vdots & \vdots & \dots & \\ \vdots & \vdots & \dots & \\ s_{2q} H_r(s_{2q}) & s_{2q}^2 H_r(s_{2q}) & \dots & -s_{2q}^{q-1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ b_{q-1} \end{bmatrix} = \begin{bmatrix} H(s_1) \\ H(s_2) \\ \vdots \\ H(s_{2q}) \end{bmatrix} \quad (2.21)$$

Here in this matrix each row represents different test points and each column represents reduced transfer function coefficients. But, each column corresponds to progressive higher powers of s and the columns are tending to be linearly dependent.

Internally $s = j\omega$ and $\omega = 2\pi f$. As we increase the order of s the order of frequency f also increases. When s tends to approach ∞ the output is zero [15].

In a dynamic system with the state variables the inputs has to first charge up the state and the state gives the output. But, if the input was put at a high frequency, it won't charge up the state and there will be no output. So, point matching will not be reliable at the high frequencies. Also, in point matching it's only accurate at the points where the reduced transfer function was matched with the original transfer function. It will be very inaccurate between the points.

2.4 MOMENT MATCHING

The initial approach is similar to the Point matching. First, consider an ordinary differential equation of a general dynamic linear time-invariant state-space model with only one input and one output

$$\begin{aligned} \frac{dx(t)}{dt} &= Ax(t) + bu(t), \\ y(t) &= C^T x(t). \end{aligned} \tag{2.22}$$

Where u is the input variable, y is the output variable, A is a $n \times n$ matrix and b is a $n \times 1$ matrix. x is a $n \times 1$ vector of state variables. After applying the Laplace properties the ordinary differential equation is written as,

$$Y(s) = C^T (sI - A)^{-1} bU(s) \tag{2.23}$$

Where,

$$H(s) = C^T (sI - A)^{-1} b \tag{2.24}$$

In point matching the reduced transfer function $H_r(s)$ was blindly matched with original transfer function $H(s)$ at certain $2q$ points. In moment matching a different approach is followed,

First consider the transfer function,

$$H(s) = C^T (sI - A)^{-1} b ,$$

This can also be written in the form,

$$H(s) = -C^T (I - sA)^{-1} A^{-1} b \quad (2.25)$$

and now the term $(I - sA)^{-1}$ is the form similar to $(1 - x)^{-1}$, which was nothing but Taylor series. The Taylor series expansion was given below,

$$(1 - x)^{-1} = 1 + x + x^2 + \dots . \quad (2.26)$$

Now using the concept of Taylor series the term $(I - sA)^{-1}$ is expanded as

$$(I - sA)^{-1} = 1 + sA^{-1} + s^2 A^{-2} + s^3 A^{-3} + \dots \quad (2.27)$$

Plugging the above expansion 2.27 back in the transfer function,

$$H(s) = -C^T (I - sA)^{-1} A^{-1} b . \quad (2.28)$$

The resultant transfer function is,

$$H(s) = C^T A^{-1} b + C^T A^{-2} b s + C^T A^{-3} b s^2 + \dots \quad (2.29)$$

It can also be written as,

$$H(s) = \sum_{k=0}^{\infty} m_k s^k , \text{ where } m_k = C^T A^{-(k+1)} b \quad (2.30)$$

m_k is the k^{th} moment. These moments are the coefficients of the Taylor series. For a system the more the moments match the higher end frequencies are expected to match

[1]. Now the reduced order transfer function is expanded using the Taylor series mentioned above and the derived moments are matched with the moments of the original transfer function.

$$H_r(s) = \frac{b_0^r + b_1^r s + \dots + b_{N-1}^r s^{N-1}}{1 + a_1^r s + \dots + a_n^r s^N} = m_0 + m_1 s + \dots + m_{2q-1} s + \dots \quad (2.31)$$

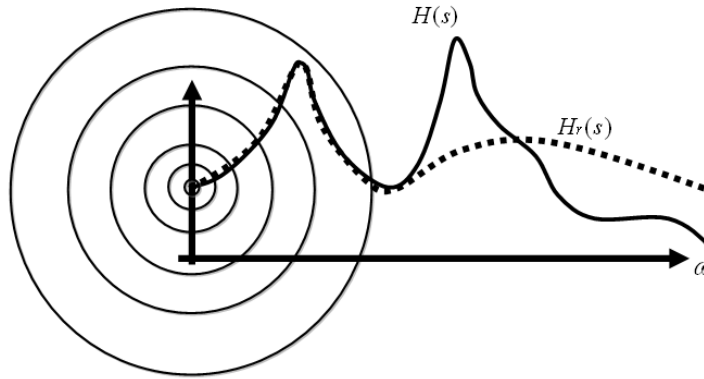


Figure 7. Rational transfer function fitting by Moment matching [9].

Once again by cross multiplying the denominator polynomial with the moments of the original transfer function and matching the terms with the powers of s , a system of equations with the coefficients of a and a matrix with the moments of the reduced transfer function that were to be matched with the moments of the original transfer function.

$$\begin{bmatrix} m_0 & m_1 & \dots & m_{k-1} \\ m_1 & \dots & \dots & \vdots \\ \vdots & \dots & \dots & m_{2q-3} \\ m_{k-1} & \dots & m_{2q-3} & m_{2q-2} \end{bmatrix} \begin{bmatrix} a_q \\ a_{q-1} \\ \vdots \\ a_1 \end{bmatrix} = \begin{bmatrix} m_q \\ m_{q+1} \\ \vdots \\ m_{2q-1} \end{bmatrix} \quad (2.32)$$

Now consider the first moment in detail i.e. $m_0 = C^T A^{-1} b$

Looking at the part $A^{-1}b$, the vector b tends to be pushed in the direction of the eigenvector with the largest eigenvalue of A . Also b can be written as weighted combination of eigenvectors as below,

$$A^{-1}b = A^{-1}[\alpha_1 \vec{e}_1 + \dots + \alpha_n \vec{e}_n] \quad (2.33)$$

After multiplying with A^{-1} , the $A^{-1}b$ is represented as

$$A^{-1}b = \alpha_1 \lambda^{-1} \vec{e}_1 + \dots + \alpha_n \lambda_n^{-1} \vec{e}_n \quad (2.34)$$

Where λ is the eigenvalue.

If the power of A^{-1} is raised to the k^{th} order, then

$$A^{-k}b = \alpha_1 \lambda^{-k} \vec{e}_1 + \dots + \alpha_n \lambda_n^{-k} \vec{e}_n \quad (2.35)$$

If one of these eigenvectors dominates then the vector b tends to align with the dominant eigenvector there by shrinking the other components by the order of the magnitude. At the higher orders, all the moments represented in the matrix form are related with the dominant eigenvalue λ [15]. If the last to the second row of the matrix with the reduced order transfer function moments were taken and multiplied by λ , the last row of the moments show up. This tells us that these moments were very much linearly dependent and they generate ill conditioned matrices. Each new moment was the resultant of the scaled old moment with the dominant eigenvalue λ .

In moment matching, a single point was picked from the reduced transfer function and the derivatives of the transfer function were matched with the original transfer function at progressively higher order. It matches well at the point picked and at the near frequency. But, it no longer matches at the higher frequency.

2.5 FRAME WORK OF PROJECTION BASED MODEL ORDER

REDUCTION

Consider a general linear time-invariant state-space model with only one input and one output

$$\begin{aligned}\frac{dx(t)}{dt} &= Ax(t) + bu(t) \quad , \\ y(t) &= C^T x(t) .\end{aligned}\tag{2.36}$$

Where u is the input variable, y is the output variable, A is a $n \times n$ matrix and b is a $n \times 1$ matrix. x is a $n \times 1$ vector of state variables. Then the transfer function from $u(t)$ to $y(t)$ can be given as

$$H(s) = C^T (sI - A)^{-1} b \tag{2.37}$$

Typically, the number of state variables n is very large so that the simulation and synthesis of the whole systems was very low. The intention was to build a much smaller system, such that the transient response $y(t)$ to some given input signal $u(t)$ is approximated to that by the original system. The concepts of controllability and observability can give good answer. The uncontrollable and unobservable parts of the system can be removed without affecting the transfer function.

The state transformation of $x = Tz$, where T is the matrix of eigenvectors of A and is assumed that the eigenvalues of A are denoted as $\lambda_1, \dots, \lambda_n$ and they were simple and unique. Then,

$$\frac{dz(t)}{dt} = T^{-1}ATz(t) + T^{-1}bu(t) = \begin{bmatrix} \lambda_1 & & \\ & \cdot & \\ & & \lambda_n \end{bmatrix} z(t) + \begin{bmatrix} \bar{b}_1 \\ \cdot \\ \bar{b}_n \end{bmatrix} u(t),$$

$$y(t) = C^T Tz(t). \quad (2.38)$$

If \bar{b}_i is zero, then the state variable z_i is not controllable and can be removed. If $\bar{c} = C^T$, if \bar{c}_i is zero, then the state variable z_i is unobservable and can be removed. So, the key issue of the model order reduction is to remove uncontrollable or unobservable parts or practically weakly controllable or observable parts.

2.5.1 PROJECTION BASED METHOD OF POLE COMPUTATION

This is the other way of finding poles in a numerically reliable way. Here poles are computed from the eigendecomposition of the reduced system.

In the Krylov subspace projection based model order reduction process, moment vectors generated recursively are first orthonormalized during the generation process and then used to build a projection matrix [7, 10]. The projection matrix is then used to reduce the original circuit matrices by congruence transformation and ensures that the reduced system is passive [11]. By using this method, it requires only q moments to find q poles. It guarantees that all the poles computed are stable owing to the nature of congruence transformation and the MNA formulation of the original RLC circuit matrices.

First a $n \times q$ matrix is formed, where each moment vector is a column.

$$M = [m_0, m_1, \dots, m_{q-1}]_{n \times q}. \quad (2.39)$$

Here $q \ll n$, and n is the number of state variables in the original circuit and also the dimension of the moment vectors. Then M is orthonormalized into a $n \times q$ projection matrix V , such that the columns of V are mutually orthogonal. Once the projection matrix V was obtained, the original circuit matrix G and C from the beginning of the topic can be reduced to two $q \times q$ order-reduced matrices by the congruence transformation

$$\hat{G} = V^T G V, \quad \hat{C} = V^T C V \quad (2.40)$$

After the reduction process, the eigenvalues of the matrix $\hat{G}^{-1} \hat{C}$ are related to the dominant poles as

$$p_i = -\frac{1}{\lambda_i} \quad (2.41)$$

Where p_i and λ_i are the i^{th} pole and eigenvalue. This was easily obtained by performing the eigendecomposition of $\hat{G}^{-1} \hat{C}$.

2.5.2 KRYLOV SUBSPACES

A subset of a vector space is called a vector subspace or subspace. The subspace was uniquely defined by a set of vectors $V = \{v_1, v_2, \dots, v_n\}$. The set of all linear combinations of these vectors were referred to as the span of V [8]

$$\begin{aligned} \text{span}\{V\} &= \text{span}\{v_1, v_2, \dots, v_n\}, \\ &= \left\{ x \mid x = \sum_{i=1}^n \alpha_i v_i \right\}, \end{aligned} \quad (2.42)$$

Where α_i are real numbers. If v_i are linearly independent, then each vector of span $\{V\}$ admits a unique expression as a linear combination of v_i . The set V is then called a basis of subspace $\text{span}\{V\}$. For $n \times n$ matrix A and a vector b , the Krylov subspace $K_q(A, b)$ is defined as,

$$K_q(A, b) = \text{span}\{b, Ab, A^2b, \dots, A^qb\}, \quad (2.43)$$

Where q is a positive integer.

2.6 PRIMA (PASSIVE REDUCED ORDER INTERCONNECT MACRO MODELLING ALGORITHM)

PRIMA [13] is a projection based model order reduction algorithm which performs a general passive projection based model order reduction on multi-input and multi-output (MIMO) RLC linear dynamic systems. In this algorithm the unstable poles were discarded and residues were readjusted using block Arnoldi algorithm. Also with the help of congruence transformation the passivity was preserved [13].

Getting into PRIMA in detail, without loss of generality, a linear m -port RLC dynamic multi-input and multi-output (MIMO) as

$$\begin{aligned} C\dot{x}_n &= -Gx_n + Bu_m, \\ i_m &= L^T x_n, \end{aligned} \quad (2.44)$$

Where x is the vector of state variables and n is the number of state variables, m is the number of independent sources specified as ports. C, G are matrices for conductance and storage elements, B and L indicate input and output ports. Typically $B = L$ for

interconnect circuits and both of them are $n \times N$ matrices. N is the number of terminals (as input and output ports).

Let $A = -G^{-1}C$, $A \in \mathfrak{R}^{n \times n}$ and $R = G^{-1}B$, $R = [r_0 r_1 \dots r_m] \in \mathfrak{R}^{n \times m}$. The y parameter matrix after Laplace transformation is,

$$Y(s) = L^T (G + sC)^{-1} B = L^T (I_n - sA)^{-1} R \quad (2.45)$$

Where I_n is the $n \times n$ identity matrix. The block moments of $Y(s)$ were defined as the coefficients of the Taylor expansion of $Y(s)$ around $s = 0$

$$Y(s) = M_0 + M_1 s + M_2 s^2 + \dots, \quad (2.46)$$

Where $M_i \in \mathfrak{R}^{N \times N}$ and can be computed as $M_i = L^T A^i R$.

The idea of model order reduction was to find a compact system much smaller than the original system. The Krylov subspace-based method accomplished this by projecting the original system onto a special subspace, which expands the same space as the block moments of the original system [11]. Specifically, the Krylov subspace was defined as,

$$Kr(A, R, q) = \text{colsp}[R, AR, AR^2, \dots, A^{k-1}R, \\ A^k r_0, A^k r_1, \dots, A^k r_l] \quad (2.47)$$

$$k = [q/m], \quad l = q - km.$$

For simplicity of expression, assume that $q = m \times k$ in the following context. In reality any k can be chosen. Then, with PRIMA the orthogonal matrix $X \in \mathfrak{R}^{n \times q}$ was found such that $\text{colsp}(X) = Kr(A, R, q)$. with,

$$\tilde{C} = X^T C X \quad \tilde{G} = X^T G X$$

$$\tilde{B} = X^T B \quad \tilde{L} = X^T L,$$

The reduced system of size q is found as

$$\begin{aligned} \tilde{C}\dot{\tilde{x}}_n &= -\tilde{G}\tilde{x}_n + \tilde{B}u_m, \\ i_m &= \tilde{L}^T x_n. \end{aligned} \tag{2.48}$$

The reduced system has q poles, which were the dominant poles of the original system. The order of block moments k is related to q by $k = \lceil q/m \rceil$. At least m moments were needed to match the first block moment ($k = 1$). For every one order of block moment increase, an addition of m additional poles was needed. This is a highly inefficient reduction process.

2.7 DISADVANTAGES OF PRIMA

The problem with existing projection based model order reduction techniques is that they are not efficient at reducing circuits with many ports. This was reflected in several aspects of the existing Krylov subspace algorithms like PRIMA.

First, the time complexity of PRIMA is proportional to the number of ports of the circuits as moments excited by every port need to be computed and matrix valued transfer functions are generated [1]. Second, the poles of the reduced models increase linearly with the number of ports, and this makes the reduced models much larger than necessary [14].

The fundamental reason is that all the Krylov-based projection methods were working directly on the moments, which contain the information of both the poles and residues for the corresponding transfer function [13]. To deal with more ports, there are more transfer functions to be dealt and more poles and residues to be computed.

However, poles among different transfer functions are the same for the same circuits as poles are characteristics of a system. But projection-based methods cannot take advantage of this as they operate directly on moments. As more residues were computed for more transfer functions, more poles also generated. However, generating more poles does not always help to improve the accuracy of the reduced models, as more block moments are not always matched as the number of poles increases [5]. As a result, projection-based methods lead to larger reduced models than necessary when the numbers of ports are larger. Also note that moments were very much linearly dependent and they generate ill conditioned matrices. Each new moment is the resultant of the scaled old moment with the dominant eigenvalue λ . The higher order moments will become less accurate. Thus, stating that PRIMA will be inaccurate at the higher frequencies.

2.8 CHAPTER SUMMARY

In this chapter, projection based model order reduction techniques were discussed. First, the concept of eigendecomposition which was used to derive the eigenvalues and eigenvectors are discussed. Secondly, the concept of moments, where the transfer function of a system was expanded in the terms of Taylor series was discussed. This was continued with the discussion of the techniques like point matching where the reduced transfer function was matched with the original transfer function at certain $2q$ points. Also, the moment matching technique was discussed where the function and its derivatives of the reduced order system was matched with the original system. Thirdly, the frame work for the projection based model order reduction was discussed, based on this concept the PRIMA is discussed along with its disadvantages.

CHAPTER 3

STATISTICAL SPECTRUM MODEL ORDER REDUCTION

In this chapter, a new statistical spectrum based method, called Statistical Spectrum Model Order Reduction (SSMOR), to generate the order-reduced variational models in consideration of both inter-die and intra-die variations with spatial correlation was discussed [16]. This method consists of the statistical spectrum method, which includes the Krylov subspace based model order reduction technique, and Monte Carlo sampling to generate order-reduced variational models. To consider the spatial correlation, the orthogonal decomposition is applied via principal component analysis (PCA) [17] to map the correlated random variables into independent and uncorrelated variables.

The SSMOR uses the statistical spectrum method to compute the variational moments, which do not suffer the problems of over-pessimism, or the accumulated inaccuracy. Also, this method addresses the issue of spatial correlation. After variational moments were generated, Monte Carlo sampling method is applied by using modified Krylov subspace reduction approach to generate the variational order-reduced models. Since Monte Carlo only operates on the order-reduced space (namely, within a few moments), therefore the cost of high computing diminishes.

3.1 PROBLEM FORMULATION

Considering the following state equation for a given RLC interconnect circuit using modified nodal analysis (MNA) formulation:

$$Gv(t) + C \frac{dv(t)}{dt} = Bu(t) \quad (3.1)$$

Where $G \in \mathfrak{R}^{n \times n}$ is the conductance matrix, $C \in \mathfrak{R}^{n \times n}$ is the matrix resulting from storage elements, and $v(t)$ is the vector of time-varying node voltages and branch currents of voltage sources. $u(t)$ is the vector of independent power sources, and B is the input selector matrix. The G, C matrices, and input currents $u(t)$ depend on the circuit parameters, such as metal wire width, length, thickness on interconnects, as well as transistor parameters, like channel length, width, gate oxide thickness, etc.

In this method, all the circuit parameter variations are treated as correlated Gaussian random variables. The spatial correlations are removed by using a set of independent random variables via the orthogonal mapping technique, i.e. Principal Component Analysis (PCA).

Assume that there are n number of dependent, correlated random Gaussian variables to indicate the variations in manufacturing process. After applying PCA, those correlated variables are transformed into independent, uncorrelated orthonormal random Gaussian variables ξ_i ; $i = 1, \dots, n$, which actually are model variation parameters, such as the channel length and the device threshold voltage. This implies that the matrices G and C are functions of ξ , i.e. $G(\xi)$ and $C(\xi)$ caused by the process variations. Therefore, we may rewrite the equation (3.1) as:

$$G(\xi)v(t) + C(\xi) \frac{dv(t)}{dt} = Bu(t) \quad (3.2)$$

Note that input $u(t)$ is also subject to variations. Thus, the problem was to develop a variational order-reduced system in terms of $\hat{G} \in \mathfrak{R}^{k \times k}$ and $\hat{C} \in \mathfrak{R}^{k \times k}$, subject to $k \ll n$, such that

$$\hat{G}v(t) + \hat{C} \frac{dv(t)}{dt} = \hat{B}u(t) \quad (3.3)$$

Where \hat{G} and \hat{C} consist of variational matrix elements, which are treated as uncorrelated variables after applying PCA. The input sources may be variational, thus the reduced models may be combined with the Monte Carlo method to calculate the variational responses. In addition, the reduced system can be represented in terms of variational pole/residue forms. Therefore, transfer functions were evaluated through fast transient waveform computation by using the recursive convolution method.

3.2 SPATIAL CORRELATION

In this section, the spatial correlation was considered among various variations within a RLC interconnect. Spatial correlations existing between the interconnects of different forms have been modeled in timing analysis [18, 19]. The general way to consider spatial correlation is by means of mapping the correlated random variables into a set of independent variables. This can be done by using some orthogonal mapping techniques, such as principal component analysis (PCA).

3.2.1 CONCEPT OF PRINCIPAL COMPONENT ANALYSIS

The concept of principal component analysis is briefly reviewed. It is used here to transform the random variables with correlation into uncorrelated random variables [20]. Suppose that x is a vector of n random variables, $x = [x_1, x_2, \dots, x_n]$, with covariance matrix C and mean vector $\mu_x = [\mu_{x1}, \mu_{x2}, \dots, \mu_{xn}]$. To find the orthogonal random variables, first the eigenvalue and corresponding eigenvector are calculated. Then, by

ordering the eigenvectors in descending order eigenvalues, the orthogonal matrix A was obtained. Here, A is expressed as,

$$A = [e_1^T, e_2^T, \dots, e_n^T]^T \quad (3.4)$$

where e_i is the corresponding eigenvector to eigenvalue λ_i , which satisfies

$$\lambda_i e_i = C e_i, i = 1, 2, \dots, n. \quad (3.5)$$

And

$$\lambda_i < \lambda_{i-1}, i = 2, 3, \dots, n. \quad (3.6)$$

With A , the transformation is performed to get orthogonal random variables

$y, y = [y_1, y_2, \dots, y_n]^T$, By using

$$y = A(x - \mu_x) \quad (3.7)$$

where, y_i is a random variable with Gaussian distribution. The mean, μ_{y_i} is zero and

the standard deviation, σ_{y_i} , is $\sqrt{\lambda_i}$ under the condition that [20]

$$e_i^T e_i = 1, i = 1, 2, \dots, n \quad (3.8)$$

Here, due to the orthogonal property of matrix A , i.e.

$$A^{-1} = A^T \quad (3.9)$$

To reconstruct the original random variables, the following equation is used:

$$x = A^T y + \mu_x \quad (3.10)$$

Therefore, the (3.47) with correlated random variables ϕ in current source

$$Gv(t) + C \frac{dv(t)}{dt} = I(t, \phi) \quad (3.12)$$

can be expressed in terms of uncorrelated random variables ,

$$Gv(t) + C \frac{dv(t)}{dt} = I(t, \xi) \quad (3.13)$$

3.3 FRAMEWORK FOR STATISTICAL MODEL ORDER REDUCTION

In this section, a modified Krylov subspace model order reduction (MOR) framework is discussed [23]. This is suitable for variational modeling of interconnect circuits, followed by the proposed variational moment computation method.

3.3.1 MODIFIED KRYLOV SUBSPACE MODEL ORDER REDUCTION

The idea of Krylov subspace based MOR method was to project the given circuit states into the dimension-reduced Krylov subspace of circuit states. The Krylov subspace essentially is spanned by the dominant moment vectors of the circuit transfer function. For a state space equation of an RLC circuit in equation (3.2), Krylov subspace is defined as

$$K_q(A, b) = \text{span} \{b, Ab, A^2b, \dots, A^q b\}, \quad (3.14)$$

where $A = G^{-1}C$ and $b = G^{-1}B$ and q is some given positive integer. Note that $A^i b$ is the i^{th} block moment defined as

$$m_i = A^i b = (-G^{-1}C)^i G^{-1}B, \quad (3.15)$$

of the circuit state transfer function, namely, $H(s) = (G + Cs)^{-1}B$, moment m_i can be directly computed in a recursive way

$$\begin{aligned} m_0 &= G^{-1}B; \\ m_1 &= -G^{-1}Cm_0; \\ m_i &= -G^{-1}Cm_{i-1}; \text{ for } i > 0 \end{aligned} \quad (3.16)$$

One way to build the reduced model was by means of Pade approximation, which computes the poles/residues of the transfer functions by using the moment information directly [21]. However, this explicit moment matching method was not stable numerically for computing higher order models. In modified Krylov subspace projection based model order reduction method to generate the reduced model. Specifically, the moment matrix M is defined as,

$$M = [m_0, m_1, \dots, m_{q-1}] \quad (3.17)$$

As it is known that the standard Krylov subspace projection method is to orthonormalize the vectors in M in order to generate a projection matrix V with the same dimension. Numerical methods like Arnoldi and Lanczos methods are typically used for the orthonormalization process, where the moment vectors were orthonormalized immediately after generation against all the previously-generated moment vectors. Such an orthonormalization process, however, was not suitable for this variational modeling process, as it was difficult to pass the variational characteristics through the orthonormalization process using the aforementioned statistical spectrum method.

Instead, all variational moments were computed by using statistical spectrum method. After all the block moments and associated variations were obtained, Monte Carlo sampling is used to generate the variational reduced models. In each sampling run, moment vectors in M were orthonormalized by using Gram-Schmidt or modified Gram-Schmidt orthonormalization algorithms, to compute the projection matrix V . Once the

projection matrix V was obtained, the original circuit matrices G and C are transformed into dimension-reduced matrices by congruence transformation:

$$\hat{G} = V^T G V; \hat{C} = V^T C V; \hat{B} = V^T B \quad (3.18)$$

Due to the nature of congruence transformation, the reduction process guarantees the passivity of all the reduced models. To compute the poles and residues, further eigen-decomposition of $\hat{G}^{-1}\hat{C} = S\Lambda S^{-1}$, where $\Lambda = \text{diag}(\lambda_1, \lambda_2 \dots, \lambda_i)$, are the reciprocals of the dominant poles.

To find the residues, w was solved as $\hat{G}w = V^T B$. Then the residues were simply the multiplications of $S^T V^T B$ and $S^{-1}w$. However, note that when generating the variational reduced models using Monte-Carlo sampling, it is necessary to consider the variations in both moments (i.e. the projection matrix V) and the given G and C matrices in state equations. One important remark is that those variations are correlated in nature, thus they are treated as correlated data during the Monte Carlo sampling.

3.3.2 STATISTICAL MODEL ORDER REDUCTION FLOW

In the statistical model order reduction flow, Statistical spectrum model order reduction (SSMOR) [23] is shown in the Figure.8. In the flow, the statistical spectrum method was used to compute the variational moments. After that, the Monte Carlo sampling method was adapted to generate the variational reduced models by using the modified Krylov subspace method. The samplings are done based on the computed means and variances of Gaussian distributions for each corresponding moment.

3.3.3 STATISTICAL MOMENT COMPUTATION

The statistical moment computation is based on the Hermite polynomials method [17]. The method is applied to compute the statistical moments and the statistical reduced models. In this approach, the spatial correlated random variables are transformed into uncorrelated variables by PCA. As a result, the random variables are assumed to be uncorrelated in the discussion of this section.

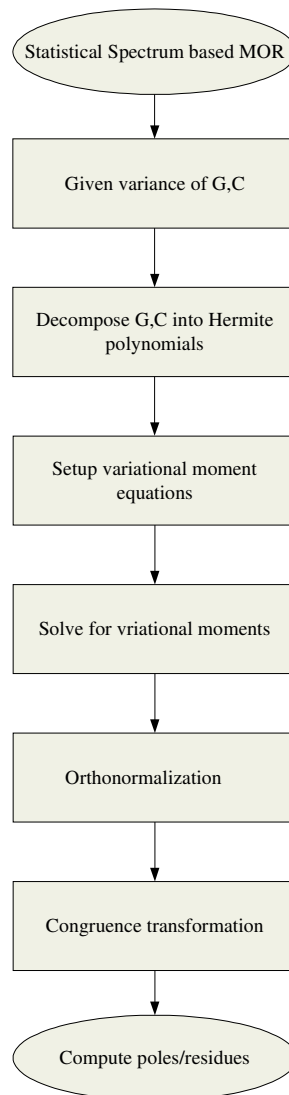


Figure 8. Flow chart of Statistical spectrum model order reduction

3.3.4 SECOND ORDER EXPANSION OF TWO RANDOM VARIABLES

In the case of two random variables, the matrices G and C are functions of two random variables, ξ_1 and ξ_2 . The Hermite polynomial basis for the order of two is $[1, \xi_1, \xi_2, \xi_1^2 - 1, \xi_1\xi_2, \xi_2^2 - 1]^T$. The variational matrices G and C can be expressed as $G = g_0 + g_1\xi_1 + g_2\xi_2$ and $C = c_0 + c_1\xi_1 + c_2\xi_2$ respectively.

Applying the equalities of Gaussian distribution for two random variables and the principle of orthogonality, the zero moment can be solved with the following equation:

$$\begin{bmatrix} g_0 & g_1 & g_2 & 0 & 0 & 0 \\ g_1 & g_0 & 0 & 2g_1 & g_2 & 0 \\ 0 & 2g_1 & 0 & 2g_0 & 0 & 0 \\ g_2 & 0 & g_0 & 0 & g_1 & 2g_2 \\ 0 & 0 & g_2 & 0 & 0 & g_0 \\ 0 & g_2 & g_1 & 0 & g_0 & 0 \end{bmatrix} \begin{bmatrix} a_{m0} \\ a_{m1} \\ a_{m2} \\ a_{m3} \\ a_{m4} \\ a_{m5} \end{bmatrix} - \begin{bmatrix} B \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = 0 \quad (3.19)$$

where $\vec{a}_m = [a_{m0}, a_{m1}, \dots, a_{m5}]$ is the coefficient vector in terms of second order Hermite polynomial for the zero moment and B is the input selector matrix. Similarly, the $2q^{th}$ moment can be derived from $(2q-1)^{th}$ moment in the following recursive way.

$$\begin{bmatrix} g_0 & g_1 & g_2 & 0 & 0 & 0 \\ g_1 & g_0 & 0 & 2g_1 & g_2 & 0 \\ 0 & 2g_1 & 0 & 2g_0 & 0 & 0 \\ g_2 & 0 & g_0 & 0 & g_1 & 2g_2 \\ 0 & 0 & g_2 & 0 & 0 & g_0 \\ 0 & g_2 & g_1 & 0 & g_0 & 0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} + \begin{bmatrix} c_0 & c_1 & c_2 & 0 & 0 & 0 \\ c_1 & c_0 & 0 & 2c_1 & c_2 & 0 \\ 0 & 2c_1 & 0 & 2c_0 & 0 & 0 \\ c_2 & 0 & c_0 & 0 & c_1 & 2c_2 \\ 0 & 0 & c_2 & 0 & 0 & c_0 \\ 0 & c_2 & c_1 & 0 & c_0 & 0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = 0 \quad (3.20)$$

Where $\vec{a} = [a_0, a_1, \dots, a_5]$ and $\vec{b} = [b_0, b_1, \dots, b_5]$ are $2q^{th}$ and $(2q-1)^{th}$ moment, respectively.

3.3.5 FIRST ORDER EXPANSION ON MULTIPLE RANDOM VARIABLES

Since the correlations may be removed by PCA, let's consider n uncorrelated random variables. In this case, first order Hermite expansion was used. The variational G and C matrices now become

$$G = g_0 + \sum_{i=0}^n g_i \xi_i; C = c_0 + \sum_{i=0}^n c_i \xi_i \quad (3.21)$$

where, ξ_i is the random variable with Gaussian distribution with zero mean and standard deviation 1. g_0 and c_0 denote the means of G and C respectively. g_i and c_i are the variances of the associated ξ_i respectively.

For n random variables, it is known that the basis of Hermite polynomials with expansion to first order is $[1, \xi_1, \xi_2, \dots, \xi_n]$. Thus,

$$m_0 = a_{m_0} + \sum_{i=1}^n a_{m_i} \xi_i; m_{2q} = a_0 + \sum_{i=1}^n a_i \xi_i; \quad (3.22)$$

$$m_{2q-1} = b_0 + \sum_{i=1}^n b_i \xi_i; \quad (3.23)$$

where, $[a_{m_0}, a_{m_1}, \dots, a_{m_n}]$, $[a_0, a_1, \dots, a_n]$, and $[b_0, b_1, \dots, b_n]$ are coefficients with respect to the first order Hermite polynomial basis. Applying the principle of orthogonality and equalities of Gaussian distributions, the zero moment can be computed with the following equation:

$$\begin{bmatrix} g_0 & g_1 & g_2 & \cdots & g_i & \cdots & g_n \\ g_1 & g_0 & 0 & \cdots & 0 & \cdots & 0 \\ g_2 & 0 & g_0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_i & 0 & 0 & \cdots & g_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_n & 0 & 0 & \cdots & 0 & \cdots & g_0 \end{bmatrix} \begin{bmatrix} a_{m0} \\ a_{m1} \\ a_{m2} \\ \vdots \\ a_{mi} \\ \vdots \\ a_{mn} \end{bmatrix} - \begin{bmatrix} B \\ 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix} = 0 \quad (3.24)$$

Once the zero moment was computed, the $2q^{\text{th}}$ moment can be evaluated from $(2q-1)^{\text{th}}$ moment recursively with the following equation:

$$\begin{bmatrix} g_0 & g_1 & g_2 & \cdots & g_i & \cdots & g_n \\ g_1 & g_0 & 0 & \cdots & 0 & \cdots & 0 \\ g_2 & 0 & g_0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_i & 0 & 0 & \cdots & g_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_n & 0 & 0 & \cdots & 0 & \cdots & g_0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_i \\ \vdots \\ a_n \end{bmatrix} + \begin{bmatrix} c_0 & c_1 & c_2 & \cdots & c_i & \cdots & c_n \\ c_1 & c_0 & 0 & \cdots & 0 & \cdots & 0 \\ c_2 & 0 & c_0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_i & 0 & 0 & \cdots & c_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_n & 0 & 0 & \cdots & 0 & \cdots & c_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_i \\ \vdots \\ b_n \end{bmatrix} = 0 \quad (3.25)$$

In the last step, the variational poles and residues via Monte Carlo methods were evaluated by using modified Krylov subspace projection methods, as mentioned in the earlier part of this section.

3.4 CHAPTER SUMMARY

In this chapter, the statistical model order reduction technique, called Statistical Spectrum Model Order Reduction (SSMOR) was discussed. This technique is capable of considering both intra-die and inter-die process variations with spatial correlations for interconnect circuits. This model order reduction technique generates order-reduced variational models from the given variational interconnect circuits. The reduced model can be used for fast statistical performance analysis of interconnect circuits, for instance, on-chip power grid and clock distribution networks with variational input power sources.

Furthermore, the general way to consider spatial correlation was by means of mapping the correlated random variables onto a set of independent and uncorrelated variables. In consideration of spatial correlation in connection with this model order reduction technique, an orthogonal mapping based on principal component analysis is applied to eliminate the correlation among variational random variables. This SSMOR algorithm consists of several important techniques, including the statistical spectrum analysis method, Monte Carlo sampling method, and modified Krylov subspace model order reduction technique. The purpose was to generate the statistical reduced model for faster interconnect simulation and optimization.

CHAPTER 4

PROPOSED ALGORITHM

4.1 ORIGINAL DYNAMIC SYSTEM

Consider a multiport dynamic linear time variant RLC system, the modified nodal analysis was applied to formulate into the state space equations,

$$C \frac{dx(t)}{dt} = -Gx(t) + Bu(t),$$
$$y(t) = L^T x(t). \quad (4.1)$$

Where G and C are $n \times n$ conductance and susceptance matrices, B and L are the $n \times N$ input and output positions matrices, and, typically, $B = L$ or $B = -L$; N is the number of input or output ports.

But, for the convenience it is represented in the for of most general state space representation of a linear time variant system with p inputs, q outputs and n state variables is written in the following form:

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t),$$
$$y(t) = L^T x(t). \quad (4.2)$$

Where $x(t)$ is called the "state vector", $y(t)$ is called the "scalar output", $u(t)$ is called the "scalar input", A is the "state matrix", B is the "input vector", C is the "output vector".

This ordinary differential is shown in detail below in the matrix form,

$$dt \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 2 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} u(t),$$

$$y(t) = [C_1 \quad \dots \quad C_n] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}. \quad (4.3)$$

The values $x_1(t), \dots, x_n(t)$ represent the N states. The output $y(t)$ is the weighted combination of those states at each point in time. Here, the main goal is to match the input –output relationship of the system with N states and the system with q states, where $q \ll N$.

4.2 EIGEN ANALYSIS

Consider, the input of the system doesn't vary with time. It means, that the system was being verified for the steady state condition. The time derivative of the state vector

$x_1(t), \dots, x_n(t)$ is zero i.e. $dt \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = 0$. Assuming this condition, the ordinary differential

equation can be written as,

$$0 = Ax + bu \quad (4.4)$$

The output of the system can be written as,

$$y = -C^T A^{-1} bu. \quad (4.5)$$

By inverting the A matrix the interior states were eliminated making the system much simpler. Now, eigendecomposition was applied on the matrix A , which results in the following,

$$A = \begin{pmatrix} \vdots & \vdots & \vdots \\ E_1 & \cdots & E_N \\ \vdots & \vdots & \vdots \end{pmatrix} \begin{pmatrix} \lambda_1 & \cdots & 0 \\ 0 & \ddots & \vdots \\ 0 & \cdots & \lambda_N \end{pmatrix} \begin{pmatrix} \vdots & \vdots & \vdots \\ E_1 & \cdots & E_N \\ \vdots & \vdots & \vdots \end{pmatrix}^{-1} \quad (4.6)$$

Where A is written as the product of matrix containing its eigenvectors, eigenvalues and inverse of the eigenvectors. Also, the states $x(t)$ are written in the terms of modes. This was explained below,

Changing variables, $Ew(t) = x(t)$ into $w(t) = E^{-1}x(t)$. The differential equation system is written in terms of modes as below,

$$\vec{E}_1 w_1(t) + \vec{E}_2 w_2(t) + \cdots + \vec{E}_n w_n(t) = x_1(t), \dots, x_n(t). \quad (4.7)$$

Now substituting $Ew(t) = x(t)$ in the ordinary differential equation, it was rewritten as,

$$\frac{dEw(t)}{dt} = AEw(t) + bu(t). \quad (4.8)$$

The equation above was multiplied by E^{-1} , which results as,

$$dw(t) / dt = E^{-1}AEw(t) + E^{-1}bu(t). \quad (4.9)$$

The resultant was a new decoupled equations and can be written as,

$$\frac{d}{dt} \begin{pmatrix} w_1 \\ \vdots \\ w_N \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 & \mathbf{0} \\ 0 & \ddots & 0 \\ \mathbf{0} & 0 & \lambda_N \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_N \end{pmatrix} + \begin{pmatrix} (E^{-1}b)_1 \\ \vdots \\ (E^{-1}b)_N \end{pmatrix} u(t). \quad (4.10)$$

The new decoupled equation in words can be written as the time derivative of modes was equal to the product of diagonal matrix containing eigenvalues λ and the

modes, in addition to the new input vector $E^{-1}b$. This gives separately scaled differential equations for the eigenmodes. The modes with weights $w(t)$ can be easily rewritten as $x(t)$, by multiplying each mode with the associated eigenvector. Now the output $y(t)$ can be calculated by the weighted combination of $w(t)$.

In the eigenmode, the input affects each of the states. But, they don't affect each other anymore and the output was the weighted combination of the completely independent states and the weight of the new input vector \tilde{b} i.e. $\tilde{b} = E^{-1}b$ [15]. \tilde{b} was nothing but the projection of the input vector b onto the eigenmode. The output vector was also rewritten as $\tilde{c} = CE^T$, which is nothing but the projection of the output vector C on the eigenvector space.

And the output equation will be,

$$y(t) = C^T x(t) = C^T E w(t) = (E^T C)^T w(t) = \tilde{c}^T w(t). \quad (4.11)$$

The eigenmode behavior is equal to the convolution of the exponential of the eigenvalue λ_i to the new input vector \tilde{b}_i and the input $u(\tau)$. This is written as,

$$w_i(t) = \int_0^t e^{\lambda_i(t-\tau)} \tilde{b}_i u(\tau) d\tau. \quad (4.12)$$

The output $y(t)$ is the weighted combination of those mode and is written as,

$$y(t) = \sum_{i=1}^N \tilde{c}_i w_i(t). \quad (4.13)$$

Considering the new decoupled equations, they are written as,

$$\begin{aligned}
\frac{d}{dt} w_1 &= \lambda_1 w_1 + \tilde{b}_1 u(t) \\
&\vdots \\
\frac{d}{dt} w_n &= \lambda_n w_n + \tilde{b}_n u(t)
\end{aligned} \tag{4.14}$$

If \tilde{b} is large for one of these modes, the respective mode was large or else respective mode will not show up. If $\tilde{b}_i = 0$ the mode was uncontrollable, this means that the mode was not affected by the input. In the output $y(t)$, if $\tilde{c}_i = 0$, the mode was unobservable, this means that the mode was not seen at the output. Also, if the product of $\tilde{b}_i \tilde{c}_i$ was small, those modes were almost unobservable and uncontrollable. From the above assumptions the terms that satisfy those conditions are neglected. This resulted in a reduced mode decoupled equations as below,

$$\begin{pmatrix} \dot{w}_1 \\ \vdots \\ \dot{w}_N \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 & \mathbf{0} \\ 0 & \ddots & 0 \\ \mathbf{0} & 0 & \lambda_q \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_q \end{pmatrix} + \begin{pmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_q \end{pmatrix} u(t) \tag{4.15}$$

The reduced output equation was written as,

$$y(t) = \sum_{i=1}^q \tilde{c}_i w_i(t) \tag{4.16}$$

Here, the eigenvalues λ we re sorted according to bubble sort algorithm. By, considering the weights of bi and the values of λ , the values of λ were represented in the order:

$$\lambda_1 > \lambda_2 > \lambda_3 > \lambda_4 \dots > \lambda_q \tag{4.17}$$

Physically, large eigenvalues (λ) represent dominate responses. The terms with dominate response were kept and ignore small ones to reduce the complexity of modified nodal analysis as our first step in the proposed algorithm.

On the other hand the least negative eigenvalues were preserved. From the equation representing the eigenmode behavior, it was rewritten as,

$$w_i(t) = \int_0^t e^{\lambda_i(t-\tau)} \tilde{b}_i u d\tau = \frac{1}{\lambda_i} (\tilde{b}_i u - \tilde{b}_i u e^{\lambda_i t}) \quad (4.18)$$

At a constant input the eigenmode with most negative eigenvalues tends to decay faster than the eigenmode with the least negative eigenvalue. So, eigenmodes with least negative eigenvalues were called as slow modes [5]. Form the equation above the output is scaled the value $\frac{1}{\lambda_i}$. If the $|\lambda_i|$ is large then the value of the mode w_i will be small.

If $\tilde{b}_{k+1}, \dots, \tilde{b}_N$ and $\tilde{c}_{k+1}, \dots, \tilde{c}_N$ are large, it will cause inaccuracy under the assumption that \tilde{b}_i was not considered in the first step. This is the primary reason that, bubble sort was used to consider the weight of b_i in the first step of the proposed algorithm

4.3 BUBBLE SORT ALGORITHM

Bubble sort is a simple sorting algorithm. It works by repeatedly stepping through the list to be sorted, comparing two items at a time and swapping them if they are in the wrong order. The pass through the list was repeated until no swaps are needed, which indicates that the list was sorted [22]. The algorithm gets its name from the way smaller elements "bubble" to the top of the list. Because it only uses comparisons to operate on

elements, it was a comparison sort. Bubble sort has worst-case complexity $O(n^2)$, where n is the number of items being sorted.

Let us take the array of numbers "5 1 4 2 8", and sort the array from lowest number to greatest number using bubble sort algorithm. In each step, elements written in bold are being compared.

First Pass:

(5 1 4 2 8) \rightarrow (1 5 4 2 8) Here, algorithm compares the first two elements, and swaps them.

(1 5 4 2 8) \rightarrow (1 4 5 2 8)

(1 4 5 2 8) \rightarrow (1 4 2 5 8)

(1 4 2 5 8) \rightarrow (1 4 2 5 8) Now, since these elements are already in order, algorithm does not swap them.

Second Pass:

(1 4 2 5 8) \rightarrow (1 4 2 5 8)

(1 4 2 5 8) \rightarrow (1 2 4 5 8)

(1 2 4 5 8) \rightarrow (1 2 4 5 8)

(1 2 4 5 8) \rightarrow (1 2 4 5 8)

Now, the array is already sorted, but our algorithm does not know if it is completed. Algorithm needs one whole pass without any swap to know it is sorted.

Third Pass:

(1 2 4 5 8) \rightarrow (1 2 4 5 8)

(1 2 4 5 8) \rightarrow (1 2 4 5 8)

(1 2 4 5 8) \rightarrow (1 2 4 5 8)

(1 2 4 5 8) → (1 2 4 5 8)

Finally, the array was sorted, and the algorithm can terminate.

4.4 IMPROVEMENT TO THE PROPOSED ALGORITHM

First, Laplace transformation properties are applied on the ordinary differential equation

$$\begin{aligned}\frac{dx(t)}{dt} &= Ax(t) + bu(t), \\ y(t) &= C^T x(t).\end{aligned}\tag{4.19}$$

Where u is the input variable, y is the output variable, A is a $n \times n$ matrix and b is a $n \times 1$ matrix. x is a $n \times 1$ vector of state variables. After applying the Laplace properties given in (2.10) and (2.11) the ordinary differential equation was written as,

$$Y(s) = C^T (sI - A)^{-1} bU(s)\tag{4.20}$$

Where, Transfer function $H(s)$

$$H(s) = C^T (sI - A)^{-1} b\tag{4.21}$$

Also $H(s)$ can be written in the form of rational transfer function as below,

$$H(s) = \frac{b_1 + b_2s + \dots b_{n-1}s^{n-1}}{1 + a_2s + \dots a_n s^n}\tag{4.22}$$

Upon applying Eigen decomposition on $H(s)$ the resultant transfer function was written as,

$$H(s) = C^T E (sI - A)^{-1} E^{-1} b\tag{4.23}$$

Where $\tilde{c} = C^T E$ and $\tilde{b} = E^{-1} b$.

Substituting \tilde{c} and \tilde{b} in the above equation. The resultant transfer function $H(s)$

was written as,

$$H(s) = \tilde{c} \begin{bmatrix} \frac{1}{s - \lambda_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{s - \lambda_n} \end{bmatrix} \tilde{b}, \quad H(s) = \sum_{i=1}^n \frac{\tilde{c}_i \tilde{b}_i}{s - \lambda_i} \quad (4.24)$$

However, direct eigendecomposition will pay a huge calculation penalty if G and C matrixes from the MNA formulated RLC network are large. Large G and C matrixes obviously means more number of nodes and eigenvalues to be computed. From the concept of projection based pole computation, it is learned that at higher orders the eigenvalues were linearly dependent and most of the pole information is lost. Instead, the transfer function $H(s)$ in the pole-residue form is considered. This is represented as

$$H(s) = \frac{\tilde{c}_1 \tilde{b}_1}{s - \lambda_1} + \frac{\tilde{c}_2 \tilde{b}_2}{s - \lambda_2} + \dots + \frac{\tilde{c}_n \tilde{b}_n}{s - \lambda_n} \quad (4.25)$$

The idea here was to define a constant as the key in the sorting algorithm, say $c_N b_N$ as one of correlations in order to sort the order of the eigenvalues in the reduced decoupled equations shown below,

$$\begin{pmatrix} \dot{w}_1 \\ \vdots \\ \dot{w}_N \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 & \mathbf{0} \\ 0 & \ddots & 0 \\ \mathbf{0} & 0 & \lambda_q \end{pmatrix} \begin{pmatrix} w_1 \\ \vdots \\ w_q \end{pmatrix} + \begin{pmatrix} \tilde{b}_1 \\ \vdots \\ \tilde{b}_q \end{pmatrix} u(t) \quad (4.26)$$

To re-arrange the $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_q$, they were sorted according to the correlation of that constant and λ_i . From the eigendecomposition it was known that the input vector \tilde{b} affects the modes, written in the form of product of eigenvalues and the mode vector. If

the correlation coefficient was positive, it represents that the constant key and the eigenvalue were linearly dependent. This applies even for the correlation of the small value of $c_N b_N$ and λ_i . This algorithm demonstrates that the lower residue values, which were the resultants from the higher order of frequency, are also taken into account in the calculation of the dominant poles. From rational transfer function, it was demonstrated that a faster and lower calculation penalty is achieved by finding the dominated poles in the proposed algorithm. As a result, after keeping all the dominate poles, the output functions were clustered in order to reduce the redundant poles. Thus the reduced transfer function was written as,

$$H_r(s) = \frac{b_0^r + b_1^r s + \dots + b_{q-1}^r s^{q-1}}{1 + a_1^r s + \dots + a_q^r s^q} \quad (4.27)$$

Where, $H_r(s)$ was matched with the $H(s)$ using only $2q$ states

4.5 CHAPTER SUMMARY

In this chapter, the new model order reduction technique using eigenmode analysis and rational transfer function fitting was discussed. The RLC interconnect circuits were written in the form of state space equations using modified nodal analysis. Eigendecomposition was applied on these equations. Modes that were unobservable and uncontrollable were truncated. The resulting modes were then written in the pole/residue form. A constant key was defined and the dominant eigenvalues representing the $2q$ poles of the reduced system are obtained by sorting those eigenvalues obtained by correlating the key and the eigenvalues. This response obtained from $2q$ poles was matched with the response of the original system.

CHAPTER 5

PROPOSED ALGORITHM TESTING AND VALIDATION

5.1 SIMULATION RESULTS

All experimental data were measured in a Microsoft Windows desktop with 1.9GHz CPU and 2GB memory using Matlab R2007a. The results of the proposed algorithm were compared with PRIMA. Table-1 summarizes the runtime results with RC benchmarks. RT indicates the computation run time in seconds. As it was seen, the improvement on accuracy was significant, roughly 90% more accurate than PRIMA.

Node #	Frequency	PRIMA		Proposed Algorithm	
		Error	RT(s)	Error	RT(s)
1208	$0 - 5 \times 10^{10}$ Hz	17%	132	0.1%	342
4800	$0 - 5 \times 10^{10}$ Hz	33%	203	1.4%	688
12000	$0 - 5 \times 10^{10}$ Hz	47%	437	2.3%	1427
100000	$0 - 5 \times 10^{10}$ Hz	49%	688	3.3%	1930

Table 1 Runtime (RT) and accuracy testing result in proposed algorithm and PRIMA (RC mesh)

It was evident from the table that the accuracy was increased. But when coming to the runtime in seconds, the proposed model has more runtime than PRIMA. In the proposed model runtime time was increased up to 3X times when compared to PRIMA. Because, as the constant key was chosen, correlation had been performed between the eigenvalue and constant key. Then the eigenvalues were sorted out using bubble sort

algorithm. The sorting algorithm like bubble sort have bad time complexity and the runtime was increased.

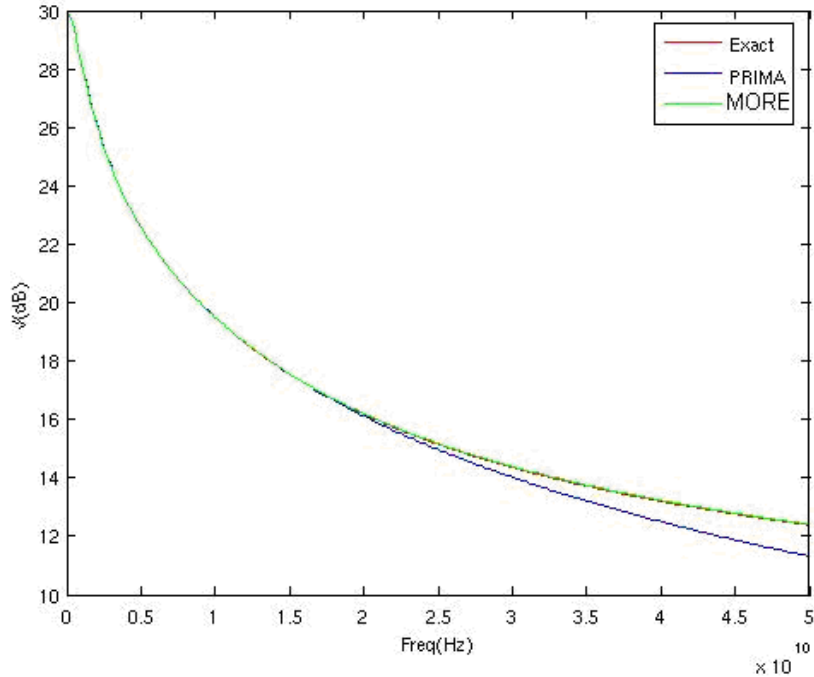


Figure 9. Proposed algorithm vs. PRIMA with RC benchmark of 4800 nodes in wide frequency band

The Figure-9 was the response of a RC mesh with a bench mark of 4800 nodes. The output was taken at the node: 4352 and the edge: 5736 and at the port number: 1157. The original size of the circuit was given as 4422×4422 moments. After it was processed by PRIMA, it was reduced to 1157×1157 matched moments. The size of the proposed model was 121×121 matched moments. It was observed that there was significant decrease in the number of matched moments and the accuracy of the proposed model clearly dominated the accuracy of the PRIMA with the original circuit. The main reason for this decrease of moments using the proposed models was that the dominant poles were being identified much efficiently by using the correlation of the key, which

constitutes the projected input vector onto the eigenmode along with the weighted combination of the projected output vector and the eigenvalues of the state matrix. Thus, by this concept the lower residues from the higher order frequency were identified and accuracy of the proposed model was increased when compared to PRIMA, which does not consider the residues at higher order frequency.

Node #	Frequency	PRIMA		Proposed Algorithm	
		Error	RT(s)	Error	RT(s)
1208	$0 - 5 \times 10^{10}$ Hz	60%	214	0.4%	437
4800	$0 - 5 \times 10^{10}$ Hz	77%	383	3.4%	597
12000	$0 - 5 \times 10^{10}$ Hz	77%	497	4.8%	1734
100000	$0 - 5 \times 10^{10}$ Hz	89%	644	6.1%	2840

Table 2 Runtime (RT) and accuracy testing result in proposed algorithm and PRIMA (RLC mesh)

Table-2 summarizes the runtime results, and the runtime results were compared for different RLC benchmarks. The accuracy improvement was also significant, over 95%. But, at the cost of increased runtime up to 3X times. The Figure-10 shows the frequency domain response of a RLC mesh at 600 nodes. At the node: 600, edge: 76, Port number: 110, the original size if the mesh is 988×988 moments. After the mesh was processed through PRIMA the number of matched moments was given by 220×220 . The number of matched moments for the proposed method was given by 152×152 . The proposed method was working great to identify the dominant poles with state matrix containing large conductance and susceptance values.

The proposed algorithm matched the original response exactly, while PRIMA method showed a significant inaccuracy. The tested results show the proposed algorithm was more accurate than PRIMA in wide frequency environments. The error (%), is the error calculated between the actual response of the system considering all the states and the response of the reduced order system, was reduced moderately.

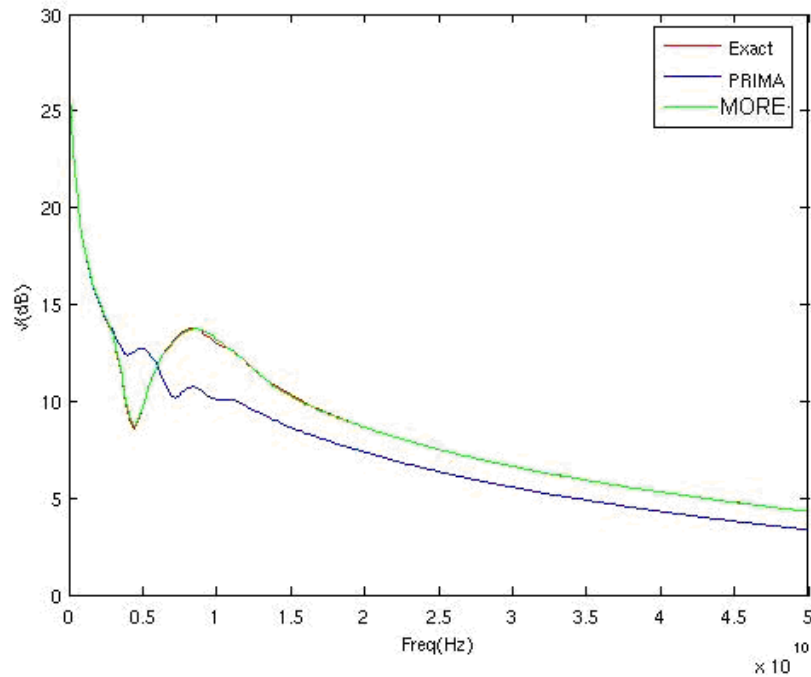


Figure 10. Proposed algorithm vs. PRIMA with benchmark of RLC circuits

5.2 CHAPTER SUMMARY

In this chapter, the results of the proposed algorithm were compared with PRIMA. The simulation was done using Matlab R2007a. The results of RC and RLC meshes obtained from the proposed algorithm were compared individually with results of PRIMA. It was observed that there was 90% increase in the accuracy. Also, this accuracy was obtained with less number of poles in comparison with the number poles compared using PRIMA.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

An extended model order reduction technique using rational transfer function fitting and eigenmode analysis order analysis of linear circuits with a large number of independent sources has been presented. In this method the concept of eigenmode analysis like representing state vector in the form of eigenmodes was used. Also, the concept of representing the transfer function in pole-residue form was obtained from the projection framework. The constant key was defined and the dominant eigenvalues representing the $2q$ poles of the reduced system were obtained by sorting those eigenvalues obtained and correlating the key with the eigenvalues.

The Passive Reduced Order Interconnect Macromodeling Algorithm (PRIMA), which was being compared, has some disadvantages like,

- The time complexity of PRIMA was proportional to the number of ports of the circuits as moments excited by every port need to be computed and matrix valued transfer functions were generated.
- The poles of the reduced models were increased linearly with the number of ports, and this makes the reduced models much larger than necessary.
- Generating more poles does not always help to improve the accuracy of the reduced models, as more block moments were not always matched as the numbers of poles were increased.

- The moments generated were very much linearly dependent and they generate ill conditioned matrices. Each new moment is the resultant of the scaled old moment with the dominant eigenvalue λ . The higher order moments will become less accurate. Thus resulting, PRIMA inaccuracy at higher frequencies.

The proposed algorithm has the advantages like,

- The concept of moments was not used in this algorithm. This helped us to eliminate the problem of generating ill conditioned matrices.
- The concept of using the constant key and correlation of that constant key and λ helped to sort the values of λ_i . Then these eigenvalues in the pole residue form were written in the form of rational transfer function. By this concept a faster and lower calculation penalty was demonstrated for finding the dominated poles in the proposed algorithm.

By the following advantages, this method can handle a much more wide frequency than the existing algorithms accomplish.

Experimental results showed that the proposed method achieved a moderate accuracy improvement over circuit simulation with wide frequency variations and reduced up to 90% errors compared to the existing model order reduction algorithm like PRIMA. . But, the runtime of the proposed model order technique was costing 3X times more when compared to runtime of PRIMA. This was primarily because of the state space representation of the larger matrices and also the sorting algorithm that was being used had very low time complexity.

In future this work can be extended by implementing new way of formulating the RLC mesh state space equations and introducing new sorting algorithms. Also, this method should be improved for the model order reduction of nonlinear dynamic systems.

LIST OF REFERENCES

- [1] Sheldon X.-D. Tan and Lei He, "Advanced Model Order Reduction Techniques in VLSI Design", Cambridge University Press, ISBN 0-521-86581-6, pg 5-31 ,2007.
- [2] Raymond A. Decarlo, Pen-min, "Linear Circuit Analysis", Volume one, Prentice Hall, ISBN 0-13-043134-6, pp 102 – 108, 1996.
- [3] Sheldon X-D. Tan, Chung –Kuan Cheng, Zhanhai Qin, "Symbolic Analysis and Reduction of VLSI Circuits", Springeronline.com, ISBN-0-387-23904-9, pg 16-24, 2005.
- [4] Gabriela Ciuprina, Daniel Ioan, " Scientific Computing in Electrical Engineering", Springer Berlin Heidelberg, ISBN 978-3-540-71979-3, pg 139-152,2007.
- [5] Guoyong Shi and C.J.Richard Shi, "Model-Order Reduction by Dominant Subspace Projection: Error Bound, Subspace Computation, and Circuit Applications," IEEE transaction on Circuits and Systems-1: Regular papers, Vol.52, No.5, May 2005.
- [6] Zu-Quing Qu, "Model Order Reduction Techniques: with Applications in Finite Element Analysis", 1st edition, Springer Publications, Chpt-9, pages 237-246, August 2005.
- [7] Guoyong Shi, Bo Hu, and C.-J. Richard Shi, "On Symbolic Model Order Reduction", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol.25, No.7, July 2006
- [8] Bradley N Bond, Luca Daniel. "Parameterized Model Order Reduction of linear dynamical systems", Proceedings of the IEEE Conference on Computer-Aided Design, November, 2005.
- [9] Michal Rewien´ski and Jacob White, "A Trajectory Piecewise-Linear Approach To Model Order Reduction and Fast Simulation of linear Circuits a Micromachined Devices", IEEE transactions on Computer-aided design of Integrated circuits and Systems, Vol. 22, No.2, February 2003.
- [10] Janet Meiling Wang, Chia-Chi Chu, Qingjian Yu, and Ernest S. Kuh, "On Projection-Based Algorithms for Model-Order Reduction of Interconnects," IEEE Transaction on Circuits and Systems - 1: Fundamental theory and applications Vol.49, No.11, November 2002.

- [11] Z. Bai, "Krylov subspace techniques for reduced-order modeling of large-scale Dynamical Systems," *Applied Numerical Mathematics*, Vol. 43, pp. 9-44, May 2002.
- [12] L. M. Silveira, "Characterizing substrate coupling in deep-submicron designs," *IEEE Design & Test of Computers*, March-April 2002.
- [13] A. Odabasioglu, M. Celik, L. T. Pileggi, "PRIMA Passive Reduced-order Interconnect Macromodeling Algorithm," 34th DAC, pp. 58-65, 1997.
- [14] R. Achar, P. K. Gunupudi, M. Nakhla, and E. Chiprout, "Passive interconnect reduction algorithm for distributed/measured networks," *IEEE Transactions on Circuits and Systems: Analog and Digital Signal Processing*, 47,(4),287-301, April 2000.
- [15] Lecture 1: "Why Model order reduction?"
<http://web.mit.edu/mor/> date cited 05/20/2008
- [16] R. Ghanem. "The nonlinear Gaussian spectrum of log-normal stochastic processes and variables". *Journal of Applied Mechanics*, 66:964{973, December 1999.
- [17] R. G. Ghanem and P. D. Spanos. "Stochastic Finite Elements: A Spectral Approach". Dover Publications, 2003.
- [18] S. R. Nassif and J. N. Kozhaya. "Fast power grid simulation". In *Proc. Design Automation Conf. (DAC)*, pages 156-161, 2000.
- [19] H. Chang and S. Sapatnekar. "Statistical timing analysis under spatial correction". *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 24(9):1467{1482, Sept. 2005.
- [20] I. T. Jolliffe. "Principal Component Analysis". Springer-Verlag, 1986.
- [21] L. T. Pillage and R. A. Rohrer. "Asymptotic waveform evaluation for timing Analysis". *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, pages 352{366, April 1990.
- [22] K. Chaudhary and P. Robinson, "Channel Routing by Sorting." *IEEE Trans. Computer-Aided Design* Vol. 10, pp. 754-760, June 1991.
- [23] Jeffrey Fan, "Process variation-aware interconnect simulation and optimization in VLSI design", PhD dissertation, Department of Electrical Engineering, University of California, Riverside, June 2007