A SIMULATION PROGRAM FOR EFFICIENT
ANALYSIS OF LINEAR CIRCUITS

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND

ELECTRONICS ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Mustafa Sungur

September 1996

A SIMULATION PROGRAM FOR EFFICIENT ANALYSIS OF LINEAR CIRCUITS

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND

ELECTRONICS ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

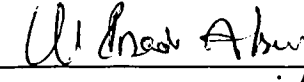FOR THE DEGREE OF

MASTER OF SCIENCE

By

Mustafa Sungur

September 1996

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.
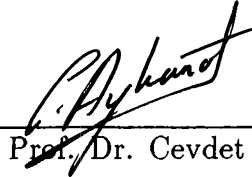
—————————————————
Prof. Dr. Abdullah Atalar(Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

—————————————————
Assoc. Prof. Dr. M. İrşadi Aksun

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

—————————————————
Assoc. Prof. Dr. Cevdet Aykanat

Approved for the Institute of Engineering and Sciences:

—————————————————
Prof. Dr. Mehmet Baray
Director of Institute of Engineering and Sciences

ii

ABSTRACT

A SIMULATION PROGRAM FOR EFFICIENT ANALYSIS OF LINEAR CIRCUITS

Mustafa Sungur
M.S. in Electrical and Electronics Engineering
Supervisor: Prof. Dr. Abdullah Atalar
September 1996

A circuit simulation program using generalized asymptotic waveform evaluation technique is introduced. The program analyzes circuits with lumped and distributed components. It computes the moments at a few frequency points and extracts the coefficients of an approximating rational by employing one of the two different methods. One of the examined methods is proposed to compare the accuracy of results and the execution times with conventional simulators and several examples are demonstrated, indicating that our simulator provides a speed improvement without a significant loss of accuracy.

Keywords : Circuit Simulation, Asymptotic Waveform Evaluation, Multi point Padé Approximation, AC Analysis, MAWE, Spice, Computer Aided Design, CAD

ÖZET

LİNEER DEVRELERİN VERİMLİ ANALİZİ İÇİN BİR BENZETİM PROGRAMI

Mustafa Sungur
Elektrik ve Elektronik Mühendisliği Bölümü Yüksek Lisans
Tez Yöneticisi: Prof. Dr. Abdullah Atalar
Eylül 1996

Genelleştirilmiş asimtotik dalgaşekli hesaplaması tekniğini kullanan bir devre benzetim programı sunulmuştur. Program, dağılmış ve ortak parametreli devrelerin analizini yapar. Bu program, devrenin momentlerini birkaç frekans noktasında hesaplar ve kesirli yaklaşım fonksiyonunun katsayılarını anlatılan iki metottan birisini kullanarak ortaya çıkarır. Denenen metotlardan biri, sonuçların doğruluğu ve çalışma zamanı bakımından bilinen benzetim programları ile karşılaştırmak için önerilmiş ve bizim benzetim programımızın doğruluğundan birşey kaybetmeden, zamanda iyileştirme yaptığını gösteren bazı örnekler verilmiştir.

Anahtar Kelimeler : Devre Benzetimi, Asimtotik Dalgaşekli Hesaplama, Çok noktalı Padé Yaklaşımı, AC Analiz, MAWE, Spice, Bilgisayar Destekli Tasarım, CAD

ACKNOWLEDGEMENT

I would like to express my deep gratitude to my supervisor Dr. Abdullah Atalar for his supervision, guidance and suggestions throughout the development of this work.

I would like to thank A. Suat Ekinci for his colloboration, numerous suggestions and invaluable help. I would also like to thank previous members of the CAD group at Bilkent University; Mustafa Yazgan, Ogan Ocalı, Mustafa Çelik and Satılmış Topçu for their useful previous work.

Finally, many thanks to my friends for their moral support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

To my family...

Chapter 1

INTRODUCTION

Accurate simulation of VLSI circuits is an expensive task for the large circuit sizes of today. With the advances in integrated circuit technology, the physical circuit sizes are reduced and the operating speeds are increased. Shrinking device sizes and increasing operating speeds require faster circuit simulation programs which do not trade execution time for accuracy. Spice-like programs with high accuracy are needed for intensive verification and design of VLSI circuits, but for reducing execution times, new circuit solving algorithms were introduced. While Spice-like simulators predict the behavior of the circuit at a large number of discrete points both in frequency and time domain analysis, most of the new simulators employ faster algorithms to solve the circuit matrix at lower number of points. The drawback of these algorithms is the loss of accuracy, and the effort is to reduce the execution time without losing much accuracy.

Asymptotic Waveform Evaluation (AWE) technique[1], is used in some new simulators in order to reduce the execution time of the simulation. Instead of solving the circuit at many discretized points, AWE seeks to capture the behavior by approximating the dominant poles of the circuit with a lower order model. The reduced order model is matched to the moments of the linear circuit, which are obtained from the Taylor series expansion of the circuit response around $s = 0$. Since the information carried by the moments is accurate

at low frequency region, the AWE technique will be efficient in extracting the low frequency poles of the circuit. At relatively higher frequencies the AWE technique becomes inefficient and several methods are proposed to improve AWE's accuracy. AWE is extended to handle distributed elements [2, 3] in order to analyze circuits that cannot be modeled by only lumped components. Also, Laurent series expansion ($s = \infty$) is added to improve the accuracy of transient analysis in the vicinity of $t = 0$ [4]. The stability of approximations is improved by manipulating the moment matching techniques [5].

Recently, the Complex Frequency Hopping technique is introduced in order to find all of the dominant poles of the circuit in a frequency range of interest [6]. The PVL algorithm, Padé Approximation via Lanczos Process, is introduced to provide high numerical stability to the Padé Approximants [7].

In the recent past, a multi point Padé Approximation was proposed [8] for analysis of interconnect networks with transmission lines not only in low frequencies but also in high frequency regions. Apart from the moments at $s = 0$ (DC), this method uses shifted moments as well. This property provides the necessary information about all frequency range. This approach requires the solution of the circuit matrix at several frequency points determined by the complex frequency hopping technique.

In this study, we introduce a simulation program for multi point Padé approximation of linear circuits. We compute the frequency shifted moments at several expansion points and match those to a lower order approximating rational by using two methods. The program is implemented in C++ language running on UNIX and uses the moment matching algorithms given in [8]. The theoretical background of the work and the methods are introduced in the next 3 chapters. The methods introduced are compared with Spice simulators in respect to their accuracy and execution times. Beginning from Section V, we present the simulations and computational results on several examples to demonstrate the efficiency of the proposed simulator.

Chapter 2

ASYMPTOTIC WAVEFORM EVALUATION

2.1 Linear Circuits and AWE

The asymptotic waveform evaluation is an approximation technique used for representing the behavior of a linear circuit. The approximation is achieved by extracting some s-domain properties of the circuit and matching them to a reduced (qth) order model of the original response. In this section, we briefly outline the basic properties of AWE. If we consider state equations for a linear circuit,

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u$$
$$y = \mathbf{c}^{\mathrm{T}}\mathbf{x} + Du \tag{2.1}$$

where the entries stand for;

$\mathbf{x}$: $n-$dimensional column vector (state vector)

$\mathbf{A}$: $n \times n$ state matrix

$\mathbf{b}$ n−dimensional vector coupling input to states

y : output variable

$\mathbf{c}$ n−dimensional vector of states

D : scalar for expressing the effect of input on output

D can be neglected for simplicity. The zero state impulse response of the linear circuit is defined as [9]

$$H(s) = \mathbf{c}^{\mathrm{T}}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}, \qquad (2.2)$$

which can be expanded into Taylor series around $s = 0$:

$$\begin{aligned}
H(s) &= -\mathbf{c}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{b} - \mathbf{c}^{\mathrm{T}}\mathbf{A}^{-2}\mathbf{b}s - \cdots - \mathbf{c}^{\mathrm{T}}\mathbf{A}^{-j-1}\mathbf{b}s - \cdots \\
&= \textstyle\sum_{j=0}^{\infty} -\mathbf{c}^{\mathrm{T}}\mathbf{A}^{-j-1}\mathbf{b}s^j \qquad (2.3) \\
&= \textstyle\sum_{j=0}^{\infty} m_j s^j
\end{aligned}$$

where

$$m_j \equiv -\mathbf{c}^{\mathrm{T}}\mathbf{A}^{-i-1}\mathbf{b}, \qquad \text{for } i = 1 \geq 0. \qquad (2.4)$$

2.2   Computation of moments

It can be shown that the m_i's are the moments of $h(t)$ and they can be computed using the following recursion:

$$\begin{cases}
\mathbf{x_0} &= -\mathbf{A}^{-1}\mathbf{b} \\
\mathbf{x_j} &= \mathbf{A}^{-1}\mathbf{x_{j-1}} \\
m_j &= \mathbf{c}^{\mathrm{T}}\mathbf{x_j}
\end{cases}$$

Above, $\mathbf{x_j}$ denotes the ith moment of the individual state variables. To start the recursion, we need to compute $\mathbf{x_0}$. This is realized by replacing the input source by a constant value of 1, the capacitors by current sources of value zero

4

and the inductors by voltage sources of value zero. This corresponds to $u = 1$ and $\dot{x} = 0$ in 2.1. The capacitor voltages and inductor currents, state variables, are found to be $\mathbf{A}^{-1}\mathbf{b}$. The value of the output is m_0. When computing higher order moments m_{j+1}, we use the preceding moments m_j. The input is set zero, a capacitor which is the ith state variable is replaced by a current source of value Cx_{ji}, and an inductor by an voltage source of value Lx_{ji}. This is equivalent to setting $u = 0$ and $\dot{x} = \mathbf{x_j}$. The new moments are the voltages across the independent current sources replaced for capacitors. and currents across the independent voltage sources replaced for inductors. New moments are computed according to $\mathbf{x} = \mathbf{A}^{-1}\mathbf{x_j}$. Computationally, finding $\mathbf{x_0}$ costs to an LU factorization and forward backward substitutions, while addition of each moment costs forward and backward substitutions only.

2.3 Order Reduction

In a linear system modeled in Laplace domain, we have the following equation,

$$\mathbf{T(s)x(s)} = \mathbf{w} \qquad (2.5)$$

where $\mathbf{T(s)}$ is the modified nodal analysis (MNA)[10] matrix of the circuit, with $\mathbf{x}$ and $\mathbf{w}$ the unknowns and excitation vectors, respectively. If the circuit contains lumped components only, i.e., $\mathbf{T} = \mathbf{T_1} + s\mathbf{T_2}$, the elements of the system matrix $\mathbf{T}$ are polynomials. With an output that is a linear combination of the unknowns vector $H(s) = \mathbf{c}^T\mathbf{x(s)}$, the impulse response becomes a rational:

$$H(s) = \frac{\sum a_i s^i}{\sum b_j s^j}$$

It is the objective of AWE to approximate the response of the high order network function with a lower order model. The approximation function is

$$\hat{H}(s) = \frac{b_0 + b_1 s + \cdots + b_{q-1} s^{q-1}}{1 + a_1 s + \cdots + a_q s^q}$$

and $\hat{H}(s)$ has similar characteristics to $H(s)$. Since, the aim is to find $\hat{H}(s)$, we have to find $2q$ coefficients of the approximating function. These coefficients

are obtained by matching the $2q$ moments to $\hat{H}(s)$ and this yields the following set of linear equations for a_i's. [1]

$$
\begin{bmatrix}
m_0 & m_1 & & m_{q-1} \\
m_1 & m_2 & \cdots & m_q \\
\vdots & & & \\
m_{q-1} & m_q & & m_{2q-2}
\end{bmatrix}
\begin{bmatrix}
a_q \\
a_{q-1} \\
\vdots \\
a_1
\end{bmatrix}
= -
\begin{bmatrix}
m_q \\
m_{q+1} \\
\vdots \\
m_{2q-1}
\end{bmatrix}
\tag{2.6}
$$

The b_j are computed from the following set of equations:

$$
\begin{aligned}
b_0 &= m_0 \\
b_1 &= m_0 a_1 + m_1
\end{aligned}
\tag{2.7}
$$

$$
b_{q-1} = m_0 a_{q-1} + m_1 a_{q-2} + \cdots + m_{q-1}
$$

The poles are found using the root finding algorithms from the denominator. The residues can be found from the poles and the moments with a scheme given in [1].

6

Chapter 3

MULTI POINT PADÉ APPROXIMATION

This chapter introduces both multi point moment generation and multi point moment matching techniques. The evaluated moments are used by two methods in order to perform the approximation. In the last section we review the Complex Frequency Hopping (CFH) technique for completeness.

3.1   Frequency Shifted Moments

The system response of a linear circuit in Eq. 2.5 is $x(s) = T^{-1}(s)w$ and can be written in Taylor series form around $s = s_k$.

$$x(s) = \sum_{i=0}^{\infty} x_{ki}(s - s_k)^i$$

where

$$x_{ki} = \frac{\frac{\partial^i}{\partial s^i}[T^{-1}]|_{s=s_k}}{i!}w.$$

7

In these equations, x_{ki} stands for the ith frequency shifted moments at $s = s_k$. The first moment set is simply the solution at that point

$$\mathbf{x_{k0}} = \mathbf{T}^{-1}(\mathbf{s_k})\mathbf{w}$$

The higher order moments can be computed recursively as

$$\mathbf{x_{ki}} = -\mathbf{T}^{-1}(\mathbf{s_k}) \sum_{r=1}^{i} \frac{\mathbf{T}^{(r)}(\mathbf{s_k})\mathbf{x_{k(i-r)}}}{\mathbf{r!}} \tag{3.1}$$

where $\mathbf{T}^{(r)}$ stands for the rth derivative of the $\mathbf{T}$ matrix with respect to s and evaluated at $s = s_k$. If the circuit has lumped components only, then $\mathbf{T^r} = 0$ for $r > 1$. Otherwise. the derivatives can be found using some methods proposed in literature [2, 11]. The frequency shifted moments of the output are obtained from the moment vectors $\mathbf{x_{ki}}$ using the linear equation

$$m_{ki} = \mathbf{c}^{\mathbf{T}}\mathbf{x_{ki}} \qquad i = 0, 1, \ldots, n_k - 1 \tag{3.2}$$

where n_k is the number of moments at $s = s_k$. So, we obtain

$$H(s) = m_{k0} + m_{k1}(s - s_k) + m_{k2}(s - s_k)^2 + \cdots$$

The moments at $s = 0$ (s_0) is denoted by m_{0i}, while m_{ki} and m_{-ki} represent the moments at $s = s_k(s_k)$ and $s = s_k^*(s_{-k})$, respectively. If the total number of moments is N, we have

$$\sum_{k=-n}^{n} n_k = n_0 + 2\sum_{k=1}^{n} n_k = N$$

where n is the number of expansion points in upper half plane.

3.2 Multi point Moment Matching

Similar to AWE methods, we are trying to find the qth order rational

$$
\begin{aligned}
\hat{H}(s) &= \frac{b_0 + b_1 s + \cdots + b_{q-1} s^{q-1}}{1 + a_1 s + \cdots + a_q s^q} = \\
&= m_{k0} + m_{k1}(s - s_k) + m_{k2}(s - s_k)^2 + \cdots + m_{k(n_k-1)}(s - s_k)^{n_k-1} \\
&\quad k = -n, \ldots, 0, \ldots, n
\end{aligned}
\tag{3.3}
$$

where $2q = N$ (N must be even) moments obtained at $n + 1$ expansion points.

In the following sections we describe two methods for calculation of the coefficients b_i and a_i directly from the moments.

3.2.1 Method I

For each expansion point $s = s_k$ we have the following equation:

$$\frac{b_0 + b_1 s + \cdots + b_p s^p}{1 + a_1 s + \cdots + a_q s^q} = m_{k0} + m_{k1}\hat{s} + \ldots + m_{k(n_k-1)}\hat{s}^{n_k-1} \qquad (3.4)$$

Here $\hat{s} = s - s_k$.

If we rewrite the left-hand-side of (3.4) we obtain

$$\hat{a}_i = \sum_{l=i}^{q} a_l \binom{l}{i} s_k^{l-i}, \qquad i = 0, 1, \ldots, q, a_0 = 1; \qquad (3.5)$$

$$\hat{b}_i = \sum_{l=i}^{p} b_l \binom{l}{i} s_k^{l-i}, \qquad i = 0, 1, \ldots, p$$

There are n_k constraints for choosing $p + q + 1$ unknowns. This gives the equation $\hat{\mathbf{b}} = \mathbf{B}\hat{\mathbf{a}}$, where

$$\hat{\mathbf{b}} = \begin{bmatrix} \hat{b}_0 \\ \hat{b}_1 \\ \\ \hat{b}_{n_k-1} \end{bmatrix} \qquad \hat{\mathbf{a}} = \begin{bmatrix} \hat{a}_0 \\ \hat{a}_1 \\ \\ \hat{a}_{n_k-1} \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} m_{k0} & & \\ m_{k1} & m_{k0} & \\ & & \\ m_{k(n_k-1)} & m_{k(n_k-2)} & m_{k0} \end{bmatrix}$$

We form the $\mathbf{C_p}$ and $\mathbf{C_q}$ matrices and the $\mathbf{M_k}$ matrix.

$$\mathbf{M_k} = \begin{bmatrix} \mathbf{C_{p1}} & \vdots & \mathbf{C_{p2}} & \vdots & \mathbf{C_{p3}} & -\mathbf{BC_{q2}} & -\mathbf{BC_{q3}} \end{bmatrix}$$

$\mathbf{C_n}$ matrix can be defined as :

$$
\mathbf{C_n} \equiv \left[\begin{array}{c|c|c} \mathbf{C_{n1}} & \mathbf{C_{n2}} & \mathbf{C_{n3}} \end{array}\right]
$$

$$
= \left[\begin{array}{c|cccc|c}
1 & s_k & s_k^2 & s_k^3 & s_k^{n-1} & s_k^n \\[4pt]
  & 1 & \binom{2}{1}s_k & \binom{3}{1}s_k^2 & \binom{n-1}{1}s_k^{n-2} & \binom{n}{1}s_k^{n-1} \\[6pt]
  &   & 1 & \binom{3}{2}s_k & & \\[6pt]
  &   &   & \ddots & & \\[6pt]
  &   &   & & \binom{n-1}{n_k-1}s_k^{n-n_k} & \binom{n}{n_k-1}s_k^{n-n_k+1}
\end{array}\right]
$$

$$(3.6)$$

The solution of the equation

$$
\mathbf{M_k}\begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_p \\ a_1 \\ a_2 \\ \vdots \\ a_q \end{bmatrix} = \begin{bmatrix} \mathbf{m_{k0}} \\ \mathbf{m_{k1}} \\ \vdots \\ \mathbf{m_{k(n_k-1)}} \end{bmatrix}
$$

$$(3.7)$$

will give the unknown coefficients.

If there are more than one expansion point, the equations will be solved simultaneously. That is, an $N \times N$ matrix

10

$$
\begin{bmatrix}
\mathbf{M_0} \\
\mathbf{M_1} \\
\mathbf{M_{-1}} \\
\vdots \\
\mathbf{M_n} \\
\mathbf{M_{-n}}
\end{bmatrix}
\begin{bmatrix}
b_0 \\
b_p \\
a_1 \\
a_q
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{m_0} \\
\mathbf{m_1} \\
\mathbf{m_{-1}} \\
\mathbf{m_n} \\
\mathbf{m_{-n}}
\end{bmatrix}
\tag{3.8}
$$

where $\mathbf{m_0}$ is the moment vector at $s = 0$ and $\mathbf{M_0}$ is the corresponding $\mathbf{M_k}$ matrix. Notice that, $\mathbf{m_k^*} = \mathbf{m_{-k}}$ and $\mathbf{M_k^*} = \mathbf{M_{-k}}$ are the conjugate moment vectors and matrices, respectively.

In AWE methods, usually $[q - 1/q]$ Padé Approximation is used [1, 8, 2].

This corresponds to taking simply $p = q - 1$ in the formulas above, we used this order in all of our simulations performed by method 1.

This method can be extended for the solution of the system in the least squares sense.

3.2.2   Method II

A faster but less accurate solution method based on the rational Hermite interpolation [12] will be described in this section. We are looking for polynomials

$$
p(x) = \sum_{i=0}^{m} b_i x^i
$$
$$
q(x) = \sum_{i=0}^{n} a_i x^i
$$

where p/q irreducible and satisfies

$$
\begin{cases}
f^{(l)}(x_i) = \left(\dfrac{p}{q}\right)^{(l)}(x_i) & \text{for } l = 0, \ldots, n_i - 1 \text{ with } i = 0, \ldots, j \\
f^{(l)}(x_{j+1}) = \left(\dfrac{p}{q}\right)^{(l)}(x_{j+1}) & \text{for } l = 0, \ldots, n_{j+1} - 1
\end{cases}
\tag{3.9}
$$

11

where $f^{(l)}$ is the lth derivative of f. Here, x_i's are the interpolation points, and there are n_i interpolation conditions at x_i.

The problem is reformulated as stated in the following lines:

$$y_l \quad = \quad x_0 \qquad \text{for} \quad l = 0, \ldots, n_0 - 1$$

$$y_{d(i)+l} \quad = \quad x_i \qquad \text{for} \quad l = 0, \ldots, n_i - 1$$

$$\text{with} \quad d(i) = n_0 + n_1 + \ldots + n_{i-1} \qquad (i \geq 1)$$

$$c_{ij} \quad = \quad 0 \qquad \text{for} \quad i > j$$

$$c_{ij} \quad = \quad f[y_i, \ldots, y_j] \qquad \text{for} \quad i \leq j$$

where

$$f[y_i, \ldots, y_j] = \begin{cases} \dfrac{f[y_{i+1}, \ldots, y_j] - f[y_i, \ldots, y_{j-1}]}{y_j - y_i} & \text{for} \quad y_i \neq y_j \\ \dfrac{f^{(j-i)}(y_i)}{(j-i)!} & \text{for} \quad y_i = y_{i+1} = \ldots = y_j \end{cases}$$

are the divided differences.
We will also define

$$B_j(x) = \prod_{l=1}^{j} (x - y_{l-1})$$

with

$$B_0(x) = 1$$

Then we have the Newton Series [13]

$$f(x) \quad = \quad \sum_{i=0}^{\infty} c_{0i} B_i(x)$$

$$p(x) \quad = \quad \sum_{i=0}^{m} b_i B_i(x)$$

$$q(x) \quad = \quad \sum_{i=0}^{n} a_i B_i(x)$$

such that

$$(fq - p)(x) = \sum_{i \geq m+n+1} d_i B_i(x) \tag{3.10}$$

Here d_i (divided differences) are 0 for $i = 0, 1, \ldots, m + n$. This problem is called **Newton-Padé Approximation problem** of order (m,n) for f.

Proceeding as in [12] yields the following system of equations.

12

$$c_{00}a_0 = b_0$$

$$c_{01}a_0 + c_{11}a_1 = b_1$$

$$(3.11)$$

$$c_{0m}a_0 + c_{1m}a_1 + \ldots + c_{nm}a_n = b_m$$

and

$$c_{0,m+1}a_0 + \ldots + c_{n,m+1}a_n = 0$$

$$(3.12)$$

$$c_{0,m+n}a_0 + \ldots + c_{n,m+n}a_n = 0$$

Solving the system of equations (3.12), gives the a_i ($i = 1, \ldots, n$) with a choice of $a_0 = 1$. Then substituting the a_i into system of equations (3.11) b_i ($i = 0, \ldots, m$) can be found. Note that these a_i's and b_i's are different from the ones in (3.4), since they are the coefficients of Newton series.

3.3 Complex Frequency Hopping

Another method for increasing AWE's accuracy with multi point expansions is the Complex Frequency Hopping (CFH)[6]. Different from our method, the CFH technique performs single expansions on several points and combines them into an accurate set of poles and residues. This algorithm first performs single point expansions at $s = 0$ and $s = j\omega_{max}$, the lowest and highest frequencies of interest. The poles are computed separately and a common pole is searched. If any pole is matched in both expansions, this pole is marked as accurate. Otherwise, another expansion point found using a binary search algorithm[14]. The search continues until every two successive expansions have at least one common pole. The algorithm given below summarizes the technique and it is illustrated in Fig. 3.1.

- **Step 1:** Poles from each expansion point obtained as mentioned in [14].

- **Step 2:** Residues are computed as in the AWE technique [1].

13
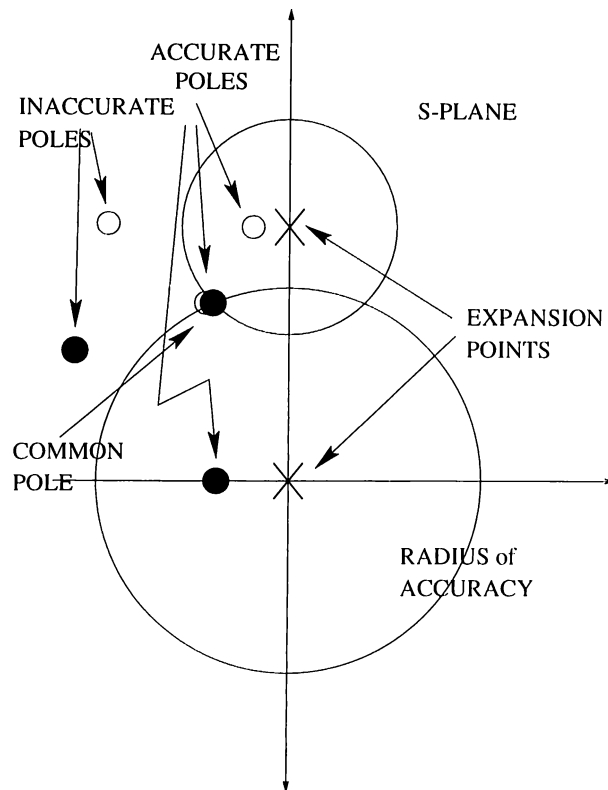
Figure 3.1: Pole selection algorithm in CFH

- **Step 3:** If the same poles are detected in two different expansions, they are marked as accurate.

- **Step 4:** The distance between an expansion point and its farthest accurate pole defines the radius of accuracy(R_{acc}). All poles within this R_{acc} are marked as accurate.

- **Step 5:** Poles that are not marked and corresponding residues are said to be inaccurate and rejected.

Chapter 4

MOMENT GENERATION

In previous chapter, we introduced two methods for multi point moment matching. Now, we will introduce the evaluation of the moments for linear circuits. This chapter begins with matrix formulation of the circuit and it proceeds with computation of moments from that matrix.

4.1 Linear Circuit Formulation

Consider a linear network π, which contains linear lumped components, and arbitrary linear subnetworks. The subnetworks may contain distributed elements. The Modified Nodal Analysis (MNA) matrix equations of the network π can be written as:

$$\mathbf{W} \frac{\mathrm{d}}{\mathrm{d}t} \mathbf{z(t)} + \mathbf{Hz(t)} + \sum_{k=1}^{N_s} \mathbf{D_k I_k} = \mathbf{b}u(t) \qquad (4.1)$$

where

$\mathbf{z(t)}$: node voltage vector appended by

independent voltage source current

15

and linear inductor current

W   matrix for energy storage

       lumped components

H   matrix for non energy storing

       lumped components

b   vector for independent sources

$\mathbf{D_k}$ selector matrix that maps i_k,

       the currents entering subnetworks

       to node space

$u(t)$: input function

and for the subnetworks, we have

$$\mathbf{A_k V_k} + \mathbf{B_k I_k} = 0 \qquad \text{for } k = 1 \ldots N_s \tag{4.2}$$

$\mathbf{V_k}$ and $\mathbf{I_k}$ are terminal voltages and currents of the kth subnetwork. Writing the Laplace transform of the equations we obtain:

$$
\begin{bmatrix}
s\mathbf{W} + \mathbf{H} & \mathbf{D_1} & \mathbf{D_2} & & \mathbf{D_{N_s}} \\
\mathbf{A_1 D_1^T} & \mathbf{B_1} & \mathbf{0} & & \mathbf{0} \\
\mathbf{A_2 D_2^T} & \mathbf{0} & \mathbf{B_2} & & \mathbf{0} \\
\vdots & & & & \vdots \\
\mathbf{A_{N_s} D_{N_s}^T} & \mathbf{0} & \mathbf{0} & & \mathbf{B_{N_s}}
\end{bmatrix}
\begin{bmatrix}
\mathbf{Z(s)} \\
\mathbf{I_1(s)} \\
\mathbf{I_2(s)} \\
\vdots \\
\mathbf{I_{N_s}(s)}
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{b} \\
\mathbf{0} \\
\mathbf{0} \\
\vdots \\
\mathbf{0}
\end{bmatrix}
U(s) \tag{4.3}
$$

We call the MNA matrix $\mathbf{T(s)}$, vector of unknowns $\mathbf{x(s)}$, and the excitation vector $\mathbf{w}$ and form the circuit equation as (2.5).

16

4.2 Evaluation of Moments

As mentioned in chapter 3, the moments of circuits with both lumped and distributed components are computed according to Eq. 3.1. That is, we need to take the derivatives of the circuit matrix T in order to evaluate the moments. The first derivative of T at $s = s_0$ is given as

$$
\mathbf{T}^{(1)} = \begin{bmatrix}
\mathbf{W} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{A}_1^{(1)}(\mathbf{s_0})\mathbf{D}_1^T & \mathbf{B}_1^{(1)}(\mathbf{s_0}) & \mathbf{0} & \mathbf{0} \\
\mathbf{A}_2^{(1)}(\mathbf{s_0})\mathbf{D}_2^T & \mathbf{0} & \mathbf{B}_2^{(1)}(\mathbf{s_0}) & \mathbf{0} \\
\vdots & & & \\
\mathbf{A}_{N_s}^{(1)}(\mathbf{s_0})\mathbf{D}_{N_s}^T & \mathbf{0} & \mathbf{0} & \mathbf{B}_{N_s}^{(1)}(\mathbf{s_0})
\end{bmatrix} \quad (4.4)
$$

and the higher order derivatives are

$$
\mathbf{T}^{(r)} = \begin{bmatrix}
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{A}_1^{(r)}(\mathbf{s_0})\mathbf{D}_1^T & \mathbf{B}_1^{(r)}(\mathbf{s_0}) & \mathbf{0} & \mathbf{0} \\
\mathbf{A}_2^{(r)}(\mathbf{s_0})\mathbf{D}_2^T & \mathbf{0} & \mathbf{B}_2^{(r)}(\mathbf{s_0}) & \mathbf{0} \\
& & & \\
\mathbf{A}_{N_s}^{(r)}(\mathbf{s_0})\mathbf{D}_{N_s}^T & \mathbf{0} & \mathbf{0} & \mathbf{B}_{N_s}^{(r)}(\mathbf{s_0})
\end{bmatrix} \quad r \geq 2
$$

If the circuit has lumped components only, $\mathbf{T}^{(r)} = 0$ for $r \geq 2$. The $\mathbf{A_k}$ and $\mathbf{B_k}$ are the entries associated with transmission line moments. The moments of the transmission lines are found by using the eigenvalue moment methods or matrix exponential method. The details of the subject can be found in literature [2, 11, 3]. In our simulations we considered lossless transmission lines only. We will illustrate the evaluation of moments for lossless transmission lines in an example.

4.2.1 Example

Assume a lossless transmission line with the parameters D, Z_0, v_p where D is the length, Z_0 is the characteristic impedance and v_p is the phase velocity (Fig. 4.1).

These parameters can be calculated from the unit electrical characteristics (L, C) if the frequency and the type of the line is determined. The terminal
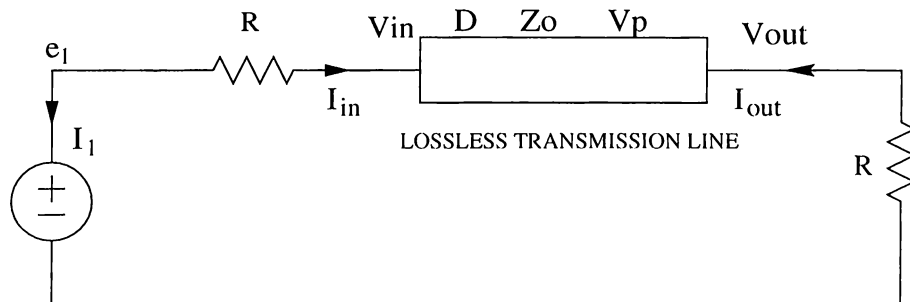

Figure 4.1: Illustrative example for moment generation

voltages of the line are related by

$$\mathbf{A} \begin{bmatrix} V_{in}(s) \\ V_{out}(s) \end{bmatrix} + \mathbf{B} \begin{bmatrix} I_{in}(s) \\ I_{out}(s) \end{bmatrix} = 0$$

where

$$\mathbf{A} = \begin{bmatrix} E_1(s_0) & -1 \\ E_2(s_0)/Z_0 & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} Z_0 E_2(s_0) & 0 \\ E_1(s_0) & 1 \end{bmatrix}$$

and

$$E_1(s) = cosh(sD/v_p)$$

$$E_2(s) = -sinh(sD/v_p)$$

The derivatives at $s = s_0$ are obtained as

$$\mathbf{A}^{(r)} = \begin{bmatrix} E_1^{(r)}(s_0) & 0 \\ E_2^{(r)}(s_0)/Z_0 & 0 \end{bmatrix}$$

$$\mathbf{B}^{(r)} = \begin{bmatrix} Z_0 E_2^{(r)}(s_0) & 0 \\ E_1^{(r)}(s_0) & 0 \end{bmatrix}$$

18

where

$$E_1^{(r)}(s_0) = \frac{1}{2}\left[\left(\frac{-D}{v_p}\right)^r e^{\frac{-s_0 D}{v_p}} + \left(\frac{D}{v_p}\right)^r e^{\frac{s_0 D}{v_p}}\right]$$

$$E_2^{(r)}(s_0) = \frac{1}{2}\left[\left(\frac{-D}{v_p}\right)^r e^{\frac{-s_0 D}{v_p}} - \left(\frac{D}{v_p}\right)^r e^{\frac{s_0 D}{v_p}}\right]$$

If we form the T matrix for this topology

$$\left[\begin{array}{cccc|cc} 1/R_1 & -1/R_1 & 0 & 1 & 0 & 0 \\ -1/R_1 & 1/R_1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1/R_2 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & E_1 & -1 & 0 & Z_0 E_2 & 0 \\ 0 & E_2/Z_0 & 0 & 0 & E_1 & 1 \end{array}\right] \left[\begin{array}{c} e_1 \\ V_{in} \\ V_{out} \\ I_1 \\ I_{in} \\ I_{out} \end{array}\right] = \left[\begin{array}{c} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{array}\right]$$

and the derivatives for $r \geq 1$ are

$$\mathbf{T^{(r)}} = \left[\begin{array}{cccc|cc} & & \mathbf{0} & & & \mathbf{0} \\ & & & & & \\ \hline 0 & E_1^{(r)} & 0 & 0 & Z_0 E_2^{(r)} & 0 \\ 0 & E_2^{(r)}/Z_0 & 0 & 0 & E_1^{(r)} & 0 \end{array}\right]$$

Since there are no energy storage passive elements in the circuit, the W matrix in Eq. 4.4 is set to zero. If we choose normalized values for the components $(R = 1, Z_0 = 0.5$ and $D/v_p = 1)$ we estimate the exact response as,

$$V_{out}(s) = \frac{0.5}{cosh(s) + 1.25 sinh(s)}$$

The response of AWE at 11th and 25th orders are depicted in Fig. 4.2 with the exact response. Since the exact response of such an circuit topology is periodic, it is impossible to approximate it by a rational. However, in most of the practical circuits, there are both energy storage elements and transmission lines and these circuits have dominant poles that enables us to approximate

19

their impulse responses with rationals. AWE methods are useful to analyze linear circuits when the dominant poles that are written in rational powers of s, exemplifies the entire response.
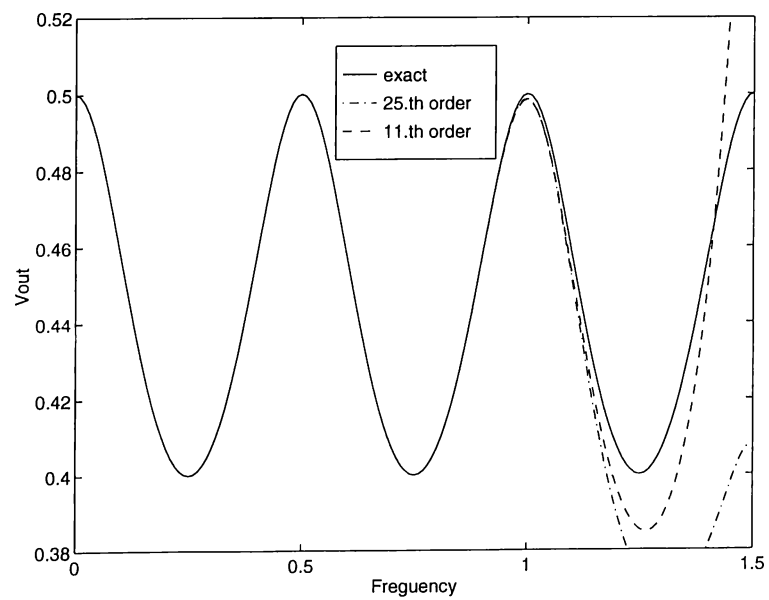
Figure 4.2: Output of the one transmission line circuit

Chapter 5

COMPUTATIONAL

CONCEPTS

This section discusses the design of the simulator in view of practical and numerical concepts. First, we explain the properties of the simulator and proceed with the numerical results.

5.1 Algorithm of the simulator

The simulator we designed performs the following tasks in order:

- Parsing the input file

- Forming the circuit matrix

- Calculation of frequency shifted moments according to the recursive scheme given in (3.1) and [2].

- Repetition of last step for each expansion point

- Matching moments to an aproximating rational and extracting the coefficients

The input parser inputs files very similar to Spice input format [15]. The format of the input circuit file and the configuration file (.config) that determines the expansion points are explained in Appendix. The circuit matrix is determined by the information arranged by the input parser. Several subroutines are used to perform the analysis as expressed in options card. A commercial matrix solver and the LU-solver implemented before are employed as external subroutines. The program produces a display output and an output file. The contents of the output is formated according to the options card as mentioned in Appendix.

5.1.1   Selection of the expansion points

Since we are employing a form of *Padé Approximation* and searching for approximating rational, only the dominant poles are crucial in our design. In time and frequency analyses, the poles closer to $j\omega$ axis are important, therefore we choose expansion points on $j\omega$ axis. Once, we set our frequency range of interest, we can apply complex frequency hopping technique (CFH) as mentioned in chapter 3. The frequency range of interest is generally between DC and a maximum frequency ($\approx GHz$ in interconnect circuits). The CFH technique gives the expansion points and the corresponding number of moments. We use this information in our method and obtain the *Multi point Padé Approximation* function.

5.1.2   Extracting the coefficients

In the first method the coefficients of the approximating function are obtained from a system of matrix equations(3.8). Each M_k and m_k element has its complex conjugate row in matrix equations. Therefore this $N \times N$ complex matrix system is equivalent to an $N \times N$ real system of equations and can be solved using the ordinary elimination algorithms such as LU [13].

22

In the second method, we form the divided difference table according to the n_i's and s_i's. The coefficients of the denominator are solved from the complex equations (3.12). This complex system is at $q \times q$ order, and could be solved using a complex matrix solver. The coefficients of the nominator are obtained by only forward and backward substitutions (Eq. 3.11).

5.2 Numerical issues

In our study, we are primarily interested in AC analysis and pole-zero extraction of interconnect circuits. Different from the conventional simulators, the program we propose solves the circuit matrix, only a few times. Since the LU factorization of the circuit matrix is known from the solution of the first moment vector, higher order moments are obtained by one forward and one backward substitution only. If the number of the expansion points is $n + 1$, we have a total of $n + 1$ LU factorizations of the circuit matrix. Obtaining the moments of the circuit at $DC + n$ points include $n + 1$ LU and $\sum_{i=0}^{n} n_i$ FBS's where n_i is the number of moments at ith expansion point.

Our method is proposed to solve complex circuits and the orders of approximations are generally large ($\sim$ 30–50) compared to the typical approximations employed by AWE technique($\sim$ 4–12)[1, 2, 4]. Since the orders of matrix sizes and the orders of approximations are high in interconnect AC analysis, we need larger memory area, higher accuracy and consume more cpu time, compared to a typical AWE transient analysis problem. Because of the very large and very small numbers appearing in matrices, the method becomes ill-conditioned. The calculation of many moments ($\sim$ 10) at one expansion point results in very small numbers as successive moments, since each consecutive moment is smaller than the previous one by an order of $\sim 10^9$ in typical networks. Also, the powers of expansion points appearing in Eq. (3.6), yields very big numbers.

As the circuit expands, the ill–conditioned behavior of the matrix in Eq. 3.8 increases. The reason for that is the deviation between the first and the last moments obtained from an expansion point. We can overcome this problem by setting a limit value for the ratio of last and first moments of expansion point

s_k.

$$\varepsilon_{ki} = \frac{m_{ki}}{m_{k0}} \qquad i = 0, \ldots, n_{k-1}$$

For ε_{kj} smaller than a reasonable limit, the moments m_{ki} for $i \geq j$ are not calculated. If j is less than what CFH requires, further action is necessary. To preserve the same accuracy. the number of expansion points must be increased beyond what results from CFH technique.

Also, frequency scaling should be applied to the energy storing elements in the circuit to increase accuracy as well as employing high precision arithmetic as explained in the next subsection.

5.2.1   Using high precision arithmetic

The most obvious method to overcome accuracy problems is to use a higher precision arithmetic. Although we use double precision arithmetic in all operations, we have accuracy problems in larger circuits, such as interconnects cascaded three times or more. A higher precision of arithmetic may be used instead of double precision, but then we have to consider the dramatically increased CPU time. This work was done for AWE transient analysis in [14] and accurate results were obtained.

5.2.2   Stability

Another observation about the method is the stability of approximated poles. Similar to the AWE technique, *Multi point Padé Approximation* technique may result in spurious right hand side poles as well. This is because of the nature of *Padé Approximations*. The typical way to overcome this problem is to discard unstable poles and solve for the remaining system of equations. However, in most of the unstable cases, the effect of the unstable pole is negligible in total approximation.

24

Chapter 6

EXAMPLES

Several examples are presented here to demonstrate the performance of the method. Since our primary concern is AC analysis, and this requires higher orders of approximations than transient analysis, the circuits demonstrated here are at considerable sizes. In run-time estimations, a SUN-SPARC20 machine on UNIX is used and the averages of several run-times are considered. The accuracy of the clock used is 16 msec.

6.1 Example 1

The first example is a well known interconnect circuit given in several references [8] [2]. As seen from Fig 6.1, the circuit has 29 lumped components, 7 lossless transmission lines and 21 nodes. Our frequency of interest is $0 - 6GHz$. By applying CFH technique to this circuit, we found the order of approximation as 35. The expansion points and moment numbers are in Table 6.1. The AC response of the circuit computed according to the moment table. Scaling was taken as 1×10^9 for frequency dependent components.

The AC response $H(s)$ of the circuit and the time comparisons are shown in Fig. 6.2 and Table 6.2, respectively. As seen from the figure, the *multi point*
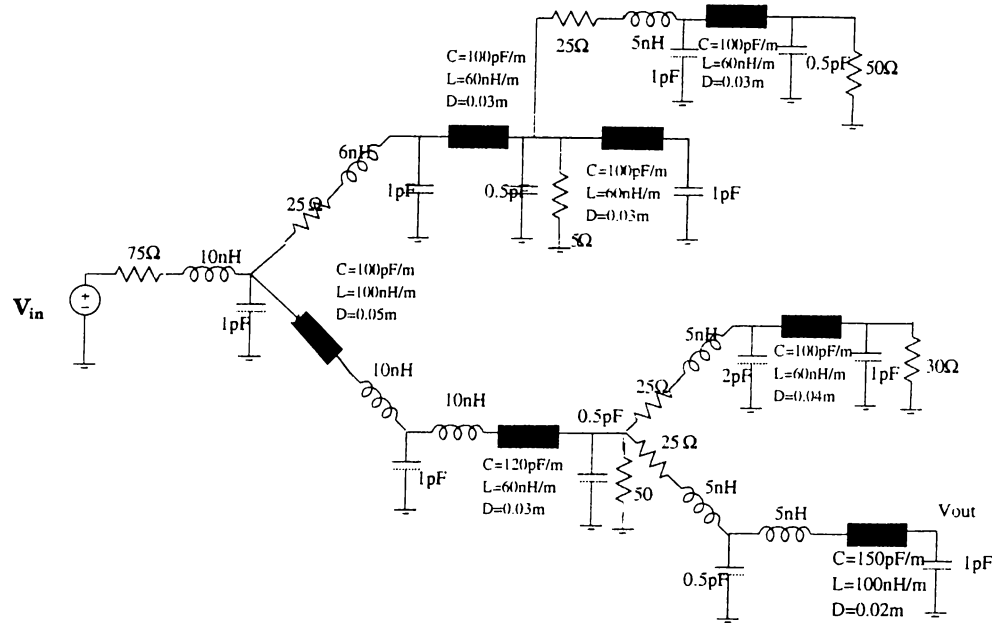
25

Figure 6.1: Example 1: Interconnect model with 7 transmission lines

Expansion point	Moments
$s = 0$	10
$s = 2\pi 1.25 \times 10^9 j$	10
$s = 2\pi 2.50 \times 10^9 j$	10
$s = 2\pi 5.00 \times 10^9 j$	10

Table 6.1: Expansion points and the moment numbers for interconnect network in Ex. 1

Padé approximation via Method1 (MAWE) and HSpice responses match each other exactly, while Method2 has significant error at the end of frequency range. We can see from Table 6.2 that MAWE and Method2 spent much less time to get the same results as HSpice and Spice3. The high speed of Method2 resulted in loss of accuracy that makes it inefficient to use in AC analysis, while MAWE (Method1) has a moderate time reduction without any loss of accuracy. When we calculate AC response at 6000 points in the frequency range $0 - 6GHz$, the proposed MAWE computed nearly 11 times faster than HSpice and nearly 7 times faster than Spice3. In a 600 point AC analysis, HSpice and Spice3 run times are closer to that of MAWE, but still MAWE has 3 or 4 times better
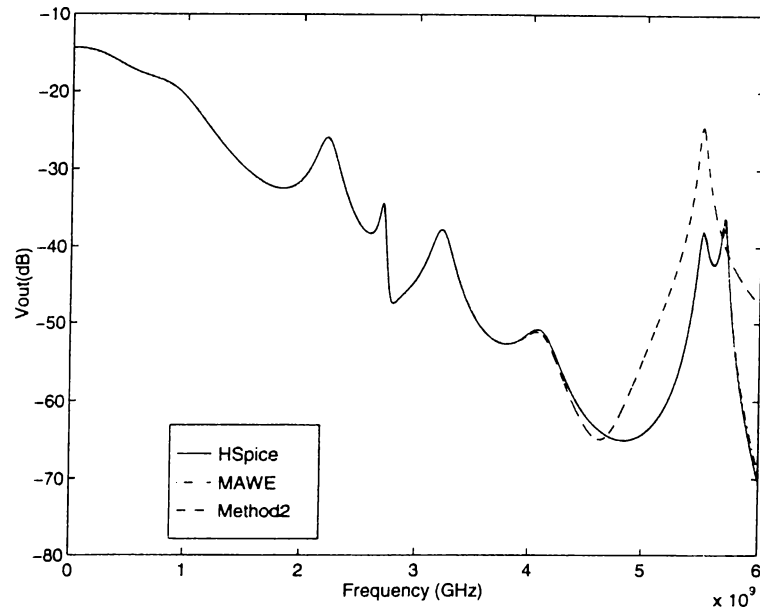
Figure 6.2: Output waveform for the interconnect network in Ex. 1

scores.

Simulator	Real Analysis Time(sec)	
	6000 points	600 points
MAWE	1.1	0.9
METHOD2	0.6	0.5
HSPICE	11.9	3.7
SPICE3	7.8	2.4

Table 6.2: Timing results for the interconnect network in Ex. 1 (I/O times excluded)

If we investigate the run time distribution diagram of the MAWE interconnect analysis (Fig. 6.3), we notice a remarkable time consumption for solving the coefficients according to Eq. 3.8. The slow behavior of the matrix solver used causes a speed disadvantage. The matrix system in (3.8) is large and nearly ill-conditioned in this example because of the high approximation order. In order to solve this bad-mannered system, we employed a matrix solver in the available LAPACK library that resulted in loss of time. The LU process

27

shared insignificant time among all processes. This is due to the special LU solver implemented before by the Bilkent University CAD group for solving the circuit matrix.
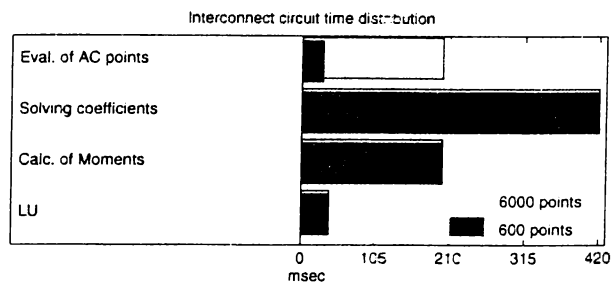

Figure 6.3: Run time distribution of MAWE for Example 1

6.2 Example 2

Second example consists of two cascaded blocks, where the previous circuit is taken as a block (Fig. 6.4). Obviously, the circuit has twice more elements and nodes, i.e., 42 nodes, 14 lossless transmission lines and 58 lumped components. We applied frequency scaling as in the first example.
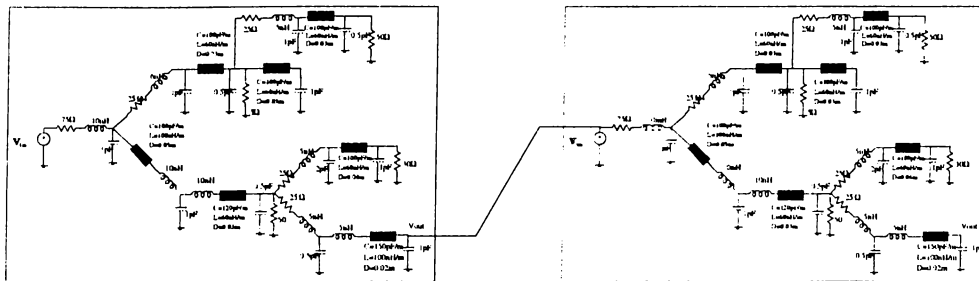

Figure 6.4: Example 2: Cascaded interconnects with 14 transmission lines

Again, our frequency range is the same $(0 - 6GHz)$. The expansion points and the number of moments for this circuit are given in Table 6.3. Since this

circuit is more complex and more stiff than the other one, we need to spread the expansion points to 8 points. The order of approximation increased as well, i.e.. we now compute a total of 78 moments.

Expansion point	Moments
$s = 0$	10
$s = 2\pi 1.00 \times 10^9 j$	5
$s = 2\pi 2.00 \times 10^9 j$	4
$s = 2\pi 2.50 \times 10^9 j$	3
$s = 2\pi 3.00 \times 10^9 j$	5
$s = 2\pi 4.00 \times 10^9 j$	5
$s = 2\pi 5.00 \times 10^9 j$	5
$s = 2\pi 5.50 \times 10^9 j$	2
$s = 2\pi 6.00 \times 10^9 j$	5

Table 6.3: Expansion points and the moment numbers for Example 2

The AC responses of cascaded interconnects computed by MAWE and Method2 are compared with HSpice in Fig. 6.5. There is a slight difference between MAWE and Hspice (exact) responses. while Method2 has noticeable error. The slight error of MAWE is acceptable since the absolute error never exceeds 3.5×10^{-4} (Fig. 6.6).

The time comparison is given in Table 6.2. Method2 scored the best time again, but, the accuracy of Method2 is not sufficient enough to verify the AC analysis. MAWE has still significant speed advantage over HSpice and Spice3. This time, the speed up over HSpice is 14 times and over Spice3 is 7 times in 6000 point analysis. We have similar results as example1 in 600 point analysis. MAWE scored faster than both Spice simulators in this analysis as well.

We were expecting MAWE to become faster as the circuit enlarges. However, as in this example, when the circuit size is doubled, the speed gain over Spice simulators do not have a noticeable change. After the investigation of the distribution in Fig. 6.7, we notice long time bars for solving coefficients according to Eq. 3.8 and moment update according to Eq. 3.1. In this example, matrix solver consumed the longest time as well. because of the high approximation order and many expansion points used. Again, this results in large and
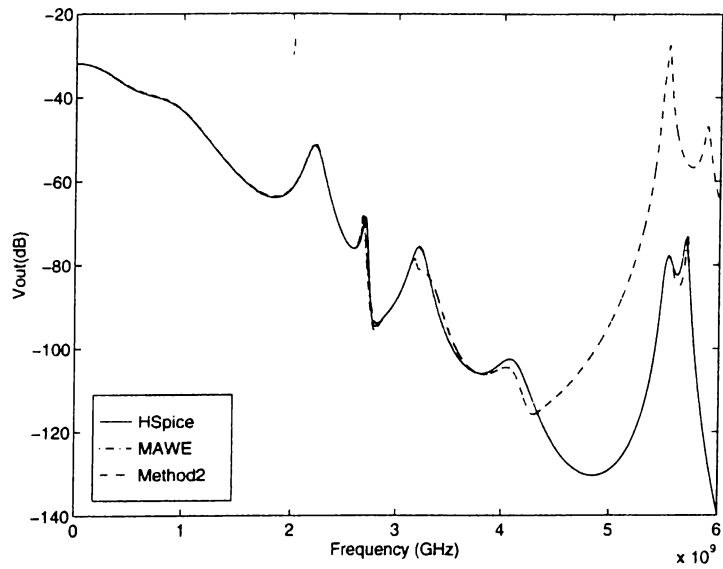
Figure 6.5: AC Response of MAWE in cascaded interconnect circuit in Ex. 2
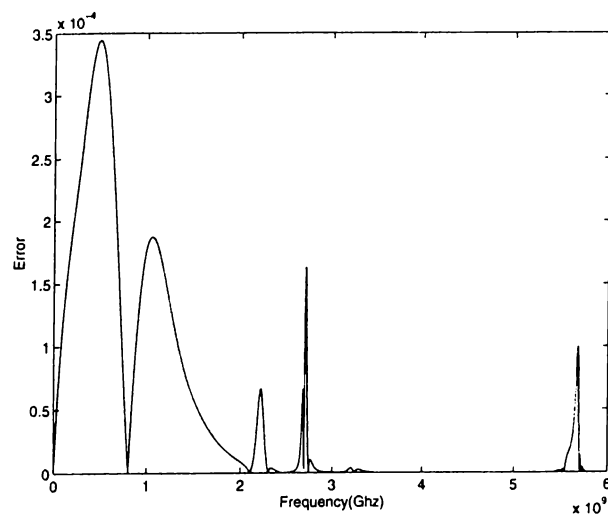

Figure 6.6: Absolute error of MAWE response for Example 2

30

Simulator	Real Analysis Time(sec)	
	6000 points	600 points
MAWE	1.6	1.5
METHOD2	1.1	1.0
HSPICE	21.9	4.9
SPICE3	11.0	4.2

Table 6.4: Timing results for cascaded interconnects in Ex. 2 (I/O times excluded)
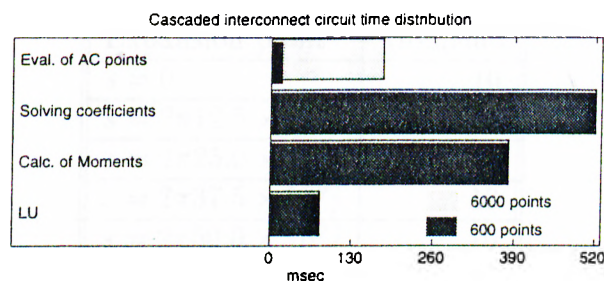

Figure 6.7: Run time distribution of MAWE for Example 2

ill-conditioned moment matrix systems that takes a long time to solve.

6.3 Example 3

Our example is a lowpass filter with five transmission lines (Fig. 6.8) which was also investigated in [8]. The filter has 2 capacitors, 2 resistors and an inductor beside 5 lossless transmission lines. The expansion points and moment numbers for the filter is given in Table 6.5. The order of the approximation is 47, hence a total of 94 moments are calculated.

In Fig. 6.9, the output response of the lowpass filter is given. Method2 could not solve this ill-conditioned system. The responses of MAWE and Hspice are indistinguishable. The frequency region of interest is $0 - 50GHz$. The time comparisons between MAWE and Spice simulators and the run-time distribution of MAWE is given in Table 6.6 and Fig. 6.10, respectively. MAWE reduced
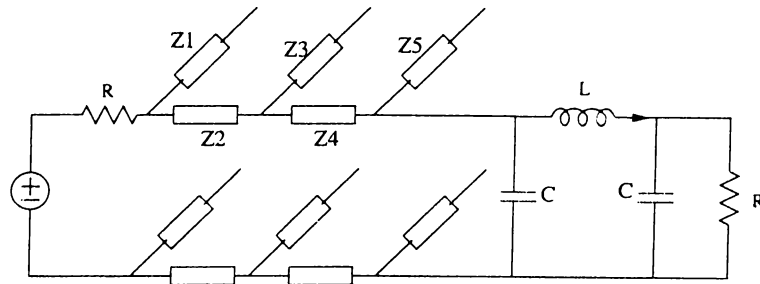
31

Figure 6.8: Example 3: Lowpass filter with 5 transmission lines

Expansion point	Moments
$s = 0$	10
$s = 2\pi 12.5 \times 10^9 j$	10
$s = 2\pi 25.0 \times 10^9 j$	10
$s = 2\pi 37.5 \times 10^9 j$	10
$s = 2\pi 50.0 \times 10^9 j$	10

Table 6.5: Expansion points and moment numbers for the filter example

analysis 4 times against Hspice and 3 times against Spice3 in 6000 point analysis. The speed recovery of MAWE against Spice simulators reduced in 600 point analysis.

The run-time distribution of MAWE in lowpass filter analysis is similar to the two preceding examples. Again, the matrix solver takes the longest time among all processes. Since there are 94 moments calculated, the matrix system in 3.8 has an order of 94×94. This takes 1000 msecs to solve the coefficients

Simulator	Real Analysis Time(sec)	
	6000 points	600 points
MAWE	1.9	1.4
HSPICE	7.5	3.4
SPICE3	5.2	1.4

Table 6.6: Timing results for the filter in Ex. 3 (I/O times excluded)
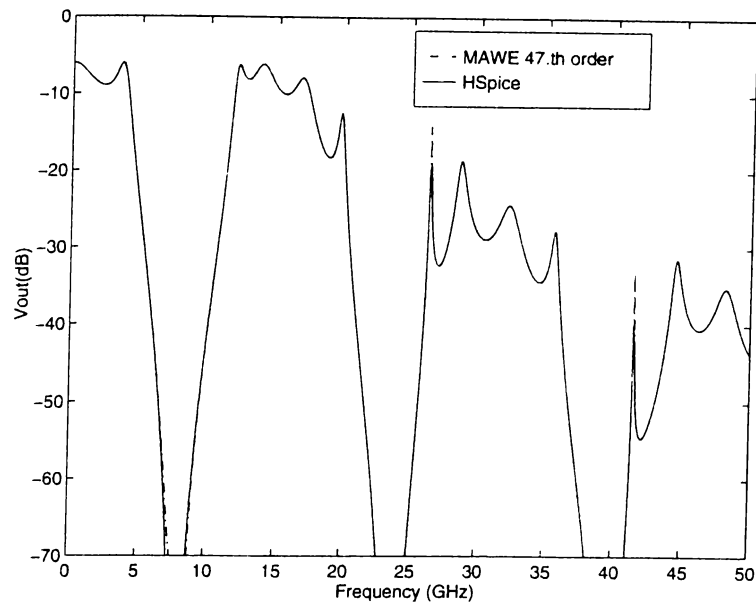
Figure 6.9: Lowpass filter output response
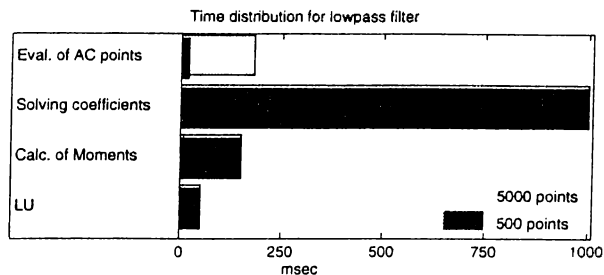
Figure 6.10: Distribution bars for the lowpass filter analysis by MAWE

in both 6000 and 600 point analyzes.

6.4 Example 4

Next, we will consider two rlc circuits in the same topology with 10 cells and 100 cells as given in Fig. 6.11. The circuits have 10 rlc elements with 21 nodes and 100 rlc elements with 201 nodes, respectively. The frequency range of interest in the first circuit is $0 - 5GHz$, while $0 - 2GHz$ in the second one. Scaling was applied to element values as 1×10^9. The moment numbers and expansion

Figure 6.11: Topology of the rlc circuits in Ex. 4

Expansion point	Moments
$s = 0$	12
$s = 2\pi 2.50 \times 10^9 j$	7
$s = 2\pi 5.00 \times 10^9 j$	7

(a)

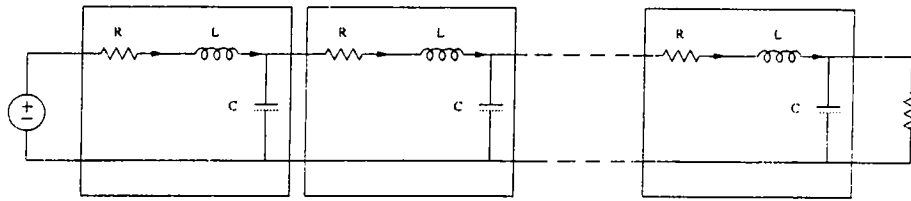| Expansion point	Moments
$s = 0$	16
$s = 2\pi 1.00 \times 10^9 j$	10
$s = 2\pi 2.00 \times 10^9 j$	10

(b)

Table 6.7: Expansion points and moment numbers for (a) 21 and (b) 201 node rlc circuits

points are given in Table 6.7. It can be seen that, 28th order approximation is needed to find the response of 201 node circuit, while only 20th order is sufficient for 21 node circuit.

The output responses of two circuits are given in Fig. 6.12. We can see that, the exact response of HSpice is matched by MAWE for both circuits, while Method2 has significant loss of accuracy in both circuits. The time comparisons for both rlc circuits are given in Table 6.8. The time results of Method2 are excluded, since it has inaccurate responses compared to other simulators.

In RLC20, MAWE's run time improvement against HSpice is in the order of 30, while in RLC200 the improvement is higher. This indicates that MAWE has better scores in a large analysis such contains 5000 points or more. This is valid for not only rlc circuits, but also interconnect circuits with distributed components.

34
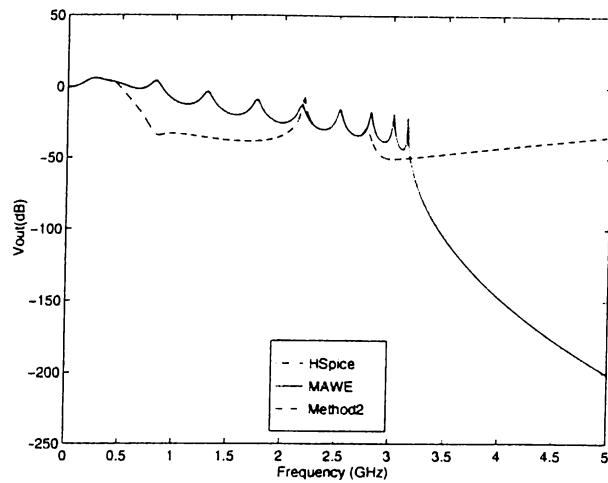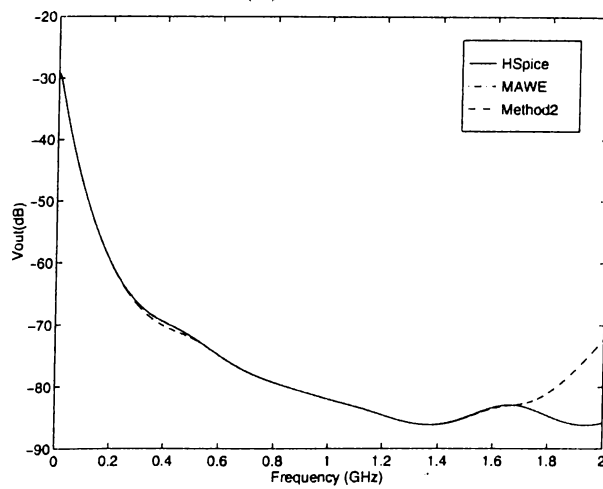
(a) RLC20

(b) RLC200

Figure 6.12: AC Responses of the rlc circuits with 21 and 201 nodes

Simulator	Real Analysis Time(sec)			
	RLC20		RLC200	
	5000 pts	500 pts	5000 pts	500pts
MAWE	0.4	0.25	2.6	2.5
HSPICE	12.5	5.4	89	12.9
SPICE3	20.4	1.1	19	2.4

Table 6.8: Timing results for the rlc networks in Ex. 4 excluding I/O times

35

Time distribution for RLC20 circuit

Eval. of AC points	
Solving coefficients	
Calc. of Moments	
LU	5000 points
	500 points

0   42.5   85   127.5   170
msec

Time distribution for RLC200 circuit

Eval. of AC points	
Solving coefficients	
Calc. of Moments	
LU	5000 points
	500 points

0   255   510   765   1020
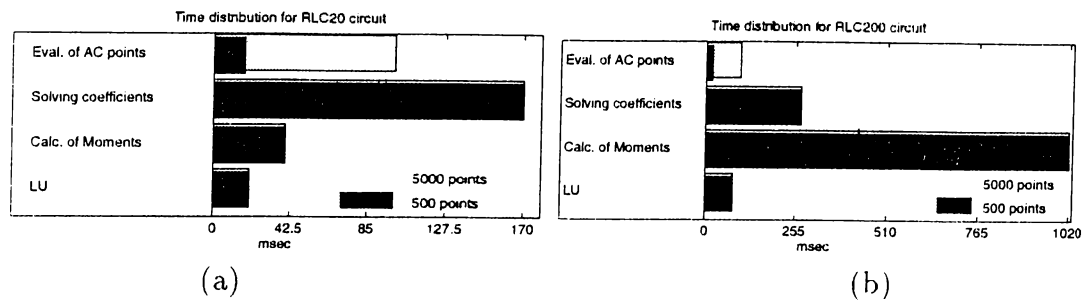msec

(a)   (b)

Figure 6.13: Run-time distributions of MAWE for (a)RLC20 (b)RLC200 in Ex. 4

When we look at time distributions of MAWE in Fig. 6.13, a major difference appears in two diagrams. The addition of the moment update process is dominant in 100 cell circuit (Fig 6.13a), while the addition of matrix solver dominates in 10 cell circuit (Fig 6.13b). In contrast to previous circuits and RLC20 circuit, the moment update takes longer time than the process of solving coefficients in RLC200 circuit. This is because of the large circuit matrix which is used in Eq. 3.1. This is due to the nearly 10 times larger circuit matrix compared to RLC20 circuit. This larger matrix which is used in recursive Eq. 3.1 causes many more multiplications than the multiplications needed for smaller size rlc network. Notice that, the data retrieving operations in moment update process consumes time as much as multiplication operations.

6.5   Discussion of the results

The examples introduced before demonstrated the accuracy and efficiency of two methods. It is observed that MAWE (Multi point via Method1) is faster than Spice like simulators in AC analysis of distributed networks especially in large analysis (more than 5000 points) without any significant loss of accuracy. The reduction in time against HSpice is approximately on the order of 10 in large analysis, and 5 in small analysis. Although, Method2 is faster than MAWE and Spice simulators, it has quite large deviations from the exact response and failed to analyze circuits successively at this size. From this point of view, we propose MAWE to compete with conventional simulators.
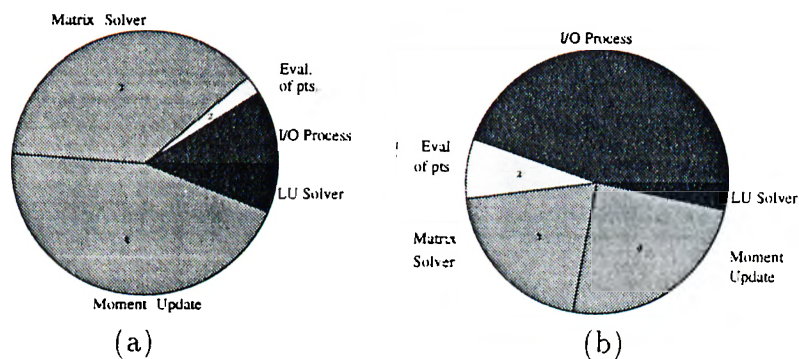
Figure 6.14: Time shares of all processes in a small and a large analysis

We also observed that the recovery of MAWE in analysis time obtained nearly without any accuracy loss. The responses of MAWE and Spice simulators are indistinguishable in the examples except a few points and the error never exceeds acceptable limits.

When we look at the time distribution diagrams of MAWE, we notice two different situations occured. If the circuit is larger compared to the approximation order, there appears a large circuit matrix, and the moment update process dominates in total time. The large circuit matrix involved in Eq. 3.1, is used in many multiplication and data retrieving operations. The sum of multiplications at one expansion point is $O(n_i^2 \times n_c^2)$. where n_c is the size of the circuit matrix T and n_i is the number of moments at that expansion point. Otherwise, higher approximation order compared to circuit size results in dominant time for matrix solver.

In the Fig. 6.14, the weights of all processes are given for a moderate MAWE analysis. The I/O process is the dominant time consumer in a large analysis. The moment update process and matrix solving process follows in order depending on the situations mentioned above. In a small analysis, these two processes have more importance than I/O process as expected. Evaluation of points costs nearly as much as LU process in the large analysis, but, obviously LU becomes expensive in the small analysis. As the circuits expand, the addition of LU on total time will increase; however, this will not cause dramatical changes in total execution time, since LU is performed once at each expansion point.

37

Chapter 7

CONCLUSION

We have introduced a program with two methods for verifying the AC response of linear circuits, using multi point moment matching techniques. Since Method2 gave inefficient results, MAWE (multi point Padé Approximation via Method1) is proposed to compete with Spice-like simulators.

Instead of calculating the frequency response at a large number of discretized points — like conventional simulators do — our program extracts the coefficients of an s-domain approximating rational of the impulse response. The proposed program can handle rlc circuits as well as the circuits with lossless transmission lines with no topological constraints, such as inductor loops, etc. The verification of the pole-zero analysis can also be done by the proposed program. The execution time of MAWE compared to Spice-like simulators is 8–10 times better. This improvement can be multipled by using a better matrix solver. The performance of the simulator increases as the matrix solver's execution time decreases. Hence, MAWE will have a better run-time advantage against Spice-like simulators by means of an LU solver faster than the one we used.

The effectiveness of the program for large sized circuits can be extended by increasing the number of expansion points and distributing the required number of moments to these points.

This program can also handle multi-conductor transmission line circuits by using an approach presented in [2]. Finally, our simulator can perform transient analysis of larger networks easily because of the nature of transient analysis which requires lower order of approximation (4-8). The recovery of the execution time in transient analysis will be similar to AC analysis as well.

APPENDIX A

I/O FORMAT OF THE PROGRAM

A.1 Configuration File (.config)

The example .config file that defines the expansion points and corresponding moment numbers is given. The order of approximation is 46 and for example the number of moments at $s = 0 + j3.0 \times 10^9$ is 8.

```
46
0 0 14
0 1.5e9 10
0 3.0e9  8
0 4.5e9 12
0 6.0e9 9
```

A.2 Circuit Input File (inlin.sp)

```
****** Example interconnect network
res1 1 2 75
lind1 2 3 10e-9
cap0 3 0 1e-12
res2 3 4 25
lind2 4 5 6e-9
cap1 5 0 1e-12
trl1 5 0 6 0  C=100e-12 L=60e-9 D=.03
cap2 6 0 .5e-12
res3 6 0 50
res4 6 7 25
lind3 7 8 5e-9
cap3 8 0 1e-12
vin 1 0 1
.ac 1e7 6e9 1e7
.options useawe scaling=1e9 lapack
.print v(8)
.end
```

Transmission line : node1 node2 node3 node4 C=value(F) L=value(H) D=length(m)

Resistor   : rxxx node1 node2 value(Ω)

Capacitor   : cxxx node1 node2 value (F)

Inductor   : lxxx node1 node2 value (H)

Voltage Input   : vxxx node1 node2 value (V)

A.2.1 Options Card

.options :

printmb   : Prints T and T^r at each moment computation

showiterno : Prints iteration number

useawe   : Employs AWE and multi point

                subroutines to verify the analysis

poles   : Prints poles and residues of the circuit

lapack   : Employs commercial Lapack package matrix solver

                instead of another solver implemented before

method2   : Employs Method2 to verify the analysis

moments   : Prints moments at each expansion point

post   : Prints every step to output file

scaling   : Inputs scaling value i,e, scaling= $1e9$

list   : Prints every step to display

A.2.2   Other Cards

.print   : Defines output parameter to be printed, i.e,

          print v(node number)

.tran   : Determines the transient analysis, i.e,

          tran time-step stoptime

.ac   : Determines the ac (frequency) analysis, i.e.,

          ac startfreq stopfreq stepfreq

.dc   : Determines the dc analysis, i.e.,

          dc startvolt stopvolt stepvolt

A.3 Output of the program

The output of the program for the input and .config files above is given next.
The output consists of an output file and display output. The output file stores
the number of discrete frequencies and the corresponding voltages in absolute
values in a neat format which could be easily loaded by graphics tools (Matlab,
Mathematica....). The display output is as follows:

```
data file is interconnect/Awefiles/inl.out
scaling
=
-------This is the beginning-------


Parsing Time=0 sec.
MNA matrix forming time for (0,0)=0 sec.
LU time=0 sec.
First FBS time=0 sec.
Other moments (13) evaluation time=0.016666 sec.
MNA matrix forming time for (0,1.5)=0 sec.
LU time=0 sec.
First FBS time=0 sec.
Other moments (9) evaluation time=0 sec.
MNA matrix forming time for (0,3)=0.016666 sec.
LU time=0 sec.
First FBS time=0 sec.
Other moments (7) evaluation time=0 sec.
MNA matrix forming time for (0,4.5)=0 sec.
LU time=0 sec.
First FBS time=0 sec.
Other moments (11) evaluation time=0.016666 sec.
MNA matrix forming time for (0,6)=0 sec.
LU time=0 sec.
First FBS time=0 sec.
Other moments (8) evaluation time=0 sec.
```

```
Lapack matrix solving time =0.799968 sec.
****rcond = 3.1096e-59
numerator of order 45
(0.333333,0) (-0.148894,-0.133465) (0.0813455,0.0139253)
 (-0.00395075,0.00700985) (-0.000353988,-0.000384566)
..... (-3.93383e-40,7.23369e-41) (1.71645e-41,6.20467e-42)
denominator of order 46
(1,0) (-0.0821817,-0.400394) (0.140346,-0.104168)
(0.056108,0.0125836) (0.00711239,0.00681182)
(0.00147754,0.00166321) .....(-1.50432e-37,1.54943e-37)
(-4.3418e-39,2.25196e-39) (-2.74136e-40,3.49853e-40)


 Evaluation time for given frequency range =0.116662 sec.


 Evaluation time without io =0.033332 sec.


Total time=1.23328 sec.
-------That's the end-------
```

REFERENCES

[1] L T. Pillage, and R. A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Transactions on Computer Aided Design*, 9:352–366, April 1990.

[2] Tak K. Tang, and Michel S. Nakhla. Analysis of high-speed vlsi interconnects using the asymptotic waveform evaluation technique. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 11(3):341–352, March 1992.

[3] J.E. Bracken, V. Raghavan, and R. A. Rohrer. Interconnect simulation with asymptotic waveform evaluation. *IEEE Transactions on Circuits and Systems-I: Fundemantal Theory and Applications*, 39:869–878, Nov 1992.

[4] X. Huang, V. Ragvahan, and R. A. Rohrer. Awesim: A program for the efficient analysis of linear(ized) circuits. In *Tech. Dig.*, pages 534–537. ICCAD, Nov. 90.

[5] D. F. Anastakis, N. Gopal, S.Y. Kim, and L.T. Pillage. On the stability of approximations in asymptotic waveform evaluation. In *Proc. Design Automation Conference*, May 1993.

[6] Michel S. Nakhla and Eli Chiprout. Analysis of interconnect networks using complex frequency hopping (cfh). *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*. 14(2):186–200, February 1995.

[7] Peter Feldmann and Roland W. Freund. Efficient linear circuit analysis by padé approximation via the lanczos process. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 14(5):639–649, May 1995.

[8] Mustafa Celik, Ogan Ocali, Mehmet A. Tan, and Abdullah Atalar. Pole-zero computation in microwave circuits using multi point padé approximation. *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, 42(1):6-13, January 1995.

[9] Chi-Tsong Chen. *Linear System Theory and Design*. Holt-Saunders International Editions, 1984.

[10] Jiri Vlach and Kishore Singhal. *Computer methods for circuit analysis and design*. Van Nostrand Reinhold Company, 1983.

[11] Eli Chiprout and Michel S. Nakhla. Addressing high-speed interconnect issues in asymptotic waveform evaluation. In *Design Automation Conference*, 1993.

[12] Annie Cuyt and Luc Wuytack. *Nonlinear Methods in Numerical Analysis*. North-Holland, 1988.

[13] S. D. Conte and C. De Boor. *Elementary numerical Analysis*. McGrawHill, Singapore, 1981.

[14] Eli Chiprout and Michel S. Nakhla. *Asymptotic Waveform Evaluation And Moment Matching for Interconnect Analysis*. Kluwer Academic Publishers, 199- .

[15] Meta-Software. *Hspice User's Manual*, 1992.