

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

**Scattering-Parameter-Based Macromodel
for Transient Analysis of Interconnect Networks with
Nonlinear Terminations**

A dissertation submitted in partial satisfaction
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

COMPUTER ENGINEERING

by

Haifang Liao

May 1995

The dissertation of Haifang Liao is
approved:

Wayne Wei-Ming Dai

Pak K. Chan

Patrick Mantey

Dean of Graduate Studies and Research

Copyright © by
Haifang Liao
1995

Contents

Contents	iii
List of Figures	v
List of Tables	viii
ABSTRACT.....	ix
Acknowledgment.....	x
Chapter 1. Introduction	1
Chapter 2. Scattering-Parameter-Based Macromodel	6
2.1. Scattering Parameters of Distributed-Lumped Components	6
2.2. Scattering-Parameter-Based Macromodel	9
Chapter 3. Capturing Time-of-flight Delay	15
3.1. Properties of Time-of-Flight	16
3.2. Scattering Parameters of Components with Time-of-Flight Captured	19
3.3. Keeping Track of Time-of-Flight.....	20
Chapter 4. Time Domain Synthesis of the Macromodel	23
4.1. Synthesis with Padé Approximation	24
4.2. Synthesis with Exponentially-Decayed Polynomial Functions.....	25
4.3. Synthesis with Mixed-Exponential Function.....	27
4.4. Accuracy and Stability Issues	28
4.5. Experiment Results	29
Chapter 5. Norton Equivalent Circuits Based on Recursive Convolution.....	37
5.1. Recursive Convolution Based on the EDPF	38
5.2. Norton Equivalent Circuits	39
5.3. Stability of Macromodel	40
5.4. Experiment Results	42
Chapter 6. Partitioning of Interconnect Networks.....	47
6.1. Circuit Partitioning	48
6.2. Efficiency and Accuracy of Reduced Networks	50

Chapter 7. Circuit Synthesis of RC Networks	52
7.1. Circuit Synthesis of RC Interconnect Networks	52
7.2. Stability of Synthesized Circuits	56
7.3. Experimental Results	58
Chapter 8. A CMOS Driver Model for Transient and Power Dissipation Analysis . . .	62
8.1. Driver Modeling.	63
8.2. Transient Analysis with Distributed Interconnect Load	68
8.3. Power Dissipation Analysis with Transient Leakage Current.	72
8.4. Experimental Results	74
Chapter 9. Conclusions	80
9.1. Concluding Remarks	80
9.2. Future Research	81
Appendix A. Scattering Matrix of an Ideal Interconnect Node.	83
Appendix B. Recursive Convolution Based on EDPF.	87
References	90

List of Figures

Figure 2.1. Four basic elements.	7
Figure 2.2. S-parameter-based macromodel.	10
Figure 2.3. Adjoined merging.	10
Figure 2.4. Self merging.	11
Figure 2.5. A two port network.	13
Figure 3.1. Time-of-flight delay	16
Figure 3.2. Transmission line circuit.	18
Figure 3.3. For $i, j \in X$, there are two paths for wave to propagate from port i to port j	21
Figure 3.4. For $i \in X, j \in Y$, there is only one path for wave to propagate from port i to port j	22
Figure 3.5. For $i, j \in X$, there are two paths for wave to propagate from port i to port j	22
Figure 4.1. An RC circuit with floating capacitance loops.	30
Figure 4.2. Output response of the RC circuit.	30
Figure 4.3. The responses of the RLC circuit.	31
Figure 4.4. A clock network.	32
Figure 4.5. Output response at node U1.	32
Figure 4.6. MCMC-93 benchmark.	33
Figure 4.7. The result comparison of the MCMC-93 benchmark by SPICE3e2 and the 5th order approximation without time-of-flight extraction (nonTOF).	34
Figure 4.8. The result comparison of the MCMC-93 benchmark by SPICE3e2 and the 5th order approximation with time-of-flight extraction (TOF).	34
Figure 4.9. A transmission line circuit.	35
Figure 4.10. The result comparison of the lossy line circuit by SPICE3e2 and	

the 4th order approximation without time-of-flight extraction (nonTOF).....	35
Figure 4.11. The comparison of the lossy line circuit by SPICE3e2 and the 4th order approximation with time-of-flight extraction (TOF).	36
Figure 5.1. Macromodel with virtual nodes.	39
Figure 5.2. Equivalent circuit of a two port macromodel..	41
Figure 5.3. A lossy transmission line circuit with nonlinear loads..	42
Figure 5.4. Near end response of the lossy transmission line circuit.	43
Figure 5.5. Far end response of the lossy transmission line circuit.	43
Figure 5.6. Grid-type clock network..	44
Figure 5.7. Near end response of the clock network..	45
Figure 5.8. Far end response of the clock network..	45
Figure 5.9. Comparison of different order approximation.	46
Figure 6.1. Several smaller components are more efficient than one large component to model complex interconnects.	48
Figure 6.2. Bucket list structure.	50
Figure 7.1. Circuit models between pairs of ports.	53
Figure 7.2. RC parallel model of a port.	55
Figure 7.3. Number of the reduced circuit elements versus number of ports..	59
Figure 7.4. Simulation time of the reduced circuits versus number of ports.	59
Figure 7.5. Simulation results of the clock network..	60
Figure 7.6. A clock network with nonlinear drivers and loads..	61
Figure 7.7 Comparison of results of the clock network system and its equivalent circuit.	61
Figure 8.1. Input voltage transition and the current source model.	64
Figure 8.2. pMOS current when $t_{pl} < t_r$	66
Figure 8.3. pMOS current i_p and nMOS current (leakage current) i_n	73

Figure 8.4. An RC circuit with floating capacitors.....	75
Figure 8.5. Voltage response of the RC circuit.....	76
Figure 8.6. A grid-type clock network.	76
Figure 8.7. Voltage response of the clock network.....	77
Figure 8.8. A transmission line circuit with a DC path from source to ground.	77
Figure 8.9. pMOS and nMOS currents of the transmission line circuit.....	78
Figure 8.10. Voltage response of the transmission line circuit.....	78
Figure 8.11. Voltage response of a large clock network.....	79
Figure A.1. An ideal interconnect node.	83

List of Tables

Table 5.1. Comparison of the macromodel and TIM.	46
Table 7.1. The clock network reduction for different number of ports	58
Table 7.2. The loop circuit and the mesh circuit reduction results	60

Scattering-Parameter-Based Macromodel for Transient Analysis of Interconnect Networks with Nonlinear Terminations

Haifang Liao

ABSTRACT

An efficient method for analyzing general distributed-lumped interconnect networks with linear or nonlinear loads for transient simulation is presented. The method is based on scattering parameter techniques. The reduced-order approximate models of linear networks with multiple inputs and outputs can be obtained in one pass of reduction. Only two operations are used to do circuit reduction. The circuit partition is efficiently combined with the circuit reduction process to speed up the reduction process, decrease the simulation time and achieve the high accuracy using lower order approximation. The partition is very suitable for large interconnect networks with large number of external ports. For transmission line networks, we can easily capture time-of-flight delay explicitly during the reduction, which greatly improves the accuracy of the macromodel. Mixed Exponential Functions (MEFs) are used to approximate the scattering parameters of the macromodel. MEF preserves the high accuracy of Padé approximation and yield a guaranteed stable solution. With recursive convolution formulas, the macromodel has been integrated into SPICE-like simulators. The interconnect networks with nonlinear terminations also can be analyzed by replacing the linear parts with lower order equivalent circuits. We developed a practical circuit reduction method to reduce large RC interconnect networks into lower order equivalent RC circuits. In conjunction with the macromodel with explicit convolution formulas, a CMOS driver model is also presented for the transient analysis and power dissipation analysis. The model takes into account the input slope effects, CMOS nonlinear effects and load interconnect effects. The macromodel and the driver model provide accuracy comparable to that of SPICE, with one or two orders of magnitude less computing time.

Keywords: macromodel, scattering parameter, circuit reduction, circuit partition, circuit synthesis, time-of-flight, recursive convolution, transmission line, CMOS driver model, transient analysis, timing analysis, power dissipation analysis.

Acknowledgment

I would like to thank Professor Wayne Dai, my advisor, for his guidance, encouragement, enthusiasm and unceasing support throughout the course of my degree study. His insights and suggestions have given me an enormous amount of help in pursuing this research. I want to thank Professor Pak Chan and Professor Patrick Mantey for reviewing this thesis and giving very helpful suggestions. Many thanks also to all members of our research group, but particularly, Jimmy Wang for his suggestion regarding capturing of time-of-flight delay of transmission line networks.

I would like to thank Dr. Rui Wang of Intel for many useful discussion at the early stage of this research, Dr. Peter Saviz of Intel and Dr. Norman Chang of HP for providing industry circuits and their help to integrate the macromodel into Intel and HP internal simulators, and Zhong L. Mo of EPIC and John Chow of Archer for providing testing benchmarks and evaluating simulation results. I also like to thank Professor Andrew Yang of University of Washington for providing their Model Independent SIMulator (MISIM).

Mostly, I would like to thank my wife, Yanhua, for her discussion, understanding, tolerance and much needed encouragement, and my daughter, Celina, who more than anyone inspired me to finish this work.

CHAPTER 1 Introduction

As circuit switching speed increases, the interconnect behavior has become the main factor to impose limitations on high performance systems. However, detailed analyses of interconnects are usually computationally expensive due to the distributed and dispersive nature of the network. To extract accurately the interconnect networks, the large number of internal nodes or circuit elements can overwhelm any circuit simulator. But the detailed inner working of these nodes are usually of little interests to the system designer and, therefore, need not be explicitly represented into computational models. In Chapter 2, we introduce a scattering-parameter-based macromodel of distributed-lumped networks. Scattering parameters are well suited for the characterizing and modeling of linear high frequency devices. The scattering parameters of passive elements always have an absolute values less than one. This dramatically increases the numerical stability of the algorithm. They are also easier to directly measure on broad frequency bands [36]. By introducing a special circuit component called “multiport interconnect node” [46], an efficient network reduction algorithm is developed to reduce the original network into a network containing one multiport component (macromodel) together with sources and loads of interest, which may be nonlinear [48, 49, 51]. The method can handle general RLC and transmission line networks including capacitive or inductive cutsets and loops. Unlike the method in [41] where system admittance matrix is computed one column at a time, our method computes the system matrix in one pass of reduction.

When the electrical length of interconnects becomes a significant fraction of signal wavelength during the fast transient, the conventional lumped-impedance interconnect

model becomes inadequate and transmission line effects must be taken into account for both on-chip and off-chip interconnects. The delay associated with transmission line networks consists of the exponentially charging time and a pure propagation delay representing the finite propagating speed of electromagnetic signals in the dielectric medium. This propagation delay, so called “*time-of-flight delay*”, is particularly evident in long lines. As the time-of-flight of the signal across the interconnect is greater than, or comparable to, the input signal rise-time (i.e. long interconnects), it is most difficult to capture the time-of-flight delay with a finite order approximation [80, 83], whether a finite sum of exponentials [9, 54] or an exponentially decayed polynomial function [19, 49]. Hence, the time-of-flight must be captured explicitly from the transfer function of the circuit. While finding the explicit analytical expression of the transfer function for an arbitrary interconnect system to compute the time-of-flight is impractical, several attempts have been made by others to extract the time-of-flight delay. They either require an explicit analytical expression of the transfer function [18] or can only deal with one set of transmission lines [54, 10]. The extracted time-of-flight of one transmission line is also used as the lower bound of the delay for the lossy transmission lines [80, 83]. In Chapter 3, we present a new method based on a scattering-parameter-macromodel to compute the time-of-flight for arbitrary interconnect systems, not limited to one transmission line [52]. The accuracy of output responses, due to the extraction of the time-of-flight, is greatly improved.

While integrating transmission line simulation in a transient circuit simulator, the fundamental difficulty arises because the circuits containing nonlinear devices must be characterized in the time domain while transmission lines with loss, dispersion, and interconnect discontinuities are best modeled in the frequency domain. To cope with this difficulty, direct convolution techniques are used. The system outputs are the convolutions of the inputs with the impulse responses. While explicit analytical expression of the impulse responses is impractical, the numerical inverse Fast Fourier Transformation technique suffers from the fact that excessive number of frequency points are needed to avoid aliasing effects. Another drawback of the direct convolution is the computing time it consumes.

In order to deal with these difficulties, Padé technique is used to get an approximated explicit analytical expression of the transfer function [60, 54, 21, 48, 64]. Impulse response functions are approximated with a sum of exponential functions in time domain.

However, the Padé technique suffers from the instability problem: unstable poles may be generated for known stable networks [16]. Instead of using Padé technique, based on the method of inversion of Laplace transform, F. Y. Chang [19] introduced Laguerre functions (or Exponentially-Decayed Polynomial Functions) to approximate the impulse response functions, together with a recursive convolution formula. But the accuracy of the Laguerre approximation is very sensitive to the time constant, which is chosen based on a very rough empirical formula. In Chapter 4, we propose an improved method to choose the time constant by introducing an error function [49]. The further improvement has been done by introducing a Mixed-Exponential Function (MEF) [51] which takes advantage of the best properties of Padé approximation and Exponentially-Decayed Polynomial Function (EDPF), this method efficiently achieves the high accuracy afforded by the former, and high stability by the latter. Furthermore, when the macromodel is incorporated into a SPICE-like simulator, more constraints must be imposed on the macromodel. The moment matching techniques including Padé approximation and EDPF approximation may create a non-positive real Y-matrix (or equivalently non-bounded real S-matrix). The positive-real Y-matrix (or bounded real S-matrix) is the necessary and sufficient condition for the corresponding network to be passive [40]. A non-positive-real Y-matrix may result in unstable simulation. A practical testing method is introduced in Chapter 5.

To avoid folding, sliding, multiplication, and integration in the progressively time-consuming direct convolution integration process for incorporating the macromodel into SPICE-like simulators, the recursive formula [73] is rediscovered [54, 64] for computing convolution of exponential function with any other function. A recursive convolution formula of exponential decayed polynomial function with any other function is also derived by F. Y. Chang [19]. In Chapter 5, we simplify recursive convolution formula [50], and derive the Norton equivalent circuit of the macromodel to handle nonlinear terminations. This macromodel has been incorporated into MISIM [82], the Intel internal simulator TIM and the HP internal simulator HSPICE, with a simplified recursive convolution formula [50].

The macromodels generated by moment matching technique are the approximation of the port behavior of the interconnects. In order to incorporate the macromodels into SPICE-like simulator, they are usually described by an admittance matrix (Y-matrix) [64, 40]. The Y-matrices are generally full matrices. When the number of external ports of an interconnect network is large, the number of entries of the matrix may be larger than the

number of circuit elements of the original interconnect network. For example, a clock network with 500 fanouts will create more than 250,000 entries of the corresponding Y-matrix. This huge full matrix may even increase the computational burden of traditional SPICE simulation. On the other hand, trying to model the huge full matrix by lower order approximation with moment matching techniques is impractical. Severe numerical problems may result in extremely ill-conditioned matrices for computing high order moments. The “complex-frequency hopping” method [22] and the PVL (Padé approximation via the Lanczos process) [29] make great efforts to compute high order moments by avoiding the ill-conditioning problem. However, increasing the approximation order will increase the modeling and simulation time. There is no efficient method to obtain the reduced-order approximate models of linear networks, especially when a large circuit with large number of ports is terminated with nonlinear drivers and loads. The difficulty of reducing a large linear network with large number of ports can be dealt with by partitioning the network into smaller subnetworks. A linear time algorithm is proposed in Chapter 6 to partition a large interconnect network into subnetworks. Each of them can be approximated with a lower-order model. Circuit partitioning speeds up the reduction process, decreases the simulation time and achieves the high accuracy using lower order approximation. The accuracy is controlled by adjusting the number of circuit elements in each subnetwork. The circuit partitioning provides a feasible and efficient way to handle large circuits with large number of external ports.

A new strategy to analyze large interconnect circuits with nonlinear terminations is to synthesize the interconnect networks with lower-order equivalent circuits. A circuit synthesis method based on moment matching technique is presented in Chapter 7 to generate RC interconnects. Since the resistance and capacitance of the synthesized circuits are positive, the circuits are always stable. The reduced circuits can be simulated using existing circuit simulators without modification.

Despite the increasing importance of interconnects in transient analysis, nonlinear active devices in a system also contribute to the system behavior significantly. A new CMOS driver model is proposed in Chapter 8 to be used in conjunction with the macromodel of interconnect networks for transient and power dissipation analysis. This model considers the input slope effects, CMOS nonlinear effects and load interconnect effects. The driver output current is represented by a linear-quadratic-exponential piecewise model. Based on this model, we can accurately evaluate the CMOS transient

leakage (short-circuit) current and short-circuit power dissipation.

Overall, the main contributions of this research include

- Development of an efficient network reduction method to reduce general interconnect networks into a macromodel.
- Specification of a precise definition of time-of-flight and creation of a lower order computational model of general interconnect systems keeping the track of time-of-flight delay, which greatly improve the macromodels of transmission line networks.
- The improved Exponential Decayed Polynomial Function approximation obtained by choosing the optimal time constant, and a proposed Mixed-Exponential Function to approximate the higher order circuit systems.
- Simplification of the formula of recursive convolution with exponential decayed polynomial function and derivation of a Norton equivalent circuit to incorporate the macromodel into SPICE-like simulator.
- Development of a linear time partitioning algorithm and efficiently combine the partitioning with reduction process, which is very efficient to analyze large networks with large number of external ports.
- Synthesis of multiport interconnect networks without transformers. Complex networks are approximated with lower order equivalent circuits, and the equivalent circuits are always stable.
- Creation of a new CMOS driver model which can be used in conjunction with interconnect macromodel. Leakage current and power dissipation can be analyzed.

CHAPTER 2 Scattering-Parameter-Based Macromodel

Linear high speed interconnect networks have been studied quite extensively [15, 19, 22, 54, 60, 69] because of the ever increasing demand of high performance systems. Detailed analyses of interconnects are usually computationally expensive due to the distributed and dispersive nature of the network. To extract accurately the interconnect networks, the large number of internal nodes or circuit elements can overwhelm any circuit simulator. However, the detailed inner working of these nodes are usually of little interests to the system designer and, therefore, need not be explicitly represented into computational models. In this chapter, we introduce a scattering-parameter-based macromodel of distributed-lumped networks. By introducing a special circuit component called multiport interconnect node [46], an efficient network reduction algorithm with only two merging rules is developed to reduce the original network into a network containing one multiport component (macromodel) together with sources and loads of interest, which may be nonlinear [48, 49, 51]. The method can handle general RLC and transmission line networks including capacitive or inductive cutsets and loops. Unlike the method in [41] where system admittance matrix is computed one column at a time, our method computes the system matrix in one pass of reduction.

2.1 Scattering Parameters of Distributed-Lumped Components

We use scattering parameters (S-parameters) to describe the components of

interconnect systems. Scattering parameters are a powerful way to describe and model interconnects. They can be measured directly at high frequencies and they exist for all distributed-lumped circuit elements including open and short circuits. They can also describe transmission lines which is critical in today's high-speed designs. A scattering matrix is employed to relate outgoing waves to incoming waves of a multiport [26]. For an n port component, the scattering matrix of the component can be defined as

$$S_{ji}(s) = \left. \frac{b_j}{a_i} \right|_{a_k = 0, k \neq i} \quad i, j = 1, 2, \dots, n \quad (2.1)$$

where s is the complex frequency, a_i and b_j are the incoming voltage wave at port i and the outgoing wave at port j , respectively. The wave parameters a_i and b_i have a clear and definite relationship with circuit parameters. Let V_i and I_i be voltage and current respectively at port i . Then they are related

$$a_i + b_i = V_i \quad \text{and} \quad a_i - b_i = Z_0 I_i \quad (2.2)$$

where Z_0 is the reference impedance. Eqs. (2.1) and (2.2) can be used to derive scattering parameters of some basic components.

The components utilized to characterize a general interconnect network can be classified into four types [49]: 1) one-port impedance, 2) two-port impedance, 3) lossy transmission line and 4) multiport interconnect node (See Fig. 2.1). The scattering parameters of the first three components can be derived as follows [26]:

For the one port element in Fig. 2.1(a), $V = ZI$. Considering Eq. (2.2), we have

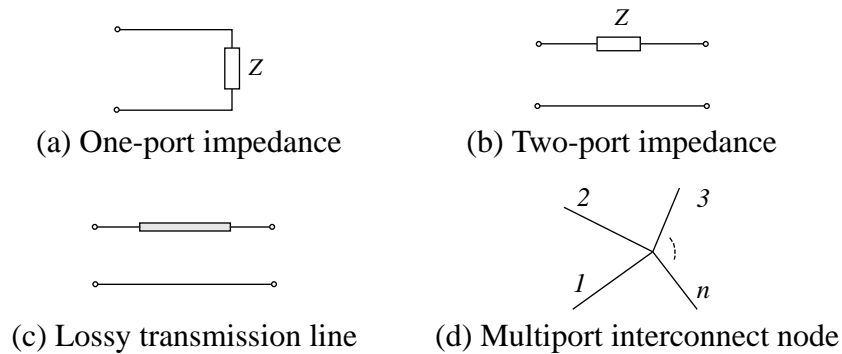


Figure 2.1. Four basic elements.

$$a + b = Z(a - b) / Z_0 \quad (2.3)$$

Thus,

$$S = \frac{b}{a} = \frac{Z - Z_0}{Z + Z_0} \quad (2.4)$$

For the two-port, according to Kirchoff's voltage law and current law,

$$\begin{aligned} V_1 &= ZI_1 + V_2 \\ I_1 + I_2 &= 0 \end{aligned} \quad (2.5)$$

Transferring the circuit parameters to wave parameters with Eq. (2.2),

$$\begin{aligned} a_1 + b_1 &= Z(a_1 - b_1) / Z_0 + a_2 + b_2 \\ a_1 - b_1 + a_2 - b_2 &= 0 \end{aligned} \quad (2.6)$$

and solving the above equation, we have

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = S \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \quad (2.7)$$

where S-matrix

$$S = \frac{1}{Z + 2Z_0} \begin{bmatrix} Z & 2Z_0 \\ 2Z_0 & Z \end{bmatrix} \quad (2.8)$$

For an RLC transmission line, the telegrapher's equations are

$$\begin{aligned} \frac{\partial}{\partial x} V(s, x) &= -(sL + R) \\ \frac{\partial}{\partial x} I(s, x) &= -sC \end{aligned} \quad (2.9)$$

where R , L and C are per-unit-length resistance, inductance and capacitance, respectively. x is the distance from the left end of the transmission line. For the line with length l , the solution of Eq. (2.9), with boundary conditions of (V_1, I_1) and (V_2, I_2) at two ends of the line respectively, is

$$\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \frac{1}{Z_c} \begin{bmatrix} \coth(\gamma l) & -\operatorname{csch}(\gamma l) \\ -\operatorname{csch}(\gamma l) & \coth(\gamma l) \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} \quad (2.10)$$

where $Z_c = \sqrt{(R + sL)/sC}$ is the characteristic impedance, $\gamma = \sqrt{(R + sL)sCl}$ is the propagation constant. Combining Eq. (2.2) and (2.10), we obtain the S-matrix of the lossy transmission line,

$$\mathbf{S} = \frac{1}{2Z_0Z_c \cosh(\gamma) + (Z_c^2 + Z_0^2) \sinh(\gamma)} \begin{bmatrix} (Z_c^2 - Z_0^2) \sinh(\gamma) & 2Z_cZ_0 \\ 2Z_cZ_0 & (Z_c^2 - Z_0^2) \sinh(\gamma) \end{bmatrix} \quad (2.11)$$

The interconnect topology as shown in Fig. 2.1(d) can be used to connect components described in Fig. 2.1(a) through Fig. 2.1(c) to construct a generalized interconnection network. The n port interconnect node can also be described by the scattering matrix [46]:

$$\mathbf{S} = \frac{1}{n} \begin{bmatrix} 2-n & 2 & \dots & 2 \\ 2 & 2-n & \dots & 2 \\ \dots & \dots & \dots & \dots \\ 2 & 2 & \dots & 2-n \end{bmatrix} \quad (2.12)$$

(Please see Appendix A for the detailed derivation). Combining these four basic elements and all multiport elements described by S-parameters, one can represent a variety of distributed-lumped network topology including capacitive cutsets, inductive loops, and lossy transmission lines.

2.2 Scattering-Parameter-Based Macromodel

Given the individual component scattering parameters, we describe a systematic reduction algorithm to reduce a distributed-lumped network to a multiport with sources and loads of interest.

The network reduction problem can be defined as follows[48, 49]: given a linear distributed-lumped network described by S-parameters, find a multiport representation of the network as illustrated in Fig. 2.2., where the multiport is characterized by its S-parameters. All nodes in the network are internal to the multiport except the node connected to the driving source (node 1) and the loads of interest (nodes 2 through n).

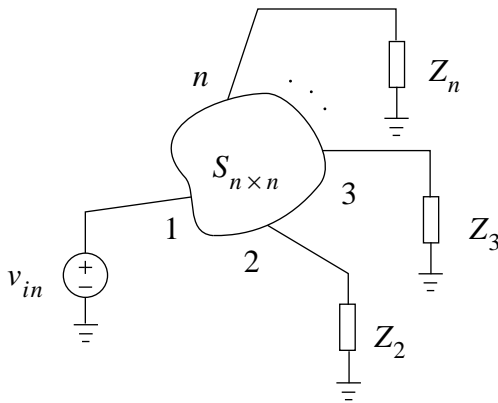


Figure 2.2. S-parameter-based macromodel.

These external nodes are specified by the user.

To obtain such a multiport representation with external ports from an arbitrary distributed-lumped network of original nodes, the network is reduced by merging the nodes into the multiport one at a time while keeping all user specified nodes external. There are two basic reduction rules[48, 49]:

Adjoined Merging Rule: Let X and Y be two adjacent multiports, with m ports and n ports respectively. Assume port k of X is connected to port l of Y , as shown in Fig. 2.3. After merging X and Y , the resultant $(n + m - 2)$ port has the following S-parameters:

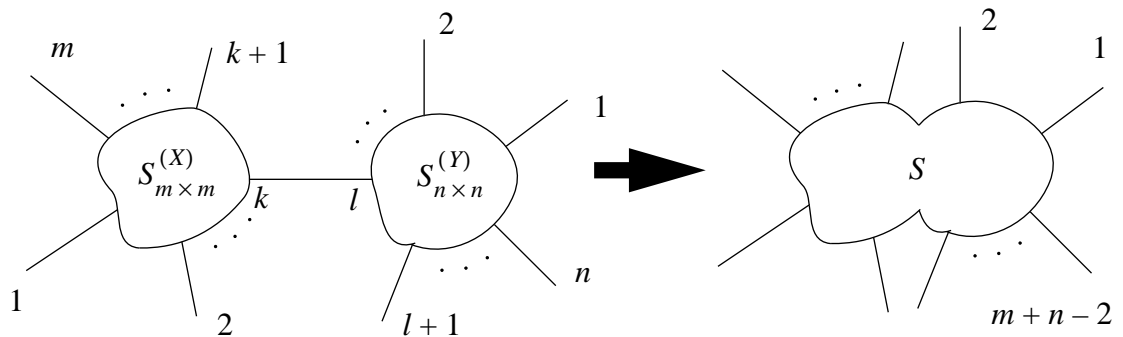


Figure 2.3. Adjoined merging.

$$S_{ji} = \begin{cases} S_{ji}^{(X)} + \frac{S_{ki}^{(X)} S_{ll}^{(Y)} S_{jk}^{(X)}}{1 - S_{kk}^{(X)} S_{ll}^{(Y)}} & i, j \in X \\ S_{ji}^{(Y)} + \frac{S_{li}^{(Y)} S_{kk}^{(X)} S_{jl}^{(Y)}}{1 - S_{kk}^{(X)} S_{ll}^{(Y)}} & i, j \in Y \\ \frac{S_{ki}^{(X)} S_{jl}^{(Y)}}{1 - S_{kk}^{(X)} S_{ll}^{(Y)}} & i \in X, j \in Y \end{cases} \quad (2.13)$$

Self Merging Rule: Let X be an m port with a self loop connected to port l and k , as shown in Fig. 2.4. After eliminating the self loop, the resultant $(m-2)$ port has the

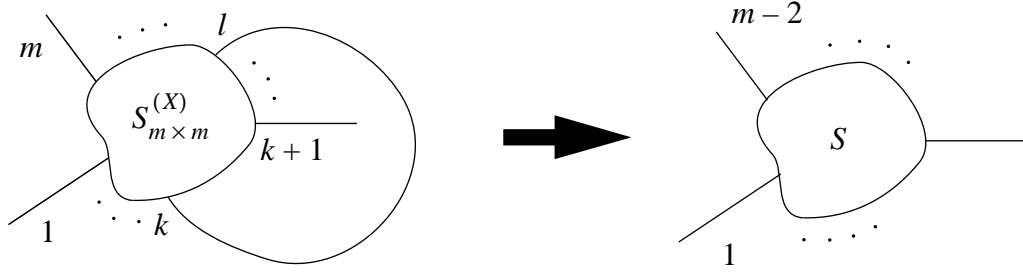


Figure 2.4. Self merging.

following S-parameters:

$$S_{ji} = S_{ji}^{(X)} + S_{jl}^{(X)} a_l + S_{jk}^{(X)} a_k \quad i, j = 1, 2, \dots, m-2 \quad (2.14)$$

where

$$\begin{aligned} a_l &= \frac{1}{\Delta} (S_{li}^{(X)} S_{kk}^{(X)} + S_{ki}^{(X)} (1 - S_{lk}^{(X)})) \\ a_k &= \frac{1}{\Delta} (S_{ki}^{(X)} S_{ll}^{(X)} + S_{li}^{(X)} (1 - S_{kl}^{(X)})) \\ \Delta &= (1 - S_{lk}^{(X)}) (1 - S_{kl}^{(X)}) - S_{ll}^{(X)} S_{kk}^{(X)} \end{aligned} \quad (2.15)$$

For an arbitrary distributed-lumped network described by the linear components, the Adjoined Merging Rule is used to merge all internal components, and the Self Merging Rule is applied to eliminate the self loops introduced by the Adjoined Merging process.

The macromodel, or the voltage transfer function of the network, can be characterized by the S-parameters of the multiport component resulted from the reduction process, together with the S-parameters of the loads.

It should be pointed out that the above reduction process does not require the network be an RC tree. Since we start with the S-parameter description of the system, which always exists for any physically realizable system, the formulation is completely general for any linear distributed-lumped network with scattering parameter descriptions. Another advantage is that the need for using lumped representation of transmission lines is eliminated since lossy transmission lines can be represented in a distributed form.

Carrying out the Taylor series expansion, all components can be represented in the form of truncated series:

$$S(s) \approx \hat{S}(s) = \sum_{i=0}^n q_i s^i \quad (2.16)$$

where q_i is the coefficient of the expansion.

In order to match the initial condition of the system, the asymptotic frequency point ($s = \infty$) should be added into the above expansion:

$$\hat{S}'(s) = S(\infty) + \sum_{i=0}^n q_i s^i \quad (2.17)$$

In the above equation, the component steady state response is represented by setting $s = 0$ and the initial condition is satisfied by $S(\infty)$ as required by the initial and final value theorems of Laplace transforms.

To complete the series expansion for each component and network reduction, we define two types of series operations. Let

$$A = \sum_{i=0}^n a_i s^i, \quad B = \sum_{i=0}^n b_i s^i \quad (2.18)$$

Then, a *Multiplication Operation* is defined as

$$C = A \times B \approx \sum_{i=0}^n c_i s^i \quad \text{with } c_i = \sum_{j=0}^i a_j b_{i-j} \quad (2.19)$$

And a *Division Operation*

$$D = A/B \approx \sum_{i=0}^n d_i s^i \text{ with } d_i = \frac{1}{b_0} (a_i - \sum_{j=0}^{i-1} d_j b_{i-j}) \quad (2.20)$$

From Eq. (2.20), it seems that a negative first moment d_{-1} would be created if $b_0 = 0$. This case may happen when using admittance matrix or state equations. However, since we use scattering parameters to characterize all components, all scattering parameters of passive components have no poles at $s = 0$ based on the principle of energy conservation [67]. Therefore, if D represents an element of a scattering matrix, it is guaranteed that there is no pole at $s = 0$. This implies that if $b_0 = 0$, a_0 must be zero, so we can shift all moments of A and B to the left and keep $b_0 \neq 0$.

Several network reduction algorithms have been reported. The multiport connection method was introduced in [26, 34, 57]. In this method, the scattering-matrix of the network is partitioned into blocks based on classification of internal and external ports. The scattering-matrix can be obtained using block operations including time-consuming matrix inversion. Connecting two multiports at a time from bottom up may reduce the computation time. However, matrix inversion is still unavoidable. Kuhn [43] introduced a flow graph reduction method. A microwave network is represented by a flow graph and the graph can be reduced step-by-step based on four reduction rules. This method, however, has not been widely used because of its complexity with large networks. Here, we introduce two simple rules for fast network reduction. While Kuhn reduces the network one branch at a time, we reduce the network one node at a time. Unlike the previous approaches, our method approximates S-parameter by expanding them into Taylor series and reduces networks with series operations, which further improves the efficiency of our reduction process. In a later section, we will show the approximation is accurate for delay computation.

Without loss of generality, assume a two-port network is the result of the network

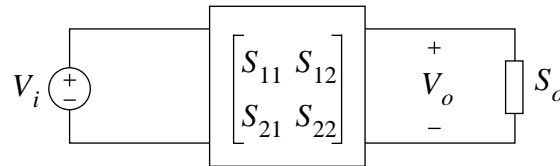


Figure 2.5. A two port network.

reduction process (See Fig. 2.5). The the voltage transfer function of the network can be characterized by the S-parameters of the multiport component resulted from the reduction process, together with the S-parameters of the loads. From this reduced network, we can easily get the transfer function.

$$H(s) = \frac{S_{21}(1 + S_o)}{1 + S_{11} + S_o(S_{12}S_{21} - S_{11}S_{22} - S_{22})} \quad (2.21)$$

where S_o is the S-parameter of the load.

CHAPTER 3 Capturing Time-of-flight Delay

In last chapter, a general interconnect network reduction method is described. A large network is reduced to a multiport macromodel. The Padé technique [60] may be used to approximate the macromodel to analyze the system transient responses. However, as the electrical length of interconnects becomes a significant fraction of signal wavelength during the fast transient, the conventional lumped-impedance interconnect model becomes inadequate and transmission line effects must be taken into account for both on-chip and off-chip interconnects. The delay associated with transmission line networks consists of the exponentially charging time and a pure propagation delay representing the finite propagating speed of electromagnetic signals in the dielectric medium. This propagation delay, so called time-of-flight delay, denoted by τ_f , is particularly evident in long lines (See Fig. 3.1). As the time-of-flight of the signal across the interconnect is greater than, or comparable to, the input signal rise-time (i.e. long interconnects), it is most difficult to capture the time-of-flight delay with a finite order of approximation[80, 83], whether a finite sum of exponentials[9, 54] or an exponentially decayed polynomial function [19, 49].

Hence, the time-of-flight τ_f , (more precisely the factor $e^{-s\tau_f}$), must be captured explicitly from the transfer function of the circuit. As we know, a transfer function will be called ideal if it is of the form $H(s) = e^{-s\tau_f}$. For $s = j\omega$, the magnitude identically equals to one, and the angle is proportional to the angle frequency ω . According to the shifting

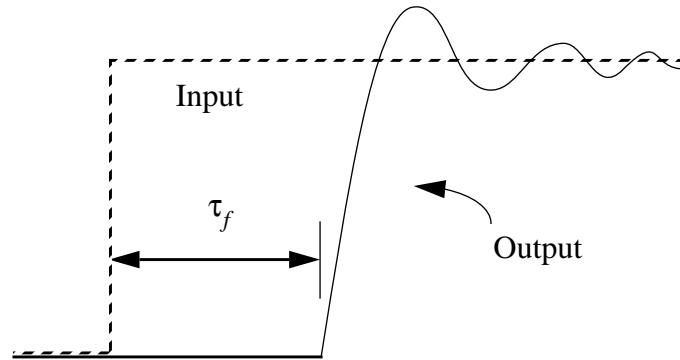


Figure 3.1. Time-of-flight delay τ_f .

theorem of Laplace transform, if this ideal network is excited by a signal $e(t)$, the corresponding response of the network will be $e(t - \tau_f)$. The response is the same as the excitation except that it is delayed in time by an amount τ_f . That is, the response for $t < \tau_f$ is equal to zero. Therefore, the time-of-flight is defined as the maximum delay during which the output response is zero for any finite input. The corresponding factor in the frequency domain is $e^{-s\tau_f}$.

While finding the explicit analytical expression of the transfer function for an arbitrary interconnect system to compute the time-of-flight is impractical, several attempts have been made to extract the time-of-flight delay. They either require an explicit analytical expression of the transfer function[18] or can only deal with one set of transmission lines[54, 10]. The extracted time-of-flight of one transmission line is also used as the lower bound of the delay for the lossy transmission lines[80, 83].

Here, we present a new method to compute the time-of-flight for arbitrary interconnect systems, not limited to one transmission line [52]. The method is based on a scattering-parameter-macromodel described in last Section. The accuracy of output responses, due to the extraction of the time-of-flight, is greatly improved.

3.1 Properties of Time-of-Flight

Recall that the time-of-flight is the maximum delay during which the output response is zero for a finite input signal. The computation can be approached from the properties of transfer function in the frequency domain. The following theorem is useful

to capture the time-of-flight for the transfer function $H(s)$.

Theorem 3.1: For any $\varepsilon > 0$, there exists positive constant s_0 , such that the time-of-flight τ_f of the transfer function $H(s)$ satisfies the following

$$e^{-\varepsilon s} < |H(s)|e^{\tau_f s} < e^{\varepsilon s} \text{ for all } s \geq s_0 \quad (3.1)$$

Proof: Let $v_i(t)$ be a finite input signal, and $V_i(s)$ be its Laplace transform. According to the initial value theorem of Laplace transform, $\lim_{s \rightarrow \infty} sV_i(s) = v_i(0^+)$.

From the definition of the time-of-flight, the corresponding time domain response $v_o(t)$ is zero for $t < \tau_f$. According to the shifting theorem, the Laplace transform of the function $\tilde{v}_o(t) = v_o(t + \tau_f)$ is

$$\begin{aligned} \tilde{V}_o(s) &= e^{\tau_f s} V_o(s) \\ &= e^{\tau_f s} H(s) V_i(s) \end{aligned}$$

Obviously, $\tilde{v}_o(0^+)$ is finite for any passive networks when the input is finite. Now, let us prove Eq. (3.1) by contradiction. First, assume that there exists $\varepsilon > 0$ such that $|H(s)|e^{\tau_f s} \geq e^{\varepsilon s}$ for all $s \geq s_0$, then

$$\begin{aligned} \left| \tilde{v}_o(0^+) \right| &= \lim_{s \rightarrow \infty} s \left| \tilde{V}_o(s) \right| \\ &= \lim_{s \rightarrow \infty} s e^{\tau_f s} |H(s)| |V_i(s)| \\ &= \left| v_i(0^+) \right| \lim_{s \rightarrow \infty} e^{\tau_f s} |H(s)| \\ &\geq \left| v_i(0^+) \right| \lim_{s \rightarrow \infty} e^{\varepsilon s} \rightarrow \infty \end{aligned}$$

This contradicts the fact that $\tilde{v}_o(0^+)$ is a finite value. Thus, $|H(s)|e^{\tau_f s} < e^{\varepsilon s}$ is held.

On the other hand, assume that there exists $\varepsilon > 0$ such that $e^{-\varepsilon s} \geq |H(s)|e^{\tau_f s}$ for all $s \geq s_0$, then the initial value of the function $\bar{v}_o(t) = v_o(t + \tau_f + \sigma)$ for some positive $\sigma < \varepsilon$ is

$$\begin{aligned}
|\bar{v}_o(0^+)| &= \lim_{s \rightarrow \infty} s |\bar{V}_o(s)| \\
&= \lim_{s \rightarrow \infty} s e^{(\tau_f + \sigma)s} |H(s)| |V_i(s)| \\
&= |v_i(0^+)| \lim_{s \rightarrow \infty} e^{(\tau_f + \sigma)s} |H(s)| \\
&\leq |v_i(0^+)| \lim_{s \rightarrow \infty} e^{(\sigma - \varepsilon)s} \rightarrow 0
\end{aligned}$$

That is, the output response $v_o(t)$ is still zero at $t = \tau_f + \sigma$ for any finite input signal. This contradicts the definition: the time-of-flight is the maximum delay during which the output response is zero for a finite input signal. Thus $|H(s)|e^{\tau_f s} > e^{-\varepsilon s}$ is held. \square

Notice that the definition of time-of-flight in [6], referred by others (e.g. in [18]), is

$$\tau_f = \lim_{s \rightarrow \infty} -\frac{1}{2} \frac{d}{ds} \left[\ln \frac{H(s)}{H(-s)} \right] \quad (3.2)$$

This definition may be incorrect for some special cases. For example, the transfer function of the transmission line circuit shown in Fig. 3.2 is

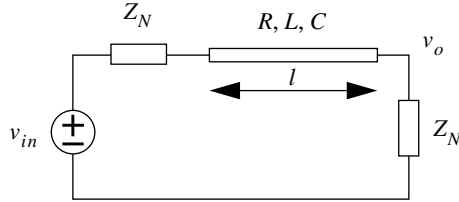


Figure 3.2. Transmission line circuit.

$$H(s) = \frac{2Z_N Z_c}{(Z_N + Z_c)^2 e^{\gamma} - (Z_N - Z_c)^2 e^{-\gamma}} \quad (3.3)$$

where $Z_c = \sqrt{(R + sL) / (sC)}$ and $\gamma = \sqrt{(R + sL) sCl}$. Obviously, its time-of-flight is $\tau_f = \sqrt{LCl}$ according to the above theorem. This can also be easily verified because of the speed of electromagnetic wave is $1/\sqrt{LC}$. But according to Eq. (3.2), it is zero.

Theorem 3.1 gives a precise description of time-of-flight. But it is difficult to compute the time-of-flight for a large network directly based on the theorem since it is

impractical to get an explicit expression of the transfer function. To cope this problem, we introduce following three corollaries.

Corollary 3.1: If the time-of-flight delays of non-zero functions $F_1(s)$ and $F_2(s)$ are $\text{TOF}(F_1(s))$, and $\text{TOF}(F_2(s))$ respectively, then the time-of-flight delay of $F_1(s) \pm F_2(s)$ is

$$\text{TOF}(F_1(s) \pm F_2(s)) = \min(\text{TOF}(F_1(s)), \text{TOF}(F_2(s))) \quad (3.4)$$

Corollary 3.2: If the time-of-flight delays of non-zero functions $F_1(s)$ and $F_2(s)$ are $\text{TOF}(F_1(s))$, and $\text{TOF}(F_2(s))$ respectively, then the time-of-flight delay of $F_1(s)F_2(s)$ is

$$\text{TOF}(F_1(s)F_2(s)) = \text{TOF}(F_1(s)) + \text{TOF}(F_2(s)) \quad (3.5)$$

Corollary 3.3: If the time-of-flight delays of non-zero functions $F_1(s)$ and $F_2(s)$ are $\text{TOF}(F_1(s))$, and $\text{TOF}(F_2(s))$ respectively, then the time-of-flight delay of $F_1(s)/F_2(s)$ is

$$\text{TOF}(F_1(s)/F_2(s)) = \text{TOF}(F_1(s)) - \text{TOF}(F_2(s)) \quad (3.6)$$

Above corollaries can be easily proved according to the Theorem 3.1. These results will be used later to keep track of the time-of-flight during the network reduction. Later we will show that these corollaries have clear physical meaning during network reduction. Let us first describe the time-of-flight delays of scattering parameters of basic circuit components.

3.2 Scattering Parameters of Components with Time-of-Flight Captured

In order to extract time-of-flight delays of interconnect systems, let us review the four basic circuit elements shown in Fig. 2.1. Their S-parameter are expressed in Eqs. (2.4-2.12). In these four basic elements, the one-port impedance, the two-port impedance and multiport interconnect node are all lumped components, i.e., electromagnetic waves propagate across the component virtually instantaneously. Therefore, the time-of-flight delays of S-parameters described in Eqs. (2.4, 2.8, 2.12) are all equal to zero.

Applying Theorem 3.1 to the S-matrix of the lossy transmission line (See Eq. (2.11)), we find that the time-of-flight delays of both S_{11} and S_{22} are zero, but the time-of-flight delays of S_{12} and S_{21} are $\tau_f = \sqrt{LC}l$. Let us rewrite Eq. (2.11) as

$$S = \begin{bmatrix} S'_{11} & e^{-\tau_f s} S'_{12} \\ e^{-\tau_f s} S'_{21} & S'_{22} \end{bmatrix} \quad (3.7)$$

where

$$S'_{11} = S'_{22} = \frac{(Z_c^2 - Z_0^2) \sinh(\gamma)}{2Z_0 Z_c \cosh(\gamma) + (Z_c^2 + Z_0^2) \sinh(\gamma)} \quad (3.8)$$

$$S'_{12} = S'_{21} = \frac{2Z_c Z_0 e^{\tau_f s}}{2Z_0 Z_c \cosh(\gamma) + (Z_c^2 + Z_0^2) \sinh(\gamma)} \quad (3.9)$$

3.3 Keeping Track of Time-of-Flight

For an arbitrary distributed-lumped network described with S-parameters, the Adjoined Merging Rule and the Self Merging Rule are employed to reduce the original network into a multiport component. For scattering parameters of basic components, we can find the corresponding time-of-flight delays according to the Theorem 3.1 as described in Section 3.2. During network reduction, there are only four fundamental operations: addition, subtraction, multiplication and division. The three corollaries we derived in Section 3.1 are applied to keep the track of the time-of-flight.

Time-of-flight is the maximum delay during which the output response is zero for any finite input, or is the minimum time at which the output has non-zero response. In order to explain the physical meaning of the three corollaries, let us go back to Eq. (2.13). S_{ji} relates the outgoing wave at port j to the incoming wave at port i . If both port i and port j belong to the same component, say X , then S_{ji} is

$$S_{ji} = S_{ji}^{(X)} + \frac{S_{ki}^{(X)} S_{ll}^{(Y)} S_{jk}^{(X)}}{1 - S_{kk}^{(X)} S_{ll}^{(Y)}} \quad i, j \in X \quad (3.10)$$

S_{ji} consists of two terms, in the other word, there are two paths for electromagnetic wave to propagate from port i to port j . The first term of Eq. (3.10) represents the first path on which the wave directly propagates from port i to port j . The second term represents the second path on which the wave propagates from port i to port k , then reflected to port j (See Fig. 3.3). Obviously, the time-of-flight of the S_{ji} is the minimal of

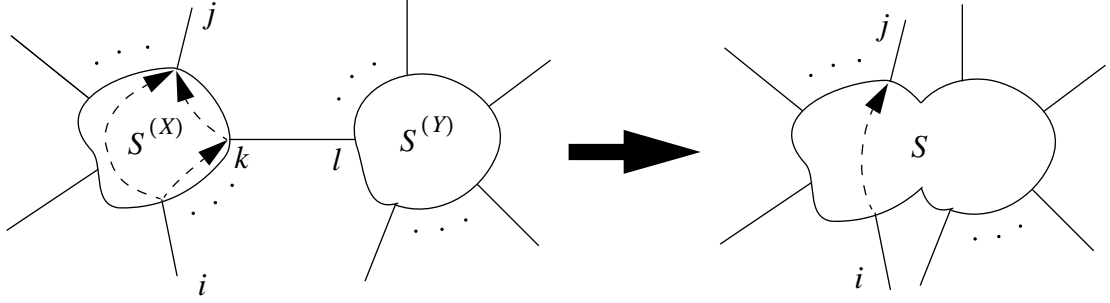


Figure 3.3. For $i, j \in X$, there are two paths for wave to propagate from port i to port j .

the time-of-flight of these two paths, since the time-of-flight is the minimum time at which the output has non-zero response. This is the physical meaning of the Corollary 3.1. The time-of-flight of the second path is the sum of those of $S_{ki}^{(X)}$ and $S_{jk}^{(X)}$, since the path consists of two sub-paths. This is the physical meaning of the Corollary 3.2. From Eqs. (2.13, 2.14 and 2.15), we can see that the S-parameters in denominators are constants or related to the same ports, so the time-of-flight delays of denominators are always zero. Thus, the corollary 3 can be simplified as

Corollary 3.3’: If the time-of-flight delays of the non-zero functions $F_1(s)$ and $F_2(s)$ are $\text{TOF}(F_1(s))$, and $\text{TOF}(F_2(s)) = 0$ respectively, then the time-of-flight delay of $F_1(s)/F_2(s)$ is

$$\text{TOF}(F_1(s)/F_2(s)) = \text{TOF}(F_1(s)) \quad (3.11)$$

Similarly, if port i belongs to component X and port j belongs to component Y , then

$$S_{ji} = \frac{S_{ki}^{(X)} S_{jl}^{(Y)}}{1 - S_{kk}^{(X)} S_{ll}^{(Y)}} \quad i \in X, j \in Y \quad (3.12)$$

There is only one path for electromagnetic wave to propagate from port i to port j . Eq. (3.12) shows that wave propagates from port i to port k of component X , then from port k of component X to port l of component Y , then from port l to port j of component Y (See Fig. 3.4). The time-of-flight of the path is the sum of those of $S_{ki}^{(X)}$ and $S_{jl}^{(Y)}$. The same analysis methods could be applied to Eq. (2.14) for self merging where there are five

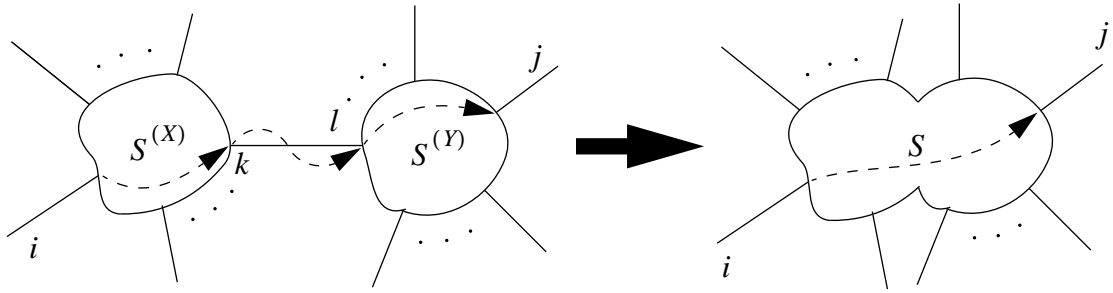


Figure 3.4. For $i \in X, j \in Y$, there is only one path for wave to propagate from port i to port j .

paths for wave to propagate from port i to port j (See Fig 3.5) and Eq. (2.21) for transfer function where there is only one path from the source to the destination.

Based on Corollary 3.1-3.3 of Theorem 3.1, network reduction can be completed with the track of the time-of-flight. Finally, we get the transfer function $H(s)$ in the Taylor series form (that is, the first several moments) with the time-of-flight τ_f . Instead of matching moments of $H(s)$, we match the moments of $H'(s) = e^{\tau_f s} H(s)$. The corresponding time domain function $h'(t)$ can be obtained with the Padé technique [60] or EDPF technique [19, 49]. Since $H(s) = e^{-\tau_f s} H'(s)$, according to the shifting theorem of Laplace transform, the time domain transfer function is

$$h(t) = h'(t - \tau_f)u(t - \tau_f) \quad (3.13)$$

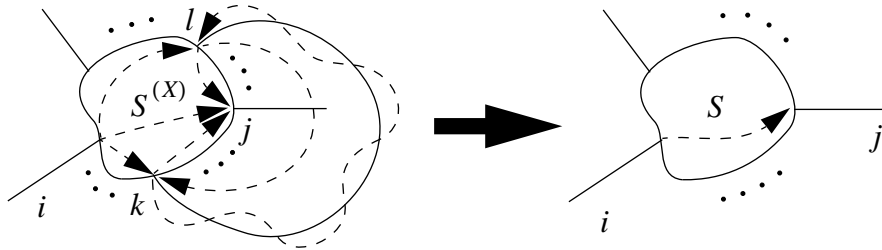


Figure 3.5. For self merging, there are five paths for wave to propagate from port i to port j .

CHAPTER 4 Time Domain Synthesis of the Macromodel

In 1892, in the Scientific Transactions of the Ecole Normale Supérieure in Paris, the French mathematician Henri Padé published an article concerning the approximate representation of a function by rational fractions. Since then, Padé approximation technique has been widely used in numerical analysis, theoretical physics, fluid mechanics and control theory [e.g. 3, 4, 5, 13, 14, 20, 38, 61, 77]. Recently, the Padé technique has been used to get an approximated explicit analytical expression of the transfer function to evaluate charging delay of interconnect networks [60, 54, 21, 48, 64, 29]. However, the Padé technique suffers from an instability problem: unstable poles may be generated for known stable networks [16]. Instead of using the Padé technique, based on the method of inversion of Laplace transform, F. Y. Chang [19] introduced Laguerre function (or Exponentially-Decayed Polynomial Function) to approximate the impulse response functions, together with a recursive convolution formula. But the accuracy of the approximation is very sensitive to the time constant which is chosen based on a very rough empirical formula. We have proposed an improved method [49] to choose the time constant by introducing an error function. The further improvement has been done by introducing a Mixed-Exponential Function (MEF) [51] which takes advantage of the best properties of Padé approximation and Exponentially-Decayed Polynomial Function (EDPF). This method efficiently achieves high accuracy afforded by the former, and high stability by the latter.

4.1 Synthesis with Padé Approximation

Starting from transfer function or S-parameters in the Taylor series form, Padé approximation can be used to analyze systems [68, 5, 60]. A frequency domain transfer function $H(s)$ can be approximated with the following summation of time domain exponential functions using the q -th order Padé approximation [68]:

$$h_p(t) = \sum_{i=1}^q k_i e^{p_i t} \quad (4.1)$$

where p_i and k_i are the poles and residues, respectively. Its corresponding expression in the frequency domain is

$$H_p(s) = \sum_{i=1}^q \frac{k_i}{s - p_i} \quad (4.2)$$

These q poles in Eq. (4.1) are approximated by matching the moments m_0 through m_{2q-1} of the exact impulse response $H(s)$ with those of the lower-order model. This yields the following set of linear equations [60]:

$$\begin{bmatrix} m_0 & m_1 & \dots & m_{q-1} \\ m_1 & m_2 & \dots & m_q \\ \dots & \dots & \dots & \dots \\ m_{q-1} & m_q & \dots & m_{2q-2} \end{bmatrix} \begin{bmatrix} a_q \\ \dots \\ a_2 \\ a_1 \end{bmatrix} = - \begin{bmatrix} m_q \\ m_{q+1} \\ \dots \\ m_{2q-1} \end{bmatrix} \quad (4.3)$$

The roots of the characteristic polynomial formed by the solution of Eq. (4.3):

$$1 + a_1 s + a_2 s^2 + \dots + a_q s^q = 0 \quad (4.4)$$

yield the pole approximation. With the set of pole approximations, the corresponding q residues are obtained by matching the first q moments, resulting in the linear system:

$$\begin{aligned}
\frac{k_1}{p_1} + \frac{k_2}{p_2} + \dots + \frac{k_q}{p_q} &= -m_0 \\
\frac{k_1}{p_1^2} + \frac{k_2}{p_2^2} + \dots + \frac{k_q}{p_q^2} &= -m_1 \\
\dots & \\
\frac{k_1}{p_1^q} + \frac{k_2}{p_2^q} + \dots + \frac{k_q}{p_q^q} &= -m_{q-1}
\end{aligned} \tag{4.5}$$

4.2 Synthesis with Exponentially-Decayed Polynomial Functions

Padé approximation provides a relatively accurate and efficient way to evaluate the response of linear interconnect systems. However, it suffers two known problems. Right-half plane approximate poles may be generated for known stable networks; and the approximation error depends on network topology, element values, input waveform, and output choice [16].

In order to overcome these problems, Exponentially-Decayed Polynomial Functions (EDPF) are used to approximate the transfer function of the macromodel. A q -th order EDPF has the following form:

$$h_e(t) = \left[c_0 + c_1 \left(\frac{t}{d} \right) + c_2 \left(\frac{t}{d} \right)^2 + \dots + c_q \left(\frac{t}{d} \right)^q \right] e^{-(t/d)} \tag{4.6}$$

where d is the time constant. The Laplace transform of EDPF is

$$H_e(s) = \frac{c_0 d}{1 + ds} + \frac{c_1 d}{(1 + ds)^2} + \dots + \frac{q! c_q d}{(1 + ds)^{q+1}} \tag{4.7}$$

EDPF can approximate the time domain transfer function with any degree of accuracy. And its corresponding frequency domain function has only one repeated stable pole, so it is always stable for stable systems.

We will use q -th order EDPF to approximate transfer function in the Taylor series form with $n + 1$ ($n > q$) moments. For a fixed time constant d (we will describe how to determine d later), we compute c_i ($i = 0, \dots, q$) by expanding $H_e(s)$ into Taylor series and matching the first corresponding $q + 1$ moments of Eq. (4.7). The remaining $n - q$ moments will be used to determine time constant d . Let m_0, m_1, \dots, m_n be the first $n + 1$

moments of the transfer function $H(s)$, then we have

$$\begin{bmatrix} x_{00} & x_{01} & \cdots & x_{0q} \\ x_{10} & x_{11} & \cdots & x_{1q} \\ \cdots & \cdots & \cdots & \cdots \\ x_{q0} & x_{q1} & \cdots & x_{qq} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \cdots \\ c_q \end{bmatrix} = \begin{bmatrix} m_0 \\ m_1 \\ \cdots \\ m_q \end{bmatrix} \quad (4.8)$$

or

$$\mathbf{X} \cdot \mathbf{C} = \mathbf{M} \quad (4.9)$$

where $\mathbf{C} = [c_0, c_1, \dots, c_q]^T$, $\mathbf{M} = [m_0, m_1, \dots, m_q]^T$, $\mathbf{X} = [x_{ij} | i, j = 0, \dots, q]$ is the coefficient matrix, where x_{ij} is a function of time constant d and can be easily determined with series multiplication and division operations given in Section 2.2. Notice that we use only $q + 1$ moments to compute the coefficient vector \mathbf{C} . Additional moments will be used to determine the time constant d .

An error criterion is introduced in [21] to reduce error introduced by moment-matching. The criterion is based on the moment skew. For a given moment m_i , and respective approximate moment \hat{m}_i , the i -th moment skew, ε_i , is defined as

$$\varepsilon_i = \left| \frac{m_i - \hat{m}_i}{m_i} \right| \quad (4.10)$$

where \hat{m}_i is the i -th moment of $H_e(s)$. The total moment skew, ε , as an error criterion, is given by

$$\varepsilon = \left(\sum_{i=q+1}^n \varepsilon_i^2 \right)^{1/2} \quad (4.11)$$

For EDPF approximation, ε is the function of the time constant d , that is,

$$\varepsilon = f(d) \quad (4.12)$$

A ‘‘golden section’’ search technique [62] is used to find a time constant which minimize the total moment skew by iteratively computing Eq. (4.9) and Eq. (4.12). The $h_e(t)$ which minimizes the total moment skew ε is selected as the impulse response of the system.

4.3 Synthesis with Mixed-Exponential Function

While EDPF gives a stable approximation, it is found that to obtain the same degree of accuracy, a higher order EDPF function is required compared to Padé approximation when a stable solution can be found for the latter. Hence, it is of computational advantage to choose Padé approximation over EDPF when possible. Comparing the characteristics of Padé approximation and EDPF, we propose a Mixed-Exponential Function (MEF) approximation for the analysis of interconnect networks. MEF is the combination of exponential functions and exponentially-decayed polynomial functions.

As we have described, a frequency domain transfer function $H(s)$ with first $2q$ moments can be approximated in the time domain with the following summation of exponential functions using the q -th order Padé approximation:

$$h_p(t) = \sum_{i=1}^q k_i e^{p_i t} \quad (4.13)$$

where p_i and k_i are the poles and residues, respectively.

We use MEF to approximate the transfer function $H(s)$ with $2q$ moments in two steps. First, a q -th order exponential function $h_p(t)$ is used to match $H(s)$ in time domain with Padé technique. Clearly, $h_p(t)$ completely matches all $2q$ moments of the transfer function $H(s)$. If all poles of $h_p(t)$ are stable, the process of time domain synthesis of the transfer function is completed.

If there exist unstable poles, the corresponding terms in $h_p(t)$ are discarded. Let there be q_p ($q_p < q$) stable poles, then $h_p(t)$ becomes

$$h_p(t) = \sum_{i=1}^{q_p} k_i e^{p_i t} \quad (4.14)$$

Transfer $h_p(t)$ into frequency domain and expand it around $s = 0$, we have

$$H_p(s) \approx \sum_{i=0}^{2q-1} m_{pi} s^i \quad (4.15)$$

where m_{pi} is the i -th moment of $H_p(s)$, and

$$m_{pi} = -\sum_{j=1}^{q_p} k_j p_j^{-i-1} \quad (4.16)$$

Since unstable poles in $h_p(t)$ are not included, $H_p(s) \neq H(s)$. Let

$$H_e(s) = H(s) - H_p(s) \approx \sum_{i=0}^{2q-1} m_{ei} s^i \quad (4.17)$$

where $m_{ei} = m_i - m_{pi}$. A q_e -th ($q_p + q_e \geq q$) order EDPF function is then used to match $H_e(s)$ [49]. In the time domain, we have:

$$h_e(t) = \sum_{i=0}^{q_e} c_i \left(\frac{t}{d}\right)^i e^{-(t/d)} \quad (4.18)$$

Finally, $h(t) = h_p(t) + h_e(t)$ is the time domain synthesis of transfer function $H(s)$.

As pointed out earlier, MEF preserves the high accuracy of Padé approximation with guaranteed stable solution.

4.4 Accuracy and Stability Issues

Although the moment matching technique is a fast and simple approximation or reduction method and is widely used, there are no efficient means of determining the appropriate order of the reduced model for a desired accuracy or tolerance [16]. The reason for this is that the finite moments, which are the first several coefficients of a Taylor series expansion around some frequency point, do not have enough information about the original function along the entire frequency axis.

Great effort has been made to estimate error in Padé approximations via Kronrod's procedure [42, 11, 12, 8]. However, since there is no explicit expression of error and accurate evaluation of the error, would require computing all moments of the original transfer function, it is impractical to determine the appropriate order of approximation of large networks. Glover's reduction method [33] gives an approximation which minimizes the Hankel-norm. The optimal Hankel-norm approximation and its error bound evaluation are based on first few Hankel singular values. These singular values are square roots of eigenvalues of a matrix which may be obtained by solving linear matrix equations (Lyapunov equations). When the system is large, the matrix is large. The accurate computation of eigenvalues can become computational prohibitive expensive for this application as the size of the matrix reaches a few hundreds, and therefore, the only practical way to obtain eigenvalues is through an approximation. Lanczos method [45, 24] is an efficient method to compute the first few dominant eigenvalues through an iterative procedure for the successive reduction of a square matrix to a sequence of tridiagonal matrices. However, Lanczos method may create a non-positive eigenvalues even though

all eigenvalues of the original matrix are positive [29]. That is, it may create nonstable approximation.

Though it is not easy to find a criteria to automatically determine the approximation order, efforts to increase approximation accuracy have been made [22, 29]. The general way is to increase the approximation order by avoiding numerical ill-condition problem. However, increasing the order usually increases reduction time and simulation time. This becomes evident for large networks with large number of external ports. The difficulty can be dealt with by partitioning. In Chapter 6, an efficient partitioning and reduction method is presented. After partitioning, high accuracy can be achieved while using lower order to approximate each subnetworks.

Though many papers claim to generate stable approximations when all poles of a transfer function are in the left half plane, more constraints have to be imposed on the macromodel when the model is to be integrated into SPICE-like simulators. The macromodels are the approximation of the port behavior of the interconnects. In order to incorporate the macromodels into SPICE-like simulator, they are usually described by an admittance matrix (Y-matrix). The approximation of the Y-matrix may result in that the Y-matrix becomes non-positive-real. A non-positive-real Y-matrix may result in an unstable simulation. A practical method to test positive-real condition is addressed in Chapter 5. A method which guarantees the macromodel to be stable for RC interconnect networks will be presented in Chapter 7.

4.5 Experiment Results

Several benchmark examples are used to verify the efficiency and generality of the macromodel. These testing circuits include various topologies commonly encountered in the delay modeling of VLSI interconnects. The first circuit, shown in Fig. 4.1, is an RC network. Floating capacitors are present to form several loops. It took 0.2 CPU second to analyze the circuit with one external output node and 0.3 CPU second for two external output nodes, both using a second order macromodel. The output responses with a unit step input, together with the results obtained by SPICE3e2 [39], are plotted in Fig. 4.2. While there is little difference between the results based on macromodel and the SPICE model, it took SPICE3e2 3.2 CPU second to compute the response. (CPU times are measured on SUN-SPARC 1+).

We also compared our method with AWE-like simulator which is implemented in

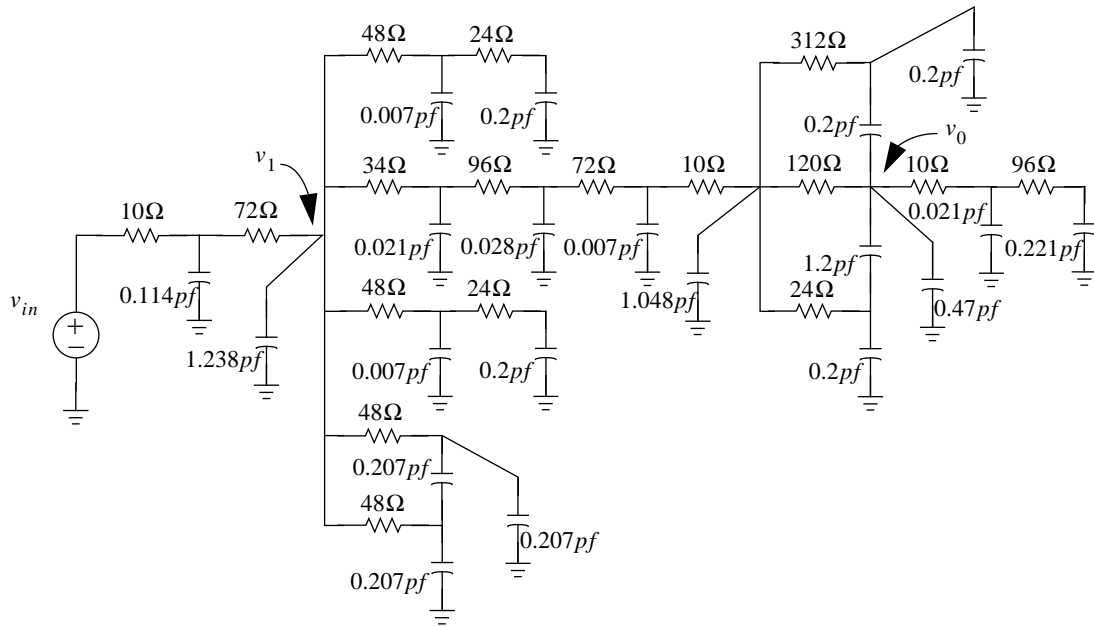


Figure 4.1. An RC circuit with floating capacitance loops.

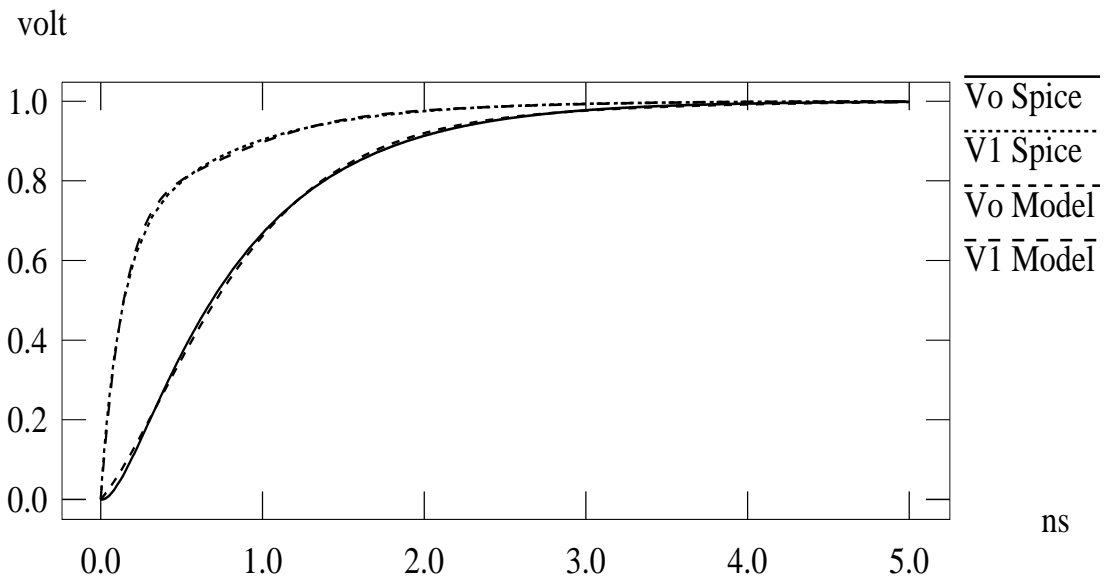


Figure 4.2. Output response of the RC circuit.

industry. Since the simulator cannot handle transmission line networks, we compare our method and AWE-like simulator with lumped circuits. Overall, both simulators have the same accuracy, but the approach presented here is much faster than the SPICE-like simulator. Fig. 4.3 is a simulation result of an industry circuit which consists of 371 RLC elements. While the SPICE-like simulator took 257.18 CPU seconds on an IBM RS6000/370, the AWE-like simulator took 3.76 seconds and our method 2.97 seconds. The result of our method and that of AWE-like simulator are almost identical.

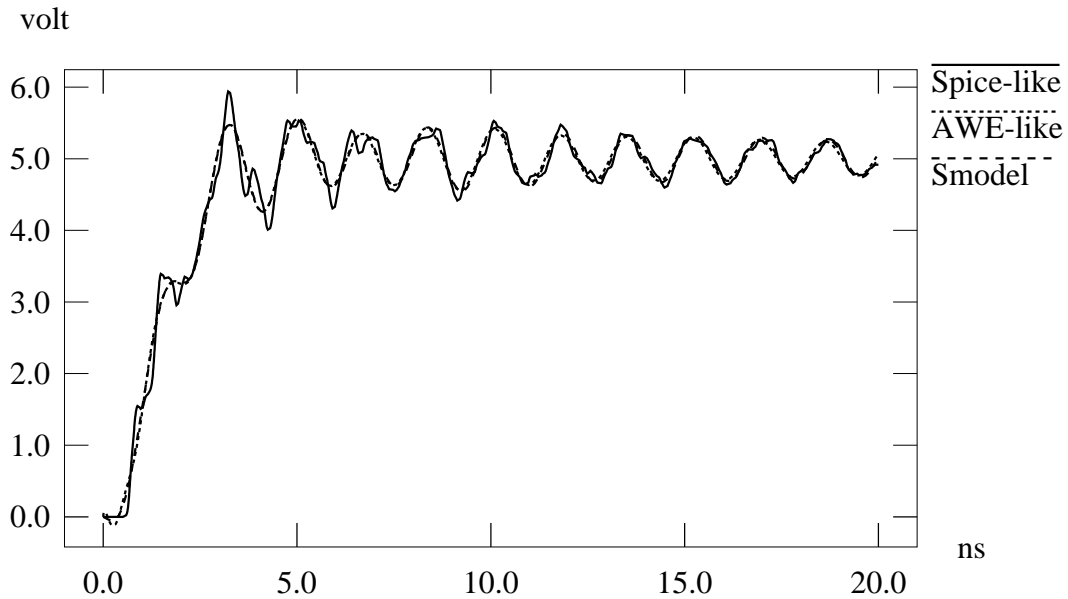


Figure 4.3. The responses of the RLC circuit.

Fig. 4.4 illustrates a clock network. Padé approximation failed to analyze the circuit since the right-hand plane approximate poles are often generated when we use different order approximation to analyze node U1 (for example, two real poles 1.2302×10^8 and 8.5137×10^{10} are generated with fifth order Padé approximation). While SPICE3e2 needed 115.6 CPU seconds to analyze the node U1 response where transmission lines are modeled by lumped sections (0.7mm/section) and 61.3 seconds with distributed model, it took 0.5 CPU second to do so with the twelfth order EDPF model and 0.4CPU second with the eighth order MEF model (the sixth order Padé approximation plus the second order EDPF approximation). The response to 0.4ns rise time ramp input signal is shown in Fig. 4.5.

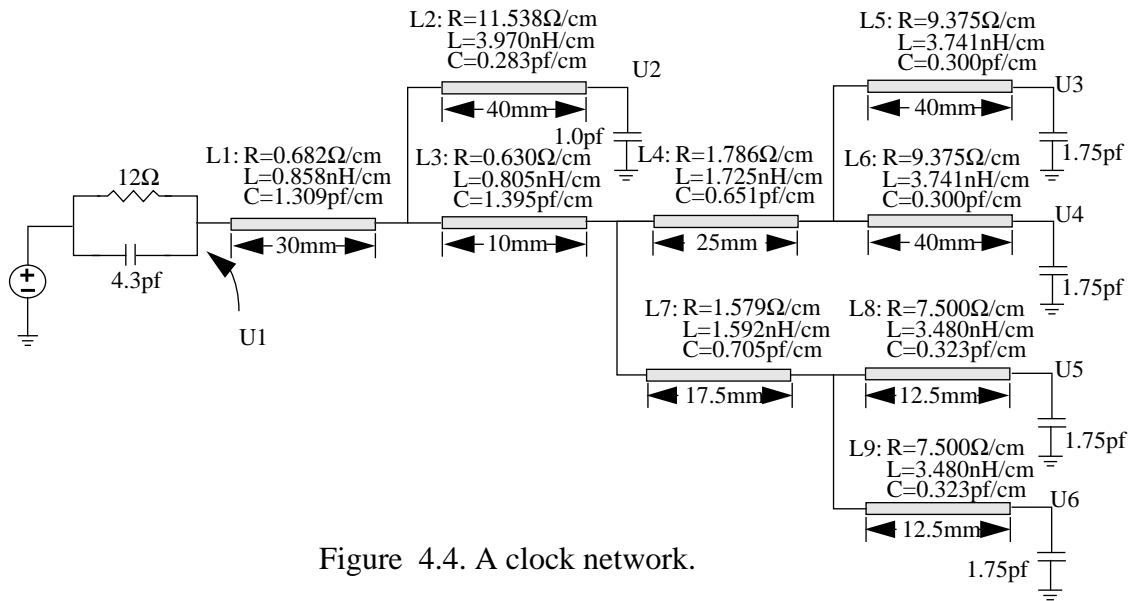


Figure 4.4. A clock network.

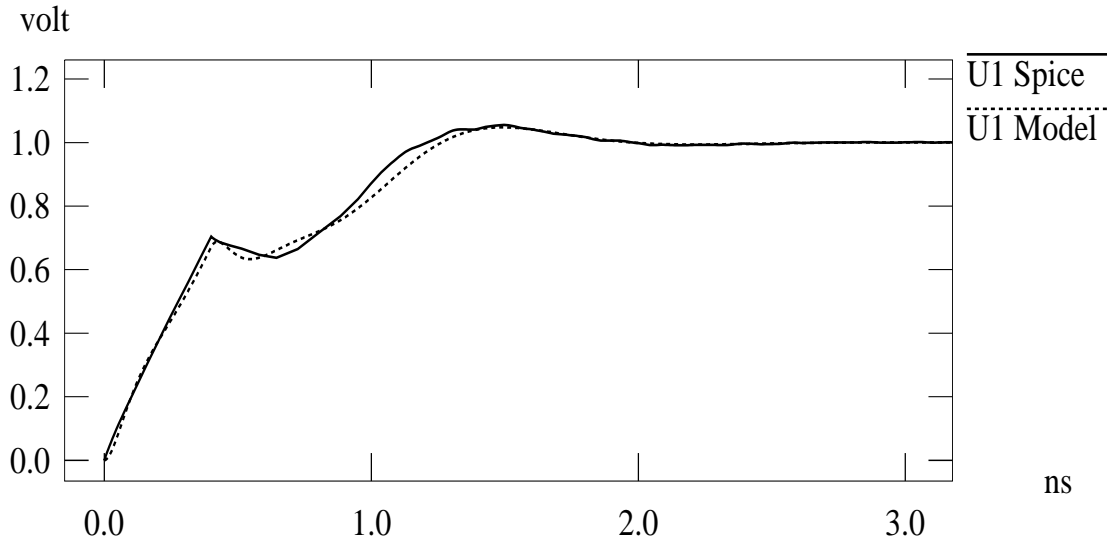


Figure 4.5. Output response at node U1.

The following two testing circuits are used to verify the efficiency and generality of the macromodel with the time-of-flight captured explicitly. The first one is a transmission line circuit which was one of the benchmarks of 1993 IEEE Multi-Chip Module

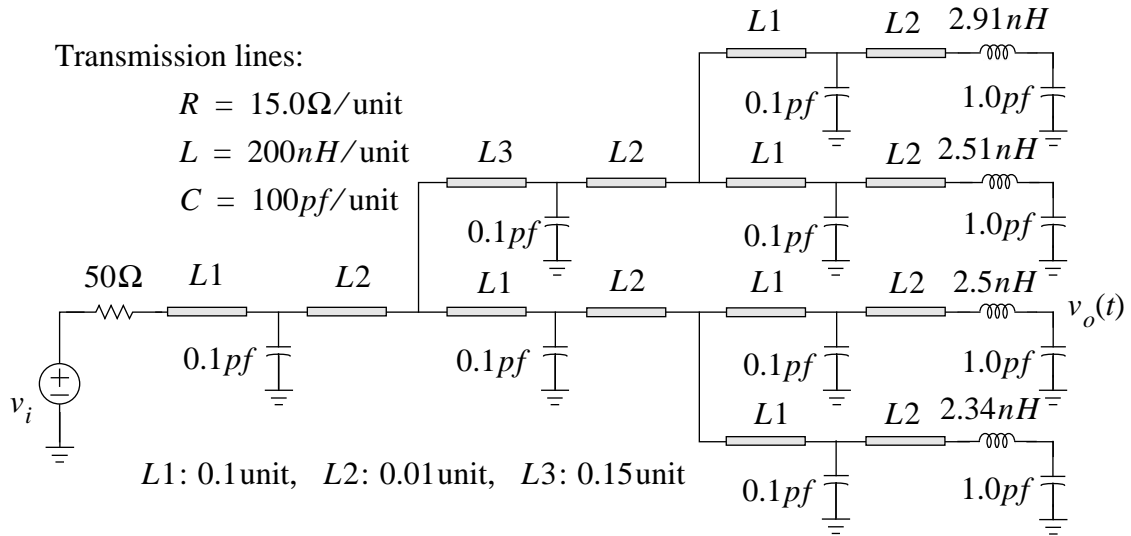


Figure 4.6. MCMC-93 benchmark.

Conference (MCMC-93) (See Fig. 4.6). The circuit was provided by Performance Signal Integrity, Inc. SPICE3e2[39] took more than 5 minutes to compute the output waveform $v_o(t)$ and our method took 1.2 seconds including 0.7 second for plotting the result. Fig. 4.7 is the analysis result of the 5th order approximation without time-of-flight extraction compared to the result of SPICE3e2. The input is a ramp signal with $1.0ns$ rise time. The experiment shows that the time-of-flight delay is difficult to capture with a finite order of approximation. This deficiency affects matching not only the flat section but also the whole curve. Fig. 4.8 is the result of the 5th order approximation with extraction of the time-of-flight ($1.476ns$) compared to the result of SPICE3e2. Using the same order approximation, with no increasing in running time, the simulation has much higher accuracy by capturing the time-of-flight, since it only needs to match the exponentially charging section of the response curves.

Fig. 4.9 is a lossy transmission line circuit. While SPICE3e2[39] took 3.7 seconds to compute the output waveform $v_o(t)$, our method took 1.1 seconds including 0.7 second for plotting the result. Fig. 4.10 is the analysis results of the 4th order approximation without time-of-flight extraction compared with SPICE3e2. The input rise time is $0.8ns$. Fig. 4.11 is the result of the 4th order approximation with extraction of the

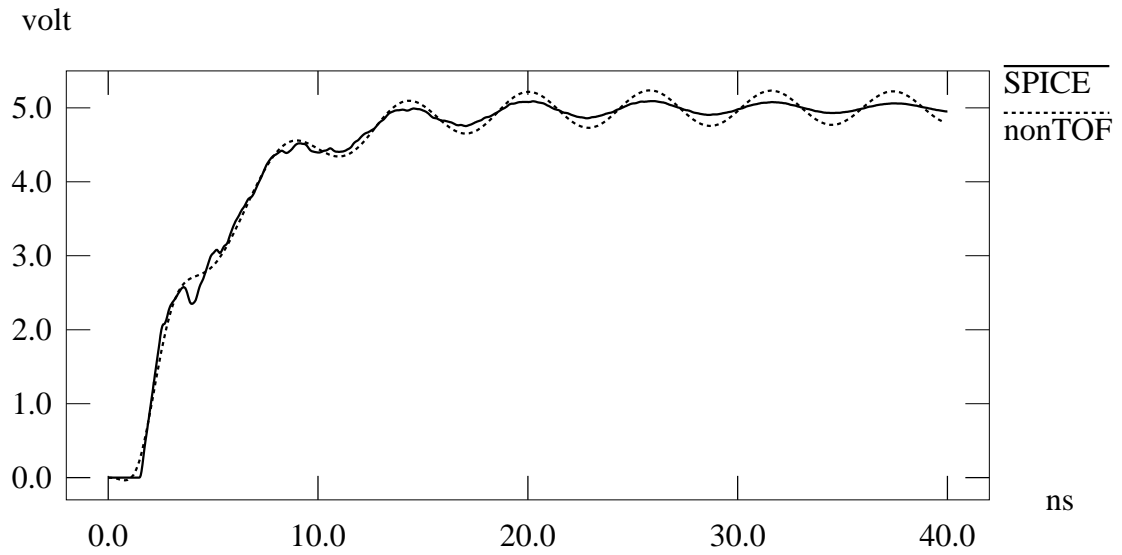


Figure 4.7. The result comparison of the MCMC-93 benchmark by SPICE3e2 and the 5th order approximation without time-of-flight extraction (nonTOF).

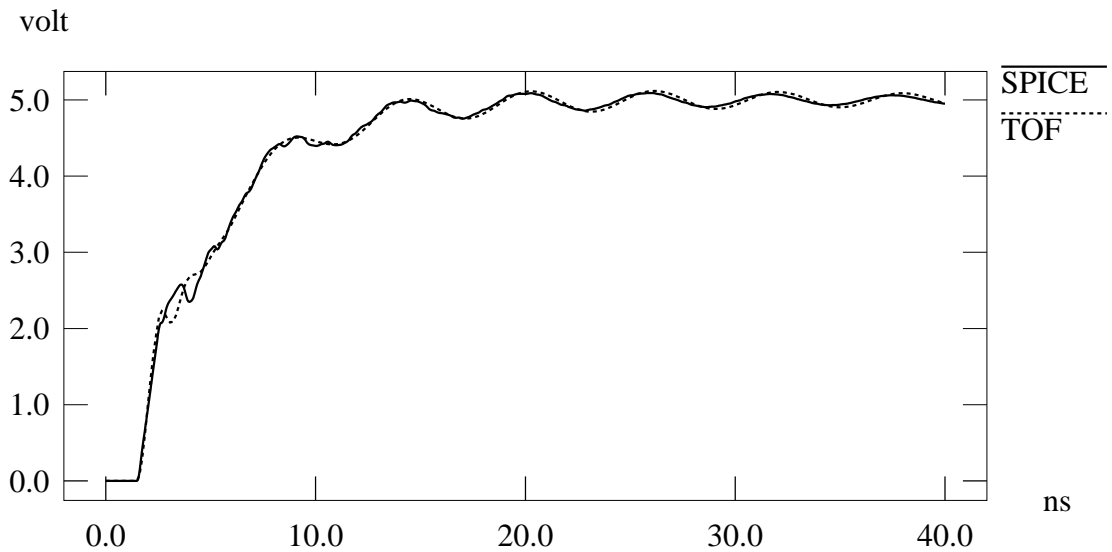


Figure 4.8. The result comparison of the MCMC-93 benchmark by SPICE3e2 and the 5th order approximation with time-of-flight extraction (TOF).

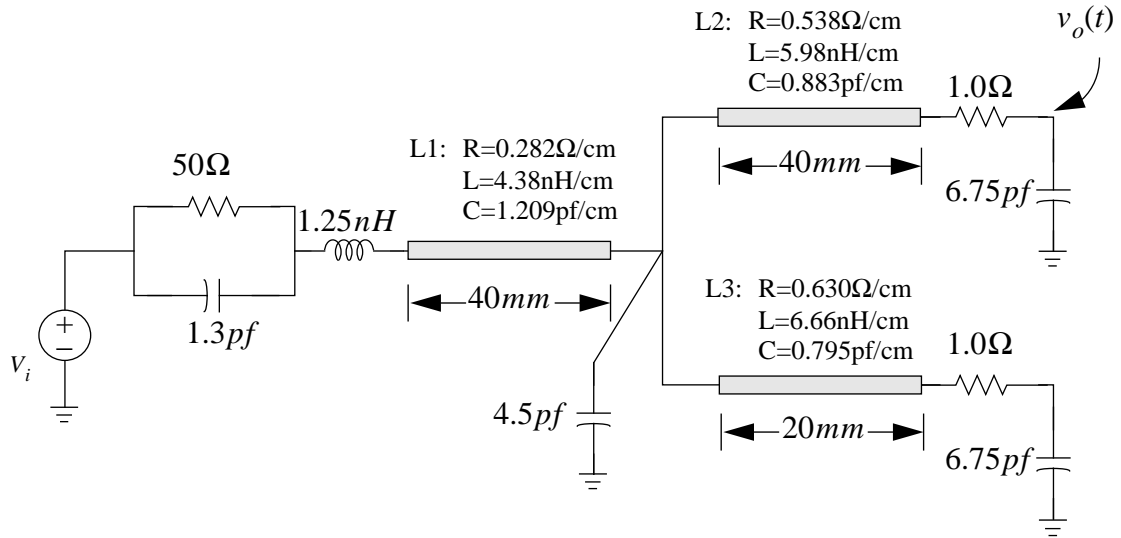


Figure 4.9. A transmission line circuit.

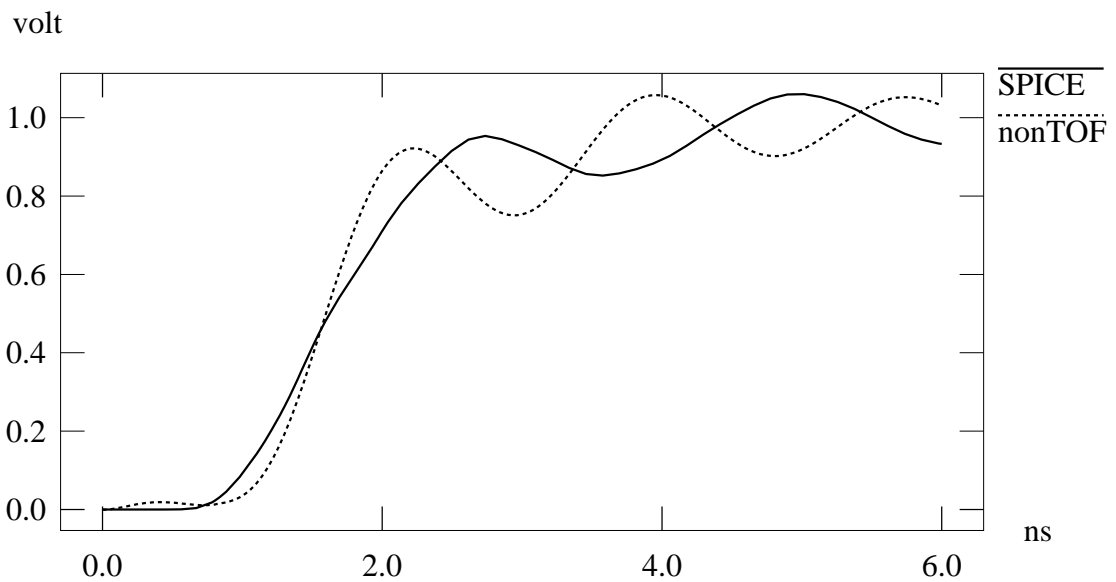


Figure 4.10. The result comparison of the lossy line circuit by SPICE3e2 and the 4th order approximation without time-of-flight extraction (nonTOF).

time-of-flight (0.582ns) compared with SPICE3e2. Again, the accuracy of response curves (not only the flat section, but whole curves) has been greatly improved.

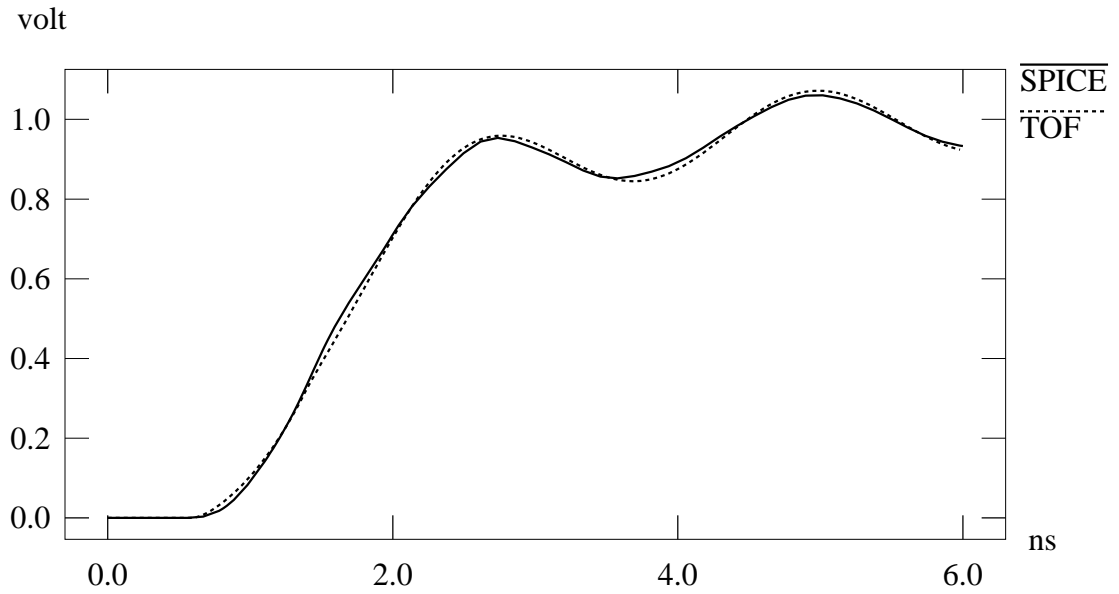


Figure 4.11. The comparison of the lossy line circuit by SPICE3e2 and the 4th order approximation with time-of-flight extraction (TOF).

CHAPTER 5 Norton Equivalent Circuits Based on Recursive Convolution

One of the fundamental difficulties encountered in integrating transmission line simulation in a transient circuit simulator arises because the circuits containing nonlinear devices must be characterized in the time domain while transmission lines with loss, dispersion, and interconnect discontinuities are best modeled in the frequency domain. To cope with this difficulty, direct convolution techniques are used. The system outputs are the convolutions of the inputs with the impulse responses. The fundamental problem lies in how to determine the impulse response of an arbitrary interconnect system. Another drawback of the direct convolution is time consuming. While explicit analytical expression of the impulse responses is impractical, the numerical inverse Fast Fourier Transformation technique suffers from the fact that an excessive number of frequency points are needed to avoid aliasing effects [54].

We have presented a scattering-parameter-based macromodel to analyze interconnect systems. The Exponentially-Decayed Polynomial Function (EDPF) or Mixed-Exponential Function (MEF) are used to get an approximated explicit analytical expression of the transfer function. MEF is the combination of exponential functions and EDPF. Reference [73] gives the derivation of the recursive convolution formula of an exponential with any input function. Actually, an exponential is a special function of the

EDPF when the order is zero. Here, we simplify the recursive formula introduced in [19] for the EDPF and derive Norton equivalent circuits based on the recursive formula. Thus, this model can be easily integrated into a SPICE-like simulator.

When the macromodel is integrated into SPICE-like simulators, the system may still be unstable even though all poles are on the left half plane since the approximation may create a non-bounded-real S-matrix (or non-positive-real Y-matrix). A real rational, bounded-real S-matrix (or positive-real Y-matrix) is the necessary and sufficient condition for the corresponding network to be linear and passive. A non-bounded-real S-matrix (or non-positive-real Y-matrix) may result in unstable simulation. A practical method is presented in this chapter to test bounded real conditions. A method which guarantees the macromodel to be stable for RC interconnect network will be presented in Chapter 7.

5.1 Recursive Convolution Based on the EDPF

It has been shown that convolution integration for an EDPF and a piecewise linear function can be computed by a recursive formula[19]. In the following, we state a simplified recursive convolution formula. For the derivation of the formula, see Appendix B. Consider the following convolution integration

$$y(t) = h(t)*x(t) = \left(\sum_{i=0}^n c_i \left(\frac{t}{d}\right)^i e^{-t/d}\right) * x(t) = \sum_{i=0}^n c_i w_i(t) \quad (5.1)$$

where

$$w_i(t) = \int_0^t \left(\frac{t-\lambda}{d}\right)^i e^{-(t-\lambda)/d} x(\lambda) d\lambda. \quad (5.2)$$

$y(t)$ can be computed by

$$y(t + \Delta t) = \phi(t + \Delta t) + \rho(t + \Delta t)x(t + \Delta t) \quad (5.3)$$

where $\rho(t + \Delta t) = c_0 \Delta t / 2$ and c_0 is the first coefficient of $h(t)$, and $\phi(t + \Delta t)$ can be computed with the following recursive formulas:

$$\begin{aligned} \phi(t + \Delta t) &= \sum_{i=0}^n c_i \zeta_i(t + \Delta t) \\ \zeta_i(t + \Delta t) &= e^{-\Delta t/d} \left(\sum_{k=0}^i \binom{i}{k} \left(\frac{\Delta t}{d}\right)^{i-k} w_k(t) \right) + \frac{\Delta t}{2} \left(\frac{\Delta t}{d}\right)^i x(t) \\ w_k(t + \Delta t) &= \zeta_k(t + \Delta t) + \frac{\Delta t}{2} x(t + \Delta t) \end{aligned} \quad (5.4)$$

5.2 Norton Equivalent Circuits

In order to incorporate the macromodel described with $n \times n$ S-matrix, shown in Fig. 2.2, into SPICE like simulators, we derive a Norton equivalent circuit. By modifying the approach in [81], we insert two series impedances Z_0 and $-Z_0$ at each port to remove the effect of the reference impedance. Thus a virtual node is created between the load and the macromodel as shown in Fig. 5.1 where V'_i is the node voltage in frequency domain at

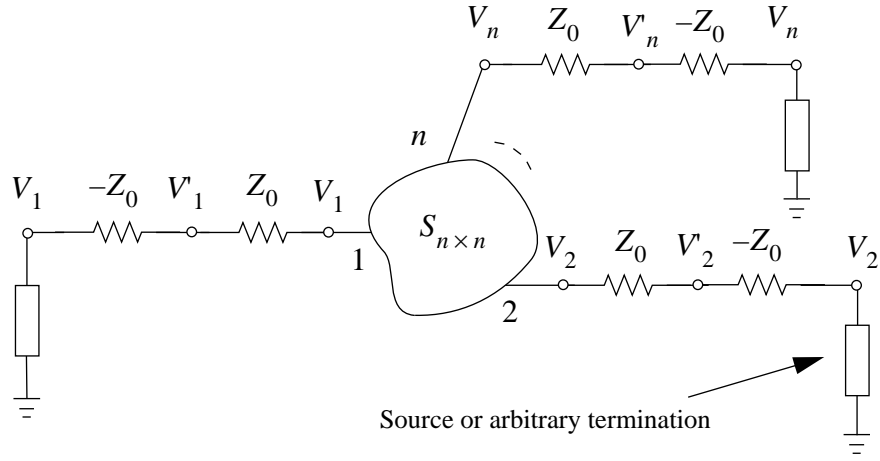


Figure 5.1. Macromodel with virtual nodes.

port i and V'_i is referred to as the virtual voltage at port i . These variables are related by

$$V'_i = V_i + Z_0 I_i \quad i = 1, \dots, N \quad (5.5)$$

where I_i is the current flowing into the macromodel at port i . From the definition of S-parameters and the relation between circuit parameters and wave parameters, we have

$$b_i = \sum_{j=1}^N S_{ij} a_j \quad (5.6)$$

$$V_i = a_i + b_i, \quad Z_0 I_i = a_i - b_i \quad (5.7)$$

where S_{ij} is an element of the S-matrix of the multiport component, and a_i and b_i are the incoming wave and outgoing wave at port i , respectively. From Eqs. (5.5) and (5.7), we get

$$a_i = V'_i / 2 \quad (5.8)$$

Rewriting Eq. (5.5) and combining it with Eqs. (5.6), (5.7) and (5.8), the macromodel system can be described by the following equation:

$$\begin{aligned}
I_i &= Z_0^{-1} V_i - Z_0^{-1} V_i \\
&= Z_0^{-1} V_i - Z_0^{-1} (a_i + b_i) \\
&= Z_0^{-1} V_i - Z_0^{-1} (a_i + \sum_{j=1}^N S_{ij} a_j) \\
&= \frac{1}{2} Z_0^{-1} V_i - \frac{1}{2} Z_0^{-1} \sum_{j=1}^N S_{ij} V_j
\end{aligned} \tag{5.9}$$

Writing this equation in the time domain leads to the convolution equation

$$i_i(t) = \frac{Z_0^{-1}}{2} v'_i(t) - \frac{Z_0^{-1}}{2} \sum_{j=1}^N S_{ij} * v'_j(t) \tag{5.10}$$

Since S-parameters are modeled with EDPF, using the recursive convolution formula Eqs. (5.3) and (5.4), Eq. (5.10) can be written with

$$\begin{aligned}
i_i(t) &= \frac{Z_0^{-1}}{2} v'_i(t) - \frac{Z_0^{-1}}{2} \sum_{j=1}^N (\phi_j(t) + \rho_j(t) v'_j(t)) \\
&= \sum_{j=1}^N g_{ij}(t) v'_i(t) - I_{si}(t)
\end{aligned} \tag{5.11}$$

where

$$I_{si}(t) = \frac{Z_0^{-1}}{2} \sum_{j=1}^N \phi_j(t) \tag{5.12}$$

$$g_{ij}(t) = \begin{cases} Z_0^{-1} (1 - \rho_j(t)) / 2 & i = j \\ -Z_0^{-1} \rho_j(t) / 2 & i \neq j \end{cases} \tag{5.13}$$

A Norton equivalent circuit can be derived from Eq. (5.11) and it can be easily integrated into SPICE-like simulators. The equivalent circuit of a two port macromodel is shown in Fig. 5.2.

5.3 Stability of Macromodel

As pointed out in [16, 1, 22], the Padé approximation suffers from the instability

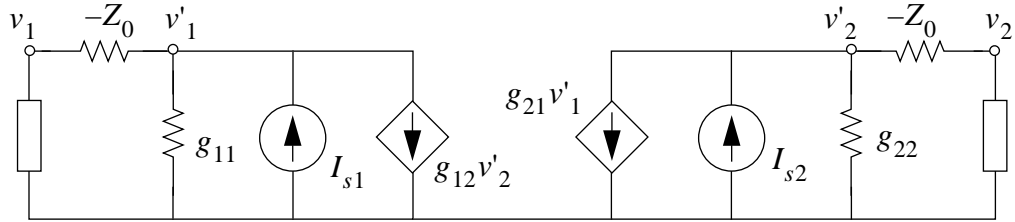


Figure 5.2. Equivalent circuit of a two port macromodel.

problem: unstable poles may be generated for known stable networks. This problem can be overcome by using exponentially-decayed polynomial functions or mixed-exponential functions to approximate the transfer function of the macromodel (See Chapter 4). However, if the macromodel is integrated into SPICE-like simulators, a more severe problem arises: the approximation may create a non-bounded-real S-matrix (or non-positive-real Y-matrix). The real rational, bounded-real S-matrix (or positive-real Y-matrix) is the necessary and sufficient condition for the corresponding network to be linear, solvable, time-invariant, finite, and passive [2, 40, 58]. Non-bounded-real S-matrix (or non-positive-real Y-matrix) may result in an unstable simulation.

An $n \times n$ matrix S is called bounded-real (BR) if it satisfies the condition that [58]

- For $Re(s) > 0$, $E - S'^*(s)S(s)$ is positive semidefinite (denoted as $E - S'^*(s)S(s) \geq 0$).

or the equivalent conditions that

- $S(s)$ is holomorphic in $Re(s) > 0$.
- $E - S'^*(j\omega)S(j\omega)$ is positive semidefinite (denoted as $E - S'^*(j\omega)S(j\omega) \geq 0$) for all (real) ω .

where the superscript asterisk (*) denotes the complex conjugate, and the prime (') denotes the matrix transpose.

The testing of bounded-real conditions contains two aspects. First, we have to check if poles of all S-matrix elements are on the left half plane. Hurwitz's method [40] could be applied. Actually, since we use mixed-exponential functions to approximate the system

matrix, the poles of all S-matrix elements are always in the left half plane. The second step is to test if $E - S'^*(j\omega)S(j\omega)$ is positive semidefinite for all ω . Let $D(j\omega) = E - S'^*(j\omega)S(j\omega)$. The positive semidefinite $D(j\omega)$ means $X'^*D(j\omega)X \geq 0$ for any $n \times 1$ complex vector X . When n is small, $X'^*D(j\omega)X$ can be expanded into a polynomial function. Sturm's method [40] could be used to test if $X'^*D(j\omega)X \geq 0$. However, when $n \geq 3$, there is no efficient method that could be used.

A practicable method is testing $D(j\omega) \geq 0$ at sample frequency points. According to [44], $D(j\omega) \geq 0$ if and only if all eigenvalues of $D(j\omega)$ are in the right half plane.

Complex computation of eigenvalues of $D(j\omega)$ can be avoided. For a reciprocal interconnect network, the S-matrix is symmetrical. And

$$D'^*(j\omega) = E'^* - (S'^*(j\omega)S(j\omega))'^* = E - S'^*(j\omega)S(j\omega) = D(j\omega) \quad (5.14)$$

that is, $D(j\omega)$ is Hermitian. For the Hermitian matrix $D(j\omega)$, the eigenvalues are all real and equal to those of the following real symmetrical matrix [62]

$$\begin{bmatrix} Re(D(j\omega)) & -Im(D(j\omega)) \\ Im(D(j\omega)) & Re(D(j\omega)) \end{bmatrix} \quad (5.15)$$

5.4 Experiment Results

The model has been built and added to the Model Independent SIMulator (MISIM) [82] which is based on the Modified Nodal Analysis (MNA). Fig. 5.3 is a lossy transmission line circuit driven by a CMOS inverter. Fig. 5.4 and Fig. 5.5 are the response

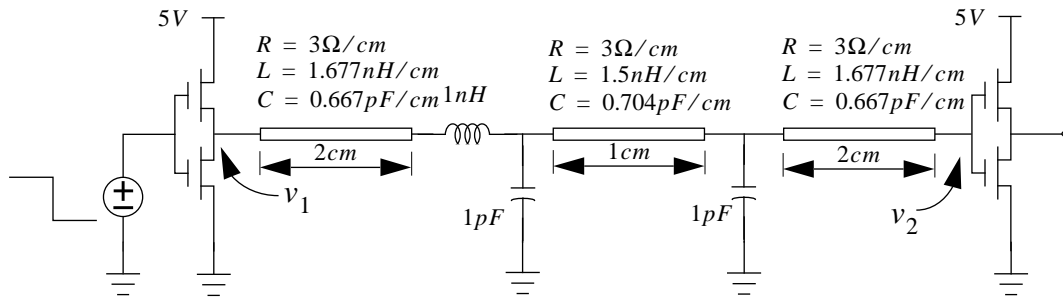


Figure 5.3. A lossy transmission line circuit with nonlinear loads.

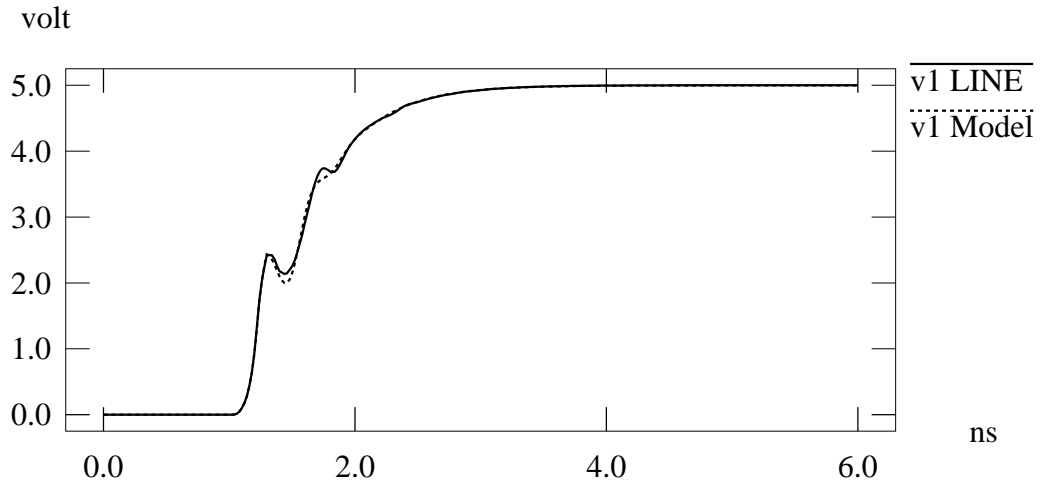


Figure 5.4. Near end response of the lossy transmission line circuit.

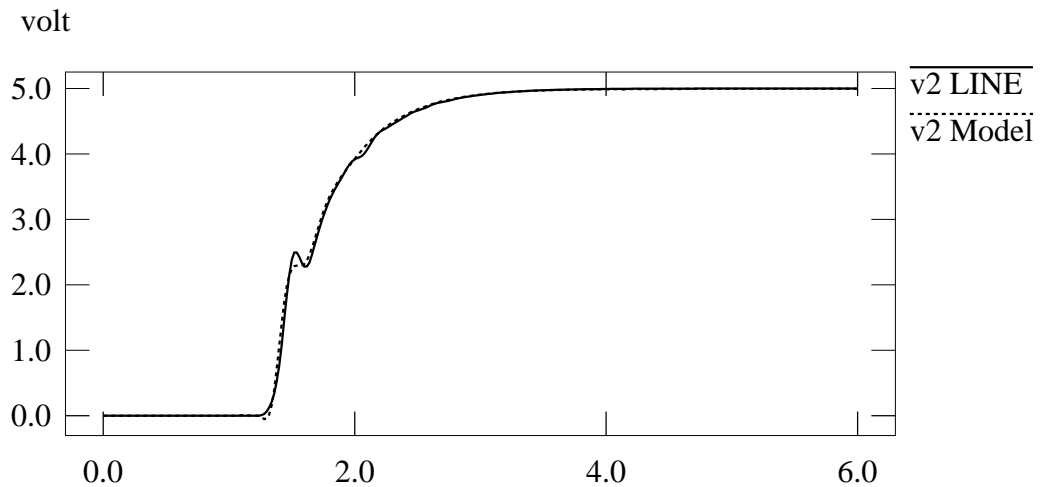


Figure 5.5. Far end response of the lossy transmission line circuit.

of the near end and the far end respectively. Comparing the direct convolution method (solid line)[32], our macromodel (dashed line) with recursive convolution has almost identical results with order of magnitudes less computing time.

The another example is a grid-type clock network (See Fig. 5.6) which is distributed around the periphery of a $10\text{mm} \times 10\text{mm}$ chip. The vertical runs are on metal layer 1 ($R = 0.8\Omega/\text{mm}$, $L = 0.66\text{nH}/\text{mm}$ and $C = 0.868\text{pF}/\text{mm}$) and the horizontal runs

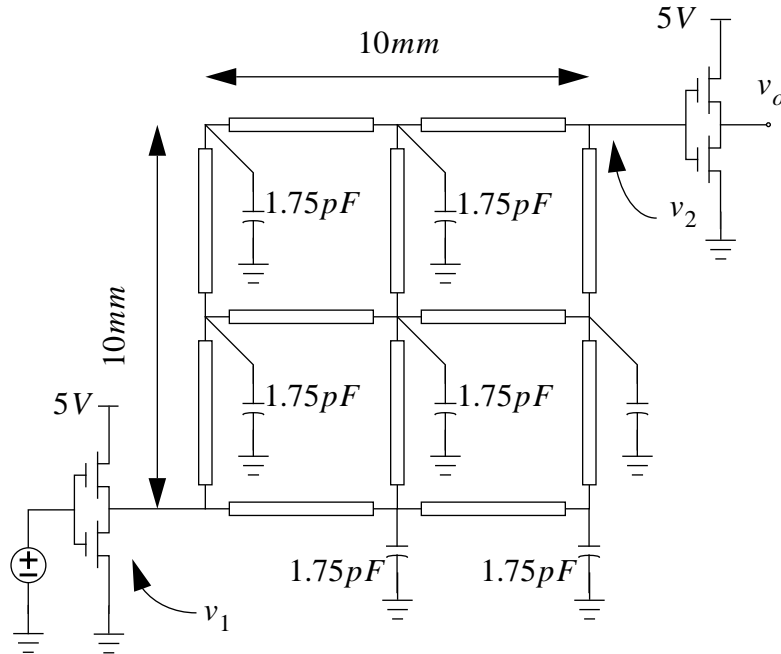


Figure 5.6. Grid-type clock network.

are on metal layer 2 ($R = 1.0\Omega/mm$, $L = 0.826nH/mm$ and $C = 0.694pF/mm$). The network is represented by distributed lossy transmission lines driven by a CMOS inverter. Fig. 5.7 is the curve of the driver output v_1 and Fig. 5.8 is the curves of v_o . There is little difference between the results based on our macromodel and the SPICE model. While it took more than 500 CPU seconds on SUN SPARC 1+ for SPICE3e2 to get the answer using the direct convolution [70], our program took 56.3 seconds to analyze the circuit using 12th order MEF approximation.

We also integrated the macromodel into Intel internal simulator TIM. The same circuit topology of Figure 5.6 is also tested in TIM with different order of approximation. The vertical runs are on metal layer 1 ($R = 1.6\Omega/mm$, $L = 0.66nH/mm$ and $C = 0.868pF/mm$) and the horizontal runs are on metal layer 2 ($R = 2.0\Omega/mm$, $L = 0.826nH/mm$ and $C = 0.694pF/mm$). The ratios of channel width and length of the driver are 400 for pMOS and 200 for nMOS. The ratios of channel width and length of the receiver are 40 for pMOS and 20 for nMOS. The output responses of v_2 (See Figure 5.6) with different order approximation are shown in Figure 5.9.

Several more circuits driven by nonlinear drivers are tested in TIM. The results are

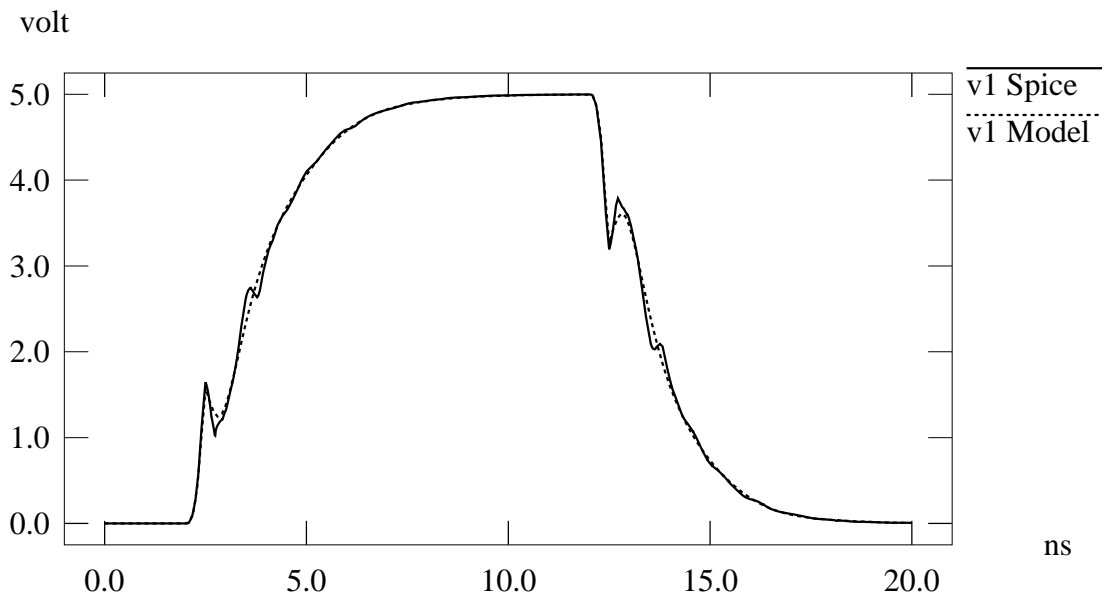


Figure 5.7. Near end response of the clock network.

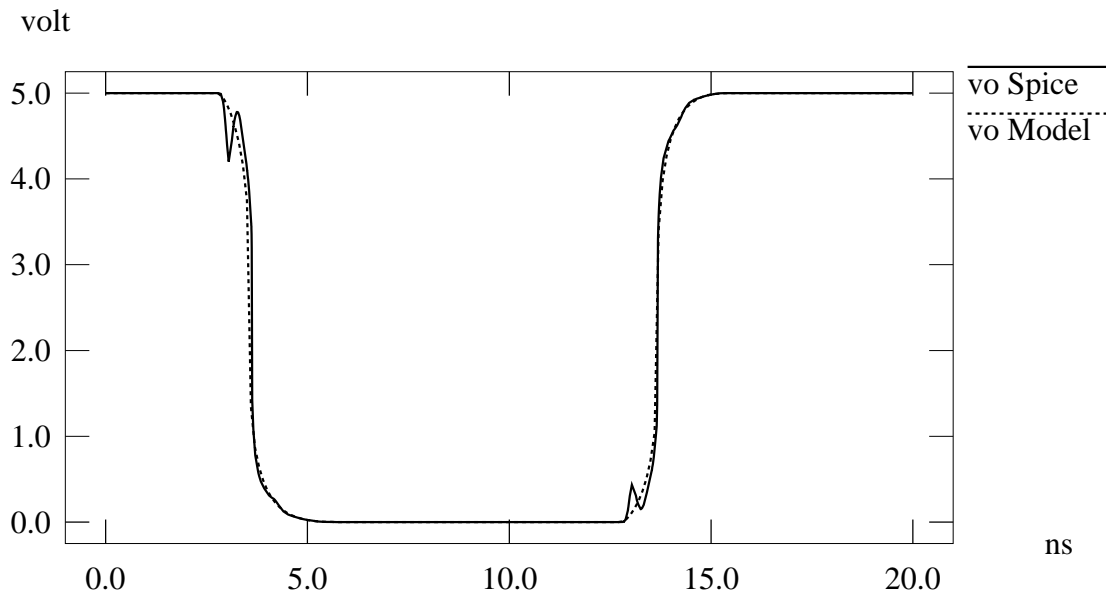


Figure 5.8. Far end response of the clock network.

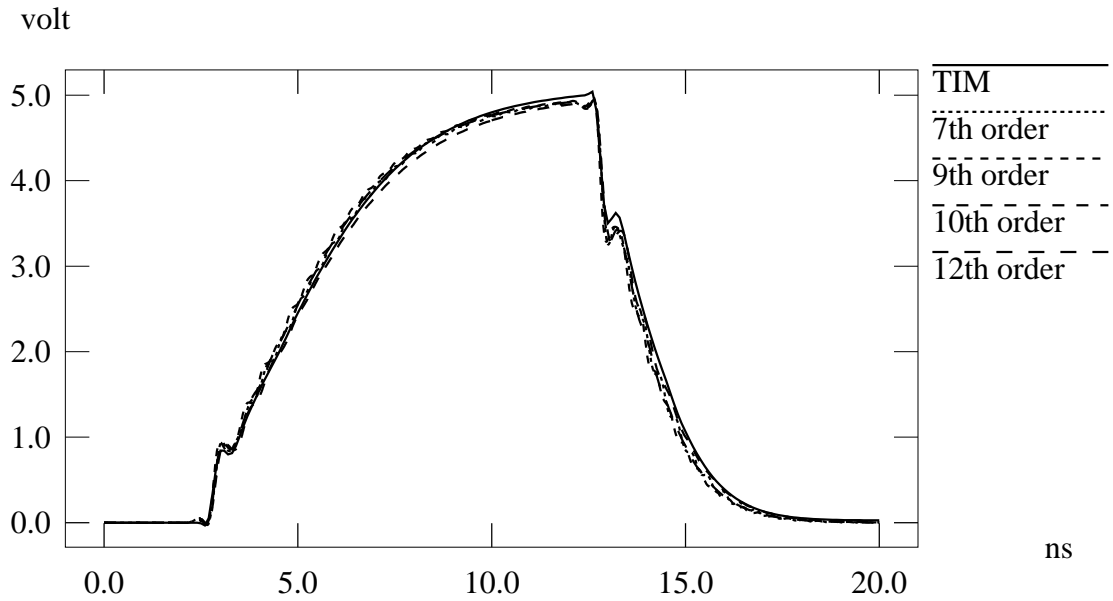


Figure 5.9. Comparison of different order approximation.

shown in Table 5.1. Note that the time for macromodel to do DC analysis includes the time of model reduction. From the table, we can see that the efficiency is more evident for the larger circuits.

circuits	# of elements	time for DC analysis		time for transient analysis	
		TIM	S-model*	TIM	S-model
Circuit 1	82	0.01	0.18	0.26	0.06
Circuit 2	180	0.47	0.46	2.44	0.80
Circuit 3	373	0.03	0.78	23.64	0.78
Circuit 4	11216	28.09	1.07	135.72	19.82

Table 5.1: Comparison of the macromodel and TIM.

*. The number includes the time for model reduction.

CHAPTER 6 Partitioning of Interconnect Networks

When fast timing analysis is desired for today's complex interconnect structures, moment matching techniques are widely used to approximate a higher order linear network using the waveforms generated by its lower order moments [49, 54, 60]. These approaches can also be used to improve the efficiency of existing time-domain circuit simulators by incorporating macromodels of the interconnect networks together with recursive convolution [49, 54, 64], or generating an equivalent circuit based on a numerical integration method [40].

The macromodels generated by moment matching technique are approximations of the port behavior of the interconnects. In order to incorporate the macromodels into SPICE-like simulators, they are usually described by an admittance matrix (Y-matrix) [64, 40]. The Y-matrices are generally full matrices. When the number of external ports of an interconnect network is large, the number of entries of the matrix may be larger than the number of circuit elements of the original interconnect network. For example, a clock network with 500 fanouts will create more than 250,000 entries of the corresponding Y-matrix. This huge full matrix may even increase the computational burden of traditional SPICE simulation.

On the other hand, trying to model the huge full matrix by lower order approximation with moment matching techniques is impractical. Severe numerical problems may result in extremely ill-conditioned matrices for computing high order

moments. The complex frequency hopping method [22] and the PVL (Padé approximation via the Lanczos process) [29] are efforts to compute high order moments while avoiding the ill-condition problem. However, increasing the approximation order will increase the modeling and simulation time. Especially, when a large circuit with large number of ports is terminated with nonlinear drivers and loads, there is no efficient method to obtain the reduced-order approximate models of linear networks.

The difficulty of reducing a large linear network with large number of ports can be dealt with by partitioning the network into smaller subnetworks. Instead of modeling a single large interconnect networks with a higher order approximation, we develop a linear time algorithm to partition the network and use lower-order models to approximate multiple small subnetworks. High accuracy can be achieved using lower-order models approximating the all subnetworks. Thus partitioning reduces the simulation time significantly. Furthermore, since partitioning optimally chooses the reduction order and does not merge large components, the reduction time actually decreases as well.

6.1 Circuit Partitioning

The description of an N -port component is generally a full S-matrix or Y-matrix. When N is very large, the synthesized equivalent circuit could have a very large number of elements. In order to minimize the number of elements of the synthesized circuit, we propose to partition a multiport component into several subcomponents with a small number of ports (See Fig. 6.1). Since the number of elements of the reduced circuit is

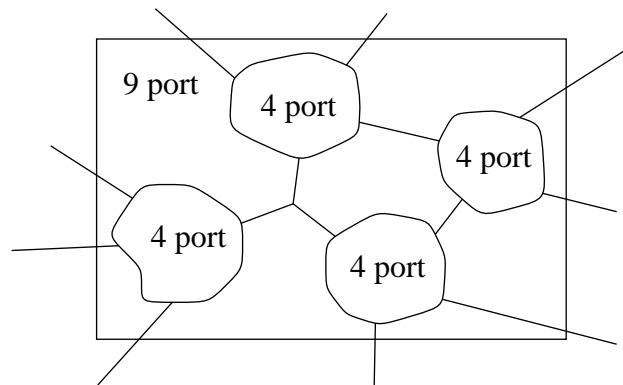


Figure 6.1. Several smaller components are more efficient than one large component to model complex interconnects.

proportional to the number of the entries of S-matrix, the objective of the partitioning problem is to minimize the total number of entries of the S-matrix. More precisely,

$$\text{minimize} \quad \sum_{i=1}^{\chi} N_i^2 \quad (6.1)$$

where χ is the number of subcomponents and N_i the number of ports of the component i . We propose a linear-time heuristic algorithm to partition the interconnect network into several multiport components.

Consider the circuit graph of a network $G(V, E)$, where vertices V consists of circuit elements shown in Fig. 2.1 and edges E represent the adjacency between elements in the circuit. Define the weight $W(e)$ of an edge $e \in E$ as the number of ports of the resultant component after contracting e with Adjoined Merging (Fig. 2.3) or Self Merging (Fig. 2.4). We bound the weight of “contractible” edges by W_{max} . When an edge has the weight more than W_{max} , it will not be contracted during network reduction, and when weights of all edges are greater than W_{max} , the reduction process terminates. Here W_{max} is chosen in such a way that the number of matrix entries does not increase too much when contracting the edges. Let N_x and N_y be the port numbers of the two different components connected by an edge e . According to the definition, $W(e) = N_x + N_y - 2$. W_{max} is chosen such that for any e , the following inequality holds,

$$N_x^2 + N_y^2 > \beta (N_x + N_y - 2)^2 \quad (6.2)$$

where $N_x^2 + N_y^2$ is the number of entries of S-matrices before contracting e , $(N_x + N_y - 2)^2$ is that after contracting e , and β the contracting factor. Based on experience, the typical β is between 0.75 and 1.0, or the corresponding W_{max} is between five and eight. The following is the pseudo code of the network reduction algorithm:

Algorithm Network-Reduction

 Compute weights of all edges

$e_{min} \leftarrow$ the edge with minimal weight

 while ($W(e_{min}) < W_{max}$)

 Contract e_{min}

 Update weights of all edges incident to e_{min}

$e_{min} \leftarrow$ the edge with minimal weight

end-while

Now we have two problems: one is how to select efficiently an edge with minimum weight, and the other is how to update the edge weights after contracting an edge. Those operations can be done in constant time with a bucket list structure shown in Fig. 6.2. This

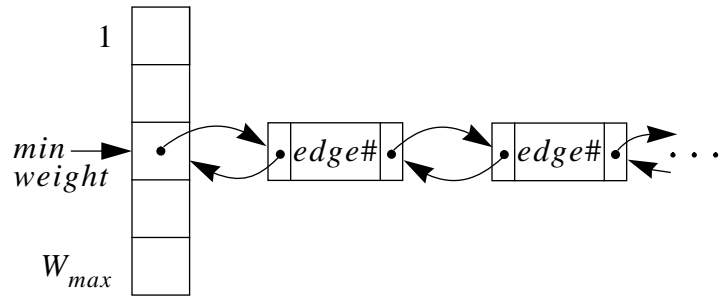


Figure 6.2. Bucket list structure.

structure includes a bucket array, whose i th entry contains a doubly-linked list of edges with weights equal to i . For the bucket array, a *minweight* index is used to keep track of the bucket containing edges with the minimum weight. Since only neighboring edges change their weights, these weights can be updated in constant time [31].

6.2 Efficiency and Accuracy of Reduced Networks

The common method to increase accuracy of modeling a large interconnect network with moment matching techniques is to increase approximation order [22, 29, 41]. This will increase the modeling and simulation time. The problem becomes evident when the number of port is very large. As stated in [41],

“The generation of the y-parameters for a specified multiport network is achieved by exciting the network with an impulse voltage at one of the ports while shorting all other ports. Measuring the currents at each of the ports then yields one column of the Y-matrix. This procedure is repeated at each of the network ports to obtain the complete description”.

Only recently, an improved PVL method [30] were proposed to compute the reduced-order approximate models of linear networks with multiple inputs and outputs.

However, the huge full Y-matrix with high order of approximation will still be a heavy burden on simulation.

Instead of modeling a single large interconnect networks with higher order approximation, we partition the network and use lower-order models to approximate multiple small subnetworks. High accuracy can be achieved using lower-order models approximating the all subnetworks. The accuracy can be controlled by adjusting the number of subnetworks, or equivalently by adjusting the number of original circuit elements in each subnetwork. The partitioning reduces the simulation time significantly. Furthermore, since partitioning optimally chooses the reduction order and does not merge large components, the reduction time actually decreases.

In addition, partitioning greatly reduces the number of elements in the reduced circuits. When using a Y-matrix to describe the port behavior of a network, the number of y-elements is $O(N^2)$, where N is the number of external ports. However, as we will show in experimental results in the Chapter 7, for typical clock networks, the number of circuit elements in reduced circuits with partitioning is $O(N)$.

CHAPTER 7 Circuit Synthesis of RC Networks

To extract accurately the interconnect networks on chips, the number of parasitic resistors and capacitors can overwhelm any circuit simulator. A new transient analysis strategy for complex RC interconnect networks is proposed in this Chapter. With the circuit partitioning we proposed in Chapter 6, a large RC interconnect network is reduced into subnetworks which are approximated with lower-order equivalent RC circuits. The number of RC elements can be reduced between 50% and 90% (80% is typical). The reduced circuits are guaranteed to be stable. The reduced-order approximate models of linear networks with multiple inputs and outputs can be obtained in one pass of reduction. The reduced circuits can be simulated using existing circuit simulators without modification. The experiment results show that the number of circuit elements in a reduced network is $O(N)$ for typical clock networks, where N is the number of external ports. The simulation time can be reduced by two to three orders of magnitude while error is less than 5% compared to the original circuits.

7.1 Circuit Synthesis of RC Interconnect Networks

Given the scattering matrix S of a multiport component, the admittance matrix is

$$Y = Z_0^{-1} (E + S)^{-1} (E - S) \quad (7.1)$$

where Z_0 is the reference impedance, and E identity matrix. The synthesis of a general Y-

matrix without using a transformer is still an open-problem [2, 40, 58]. Here we propose a new method to avoid using transformers. The admittance to ground of the i th port is given by the sum of the i th row (or column) of its Y-matrix. The admittance connecting port- i and port- j is y_{ij} . Thus, the circuit synthesis problem amounts to synthesizing admittances between a port and the ground and between pairs of ports with lumped R and C elements. Unlike the method proposed in [66] where the connection between any two ports is synthesized with a fixed circuit configuration and the values of circuit elements are determined based on the admittance at two frequency points, we use moment matching technique to synthesize the admittance matrices of an RC network. With Taylor series expansion, y_{ij} can be described as

$$y_{ij} = m_0^{(ij)} + m_1^{(ij)} s + \dots \quad (7.2)$$

A method of synthesizing a lower-order circuit of a transfer function based on the moment matching technique was proposed by Roberge [68]. It can not guarantee to have a stable approximation. The lower-order circuit, which is synthesized with operational amplifiers, will take more time to simulate than the original circuit.

We use the T-model shown in Fig. 7.1(a) to synthesize the circuits between pairs of ports, by matching the moments of y_{ij} . The T-model together with the capacitors at two ports, to be synthesized with y_{ii} , forms a 2Π model which provides an accurate description of connection between pairs of ports. The admittance matrix of the T-model is

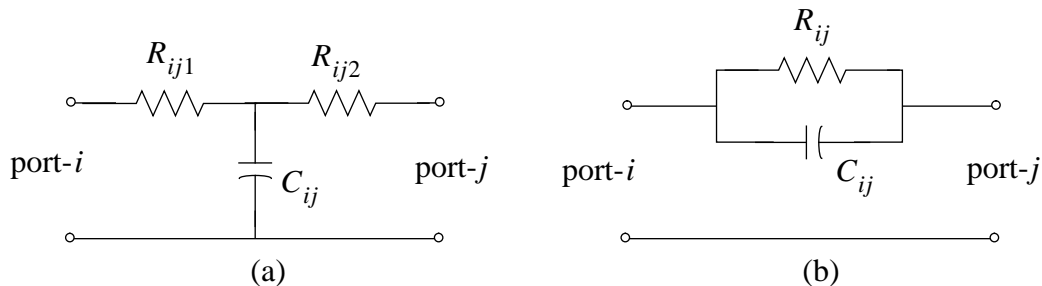


Figure 7.1. Circuit models between pairs of ports.

$$\begin{aligned}
\tilde{y}_{ii} &= \frac{1}{R_{ij1}} - \frac{R_{ij2}}{R_{ij1}(R_{ij1} + R_{ij2} + sC_{ij}R_{ij1}R_{ij2})} \\
\tilde{y}_{jj} &= \frac{1}{R_{ij2}} - \frac{R_{ij1}}{R_{ij2}(R_{ij1} + R_{ij2} + sC_{ij}R_{ij1}R_{ij2})} \\
\tilde{y}_{ij} &= \tilde{y}_{ji} = \frac{-1}{R_{ij1} + R_{ij2} + sC_{ij}R_{ij1}R_{ij2}}
\end{aligned} \tag{7.3}$$

By expanding Eq. (7.3) into Taylor series, we have

$$\begin{aligned}
\tilde{m}_0^{(ii)} &= \tilde{m}_0^{(jj)} = -\tilde{m}_0^{(ij)} = \frac{1}{R_{ij1} + R_{ij2}} \\
\tilde{m}_1^{(ii)} &= \frac{C_{ij}R_{ij2}^2}{(R_{ij1} + R_{ij2})^2} & \tilde{m}_1^{(jj)} &= \frac{C_{ij}R_{ij1}^2}{(R_{ij1} + R_{ij2})^2} \\
\tilde{m}_1^{(ij)} &= \frac{C_{ij}R_{ij1}R_{ij2}}{(R_{ij1} + R_{ij2})^2}
\end{aligned} \tag{7.4}$$

In order to determine the values of R_{ij1} , R_{ij2} and C_{ij} , let

$$\tilde{m}_0^{(ij)} = m_0^{(ij)} \quad \tilde{m}_1^{(ij)} = m_1^{(ij)} \quad \frac{\tilde{m}_1^{(ii)}}{\tilde{m}_1^{(jj)}} = \frac{m_1^{(ii)}}{m_1^{(jj)}} \tag{7.5}$$

The reason why we match $m_1^{(ii)}/m_1^{(jj)}$, and not match $m_1^{(ii)}$ and $m_1^{(jj)}$ directly, is that $m_1^{(ii)}$ and $m_1^{(jj)}$ are the second moments of y_{ii} and y_{jj} which are total contribution of the circuit, not just a branch between port- i and port- j . Solving Eqs. (7.3), (7.4) and (7.5), we have

$$\begin{aligned}
R_{ij1} &= \frac{-\sqrt{m_1^{(jj)}}}{m_0^{(ij)} (\sqrt{m_1^{(ii)}} + \sqrt{m_1^{(jj)}})} \\
R_{ij2} &= \frac{-\sqrt{m_1^{(ii)}}}{m_0^{(ij)} (\sqrt{m_1^{(ii)}} + \sqrt{m_1^{(jj)}})} \\
C_{ij} &= \frac{(\sqrt{m_1^{(ii)}} + \sqrt{m_1^{(jj)}})^2 m_1^{(ij)}}{\sqrt{m_1^{(ii)}} m_1^{(jj)}}
\end{aligned} \tag{7.6}$$

The above formulas could be used to synthesize all off-diagonal elements of Y -matrix with $m_1^{(ij)} \geq 0$. However, if there are floating capacitors, $m_1^{(ij)}$ may be negative. In this case, we can also use a floating capacitor to model the y_{ij} between port- i and port- j (see Fig. 7.1(b)). The admittance matrix of the model is

$$\begin{aligned}\tilde{y}_{ii} &= \tilde{y}_{jj} = -\tilde{y}_{ij} \\ &= \frac{1}{R_{ij}} + sC_{ij}\end{aligned}\quad (7.7)$$

Thus, we have

$$R_{ij} = \frac{-1}{m_0^{(ij)}} \quad C_{ij} = -m_1^{(ij)} \quad (7.8)$$

For the diagonal elements y_{ii} , we need to synthesize y_{i0} , where

$$\begin{aligned}y_{i0} &= y_{ii} - \sum \tilde{y}_{ii} \\ &= m_0^{(i0)} + m_1^{(i0)}s + \dots\end{aligned}\quad (7.9)$$

The y_{i0} is modelled with a parallel RC elements of Fig. 7.2. If there is no DC path from port- i to ground, $m_{i00} = 0$. The y_{i0} is modelled with a single capacitor. In general, the parameters of the model are

$$R_{ii} = \frac{1}{m_0^{(i0)}} \quad C_{ii} = m_1^{(i0)} \quad (7.10)$$

In case the capacitor C_{ii} computed in Eq. (7.10) is negative, we can set $C_{ii} = 0$ and scale up all C_{ij} to keep total capacitance unchanged. Let C'_{ij} be the new capacitance

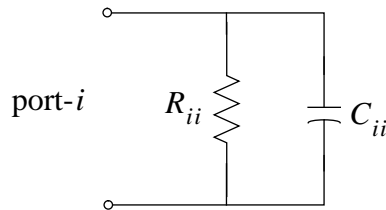


Figure 7.2. RC parallel model of a port.

corresponding to C_{ij} , then we have

$$\begin{aligned} C'_{ii} &= 0 \\ \sum_{i \neq j} C'_{ij} &= C_{ii} + \sum_{i \neq j} C_{ij} \end{aligned} \quad (7.11)$$

By setting $C'_{ij} = \alpha C_{ij}$, we have

$$\alpha = \frac{C_{ii} + \sum_{i \neq k} C_{ik}}{\sum_{i \neq k} C_{ik}} \quad (7.12)$$

and

$$C'_{ij} = \frac{C_{ii} + \sum_{i \neq k} C_{ik}}{\sum_{i \neq k} C_{ik}} C_{ij} \quad (7.13)$$

7.2 Stability of Synthesized Circuits

The sufficient condition for the reduced circuits to be stable is that all synthesized circuit parameter are non-negative [40, 58, 66]. In this section, we prove indeed that the reduced circuits are stable.

Lemma 7.1: The matrix obtained by replacing each element of the Y-matrix by its first moment is diagonal dominant and all off-diagonal elements are nonpositive. That is,

$$\begin{aligned} m_{ii0} &\geq \sum_{i \neq j} |m_{ij0}| \\ m_{ij0} &\leq 0 \quad i \neq j \end{aligned} \quad (7.14)$$

Proof: Actually, the matrix consisting of first moments of the Y-matrix of an N-port interconnect network is a symmetric admittance matrix of a resistive and grounded network. According to [40], the necessary and sufficient condition for the realization of such a matrix is that the matrix is diagonal dominant, with all off-diagonal elements nonpositive. \square

Lemma 7.2: For RC circuits, the second moments m_{ii1} of the diagonal elements y_{ii} ($i = 1, \dots, N$) of the Y-matrix are positive.

Proof: y_{ii} is equivalently the driving-point admittance of an one-port RC circuit. It can be described as in [40]

$$y_{ii} = h + k_{\infty}s + \sum_q \frac{k_q}{s + p_q} \quad (7.15)$$

Thus,

$$m_{ii1} = k_{\infty} - \sum_q \frac{k_q}{p_q} \quad (7.16)$$

Since $k_{\infty} \geq 0$ and all $k_q < 0$ for a positive-real one-port RC circuits (See Section 6.3 of [40]), the lemma follows. \square

Theorem 7.1: The resistances R_{ij1} and R_{ij2} computed by Eq. (7.6), R_{ij} by Eq. (7.8) and R_{ii} by Eq. (7.10) are non-negative.

Proof: According to Lemma 7.1, $m_0^{(ij)} \leq 0$ for $i \neq j$ and

$$\begin{aligned} m_0^{(i0)} &= m_0^{(ii)} - \sum \tilde{m}_0^{(ii)} \\ &= m_0^{(ii)} + \sum \tilde{m}_0^{(ij)} \\ &= m_0^{(ii)} + \sum m_0^{(ij)} \geq 0 \end{aligned} \quad (7.17)$$

According to Lemma 7.2, $m_1^{(ii)} > 0$. Thus, R_{ij1} , R_{ij2} , R_{ij} and R_{ii} are all non-negative. \square

Theorem 7.2: The capacitances C_{ij} computed by Eq. (7.6) or Eq. (7.8) and C_{ii} by Eq. (7.10) or Eq. (7.13) are non-negative.

Proof: When $m_{ij1} \geq 0$, and the T-model of Fig. 7.1(a) is used, the capacitance C_{ij} computed by Eq. (7.6) is nonpositive. In case $m_{ij1} < 0$, the floating RC model of Fig. 7.1(b) is used. The capacitance C_{ij} computed with Eq. (7.8) is positive. In most of time, the capacitance C_{ii} computed by Eq. (7.10) is non-negative. In case that C_{ii} is negative, the formula of Eq. (7.13) is used to assure all capacitances to be non-negative. \square

Therefore, all parameters of the reduced circuits are non-negative. According to [40, 58, 66], we have

Theorem 7.3: The reduced RC circuits are stable.

7.3 Experimental Results

To illustrate the efficiency and generality of the proposed approach, some industry circuits were tested. We compare results of the original circuits and the reduced circuits using SPICE3e2 [39] on a Sun Sparc 20 workstation.

The first circuit is a clock network which consists of 26747 RC elements and has 547 external ports. The number of RC elements is reduced to 2084 which is less than 10% of that of the original circuit. Consequently, the simulation time (SPICE3e2) is reduced from 202.8 seconds to 4.4 seconds. The number of elements and simulation time could be further reduced if fewer external ports were of interest. For example, only ports on critical paths may be of interest. Table 7.1 shows the reduction results for different number of external ports. The number of circuit elements in reduced circuits versus the number of

# of ports	# of elements	reduction time (sec.)	simulation time (sec.)
2	11	39.1	0.2
6	24	38.0	0.3
12	43	40.6	0.3
22	84	38.2	0.3
52	198	36.8	0.5
102	387	38.6	0.8
202	765	44.0	1.5
302	1154	42.9	2.4
402	1537	45.3	3.3
502	1906	57.3	3.9
547	2084	55.7	4.4
Original	26747	—	202.8

Table 7.1: The clock network reduction for different number of ports

external ports is plotted in Fig. 7.3. The figure shows that the number of reduced circuit elements is $O(N)$, where N is the number of external ports. The simulation time versus the number of external ports is plotted in Fig. 7.4. Simulation results at the same port of those reduced circuits with different number of ports are almost identical (See Fig. 7.5). The error of the simulation results is less than 2%.

The reduction results of a loop-circuit and a mesh-circuit are shown in Table 7.2.

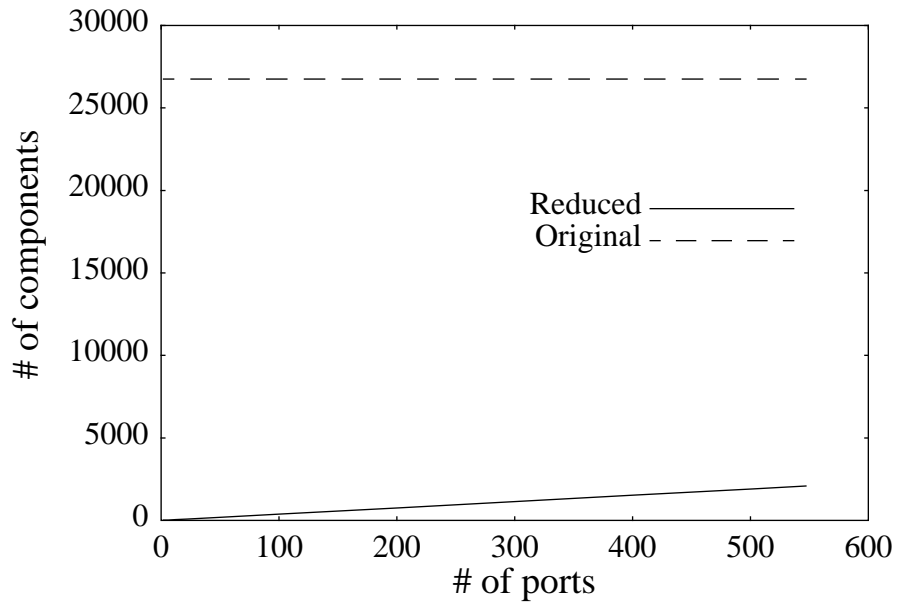


Figure 7.3. Number of the reduced circuit elements versus number of ports.

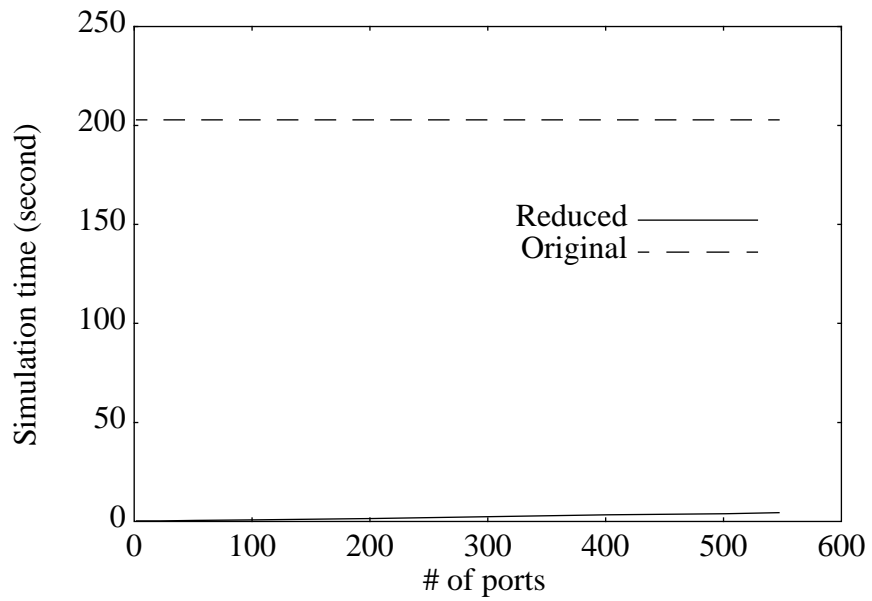


Figure 7.4. Simulation time of the reduced circuits versus number of ports.

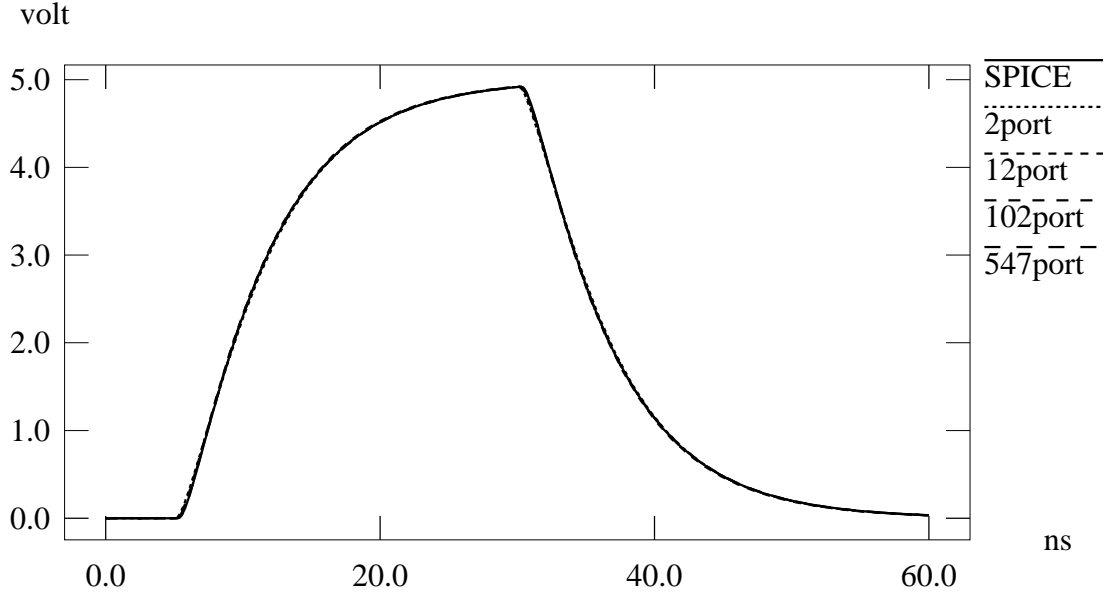


Figure 7.5. Simulation results of the clock network.

circuit	# of ports	# of elements		simulation time (sec.)		error
		original	reduced	original	reduced	
loop	102	2100	321	5.4	0.7	< 2%
mesh	102	4973	372	17.9	0.9	< 2%

Table 7.2: The loop circuit and the mesh circuit reduction results

After reduction, the simulation time of the loop-circuit and the mesh-circuit was reduced 7.7 times and 20 times, respectively. The error of the simulation results is less than 2%, compared with the original circuits.

The circuit partitioning plays a very important role in the circuit reduction. Another clock circuit consists of 14034 RC elements with 1953 external ports (7.19 elements per port). If we use only a simple multiport macromodel, the number of elements in the synthesized circuit would be much more than that of the original circuit. However, since the efficient combination of partitioning and reduction, the reduced circuit which keeps all 1953 ports external has only 8186 RC elements (4.19 elements per port).

The circuit reduction provides a new strategy to handle nonlinear circuits. A seven-port clock network with nonlinear drivers and loads is shown in Fig. 7.6. The interconnect

consists of 9427 RC elements. After reduction, the circuit has only 73 lumped elements. The simulation time for this reduced circuit is 0.6 second, while simulation time for the original circuit is 43.0 seconds. Again, the results (See Fig. 7.7) show excellent agreement in response waveforms.

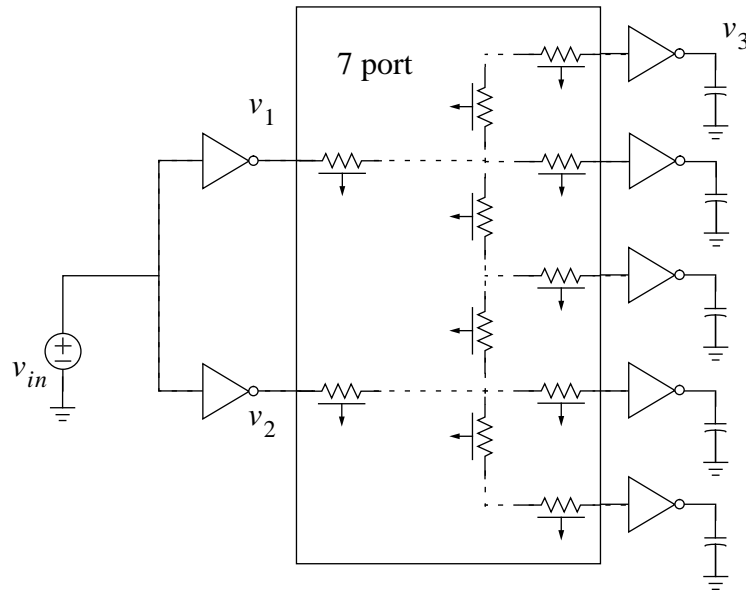


Figure 7.6. A clock network with nonlinear drivers and loads.

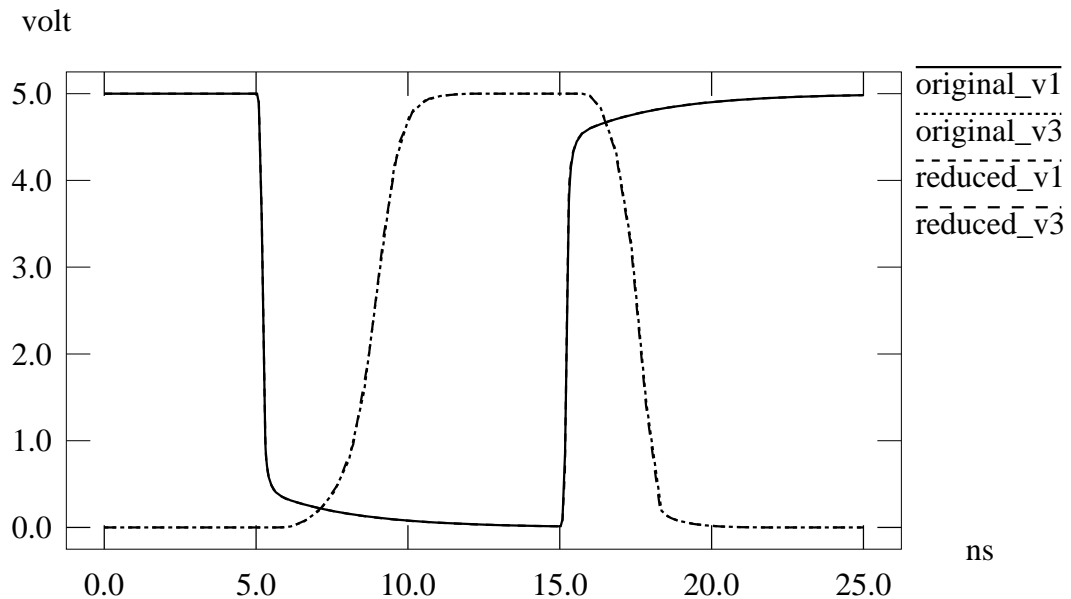


Figure 7.7. Comparison of results of the clock network system and its equivalent circuit.

CHAPTER 8 A CMOS Driver Model for Transient and Power Dissipation Analysis

Despite the increasing importance of interconnects in transient analysis, nonlinear active devices in a system also contribute to the system behavior significantly. While most transient analysis techniques of interconnect networks ignore the nonlinearity of the driving gates [69, 60], most CMOS driver models do not take into account the distributed loads [27, 35, 37]. Delay computation of a CMOS driver is based on the assumption that the output current is only dependent on the lumped load capacitance [27] and the rise time of the input waveform [35, 37]. A model for the nMOS driver with a lumped capacitive load is also proposed in [55, 56]. The output waveform is simply characterized as time-shifted ramps with exponential tails. The static power dissipation is assumed to be roughly proportional to the ratio of channel width and length. These assumptions will result in significant error for distributed interconnect loads. A current model of a nonlinear driver is proposed in [63], where a distributed interconnect load is modeled with summation of exponential functions. Since using a linearly increasing current source to model quadratically increasing output current, this model deviates from SPICE results significantly. All of these driver models ignore transient leakage (short-circuit) current. When the inverter is slightly loaded, and output rise and fall time are relatively short as compared to the input rise and fall time, then the short-circuit dissipation will increase to the same order of magnitude as the dynamic dissipation [76]. Therefore, the analysis of

the short-circuit dissipation is important for the low power design.

In this chapter, we present a new CMOS driver model used in conjunction with the macromodel of interconnect networks for transient and power dissipation analysis. This model considers the input slope effects, CMOS nonlinear effects and load interconnect effects. The driver output current is represented by a linear-quadratic-exponential piecewise model. Based on this model, we can accurately evaluate the CMOS transient leakage (short-circuit) current and short-circuit power dissipation. The model provides accuracy comparable to that of SPICE3e2 with one or two orders of magnitude less computing time.

8.1 Driver Modeling

We propose an efficient nonlinear current model here to match the output current waveform of a CMOS inverter. By lumping the input and output capacitors of the active gates into the interconnect networks, we assume that the magnitude of the current flowing out of a driver at any instant is a function of the input and output voltages of the driver at that instant. Therefore, the driver is memoryless, and the current being pumped into the interconnect is well defined.

The response of an arbitrary linear network with distributed-lumped elements can be efficiently handled by the scattering-parameter-based macromodel paradigm [48, 49]. As the load of a CMOS inverter, the linear network needs to be analyzed only once. The voltage response of the nonlinear input current, referred to as v_o , can be expressed in the frequency domain as

$$V_o(s) = Z(s)I_{out}(s) \quad (8.1)$$

where $I_{out}(s)$ is the output current of the driver, and $Z(s)$ the driving-point impedance or transfer impedance. The corresponding expression in the time domain is

$$v_o(t) = z(t)*i_{out}(t) \quad (8.2)$$

Since $z(t)$ can be approximated by a set of exponential functions, or an exponentially-decayed polynomial function [49], the remaining problem is to develop a current model to match the output current of the driver.

Let us consider a nonlinear CMOS driver shown in Fig. 8.1(a). We assume the input voltage to be a falling ramp signal or rising ramp signal. For a falling input voltage

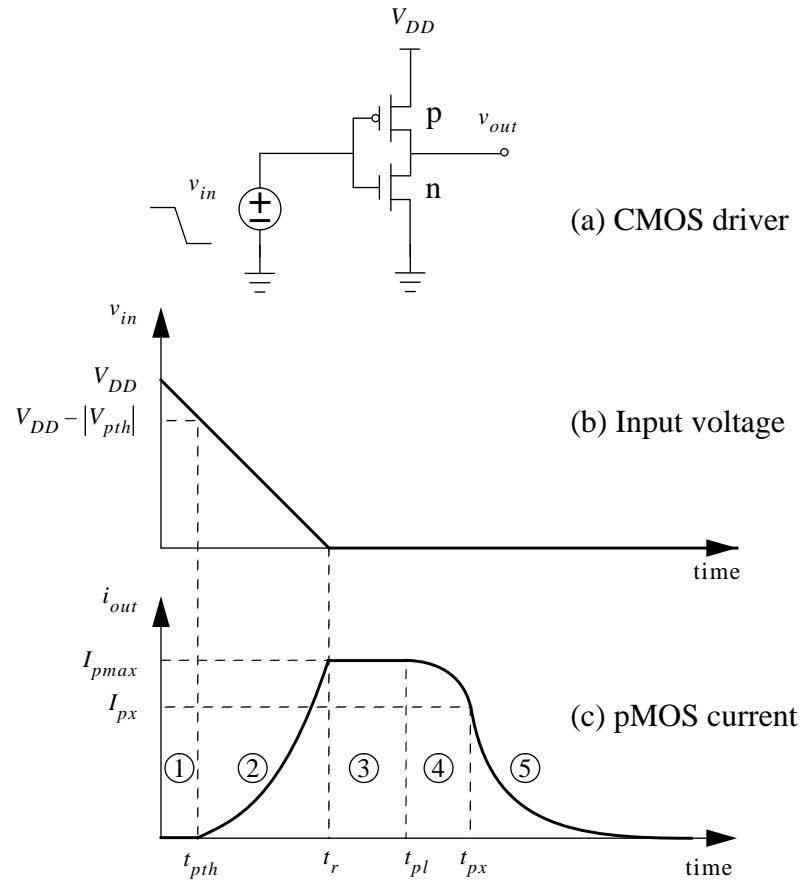


Figure 8.1. Input voltage transition and the current source model.

transition shown in Fig. 8.1(b), the corresponding output current is represented by a five-section piecewise model.

Section 1: $0 \leq t \leq t_{pth}$. As illustrated in Figure 8.1(c), the output current of the driver in this section is equal to zero since the source-gate voltage of the pMOS transistor ($V_{DD} - v_{in}(t)$) is less than its threshold voltage $|V_{pth}|$, the pMOS works in the cutoff region.

Section 2: $t_{pth} < t \leq t_r$. As the input voltage decreases, the source to gate voltage of the pMOS transistor exceeds its threshold voltage, causing it to turn on in saturation. Thus, the pMOS current increases quadratically in this section [79]. While the nMOS transistor starts from the linear region, enters saturation and finally, into cutoff region, its

current increases for a while then quadratically decreases to zero. This nMOS current is called transient leakage (short-circuit) current which will be taken into consideration in Section 8.3. Ignoring this leakage current for now, the current flowing into the interconnect is

$$i_{out}(t) = \frac{I_{pmax}}{(t_r - t_{pth})^2} (t - t_{pth})^2 \quad (8.3)$$

where I_{pmax} is the maximum current that the pMOS transistor could provide, t_r the rise time (to be precise, the fall time here) of the input signal, and t_{pth} the time when the pMOS transistor starts conducting current, or when the following condition is true [79],

$$V_{DD} - v_{in}(t_{pth}) = |V_{pth}| \quad (8.4)$$

If the output voltage is low enough, the section 2 model is employed until the time t_r at which the input transition reaches its final value, then switches to section 3. If the output voltage goes up quickly, pMOS turns on in the linear region, the gate thus enters section 4 directly. The method to determine which case will occur is by computing t_{pl} , which meets the following equation [79]:

$$v_{in}(t_{pl}) + |V_{pth}| = v_{out}(t_{pl}) \quad (8.5)$$

If $t_{pl} > t_r$, pMOS is in the saturation region for $t < t_r$, the gate enters section 3 at $t = t_r$. Otherwise, if $t_{pl} \leq t_r$, the pMOS turns on in the linear region at $t = t_{pl}$, the gate thus enters section 4 directly (See Fig. 8.2). In the latter case, the initial current value in section 4 is

$$I_{pl} = \frac{I_{pmax}}{(t_r - t_{pth})^2} (t_{pl} - t_{pth})^2 \quad (8.6)$$

Section 3: $t_r < t \leq t_{pl}$. This section begins at t_r , when the input voltage is down to zero and the pMOS transistor is still in the saturation region. A constant current source with magnitude I_{pmax} represents the model. The I_{pmax} is the maximum current that can through the pMOS [79]

$$I_{pmax} = \frac{1}{2} \beta (V_{DD} - |V_{pth}|)^2 \quad (8.7)$$

where β is the MOS transistor gain factor. The factor is dependent on both the process

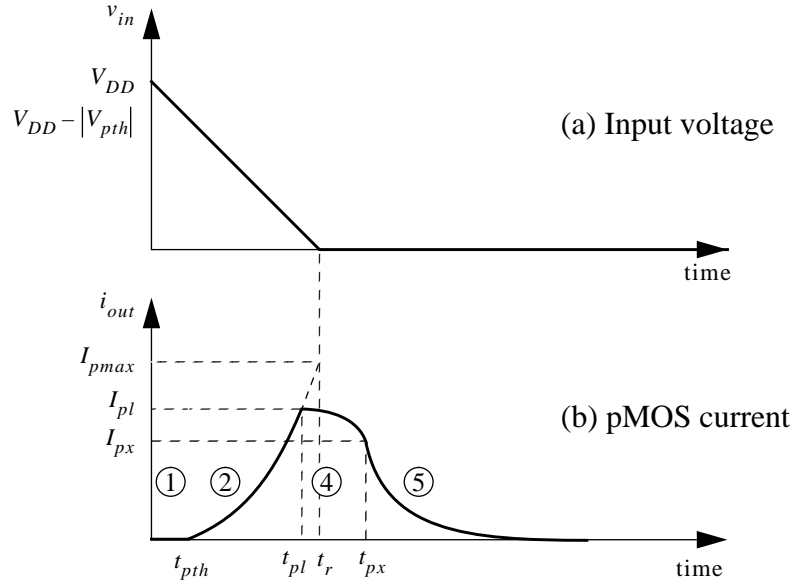


Figure 8.2. pMOS current when $t_{pl} < t_r$.

parameters and the device geometry, and is given by

$$\beta = \frac{\mu\epsilon}{t_{ox}} \left(\frac{W}{L} \right) \quad (8.8)$$

where μ is the effective surface mobility of the carriers in the channel, ϵ the permittivity of the gate insulator, t_{ox} the thickness of the gate insulator, W the width of the channel, and L the length of the channel. In this case, the initial current of section 4 is $I_{pl} = I_{pmax}$.

When the output voltage rises to a value such that pMOS transistor enters its linear region (See Eq. (8.5)), the output current of the pMOS becomes [79]

$$i_{out}(t) = \beta \left[(v_{sg}(t) - |V_{pth}|) v_{sd}(t) - v_{sd}^2(t)/2 \right] \quad (8.9)$$

where $v_{sg}(t) = V_{DD} - v_{in}(t)$ and $v_{sd}(t) = V_{DD} - v_{out}(t)$. Although this region is commonly called the linear region, $i_{out}(t)$ varies linearly with $v_{sd}(t)$ only when the quadratic term $v_{sd}^2(t)/2$ is small. Instead of modeling the remaining pMOS current in one section such as [63], we model this region into two sections.

Section 4: $pl < t \leq t_{px}$. t_{px} is the time at which the $v_{sd}(t)$ is a half of the voltage

$v_{sg}(t) - |V_{pth}|$. That is, $v_{sd}(t_{px}) = (v_{sg}(t_{px}) - |V_{pth}|) / 2$, or

$$V_{DD} - v_{out}(t_{px}) = (V_{DD} - v_{in}(t_{px}) - |V_{pth}|) / 2 \quad (8.10)$$

Thus,

$$v_{out}(t_{px}) = (V_{DD} + v_{in}(t_{px}) + |V_{pth}|) / 2 \quad (8.11)$$

In this section, the output current is a nonlinear function of the output voltage. We model this output current with a quadratic curve, that is

$$i_{out}(t) = I_{pl} - \frac{I_{pl} - I_{px}}{(t_{px} - t_{pl})^2} (t - t_{pl})^2 \quad (8.12)$$

where $I_{px} = i_{out}(t_{px})$ is the pMOS output current at $t = t_{px}$. $i_{out}(t_{px})$ and $v_{out}(t_{px})$ meet the DC property (Eq. (8.9)), and the load property (Eq. (8.2)).

Section 5: $t > t_{px}$. When output voltage continuously goes up, the transistor with the constant input voltage behaves like a linear resistor. The output current is modeled by an exponential decay function as the output voltage reaches its final value. The time constant τ is initially computed based on the charge which has been pumped into the loading network. Since the initial value of the current is known, the exponential decay function is determined solely by its time constant. Assume that output current in this section is

$$i_{out}(t) = I_{px} e^{-(t-t_{pl})/\tau} \quad (8.13)$$

The charge pumped into the interconnect in section 5 is $I_{px}\tau$. On the other hand, the total charge previously pumped into the interconnect in section 1, 2, 3 and 4 is

$$Q_{prev} = \begin{cases} \frac{1}{3} (2t_{px} + t_{pl} - 2t_r - t_{pth}) I_{pmax} & (t_{pl} \geq t_r) \\ \frac{1}{3} (2t_{px} - t_{pl} - t_{pth}) I_{pl} & (t_{pl} < t_r) \end{cases} \quad (8.14)$$

Thus, if there is no DC path from source to the ground, we can determine the time constant τ by

$$I_{px}\tau + Q_{prev} = C_{eqv} V_{DD} \quad (8.15)$$

where C_{eqv} is the equivalent driving-point capacitance of the interconnect network and can be computed as the load of the driver by the scattering-parameter-based macromodel.

In case that there is a DC path from the source to the ground, there is a non-zero steady output current, and C_{eqv} is infinite. The pMOS steady current I_{pinf} and output voltage $v_{out}(\infty)$ should meet Eq. (8.9) and

$$v_{out}(\infty) = R_{eqv} I_{pinf} \quad (8.16)$$

where R_{eqv} is the equivalent driving-point resistance. The output current described in Eq. (8.13) can be modified to

$$i_{out}(t) = I_{pinf} + (I_{px} - I_{pinf}) e^{-(t-t_{pl})/\tau} \quad (8.17)$$

The time constant τ of the exponentially-decayed current can also be determined based on Eq. (8.9), and Eq. (8.2). The method to determine if there is a DC path from source to the ground, as well as the C_{eqv} or R_{eqv} , will be discussed in the next section.

8.2 Transient Analysis with Distributed Interconnect Load

The response of an arbitrary linear network with distributed-lumped elements can be efficiently handled with the scattering-parameter-based macromodel paradigm [49]. As a load of CMOS inverter, the linear network has to be analyzed only one time.

8.2.1 Computation of Equivalent Resistance or Capacitance

In order to use the proposed model, we need to know if there is DC path from source to the ground. This test can be easily done by examining the driven-point scattering parameter $S_{in}(s)$ of the interconnect network as described by the following theorems.

From $S_{in}(s)$, we can easily determine if there is a grounded resistive element.

Theorem 8.1: An interconnect system has no DC path from the source to the ground if and only if $S_{in}(0) = 1$, where $S_{in}(s)$ is the driving-point S-parameter at frequency s , and $S_{in}'(0)$ is the first moment of $S_{in}(s)$.

Proof: Consider the relation between the scattering parameter and the impedance for one port component, we have

$$S_{in}(s) = \frac{Z_{in}(s) - Z_0}{Z_{in}(s) + Z_0} \quad (8.18)$$

where Z_0 is the reference impedance. If there is no grounded resistive elements, the driving-point impedance $Z_{in} = \infty$ at $s = 0$, thus $S_{in}(0) = 1$. On the other hand, rewrite

the above equation

$$Z_{in}(s) = \frac{1 + S_{in}(s)}{1 - S_{in}(s)} Z_0 \quad (8.19)$$

If $S_{in}(0) = 1$, then $Z_{in}(0) = \infty$, indicating no DC path from the source to the ground. \square

Theorem 8.2: For an interconnect system with at least one DC path from the source to the ground, the equivalent driving-point resistance is

$$R_{eqv} = Z_{in}(0) = \frac{1 + S_{in}(0)}{1 - S_{in}(0)} Z_0 \quad (8.20)$$

The theorem can be easily proved from Eq. (8.19). This theorem shows that the equivalent driving-point resistance is only dependent on the first moment of the driving-point S-parameter.

From the property of Taylor series, the second moment of the $S_{in}(s)$ is $\frac{d}{ds} S_{in}(0)$. The following theorem shows that the equivalent driving-point capacitance is only dependent on the second moment of S_{in} .

Theorem 8.3: For an interconnect system with no DC path from the source to the ground, the equivalent driving-point capacitance is

$$C_{eqv} = -\frac{1}{2Z_0} \frac{d}{ds} S_{in}(0) \quad (8.21)$$

Proof: From Eq. (8.19), since $S_{in}(0) = 1$, there is a pole at $s = 0$ for driving-point impedance. The corresponding residue is

$$k_0 = \left. \frac{(1 + S_{in}) Z_0}{\frac{d}{ds} (1 - S_{in})} \right|_{s=0} = -\frac{(1 + S_{in}(0)) Z_0}{\frac{d}{ds} S_{in}(0)} = -\frac{2Z_0}{\frac{d}{ds} S_{in}(0)} \quad (8.22)$$

and the equivalent driving-point capacitance is

$$C_{eqv} = \frac{1}{k_0} = -\frac{1}{2Z_0} \frac{d}{ds} S_{in}(0) \quad \square$$

Consequently, the pole at $s = 0$ makes it impossible to express the driving-point impedance in Taylor series. In this case, the function to be matched with exponential func-

tions becomes

$$Z_{in} = Z_{in} - \frac{1}{C_{eqv}s} \quad (8.23)$$

These three theorems provide a method to determine if there is a DC path from the source to the ground, and the means to compute the equivalent driving-point resistance R_{eqv} or the equivalent driving-point capacitance C_{eqv} , which are used in the current model. We have thus completed the construction of the driver model. This new model is an accurate approximation for a wide range of CMOS gates. The next section focuses on how to combine this driver model with the Padé or Exponentially-Decayed Polynomial Function (EDPF) approximation of the interconnect for transient analysis.

8.2.2 Transient Analysis

A driver model should be combined with the transfer function of the interconnect network which is the load of the driver (See Eq. (8.2)). Based on the scattering-parameter-macromodel, the transfer function in the time domain can be approximated as a summation of exponentials using Padé technique [48]. A transfer function with q -th order Padé approximation has the following form:

$$h_p(t) = \sum_{i=1}^{q_p} k_i e^{p_i t} \quad (8.24)$$

where p_i are the poles and k_i the corresponding residues. The transfer function can also be approximated as an Exponentially-Decayed Polynomial Function (EDPF) [49]. A transfer function with q -th order EDPF approximation has the following forms:

$$h_e(t) = \sum_{i=0}^{q_e} c_i \left(\frac{t}{d}\right)^i e^{-(t/d)} \quad (8.25)$$

where d is the time constant of the EDPF, c_i the coefficients.

On the other hand, based on the analysis in last section, the output current of a driver can be expressed as (for $t_{pl} \geq t_r$)

$$i_{out}(t) = \begin{cases} 0 & t \leq t_{pth} \\ \frac{I_{pmax}}{(t_r - t_{pth})^2} (t - t_{pth})^2 & t_{pth} < t \leq t_r \\ I_{pmax} & t_r < t \leq t_{pl} \\ I_{pmax} - \frac{I_{pmax} - I_{px}}{(t_{px} - t_{pl})^2} (t - t_{pl})^2 & t_{pl} < t \leq t_{px} \\ I_{pinf} + (I_{px} - I_{pinf}) e^{-(t - t_{pl})/\tau} & t_{px} < t \end{cases} \quad (8.26)$$

Similar formulas can also be derived for the other case where $t_{pl} < t_r$. To symbolically convolve the transfer function with output current of the driver $i_{out}(t)$, we take each term in $h_p(t)$ or $h_e(t)$ with each term in $i_{out}(t)$. Suppose a quadratic function is

$$f(t) = \begin{cases} t^2 & t \leq t_0 \\ 0 & t > t_0 \end{cases} \quad (8.27)$$

The effect of the quadratic function on an exponential in $h_p(t)$ can be explicitly represented, for $t \leq t_0$, by

$$f(t) * (k_i e^{p_i t}) = \begin{cases} k_i t^3 / 3 & p_i = 0 \\ k_i (2e^{p_i t} - 2 - 2p_i t - p_i^2 t^2) / p_i^3 & p_i \neq 0 \end{cases} \quad (8.28)$$

For $t > t_0$, it can also be expressed by

$$f(t) * (k_i e^{p_i t}) = \begin{cases} k_i t_0^3 / 3 & p_i = 0 \\ k_i e^{p_i t} (2 - (2 + 2p_i t_0 + p_i^2 t_0^2) e^{-p_i t_0}) / p_i^3 & p_i \neq 0 \end{cases} \quad (8.29)$$

The convolution of $h_e(t)$ with input can be computed by recursive formulas. For example, the effect of the quadratic input signal on the q th order exponentially-decayed polynomial function for $t \leq t_0$ is

$$\begin{aligned}
F_q(t) &= f(t) * h_e(t) \\
&= \sum_{j=0}^q c_j (t^2 - 2td(j+1) + d^2(j+1)(j+2)) f_j(t) + \\
&\quad \sum_{j=0}^q c_j t d (t - d(j+2)) \left(\frac{t}{d}\right)^j e^{-t/d}
\end{aligned} \tag{8.30}$$

where

$$f_j(t) = \int_0^t \left(\frac{\tau}{d}\right)^j e^{-\tau/d} d\tau \tag{8.31}$$

with $f_j(t) = j f_{j-1}(t) - d(t/d)^j e^{-t/d}$ and $f_0(t) = d(1 - e^{-t/d})$. Based on Eq. (8.30) and Eq. (8.31), we have a simple algorithm to implement the convolution process:

```

value ← 0, K ← 0
for (j ← order; j ≥ 0; j ← j - 1)
    K ← (j + 1) K + c_j (t^2 - 2td(j + 1) + d^2(j + 1)(j + 2))
    value ← value × t/d + c_j t d (t - d(j + 2)) - d × K
value ← value × e^{-t/d} + d × K

```

The proof of the algorithm is omitted for brevity. The formulas for $t > t_0$, and convolution of $h_p(t)$ or $h_e(t)$ with a step, ramp and exponentially-decaying function can also be computed by time domain explicit formulas or recursive formulas.

8.3 Power Dissipation Analysis with Transient Leakage Current

Most CMOS driver models ignore transient leakage (short-circuit) current. When the rise time of the output voltage is comparable to the rise time of the input signal, the leakage current become significant, and the corresponding leakage dissipation will be similar in magnitude as the dynamic dissipation [76]. Previous methods of evaluating the leakage current are all based on the assumption that the load is a lumped capacitor [35, 76].

In order to accurately estimate the leakage current of the CMOS driver with a general interconnect load, we analyze the leakage current of the nMOS transistor for the falling ramp input signal shown in Fig. 8.1. The current through nMOS can be represented by a

four-section piecewise model (See Fig. 8.3).

Section 1: $t \leq t_{pth}$. Here i_n equals zero since i_p and output voltage are zero.

Section 2: $t_{pth} < t \leq t_{ns}$. As the pMOS transistor turns in the saturation region and output voltage increases, the nMOS goes into linear region. In this section, input voltage goes down linearly and output voltage increases nonlinearly. We use a quadratically increasing current source to model the nMOS nonlinearly increasing current. This section is employed until the nMOS enters to the saturation region at t_{ns} , where t_{ns} is the time that input voltage and output voltage meet the following condition:

$$v_{in}(t_{ns}) - v_{nth} = v_{out}(t_{ns}) \quad (8.32)$$

So, the nMOS current in this section is

$$i_n(t) = \frac{I_{ns}}{(t_{ns} - t_{pth})^2} (t - t_{pth})^2 \quad (8.33)$$

where I_{ns} is determined by t_{ns} and I_{nmax} , the maximum current that the nMOS can

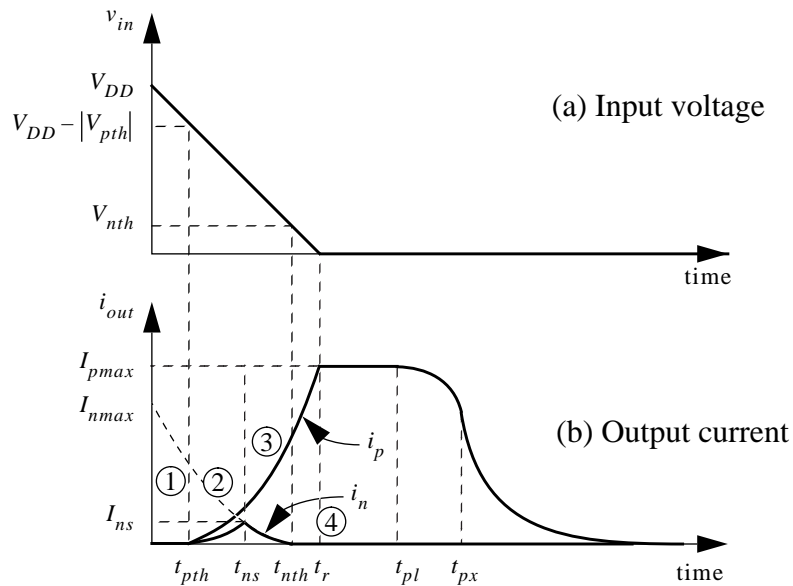


Figure 8.3. pMOS current i_p and nMOS current (leakage current) i_n .

supply.

Section 3: $t_{ns} < t \leq t_{nth}$. As the nMOS switches to the saturation region, the current quadratically decreases. This process continues until the current dies down to zero at t_{nth} , when the nMOS goes into the cutoff region. The current in this region can be expressed by

$$i_n(t) = \frac{I_{nmax}}{2} \frac{(t_{nth} - t)^2}{t_{nth}} \quad (8.34)$$

Note that

$$I_{ns} = \frac{I_{nmax}}{2} \frac{(t_{nth} - t_{ns})^2}{t_{nth}} \quad (8.35)$$

Section 4: $t > t_{nth}$. In this section, the nMOS is in the cutoff region, its current is zero.

Overall, the current through the nMOS can be expressed as

$$i_n(t) = \begin{cases} \frac{I_{ns}}{(t_{ns} - t_{pth})^2} (t - t_{pth})^2 & t_{pth} < t \leq t_{ns} \\ \frac{I_{nmax}}{2} \frac{(t_{nth} - t)^2}{t_{nth}} & t_{ns} < t \leq t_{nth} \\ 0 & otherwise \end{cases} \quad (8.36)$$

The convolution formula we have derived early can also be used here. All charges go through nMOS transistor are

$$Q_{nprev} = \int i_n(t) dt = \frac{I_{ns}}{3} (t_{nth} - t_{pth}) \quad (8.37)$$

Thus, the short-circuit power dissipation for this transient change is

$$W_s = V_{DD} Q_{nprev} = \frac{V_{DD} I_{ns}}{3} (t_{nth} - t_{pth}) \quad (8.38)$$

8.4 Experimental Results

The proposed driver models were tested on several circuits using SUN SPARC 1+ workstation.

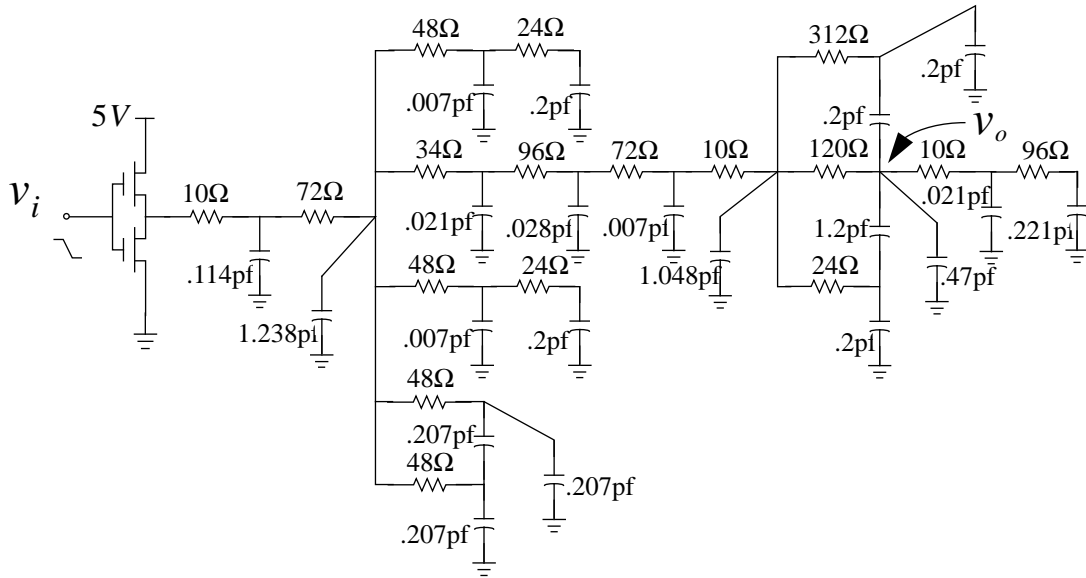


Figure 8.4. An RC circuit with floating capacitors.

An RC circuit with floating capacitors is shown in Fig. 8.4. The driver model parameter includes maximum current ($11.2mA$), threshold voltage ($1.0V$) and the rise time of the input signal ($1.0ns$). Output voltage V_o computed by our model matches very well with the SPICE result (See Fig. 8.5). SPICE3e2 [39] took 1.2 CPU seconds and our method took 0.4 second.

The second example is a grid-type clock network (See Fig. 8.6). The vertical runs are on metal layer 1 ($R = 60\Omega/m$, $L = 548.0nH/m$ and $C = 142.3pF/m$) and the horizontal runs are on metal layer 2 ($R = 40\Omega/m$, $L = 502.5nH/m$ and $C = 155.2pF/m$). The length of these transmission lines is $0.03m$. The parameters of the driver are: maximum current is $28.8mA$, threshold voltage $1.0V$ and the rise time of the input signal $1.0ns$. With direct convolution, SPICE3e2 took more than 200 CPU seconds to analyze the circuit. But our method took only 0.9 CPU second to generate the voltage responses which tracks well with the SPICE result (See Fig. 8.7).

If the CMOS is designed in such a way that the rise/fall time of output signal is comparable to that of input signal, the CMOS leakage current will increase in the same order of magnitude as the output current. This can be illustrated by the following small circuit

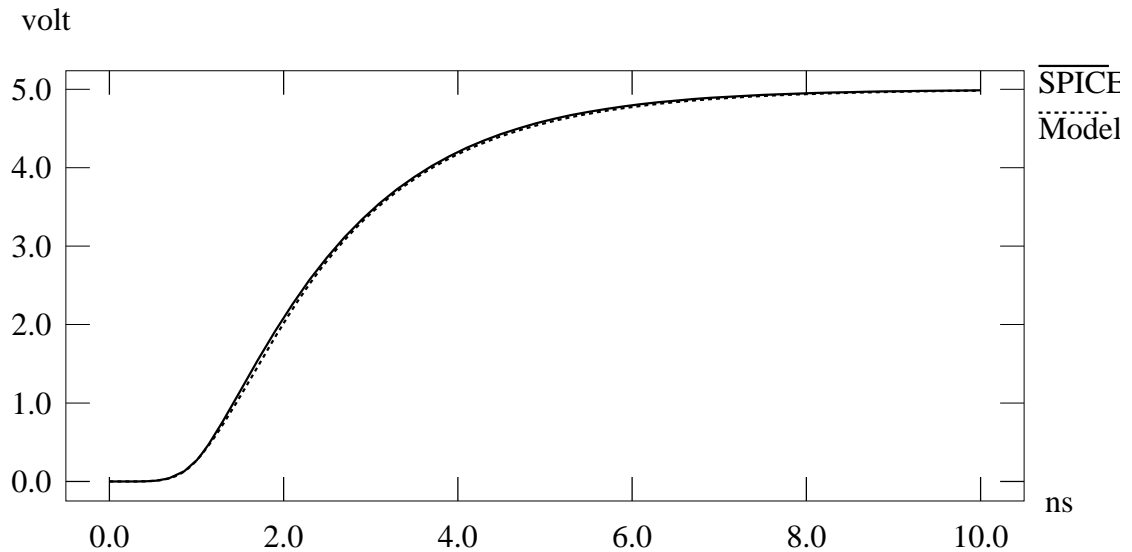


Figure 8.5. Voltage response of the RC circuit.

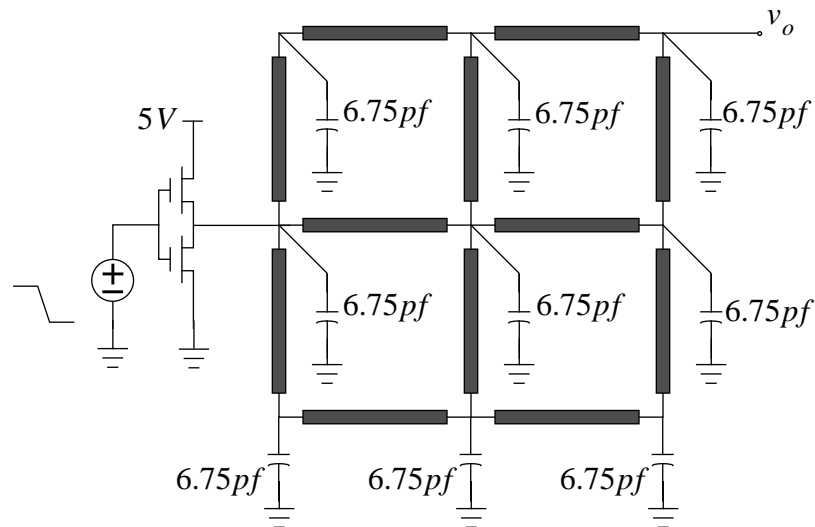


Figure 8.6. A grid-type clock network.

(See Fig. 8.8). The threshold voltages of both pMOS and nMOS are 1.0volt. The maximum current of pMOS is $3.2mA$, and that of nMOS $6.4mA$. The rise time of the input signal is 10ns. SPICE3e2 took more than 400 CPU seconds to analyze this circuit because of the “short” transmission line. Our method only took 0.4 second. The pMOS and nMOS

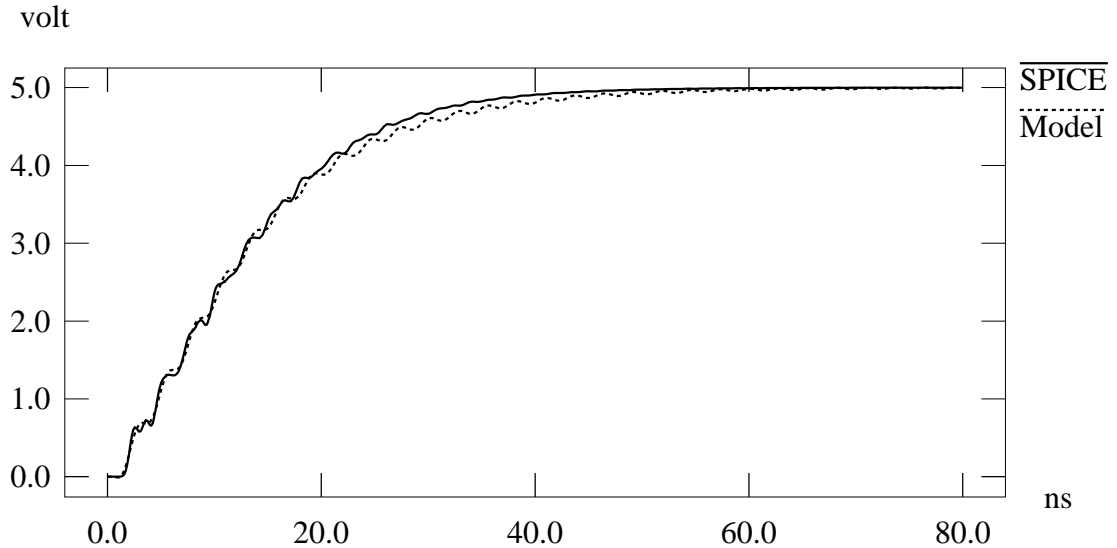


Figure 8.7. Voltage response of the clock network.

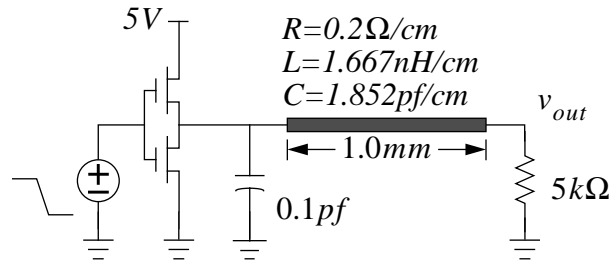


Figure 8.8. A transmission line circuit with a DC path from source to ground.

currents are shown in Fig. 8.9. The steady output current is not equal zero because there is a DC path from source to ground. Obviously, if we ignored the nMOS current (leakage current), it would create quite large error. The voltage response is shown in Fig. 8.10.

Finally, we demonstrate the efficiency of the driver model by evaluating a large industry circuit. The circuit, driven by a CMOS inverter, includes 2895 resistors, 2767 capacitors and 73 lossy transmission lines. SPICE3e2 [39] took more than 5200 seconds to analyze the circuit and our method took 86.5 seconds. The voltage response is shown in Fig. 8.11.

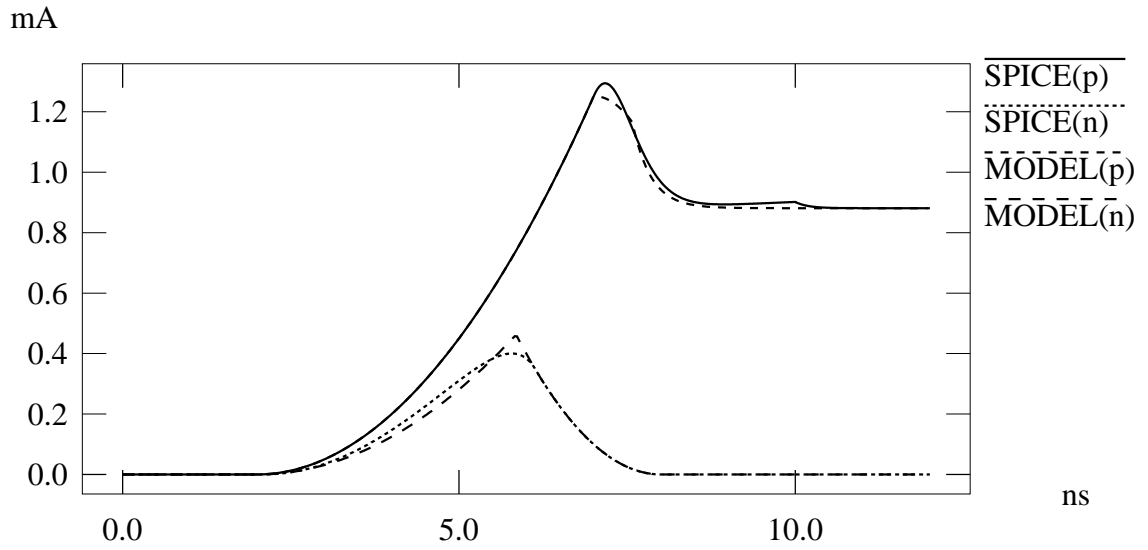


Figure 8.9. pMOS and nMOS currents of the transmission line circuit.

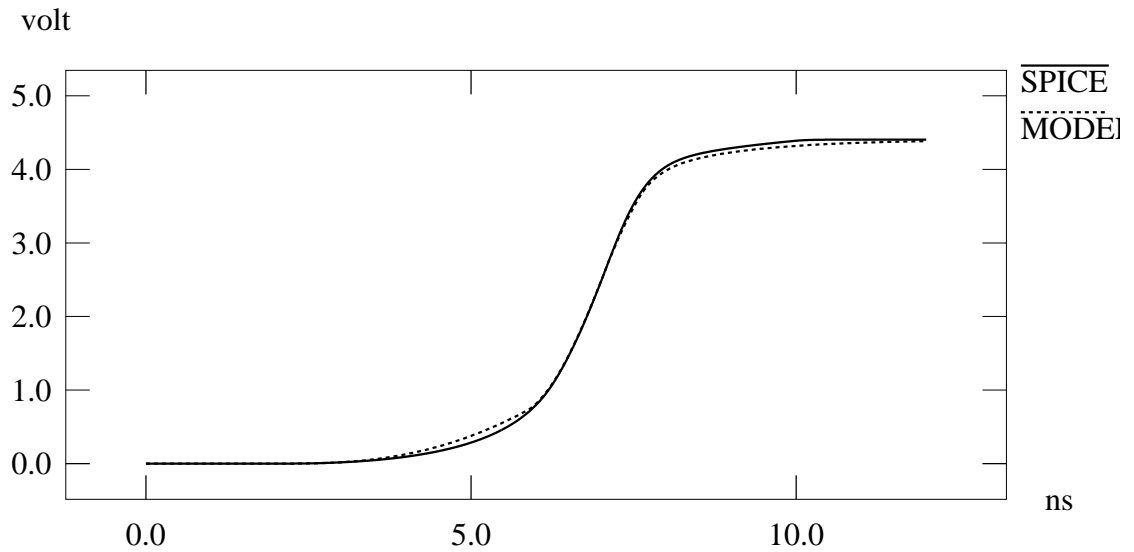


Figure 8.10. Voltage response of the transmission line circuit.

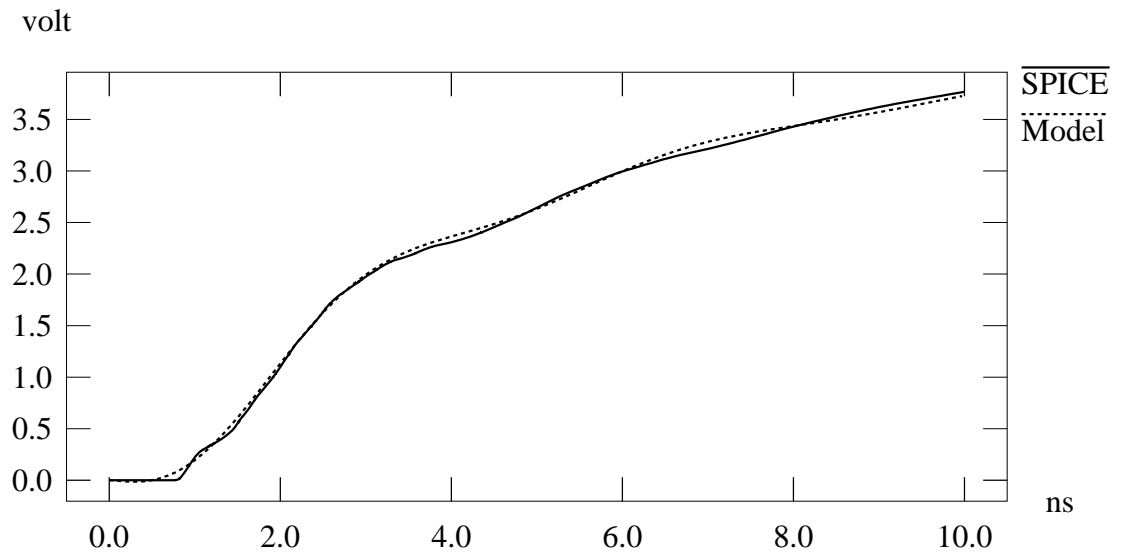


Figure 8.11. Voltage response of a large clock network.

CHAPTER 9 Conclusions

9.1 Concluding Remarks

The scattering-parameter-based macromodel provides an efficient method and strategy to analyze interconnect networks. Lossy transmission lines, floating capacitors and inductors, capacitive or inductive cutsets and loops are all easily addressed with the macromodel. Time-of-flight delay of transmission line networks can be explicitly captured, which greatly improves the accuracy of the macromodel. Large interconnect networks with large number of external ports or nonlinear terminations are easily dealt with by efficiently combining circuit reduction and partition. Circuit partitioning speeds up the reduction process, decreases the simulation time and achieves high accuracy using lower order approximation. Accuracy can be controlled by adjusting the circuit element number in each subnetwork. The interconnect networks with nonlinear terminations can be analyzed with recursive convolution techniques or by synthesizing the reduced networks. Because of its generality, the macromodel can be used for interconnect analysis on integrated circuit chips, multichip modules and printed circuit boards. Since no assumption is made regarding port voltages and currents, the proposed macromodel can be applied to the simulation of analog circuits and mixed-signal circuits. In conjunction with the macromodel, the proposed CMOS driver model, which takes into account the input slope effects, CMOS nonlinear effects, and load interconnect effects, can be used for event-driven timing analysis and power dissipation analysis for large digital circuits.

9.2 Future Research

A number of open theoretical and practical questions are raised as the results of this thesis research.

The scattering-parameter-based macromodel has been extended to analyze sensitivity of interconnect networks [75]. The sensitivity and time-of-flight can be kept during network reduction. The technique that keeps the track of information of time-of-flight and sensitivity during reduction can be used to merge all linear independent sources, that is, the ports which are connected to those sources need not to be kept external. The method will be useful for large circuits with multiple sources, for example, the power and ground planes where all sources and drains of CMOS transistors are modelled as current sources, and for the circuits with initial conditions which may exist in the form of static charge during the idle-period of logic switching and the fetch/store transition period of memory circuits [17]. Such cases are often ignored in the transient simulations of large interconnect networks.

Although Padé approximation provides a feasible and accurate model for RC circuits, it tends to yield unstable poles when the number of poles (order of approximation) exceeds seven. We have introduced mixed-exponential functions to deal this problem. However, extremely ill-conditioned matrices because of the severe numerical problems may still exist when computing high order moments. The complex frequency hopping method [22] and the PVL (Padé approximation via the Lanczos process) [29] help to compute high order moments by avoiding the ill-condition problem. However, difficulty of determining approximation order with moment matching technique hinders their use. In addition, increasing the approximation order will increase the modeling and simulation time. Partitioning and reduction is a feasible solution. We have proposed an efficient partitioning method to speed up the reduction and simulation. Efficient data modeling beyond moment matching methods still to be developed. Scattering-parameter-based macromodel can compute S-matrix at sampling frequency points [50]. Although least-squares, which can compute the error norm of approximation [74], could be used, a more robust data modeling technique is needed to avoid numerical problems when the number of poles goes larger. Linear system identification technique [71] could be one of choices.

Approximating the multiport macromodel is another open question. The more strict requirement that the reduced model is stable must impose the approximation process. This

requirement includes testing bounded-real condition of the reduced network scattering matrix and avoiding non-bounded-real results.

The most outstanding open problem in circuit synthesis is to synthesize multiport reciprocal networks without the use of transformers [58]. We have presented a lower order RC synthesis of interconnect networks with RC elements. The synthesized circuits are always stable. The technique may be extended to handle RLC circuits and transmission line networks where there are time-of-flight delays between some pairs of ports.

Appendix A Scattering Matrix of an Ideal Interconnect Node

For each vertex corresponding to an interconnect node (Figure A.1), Kirchoff's voltage law and current law hold. For the n port interconnect node, we have

$$V_1 = V_2 = \dots = V_n \quad (\text{A.1})$$

$$I_1 + I_2 + \dots + I_n = 0 \quad (\text{A.2})$$

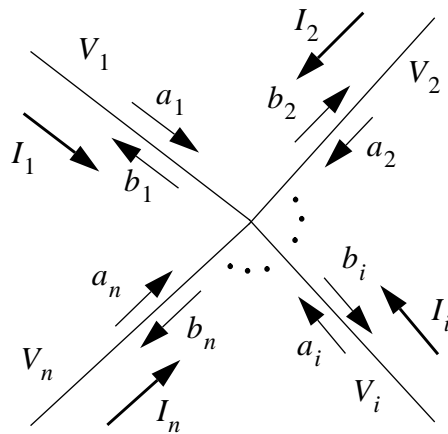


Figure A.1. An ideal interconnect node.

where V_i and I_i are voltage and current respectively at port i . On the other hand, the circuit parameters have a clear and definite relationship with the wave parameters a_i and b_i , that is

$$a_i + b_i = V_i \text{ and } a_i - b_i = Z_{0i}I_i \quad (\text{A.3})$$

where Z_{0i} is the reference impedance at port i . Combining Eqs. (A.1) and (A.2) with Eq. (A.3), we have

$$a_1 + b_1 = a_2 + b_2 = \dots = a_n + b_n \quad (\text{A.4})$$

$$\sum_{i=1}^n \frac{a_i}{z_{0i}} = \sum_{i=1}^n \frac{b_i}{z_{0i}} \quad (\text{A.5})$$

Write Eqs. (A.4) and (A.5) in a matrix form,

$$\begin{bmatrix} 1 & -1 & 0 & \dots & 0 \\ 0 & 1 & -1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 1 & -1 \\ -1 & -1 & \dots & \dots & -1 \\ z_{01} & z_{02} & \dots & \dots & z_{0n} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{bmatrix} = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & -1 & 1 \\ -1 & -1 & \dots & \dots & -1 \\ z_{01} & z_{02} & \dots & \dots & z_{0n} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_n \end{bmatrix} \quad (\text{A.6})$$

Solving the system of the equations, we have

$$\mathbf{B} = \mathbf{S}\mathbf{A} \quad (\text{A.7})$$

where $\mathbf{A} = [a_1, a_2, \dots, a_n]^T$, $\mathbf{B} = [b_1, b_2, \dots, b_n]^T$ and

$$\mathbf{S} = \frac{2}{G} \begin{bmatrix} \frac{1}{z_{01}} - \frac{1}{2}G & \frac{1}{z_{02}} & \dots & \frac{1}{z_{0n}} \\ \frac{1}{z_{01}} & \frac{1}{z_{02}} - \frac{1}{2}G & \dots & \frac{1}{z_{0n}} \\ \dots & \dots & \dots & \dots \\ \frac{1}{z_{01}} & \frac{1}{z_{02}} & \dots & \frac{1}{z_{0n}} - \frac{1}{2}G \end{bmatrix} \quad (\text{A.8})$$

where

$$G = \sum_{i=1}^n \frac{1}{z_{0i}} \quad (\text{A.9})$$

If all reference impedances are same, that is, $z_{01} = z_{02} = \dots = z_{0n}$, then

$$\mathbf{S} = \frac{1}{n} \begin{bmatrix} 2-n & 2 & \dots & 2 \\ 2 & 2-n & \dots & 2 \\ \dots & \dots & \dots & \dots \\ 2 & 2 & \dots & 2-n \end{bmatrix} \quad (\text{A.10})$$

The S-matrix of Eq. (A.8) or Eq. (A.10) has some properties [46]. Since $S_{ik} = S_{jk}$ for $i \neq k$ and $j \neq k$, we have

Theorem A.1: The junction vertices contain no preferred voltage wave ports; that is, outgoing waves at all other ports due to the incoming wave at one port are equal.

For example, if port 1 has an incoming wave a_1 and other ports are perfectly matched (no incoming wave), outgoing waves on all ports other than port 1 are equal to $2G^{-1}z_{01}^{-1}a_1$.

The S_{ii} is called the reflection factor at port i . We say the port i is perfectly matched (no reflection) if $S_{ii} = 0$.

Theorem A.2: If a junction vertex has degree greater than or equal to three, it is impossible for all its ports to be perfectly matched. That is, it is impossible that $S_{ii} = 0$ for $i = 1, 2, \dots, n$ if $n \geq 3$.

Proof: Assume that all $S_{ii} = 0$ ($i = 1, 2, \dots, n$) for a junction vertex, that is,

$$\frac{2}{Gz_{0i}} - 1 = 0 \quad i = 1, 2, \dots, n \quad (\text{A.11})$$

Since $G \neq 0$, we can rewrite Eq. (A.11) as:

$$\frac{2}{z_{0i}} - G = 0 \quad i = 1, 2, \dots, n \quad (\text{A.12})$$

the matrix form of these equations is

$$\begin{bmatrix} 1 & -1 & -1 & \dots & -1 \\ -1 & 1 & -1 & \dots & -1 \\ -1 & -1 & 1 & \dots & -1 \\ \dots & \dots & \dots & \dots & \dots \\ -1 & -1 & -1 & \dots & 1 \end{bmatrix} \begin{bmatrix} 1/z_{01} \\ 1/z_{02} \\ \dots \\ 1/z_{0n} \end{bmatrix} = \mathbf{0} \quad (\text{A.13})$$

Since the determinant of the coefficient matrix is $-(n-2)2^{n-1} \neq 0$ when $n \geq 3$, there exists no non-trivial solution to these equations. Therefore, the assumption that all $S_{ii} = 0$ ($i = 1, 2, \dots, n$) can not be true in general. \square

In [23], S-parameter analysis of multi-conjugate networks is based on the assumption, among others, that all interconnect elements must be perfectly matched. Clearly, it contradicts to the above theorem.

Obviously, the ideal junction vertices neither store nor consume energy. We can prove this with the S-matrix described in Eq. (A.8).

Theorem A.3: The junction vertices neither store nor consume energy.

Proof: For an n port junction vertex, the input power denoted by P_{in} and output power denoted by P_{out} are

$$P_{in} = \mathbf{B}'^* \mathbf{Z}_0 \mathbf{B} \quad (\text{A.14})$$

$$P_{out} = \mathbf{A}'^* \mathbf{Z}_0 \mathbf{A} \quad (\text{A.15})$$

where \mathbf{B} and \mathbf{A} are input voltage waves and output voltage waves respectively, \mathbf{B}'^* and \mathbf{A}'^* are their corresponding conjugate vectors, and $\mathbf{Z}_0 = \text{diag} [z_{01}, z_{02}, \dots, z_{0q}]$. From Eq. (A.7),

$$P_{out} = (\mathbf{S}\mathbf{B})'^* \mathbf{Z}_0 (\mathbf{S}\mathbf{B}) = \mathbf{B}'^* \mathbf{S}'^* \mathbf{Z}_0 \mathbf{S} \mathbf{B} = \mathbf{B}'^* \mathbf{Z}_0 \mathbf{B} = P_{in} \quad (\text{A.16})$$

Therefore, the junction vertices neither store nor consume energy. \square

Appendix B Recursive Convolution Based on EDPF

Consider the following convolution

$$y(t) = h(t) * x(t) = \int_0^t h(t-\lambda)x(\lambda)d\lambda \quad (\text{B.1})$$

where $x(t)$ is the excitation function, and $h(t)$ is the transfer function approximated in the exponentially-decayed polynomial function (EDPF), that is

$$h(t) = \sum_{i=0}^n c_i \left(\frac{t}{d}\right)^i e^{-t/d} \quad (\text{B.2})$$

Let

$$w_i(t) = \int_0^t \left(\frac{t-\lambda}{d}\right)^i e^{-(t-\lambda)/d} x(\lambda) d\lambda \quad (\text{B.3})$$

then, Eq. (B.1) has the form:

$$y(t) = \left(\sum_{i=0}^n c_i \left(\frac{t}{d}\right)^i e^{-t/d} \right) * x(t) = \sum_{i=0}^n c_i w_i(t) \quad (\text{B.4})$$

Thus, the key point to efficiently compute $y(t)$ is how to update $w_i(t)$ at time $t + \Delta t$.

Let us look at the recursive convolution of Eq. (B.3) at time $t + \Delta t$

$$\begin{aligned} w_i(t + \Delta t) &= \int_0^{(t+\Delta t)} \left(\frac{t + \Delta t - \lambda}{d} \right)^i e^{-(t+\Delta t-\lambda)/d} x(\lambda) d\lambda \\ &= p_i(t + \Delta t) + q_i(t + \Delta t) \end{aligned} \quad (\text{B.5})$$

where

$$\begin{aligned} p_i(t + \Delta t) &= \int_0^t \left(\frac{(t + \Delta t) - \lambda}{d} \right)^i e^{-(t+\Delta t-\lambda)/d} x(\lambda) d\lambda \\ &= e^{-\Delta t/d} \int_0^t \left(\frac{\Delta t + t - \lambda}{d} \right)^i e^{-(t-\lambda)/d} x(\lambda) d\lambda \\ &= e^{-\Delta t/d} \int_0^t \left(\sum_{k=0}^i \binom{i}{k} \left(\frac{\Delta t}{d} \right)^{i-k} \left(\frac{t-\lambda}{d} \right)^k \right) e^{-(t-\lambda)/d} x(\lambda) d\lambda \\ &= e^{-\Delta t/d} \sum_{k=0}^i \binom{i}{k} \left(\frac{\Delta t}{d} \right)^{i-k} \int_0^t \left(\frac{t-\lambda}{d} \right)^k e^{-(t-\lambda)/d} x(\lambda) d\lambda \\ &= e^{-\Delta t/d} \sum_{k=0}^i \binom{i}{k} \left(\frac{\Delta t}{d} \right)^{i-k} w_k(t) \end{aligned} \quad (\text{B.6})$$

and

$$q_i(t + \Delta t) = \int_t^{(t+\Delta t)} \left(\frac{(t + \Delta t) - \lambda}{d} \right)^i e^{-(t+\Delta t-\lambda)/d} x(\lambda) d\lambda \quad (\text{B.7})$$

By using the Trapezoidal rule for the above integration, we have

$$q_i(t + \Delta t) = \begin{cases} \frac{\Delta t}{2} (e^{-\Delta t/d} x(t) + x(t + \Delta t)) & i = 0 \\ \frac{\Delta t}{2} \left(\frac{\Delta t}{d} \right)^i e^{-\Delta t/d} x(t) & i > 0 \end{cases} \quad (\text{B.8})$$

Now, combine Eqs. (B.4-B.8), we have the following recursive convolution

$$\begin{aligned} y(t + \Delta t) &= \sum_{i=0}^n c_i w_i(t) \\ &= \sum_{i=0}^n c_i (p_i(t + \Delta t) + q_i(t + \Delta t)) \\ &= \sum_{i=0}^n c_i \left(p_i(t + \Delta t) + \frac{\Delta t}{2} \left(\frac{\Delta t}{d} \right)^i e^{-\Delta t/d} x(t) \right) + c_0 \frac{\Delta t}{2} x(t + \Delta t) \\ &= \phi(t + \Delta t) + \rho(t + \Delta t) x(t + \Delta t) \end{aligned} \quad (\text{B.9})$$

where $\rho(t + \Delta t) = c_0 \Delta t / 2$ and $\phi(t + \Delta t)$ can be computed with the following recursive formulas:

$$\begin{aligned}\phi(t + \Delta t) &= \sum_{i=0}^n c_i \zeta_i(t + \Delta t) \\ \zeta_i(t + \Delta t) &= e^{-\Delta t/d} \left(\sum_{k=0}^i \binom{i}{k} \left(\frac{\Delta t}{d} \right)^{i-k} w_k(t) \right) + \frac{\Delta t}{2} \left(\frac{\Delta t}{d} \right)^i x(t) \\ w_k(t + \Delta t) &= \zeta_k(t + \Delta t) + \frac{\Delta t}{2} x(t + \Delta t)\end{aligned}\tag{B.10}$$

References

- [1] D. F. Anastasakis, N. Gopal, S.-Y. Kim and L. T. Pillage, “Enhancing the Stability of Asymptotic Waveform Evaluation for Digital Interconnect Circuit Applications,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 729-736, June 1994.
- [2] B. D. O. Anderson and S. Vongpanitlerd, Network Analysis and Synthesis: A Modern Systems Theory Approach, Englewood Cliffs, N.J., Prentice-Hall, 1973.
- [3] G. A. Baker, Jr. and J. L. Gammel, The Padé Approximant in Theoretical physics, New York, Academic Press, 1970.
- [4] G. A. Baker, Jr. and P. Graves-Morris, Padé Approximants, Part I: Basic Theory, Mass.: Addison-Wesley, 1981.
- [5] G. A. Baker, Jr. and P. Graves-Morris, Padé Approximants, Part II: Extensions and Applications, Mass.: Addison-Wesley, 1981.
- [6] N. Balabanian, T. A. Bickart, Electrical Network Theory, ch. 6, Malabar, Florida: Robert E. Krieger, 1985.
- [7] L. Barford, B. Troyanovsky, N. H. Chang, H. Liao and W. Dai, “System Identification Techniques for Fast, Recursive Convolution Based Circuit Simulation for Large RC Circuits,” *Proceedings of HP Design Technology Conference*, 1995.
- [8] A. Belantari, “Error Estimate in Vector Padé Approximation,” *Applied Numerical Mathematics*, pp. 457-468, 1991.

- [9] J. E. Bracken, V. Raghavan, and R. A. Rohrer, "Simulating Distributed Elements with Asymptotic Waveform Evaluation," *IEEE MTT-S International Microwave Symposium Digest*, pp. 1337-1340, 1992.
- [10] J. E. Bracken, V. Raghavan, and R. A. Rohrer, "Extension of the Asymptotic Waveform Evaluation Technique with the Method of Characteristics," Technical Digest of the *IEEE International Conference on Computer-Aided Design*, pp. 71-75, Nov. 1992.
- [11] C. Brezinski, "Error Estimate in Padé Approximation," *Proceedings of International Symposium on Orthogonal Polynomials and their Applications*, pp. 1-19, 1988.
- [12] C. Brezinski, "Procedures for Estimating the Error in Padé Approximation," *Mathematics of Computation*, pp. 639-648, Oct., 1989.
- [13] C. Brezinski, History of Continued Fractions and Padé Approximants, New York: Springer-Verlag, c1991.
- [14] H. Cabannes, Padé Approximants Method and Its Applications to Mechanics, New York: Springer-Verlag, 1976.
- [15] P. Chan, M. Schlag, "Bounds on Signal Delay in RC Mesh Networks," *IEEE Trans. on CAD*, Vol. CAD-8, no. 6, June 1989. pp. 581-589
- [16] P. Chan, "Comments on 'Asymptotic Waveform Evaluation for Timing Analysis'," *IEEE Trans. on CAD*, Aug. 1991, pp. 1078-1079.
- [17] F. Y. Chang, "Transient Analysis of Lossy Transmission Lines with Arbitrary Initial Potential and Current Distributions," *IEEE Trans. on Circuits and Systems*, vol. CAS-39, pp. 180-198, March 1991.
- [18] F. Y. Chang, "Waveform Relaxation Analysis of Nonuniform Lossy Transmission Lines Characterized with Frequency-Dependent Parameters," *IEEE Trans. on Circuits and Systems*, vol. CAS-38, pp. 1484-1500, Dec. 1991.
- [19] F. Y. Chang, "Transient Simulation of Nonuniform Coupled Lossy Transmission Lines Characterized with Frequency-Dependent Parameters, Part II: Discrete-Time Analysis," *IEEE Trans. on Circuits and Systems*, vol. CAS-39, pp. 907-927, Nov. 1992.

- [20] H. S. Chen and E.F. Thompson, Iterative and Padé Solutions for the Water-Wave Dispersion Relation, National Technical Information Service, 1985.
- [21] E. Chiprout and M. Nakhla, "Generalized Moment-matching Methods for Transient Analysis of Interconnect Networks," *29 th ACM/IEEE Design Automation Conference Proceedings*, 1992. pp. 201-206.
- [22] E. Chiprout and M.S. Nakhla, "Analysis of Interconnect Networks Using Complex Frequency Hopping (CFH)," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 186-200, Feb. 1995.
- [23] B. J. Cooke, J. L. Prince and A. C. Cangellaris, "S-Parameter Analysis of Multiconductor, Integrated Circuit Interconnect Systems," *IEEE Transaction on Computer-Aided Design*, vol. CAD-11(3), pp. 353-360, 1992.
- [24] J. K. Cullum and R. A. Willoughby, Lanczos algorithms for large symmetric eigenvalue computations, Boston: Birkhauser, 1985.
- [25] T. Dhaene, L. Martens, and D. De Zutter, "Transient Simulation of Arbitrary Nonuniform Interconnection Structures Characterized by Scattering Parameters," *IEEE Trans. on Circuits and Systems*, vol. CAS-39, pp. 928-937, Nov. 1992.
- [26] J. Dobrowolski, Introduction to Computer Methods for Microwave Circuit Analysis and Design, Artech House, 1991.
- [27] M. I. Elmasry, Digital MOS Integrated Circuits, New York: IEEE Press, pp. 4-27, 1981.
- [28] W. C. Elmore, "The Transient Response of Damped Linear Networks with Particular Regard to Wideband Amplifier," *J. Applied Physics*, 19(1), 1948.
- [29] P. Feldmann and R. W. Freund, "Efficient Linear Circuit Analysis by Padé Approximation via the Lanczos Process," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 639-649, May 1995.
- [30] P. Feldmann and R. W. Freund, "Reduced-Order Modeling of Large Linear Subcircuits via a Block Lanczos Algorithm," *Proceedings of the 32th ACM/IEEE Design Automation Conference*, June 1995.

- [31] C. M. Fiduccia and R. M. Mattheyses, "A Linear-Time Heuristic Improving Network Partitions," *Proceedings of the 19th Design Automation Conference*, pp. 241-247, 1982.
- [32] D. S. Gao, A. T. Yang, and S. M. Kang, "Accurate Modeling and Simulation of Parallel Interconnection in High-speed Integrated circuits," *IEEE Trans. on Circuits and Systems*, pp. 1-9, Jan. 1992.
- [33] K. Glover, "All Optimal Hankel-norm Approximations of Linear Multivariable Systems and Their L^∞ -error bounds," *Int. J. Control*, pp. 1115-1193, 1984.
- [34] K. C. Gupta, R. Grag, and R. Chadha, Computer Aided Design of Microwave Circuits, Dedham, MA: Artech 1981.
- [35] N. Hedenstierna, and K. O. Jeppson, "CMOS Circuit Speed and Buffer Optimization," *IEEE Trans. on CAD*, pp. 270-281, March 1987.
- [36] Hewlett-Packard. Application Note 117-1, 1969.
- [37] B. Hoppe, G. Neuendorf, D. Schmitt-Landsiedel, and W. Specks, "Optimization of High-Speed CMOS Logic Circuits with Analytical Models for Signal Delay, Chip Area, and Dynamic Power Dissipation," *IEEE Trans. on CAD*, pp. 236-246, March 1990.
- [38] C. Hwang and Y.-C. Lee, "Multifrequency Padé Approximation via Jordan Continued-Fraction Expansion," *IEEE Transactions on Automatic Control*, pp. 444-446, April 1989.
- [39] B. Johnson, T. Quarles, A. R. Newton, D. O. Pederson and A. Sangiovanni-Vincentelli, "SPICE3 Version 3e User's Manual," University of California, Berkeley, April 1991.
- [40] S. Karni, Network Theory: Analysis and Synthesis, Allyn and Bacon, Inc., 1966.
- [41] S.-Y. Kim, N. Gopal and L. T. Pillage, "Time-domain Macromodels for VLSI Interconnect Analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1257-70, Oct. 1994.
- [42] A. S. Kronrod, "Nodes and Weights of Quadrature Formulas," *Consultant Bureau*, New York, 1965.

- [43] N. Kuhn, "Simplified Signal Flow Graph Analysis," *Microwave J.*, VI, No. 11, pp. 59-66, 1963.
- [44] P. Lancaster, *Theory of Matrices*, New York, Academic Press, 1969.
- [45] C. Lanczos, "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators," *J. Res. Nat. Bur. Standards*, vol. 45, pp. 255-282, 1950.
- [46] H. Liao and W. Dai, "Wave Spreading Evaluation of Interconnect Systems," *Proceedings of the 3rd IEEE Multichip-module Conference*, pp. 128-133, 1993.
- [47] H. Liao and W. Dai, "Wave Spreading Evaluation of Interconnect Systems," Accepted by *IEEE Transactions on Microwave Theory and Techniques*.
- [48] H. Liao, R. Wang, R. Chandra, and W. Dai, "S-Parameter Based Macro Model of Distributed-Lumped Networks Using Padé Approximation," *IEEE International Symposium on Circuits and Systems*, pp. 2319-2322, May 1993.
- [49] H. Liao, W. Dai, R. Wang, and F. Y. Chang, "S-Parameter Based Macro Model of Distributed-Lumped Networks Using Exponentially Decayed Polynomial Function," *Proceedings of 30th ACM/IEEE Design Automation Conference*, pp. 726-731, June 1993.
- [50] H. Liao, W. Dai, "Scattering Parameter Transient Analysis of Interconnect Networks with Nonlinear Terminations Using Recursive Convolution," *Proceedings of the Synthesis and Simulation Meeting and International Interchange*, pp. 295-304, Oct. 1993.
- [51] H. Liao, R. Wang, and W. Dai, "Transient Analysis of Interconnects with Nonlinear Driver Using Mixed Exponential Function Approximation," *Technical Report of University of California at Santa Cruz*, UCSC-CRL-93-44, Oct. 1993.
- [52] H. Liao, and W. Dai, "Capturing Time-of-Flight Delay for Transient Analysis Based on Scattering Parameter Macromodel," *IEEE/ACM International Conference on Computer-Aided Design*, pp. 412-417, 1994.
- [53] T. Lin, C. Mead, "Signal Delay in General RC networks," *IEEE Trans. on CAD*, Vol. CAD-3, no. 4, pp. 331-349, Oct. 1984.

- [54] S. Lin, and E. S. Kuh, "Transient Simulation of Lossy Interconnects Based on the Recursive Convolution Formulation," *IEEE Trans. on Circuits and Systems*, vol. CAS-39, pp. 879-892, Nov. 1992.
- [55] M. D. Matson, "Macromodeling of Digital MOS VLSI Circuits," *Proceedings of 22th ACM/IEEE Design Automation Conference*, pp. 144-151, 1985.
- [56] M. D. Matson and L. A. Glasser, "Macromodeling and Optimization of Digital MOS VLSI Circuits," *IEEE Trans. on CAD*, Vol. CAD-5, no. 4, pp. 659-678, Oct. 1986.
- [57] V. A. Monaco, and P. Tiberio, "Automatic Scattering Matrix Computation of Microwave Circuits," *Alta Frequenza*, vol. 39, pp. 59-64, 1970.
- [58] R. W. Newcomb, Linear Multiport Synthesis, New York, McGraw-Hill, 1966.
- [59] P. R. O'Brien, T. Savarino, "Modeling the Driving-Point Characteristic of Resistive Interconnect for Accurate Delay Estimation," *Digest of Technical Papers, ICCAD-89*, Santa Clara, pp. 512-515.
- [60] L. T. Pillage, and R. A. Rohrer, "Asymptotic Waveform Evaluation for Timing Analysis," *IEEE Trans. on CAD*, vol. CAD-9, pp. 352-366, April 1990.
- [61] A. Pozzi, Application of Padé Approximation Theory in Fluid Dynamics, N.J.: World Scientific, 1994.
- [62] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, Numerical Recipes in C: The Art of Scientific Computing, Cambridge, June 1991.
- [63] V. Raghavan, and R. A. Rohrer, "A New Nonlinear Driver Model for Interconnect Analysis," *Proceedings of 28th ACM/IEEE Design Automation Conference*, pp. 561-566, 1991.
- [64] V. Raghavan, J. E. Bracken, and R. A. Rohrer, "AWESpice: A General Tool for the Efficient and Accurate Simulation of Interconnect Problems," *Proceedings of 29th ACM/IEEE Design Automation Conference*, pp. 87-92, June 1992.
- [65] C. Ratzlaff, N. Gopal, and L. Pillage, "RICE: Rapid Interconnect Circuit Evaluator," *28th ACM/IEEE Design Automation Conference Proceedings*, 1991. pp. 555-560.
- [66] J. C. Rautio, "Synthesis of Lumped Models from N-Port Scattering Parameter Data," *IEEE Transactions Microwave Theory and Techniques*, vol. 42, pp. 535-537, March 1994.

- [67] P. A. Rizzi, Microwave Engineering, Englewood Cliffs, NJ. Prentice-Hall, 1988.
- [68] J. K. Roberge, Operational Amplifiers: Theory and Practice, John Wiley & Sons Inc., pp. 530, 1975.
- [69] J. Rubinstein, P. Penfield, and M. Horowitz, "Signal Delay in RC Tree Networks," *IEEE Trans. On CAD*, Vol. CAD-2, no. 3, pp. 202-211, July 1983.
- [70] J. S. Roychowdury, and D. O. Pederson, "Efficient Transient Simulation of Lossy Interconnect," *Proceedings of 29th ACM/IEEE Design Automation Conference*, pp. 740-745, 1992.
- [71] J. Schoukens and R. Pintelon, Identification of Linear Systems: A Practical Guideline to Accurate Modeling, New York: Pergamon Press, 1991.
- [72] J. E. Schutt-Aine, and R. Mittra, "Scattering Parameter Transient Analysis of Transmission Lines Loaded with Nonlinear Terminations," *IEEE Trans. on Microwave Theory Tech.*, vol. MTT-36, pp. 529-535, March 1988.
- [73] A. Semlyen, and A. Dabuleanu, "Fast and Accurate Switching Transient Calculation on Transmission Lines with Ground Return Using Recursive Convolution," *IEEE Trans. on Power Apparatus and Systems*, vol. PAS-94, pp. 561-571, 1975.
- [74] D. E. Stewart and A. Leyk, Meschach Library, Internet, April, 1994.
- [75] Y. Sun and W. Dai, "Sensitivity Analysis of Interconnect Networks Based on Scattering Parameter Macromodel," *Proceedings of the IEEE Multichip-module Conference*, pp. 87-92, 1995.
- [76] H. J. M. Veendrick, "Short-Circuit Dissipation of Static CMOS Circuitry and Its Impact on the Design of Buffer Circuits," *IEEE Journal of Solid-State Circuits*, Vol. SC-19, No. 4, pp. 468-473, Aug. 1984.
- [77] K. Waranica, Solution of Nonlinear Differential Equations by Means of the Padé Approximation, dissertation, 1965.
- [78] F. L. Warner, Microwave Attenuation Measurement, Peter Perefrinus, 1977. pp. 276.
- [79] N. Weste, and K. Eshraghian, Principles of CMOS VLSI Design: A systems Perspective, Chapter 2, Addison-Wesley, 1993.

- [80] J. White, L. Pillage, and J. Cohn, "Computer-Aided Analysis and Design of Interconnect" *Tutorial of IEEE/ACM International Conference on Computer-Aided Design*, Nov. 1993.
- [81] D. Winklestein, M. B. Steer, and R. Pomerleau, "Simulation of Arbitrary Transmission Line Networks with Nonlinear Terminations," *IEEE Trans. on Circuits and Systems*, vol. CAS-38, pp. 418-422, April 1991.
- [82] A. T. Yang, J. Yao, et al, User's Manual of MISIM, 1992.
- [83] I. Young, L. Pillage and J. Cohn, "Digital Circuit Interconnect: Issues, Models, Analysis and Design," *Tutorial of IEEE/ACM International Conference on Computer-Aided-Design*, Nov. 1994.