

## INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

**The quality of this reproduction is dependent upon the quality of the copy submitted.** Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

# UMI

A Bell & Howell Information Company  
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA  
313/761-4700 800/521-0600



**Automatic Measurement-Based Characterization of Off-Chip  
Interconnect Circuitry using Lumped Elements**

**by  
Steven D. Corey**

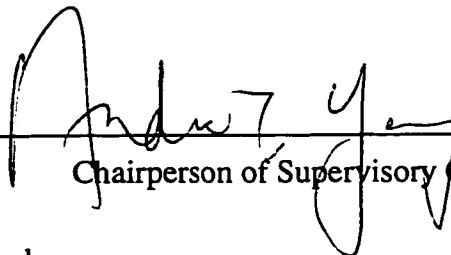
A dissertation submitted in partial fulfillment  
of the requirements for the degree of

Doctor of Philosophy

University of Washington

1997

Approved by



Chairperson of Supervisory Committee

Program Authorized  
to Offer Degree

Electrical Engineering

Date

10/23/97

**UMI Number: 9819222**

**Copyright 1997 by  
Corey, Steven David**

**All rights reserved.**

---

**UMI Microform 9819222  
Copyright 1998, by UMI Company. All rights reserved.**

**This microform edition is protected against unauthorized  
copying under Title 17, United States Code.**

---

**UMI**  
**300 North Zeeb Road**  
**Ann Arbor, MI 48103**

© Copyright 1997

Steven D. Corey

In presenting this dissertation in partial fulfillment of the requirements for a Doctoral degree at the University of Washington, I agree that the Library shall make its copies freely available for inspection. I further agree that extensive copying of this dissertation is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for copying or reproduction of this dissertation may be referred to University Microfilms, 1490 Eisenhower Place, P.O. Box 975, Ann Arbor, MI 48106, to whom the author has granted "the right to reproduce and sell (a) copies of the manuscript in microform and/or (b) printed copies of the manuscript made from microform."

Signature Steven D. Grey  
Date 10/23/97

University of Washington

Abstract

Automatic Measurement-Based Characterization of Off-Chip  
Interconnect Circuitry using Lumped Elements

by Steven D. Corey

Chairperson of the Supervisory Committee  
Professor Andrew T. Yang  
Department of Electrical Engineering

As clock speeds of digital circuits continue to increase, the performance of board-level and package-level circuitry has become an important factor in digital design. The same is true in the consumer wireless communications market, in which low-cost designs necessitate the use of low-performance interconnect solutions. In order to consider the effects of interconnect circuitry in the design stage of a circuit or system, interconnections must first be effectively modeled for simulation.

Many different interconnect modeling solutions have been proposed and implemented in the history of circuit design. Key issues faced in developing a modeling approach are the compatibility of the model with existing simulation tools, the stability of the model, the passivity of the model, and the ability of the modeling approach extract as simple a model as is acceptable for a given application. Furthermore, as is requisite in any CAD (computer aided design) application, the approach must be reliable and predictable.

In this dissertation a robust modeling approach is presented which extracts a lumped-element model from measured time-domain data. The extracted model is in SPICE-netlist format, which ensures compatibility with the large majority of existing industry tools. The model is shown heuristically to be asymptotically stable, and to a certain degree to be absolutely stable. The modeling approach presented includes the ability to reduce the size of the model for use in a frequency range narrower than that from which it was extracted. Extraction examples are given of typical interconnect structures to verify the accuracy and practicality of the approach.

# Table Of Contents

	<i>Page</i>
List of Figures .....	iv
Chapter 1: MOTIVATION FOR OFF-CHIP INTERCONNECT MODELING .....	1
Chapter 2: REVIEW OF EXISTING INTERCONNECT MODELING APPROACHES .....	10
2.1 Criteria for Evaluation of Existing Interconnect Modeling Solutions .....	12
2.1.1 Asymptotic Stability .....	13
2.1.2 Accuracy/Complexity Tradeoff .....	14
2.1.3 Compatibility with Existing Simulation Tools .....	16
2.1.4 Absolute Stability .....	17
2.2 Modeling Approaches in the Literature .....	17
2.2.1 Lumped Models .....	17
2.2.2 Distributed Models .....	19
2.2.3 Measurement-Based Models and Techniques .....	20
2.3 Overview of the Model to be Presented .....	22
Chapter 3: OVERVIEW OF THE PROPOSED INTERCONNECT MODELING APPROACH .....	23
3.1 Measurement-Based/Lumped-Element Combination .....	24
3.2 Automatic Extraction .....	27
3.3 A General Model .....	29
3.4 Asymptotic Stability .....	32
3.5 Absolute Stability .....	33
3.6 Conclusion .....	33
Chapter 4: REVIEW OF PERTINENT CIRCUIT AND SYSTEM THEORY .....	35
4.1 System Theory Overview .....	35
4.2 Linear Time-Invariant Multiport Networks .....	37
4.3 State-Space Representations of Continuous-Time Systems .....	40
4.4 Sampled-Time Representation of Continuous-Time Systems .....	43
4.5 System Impulse Response .....	45



4.6	Frequency-Domain Representation of Continuous-Time Systems. . . . .	46
4.7	Variable Definitions for the Modeling Problem at Hand . . . . .	49
4.8	Basis Transformations . . . . .	51
4.9	Summary . . . . .	53
Chapter 5: AN ALGORITHM FOR ROBUST INTERCONNECT MODEL EXTRACTION FROM TIME-DOMAIN MEASUREMENTS. . . . .		55
5.1	A Robust Algorithm . . . . .	56
5.2	Problem Formulation . . . . .	57
5.3	Choice of Internal Waveforms . . . . .	60
5.4	Internal Waveform Subset Selection . . . . .	63
5.5	Final Identification of the Sampled Model—Determination of Q and C . . . . .	68
5.6	Transformation to Continuous Time. . . . .	70
5.7	Post-processing . . . . .	74
5.7.1	Further Reduction (Complexity vs. Accuracy). . . . .	74
5.7.2	Conversion to Admittance (Compatibility) . . . . .	74
5.7.3	Branch Reduction . . . . .	77
5.7.4	Conversion to a SPICE netlist . . . . .	78
5.7.5	MNA Stamping Procedure . . . . .	78
5.7.6	Unstamping Procedure . . . . .	80
5.8	Conclusion . . . . .	81
Chapter 6: IMPLEMENTATION OF THE INTERCONNECT MODELING SYSTEM. . . . .		83
6.1	Choice of Computer Language. . . . .	84
6.2	Organization of the Interconnect Modeling Framework . . . . .	86
6.2.1	Inputs. . . . .	86
6.2.2	Pencil Preconditioning . . . . .	88
6.2.3	Conversion to Continuous Time. . . . .	89
6.2.4	Network Reduction. . . . .	90
6.2.5	Branch Reduction and Conversion to Admittance . . . . .	90
6.2.6	Outputs . . . . .	91
6.3	Computational Complexity. . . . .	92
6.4	Summary . . . . .	94
Chapter 7: CHARACTERIZATION EXAMPLES. . . . .		95
7.1	Measurement Setup and General Modeling Approach. . . . .	95

7.2 Two-port Microstrip Circuit . . . . .	98
7.3 Four-port Microstrip Circuit . . . . .	105
7.4 Lossy MCM Line Characterization . . . . .	109
7.5 Conclusion . . . . .	112
 Chapter 8 CONCLUSION. . . . .	 113
 Bibliography . . . . .	 117
 Appendix A: Underlying Circuit Theory of Multiport Measurements . . . . .	 124
 Appendix B: Fixturing Issues in High-frequency Measurements . . . . .	 128
 Appendix C: De-embedding High-Frequency Measurements . . . . .	 132

# List of Figures

<i>Number</i>	<i>Page</i>
1.1 Examples of off-chip interconnect circuitry . . . . .	2
1.2 Scattering responses to different incident wavelengths . . . . .	4
1.3 Time-domain and frequency-domain representations of a typical digital signal. . . . .	5
1.4 Ideal transmission line response to different input risetimes. . . . .	6
4.1 Pictorial representation of a system. . . . .	38
4.2 Examples of accurately-sampled and undersampled waveforms. . . . .	44
5.1 Transformer for simulation of a scattering-based model in an admittance-based simulator. . . . .	76
5.2 Conductor $g_{ij}$ to be inserted into global circuit matrix $\mathbf{G}$ . . . . .	79
6.1 Flow diagram of the interconnect modeling system as implemented. . . . .	87
7.1 General measurement setup employed. . . . .	96
7.2 Two-port microstrip example circuit. . . . .	99
7.3 Idealized setup for taking measurements at port 1 of the two-port example circuits from Sections 7.2 and 7.4. . . . .	99
7.4 Measured excitation step waveform and step response waveforms for the circuit in Figure 7.2. . . . .	101
7.5 Comparison between the excitation step used in simulation and that used in taking the measurements. . . . .	102
7.6 Comparison of model response with matched-load measured data used to extract the model originally. . . . .	103
7.7 Comparison of modeled and measured data for two-port example. . . . .	104
7.8 Evaluation of the model of the circuit in Figure 7.2. . . . .	106
7.9 Four-port microstrip example circuit. . . . .	107
7.10 Idealized setup for taking measurements at port 1 of the four-port circuit in Figure 7.9. . . . .	107

7.11 Simulated and measured open-circuit reflection at port 1 and crosstalk at port 4 for the circuit in Figure 7.9. ....	108
7.12 Frequency-domain reflection magnitude comparison of measurement and the 100 MHz, 1GHz, and 10 GHz models of the circuit in Figure 7.9. ....	109
7.13 Simulated and measured waveforms for matched reflection and matched transmission for unreduced model.....	110
7.14 Comparison of open-circuit reflection waveforms for unreduced model.....	111
7.15 Magnitude plot of $S_{11}$ for measured data as well as for 1 GHz, 5 GHz, and 20 GHz reduced models.....	112

# Acknowledgement

Completion of this research has been the result of a group effort, and thanks are in order to many people who have made significant contributions. Foremost, I express my appreciation to my faculty advisor, Andrew Yang, who above all else has taught and encouraged me by word and example to aim high and to never settle for mediocre or incremental contributions. Of those fellow students in the simulation/modeling research group, I specifically thank Kevin Kerns for many enlightening and helpful discussions regarding linear systems and numerical methods, and their application to the interconnect modeling problem. Of equal importance was the guidance and stabilizing influence of Ivan Wemple, and his generous dispensation of carefully considered advice borne out of experience. And to *all* simulation/modeling lab members, thanks for your friendship and encouragement throughout my tenure.

A word of appreciation is also in order to Tektronix, Inc., for directly supporting a portion of this work financially, and for donating the hardware and software necessary to institute a *bona fide* measurement-based research effort. I specifically thank Stan Kaveckis and others at Tektronix who have lent their support to my work throughout its course.

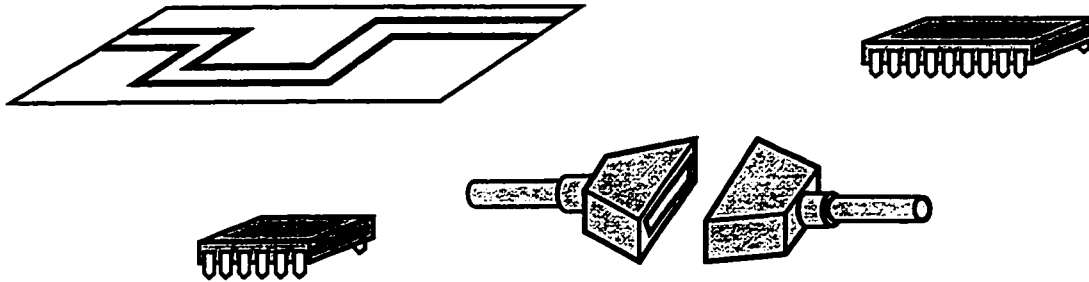
Finally, I would like to express my gratitude to family and friends who have been a continuous source of strength and encouragement over the course of this effort. And I am particularly grateful to those who have tactfully avoided the question, “Are you ever going to finish?”

# Chapter 1

## *MOTIVATION FOR OFF-CHIP INTERCONNECT MODELING*

Off-chip interconnect modeling has migrated to the forefront in electronic circuit and system design because of the need to perform system-level verification in the face of increasing circuit speed and complexity. System-level verification is the process by which a designer ensures that an entire electronic system, composed of various hierarchies of subcircuits, will function as desired when the various subcomponents are electrically and mechanically connected. Those interconnections which are not supported directly by the silicon substrate itself are considered to be off chip, such as the examples shown in Figure 1.1.

A typical example of system-level verification occurs in the design of a computer workstation at the motherboard level. Various subcomponents of the system may include a central processing unit, cache memory, random access memory, hard disk controller, floppy disk controller, video card, modem card, etc. Furthermore, these components have quite likely been purchased from several different vendors to optimize some cost-benefit figure for the motherboard manufacturer. Each vendor has guaranteed that its circuit functions according to a certain set of specifications. A designer who would naively lay out the board such that area (for example) is minimized would likely discover that although



**Figure 1.1** Examples of off-chip interconnect circuitry: circuitboard traces, packages, and connectors.

every output of every component was connected to the proper corresponding input of every other component, the system would not operate. That is, perhaps the data lines were laid too close to each other, allowing crosstalk between adjacent lines. Or perhaps the traces are too narrow, causing a resistive drop during current flow. Another possibility is that a SCSI connector enroute to the disk drive is causing reflections large enough to trip the disk controller. This example illustrates typical problems encountered by a signal integrity engineer, whose job is to perform system-level verification.

However, the signal integrity engineer is merely the first of a number of people affected by off-chip interconnect parasitics. Although he or she may have to shoulder the task of redesigning the board, the company will have to foot the bill for the redesign. If the system design parameters are relatively flexible, it may be possible to change the layout of the board to reduce parasitics at the expense of area, for example. However, in other situations it may be necessary to find different components which are less susceptible to noise, or to reduce the clock speed on the board to reduce the effects of parasitics. Any of these options will increase the development cost of the system and possibly reduce its performance, which in turn erodes profits.

More often, the increase in development cost caused by an additional design iteration is overshadowed by the expense of the associated time-to-market delay. Because of rapid advances in technology, a few months' delay in releasing a product can be the deter-

mining factor with respect to its profitability. A worst-case scenario could even result in the cancellation of the product. In contrast, accurate interconnect modeling will often reduce time-to-market significantly, with the associated increase in profits and market share.

Another advantage of accurate interconnect modeling is that it helps catch designs which are at the edge of failure. In the example above, the prototype board may operate, but the statistical variations encountered in manufacturing the system cause a significant percentage of boards to fail specifications at the factory. Such designs are not only responsible for low manufacturing yield, which reduces profit margins, but they can also lead to unreliability in the shipped product, eroding customer confidence. Without accurate modeling, the designer is forced to make a worst-case design, likely exchanging performance for reliability and manufacturability.

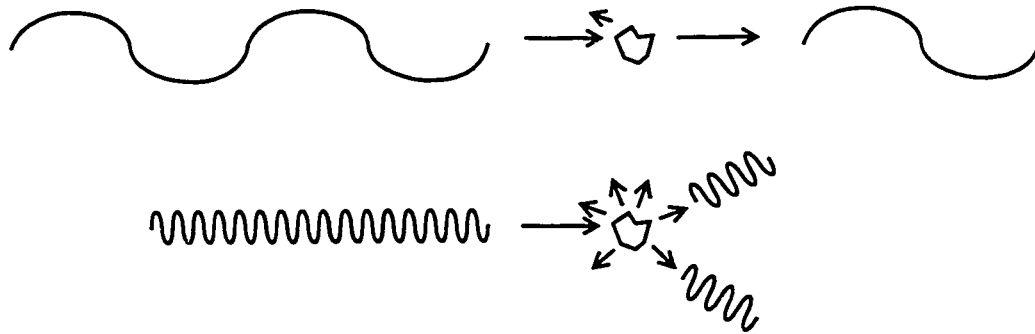
The preceding example illustrates the advantages that accurate interconnect modeling offers to system-level design. The reader may then wonder why this issue is only lately coming to the forefront in electronic design, and why it has not been an issue in the past. The answer was alluded to earlier in this section: the signal integrity engineer must perform system-level verification *in the face of increasing circuit speed and complexity*. The high circuit speeds and system complexities which are commonplace today have brought the issue of interconnect modeling to the forefront, such that it is now a significant bottleneck in circuit design.

Increased *digital* circuit speed is the primary reason why interconnect modeling has moved into prominence. The importance of increased digital speeds is due to the principles of reflection and transmission as they are embodied in Maxwell's equations, which govern the interaction of electromagnetic waves and matter [1][2]. Electromagnetic radiation of a particular frequency has an associated wavelength given by

$$\lambda = \frac{v}{f} \quad (1.1)$$



where  $v$  is the velocity of the wave in the given medium.<sup>†</sup> If the wavelength associated with an electrical signal is large compared to the size of the object encountered, that object may be treated as a point—its spatial variations are ignored. At this juncture in our discussion, it is useful to introduce the concept of electrical length, which refers loosely to how large an object appears to an electromagnetic wave inside the object. An object is considered electrically large if multiple wavelengths fit inside of it, and electrically small if only a small fraction of a wavelength fits inside of it. Given these definitions, it can be seen from (1.1) that as frequency increases, wavelength decreases, and therefore the various scattering objects which were once small compared to the wavelength (electrically small) can become comparable in size (electrically large), and therefore must be modeled as such. This concept is illustrated rather simplistically in Figure 1.2, where it is seen that



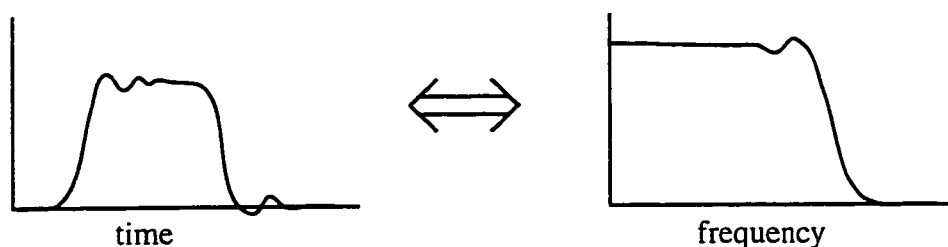
**Figure 1.2** Scattering responses in which the excitation wavelength is large compared to the scatterer (above) and small compared to the scatterer (below). In the latter case, the reflection signature is more complex.

when the wavelength is large compared to the scattering object, reflections are simple and the majority of the energy continues propagating largely unaffected, which implies that only a very simple model is necessary, if any. When the wavelength decreases, however,

<sup>†</sup> It is not uncommon for velocity to vary as a function of frequency, but generally as a second-order effect in interconnect modeling.

the same scatterer causes a more complicated reflection signature, due in part to the resonant modes which are set up internal to the scatterer. This typically requires a more complex model.

The preceding explanation was performed in the frequency domain, since concepts of frequency and wavelength are most intuitive in that context. However, since we are currently referring to the effects of interconnect parasitics on *digital* circuitry, a time domain discussion is in order. Digital signals are not generally monochromatic, and therefore have a large distribution of wavelengths present from DC up to some frequency determined by signal risetime and shape. The transition of such a signal between its high and low states can be represented by Fourier components, as depicted in Figure 1.3, and the



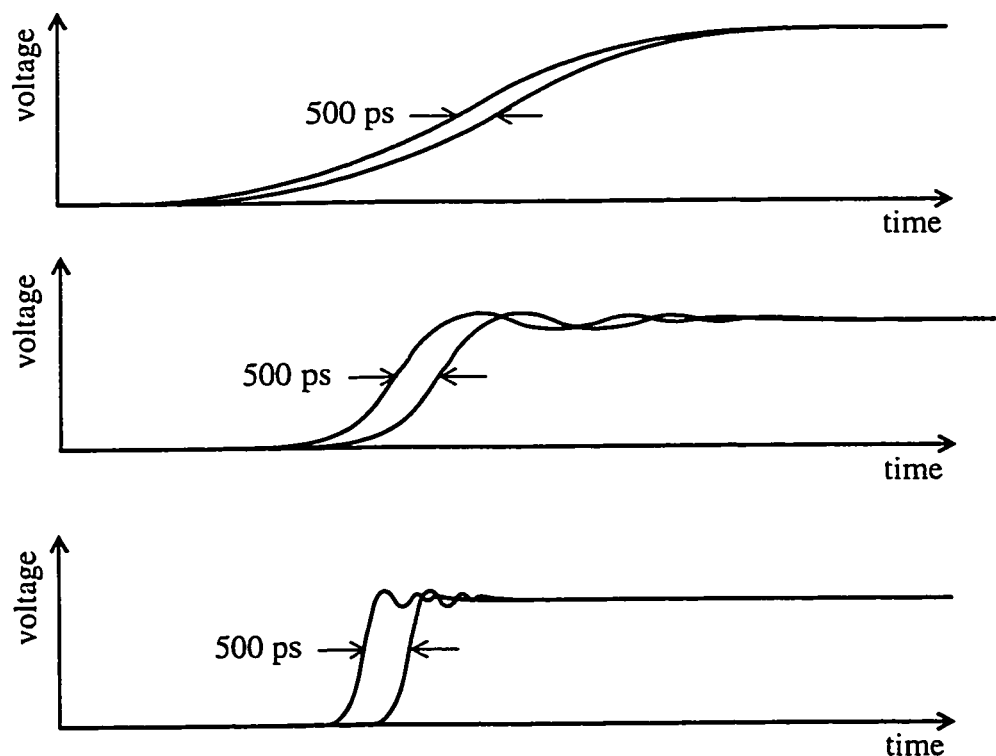
**Figure 1.3** Time-domain (left) and frequency-domain (right) representations of a typical digital signal.

arguments of the preceding paragraph may be applied to each of these components individually. (Of course, only those components which correspond to the smallest wavelength, and therefore the highest frequency, are of import to this discussion.) However, it is possibly more intuitive to develop a time-domain version of the concept of electrical length. In the time domain, electrical length refers loosely to the amount of time required for a signal to get past a scattering object.<sup>†</sup> Given this definition, it may be succinctly

<sup>†</sup> In this case, the term length is used temporally rather than spatially, to refer to a length of *time*.

stated that if the electrical length of an object is small compared to the risetime of the incident signal, the electrical contribution of the object may be ignored. On the other hand, if the electrical length of the scatterer is on the order of the signal risetime, it must be modeled, but it is unnecessary to account for its detailed spatial structure. Finally, if the signal encounters a barrier whose electrical length is significantly larger than the signal risetime, the barrier must be modeled with a certain amount of its fine spatial structure taken into account.

As an example, consider a transmission line of electrical length 500 ps, which would typically correspond to a physical length of about three inches, depending on the dielectric constant of the circuitboard. Several input and output signal pairs are shown in Figure 1.4.



**Figure 1.4** Three input/output signal pairs for an ideal 500 ps transmission line. Signal risetimes from top to bottom are 5 ns, 1 ns, and 200 ps. With decreasing risetime, the distortion due to the transmission line is seen to increase relative to the input signal.

At a risetime of 5 ns, the additional delay will go almost unnoticed, as it is a mere ten percent of the risetime itself. Given a step with a 1 ns risetime, the delay is noticeable, and must be taken into account, at least qualitatively. However, a 200 ps step is dominated by the delay of the transmission line, which must now be modeled to within perhaps ten percent so as not to incur a large amount of error. This simple example serves to emphasize that as risetimes of digital signals continue to decrease, the latter regime, as exemplified by the 200 ps signal, will continue to become more common, causing interconnect parasitics play an increasingly important role in circuit operation. When more realistic interconnect parts such as packages and connectors are added to the slurry, reflection, dispersion, and crosstalk are just as important as the ideal delay of the simple transmission line example above. In these situations accurate models are just as important, but are significantly more difficult to obtain by conventional methods.

Concurrent with the increase in digital system speeds has been increasing complexity in digital and mixed-mode systems. Complexity refers to the number of discrete elements (*i.e. transistors*) in a circuit, or perhaps the number of input/output (I/O) lines, or signal lines, in the circuit. Higher complexity has manifested itself in additional circuit-board layers, narrower and closer wire traces, and more crossovers and vias. Each of these introduces additional nonidealities in the interconnect circuitry which must be effectively modeled and simulated to achieve a robust system design. The coincident push toward miniaturization of such systems has caused the large number of associated interconnects to be compressed into increasingly smaller spaces, exacerbating the complexity problem. This is particularly evident in the multichip module (MCM), which shrinks the area occupied by packaging and wire traces by significantly reducing their sizes, will further increase the challenges presented to the signal integrity engineer.

Another reason for the increased need for interconnect modeling has been the advent of consumer radio frequency (RF) products. Past technologies limited mobile wireless communication products to primarily police and military applications, because of prohibitive expense associated with the design and manufacture of such systems under early technologies. However, the miniaturization of oscillators and phase-locked loops, as well as

advances in on-chip signal processing and low-power design, have brought wireless communication within reach of consumer markets. Predictably, the dramatic increase in volume has been accompanied by significantly reduced profit margins, requiring a corresponding reduction in time to market. (These two factors, price and time to market, were less important in the original military and police markets.) The reduced margins have required designers to utilize less expensive packaging and circuit board technologies, increasing the parasitics associated with each. Since accurate interconnect modeling is an effective way to decrease time-to-market and improve the reliability of a design, it is an important component in consumer RF design.

Electrical modeling of off-chip interconnect circuitry has rapidly come into a position of prominence in digital and mixed-mode system design. Responsibility for this change has been assigned to various market factors such as increased system clock speeds, increased system complexities, and the advent of consumer wireless communication. These factors will in all likelihood continue on their present trends toward even higher circuit speeds and complexities, and further development and availability of consumer RF applications. For example, computers with clock frequencies in the range of 1-4 GHz are already being realistically discussed, and those at the lower end of this spectrum are already in development. An interesting stated goal of one electronics company is to develop self-contained wireless communication technology the size of a credit card. As these types of applications are developed, the need for accurate interconnect modeling will continue to increase.

Although the interconnect modeling issue has become more recognized in recent years, insufficient tools and knowledge are available for the electronic design industry to effectively deal with the problem. The purpose of this work is to present a practical algorithm which can be used in an industrial setting to extract measurement-based simulation models of typical interconnect parts. However, the term "practical algorithm" is an ambiguous one. In the opinion of the author, it is impossible to definitively determine the practicality of an algorithm without implementing it in software and testing it on a variety of industry examples. To the extent possible, that was accomplished in the course of this

research, and will be presented in this dissertation. The following chapter will review existing approaches to interconnect modeling, in terms of types of models used, algorithms employed in their extraction, and their overall usefulness to the electronic design industry. The remainder of the dissertation will be concerned with the mathematical, computational, and empirical details involved in the derivation, implementation, and verification of a practical algorithm for extraction of interconnect models.

## Chapter 2

### *REVIEW OF EXISTING INTERCONNECT MODELING APPROACHES*

Interconnect modeling has been a subject of interest to electrical engineers for quite some time, certainly since wireless communication was initiated around the turn of the century. Since that time, radio frequency (RF) designers have concerned themselves with interconnect issues in the context of properly matching impedances at high frequency, since the wavelengths associated with such signals are equivalent in size to the circuit elements being used. This includes not only matching input signal lines to amplifiers, but also matching amplifier outputs to antennas. Transmission line structures are often incorporated as part of an RF design, but are typically assumed lossless for hand analysis. Since computer simulation is necessary to include anything more than the simplest of lossy transmission line characteristics, accurate modeling of such lines for simulation has been an important issue in RF design. It should be noted, however, that RF design is generally narrowband in nature, which greatly simplifies the modeling and simulation issues. Digital systems, in contrast, are inherently broadband.

Another branch of electrical engineering which has historically dealt with transmission line issues is power engineering. According to (1.1) the wavelength associated with a 60 Hz signal in free air is 5000 km. Although such distances seem large compared to

wavelengths associated with microwave applications, this order of magnitude is routinely experienced in intracontinental power distribution systems. As in RF design, reflections and delays associated with lossy transmission lines must be modeled for simulation, and the application is once again narrowband in nature.

As stated previously, interconnect modeling has only recently become an issue in *digital* circuit design. The age of digital computers is generally said to have commenced in 1946 with the development of the ENIAC by the US Army for the solution of artillery trajectory problems [3]. The only interconnect issue of concern to its designers was DC resistance, which was typically low enough to be negligible. The computer itself was 100 feet long, however, requiring a reasonable gauge of wire to support signals. Despite the considerable length of such interconnect runs, the resultant delay was dwarfed by the time constants associated with the various memory devices. The introduction of the monolithic transistor and the subsequent development of transistor-based memory circuits dramatically decreased the time required to access primary storage, or random-access memory. As a byproduct of these truly revolutionary changes, circuitboard delay was inaugurated as the first significant effect of interconnect circuitry in digital design, since the time required for a signal to travel to and from main memory was now on the same order as the time spent to fetch or write the data. However, since the delay was ideal, it could be accurately handled by a digital timing simulator, and this continued to be the case for most digital systems well into the 1980's. Even this delay was often dominated by on-chip delays across nonlinear devices in the central processing unit (CPU), which held CPU clock speeds to the low megahertz range.

However, current clock speeds in excess of 100 MHz have caused risetimes to dip below 1 ns, allowing reflection, crosstalk, and risetime degradation to join delay as significant factors at the board and package levels. At the same time increased demand for high-speed networking capabilities, as well as dramatic speed increases in secondary storage devices such as magnetic disk drives, have required significantly higher data rates of transmission cables and associated connectors. At these increased data rates, the behavior of such devices becomes less and less ideal, which significantly complicates circuit



design. Since these effects have only recently become problematic in broadband applications such as high-speed digital circuit design, there is not only a need for study and understanding of the interconnect parasitic problem itself—there is also significant space for innovation and improvement in the modeling and simulation of such systems. This chapter will begin by introducing a set of criteria for evaluation of a general interconnect modeling solution. Current theoretical approaches to the problem will be evaluated in light of these criteria, and the approach to be proposed in this dissertation will be cursorily evaluated as well. A more in-depth evaluation of the approach, however, will be reserved until the following chapter.

## *2.1 Criteria for Evaluation of Existing Interconnect Modeling Solutions*

In order to systematically review various approaches to interconnect modeling, it is useful to first develop a set of criteria for comparison. In this review, the criteria will be stated as a set of desired characteristics of a general, robust, and practically useful modeling solution. The following four aspects have been identified and selected as the basis for comparison:

1. Asymptotic Stability
2. Accuracy/Complexity tradeoff
3. Compatibility with Existing Simulation Tools
4. Absolute Stability

These four characteristics do not necessarily span the entire space of interconnect modeling, nor will they elucidate every strong point or every weak point of every modeling solution. However, they represent the major hurdles faced by anyone attempting to develop a general modeling solution, since they are broad enough to apply to any modeling approach. Each is described in greater detail in the following subsections.

### *2.1.1 Asymptotic Stability*

If a network is asymptotically stable for a specific set of terminations<sup>†</sup>, its output will not grow without bound in response to a bounded input.<sup>‡</sup> Any system which is not asymptotically stable for the set of terminations to which it will be subjected during simulation is

useless, since the output of such a circuit will generally grow without bound during time-domain simulation until the integration scheme breaks down. If a circuit is stable and accurate in the termination regimes to which it will be subjected during simulation, it is useful as a model. However, such a property can be difficult to guarantee, especially if the model is meant to be a general one for use with many different terminations. This is because the termination scheme includes the not only any linear or nonlinear impedances present at the ports, but injected voltages and currents as well. Each of these factors is highly independent of the model itself, and for the most part unknown prior to simulation, which is when the model is extracted.

It should be noted that a weaker definition of asymptotic stability is employed at times. This form refers only to a single termination scheme, namely no termination at all. Under this definition, a network is said to be asymptotically stable if it does not blow up when a Dirac delta function is placed as the input variable at one of the ports while all other port input variables are set to zero. Examples would include an impedance network driven at one port with a delta function of current while all other ports are open circuited. Predictably, the dual consists of an admittance network driven at one port with a delta function of voltage while all other ports are short circuited. Asymptotic stability under this termination regime (*i.e.*, no termination) is synonymous with requiring that the network have no poles with positive real parts, since positive poles correspond directly to exponential growth in the time domain. If this termination scheme describes the complete set of conditions under which the model is to be driven during usage, such a property (that is, no positive poles) is valuable. However, this is typically not the case, since most models will be used under more arbitrary terminations, and for this reason having all poles with negative real parts is not generally a strong enough condition to ensure accurate simulation.

---

<sup>†</sup> Note that the definition of a termination is general enough to include any device which is connected to the ports, *i.e.* linear elements, nonlinear elements, power supplies, etc.

<sup>‡</sup> Typically a Dirac delta function is used as excitation, since it contains significant signal energy at every frequency.

It may also be argued that if the termination scheme to be used is close to one under which the model is known to be asymptotically stable, then the simulation will not likely suffer from instability. For example, an asymptotically stable impedance network should be stable when driving a high-impedance input buffer. However, this is an entirely heuristic argument, especially if the buffer impedance is nonlinear. If simulation results can be closely monitored at least at the ports, asymptotic instability can be caught during simulation and perhaps the model can be re-extracted or tweaked. However, this precludes use of the model in system which is too large for all simulation results to be specifically monitored.

### *2.1.2 Accuracy/Complexity Tradeoff*

A general-purpose modeling technique should have the ability to vary accuracy by inversely varying model complexity. Since engineering must in many cases be governed by pragmatism, the question “why worry about what will never occur?” arises quite naturally when modeling a system. In simulation, a model which predicts more information than necessary is often more difficult to extract and takes longer to evaluate during simulation. And in reality, the extraneous information in the model will never be accessed. However, to effectively trade off accuracy and complexity, the modeler must be able to knowledgeably determine what behavior must necessarily be predicted, and what behavior is unnecessary to predict.

In interconnect modeling it is natural to present the tradeoff between accuracy and complexity of a model as a per cent deviation over a frequency range. This is especially true since interconnect circuits are generally linear for the range of signals they carry, which implies that signal energy present in the system at one frequency will never “leak” or transform into signal energy at another frequency.<sup>†</sup> That is, the circuit input/output relationships at a given frequency are proportional, and linearity furthermore implies that

---

<sup>†</sup> Of course, this is only true of the interconnections themselves, and generally untrue of the nonlinear digital circuits which they connect. One need look no further than the digital comparator to see energy converted from near DC into the high megahertz range.

the proportionality is independent of the actual amplitudes of the signals. The only variation of the transfer function of such a network is with respect to frequency. Since all real-life signals are band limited, there is an infinite range of input-signal frequencies to which the model will never be subjected. For this reason, it is useful to tailor the model to be valid over a finite bandwidth which contains the frequency spectrum of the signals that the model will encounter.

While in the previous subsection it was noted that not all input vectors or termination schemes are generally known prior to simulation, it is typical to have approximate knowledge prior to simulation of the frequency content to which the model will be subjected. In analog circuit design, such information is directly available from the specifications of the device to be designed. In digital design, it must be derived from specification information such as signal risetimes and shapes, and often from knowledge of the output characteristics of the various output buffers to which it will be connected. Although determination of an effective frequency range in digital design is not as precise as in analog design, it is quite possible to choose a cutoff frequency above which no significant energy is present in the system.

For a qualitative illustration of how the level of necessary model complexity depends on the frequencies of the input signal, the reader is referred to the transmission-line example in Chapter 1, illustrated in Figure 1.4. It was determined in that example that the device had a negligible effect on the 5 ns input signal—that is, the frequency content of the input signal was low enough that the device looked very similar to a short circuit between its input and output nodes. In such a case, a very simple model will suffice for the device. However, for the faster input signals, the effects of the line are more noticeable. Alternately stated, since the input signal contains higher frequency content, the corresponding higher-frequency behavior of the device must be modeled, which implies a more complex model. Quantitative illustrations of this concept are presented in Chapter 6.

### 2.1.3 Compatibility with Existing Simulation Tools

Much of digital and analog simulation is done using SPICE-based simulators, which have been a de facto industry standard since the adoption of SPICE (Simulation Program with Integrated Circuit Emphasis), and particularly SPICE2, which came from the University of California at Berkeley in 1975.<sup>†</sup> Many derivative and offshoot simulation programs and tools have evolved in the meantime, but the current generation of SPICE-based simulators still uses a Modified Nodal Analysis (MNA) [5][6] formulation to convert a textual circuit description (netlist) into a system of ordinary differential equations.<sup>‡</sup> If a model is intended for general use in the electronics industry, for all practical purposes it has to be compatible with SPICE-based simulation tools, since virtually no modeling tool is good enough to convince designers to switch to an alternate simulation tool. A practical modeling approach should be compatible with the SPICE netlist input format or at least the MNA matrix representation, given minimal integration effort. Preferably the integration should be transparent.

A model which consists of standard SPICE elements will be compatible with the vast majority of simulators in use in the electronics industry. It is almost safe to say that a *linear* model composed of standard SPICE elements<sup>††</sup> will be compatible with every time-step-driven simulation tool in use in the digital electronics industry today, particularly in interconnect design.

---

<sup>†</sup> For an interesting and impressively thorough review of the early history of circuit simulation, the reader is referred to Reference [4].

<sup>‡</sup> MNA refers to a specific algorithm for converting a textual netlist which represents a circuit transfer function into a format used internally by a simulation program. While the same information is contained in both representations, the latter is more suitable for efficient computer storage and evaluation.

<sup>††</sup> A linear SPICE-compatible model would consist of linear resistors, capacitors, inductors, and controlled sources. Elements of a nonlinear model might include diodes, transistors, nonlinear resistors, nonlinear capacitors, nonlinear inductors, or nonlinear controlled sources.

### 2.1.4 *Absolute Stability*

Absolute stability, or passivity, is a generalization of asymptotic stability. A passive system is one which does not generate energy for any set of inputs or terminations. For this reason a passive circuit is asymptotically stable for any and every passive termination—an absolutely stable model will never be the source of asymptotic instability in a circuit simulation. This is also a good approximation to reality, since no closed physical system is active due to the conservation of energy, succinctly stated in the first law of thermodynamics.<sup>†</sup> However, absolute stability is mathematically more difficult to verify and enforce than asymptotic stability, and will not be treated rigorously in this dissertation.

## 2.2 *Modeling Approaches in the Literature*

The purpose of this section is to review interconnect modeling approaches presented in the literature. However, it is not intended to address each individual approach, but rather to group them into general categories which more or less represent the whole of the existing literature on the subject. Indeed the body of literature is a vast one, and no guarantee is given that this section covers every facet. In an attempt to effect a useful comparison of the various types of modeling approaches in existence, care will be taken to evaluate the performance of each in terms of the four criteria stated in Section 2.1.

### 2.2.1 *Lumped Models*

Interconnect circuitry has traditionally been characterized using models composed of standard linear RLC elements, coupled inductors, and lossless transmission lines. Positive-valued resistors, inductors, and capacitors are passive, and use of passive elements guarantees an absolutely stable (and therefore asymptotically stable) model which is completely compatible with a SPICE-based simulator. While use of negative-valued elements

<sup>†</sup> The first law states that energy can neither be created nor destroyed. The second law of thermodynamics, which states that all processes serve to increase the amount of entropy in the universe, implies that all physical systems are lossy if the entropy is not included as one of the state variables, which is always the case in electrical circuit analysis.

does not preclude absolute stability, it greatly complicates the evaluation of whether or not a circuit is passive.

If the topology and element values of a lumped model are well chosen, the model will accurately represent the behavior of the actual circuit within a frequency range. In practice, however, the choice of model topology and element values is not necessarily a trivial task. For simple circuits with regular geometries, topology and element values may be determined by inspection or by simple formulas. For more complex circuits, automatic extraction is necessary [7][8], generally by discretization of Maxwell's equations. However, it should be noted that in order for automatic extraction methods to maintain accuracy up to a desired frequency, the circuit in question must be discretized into pieces smaller than the smallest propagating wavelength of interest due to equation (1.1) and the accompanying discussion from Chapter 1. This usually results in a circuit which is larger than necessary, since many of the circuit resonances occur at frequencies far above the frequencies at which the model will be evaluated. This is poignantly obvious in that automatically generated netlists resulting from such techniques can be prohibitively large, even for a circuit of modest physical size. Strategies exist [9][10][11][12] for reducing a large netlist of this type to a smaller system while maintaining the important low-frequency behavior. These may be seen as valid methods for trading away model accuracy in return for reduced complexity. However, it is not uncommon for the original complex model to be too large to even perform the reduction, in which case simulation remains impossible. If reduction is possible and the original extracted model is accurate, an accurate reduced model can be obtained. However, if the original model is inaccurate, the reduced model is difficult to modify by adjusting or tweaking if greater accuracy is desired.

In some sense, lumped element models represent the discretization of the spatial variables present in Maxwell's equations, which transforms them from a small set of partial differential equations in space and time into a typically larger set of ordinary differential equations in time. Since the discretization size (as discussed in the previous paragraph) used in the transformation must be based on the wavelengths with which the

device is to be excited, it follows that a lumped model of an electrically large structure will tend to be large—that is, it will have many branches. For this reason evaluation of a lumped-element model becomes computationally expensive for devices which are large compared to the wavelengths present in the excitation signals. In many cases, a distributed model may be more appropriate to characterize such a circuit.

### 2.2.2 *Distributed Models*

Although lumped-element models can become prohibitively expensive for a circuit whose size is large compared to the excitation wavelengths, distributed models are well suited to the characterization of such circuits, particularly if they behave more or less as uniform transmission lines.<sup>†</sup> So many different distributed models have been presented for use in place of and in conjunction with lumped-element models that the literature on the various distributed models is too vast to rigorously reference. The reader is simply referred to two of the many books on transmission-line theory [13][14] and some typical example models in the literature [15][16].

Distributed models are generally derived analytically from Maxwell's equations, subject to particular sets of boundary conditions and simplifying approximations. This type of model often maintains variable parameters for specific adjustments such as geometric dimensions and dielectric constant. Parameters are also typically available for inclusion of various types of loss such as ohmic loss, dispersion, and skin effect. For a well-designed distributed model, any physically reasonable choice of these parameters results in a passive representation. However, this property will vary from model to model, and is *not* a general property of distributed models. The range of accuracy for a distributed model is bounded by the frequency at which dominant effects of the device's behavior are no longer modeled. Because this range is fixed, there is no real tradeoff available between model accuracy and complexity. However, trading accuracy for complexity is often of minor importance when dealing with distributed models, since a distributed

<sup>†</sup> An interesting approach which generalizes the concept of the distributed model to any type of circuit, like or unlike a transmission line, is presented in [8]. This approach can be viewed as a hybrid between lumped and distributed models.



model may usually be very efficiently evaluated (in terms of CPU time) in a simulation tool.

The area in which distributed models typically suffer is that of compatibility with existing simulation tools, since no distributed models (except arguably the ideal uncoupled transmission line) are standard SPICE elements. For this reason the model must be coded into the simulation program before it may be used in conjunction with a particular simulator. Despite this drawback, distributed models are quite useful in interconnect simulation since they allow efficient simulation of typical transmission line structures, which constitute a significant portion of interconnect circuitry. In fact, some commercially available simulation tools which are tailored for use in the interconnect design community implement multiple distributed models. Nevertheless, traditional distributed models are not flexible enough to represent more complex structures, such as packaging and connectors, for which modeling has become important.

### 2.2.3 *Measurement-Based Models and Techniques*

Models based on measured data have increased in popularity due to the wide variety of structures which must be modeled to ensure signal integrity. Highly complex structures often create intractably large models if an attempt is made to automatically generate a lumped-element netlist from geometry and material constants. Furthermore, atypical or unorthodox structures neither lend themselves well to hand analysis, nor are they well represented by typical distributed models. A place has traditionally been reserved for measurement-based models to be used in cases such as these, but only when all else fails. As these cases become more common, however, academia and industry are taking a more serious look into measurement-based models.

In pure measurement-based models, the model parameters themselves consist primarily of measured data points. Obviously there will be some compatibility problems with most simulation tools, since this type of model is not a standard feature of SPICE, and furthermore such elements are not utilized in traditional passive circuit design. Consequently, the ability to evaluate such a model must be coded into a simulation tool for the

model to be useful. The more general set of measurement-based modeling *techniques* consists of those which use measured data not just to optimize hard-coded model parameters, in the traditional sense of model extraction, but instead use measured data to determine the parameters of a dynamic model—one whose form is set based on the measured data itself.

It is not uncommon for the measured data in question to be approximate time-domain impulse response samples. In this case, evaluation of a measurement-based model consists of convolving the device impulse response with the input waveforms in order to compute the output waveforms. The range of accuracy of the model may be reduced by filtering and downsampling the approximate impulse response. However, only specific convolution implementations may take advantage of this to reduce the computational burden of the model evaluation. In fact, a primary drawback of techniques which require full convolution [17][18][19] is that they can be computationally expensive if the system impulse response waveforms are long, since the number of operations to simulate to the  $n^{\text{th}}$  timestep is proportional to  $n^2$ . If techniques based on recursive evaluation of the convolution integral [20][21] are employed,<sup>†</sup> the required number of operations is reduced to being proportional to  $n$ . However, a large number of operations is still required (that is, the constant of proportionality remains quite large), and the necessary computer code for model evaluation becomes significantly more complicated.

Another approach to measurement-based modeling is to use samples of the frequency-domain impulse response as model parameters. In many frequency-domain simulation applications, such an approach is quite adequate, and in [22] it was extended to include simulation with nonlinear loads via harmonic balance techniques. In time-domain simulation, however, it is necessary to perform an FFT and inverse FFT at every time step, as outlined in [23] and [24], which is effectively as expensive as the full convolution techniques mentioned above. Extension to nonlinear loads would require that the FFT and its

<sup>†</sup> Such models are not purely measurement based, since they transform the model data into an intermediate representation. However, the *technique* is measurement based, since the model parameters *and model size* are determined at extraction time.

inverse be performed for *every nonlinear iteration*, increasing the computational burden even further.

Since numerical waveforms are used which decay to zero for large time, asymptotic stability of an accurate measurement-based model is almost guaranteed, since the model response must also decay to zero. However, the absolute stability of the model is not mathematically ensured, and the model could therefore cause a simulation to become unstable given the correct passive termination. In practice, however, if the data used in the model is sufficiently passive, an accurate model will very likely be passive as well, for the same reason that it will be asymptotically stable.

### 2.3 Overview of the Model to be Presented

The model to be presented in this dissertation is a hybrid which uses a measurement-based technique to extract a lumped-element model. Use of lumped elements ensures compatibility with existing SPICE-based simulators, and also allows use of the various tools available for balancing accuracy and complexity of a lumped-element circuit. Asymptotic and absolute stability, however, are subject to the heuristic constraints inherited from measurement-based models. These constraints dictate that an accurate model will in many cases be asymptotically and absolutely stable, although there is no guarantee of such. These and other properties of the model will be discussed in detail in the following chapter.

## Chapter 3

### *OVERVIEW OF THE PROPOSED INTERCONNECT MODELING APPROACH*

In this dissertation, a reliable and practical approach is presented for modeling interconnect circuits including printed circuit board traces, various connectors, IC packages, and MCM/hybrid substrates. Precursory work to this was published in [21] and [25], while the current work itself was published in [26], [27], and [28]. The modeling technique automatically extracts a subcircuit composed of standard SPICE elements from the measured time-domain scattering step response waveforms of a multiport device under test (DUT). Because the model is extracted from decaying time-domain waveforms, asymptotic stability is guaranteed if the fit is accurate. Use of standard elements ensures complete compatibility with existing SPICE-based simulators, and the model can be incorporated into any such simulator and connected by its port nodes to any other circuit. The response of the model is valid up to a specified maximum frequency, and network reduction methods allow the model to be automatically simplified for use in narrower frequency ranges.

The model is completely general in the sense that it is extracted solely from the measured port behavior of the DUT, so that linear circuits with arbitrary geometry and cross-coupling can be modeled without knowledge of their internal characteristics. However, as

will be discussed in this chapter, the generality of the approach is limited in practice by the size of model it can extract because of the finite compute power available. Asymptotic stability of the scattering parameters is guaranteed for an accurate model, but absolute stability is not guaranteed. However, if the original measurements are passive and the model is accurate, absolute stability is likely. The subsequent sections of this chapter explain each of the above attributes of the modeling approach in greater qualitative detail.

### *3.1 Measurement-Based/Lumped-Element Combination*

The modeling approach to be presented is to some extent a hybrid between measurement-based and lumped-element modeling, two of the categories introduced in Chapter 2. That is, although the model is composed of lumped elements, its size and form are dictated by the measurement data, and not by materials constants, geometric considerations, or the whim of the modeler.

The primary reason for a measurement-based approach is that it allows any device to be modeled, since it is based only on the port behavior, which is a characteristic common to every device. Of course there are many different types of measurements available upon which to base such a modeling scheme. It was decided for the purposes of this work to use the time-domain scattering response of the device in question, since this response is directly obtainable at high frequency using time-domain reflectometry (TDR) [29][30]. Additionally, scattering parameters often allow a more concise time-domain representation of a typical interconnect circuit than other representations such as admittance or impedance. Although scattering parameters were chosen for the above reasons, the modeling approach is general in that a model may be derived from any full (*i.e.*, linearly independent) set of measurements with only minor alterations to the derivation and resulting algorithms.

The data used for model extraction is measured using TDR methods, and the various algorithmic steps and numerical methods outlined in Chapters 5 and 6 are used to operate on this data until the final result is realized in the form of a useful model. As previously discussed, using electrical measurements to build an electrical model allows sim-

ple, direct verification of model accuracy, provided the measurements are accurate.<sup>†</sup> This is because not only are the original measurements available, but the measurement apparatus (that is, the fixturing setup) is already designed and functional, and can often be used to make additional independent supporting measurements if necessary. Some issues involved in obtaining measurements necessary for model extraction are discussed in Appendices A-C.

At the same time, using lumped elements to represent the model ensures compatibility with existing simulation tools, allowing the designer to draw on simulation resources which have been developed and improved over the past 25 years. An incompatible model, on the other hand, would find very little acceptance outside of academic circles, since proving its usefulness to the point of custom integration into existing industry CAD frameworks would be difficult, if not impossible. From a theoretical perspective, because a lumped-element circuit is a representation of a set of coupled constant-coefficient differential equations, an enormous body of applicable mathematical knowledge and methods exists for development and analysis of such models. This paves the way for further refinement of lumped-element modeling approaches, based on existing work. One important aspect of this existing work is various black-box network reduction algorithms, such as those proposed in [10] and [12], which operate directly on the state space of such a model. These reduction schemes form the mathematical basis for realizing an accuracy/complexity tradeoff for lumped models, the importance of which was emphasized in Section 2.1.2. Some care must be taken, however, in that the most general reduction algorithms have not yet been sufficiently studied and tested to allow for blind implementation as a subsystem of an overall modeling scheme.

As was discussed in previous sections, there are specific drawbacks to measurement-based and lumped-element modeling approaches. Foremost, measurement-based approaches are hampered by the inherent necessity to take accurate, high-frequency mea-

<sup>†</sup> In contrast, if the measurements are inaccurate, modeling and model verification are inaccurate as well. Various measurement accuracy issues are mentioned in Appendices B and C.

surements. This issue is important because such measurements can not be typically made by a technician or untrained engineer, but are best made by an engineer with experience in high-frequency measurements or preferably high-frequency design. Difficult fixturing issues which require creative, effective solutions cause high-frequency measurement to take on the flavor of an art as well as a science. Furthermore, taking measurements implies the existence of a part to measure, which rules out the use of measurement-based approaches to model a part that has not yet been prototyped or built.

While these drawbacks might seem likely to rule out the applicability of measurement-based modeling to real-world design, neither drawback is especially significant in modeling of critical off-chip interconnect circuits. This is because in practice, any critical model must be compared against measurement to verify its accuracy; that is, *measurement is already present in the modeling loop*. This is certainly true in the area of nonlinear device modeling, and is often true in linear modeling as well because of the limited accuracy of models developed by hand analysis, and the high complexity of models computed with field solvers. If it is then necessary to take measurements for verification, it will be necessary to build some form of prototype. A major digression is not required to build a model from such a prototype.

In some aspects of circuit design, the part to be modeled is already in production, and the issue of prototyping is irrelevant. An example would be deciding what type of package or connector to use for a given application. In many cases the package or connector is already built and available, and in a simplistic case the decision is reduced to obtaining an accurate model of each prospect and simulating the circuit with the various models in place. Although it is becoming more common for vendors of such parts to provide models to their customers, there is no guarantee that such a model is sufficiently accurate over the frequency range under which it will be used. Furthermore, when the part is introduced into a particular application, shifted ground and power planes will often cause the model to become inaccurate. The best recourse is often to develop an in-house measurement-based model under signal conditions and geometrical considerations appropriate to the intended use of the part.<sup>†</sup>

The significant drawback to lumped-element models, as discussed in Section 2.2.1, is the complexity which is introduced if the model represents an object which is large in comparison to the smallest wavelength present in the incident signal. Such circuits—a long transmission line, for example—are generally better represented by a distributed model, if one may be extracted.<sup>†</sup> In practice, however, many interconnect circuits and subcircuits are small enough compared to the wavelengths in question that lumped models are not only sufficient but are preferable, because of the tendency of distributed models to introduce time-consuming breakpoints into time-domain simulations [33].

### 3.2 Automatic Extraction

The usefulness of *automatic* extraction of off-chip interconnect models cannot be understated, since building a model manually can be a time-consuming and therefore expensive process, and the accuracy of the resulting model is often less than desired. The lack of a reliable automated approach becomes a limiting factor in design when inaccurate models are used, or when no extraction is attempted at all due to the amount of effort required to obtain questionable results. The motivation for this work is to provide an automatic interconnect modeling capability to circuit designers where seemingly no such capability exists.

Attempting automatic extraction of a ratio-of-polynomials transfer function in the frequency domain, where the problem is more easily stated, usually leads to problems with asymptotic instability. This is because in the frequency domain if a pole happens to slip across from the left half of the complex frequency plane to the right half, a relatively small error is introduced on the imaginary axis, where the Fourier transform is observed. However, this often leads to irrecoverable error in the time domain, since the pole will

<sup>†</sup> This scenario points to another broad application of measurement-based modeling: the modeling of various interconnect parts by their original vendors, using typical usage conditions. An accurate industry-compatible model would significantly enhance the stature and value of such a product.

<sup>†</sup> The subject of distributed models and associated parameter extraction is outside of the scope of this dissertation, and the interested reader is referred to references [31] and [32].



eventually cause exponential growth to entirely dominate the response and, in practical terms, to halt the circuit simulation session. In other words, the model is practically useless.

Significant work has been done in the area of robust system identification [34], which refers generally to determining information about a system based on measurements. However, to the author's knowledge it has been developed primarily for scalar functions (analogous to the humble one-port in circuit theory) to interpolate between data points, and also for identification of radar targets according to their resonant modes. Very little system identification work has been concerned with electrical modeling, and certainly none has been applied toward development of SPICE-based electrical models.

The most practically significant work in automatic SPICE-compatible time-domain extraction of interconnect models was the work on impedance peeling by Jong, Hayden, and Tripathi [35][36]. This work was encapsulated in two successive industry modeling tools from Tektronix, IPA310 and IPA510 [37], and marketed by Tektronix under the trade name Z-profile [38]. Although in its most direct implementation impedance peeling is known to suffer from numerical instability due to ill conditioning, standard deconvolution techniques [39][40] may be applied to improve the numerical properties of the algorithm if necessary. Perhaps the most significant limitation of the approach is its current applicability in automated form to only near-lossless transmission line structures, which must be either uncoupled or symmetric.<sup>†</sup> In reality, however, a large number of circuitboard applications fall into this category, making impedance peeling quite useful in many modeling situations. Many additional applications have been developed [42][43] which allow the impedance waveform generated by the peeling algorithm to be used as a starting point for models of other types of circuits. However, because significant manual intervention is required in such applications, impedance peeling in its current form cannot be considered an automatic extraction technique.

<sup>†</sup> In reference [41], the algorithm was extended to include ohmic (series DC) loss. However, no variation as of yet deals rigorously with dispersion or skin effect losses.

To achieve automatic extraction as outlined in this work, it was necessary to use generalized forms of the typical system identification formulations to make them applicable to matrix functions (and therefore multiport circuits). Since automatic results must be reliable to be of practical use, preconditioning techniques were developed to make the extraction problem feasible and well conditioned. Finally, significant post-processing steps were necessary to further reduce the size of the model, to transform it into a representation which is useful to a circuit simulator, and to reduce the simulation problem to a tractable size. This approach represents a significant deviation from currently available modeling approaches. However, such an alternative approach seems necessary in order to achieve reliable automation in interconnect modeling where none currently exists.

### 3.3 *A General Model*

As stated at the beginning of this chapter, the model used in this work is theoretically general enough to characterize any linear multiport circuit. Use of such a general model is seemingly the simplest way to implement a truly automatic model extraction algorithm, since automatic generation of circuit-specific models requires a computer to effectively decide which model (if any) is appropriate to characterize a given data set. This is a difficult task indeed, if the decision is to be based solely on measured data. Furthermore, if the approach is to be considered an “interconnect modeling” approach, it must consist of a set of models which together span the space of linear multiport circuits. This coverage is achieved most simply if that set consists of a single, sufficiently general model.

The model which was developed in this work is general in that a single extraction technique is used regardless of what circuit is being modeled. This results in all extracted models having a very similar form when seen from a general enough viewpoint. The ability of the model to characterize any data set is preserved by varying the number of modeling sections used and the values of the elements. The decisions as to the number of sections and values of elements are made by the computer through application of numerical techniques to the measured data used as inputs.

To the engineer who desires a model which will recreate the circuit's behavior inside of a simulator, any model is as good as any of the other available models (an infinite number exist) which will do the same. A model created with this purpose in mind is called a *simulation* model or *black-box* model—it accurately represents the behavior of the circuit at its ports, so that a simulator can quantify the effects of the circuit when it is connected to any other circuitry via its ports. Any non-port waveforms, however, are considered unimportant and arbitrary. If the effects of finite precision element values are ignored, the only noticeable difference between simulation models of the same device is the variation in compute time for model evaluation—a model which evaluates more rapidly will obviously be preferred. With the concept of a black-box model in mind, we are free to choose any model from the available set of models corresponding to the device being characterized.

The term *physical* model, on the other hand, is often used to refer to a model which geometrically “makes sense” to the engineer who is using it. That is, the engineer can more or less map each element in the model to a physical section of the circuit in question. Such a model would likely be used to troubleshoot a design, to see what portion of the circuit needs to have, for example, its parasitic inductance reduced. It will be seen in subsequent chapters that the general black-box model presented in this work would likely not be useful for such applications. However, it is the author's contention that the probability of automatically generating anything as ambiguous as a “physical” model is quite low. This is because such a model is not defined by its quantifiable inputs and outputs, but instead by the subjective requirement that it “make sense” to the person using it. That is, it is not really a simulation model to be evaluated by a computer, but instead is a mental model to be evaluated by an engineer. For this reason, determining a physical model will require manual intervention, preferably by the individual who will be evaluating the model. In summary, the general model proposed and employed in this dissertation is useful for developing accurate simulation models, but will not likely be useful for developing a physical model. This seems to be the price paid in exchange for automatic extraction capability.

It should be noted that only in theory may the above model be considered completely general. In practice it is limited by two factors which tend to be correlated. The first is the inherent difficulty in taking a large number of measurements. Appendix A outlines in some detail the steps necessary for taking multiport scattering measurements, and Appendix B attests to potential difficulty in implementing a measurement scheme. For the sake of this discussion, it is sufficient to say that  $\frac{m(m-1)}{2}$  measurements are necessary to completely characterize a symmetric  $m$ -port circuit. For a two-port circuit, three measurements are necessary, and for a four-port circuit, ten measurements are necessary. However, for a ten-port circuit, 45 measurements are necessary. So it is obvious that complete and rigorous characterization of, for example, a 512-pin package, is out of the question, as it would require some 130,000 measurements. For this type of package, it is typical to utilize geometric symmetry, to ignore weak coupling between distant lines, and to only model critical paths. If implemented wisely, such an approach will maintain the number of measurements at a tractable level while only slightly compromising model accuracy. The difficulty of performing a large number of measurements limits not only the generality of the model when applied to circuits with a large number of ports, but it also limits the ability to automate the modeling of such circuits, since manual intervention is likely required to determine critical paths and weakly coupled ports.

The second practical reason which inhibits the generality of the model is computational power. It will be shown in Chapter 6 that the number of operations required to perform a model extraction tends to increase as the cube of the number of ports in the device being characterized, and also as the cube of the number of data points in each measured waveform. The proportionality constant may be altered by choosing from various methods of preconditioning the problem, but the fact remains that circuits with large numbers of ports or long time-domain responses will be increasingly expensive to characterize, and at some point will be effectively impossible to characterize.

### 3.4 *Asymptotic Stability*

For a model to be even partially useful, it must be asymptotically stable. The approach to be presented in this dissertation cannot mathematically guarantee asymptotic stability, but must instead rely on a heuristic tendency toward asymptotic stability, stated in the following paragraph:

It can be shown that the approach presented accurately models the time-domain waveforms used as input data. Since these waveforms are measurements of asymptotically stable systems, they decay to zero with time, and therefore the model response decays to zero with time, since it accurately recreates them. It will be shown later that the response of a lumped-element model can actually be represented as a sum of weighted exponentials. If any one of them is growing significantly within the time window in which the model was built and evaluated, then the entire response will be growing, which would cause inaccuracy and contradict the initial statement that the approach accurately models the input waveforms.

However, this argument ignores the possibility that perhaps the model response grows significantly outside of the window of evaluation, so that the inaccuracy is not seen. If this is the case, the exponentials responsible for the growth may be removed from the model, since they obviously have little or no effect on the behavior of the model within the evaluation window, and therefore their removal will not incur significant error. In fact, the algorithm employed for initial reduction was chosen in part for its ability to weed out such portions of the model, which are extraneous in that they don't affect model behavior in the time window of interest, but do serve to increase the size and complexity of the model.

These two arguments are sufficient to show that the algorithm tends to produce asymptotically stable models if the time-domain data used to build the model is itself decaying, as well it should be. These arguments also make a strong case for time-domain extraction of this type of model, since when applied directly in the frequency domain, they have little bearing on asymptotic stability. This is because a system in the frequency domain is characterized by its singularities, and a small error near a singularity tends to cause asymptotic instability. In the time domain, however, asymptotic instability causes

such egregious error that significant instability is very unlikely if the time-domain error is held below a reasonable limit. Furthermore, if there is insignificant instability causing potential exponential growth in the model, it may be safely and methodically removed. Techniques for removal of asymptotic instability will be discussed in Chapter 6 after the model and the extraction algorithm have been presented in detail.

### 3.5 *Absolute Stability*

Absolute stability, which is the inability of a system to generate energy, must be implied in a similar fashion to asymptotic stability, but the correlation is not as strong. To support the tendency toward *asymptotic* stability, we utilized the fact that the waveforms used to build the model are decaying, so that an accurate model will also decay. To imply *absolute* stability, we must rely on the fact that the waveforms represent a passive circuit, and therefore an accurate model will be passive as well. However, since the correlation between passive data and a passive model is not as strong, neither is the argument. Unless the amount of error which can be absorbed before the model becomes active is known, it is difficult to know how much is allowable in the extraction step while still maintaining passivity. Furthermore, errors occurring in measurement and de-embedding can cause the original time-domain data to be active, in which case an accurate model is likely to be active as well. In short, it is difficult to guarantee passivity for an extracted model, although a device with significant loss over the entire frequency spectrum (a resistor, for example) will likely produce a passive model. However, many interconnect circuits are nearly lossless, especially at low frequency, and do not fall into the category of broadband lossy circuits. For this reason, extraction of passive models is an area of continuing research.

### 3.6 *Conclusion*

In conclusion, the modeling approach presented in this dissertation is a general technique which is compatible with industry standard circuit simulators. Because it is automatic, it is quite useful for developing black-box simulation models with little manual intervention, given a set of measurements. Although it is not likely to produce a physical

model, it was emphasized that automatic extraction of something as ambiguous as physical model is nearly impossible—the required manual intervention excludes complete automation. It was noted that taking a full set of multiport measurements can be an arduous task and requires experience or training. However, building and measuring a circuit is already a necessary aspect of model verification, and therefore does not significantly increase the cost of many design cycles. Finally, the method tends strongly toward asymptotically stable models, but somewhat less strongly toward absolutely stable models. Many of the concepts and issues introduced in this chapter will be discussed in greater depth as the algorithms and flow of the modeling approach are presented in Chapters 5 and 6. However, these discussions must be postponed until a basis has been established in circuit theory, which is the subject of the following chapter.

## Chapter 4

### *REVIEW OF PERTINENT CIRCUIT AND SYSTEM THEORY*

#### *4.1 System Theory Overview*

A basic concept of engineering is that of the *system* which responds to a set of inputs by producing a predictable set of outputs. Regardless of the origin or destination of the system's inputs and outputs, they can be viewed as purely mathematical abstractions. Likewise, any physical characteristics or properties of the system itself, such as size, shape, and materials, can be completely abstracted away and the system represented only as a mathematical mapping of inputs to outputs. *System theory*, also referred to as *network theory*, refers to the mathematical methods and tools used to rigorously analyze such systems, as well as the properties and definitions used to define different types of systems.

Only a few of these methods and tools are general enough to apply to any system. A more typical approach to system analysis is to break the complete set of systems into subsets delineated according to their mathematical characteristics, so that specific mathematical tools may then be used to treat the different subsets. Some of these tools were developed independently as purely mathematical, system-theoretical entities to operate on a particular abstract set of systems. Others, however, were concocted to solve a problem



occurring as part of a particular application, and later generalized to become part of the body of knowledge of system theory.

In this sense it is helpful to view solutions to various engineering problems in the context of system theory, so that innovations will more quickly find their way into the general body of knowledge. This makes solutions more accessible to scientists and engineers in many disciplines, because many of the concepts, theoretical results, and applications may be presented in the more universal language of system theory, rather than being obscured by the local dialect of the innovator's particular engineering discipline.

The converse is true as well, that a researcher who chooses to investigate a subject in the context of system theory has at his or her disposal the same large body of system theory knowledge. However, in the experience of the author there is no universal "system theory clearinghouse" by which to access this body of knowledge, with the unfortunate result that it is quite scattered across various disciplines and subdisciplines of engineering, mathematics, and the physical sciences. As a result, many powerful aspects of system theory are utilized only by a relatively small number of individuals, and a large number of researchers are constrained to re-invent the wheel as it were by duplicating the results of others in parallel fields.

This is largely the case because the various technical journals in which such innovations are published are necessarily defined according to specific applications, and inclusion of verbiage to translate system theory into the dialect of the journal readership is costly in terms of page count (and consequently page charges!). Furthermore, although many engineering texts have been written [45][46] which present their subjects in terms of system theory, they are often avoided for fear of confusing application-oriented students with extraneous mathematical abstraction. (Or, if such a text is used, course content typically downplays system theory aspects in favor of application-specific problem solutions.) Such an approach is understandable at the undergraduate level, where students are groping to learn the basic methods for solving problems typical to their discipline. However, it is unconscionable at the graduate level, where students must learn general problem-solving

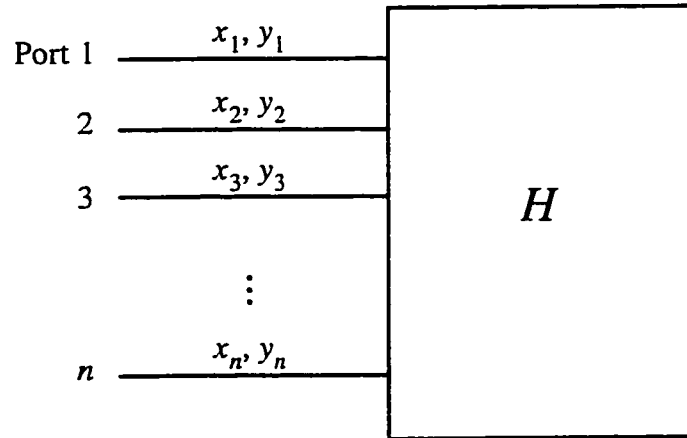
techniques, and are often called upon to borrow results from other disciplines and subdisciplines and tailor them to solve a specific problem.

Circuit theory is the brand of system theory which is specific to electrical circuit analysis and design. In circuit theory, the system in question is generally assumed to be some electrical circuit, and the input and output variables are usually some electrical quantities closely related to voltage and current. The remainder of this discussion will be based on these assumptions, since the application at hand is indeed the *electrical* modeling of off-chip interconnect. However, every effort will be made to keep all derivations as general as possible, allowing a straightforward change of variables to transform the results to apply to non-electrical systems as well. The remainder of this chapter is devoted to introducing the mathematical and circuit-theoretical framework necessary to support the derivation of the model extraction algorithm in Chapter 5.

## 4.2 Linear Time-Invariant Multiport Networks

An  $m$ -port multiport network is one which maps  $m$  inputs to  $m$  corresponding outputs. In circuit theory, the input and output variables are generally some electrical quantities, such as voltage and current. The  $m$  ports correspond to points where the circuit can be excited and its response measured. A typical pictorial representation of an  $n$ -port system is shown in Figure 4.1, in which the system itself is represented as a “black box” that operates independently on the inputs  $x$  introduced at its ports, resulting in the outputs  $y$  which are measured at the same ports.

Mathematically, such a system is described by its *transfer function*  $H$ , which is a statement (usually mathematical) of the mapping from the inputs to the outputs. It should be noted that since the transfer function describes the mapping from input to outputs, complete knowledge of both the inputs to the system and the system transfer function allows unambiguous determination of the outputs. It is also worth mentioning at this juncture that two complementary methods are typically used to describe such a system: time domain and frequency domain.<sup>†</sup> The time domain is typically a more intuitive representation of a signal to most casual observers, since a large part of human experience involves



**Figure 4.1** Pictorial representation of a system. The transfer function  $H$  operates on the inputs  $x$  to produce the outputs  $y$ .

the continuous, relentless march of time, and for this reason we will use it to define various properties of systems.

The relationship between the inputs and outputs of an  $m$ -port system may be stated in the mathematically in the time domain as

$$\mathbf{y}(t) = H[\mathbf{x}(t)] \quad (4.1)$$

where  $\mathbf{x}(t)$  is a vector of input signals,  $\mathbf{y}(t)$  is a vector of output signals, and  $H$  represents some functional mapping between the two. Two important properties which describe subclasses of networks are *linearity* and *time-invariance*, defined as follows:

#### *LINEARITY:*

Linearity is composed of the two concepts of scaling and superposition. Scaling dictates that if the input to a system is scaled by a constant factor, then the output is scaled by the same factor. The second component of linearity is superposition: if it is known that a system maps two arbitrary input signals to two corresponding output signals, then super-

<sup>†</sup> Although nomenclature for these two domains for system analysis varies across different fields of system theory, they are always related by the Laplace transform [44].

position requires that the sum of the two input signals map to the sum of the corresponding output signals. These verbal descriptions may be stated more rigorously in the following form:

Given a system whose transfer function  $H$  maps  $x(t) \rightarrow y(t)$  such that  $y_1(t) = H[x_1(t)]$  and  $y_2(t) = H[x_2(t)]$ , the system is linear if  $ay_1(t) = H[ax_1(t)]$  and  $y_1(t) + y_2(t) = H[x_1(t) + x_2(t)]$  for every  $x_1(t)$  and  $x_2(t)$ , and for every constant  $a$ .

#### *TIME INVARIANCE:*

Time invariance is more intuitive, in that it requires that the only effect of delaying the input to a system by a certain time interval is that the output is delayed by the same time interval. The following mathematical explanation is more rigorous:

Given a system whose transfer function  $H$  maps  $x(t) \rightarrow y(t)$  such that  $y(t) = H[x(t)]$ , the system is time invariant if  $y(t - t_0) = H[x(t - t_0)]$  for every  $x(t)$  and every  $t_0$ .

Although many analog circuits are time-invariant, it should be apparent that many physical systems in which we find ourselves are not. As an example, imagine that an individual suffers a massive heart attack and the paramedics find him collapsed on the sidewalk. Now consider a system whose input is Cardiopulmonary Resuscitation (CPR) repetitions, and whose output is the general health of the subject. Time invariance dictates that delaying the commencement of CPR for 30 minutes will find the patient recovering to the same state as he would have, only 30 minutes later than he would have, which is clearly not realistic in this case. Furthermore, linearity would imply that if the paramedics were to apply ten times the normal force in their CPR repetitions, the patient would find himself ten times healthier when all is said and done, which is also highly unlikely. Obviously such a system exhibits highly nonlinear and time-varying mapping between input and output variables.

However, in the analysis of a surprisingly large set of systems, particularly for a large set of electronic circuits, very little error is incurred by assuming linearity and time invariance. As a result, the analysis of such circuits is greatly simplified. Since the large majority of electronic interconnect circuits falls into this category,<sup>†</sup> we continue our discussion of pertinent circuit and system theory for the most part under the general assumption that all systems of interest are linear and time invariant. The concepts of linearity and time invariance are discussed in more detail in [47].

### 4.3 State-Space Representations of Continuous-Time Systems

A convenient representation of a network transfer function of a system with a single independent variable is a set of coupled ordinary differential equations

$$\dot{\mathbf{x}} = f(t, \mathbf{x}), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad (4.2)$$

where  $\mathbf{x}$  is a vector of unknown state variables called the *state vector*, known only at a single timepoint  $t_0$ . It is worth noting explicitly that an  $n$ th-order differential equation (that is, one containing the  $n$ th derivative of a scalar function  $x(t)$ ) may be converted into a coupled set of  $n$  first-order equations such as the set shown above by a standard change of variables [3]. For this reason, (4.2) is general enough to represent any set of ordinary differential equations.

The system of equations which will be investigated and utilized in this work is a subset of the set of systems represented by (4.2), such that the function  $f(t, \mathbf{x})$  will be restricted to a scalar matrix multiplied by  $\mathbf{x}$ . That is,

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}, \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad (4.3)$$

---

<sup>†</sup> The only exception of significance is those interconnect circuits with nonlinear magnetic properties. However, these nonlinearities are typically second-order effects when present.

where  $\mathbf{A}$  is a constant matrix known as the *state matrix*, and  $\mathbf{x}$  is the state vector as defined previously. This is a significant simplification indeed; however, application of the principles of Fourier analysis tells us that the formulation in (4.3) represents a set of basis functions which is sufficient in theory to represent any well-behaved linear system of a single independent variable. Practically speaking, as applies to the problem at hand, (4.3) is able to represent the large majority of interconnect circuits, and arguably any such circuit encountered in practical circuit design.

The solution to (4.3) is [44]

$$\mathbf{x} = e^{\mathbf{A}t} \mathbf{x}(0) = e^{\mathbf{A}(t-t_0)} \mathbf{x}_0 \quad (4.4)$$

where the notation  $e^{\mathbf{A}t}$  denotes the matrix exponential. A cursory definition of the matrix exponential may be given as

$$e^{\mathbf{A}t} = \mathbf{X} e^{\mathbf{\Lambda}t} \mathbf{X}^{-1} \quad (4.5)$$

where  $\mathbf{A} = \mathbf{X} \mathbf{\Lambda} \mathbf{X}^{-1}$  is an eigendecomposition of  $\mathbf{A}$ , and

$$e^{\mathbf{\Lambda}t} = \begin{bmatrix} e^{\lambda_1 t} & & & \\ & e^{\lambda_2 t} & & \\ & & \dots & \\ & & & e^{\lambda_n t} \end{bmatrix} \quad (4.6)$$

where  $\lambda_1, \lambda_2, \dots, \lambda_n$  are the diagonal elements of the eigenvalue matrix  $\mathbf{\Lambda}$ . Although the above definition breaks down in the case in which  $\mathbf{A}$  is defective (that is,  $\mathbf{X}$  is singular), that case is treated rigorously in both [44] and [48]. Finally, as a matter of nomenclature, it is worth noting that the system in (4.3) is often referred to as a *homogeneous* system—one with no external excitation, and the values  $\lambda_1, \lambda_2, \dots, \lambda_n$  are referred to as the *poles* of the system.

It can be seen by inspection of equations (4.4)-(4.6) that  $\mathbf{x}(t)$  actually consists of a sum of weighted exponentials. It is known from Fourier theory that in the limit such a sum is a set of basis functions capable of representing a very broad range of time-domain signals [47], including all well-behaved signals. This supports the earlier statement that (4.3) is sufficient to represent the large majority of impulse responses encountered in interconnect modeling, if not all such responses.

To generalize the formulation in (4.3) into a system which may be excited and whose response may be measured, we introduce the concept of *forcing functions*. A forcing function is an input into the system which is independent of any state variable, and as such may be summed with the response. A typical representation of such a system is

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}(t), \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad (4.7)$$

where  $\mathbf{B}$  is a constant matrix and  $\mathbf{u}(t)$  is the time-varying excitation vector. The solution to (4.7) is [44]

$$\mathbf{x} = e^{\mathbf{A}(t-t_0)} \mathbf{x}_0 + \int_{t_0}^t e^{\mathbf{A}(t-\tau)} \mathbf{B}\mathbf{u}(\tau) d\tau \quad (4.8)$$

The above equation is a surprisingly clean representation of certain arbitrarily large systems. Various SPICE-like circuit simulators (and in general, ordinary differential equation solvers) evaluate (4.8) quite effectively by simultaneously approximating the matrix exponential and discretizing the integral. These solvers utilize robust error control and stability properties built in over several decades of algorithmic development and refinement. Some of the algorithms employed by such software packages are presented in [6] and [3].

However, the purpose of this work is not to evaluate (4.8), but to instead determine the parameters  $\mathbf{A}$  and  $\mathbf{B}$ . (Having done so, (4.8) may be evaluated multiple times as necessary by the pre-existing software packages mentioned above.) For this reason, (4.8) is not particularly useful in its current form for practical evaluation of such systems, primarily

due to the presence of the integral, which is furthermore complicated by the presence of the matrix exponential. In order to determine the parameters **A** and **B**, it is necessary to discretize the integral in (4.8) symbolically, which is the subject of the next section.

#### 4.4 *Sampled-Time Representation of Continuous-Time Systems*

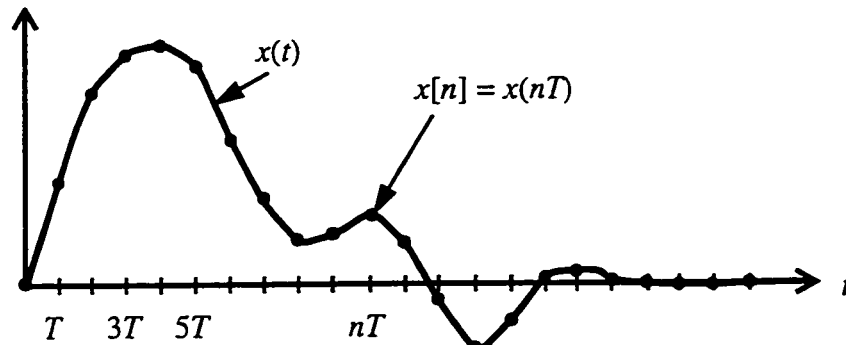
The stated goal of this research work is to develop a model of a circuit based on time-domain measurements. As is the case in most efforts to model circuits for computer simulation, the extraction will be done with the aid of a digital computer, which implicitly requires that the raw data be sampled. That is, some analog-to-digital conversion must take place before the computer can be of use in analyzing the data (in this case, the circuit response) and developing a model. Since the goal of this chapter is to build a theoretical basis to support the extraction of a model from such data, it becomes necessary to extend the developments of the previous section for application to systems characterized by sampled data.

To rigorously handle sampled data, the input and output signals of equation (4.8) are sampled at evenly-spaced timepoints which are separated by a short interval of time  $T$ , known as the sampling interval. The sampling process, depicted visually in Figure 4.2(a), maps a continuous-time signal  $x(t)$  into a discrete-time signal  $x[n]$ , and is represented mathematically by the relationship

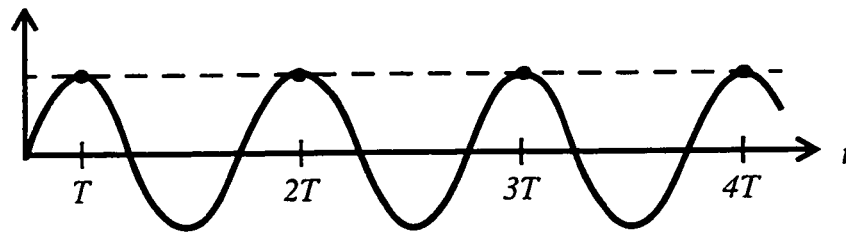
$$x[n] = x(nT), \quad n = 0, 1, 2, \dots \quad (4.9)$$

The sampling interval must be “short” in comparison to the shortest time constants in the signal, which ensures that the complete signal can be rebuilt from the sampled signal if and when desired. In contrast, Figure 4.2(b) depicts a sinusoidal signal which is not sampled frequently enough, introducing ambiguity into the reconstruction process and therefore clouding our knowledge of the true nature of the underlying continuous-time signal. Stated in a more quantitative manner, the sampling scheme must adhere to the Nyquist criterion, as described in [51], in order to avoid aliasing.





(a)



(b)

**Figure 4.2** (a) Accurately sampled waveform. (b) Undersampled waveform, leading to ambiguity in the reconstruction of the original continuous-time waveform. Given the samples shown, the dotted line is a more likely candidate than the actual original waveform.

Having accurately sampled the waveform as shown in Figure 4.2(a), (4.8) is now evaluated over a single sampling interval  $(n-1)T < t < nT$ . We begin by making the following definitions:

$$\begin{aligned} \mathbf{x}_n &\equiv \mathbf{x}(nT) \\ \mathbf{u}_n &\equiv \mathbf{u}(nT) \end{aligned} \tag{4.10}$$

We furthermore assume that  $\mathbf{u}(t) = \mathbf{u}(nT) = \mathbf{u}_n$  is constant on the same interval, and note that (4.10) implies that  $\mathbf{x}_{n-1} = \mathbf{x}((n-1)T)$ . Equation (4.8) is now evaluated on the sampling interval by substituting

$$\begin{aligned} t_0 &\rightarrow (n-1)T \\ t &\rightarrow nT \end{aligned} \quad (4.11)$$

which results in

$$\mathbf{x}_n = e^{\mathbf{A}(nT - (n-1)T)} \mathbf{x}_{n-1} + \int_{(n-1)T}^{nT} e^{\mathbf{A}(nT - \tau)} \mathbf{B} \mathbf{u}(\tau) d\tau \quad (4.12)$$

Substitution of  $\alpha = nT - \tau$  results in

$$\begin{aligned} \mathbf{x}_n &= e^{\mathbf{A}T} \mathbf{x}_{n-1} + \int_0^T e^{\mathbf{A}\alpha} d\alpha \mathbf{B} \mathbf{u}_n \\ &= e^{\mathbf{A}T} \mathbf{x}_{n-1} + (\mathbf{e}^{\mathbf{A}T} - \mathbf{I}) \mathbf{A}^{-1} \mathbf{B} \mathbf{u}_n \end{aligned} \quad (4.13)$$

where  $\mathbf{I}$  is the  $n \times n$  identity matrix. In short, the response of the system  $\mathbf{x}_n$  at any sample point  $n$  may be calculated given the input sample  $\mathbf{u}_n$  at the current timepoint and the response  $\mathbf{x}_{n-1}$  at the previous timepoint.

#### 4.5 System Impulse Response

For the purposes of the modeling application at hand, it is not necessary to thoroughly investigate the response of a system to arbitrary inputs. Rather, we restrict the excitation vector of equation (4.7) to

$$\mathbf{u}(t) = \mathbf{e}_{i,n} \delta(t) \quad (4.14)$$

where  $\mathbf{e}_{i,n}$  is one of the  $n$  columns of the  $n \times n$  identity matrix, and  $\delta(t)$  is the Dirac delta function, or unit impulse [47]. Although this may appear to be a drastic loss of generality,

the set of inputs referred to above forms a basis for all possible inputs to the system. That is, if we know the response of the system to the above set of inputs, we can determine the response to any set of inputs. To simplify the notation further, we use matrix notation to look at the complete set of inputs and corresponding outputs at once, by allowing  $\mathbf{u}(t)$  to be a matrix given by

$$\mathbf{u}(t) = \mathbf{I}\delta(t) \quad (4.15)$$

Given this specific, well-defined set of inputs, the solution to (4.7) is seen to be

$$\mathbf{H}(t) \equiv \mathbf{x}(t) = e^{\mathbf{A}t}\mathbf{B} \quad (4.16)$$

and is termed the *system impulse response*. Furthermore, it is seen that samples of the impulse response may be computed according to

$$\begin{aligned} \mathbf{H}_n \equiv \mathbf{H}(nT) &= e^{\mathbf{A}nT}\mathbf{H}_{n-1} \\ \mathbf{H}_0 &= \mathbf{B} \end{aligned} \quad (4.17)$$

That is, that matrices  $\mathbf{A}$  and  $\mathbf{B}$ , which completely describe the continuous-time system, are also sufficient to describe the corresponding sampled-time system. Furthermore, any sample  $\mathbf{H}_n$  of the impulse response (after the initial sample) may be calculated from knowledge of the previous sample,  $\mathbf{H}_{n-1}$ , and the sampling interval. The initial sample  $\mathbf{H}_0$  is simply given by the matrix  $\mathbf{B}$ .

#### 4.6 Frequency-Domain Representation of Continuous-Time Systems

Although the modeling algorithms to be presented in this dissertation are based on measured time-domain data, there are certain aspects of systems which are more simply stated in the frequency domain. In order to be able to move easily between these two representations of the same system, it is helpful to explicitly show how they are related math-

ematically. References [44] and [47] both contain material on this subject, although [44] has the most information from a state-variable point of view.

The Laplace transform  $F(s)$  of a scalar function  $f(t)$  is defined as

$$F(s) = \int_{-\infty}^{\infty} f(t) e^{-st} dt \quad (4.18)$$

where  $s$  is a complex frequency variable. Since a function such as  $\mathbf{H}(t)$  in equation (4.16) is actually a matrix whose elements are scalar functions such as  $f(t)$  above, it will be useful to develop a concise notation to handle the Laplace transformation of matrix functions of this type.

Since the Laplace transform of a matrix is computed as the element-by-element transformation of each entry of the matrix, it can be shown that the transform of equation (4.16), restated here as

$$\mathbf{H}(t) = e^{\mathbf{A}t} \mathbf{B} \quad (4.19)$$

is given by

$$\mathbf{H}_L(s) = \mathcal{L}\{e^{\mathbf{A}t} \mathbf{B}\} = (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \quad (4.20)$$

where  $\mathbf{I}$  is the identity matrix. Equation (4.20) is a remarkably simple and concise representation of the frequency-domain behavior of a potentially large linear system of constant-coefficient differential equations, which makes it a useful relationship to bear in mind when performing state variable analysis on such systems.

For purposes of illustration, we note that the same result may be achieved starting with equation (4.7) and using the fact that the relationship

$$F(s) = \mathcal{L}\{f(t)\} \quad (4.21)$$

implies that

$$sF(s) = \mathcal{L} \left\{ \frac{d}{dt} f(t) \right\} \quad (4.22)$$

Application of the above relationships to equation (4.7) results in

$$s\mathbf{X}(s) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \quad (4.23)$$

which may be rearranged as

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{B}\mathbf{U}(s) \quad (4.24)$$

Combination of (4.24) and (4.20) results in

$$\mathbf{X}(s) = \mathbf{H}_L(s) \mathbf{U}(s) \quad (4.25)$$

The above relationship (4.25) illustrates a classic concept of system theory, that the frequency-domain outputs of the system are related directly to the frequency-domain inputs of the system *via multiplication*. This is significantly simpler than equations (4.7) or (4.8), which require either differentiation or integration to compute outputs directly from inputs in the time domain.

Finally, to close the loop, we note that the impulse response is the response that corresponds to the time domain inputs of equation (4.15), which transforms into the frequency domain as

$$\mathbf{U}(s) = \mathbf{I} \quad (4.26)$$

Substitution of this set of inputs into (4.24) results in (4.20).

#### 4.7 Variable Definitions for the Modeling Problem at Hand

We now take a step from the derivations above, which for the most part represent general system theory, to a more specific set of equations and variables developed to most simply represent the problem at hand. The intent is to choose a set of variables in which each is easily associated with the physical quantity or concept it represents.

As discussed in Section 3.1, the problem will be solved in terms of scattering parameters. For this reason, subsequent derivations leading to the final modeling algorithm will be done in terms of scattering variables. For a review of scattering parameters as they relate to circuit theory and design, the reader is referred to [49] or any number of texts on general circuit theory.

We begin our change in variables by defining the following two substitutions:

$$\begin{aligned} \mathbf{A} &\rightarrow -\mathbf{C}^{-1}\mathbf{G} \\ \mathbf{B} &\rightarrow \mathbf{C}^{-1} \end{aligned} \tag{4.27}$$

This allows us to restate (4.7) as

$$\mathbf{C}\dot{\mathbf{S}}(t) + \mathbf{G}\mathbf{S}(t) = \mathbf{U}(t) \tag{4.28}$$

where we continue to use the matrix notation introduced in Section 4.5 for both the inputs  $\mathbf{U}(t)$  and the state variables  $\mathbf{S}(t)$ . The inputs  $\mathbf{U}(t)$  are incident voltage waves, and the state variables  $\mathbf{S}(t)$  (the outputs) are reflected voltage waves. The matrices  $\mathbf{C}$  and  $\mathbf{G}$  are constant. Although no formal proof will be presented in this dissertation, the above equation is sufficient to represent any circuit made up of lumped resistors, inductors, capacitors, and coupled inductors.<sup>†</sup> Such a circuit will hereafter be referred to as an *RLCM Circuit*. Typically the number of rows and columns in  $\mathbf{G}$  and  $\mathbf{C}$  need not be greater than the number of dynamic elements (*i.e.*, inductors and capacitors) in the circuit [44].

---

<sup>†</sup> A simple proof can be constructed from the basic stamping procedure for Modified Nodal Analysis described in [5] and [6].

Again restricting the inputs to  $\mathbf{U}(t) = \delta(t) \mathbf{I}$  and following the derivation of Section 4.5, we arrive at the equation for the system impulse response

$$\mathbf{S}(t) = e^{\mathbf{A}t} \mathbf{C}^{-1} \quad (4.29)$$

where

$$\mathbf{A} = -\mathbf{C}^{-1} \mathbf{G} \quad (4.30)$$

Furthermore, it is seen that samples of the system impulse response may be computed from

$$\mathbf{S}[n] = \mathbf{S}(nT) = e^{\mathbf{A}nT} \mathbf{C}^{-1} \quad (4.31)$$

which may be restated as

$$\mathbf{S}[n] = \mathbf{Q}^n \mathbf{C}^{-1} \quad (4.32)$$

where

$$\mathbf{Q} = e^{\mathbf{A}T} \quad (4.33)$$

and  $\mathbf{Q}$  is known as the *state matrix* of the sampled system in the same way that  $\mathbf{A}$  is the state matrix of the corresponding continuous-time system. Finally, the relationship expressed in (4.32) and (4.33) above may be stated recursively as

$$\mathbf{S}[n+1] = \mathbf{Q} \mathbf{S}[n], \quad \mathbf{S}[0] = \mathbf{C}^{-1} \quad (4.34)$$

which shows not only that  $\mathbf{Q}$  and  $\mathbf{C}$  completely characterize the impulse response of the sampled system, but that any sample  $\mathbf{S}[n]$  can be computed given the state matrix  $\mathbf{Q}$  and the previous sample  $\mathbf{S}[n-1]$ .

In the Laplace domain, application of (4.27) to equation (4.20) results in

$$\mathbf{H}_L(s) = (\mathbf{G} + s\mathbf{C})^{-1} \quad (4.35)$$

which provides a very simple and concise representation of the frequency-domain behavior of the system. It is worth restating that the above impulse response function provides sufficient information to compute the frequency response of the system to any and all frequency-domain inputs by direct multiplication.

#### 4.8 Basis Transformations

A well-known concept of state-variable theory is that of changing the basis from which the output vector is to be computed. In circuit theory, this corresponds to choosing what set of state variables is going to be used in the analysis of the circuit in question, since any variable set which supports all of the modes observed in the output vector is mathematically as good as any other set. In practice, however, one set may be more easily manipulated than another, or perhaps contain fewer redundancies, in which case the utility of basis transformation may be clearly seen.

To effect a basis transformation on the system represented by (4.35), we assume that only an  $m \times m$  portion of  $\mathbf{H}_L(s)$  is to be observed, presumably at the  $m$  ports of the network, and is therefore called the *port variables*. The remainder of  $\mathbf{H}_L(s)$  is assumed to be unimportant and unmeasurable since it is not accessible by any port, and is therefore termed the *internal variables*. We further assume that the matrices are ordered such that the port variables reside in the upper left-hand corner, which allows us to partition the problem according to the notation

$$\mathbf{H}_L(s) = \begin{bmatrix} \mathbf{H}_p(s) & \mathbf{H}_{pi}(s) \\ \mathbf{H}_{ip}(s) & \mathbf{H}_i(s) \end{bmatrix} \quad (4.36)$$



where as stated previously, the port variables reside in the upper left-hand corner of the matrix. Our goal in applying a basis transformation is to change the internal variables  $\mathbf{H}_i(s)$  without affecting the port variables  $\mathbf{H}_p(s)$ .

We continue by partitioning the matrices  $\mathbf{G}$  and  $\mathbf{C}$  in a similar manner to (4.36), which results in

$$\begin{bmatrix} \mathbf{H}_p(s) & \mathbf{H}_{pi}(s) \\ \mathbf{H}_{ip}(s) & \mathbf{H}_i(s) \end{bmatrix} = \left( \begin{bmatrix} \mathbf{G}_p & \mathbf{G}_{pi} \\ \mathbf{G}_{ip} & \mathbf{G}_i \end{bmatrix} + s \begin{bmatrix} \mathbf{C}_p & \mathbf{C}_{pi} \\ \mathbf{C}_{ip} & \mathbf{C}_i \end{bmatrix} \right)^{-1} \quad (4.37)$$

If we now introduce two matrices  $\mathbf{U}$  and  $\mathbf{V}$  of the general form

$$\begin{aligned} \mathbf{U} &= \begin{bmatrix} \mathbf{I} & \mathbf{U}_{pi} \\ 0 & \mathbf{U}_i \end{bmatrix} \\ \mathbf{V} &= \begin{bmatrix} \mathbf{I} & 0 \\ \mathbf{V}_{ip} & \mathbf{V}_i \end{bmatrix} \end{aligned} \quad (4.38)$$

it follows directly that their inverses will be of the same general form,

$$\begin{aligned} \mathbf{U}^{-1} &= \begin{bmatrix} \mathbf{I} & \tilde{\mathbf{U}}_{pi} \\ 0 & \tilde{\mathbf{U}}_i \end{bmatrix} \\ \mathbf{V}^{-1} &= \begin{bmatrix} \mathbf{I} & 0 \\ \tilde{\mathbf{V}}_{ip} & \tilde{\mathbf{V}}_i \end{bmatrix} \end{aligned} \quad (4.39)$$

If the matrices  $\mathbf{U}^{-1}$  and  $\mathbf{V}^{-1}$  are used to transform the impulse response  $\mathbf{H}_L(s)$  into a new system according to

$$\mathbf{H}'_L(s) = \mathbf{V}^{-1} \mathbf{H}_L(s) \mathbf{U}^{-1} \quad (4.40)$$

the port behavior of the new system will be the same as that of the original system. We further note that the same transformation can be achieved by operating directly on the matrices  $\mathbf{G}$  and  $\mathbf{C}$ , such that

$$\mathbf{H}'_L(s) = (\mathbf{G}' + s\mathbf{C}')^{-1} \quad (4.41)$$

where  $\mathbf{G}'$  and  $\mathbf{C}'$  are given by

$$\begin{aligned} \mathbf{G}' &= \mathbf{UGV} \\ \mathbf{C}' &= \mathbf{UCV} \end{aligned} \quad (4.42)$$

That is, if two matrices of the form shown in (4.38) are used to transform the matrices  $\mathbf{G}$  and  $\mathbf{C}$  as shown in (4.42), the variables of interest, the port variables, remain unchanged. In terms of system theory, the system is unchanged as observed through its ports.

An important point not explicitly stated in the preceding exercise is that  $\mathbf{U}$  and  $\mathbf{V}$  must have full column rank, or else the pencil  $(\mathbf{G}' + s\mathbf{C}')$  in (4.41) will be singular. Furthermore, in order for the port behavior to be preserved,  $\mathbf{U}$  and  $\mathbf{V}$  must be square. Although this was assumed to be the case in the preceding manipulations, the use of rectangular matrices  $\mathbf{U}$  and  $\mathbf{V}$  can be very useful, and actually forms the basis for all of the network reduction schemes presented in [50] and other related works.

#### 4.9 Summary

In this chapter we have developed a basis of circuit and system theory in order to support the subsequent derivation of the interconnect model extraction algorithm which is the subject of this dissertation. Initially the concept of a system was introduced, along with the linearity and time invariance properties. A derivation was then sketched to show the connection between a system of coupled constant-coefficient differential equations in the continuous-time domain and a system of coupled difference equations in the sampled-time domain. The Laplace transform was introduced, and the continuous-time domain results were transformed into the Laplace domain, since some aspects of a system are

more concisely stated in that domain. In Section 4.7, each of the above results in the various mathematical domains was translated into a set of state variables appropriate to the problem at hand. The purpose of the change in variables and nomenclature was to improve the reader's intuitive understanding of the extraction algorithm to follow. Finally, an approach was outlined for applying a transformation to a system's state space which changes the supporting basis of the port variables without affecting the port variables themselves. The concepts and derivations of this chapter leading to equations (4.35) and (4.41) constitute a sufficiently broad and stable mathematical platform from which we may now launch our assault on the interconnect modeling problem.

## Chapter 5

### *AN ALGORITHM FOR ROBUST INTERCONNECT MODEL EXTRACTION FROM TIME-DOMAIN MEASUREMENTS*

The previous chapter described how a set of linear constant-coefficient differential equations can be used to model a linear system, and how such a set of equations may be represented in matrix form. Furthermore, an associated system was derived whose inputs and outputs are samples of such a system. The most significant result of the chapter, in the context of this work, was that the matrices  $\mathbf{Q}$  and  $\mathbf{C}$  of equation (4.28) completely characterize a sampled system.

In this chapter an algorithm is presented for robustly extracting a continuous-time model of such a system based on samples of its time-domain response. The initial goal of the modeling algorithm will be to determine  $\mathbf{Q}$  and  $\mathbf{C}$ , thereby characterizing the associated sampled system. Having done so, the matrix  $\mathbf{G}$  may be computed by inversion of equations (4.30) and (4.33), such that  $\mathbf{G}$  and  $\mathbf{C}$  define the continuous-time model. The matrices  $\mathbf{G}$  and  $\mathbf{C}$  may be further optimized before settling on a final model, and the model is at last translated into a SPICE netlist form which is compatible with standard industry simulation tools.

### 5.1 *A Robust Algorithm*

Robustness is one of the most important aspects of any CAD application. This is true for both modeling, as described in this dissertation, and for synthesis, the process by which a circuit is designed or laid out. Probably the first thought that comes to mind at the suggestion of robustness is a program that does not terminate unexpectedly in response to an unexpected set of inputs. This form of robustness is primarily aesthetic in value, occurring for the most part at the user-interface level. Because it is of practical importance to an end user, it is *very* important to an entity attempting to market such software. Even in the academic environment, a certain level of aesthetic robustness is necessary if it is the intent of the author to have others use the code. However, this form of robustness is not the primary aim in software written for the purposes of algorithmic research and development.

Of far greater importance from a technical viewpoint is the reliability and robustness of the underlying numerical approach upon which the software is built. Since the primary purpose of CAD is to automate a process which is difficult or time-consuming when performed manually, it is of great importance that the user is able to trust the tool to return accurate outputs. If the user cannot trust the outputs returned by the tool, he or she must embark on the difficult and time-consuming task of monitoring the outputs for correctness, thereby defeating the original purpose of the tool. As a result, a software package which runs as desired but returns totally or even partially inaccurate results is even more useless than one which has the annoying tendency to terminate of its own volition.

In the analysis of a numerical approach to solving a problem, the issue of robustness is usually addressed by questions of accuracy and predictability. One aspect of accuracy is the significance of any mathematical approximations which are incurred during the solution. This is especially important when continuous-time systems are discretized for analysis on a digital computer. The other aspect of the accuracy of an approach is closely related to its conditioning—its sensitivity to small variations in inputs. A well-conditioned approach should exhibit small variations in its outputs in response to small variations in its inputs. One which behaves otherwise is considered unpredictable, and it becomes difficult to ascertain which result, if any, is correct.

Finally, it is necessary to mention the conditioning of a basic software algorithm, also referred to as its stability. A stable algorithm is one which behaves predictably for well-conditioned inputs. Many stable numerical algorithms are employed in this work in the process of extracting a model. However, as will be seen in this chapter, the order of application of each algorithm within the entire modeling approach is very important so that each is assured of receiving well-conditioned inputs and therefore behaving predictably. As the implementation of the modeling approach is presented in the following sections, it will be seen that by careful design, the overall predictability of the approach is maintained even though poorly-conditioned matrices are routinely encountered internally.

## 5.2 Problem Formulation

Derivation of the modeling algorithm begins with the final result of Section 4.7,

$$\mathbf{S}[n+1] = \mathbf{Q}\mathbf{S}[n], \quad \mathbf{S}[0] = \mathbf{C}^{-1} \quad (5.1)$$

It was noted in Section 4.7 that the size of the matrices  $\mathbf{G}$  and  $\mathbf{C}$  of (4.28) is determined by the number of dynamic elements in the circuit. It is easily shown that the size of these two matrices carries over to become the size of the matrix  $\mathbf{Q}$  in (5.1). However, the basic premise of this research is that we may extract a model from a set of measurements taken at the ports of a circuit, and that the model will be highly independent of the internal details of the circuit. This presents a discrepancy, since a set of raw waveforms measured at the ports of a device offers very little insight into how many dynamic elements are present. Furthermore, since many of those elements will typically be internal to the system itself and therefore not accessible from the ports, we have very little information if any as to the unmeasurable state variable waveforms at each internal element. In fact, we may be almost certain that no ideal lumped element even exists in the physical circuit! It is apparent that due to this lack of information, (5.1) may not be solved directly for  $\mathbf{Q}$ .

However, since the model is desired to be highly independent of this information, we are free to take the alternate approach of not caring what physical waveforms are occurring internal to the circuit, but matching only the port behavior. Since we are no longer

required to replicate an unknown and unmeasurable set of waveforms, the derivation may proceed, albeit with a large number of degrees of freedom, since the internal waveforms may now be anything which is convenient.

Since the waveforms occurring at the ports are known from measurement, while those occurring internal to the device are not, the matrix of scattering waveforms  $\mathbf{S}[n]$  is ordered and symbolically partitioned. This partitioning simplifies the derivation by allowing independent manipulation of port waveforms and internal waveforms (*i.e.*, measurable and unmeasurable waveforms). The physical differentiation between port and internal waveforms is reflected by the notation

$$\mathbf{S}[n] = \begin{bmatrix} \mathbf{S}_p[n] & \mathbf{S}_3[n] \\ \mathbf{S}_i[n] & \mathbf{S}_4[n] \end{bmatrix} \quad (5.2)$$

It should be emphasized that each of the elements of  $\mathbf{S}[n]$  is a waveform. In the partitioning above, the submatrix  $\mathbf{S}_p[n]$  contains the waveforms measured at the ports of the device. Because the waveforms present in the other three submatrices cannot be accessed for measurement through any port, they are unmeasurable and are therefore unknown. However, as stated previously, we are free to choose any convenient values for these waveforms, and it turns out that the model derivation (and subsequent implementation) is accomplished more simply by waiting until later in the algorithm to do so. For this reason we put off the determination of the waveforms represented by the submatrices  $\mathbf{S}_3[n]$  and  $\mathbf{S}_4[n]$  by noting that the equality in (5.1) holds even if they are dropped. With this fact in mind, we may restate (5.1) as

$$\begin{bmatrix} \mathbf{S}_p[n+1] \\ \mathbf{S}_i[n+1] \end{bmatrix} = \mathbf{Q} \begin{bmatrix} \mathbf{S}_p[n] \\ \mathbf{S}_i[n] \end{bmatrix} \quad (5.3)$$

which allows the derivation to continue under a simplified notation.

In order to solve for the matrix  $\mathbf{Q}$ , we must remove from (5.3) the dependence on  $n$ , and this is achieved by solving (5.3) separately for each sample point. Assuming that each of our measured waveforms has  $N$  samples, the following system of equations may be developed:

$$\begin{bmatrix} S_p[1] & S_p[2] & \dots & S_p[N-1] \\ S_i[1] & S_i[2] & \dots & S_i[N-1] \end{bmatrix} = \mathbf{Q} \begin{bmatrix} S_p[0] & S_p[1] & \dots & S_p[N-2] \\ S_i[0] & S_i[1] & \dots & S_i[N-2] \end{bmatrix} \quad (5.4)$$

For notational convenience, (5.4) may be restated as

$$\begin{bmatrix} \mathbf{S}_{1,p} \\ \mathbf{S}_{1,i} \end{bmatrix} = \mathbf{Q} \begin{bmatrix} \mathbf{S}_{0,p} \\ \mathbf{S}_{0,i} \end{bmatrix} \quad (5.5)$$

or even more succinctly as

$$\mathbf{S}_1 = \mathbf{Q}\mathbf{S}_0 \quad (5.6)$$

Despite the simplicity of the notation in (5.6), the fact remains that the entirety of the remaining internal waveforms, represented at each sample point by  $\mathbf{S}_{0,i}$  and  $\mathbf{S}_{1,i}$ , are unknown. This fact leaves us still unable to solve directly for  $\mathbf{Q}$ , since potentially sizable portions of  $\mathbf{S}_0$  and  $\mathbf{S}_1$  are unknown. However, we once again appeal to the fact that the internal waveforms may take on any convenient values since they cannot be measured and therefore cannot be rigorously characterized. That is, it is not at all necessary to know or characterize the physical internal waveforms of the device, for the purpose of modeling a circuit for simulation, and even the concept of internal waveforms of a physical circuit is ambiguous, as there are an infinite set to choose from. Although we were able previously to postpone the determination of those waveforms corresponding to the submatrices  $\mathbf{S}_3[n]$  and  $\mathbf{S}_4[n]$  in (5.2), the topology of the problem does not allow us to postpone determination of the remaining internal waveforms. For this reason it is necessary to



choose an appropriate set of internal waveforms, by exercising the degrees of freedom to which we are entitled, before it is possible to solve for  $\mathbf{Q}$ .

### 5.3 Choice of Internal Waveforms

In the process of choosing a good set of internal waveforms for the model, we fall back on the two competing modeling criteria—*accuracy* and *complexity*. Accuracy requires that the model recreate the data used to extract it, and that the model extrapolate reasonably to predict data not used to create it. On the other hand, the complexity criterion dictates that the model should be as small as possible. It is no wonder that the two are in constant competition.

At this stage of the derivation, where all numbers are assumed known to infinite precision, we choose a solution which guarantees exact accuracy, while achieving minimal complexity subject to the accuracy constraint. As stated in the previous section, a number of degrees of freedom remain at our disposal, because we are unconcerned about the waveforms internal to our model, but are only interested in the port behavior. By choosing a particular solution to (5.4), we are in effect choosing a particular (and preferably convenient) set of internal waveforms to fill the unknown entries in (5.4). The solution we have chosen corresponds to the particular set of internal waveforms shown below, which causes the unknown values in (5.4) to be specifically filled as follows:

$$\begin{bmatrix} \mathbf{s}_p[1] & \mathbf{s}_p[2] & \dots & \mathbf{s}_p\left[\frac{N}{2}\right] \\ \mathbf{s}_p[2] & \mathbf{s}_p[3] & \dots & \\ & & \dots & \\ \mathbf{s}_p\left[\frac{N}{2}\right] & \mathbf{s}_p\left[\frac{N}{2}+1\right] & & \mathbf{s}_p[N-1] \end{bmatrix} = \mathbf{Q}' \begin{bmatrix} \mathbf{s}_p[0] & \mathbf{s}_p[1] & \dots & \mathbf{s}_p\left[\frac{N}{2}-1\right] \\ \mathbf{s}_p[1] & \mathbf{s}_p[2] & \dots & \mathbf{s}_p\left[\frac{N}{2}\right] \\ & & \dots & \\ \mathbf{s}_p\left[\frac{N}{2}-1\right] & \mathbf{s}_p\left[\frac{N}{2}\right] & & \mathbf{s}_p[N-2] \end{bmatrix} \quad (5.7)$$

In the above formulation,  $\mathbf{Q}'$  is the variant of  $\mathbf{Q}$  which corresponds to the chosen set of internal waveforms. As before, for notational convenience, we restate (5.7) as

$$\begin{bmatrix} \mathbf{S}'_{1,p} \\ \mathbf{S}'_{1,i} \end{bmatrix} = \mathbf{Q}' \begin{bmatrix} \mathbf{S}'_{0,p} \\ \mathbf{S}'_{0,i} \end{bmatrix} \quad (5.8)$$

or even more succinctly as

$$\mathbf{S}'_1 = \mathbf{Q}' \mathbf{S}'_0 \quad (5.9)$$

Equation (5.7), although its dimension is only  $\frac{N}{2} \times \frac{N}{2}$ , produces a  $\mathbf{Q}$  which guarantees that all  $N$  samples are matched by the model. A review of Prony's method [52], which applies to the one-port case in which each submatrix of (5.7) is a scalar, will verify that this is the minimal number of equations which must be satisfied in order to exactly recreate the original data. Generalization of that result to the multiport case is straightforward, given a basic understanding of linear algebra. Furthermore, for the one-port case of (5.7), in which  $\mathbf{S}_p[n]$  is a scalar function, the formulation of (5.7) coincides with that shown in [34].

A further look into the structure of the formulation in (5.7) will lend insight as to the origin of its properties—why it exactly recreates the original data and why it is in some sense minimal in size. To simplify the notation as we investigate the structure of (5.7), we assume that we are dealing with a one-port problem, in which  $\mathbf{S}_p[n]$  is a scalar function. The generalization of the results to the multiport case is straightforward.

If each row of  $\mathbf{S}'_0$  or  $\mathbf{S}'_1$  is viewed as a sequence of samples, it is seen that each row is actually one of the waveforms of the model response. For this reason, any of these rows may be thought of as a response waveform, and the two terms are used interchangeably in the following discussion. By examining the structure of (5.7), it can be seen that each row of  $\mathbf{S}'_{0,i}$  and  $\mathbf{S}'_{1,i}$  is identical to the row above it, but is shifted by one sample. Further-

more, If we refer to equation (4.32) we are reminded that we are dealing with samples of a sum of weighted exponentials. In fact, the following eigendecomposition of (4.32)

$$\mathbf{S}[n] = \mathbf{Q}^n \mathbf{C}^{-1} = \mathbf{W} \mathbf{\Lambda} \mathbf{W}^{-1} \mathbf{C}^{-1} \quad (5.10)$$

reveals that each element of  $\mathbf{S}[n]$  is of the form

$$s_{ij}[n] = k_{ij,1} \lambda_1^n + k_{ij,2} \lambda_2^n + \dots + k_{ij,M} \lambda_M^n \quad (5.11)$$

where  $[\lambda_1, \lambda_2, \dots, \lambda_M]$  are the eigenvalues of  $\mathbf{Q}$ , and  $M$  is the dimension of  $\mathbf{Q}$ . This being the case, the choice of time shifted versions of the port waveforms as internal waveforms is intuitively seen to be a wise one, since it can be shown that the time-shifted waveforms are represented by the same values of  $[\lambda_1, \lambda_2, \dots, \lambda_M]$ , but with different values of  $k$ . In short, no eigenvalues (that is, modes) are injected into the internal waveforms which are not already present in the port waveforms. It is this property that makes the model minimal in size, subject to the accuracy criteria.

Up to this point, our definition of minimal has purposely been left somewhat ambiguous. The observant reader will notice that the dimension of the problem represented by (5.7)-(5.9) is actually determined by the number of sample points in conjunction with the number of ports, whereas in Section 4.7 it was stated that the size of the system is closely related to the number of dynamic elements present. Obviously these two quantities should be independent, since measuring additional samples of a system's impulse response does not change the number of dynamic elements in the system. A more precise statement is that the formulation in (5.7) is minimal for a general, arbitrary set of port waveforms. However, this begs the question as to what set of port waveforms might be deemed arbitrary and what set might not.

As an example of a non-arbitrary set of port waveforms, consider a one-port circuit which contains eight dynamic elements. The dimension of  $\mathbf{G}$  and  $\mathbf{C}$  in (4.28) and therefore  $\mathbf{Q}$  in (4.32) is  $8 \times 8$ . Now assume that we can measure 256 data points of the system

impulse response. This corresponds to  $\mathbf{S}'_0$ ,  $\mathbf{S}'_1$ , and  $\mathbf{Q}'$  in (5.7) with a dimension of 128, which is obviously not minimal for our example problem. The discrepancy can be easily isolated by investigating the rank of  $\mathbf{S}'_0$ , which must be inverted to compute  $\mathbf{Q}'$ . It can be shown by use of the Cayley-Hamilton theorem [44] that only the first eight rows of  $\mathbf{S}'_0$  are linearly independent; that is,  $\mathbf{S}'_0$  is singular and the problem is underdetermined. This being known, only the first eight rows of (5.7) would be used to compute  $\mathbf{Q}'$ , which would then be  $8 \times 8$ .

The preceding example illustrates that if the samples used to populate  $\mathbf{S}'_0$  are exact samples of the impulse response of a specific RLCM circuit, then (5.7) will *not* correspond to a minimal solution. However, it must be emphasized that the purpose of this work is not to build a model from samples of an ideal mathematical RLCM circuit, but instead to develop a model from measured data by assuming that it is the response of such a circuit. This may seem like a minor difference to emphasize, but in fact it becomes significant when the presence of measurement noise is taken into account. Because of measurement noise and the basic distributed nature of real-world circuits, measured data is generally *not* comprised of exact samples of the impulse response of a specific RLCM circuit. Although this would seem to imply that direct inversion of (5.9) yields the desired result because  $\mathbf{S}'_0$  is no longer singular, this is only true in perfect, infinite-precision mathematics. In practice it is typical for the measurements to correspond quite closely to the response of an RLCM circuit, such that in finite-precision math,  $\mathbf{S}'_0$  is nonsingular but poorly conditioned, making it more difficult to handle than the singular case. The following section outlines the approach chosen to deal with these difficulties.

#### 5.4 Internal Waveform Subset Selection

The example of the preceding section raises several interesting mathematical issues with respect to what circumstances might lead to a singular or rank-deficient  $\mathbf{S}'_0$ . In the presence of machine roundoff error, mathematical questions regarding singularities invariably correspond to numerical questions regarding the conditioning [48] of the problem at

hand. Likewise, mathematical questions of linear dependence or rank deficiency typically spawn algorithmic questions of rank determination.

The problem at hand is no exception to this principle: although direct inversion of (5.9) using perfect math yields a  $\mathbf{Q}'$  which exactly recreates the original measured data used in model extraction, the same approach done in finite-precision math will typically yield a useless result. This is because the matrix  $\mathbf{S}'_0$ , although in general nonsingular, is typically very close numerically to a singular matrix, which defines it as ill conditioned.<sup>†</sup> Steps must be taken to deal with the ill conditioning of  $\mathbf{S}'_0$  prior its inversion, or the resulting  $\mathbf{Q}'$  may vary wildly depending on seemingly unrelated parameters such as the underlying architecture of the computer hardware being used for the computations. Clearly this type of variation would result in an unreliable and unpredictable modeling algorithm.

Before presenting the numerical techniques chosen to circumvent this difficulty, it is instructive to look at the significance of the internal waveforms from a numerical point of view. Returning to equations (5.7)–(5.9), and for that matter to (5.1), we see that  $\mathbf{Q}$  effects a linear transformation on the internal waveforms and port waveforms. This transformation results in the same set of waveforms, but each is shifted by one timepoint. That is, the internal waveforms must be such that some linear combination of them with the port waveforms results in the same set of waveforms, but shifted. For this reason  $\mathbf{Q}$  is often termed the *linear prediction matrix* in system identification literature.

As illustrated in the example of the previous chapter, the use of more samples than necessary of the exact response of an RLCM circuit results in an underdetermined problem, and some of the rows of  $\mathbf{S}'_0$  must be removed in order to invert the equation. Likewise, if  $\mathbf{S}'_0$  is populated using measured data and the problem is to be solved in the presence of roundoff error, the problem is *numerically* underdetermined, and the redundant row space of  $\mathbf{S}'_0$  must likewise be removed before the problem can be solved. The approach taken in this work is to choose from the set of internal waveforms in (5.8) a lin-

<sup>†</sup> Discussions of almost every aspect and definition of ill conditioning may be found in reference [48].

ear combination which is sufficient to nearly satisfy the equality, without introducing unacceptable error in the process. This aspect of the modeling algorithm is called *preconditioning*, since it improves the condition of  $\mathbf{S}'_0$ .

Intuitively, the solution approach proceeds as follows. It can be shown that  $\mathbf{S}'_0$  spans  $\mathbf{S}'_1$ . If a subset of the rows of  $\mathbf{S}'_0$  can be found which nearly spans  $\mathbf{S}'_0$ , then that subset will also nearly span  $\mathbf{S}'_1$ . If this is the case, the matrix  $\mathbf{Q}'$  may be replaced by a smaller matrix whose job it is to transform (premultiply) the reduced number of rows in the selected subset. The approach just outlined can be stated in a more mathematical notation as follows:

1. Given  $\mathbf{S}'_1 = \mathbf{Q}'\mathbf{S}'_0$

$$\text{where } \mathbf{S}'_1 = \begin{bmatrix} \mathbf{S}'_{1,p} \\ \mathbf{S}'_{1,i} \end{bmatrix} \quad \mathbf{S}'_0 = \begin{bmatrix} \mathbf{S}'_{0,p} \\ \mathbf{S}'_{0,i} \end{bmatrix}$$

2. Find  $\tilde{\mathbf{S}}_0 = \begin{bmatrix} \tilde{\mathbf{S}}_{0,p} \\ \tilde{\mathbf{S}}_{0,i} \end{bmatrix} = \mathbf{W}\mathbf{S}'_0$  and  $\tilde{\mathbf{S}}_1 = \begin{bmatrix} \tilde{\mathbf{S}}_{1,p} \\ \tilde{\mathbf{S}}_{1,i} \end{bmatrix} = \mathbf{W}\mathbf{S}'_1$

such that  $\text{span}(\tilde{\mathbf{S}}_0) \approx \text{span}(\mathbf{S}'_0)$

and therefore  $\text{span}(\tilde{\mathbf{S}}_0) \approx \text{span}(\mathbf{S}'_1)$

and finally  $\text{span}(\tilde{\mathbf{S}}_0) \approx \text{span}(\tilde{\mathbf{S}}_1)$

3. Furthermore, we require  $\tilde{\mathbf{S}}_{0,p} \approx \mathbf{S}'_{0,p}$  and  $\tilde{\mathbf{S}}_{1,p} \approx \mathbf{S}'_{1,p}$  to approximately preserve the port behavior.

Many approaches have been suggested to solve problems similar to that outlined above, which amounts to isolating the effective null space of  $\mathbf{S}'_0$  and throwing it out. The most typical approaches involve QR decomposition with column pivoting or singular value decomposition [48] to determine an appropriate matrix  $\mathbf{W}$ . The solution to be presented in this work actually uses both algorithms and is outlined in [48] as *subset selec-*

*tion*—finding a subset of the rows of a matrix which span nearly the same space as the matrix itself. This solution was chosen from the set of available solutions for presentation since it leads to the most straightforward formulation of the problem, while in typical modeling situations (that is, excepting completely degenerate cases) its results are as accurate as the more complicated methods.

Subset selection actually places additional constraints on the matrix  $\mathbf{W}$  above by requiring that the rows of  $\tilde{\mathbf{S}}_0$  be a subset of the rows of  $\mathbf{S}'_0$ , as opposed to a general linear combination as suggested in the above problem statement. The matrix  $\mathbf{W}$  is determined as follows:

1. Compute the SVD of  $\mathbf{S}'_0$  such that  $\mathbf{S}'_0 = \mathbf{U}\mathbf{S}\mathbf{V}^T$ .
2. Estimate the rank  $R$  of  $\mathbf{S}'_0$  from its singular values.
3. Perform QR factorization with column pivoting on  $\mathbf{U}^T$ , and note the indices of the first  $R$  columns to be chosen.
4. Build  $\tilde{\mathbf{S}}_0$  from the  $R$  rows of  $\mathbf{S}'_0$  which are chosen from step 3. The matrix  $\mathbf{W}$  is implicitly the rectangular permutation matrix which accomplishes the selection.

It is shown in [48] that the smallest singular value of  $\tilde{\mathbf{S}}_0$ ,  $\sigma_R(\tilde{\mathbf{S}}_0)$ , is bounded by

$$\frac{\sigma_R(\mathbf{S}'_0)}{\|\tilde{\mathbf{U}}_{11}^{-1}\|_2} \leq \sigma_R(\tilde{\mathbf{S}}_0) \leq \sigma_R(\mathbf{S}'_0) \quad (5.12)$$

where the notation  $\sigma_R(\ )$  represents the  $R^{th}$  singular value of the corresponding matrix. The matrix  $\tilde{\mathbf{U}}_{11}$  is the upper left  $R \times R$  submatrix of

$$\mathbf{W}^T \mathbf{U} = \begin{bmatrix} \tilde{\mathbf{U}}_{11} & \tilde{\mathbf{U}}_{12} \\ \tilde{\mathbf{U}}_{21} & \tilde{\mathbf{U}}_{22} \end{bmatrix} \quad (5.13)$$

One aspect of the above algorithm is the determination of  $R$ , the rank of  $\mathbf{S}'_0$ . This was accomplished by setting  $R$  to the number of singular values of  $\mathbf{S}'_0$  which are greater than the noise floor exhibited in the data. This causes high-frequency noise, which is the primary contribution of the excess internal waveforms, to be ignored in the modeling process. It is this noise which, if it is allowed to remain in the formulation, causes asymptotic instability and inaccuracy in the resulting model. The noise floor was determined by applying the fast Fourier transform (FFT) to the input waveforms, and analyzing the data in the frequency domain. The amount of noise present in a set of de-embedded measurements is a result of many factors, including the quality of the measurement system, the bandwidth of the excitation signal, and the amount of filtering applied to the data prior to extraction.

The effectiveness of step 3 above is that QR factorization with column pivoting tends to produce a well-conditioned  $\tilde{\mathbf{U}}_{11}$ , which implies that  $\|\tilde{\mathbf{U}}_{11}^{-1}\|_2$  is small, since  $\|\tilde{\mathbf{U}}_{11}^{-1}\|_2$  is actually just the inverse of the smallest singular value of  $\tilde{\mathbf{U}}_{11}$ . This results in a large enough upper bound on  $\sigma_R(\tilde{\mathbf{S}}_0)$  that it is certain to be well conditioned. Care must be taken in performing the QR factorization that the first  $m$  rows (corresponding to the waveforms measured at the  $m$  ports of the multiport) are selected, so that the port behavior is preserved. This is appropriate since in this step we are choosing a set of *internal* waveforms, while the port waveforms have already been determined by measurement.

To complete the solution for  $\mathbf{Q}$ , we formulate the problem as

$$\tilde{\mathbf{S}}_1 \approx \tilde{\mathbf{Q}}\tilde{\mathbf{S}}_0 \quad (5.14)$$

where generally equality cannot be obtained because  $\tilde{\mathbf{S}}_0$  and  $\tilde{\mathbf{S}}_1$  are rectangular, causing the problem to be overdetermined. As a result, (5.14) is solved for  $\tilde{\mathbf{Q}}$  by least squares methods to minimize the residual column norms.



### 5.5 Final Identification of the Sampled Model—Determination of $\mathbf{Q}$ and $\mathbf{C}$

The question now arises as to whether the steps outlined in the previous three sections have led us any closer to the model we are seeking—that is, the matrices  $\mathbf{Q}$  and  $\mathbf{C}$  of equation (5.1). As it turns out, the majority of the computational work required to resolve the sampled-time model has been done. However, some additional steps required to finalize the model are outlined in this section.

Having solved for  $\tilde{\mathbf{Q}}$ , it is now substituted into equation (5.1) in place of  $\mathbf{Q}$ , the  $M \times M$  linear prediction matrix of (5.1). As a reminder, not only the elements of  $\mathbf{Q}$ , but also its size, were unknown, so we are free to assign it the values and dimension of  $\tilde{\mathbf{Q}}$  at this point. (Similarly, the size and values of  $\mathbf{S}[n]$  were also unknown, except for its port behavior.) However, some error is incurred in the process of estimating  $\tilde{\mathbf{Q}}$  from  $\mathbf{Q}'$ , so that in general the equality of (5.1) cannot be fulfilled exactly, and the recursion relation must be rewritten as

$$\mathbf{S}[n+1] \approx \tilde{\mathbf{Q}}\mathbf{S}[n] \quad (5.15)$$

However, the response of the model is typically of greater interest, and is given by

$$\tilde{\mathbf{S}}[n+1] = \tilde{\mathbf{Q}}\tilde{\mathbf{S}}[n] \quad (5.16)$$

where  $\tilde{\mathbf{S}}[n]$  is a perturbed version of  $\mathbf{S}[n]$ .

Although the majority of the computational work is finished,  $\mathbf{C}^{-1}$  must also be determined in order to completely characterize the system. We begin by noting from (5.1) that  $\mathbf{C}^{-1} = \mathbf{S}[0]$ . However, an  $\mathbf{S}[0]$  in the strictest sense, *i.e.* one which satisfies (5.1) exactly, is not known, because  $\mathbf{S}[n]$  is not known in its entirety. This is because, as stated previously, the internal waveforms are unmeasurable and therefore unknown. By comparing equations (5.1) and (5.16) we see that the correct choice is  $\mathbf{C}^{-1} = \tilde{\mathbf{S}}[0]$ , although we must still contend with the fact that a portion of  $\tilde{\mathbf{S}}[0]$  is as of yet unknown. Returning to the notation of equation (5.2), we see that  $\tilde{\mathbf{S}}_p[0]$ , the port waveforms at

$t = 0$ , are known from measurement. Furthermore,  $\tilde{\mathbf{S}}_i[0]$  has been decided in the process of choosing our internal waveforms, which was accomplished through the problem formulation shown in equation (5.7) and through the subset selection process outlined in this section. This leaves us with  $\tilde{\mathbf{S}}_3[0]$  and  $\tilde{\mathbf{S}}_4[0]$  still to be determined, since we explicitly postponed determination of  $\mathbf{S}_3[n]$  and  $\mathbf{S}_4[n]$  at the beginning of the derivation. If we recall that these two submatrices of waveforms are arbitrary with respect to the port waveforms, it is straightforward to show that the corresponding submatrices of  $\mathbf{C}^{-1}$  are also arbitrary. Once again, we are left with the task of choosing an appropriate subset of the internal waveforms. Due to numerical considerations, it is wise to choose a set which preserves the conditioning of the matrix, if for no other reason than the fact that we will later want to compute  $\mathbf{C}$  from  $\mathbf{C}^{-1}$ .

Given these constraints, it was decided to choose  $\mathbf{C}^{-1}$  according to

$$\mathbf{C}^{-1} = \begin{bmatrix} \tilde{\mathbf{S}}_p[0] & (\tilde{\mathbf{S}}_i[0])^T \\ \tilde{\mathbf{S}}_i[0] & \mu \mathbf{I} \end{bmatrix}, \quad \mu = \|\tilde{\mathbf{S}}_i[0]\|_2 \quad (5.17)$$

The particular choice of the scaling factor  $\mu$  in (5.17) serves two purposes. First, it scales  $\mu \mathbf{I}$  small enough that important information in  $\tilde{\mathbf{S}}_i[0]$  is not dominated and subsequently lost. Second,  $\mu \mathbf{I}$  is scaled large enough that its own pertinent information is not dominated by  $\tilde{\mathbf{S}}_i[0]$  and therefore lost. Both of these properties work together to maintain the conditioning of the matrix. It should also be noted that the entire subset selection process of Section 5.4 tends to cause the columns of  $\tilde{\mathbf{S}}_i[0]$  to be fairly linearly independent, so that even if  $\tilde{\mathbf{S}}_p[0] = 0$ , (which is typically the case when ideal delay is present in a system) the matrix is still well conditioned.

As a final note, the first  $m$  columns of  $\mathbf{C}^{-1}$ , which is composed of  $\tilde{\mathbf{S}}_p[0]$  and  $\tilde{\mathbf{S}}_i[0]$ , are identical to the first  $m$  columns of  $\tilde{\mathbf{S}}_0$ . From an implementational perspective, this allows a simple translation from the preconditioned  $\tilde{\mathbf{S}}_0$  matrix to the modeling matrix  $\mathbf{C}^{-1}$ , which may then be inverted in place to compute  $\mathbf{C}$ .

At this point, the sampled system which we sought to identify, described by equation (5.1), has been approximated by another sampled system, represented by the equations

$$\tilde{\mathbf{S}}[n+1] = \tilde{\mathbf{Q}}\tilde{\mathbf{S}}[n], \quad \tilde{\mathbf{S}}[0] = \mathbf{C}^{-1} \quad (5.18)$$

and this system has been completely identified by  $\tilde{\mathbf{Q}}$  and  $\mathbf{C}$ . However, a sampled system such as that of (5.18) is not particularly compatible with industry standard simulators, which prefer a continuous-time model as described in equation (4.28). Some transformations are therefore necessary to achieve compatibility with existing simulation tools. These transformations are the subject of the next section.

### 5.6 Transformation to Continuous Time

To understand how the sampled-time model of (5.18) is converted to a continuous-time model such as that in equation (4.28), it is necessary to return to the derivation of Section 4.7 and invert equations (4.30) and (4.33). This results in the relations

$$\mathbf{A} = \frac{\log(\mathbf{Q})}{T} \quad (5.19)$$

and

$$\mathbf{G} = -\mathbf{CA} \quad (5.20)$$

which may be used for conversion to continuous time. (The function  $\log(\mathbf{Q})$  refers to the matrix logarithm of  $\mathbf{Q}$ , and will be discussed shortly.) For the purposes of the remaining discussion, we drop the tilde notation of the previous section, and instead outline how a general system is converted from the sampled-time domain to the continuous-time domain, as opposed to the specific system of Sections 5.4 and 5.5.

We begin by noting that application of (5.20) to compute  $\mathbf{G}$  is straightforward once  $\mathbf{C}$  and  $\mathbf{A}$  have been determined. Since computation of  $\mathbf{C}$  was described in the preceding section, we devote the remainder of this section to the determination of  $\mathbf{A}$ .

The implementation of (5.19) to compute  $\mathbf{A}$  is by no means straightforward. The primary difficulty lies in the evaluation of the matrix logarithm, which reference [48] refers to as “an important area demanding much more work.” The matrix logarithm is defined the same way that any other function of a matrix is defined. Here we present a not-entirely-rigorous definition which is analogous to that given for the matrix exponential in Section 4.3, namely

$$\log (\mathbf{Q}) = \mathbf{X} [\log (\Lambda)] \mathbf{X}^{-1} \quad (5.21)$$

where  $\mathbf{Q} = \mathbf{X}\Lambda\mathbf{X}^{-1}$  is an eigendecomposition of  $\mathbf{Q}$ , and

$$\log (\Lambda) = \begin{bmatrix} \log (\lambda_1) & & & \\ & \log (\lambda_2) & & \\ & & \cdots & \\ & & & \log (\lambda_n) \end{bmatrix} \quad (5.22)$$

where  $\lambda_1, \lambda_2, \dots, \lambda_n$  are the diagonal elements of the eigenvalue matrix  $\Lambda$ . As was mentioned in Section 4.3, such a definition will obviously break down in the case in which  $\mathbf{Q}$  is defective. However, there is no completely rigorous recourse for computation of the logarithm of a defective matrix as there is for computation of the exponential of a defective matrix. This is because any power series representation of the logarithm converges only in a limited region of the complex plane. While this is not a significant issue in scalar computations, matrix computations require that all eigenvalues of the matrix in question lie within the region of convergence of the power series in question. Clearly this presents a problem for many matrices.

The first step in well-conditioned evaluation of a function of a matrix is the use of Parlett's algorithm [48], which makes it unnecessary to explicitly compute the potentially ill-conditioned eigenvector matrix  $\mathbf{X}$ . However, it offers only limited protection from the problem of repeat eigenvalues, in the form of a block version of the algorithm. The block version only partially addresses the problem since it still requires evaluation of the logarithm of each diagonal block, which was likely preserved as a block because it was close to a defective matrix. This implies that the logarithm of such a block must still be computed by power series, and therefore requires identification of a series which converges over a region containing all the eigenvalues in the block. Some suggestions for possible candidate power series are given in [48] and [53]. The problem is further complicated by the question of the *rate of convergence* of a series—if an eigenvalue lies too close to the boundary of the region of convergence, the series will converge very slowly.

In this work, direct application of the non-block version of Parlett's algorithm was employed to compute the matrix logarithm. All of the work done in the process of initial reduction serves to help avoid repeat or close eigenvalues for a typical set of measurements. For that matter, the eigenvector approach of equations (5.21) and (5.22) was found to be sufficient for calculating the logarithm without incurring excessive error. However, it should be emphasized that even though problems of the sort mentioned above are improbable in this particular application, they are by no means impossible to encounter. Robust software for computing the matrix logarithm should at the least include reliable error estimation, and should also have contingencies to help cover some of the possible pitfalls mentioned above.

Finally, it should be noted that even the scalar logarithm function is not a one-to-one mapping. This is because the exponential function maps many inputs to a single output. That is, given

$$e^{j\theta} = a + jb \tag{5.23}$$

it follows that

$$e^{j(\theta + 2\pi n)} = a + jb, \quad n = 0, \pm 1, \pm 2, \dots \quad (5.24)$$

If we invert (5.24), we see that

$$\log(a + jb) = j(\theta + 2\pi n), \quad n = 0, \pm 1, \pm 2, \dots \quad (5.25)$$

In this work, only a single solution to (5.25) was used to identify the eigenvalues of the continuous-time system, namely

$$\log(a + jb) = j\theta, \quad -\pi \leq \theta < \pi \quad (5.26)$$

This particular solution results in a continuous-time system with an impulse response whose complex oscillations will always have a period which is greater than the sampling interval. This can be easily seen if the continuous-time impulse response is viewed as a sum of weighted exponentials analogous to how the impulse response of the sampled system was described in Section 5.3. This interpretation of the matrix logarithm is the correct one for this modeling application, since the data was assumed (as stated in Section 4.4) to have been sampled according to the Nyquist criteria. Adherence to the Nyquist criteria when sampling a signal enforces by definition the absence of oscillations with period less than  $T$ , the sampling interval.

Having completed the transformation from the sampled-time domain, we now have a continuous-time system which approximates the original measured system in terms of the matrices  $\mathbf{G}$  and  $\mathbf{C}$  from equation (4.28). Further post-processing is necessary, however, to achieve a model with some of the properties alluded to in Section 2.1, namely industry compatibility and an accuracy/complexity tradeoff. The following section outlines the necessary procedures.

## 5.7 *Post-processing*

In Section 2.1, several criteria were discussed as being important aspects in evaluation of various interconnect modeling solutions. Until this point, none of these issues has been addressed directly (although in fairness the whole of Chapter 5 up to this point has been concerned with the issue of accuracy). In this section we look at additional steps which are implemented as post-processing to transform our newly extracted model into one which is more practically useful. In particular, we look at how to reliably trade off model accuracy against model complexity, and also at how to transform the model to be useful in conjunction with existing industry standard simulation tools.

### 5.7.1 *Further Reduction (Complexity vs. Accuracy)*

As was stated in Section 2.1.2, a model should not characterize the behavior of a device at frequencies far above the frequency range in which it will be used, since such overmodeling typically causes more complex models than necessary and unnecessarily increases simulation time. One way to avoid overmodeling is to apply a network reduction step such as one of those outlined in [50]. In general such methods apply a rectangular transformation, as was outlined in Section 4.8, to the matrices  $\mathbf{G}$  and  $\mathbf{C}$  of the newly extracted model. The subspaces spanned by the variable portions of the transforming matrices  $\mathbf{U}$  and  $\mathbf{V}$  of equation (4.38) are chosen so that the desired characteristics of the transfer function (typically the low-frequency characteristics in interconnect modeling) are preserved to within a prescribed level of accuracy. Because  $\mathbf{U}$  and  $\mathbf{V}$  are typically rectangular in this step, the matrices  $\mathbf{G}$  and  $\mathbf{C}$  are reduced in size, resulting in a simpler model. For in-depth information on network reduction, the interested reader is referred to [50], or at least to [10], [11] or [12].

### 5.7.2 *Conversion to Admittance (Compatibility)*

Section 2.1 also emphasized the importance of a model being compatible with standard industry simulation tools. Without question the electronic design industry rallies around SPICE, or more likely one of its many derivatives which are commercially avail-

able. Even some of the most recent SPICE derivatives [54], which have revamped the internal solver to achieve dramatic speed increases for on-chip digital circuit simulation, still rely on the SPICE netlist format for input. In the RF analog design world the picture is not so clear, in that a mixture of SPICE-based and scattering-based simulators is in use. Although scattering-based tools primarily take geometric descriptions of electrical components, most are also capable of handling lumped elements. This allows a lumped-element model to be interfaced to the simulator by simply filtering it through a post-processing module to alter its netlist syntax.

The emphasis in this work is on extraction of a model which is compatible with SPICE-based simulators. Internally, the primary point of commonality between such simulators is their use of Modified Nodal Analysis (MNA) [5][6] to formulate their equations of state. Because MNA is based primarily on admittance and secondarily on impedance relationships, it employs nodal voltages and possibly branch currents as its state variables. Furthermore, the branch equations implicit in MNA form immittance relationships between total current and total voltage. This exists in stark contrast to the model extracted by the means outlined in preceding sections of this chapter, whose port state variables are reflected voltage waves, belying a scattering-based model. It is apparent that a transformation is necessary if the model is to be utilized in an admittance-based SPICE-compatible simulator.

As was stated in Section 4.7, the model in equation (4.28) is formulated in terms of scattering parameters, which is a logical starting point since it will be built from sampled scattering impulse response data. As stated in Section 3.1, scattering data is more easily measured at high frequency, and often provides a more concise representation of the system in question. However, this necessitates a mapping step to move from a scattering state space to an admittance state space. The approach taken in this work was to begin by adding the  $\mathbf{G}$  and  $\mathbf{C}$  matrices directly into the global MNA (immittance) matrix in parallel, which is to basically assume that  $\mathbf{G}$  is a conductance matrix and  $\mathbf{C}$  is a susceptance matrix. (Exactly how the parallel addition is implemented is the subject of the next section.) However, we know that  $\mathbf{G}$  and  $\mathbf{C}$  are not conductance and susceptance, due to the reasons

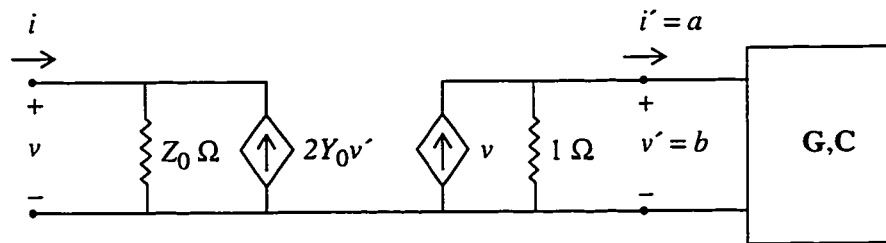


discussed in the preceding paragraphs. To remedy this discrepancy, a transformer is inserted between each of the ports of the model and its connection point to the global MNA matrix. The purpose of the transformer is to alter the network such that the current and voltage waveforms computed by the admittance-based simulator at what *used to be* the ports become the scattering waveforms if measured from the ports of the resulting system.

The above description of the transformation will no doubt become more clear if stated mathematically. To derive the topology and element values of such a transformer, we appeal to the circuit-theory equations that relate the scattering and immittance variables,

$$\begin{aligned} v &= a + b \\ i &= Y_0(a - b) \end{aligned} \quad (5.27)$$

The variables  $v$  and  $i$  are voltage and current vectors, respectively, and the vectors  $a$  and  $b$  consist of incident and reflected voltage waves, respectively. As seen in Figure 5.1, the



**Figure 5.1** Transformer for simulation of a scattering-based model in an admittance-based simulator.

inputs and outputs of the newly-extracted model represented by  $\mathbf{G}$  and  $\mathbf{C}$  are known to be incident and reflected voltage waves,  $a$  and  $b$ . However, since the  $\mathbf{G}$  and  $\mathbf{C}$  matrices were added in parallel into the global MNA matrix as conductance and susceptance, the admittance-based simulator views these quantities as voltage and current. For this reason, we

refer to them as  $i'$  and  $v'$ , and they are substituted into (5.27) for  $a$  and  $b$ , respectively. After a minor rearrangement, the resulting relationship is seen to be

$$\begin{aligned} i &= Y_0(v - 2v') \\ i' &= v - v' \end{aligned} \tag{5.28}$$

Restatement of the above relationship in terms of an admittance matrix results in

$$\begin{bmatrix} i \\ i' \end{bmatrix} = \begin{bmatrix} Y_0 & -2 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} v \\ v' \end{bmatrix} \tag{5.29}$$

Inspection of the transformer of Figure 5.1 verifies that it indeed implements the relationship expressed in (5.29).

### 5.7.3 Branch Reduction

Before the simulation netlist is actually written, a full-rank state-variable transformation such as that outlined in Section 4.8 is applied to the system. The transformation matrices  $\mathbf{U}$  and  $\mathbf{V}$  are chosen based on the eigenvectors of the internal pencil  $(\mathbf{G}_i + s\mathbf{C}_i)$  of equation (4.37) such that in the transformed system, the internal pencil is block diagonal. Because the transformation is of full rank, the port behavior of the model is unchanged. The purpose of the transformation is to reduce the number of off-diagonal elements in each matrix, which directly reduces the number of branches in the circuit from  $O(M^2)$  to  $O(M)$ . The net effect is a significant reduction in simulation time for the circuit.

Although not stated explicitly in the mathematical derivation of Section 4.8, it is necessary in the presence of machine roundoff to use well-conditioned matrices  $\mathbf{U}$  and  $\mathbf{V}$  in the basis transformation, or the port behavior will not be accurately preserved. Issues of conditioning may preclude full diagonalization of the internal pencil into  $2 \times 2$  blocks. However, even if the blocks are larger than  $2 \times 2$ , the number of branches is still reduced from  $O(M^2)$  to  $O(M)$ .

#### 5.7.4 Conversion to a SPICE netlist

As was discussed in Section 2.1.3, almost every general-purpose simulator in the electronics industry is SPICE-based, or at least reads its input from a SPICE-type netlist. It was also mentioned that SPICE-based simulators employ some form of Modified Nodal Analysis (MNA). One of the appealing aspects of MNA is the ease with which typical lumped linear elements may be entered into the state matrix formulation of equation (4.28) by a process often referred to as “stamping” or “stenciling”. Since the input format for a SPICE-based simulator is a netlist of standard elements, the stamping procedure acts as a filter to translate the input netlist into a matrix formulation suitable for computer solution. Modeling, however, is in some sense the inverse of the simulation problem, and one of our explicit modeling goals is to have our resultant model be in a SPICE-compatible netlist format. For this reason our last step in building a model is to apply the inverse of the stamping procedure to the extracted  $\mathbf{G}$  and  $\mathbf{C}$  matrices, to convert them into a SPICE netlist. We refer to this inverse stamping process as “unstamping”.

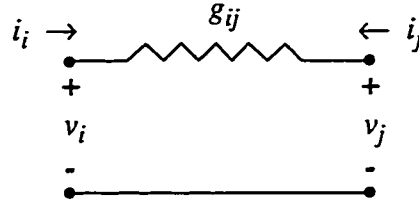
#### 5.7.5 MNA Stamping Procedure

The process of unstamping is easily understood if the process of stamping is first understood, so we begin with the latter. We commence by customizing equation (4.28) to be specific to the RC problem, *i.e.* that of a circuit containing only resistors and capacitors. The formulation for the RC problem is given by

$$\mathbf{C}\dot{\mathbf{v}}(t) + \mathbf{G}\mathbf{v}(t) = \mathbf{i}(t) \quad (5.30)$$

where  $\mathbf{v}(t)$  is the set of nodal voltages in the circuit, and  $\mathbf{i}(t)$  represents the current injected externally into each node in the circuit.

To understand the relationship between (5.30) and a physical circuit, we observe the conductor shown in Figure 5.2. The conductor connects nodes  $i$  and  $j$  of the circuit via a



**Figure 5.2** Conductor  $g_{ij}$  to be inserted into global circuit matrix  $\mathbf{G}$ .

conductance of value  $g_{ij}$ , and is governed by the twoport equations

$$i_1 = -i_2 = g_{ij}(v_i - v_j) \quad (5.31)$$

where  $v_i$  and  $v_j$  are the nodal voltages of interest, and the currents injected into each of the ports are represented by  $i_i$  and  $i_j$ . The branch immittance relationship expressed above may be cast into matrix form as

$$\begin{bmatrix} g_{ij} & -g_{ij} \\ -g_{ij} & g_{ij} \end{bmatrix} \begin{bmatrix} v_i \\ v_j \end{bmatrix} = \begin{bmatrix} i_i \\ i_j \end{bmatrix} \quad (5.32)$$

It can be shown that because the currents sum at the nodes of the circuit, in order to add a conductance between nodes  $i$  and  $j$  of an existing circuit, it is only necessary to add the  $g$  submatrix of (5.32) into the global  $\mathbf{G}$  matrix of the existing circuit at the  $i$  and  $j$  positions. In this fashion all the conductors (or actually resistors) in a netlist are entered into the state matrix. The procedure is completely analogous for capacitors if the variable  $\dot{v}$  is substituted for  $v$ .

The formulation presented above, however, overlooks the need for a datum voltage from which to reference the nodal voltages. For example, the twoport in Figure 5.2 is

characterized by the two voltage drops,  $v_i$  and  $v_j$ , and more specifically it is characterized by the difference between those voltages. For that reason, knowledge of the currents is insufficient to determine the actual voltages, unless the voltage at one of the nodes is fixed. For larger, more complicated networks, the same principle applies in that the voltage at one node in a circuit must be fixed as a datum voltage in order to determine the voltages at all the other nodes. If not, there exist an infinite number of solutions to the nodal voltages, and the system admittance matrix can be shown to be singular. The classical solution to this problem is to identify one of the nodes as “ground”, governed by

$$v_g = 0 \quad (5.33)$$

which allows row  $g$  and column  $g$  to be removed from the global matrix. The net effect may be thought of as introducing a new one-port element, a “conductor to ground”. Such a conductor of value  $g_{ii}$ , when attached to node  $i$ , is governed by the equation

$$g_{ii}v_i = i_i \quad (5.34)$$

Finally, it must be noted that the above formulation covers the case of resistors and capacitors, whose admittances are well defined at all finite frequencies. That is, the current through such an element can be determined from the voltage drop across it for all such frequencies. Inductors and voltage sources are examples of elements for which this is not the case, and for this reason they have special impedance-based stamping rules. However, this fact is inconsequential to the results of this work, and the interested reader is referred to [5] and [6] for additional insight on MNA-style stamping of the various lumped elements.

#### 5.7.6 Unstamping Procedure

The observant reader will immediately notice that the mapping from netlist to state space via Modified Nodal Analysis is many-to-one, which implies that the inverse map from state space to netlist is non-unique. Indeed this is the case, and we are therefore

required to choose an algorithm which results in one of the possible solutions. The unstamping algorithm chosen for the purposes of this work is quite simple, consisting of the following:

1. Assume each off-diagonal element  $g_{ij}$ ,  $i \neq j$ , of the global  $\mathbf{G}$  matrix represents a conductor of that value between nodes  $i$  and  $j$  of the network. Subtract the associated admittance matrix, analogous to (5.32), from the global  $\mathbf{G}$  matrix. If any asymmetry is present between the nodes, treat it as a voltage-controlled current source (VCCS) to ground.
2. Application of the above to every nonzero entry in  $\mathbf{G}$  results in a diagonal matrix, each element of which may be realized as a conductor to ground.
3. Repeat steps 1 and 2 to realize the global  $\mathbf{C}$  matrix.

Note that the symmetry of  $\mathbf{C}$  may be enforced during extraction, except possibly at the ports. This largely negates the need to use controlled sources in the realization of  $\mathbf{C}$ .

Of significance is the fact that no inductors are used in the realization, which would require additional unstamping machinery. This is accomplished by allowing the value of any conductor or capacitor to be either positive or negative, which has occasionally bred questions from those who would believe that use of negative elements implies asymptotic instability, or at the very least an active circuit. This, however, is simply not the case. As was stated previously, the mapping from netlist to state space is not one-to-one, and it is the characteristics of the state space which determine the stability properties of a system—not the value of any individual circuit element in and of itself. This is particularly true for systems defined only in terms of their port behaviors, in which case many nodes of the circuit are internal to the multiport and can never be directly excited by a source.

## 5.8 Conclusion

In this chapter a derivation was presented for a robust interconnect modeling algorithm. The derivation was for the most part constructive, in that it leads directly to an

implementation of the modeling algorithm. Having initially discussed the meaning and importance of robustness in CAD applications, our starting point for the derivation was the recursion relationship which defines the sampled-time system with which the measured data is to be modeled. Problem formulation was covered, as well as the numerical considerations necessary to improve and subsequently maintain the conditioning of the problem. The necessary post-processing steps for achieving an accuracy/complexity tradeoff and for ensuring industry compatibility for the model in its final form were also discussed. The final result is a netlist composed of standard SPICE elements which accurately characterizes the system represented by the initial data.

The implementation of any numerical algorithm is certain to raise a number of issues. The following chapter discusses those issues encountered in the process of implementing the modeling algorithm just derived.

## Chapter 6

### *IMPLEMENTATION OF THE INTERCONNECT MODELING SYSTEM*

In previous chapters we have covered the motivational issues behind interconnect modeling, reviewed the mathematical details pertinent to the derivation of the extraction algorithm, and derived the algorithm itself. However, a mathematical proof is of very limited use from a practical CAD perspective until it has been implemented on a computer. The primary reason, academically speaking, is that even the best mathematician is subject to making errors, and a good way to trap such errors is to set aside the symbolic variables used in proofs and derivations, and test the mathematical platitudes on actual numbers. This is especially important in applied math and numerical computations, where large-scale problems must be solved in the presence of roundoff error. Otherwise, an algorithm which appears to be sound on paper might turn out to be too poorly conditioned or computationally expensive to be of practical use. If so, it must be re-implemented or scrapped. Finally, from a more practical perspective, the algorithm will not generate a single model, and may hardly be called a model extraction algorithm, until it has been implemented.

There are typically multiple approaches to implementing a general algorithm on a computer, and this is indeed the case with the algorithm presented in this dissertation. Which implementation is chosen often depends on the goals of the programmer in imple-



menting the algorithm in light of the resources (hardware, software, manpower) which are available for implementation. The remainder of this chapter is devoted to presentation of specific implementation of the algorithm for the purposes of this research work, as well as explanation when appropriate as to why a particular implementation path was chosen over another.

### *6.1 Choice of Computer Language*

Prior to implementation of any computer application, a high-level decision must be made as to what computer language or framework to use to write the necessary computer code. Important issues involved in such a decision include simplicity, reliability, portability, and longevity. Simplicity refers to how easy it is to code and maintain a particular application in a particular computer language or framework, while reliability is related to whether applications coded under a language or framework behave as expected in a bug-free manner. Portability refers to whether the application as implemented in one computer architecture (or graphical user interface platform) can be easily ported to another architecture or platform with minimal rewriting of code, and is again highly dependent on the language or framework used to write the application. Finally, longevity is the question of how long the computer language or framework will be in existence, so that the CAD implementation of the algorithm can be updated easily and incrementally to reflect current user trends and expectations in terms of both user interface and technical functionality.

In an academic setting, simplicity is usually the most highly weighted of the above issues, particularly when the primary aim of the research is the proof of a concept. If this is the case, any excess effort which must be put into writing computer code can detract from the research focus. A more insidious effect of an unnecessarily complex implementation language or platform is that the possibility of wasting time and effort to code up a potential problem solution may deter the researcher from trying what could turn out to be a superior approach. However, although simplicity is a very desirable characteristic of a computing language or framework, the simplest solution is often the least flexible or the

slowest or both. For this reason some simplicity must be traded off for flexibility based on the needs of the specific application being coded.

In this work, MATLAB [55] was employed for almost all aspects of algorithm research and development. In the experience of the author, MATLAB very effectively balances simplicity of code-writing against flexibility in terms of its capabilities. This is because the syntax itself is fairly intuitive, allowing straightforward access to a large number of pre-implemented high-level functions, as well as many low-level functions. At the same time, its flexibility allows the user to code up those high- and low- level functions which are not pre-implemented. A simple, intuitive user interface increases the utility of the tool even further. MATLAB is particularly useful for coding and testing implementations which make heavy use of standard numerical algorithms such as singular value decomposition, QR factorization, matrix inversion, and eigendecomposition [48] because of the easy access it grants to these routines. For this reason it was extremely well suited to researching the algorithm presented in Chapter 5.

Having developed and tested the approach and proven it in concept via MATLAB implementation, it was decided to re-implement many portions of the algorithm in C [56] and Fortran [57]. This was done for a variety of reasons, the first being to achieve speedup and decrease memory usage in certain critical areas. Since these are important aspects in the implementation of an industry CAD tool, it was decided to address them to a reasonable, although limited, extent. Another reason for partial re-implementation was to reduce dependence on the MATLAB source code and libraries. Although the libraries' interfaces are very stable, it cannot be ignored that the internal workings are subject to change which could introduce problems for an application. Furthermore, MATLAB is not necessarily a part of the circuit designer's environment, which more typically consists of tools which are highly optimized for rapid circuit analysis and design. Because of the short life and limited industry use of typical academic software, neither reason for reducing dependence on third-party products is particularly pressing in the university setting. Finally, porting to a low-level language such as C is of limited usefulness if it is desired to do further algo-

rithmic and implementational research. For these reasons, a complete port from MATLAB to C was not implemented.

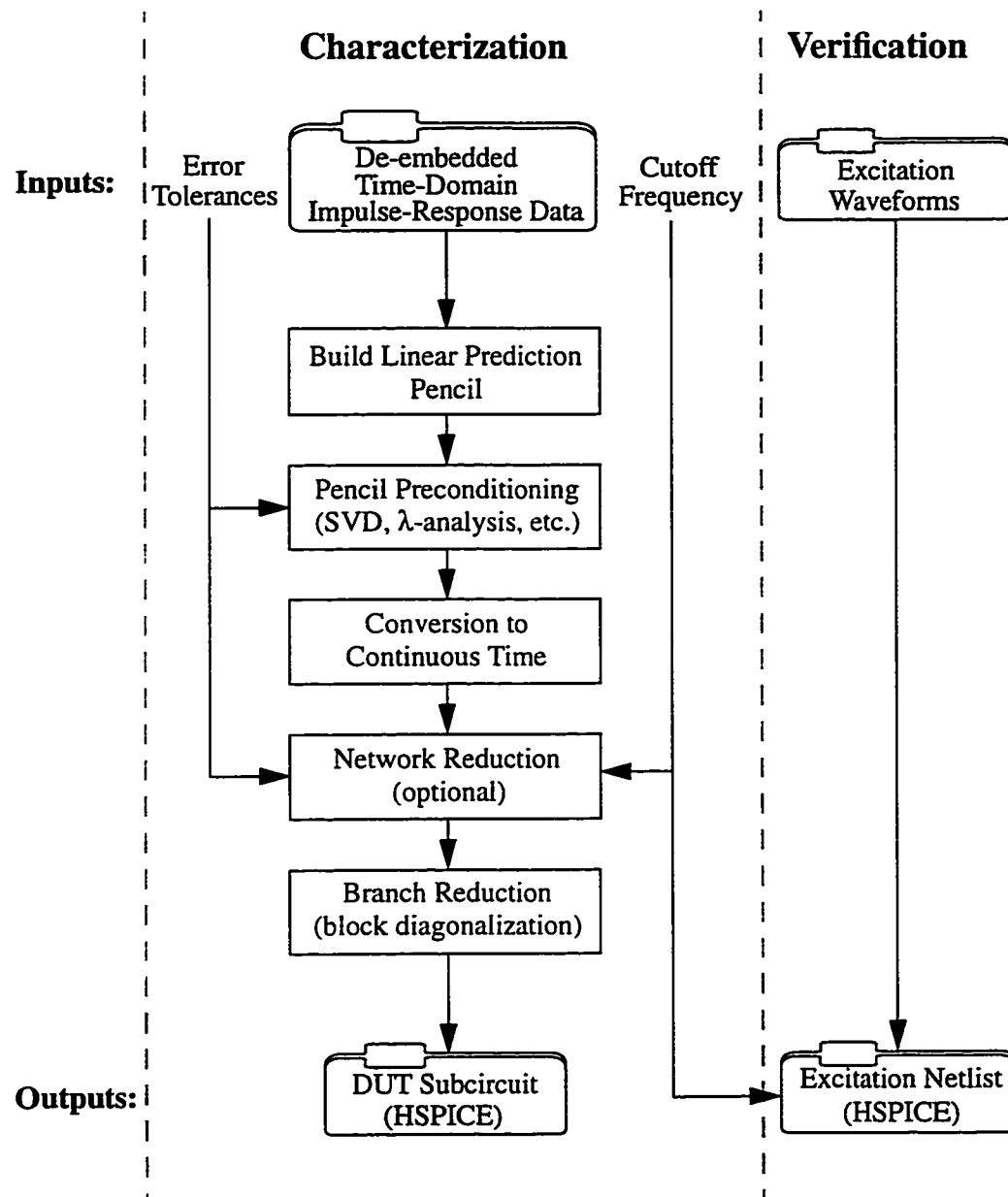
## 6.2 Organization of the Interconnect Modeling Framework

A flow diagram of the implementation of the interconnect model extraction algorithm presented in Chapter 5 is shown in Figure 6.1. The details regarding each of the various segments of the diagram are outlined in the subsequent subsections.

### 6.2.1 Inputs

The inputs to the system in Figure 6.1 are error tolerances, de-embedded data, cutoff frequency, and excitation waveforms. A significant amount of effort is required on the part of the modeler to obtain these inputs, particularly the de-embedded data. Not only must this data be measured, as outlined in Appendix A, it must also be normalized and de-embedded, as discussed in Appendices B and C. This data consists of the approximated samples of the time-domain impulse response, referred to as  $S_p[n]$  in equation (5.2). As such it must be an accurate representation of the impulse response, since the extracted model will be no more accurate than the data itself. In this particular implementation of the modeling algorithm, previously measured and de-embedded data was read from an ASCII file to begin the extraction.

The error tolerances consist of two values. The first is used in the preconditioning of the model to determine what components of the input waveforms fall into the category of *noise*, and what components fall into the category of *signal*. In choosing an appropriate error tolerance, it is important to evaluate the noise floor of the input waveforms. If the error tolerance is below the noise floor, then the algorithm will model the noise present in the input waveforms, which can lead to instability and inaccuracy. If the error tolerance is too high, the model will be inaccurate and possibly unstable. A good error tolerance choice is one which is just large enough to reject the noise present in the waveforms, yet small enough to include the pertinent signal components. An important aid in making this judgment call is a high signal-to-noise ratio in the measurements. If a high SNR is not



**Figure 6.1** Flow diagram of the interconnect modeling system as implemented.

available from the raw measurements, it may be necessary to sacrifice some of the bandwidth of the measurements by filtering out noise components to boost the SNR.

The second component of the error tolerances is used to perform the optional model reduction step. That is, the response of the reduced model in the frequency range of inter-

est is not allowed to deviate from the response of the original (unreduced) model more than the amount specified by the tolerance.

The “frequency range of interest” mentioned above is defined at its upper end by the cutoff frequency, which is the highest frequency at which the user desires the model to fit the data. The lower bound of this range for the sake of this implementation is taken to be DC. In this way, the cutoff frequency is used in conjunction with the error tolerance to control the allowable amount of error as a function of frequency. Choice of a cutoff frequency depends primarily on the bandwidth of the application in which the model is to be used; that is, the frequency should be high enough to accommodate all important DUT characteristics to be modeled.

The excitation waveforms are only provided if the user would like the framework to build an excitation netlist to drive the model. Although it is optional, this netlist is quite useful in the verification stage, and provision of the excitation waveforms does not generally require any extra work on the part of the modeler, since they are typically a byproduct of the de-embedding process. However, if the physical excitation to the DUT was quite step-like, then almost any step-like excitation with the same risetime will give similar results when the model is evaluated, especially if some care is taken to smooth the corners of the simulation input waveforms. This effectively removes from the excitation waveform any signal energy which is above the cutoff frequency.

### 6.2.2 *Pencil Preconditioning*

The next two algorithmic steps from Figure 6.1 are to build and precondition the linear prediction pencil, as outlined in Sections 5.3 and 5.4, respectively. In this work, the input waveforms were read from an ASCII file and immediately written to memory in a dense matrix structure. An implementation which was more heavily geared toward efficient memory usage could further exploit the structure of the  $S_0'$  and  $S_1'$  matrices to reduce the maximum amount of memory required.

Pencil preconditioning, or subset selection, was performed as outlined in Section 5.4 using the singular value decomposition. The symmetric eigenvalue library routine from

LINPACK [58] was used for this step. The error tolerance  $\varepsilon$  was used as a cutoff value such that  $\sigma_R(\mathbf{S}'_0)$  from equation (5.12) is the smallest singular value of  $\mathbf{S}'_0$  which is greater than the error tolerance. That is,

$$\sigma_R(\mathbf{S}'_0) > \varepsilon > \sigma_{R+1}(\mathbf{S}'_0) \quad (6.1)$$

Since the error tolerance serves to separate the noise components of the input waveforms from the signal components, the error tolerance was chosen based on the noise floor observed in the data, as discussed in Sections 5.4 and 6.2.1. Values were typically used which ranged from 1 mV to 10 mV, depending on the noisiness of the data used in the extraction.

Finally, it should be noted that the symmetric eigenvalue routine in LINPACK is a dense, complete decomposition. However, an implementation geared toward reduced memory usage or increased speed in many cases could vary the algorithm employed for subset selection to improve performance in terms of speed or memory usage or both.

### 6.2.3 *Conversion to Continuous Time*

Conversion of the sampled-time model to continuous time is performed through implementation of equations (5.19) and (5.20), and therefore requires evaluation of the matrix logarithm. As was stated earlier, computation of the matrix logarithm is a difficult task. The most robust approach is likely performed via a block version of Parlett's algorithm [48] in which problematic blocks are evaluated by means of appropriate power series. Because a fast version of Parlett's algorithm (particularly a block version) was unavailable in MATLAB, the logarithm was computed by the eigenvector method. Section 4.3 alluded to the fact that this method sacrifices accuracy if the eigenvectors are poorly conditioned. For this reason, a check on the condition number of the eigenvectors was included as part of the evaluation of the logarithm. In the various benchmark model extractions the eigenvector approach was never responsible for the loss of more than 6

digits of precision. However, such an approach may not be robust enough for industry use.

#### 6.2.4 *Network Reduction*

As was stated in Section 2.1.2, a model should not characterize the behavior of a device at frequencies far above the frequency range in which it will be used. Such over-modeling typically causes overly complex models which unnecessarily increase simulation times. To avoid this problem, a network reduction step is included in the modeling system flow. The reduction algorithm, based on material presented in [50], is implemented in MATLAB, and its inputs are an error tolerance and a cutoff frequency.<sup>†</sup> The cutoff frequency defines the region in which accurate characterization is desired and the region in which accurate characterization is unnecessary, while the error tolerance limits the amount of allowable error in the region of accuracy. As was mentioned previously, the cutoff frequency should be chosen based on the bandwidth over which the model will be evaluated.

#### 6.2.5 *Branch Reduction and Conversion to Admittance*

A transformation of the state variables corresponding to the internal nodes of the model was performed as explained in Section 5.7.3 to diagonalize the internal submatrices of the  $\mathbf{G}$  and  $\mathbf{C}$  matrices. If the  $M \times M$  matrices  $\mathbf{G}$  and  $\mathbf{C}$  are left dense, the number of nonzero entries (and therefore the number of branches in the circuit) scales as  $M^2$ . However, the above diagonalization causes number of nonzero entries in  $\mathbf{G}$  and  $\mathbf{C}$  to scale as  $M$  (assuming that  $M \gg m$ , where  $m$  is the number of model ports). This effectively reduces the number of internal branches in the model, which in turn reduces the simulation time significantly for large models.

Since the transformation is based on the eigenvectors of the system, it suffers from the same potential conditioning problems as the computation of the matrix logarithm. If a poorly-conditioned eigenvector matrix were encountered, it would become necessary to

<sup>†</sup> The author would like to express appreciation to the author of [50], Kevin Kerns, for allowing the use of his network reduction code.

group the associated eigenvalues together in a larger block. However, the scaling of the number of branches with the number of nodes in the circuit remains linear, even if some larger blocks are necessary. In the various examples run in the verification of this implementation of the modeling algorithm, loss of conditioning during this step was not found to be a significant issue, since the preconditioning and network reduction steps both tend to work against repeat eigenvalues.

A potentially convenient side effect of this step is that because it involves an eigen-decomposition of the system, any eigenvalues corresponding to exponential growth (that is, poles with positive real parts) may be removed in order to ensure the weak form of asymptotic stability discussed in Section 2.1.1. (As a reminder, this would imply no exponential growth when the system is unterminated.) It is worth noting, however, that no positive poles were extracted when running the characterization examples of Chapter 7.

#### 6.2.6 *Outputs*

The output of the system consists of either one or two simulation netlists in HSPICE [59] format, which is an extended version of standard SPICE format.<sup>†</sup> HSPICE was chosen as a simulation engine since it is widely accepted and used in the circuit design industry. Also of significance in this decision was that a donated copy of the software was available for use, for which the author would like to express appreciation to Meta-Software.<sup>‡</sup>

The model itself is contained in its own netlist (in a separate file) which is included in the main simulation file, and is accessed by the simulator as a subcircuit. Writing the model in subcircuit format has two useful side effects. The first is that multiple instances of the model may be included in a single simulation with no node-naming conflicts, since the circuit simulator does the bookkeeping to resolve such conflicts. The second reason is to provide some measure of protection for the model itself, so that it is not inadvertently

<sup>†</sup> In the case of simple resistors and capacitors the two syntaxes are identical.

<sup>‡</sup> The HSPICE simulation tool has since been acquired by the Avant! Corporation, Fremont, CA.



changed by connecting something to one of its internal nodes. Embedding it within a sub-circuit statement in a separate file largely achieves this goal.

The excitation netlist can be either a version of the input waveforms filtered to reflect the cutoff frequency of the model, or a fabricated set of waveforms whose frequency content lies within the range determined by the cutoff frequency. The excitation netlist explicitly includes the extracted DUT netlist and connects its excitation sources to the ports of the DUT for evaluation by the simulator.

### 6.3 *Computational Complexity*

Because the purpose of computer-aided design is to save time over manual design, a typical figure of merit in many CAD applications is execution time in CPU seconds. This is especially true in those areas (*e.g.*, simulation, physical design) which have matured over time to the point that a set of standard benchmark circuits exists which facilitates the comparison of different algorithms and variations of the same algorithm in terms of their CPU usages.

However, in an emerging field such as measurement-based interconnect modeling, the purpose of any one innovation is not to tweak existing algorithms, nor to implement new ones to shave seconds off those algorithms. The aim of this work is instead to offer an approach to solving a problem that was previously unsolvable within an acceptable accuracy level. For this reason, the emphasis of this research is not to develop the most efficient implementation of the extraction algorithm. Instead, research effort was primarily directed toward development of a robust algorithm, and implementation effort toward a relatively simple, illustrative package to prove the concept of the algorithm. In this context, it is more useful to look into issues of computational complexity to show how CPU usage scales with the size of the problem, rather than tabulate exact CPU seconds elapsed for every example.

The limiting algorithmic step in terms of problem size is the preconditioning of the initial pencil, since it requires some form of subset selection from a dense matrix. As a rule of thumb, in the best algorithms for operation on arbitrary dense  $n \times n$  matrices, the

number of required floating-point multiplies scales as  $n^3$ , which is denoted by stating that the algorithm is  $O(n^3)$ , or “order  $n$  cubed”. This is true for all classical algorithms such as matrix-matrix multiplication, matrix inversion, iterative factorizations such as eigendecomposition and SVD, and non-iterative factorizations such as LU, QR, or Cholesky factorization. Since the matrix to be operated upon ( $S'_0$  from equation (5.9)) is of size  $\frac{mN}{2} \times \frac{mN}{2}$ , where  $m$  is the number of ports and  $N$  is the number of data points in each waveform, the preconditioning step requires  $O((mN)^3)$  operations. This seems true for any robust approach to preconditioning the initial pencil. That is, although one technique may require half or one third the amount of computations, each scales the same way with the size of the system.<sup>†</sup> On a Sun Sparc 20 workstation, the amount of time consumed in preconditioning was on the order of ten seconds for  $mN=512$ , 100 seconds for  $mN=1024$ , and 500 seconds for  $mN=2048$ . From this progression it is clear that there is a practical limit to what systems may be modeled up to what frequency.

The second most expensive algorithmic step computationally is network reduction. This step is also  $O(n^3)$ , where  $n$  is the dimension of the  $G$  and  $C$  matrices. However, since the reduction is performed after the preconditioning step, the dimension of  $G$  and  $C$  is often ten times less than that of the original system of equations (5.7)-(5.9) before preconditioning. For this reason, the cost of network reduction is dwarfed by the cost of the preconditioning of the large initial pencil for any typical application. This is true despite the fact that the network reduction approach used in this work [50] utilizes a potentially expensive heuristic to determine whether the reduced system meets the error criteria or must be increased in complexity.

It was mentioned in Section 6.2.2 regarding memory usage that the implementation outlined in this chapter stores the initial large matrix  $S'_0$  in a dense array of double-precision floating point numbers. This implies that memory usage scales as  $O((mN)^2)$ . In the same section it was suggested that memory usage could be reduced by taking advan-

<sup>†</sup> In fact, it might easily be stated that the entire problem of accurately extracting a state space from a general set of time-domain measurements (*i.e. system identification*) is  $O(n^3)$ , where  $n$  is the initial size of the pencil, since even without the presence of the preconditioning step, the necessary matrix inversion is of the same order.

tage of the structure of the problem. However, this space reduction was not implemented since it was not considered to shed additional light on the basic model extraction algorithm.

#### 6.4 *Summary*

In this chapter some of the details involved in the implementation of the algorithm presented in Chapter 5 were discussed, and reasons were given as appropriate for the choice of this particular implementation over the many others which are available. Approximate values were given as to the computational cost in CPU seconds to perform typical problems of various sizes. More importantly, computational complexities of the limiting aspects of the algorithm were explained in terms of how the computational cost scales with the size of the problem. In order to complete the picture of the modeling algorithm, it only remains to verify empirically that it works as presented and implemented. For this final step in the presentation, we proceed to the next chapter.

# Chapter 7

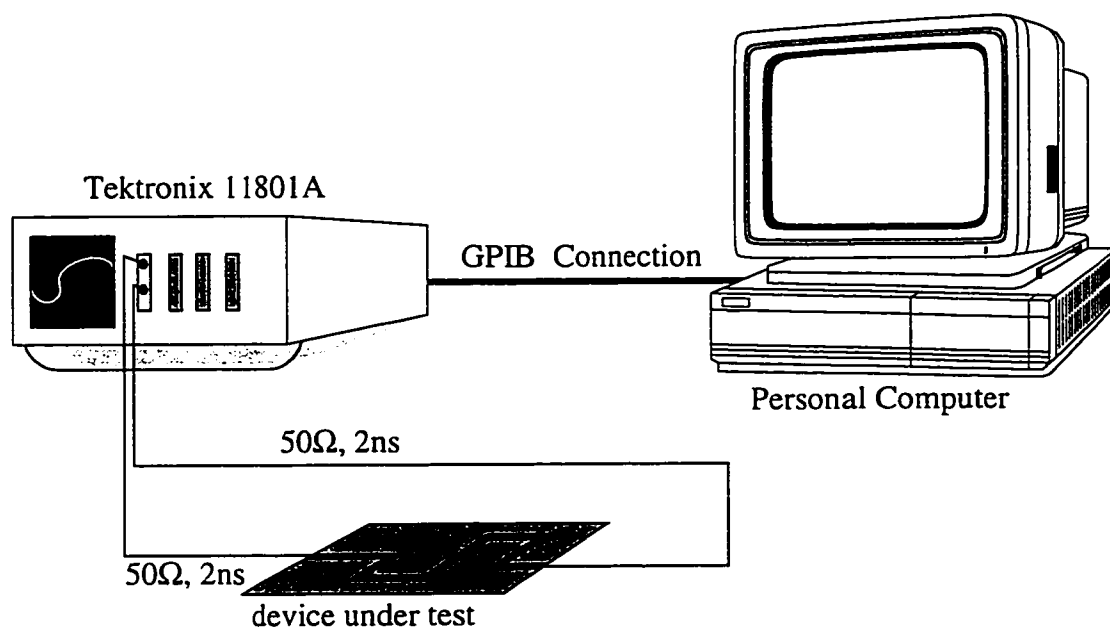
## *CHARACTERIZATION EXAMPLES*

The final step in the proof of any CAD application is its effectiveness when applied to real-life examples. Often the most rigorous proving ground is the application's intended set of users, who will invariably ferret out any algorithmic or implementation weaknesses, and will furthermore provide creative and practical suggestions for improvement. Because a full-fledged industry-ready tool was not built around the extraction algorithm developed in the course of this research work (as indeed this was well outside the scope of the research), releasing it to circuit designers for evaluation was not an option. However, many real-world characterization examples were performed in-house to evaluate the practical effectiveness of the algorithm and its implementation. In an industry setting, such examples arise out of necessity, as a byproduct of circuit design. In the academic setting, however, it was instead necessary to borrow examples from industry. Some of the examples were actual parts for use circuit designs, while others were for purposes of characterizing a process, and still others had been contrived for demonstration purposes to mimic typical applications.

### *7.1 Measurement Setup and General Modeling Approach*

Three interconnect characterization examples are presented in this chapter to verify the accuracy and applicability of the modeling approach, and to emphasize the utility of

the resulting model. In each case, TDR measurements of the device under test (DUT) were made with a Tektronix 11801A Digital Sampling Oscilloscope and SD-24 TDR Sampling Head. Signals were coupled to and from the TDR via two high-performance<sup>†</sup>  $50\pm 1\Omega$ ,  $2\pm 0.02$  ns coaxial lines with precision 3.5mm connectors. Figure 7.1 depicts the



**Figure 7.1** General measurement setup employed.

general measurement setup that was used. At the interface to the DUT, some form of fixturing was required in each case to couple the signals in and out of the device, as discussed in Appendix B. The deconvolution scheme outlined in Appendix C was applied to de-embed approximate scattering impulse-response waveforms from the TDR measurements. In reality, this approximate scattering impulse response is that of the DUT combined with the fixturing, as discussed in Appendix B. Appendix C also mentions that additional measurement and software effort allows the de-embedding of the DUT response from the

<sup>†</sup> Although the actual bandwidth of the cables and connectors was not specifically measured, it was guaranteed by the manufacturer to be greater than 20 GHz. In practice it was found to be greater than 27 GHz, the limit of the SD-24 TDR excitation step.

combined response. However, it was decided not to expend the additional effort necessary to implement and perform this aspect of de-embedding, since it would not shed any more light on the modeling algorithm itself, and therefore would not serve the primary purpose of the examples.<sup>†</sup>

For each example, the approximate scattering parameter data as measured and de-embedded from the DUT was used as input to the modeling algorithm implementation as outlined in Chapter 6. This implementation, in turn, applied the extraction algorithm presented in Chapter 5 to the data, resulting in the formulation of a scattering state-space model of the form shown in equation (4.28). In each case, the model was then reduced in complexity while preserving its behavior over different frequency ranges of interest. Finally, the model was converted to admittance and written to a SPICE-compatible netlist.

In order to verify the accuracy of the extracted models, each was simulated under a variety of excitation and termination conditions using HSPICE [59]. The simulation results were then compared to measurements taken of the corresponding DUT as subjected to similar conditions. Using termination conditions for verification which are different from those used to take the measurements used for extraction allows an independent verification of the model's ability to accurately extrapolate to those conditions. Exposure of the DUT to varying excitation conditions was not possible, however, because the output waveform of the SD-24 TDR sampling head is fixed. In order to compare the simulated response of a low-frequency model with that of the DUT, the measured high-frequency data was filtered to emulate excitation by a slower-risetime excitation step. If the system is assumed to be linear, which is indeed a very close approximation to reality, filtering the high-frequency measured response signal in software is equivalent to using a low-frequency excitation for measurement. Although it does not provide the same independent

---

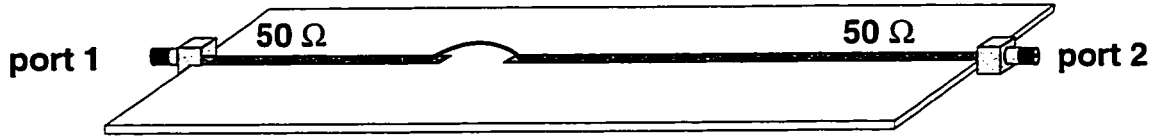
<sup>†</sup> This discussion illustrates one advantage of time-domain measurements over frequency-domain measurements, namely the decreased need for de-embedding. Due to the inherent differences in the internal design of the excitation and measurement systems, raw TDR measurements are very close to physical reality, whereas in the frequency domain, full de-embedding is all but mandatory. In all fairness, however, the software aspects of de-embedding are significantly simpler if the data is to be measured and viewed in the frequency domain.

verification that would be available from measuring with low-frequency excitation, it effectively tests the premise of performing model reduction to trade off accuracy for reduced model complexity. Furthermore, it allows the results of the model reduction, which is typically formulated and applied in the frequency domain, to be viewed and compared against measurement in the time domain. In the comparison process, the practical utility and necessity of the model reduction capability are emphasized.

In each example, the simulated results are shown to closely match with the corresponding measured data for the various termination schemes and excitation conditions under which the model was evaluated. This underscores the validity of the proposed modeling approach for application to real-world interconnect modeling problems, and verifies the accuracy of the algorithm as derived in Chapter 5. In fact, the accuracy of each model is seen to approach the practical limit in which the model is as accurate as the data used to extract it. The fact that each model was extracted automatically using the same software package without user intervention testifies to the robustness of the algorithm, as well as its potential for realization as part of a *bona fide* black-box CAD tool.

## 7.2 Two-port Microstrip Circuit

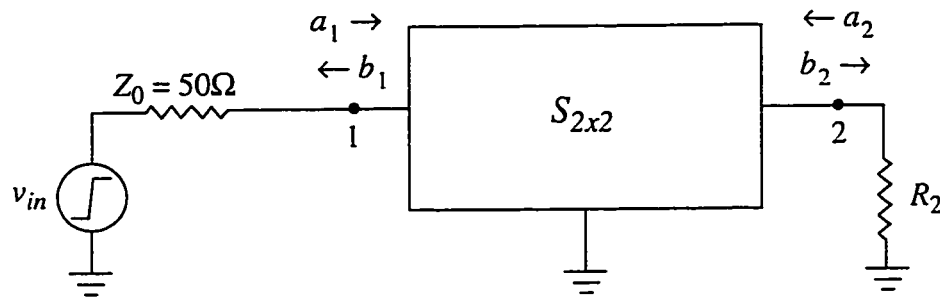
The first example to be presented is the two-port microstrip circuit in Figure 7.2. The circuit consists of two lengths of  $50\Omega$  transmission line connected by a short piece of soldered wire, and has a  $50\Omega$  SMA coaxial connector at each end. The physical length of the circuit is about 14 cm. The circuit was borrowed from a demonstration board built by Tektronix to exemplify the usefulness of TDR for this type of application—making measurements of non-ideal transmission line structures. The excess inductance introduced by the wire soldered into the microstrip run nicely resembles that of a via structure which might appear in a typical circuitboard run. The fact that its response is somewhat exaggerated compared to that of a single via hole is useful in that it ensures that the effects of the wire dominate over those of the SMA connectors present at either end. This allows the circuit to be characterized to a reasonable level of accuracy without having to rigorously



**Figure 7.2** Two-port microstrip example circuit. The physical distance between ports 1 and 2 is approximately 14 cm.

de-embed the DUT response from the combined response of the DUT and SMA connectors, as mentioned in Section 7.1.

Figure 7.3 shows an idealized version of the setup used to take measurements of the circuit in Figure 7.2, where the coaxial cables seen in Figure 7.1 are omitted. As discussed in Appendix A, port 1 is excited by the TDR, and the resultant voltage waveforms are measured at both ports.



**Figure 7.3** Idealized setup for taking measurements at port 1 of the two-port example circuits from Sections 7.2 and 7.4. The same setup, which ignores any fixturing issues, was also used to generate simulation data for model verification in each example.

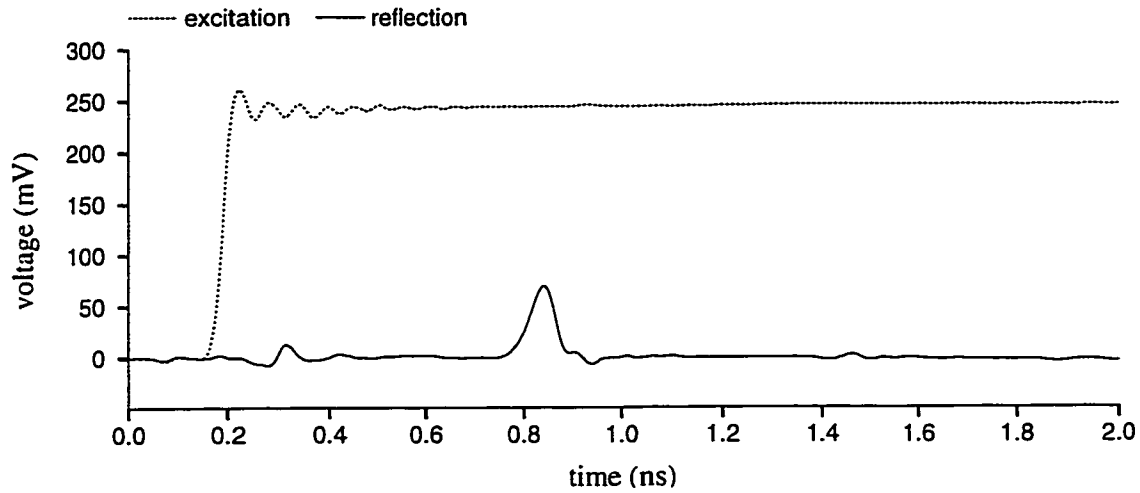
The measurement of total voltage requires that the reflected voltage be calculated from equation (5.27). A simplified calibration procedure assumes that all impedances, including that of the coaxial cable, are  $50\ \Omega$ . A  $50\ \Omega$  output impedance and  $50\ \Omega$  cable implies that the steady-state voltage measured prior to the DUT reflection consists entirely of the incident step, since reflection is by definition zero from a  $50\ \Omega$  transmission line.



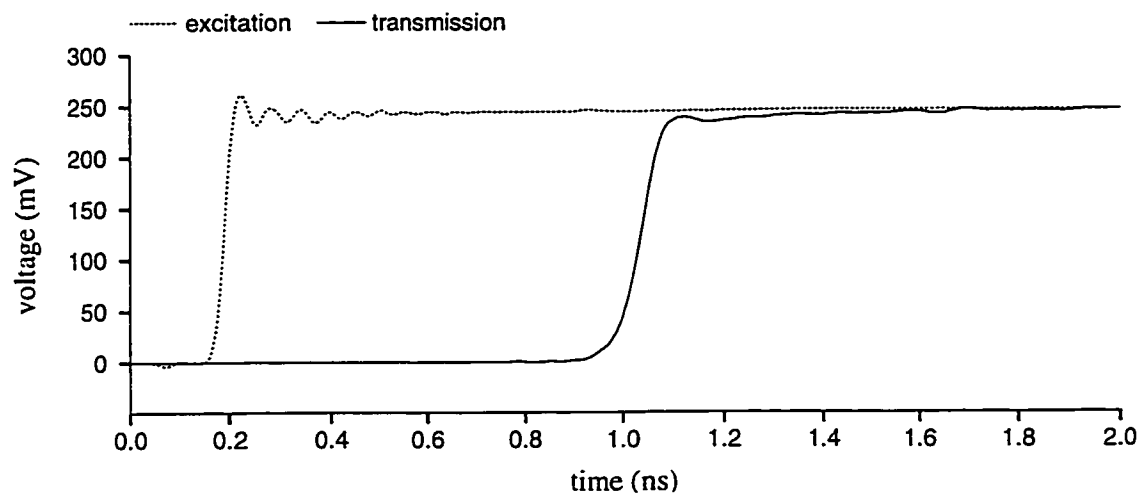
This value may then be subtracted from the voltage waveform to yield the reflection waveform of the DUT according to equation (5.27). The transmission waveform is ideally measured at port 2 across a resistor of value  $R_2=50\ \Omega$  (see Appendix A), which in practice is part of the high-frequency sampling head. If it is assumed that this resistor value is exact, then the total voltage waveform measured at port 2 is equal to the reflected voltage  $b_2$  at that port, since by definition the incident voltage  $a_2$  must be zero. If higher accuracy is desired, one of the more rigorous de-embedding schemes to which Appendix C alludes must be implemented, and this was deemed unnecessary for the purposes of this research.

Prior to discussing the model extraction and subsequent verification, it is worthwhile to look into the TDR reflection and transmission signatures of the DUT, to more effectively quantify what is most important to model. Figure 7.4 shows the measured reflection and transmission waveforms for the circuit in Figure 7.2. The important aspects of the waveforms in terms of digital circuit design are reflection and delay, as well as some dispersion. The reflection caused by the wire can be seen as an inductive bump occurring at around 0.85 ns in the reflection waveform. Excess inductance causes a transient positive bump in the reflection signature because of its higher impedance at high frequency. Delay and dispersion are observed in the transmission waveform by comparing it to the incident waveform, where it may be seen that once the reflection begins from port 1 at around 0.15 ns, an additional 0.8 ns elapses before the transmitted signal is seen at port 2, which gives an approximate measure of delay, or transit time. For such approximate measurements, we ignore the difficulty in determining exactly when the reflection begins from port 1, as well as the 20 ps uncertainty introduced by each of the coaxial cables. Dispersion is observed by noting that the risetime of the transmitted waveform is 100 ps, whereas the risetime of the input step is less than 30 ps. These aspects of the waveform should be taken into account as the accuracy of the model is evaluated.

Approximate time-domain scattering parameters were deconvolved out of the  $S_{11}$  and  $S_{21}$  waveforms shown in Figure 7.4 (and from the  $S_{12}$  and  $S_{22}$  waveforms not shown) and subsequently used as input data to the modeling algorithm. By application of the algorithm and implementation of Chapters 5 and 6, it was found that 69 poles were neces-



(a)



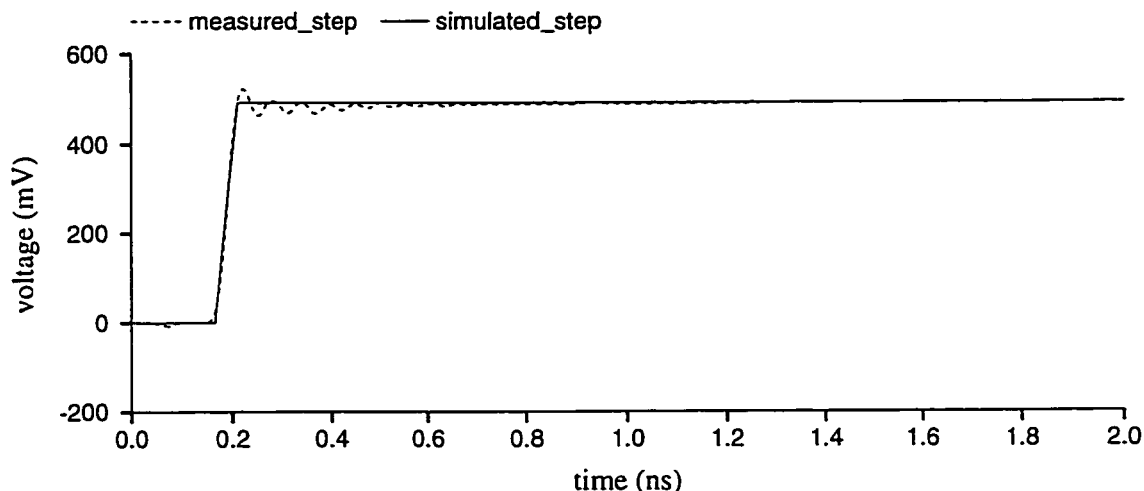
(b)

**Figure 7.4** Measured excitation step waveform and step response waveforms for the circuit in Figure 7.2. (a)  $S_{11}$  response (reflection) obtained by exciting and measuring at port 1. (b)  $S_{21}$  response (transmission) obtained by exciting port 1 and measuring at port 2.

sary to represent the four scattering parameters to within 2% accuracy from DC to 32 GHz. After adding a node at each port to transform the model into admittance as

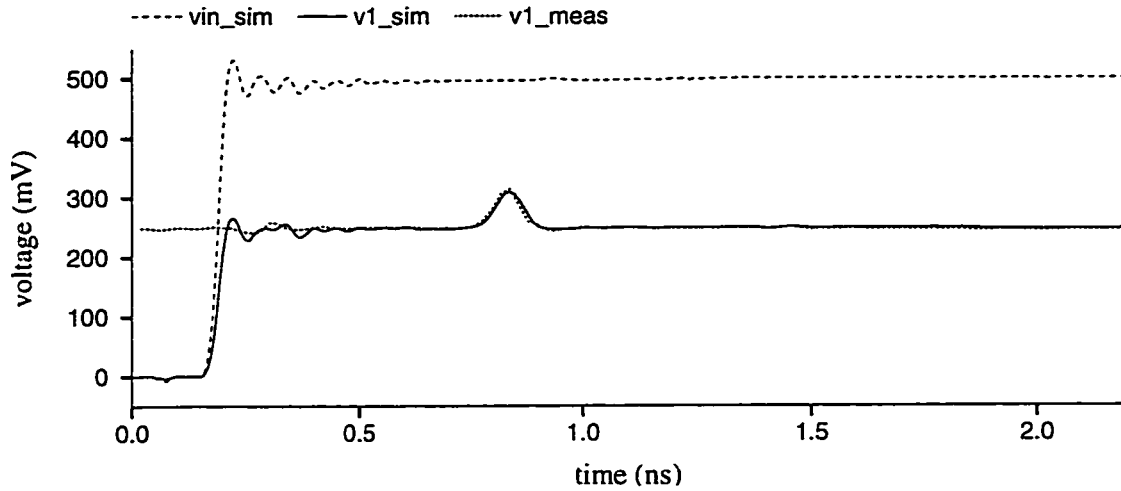
described in Section 5.7.2, the generated SPICE netlist representing the DUT had 71 nodes, and was written into its own subcircuit file.

To evaluate the accuracy of the 32 GHz extracted model described above, port 1 was driven by a 500 mV, 35 ps step input through a  $50\Omega$  output resistance, and port 2 was terminated by a resistance  $R_2=50\Omega$ , as shown in the idealized measurement setup in Figure 7.3. The excitation signal used for the simulations was a trapezoidal step whose risetime was chosen to match that of the measured input step from Figure 7.4. Figure 7.5 is a com-

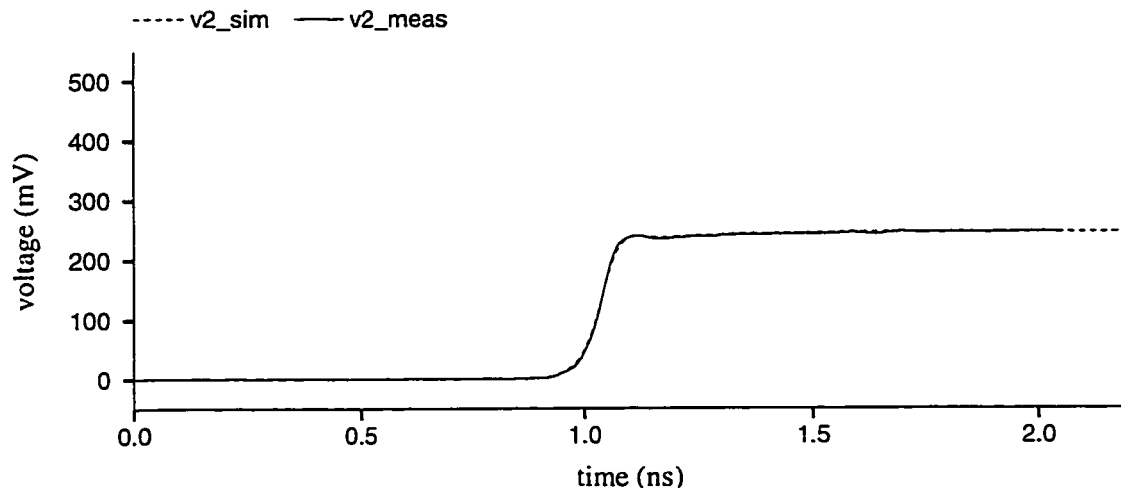


**Figure 7.5** Comparison between the excitation step used in simulation and that used in taking the measurements.

parison of the excitation waveforms used for measurement and simulation, and a minor discrepancy can be seen at the corners of the steps, the upper corners in particular. This is because the simulation step has some additional high-frequency content and fails to model the overshoot of the measurement step. However, these effects are second order, and can be ignored unless a very rigorous comparison is made between the simulated and measured responses. The step response of the model for matched terminations (that is,  $R_2=50\Omega$ ) was compared to the measured TDR step response, as shown in Figure 7.6. The reflection and transmission waveforms are seen to compare favorably, with reflection,



(a)

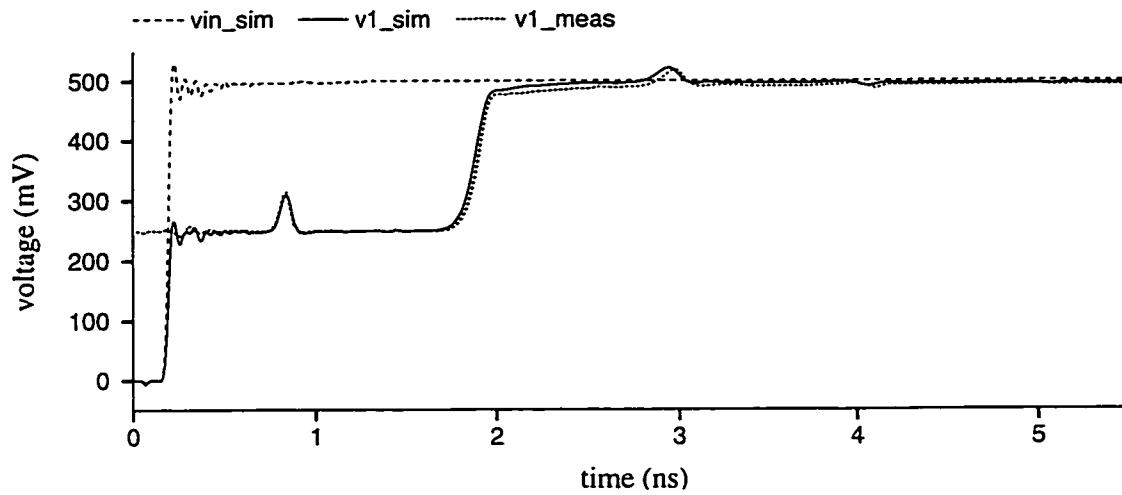


(b)

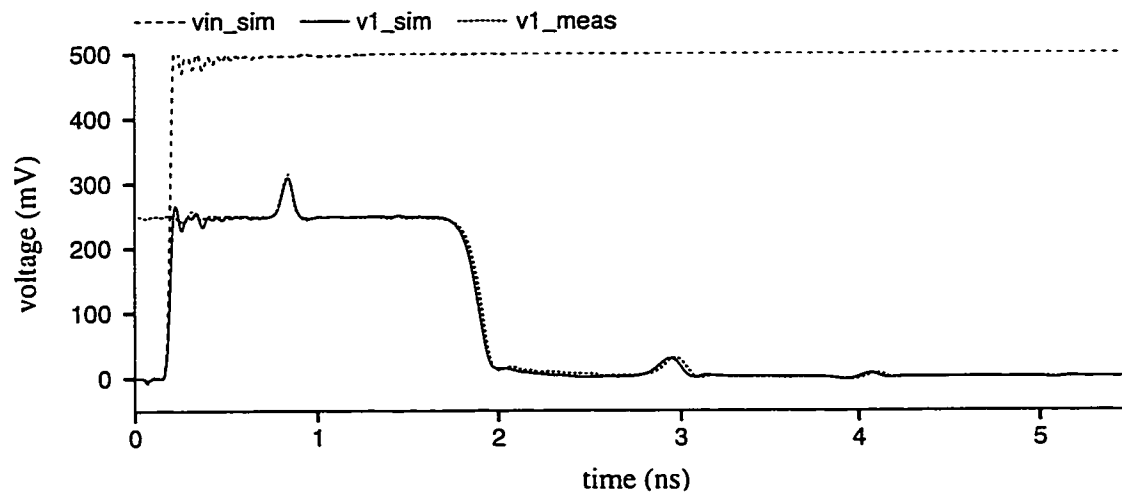
**Figure 7.6** Comparison of model response with matched-load measured data used to extract the model originally. (a) Reflection ( $S_{11}$  step response) comparison. (b) Transmission ( $S_{21}$  step response) comparison.

delay, and dispersion accurately modeled. It should be noted that this comparison is against the same data that was used in the extraction of the model, and in that sense it does not verify the ability of the model to extrapolate into different termination schemes. However, circuit theory predicts that if the scattering waveforms are accurately characterized

then the model is valid for all termination schemes, and this is verified in Figures 7.7(a) and 7.7(b). Figure 7.7(a) is an open-circuit comparison, showing simulated voltage at port



(a)



(b)

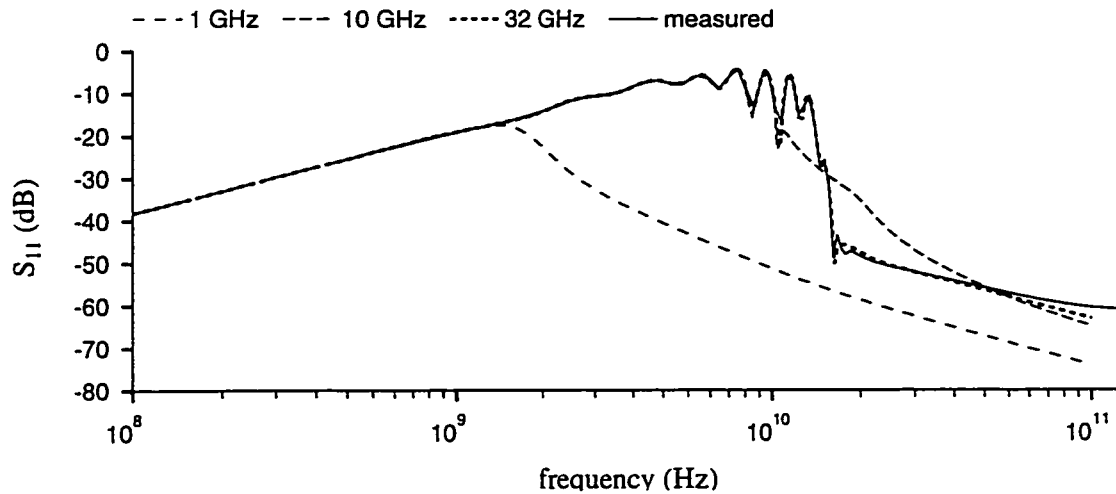
**Figure 7.7** Comparison of modeled and measured data for two-port example. (a) Open-circuit reflection at port 1 in response to a 35 ps step. (b) Short-circuit reflection at port 1 in response to a 35 ps step.

1 for  $R_2=50\text{k}\Omega$  with TDR data measured for port 2 unterminated. Figure 7.7(b) verifies short-circuit behavior by comparing simulated voltage at port 1 for  $R_2=0.001\Omega$  with TDR data measured for port 2 short-circuited. The reflections occurring internal to the DUT due to the mismatched loads at port 2 allow the opportunity to see the model response extrapolated in time (as well as in termination space), and it is seen that the successive reflections in each case are accurately modeled.

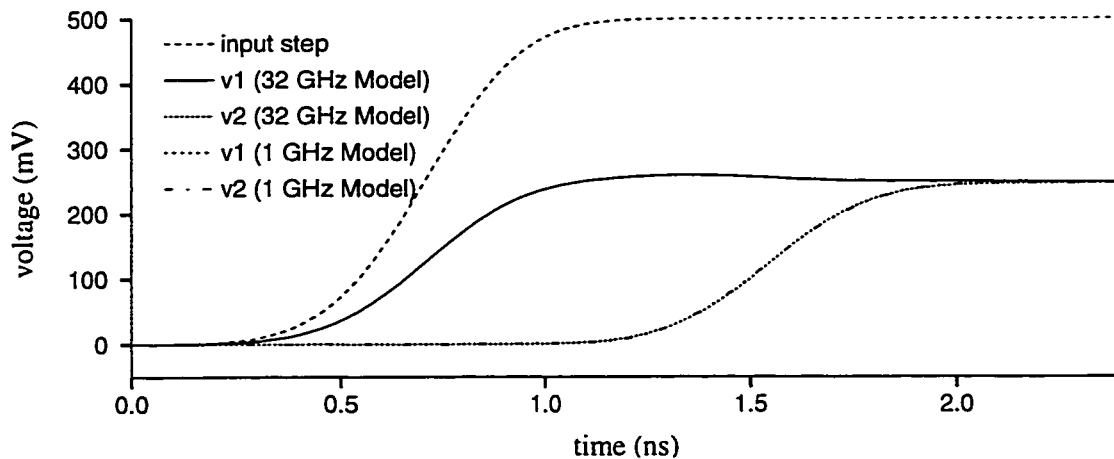
In order to test the effectiveness of the model reduction step, reduced models were created which were accurate from DC to 1 GHz and from DC to 10 GHz. The models had 12 and 49 nodes respectively, as compared to the 71 nodes required by the 32-GHz model. Figure 7.8a compares the frequency-domain reflections at port 1 of the various extracted models with that of the measured responses (obtained via FFT methods), and it can be seen that each model is valid within the frequency range of interest. Figure 7.8b gives a time-domain comparison between the reflection and transmission responses of the 1-GHz and 32-GHz models. Each model was terminated with  $R_2=50\Omega$  and excited with a 500 ps step. The two models are seen to behave identically given this excitation, which underscores the utility of the model reduction capability in trading model bandwidth for reduced model complexity in applications which would not require a high-frequency model.

### 7.3 *Four-port Microstrip Circuit*

The four-port microstrip circuit in Figure 7.9 consists of two mirrored microstrip lines of varying width terminated at each end by  $50\Omega$  SMA coaxial connectors. There is no ground plane under the majority of each trace, resulting in significant crosstalk between the two lines. The physical length of each run is about 14 cm. The circuit came from the same demonstration board as that of the previous example, but was intended to showcase the capabilities of differential TDR and time-domain transmission (TDT). For the purposes of this research, the primary reason for running this example was to investigate the ability of the algorithm to handle crosstalk between separate transmission-line structures. However, this particular example was known to be a difficult case for the modeling approach in [21] to handle. The difficulty arises due to the combination of the very



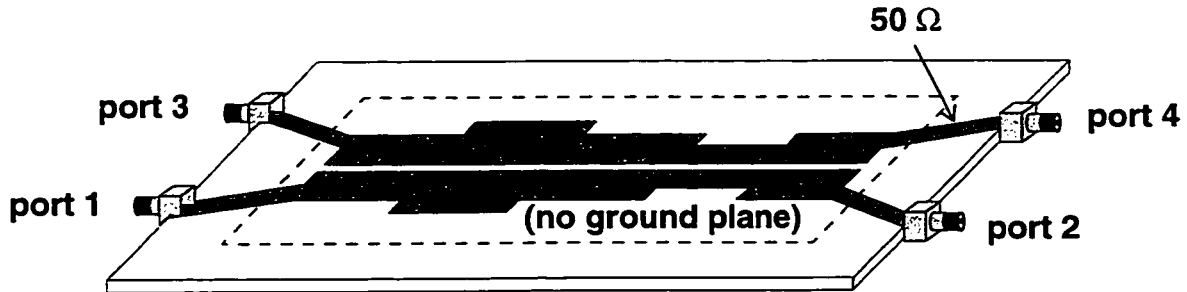
(a)



(b)

**Figure 7.8** Evaluation of the model of the circuit in Figure 7.2. (a) Comparison of 1 GHz, 10 GHz, and 32 GHz reflection magnitudes at port 1 with measured data in the frequency domain. (b) Matched-load reflection at port 1 and transmission at port 2 for 1 GHz and 32 GHz models in response to a 500 ps step.

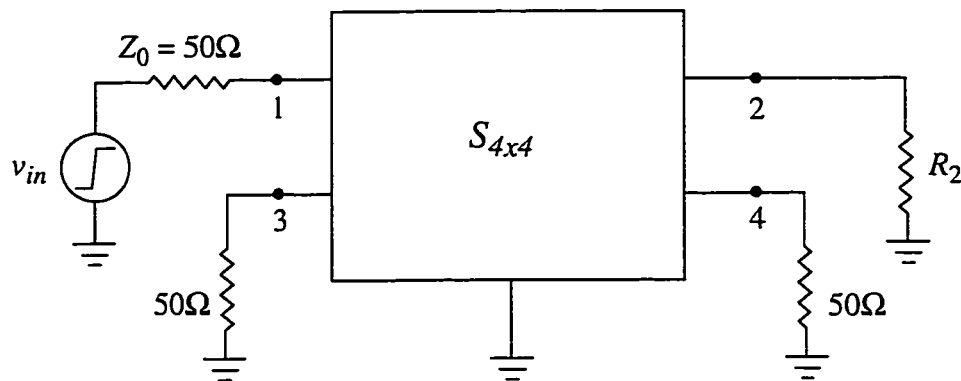
fast time constants (as compared to the sampling interval) encountered initially in the response and the lower-frequency ringing which continues for some time. In this sense it



**Figure 7.9** Four-port microstrip example circuit. The physical distance between ports 1 and 2 is approximately 14 cm.

was a good test of the robustness of the modeling algorithm presented in this work, and it served as a verification as well, since it was handled without a problem.

An idealized diagram of the setup used for measurements at port 1 and subsequent simulations is shown in Figure 7.10. A total of 148 internal nodes were required to model

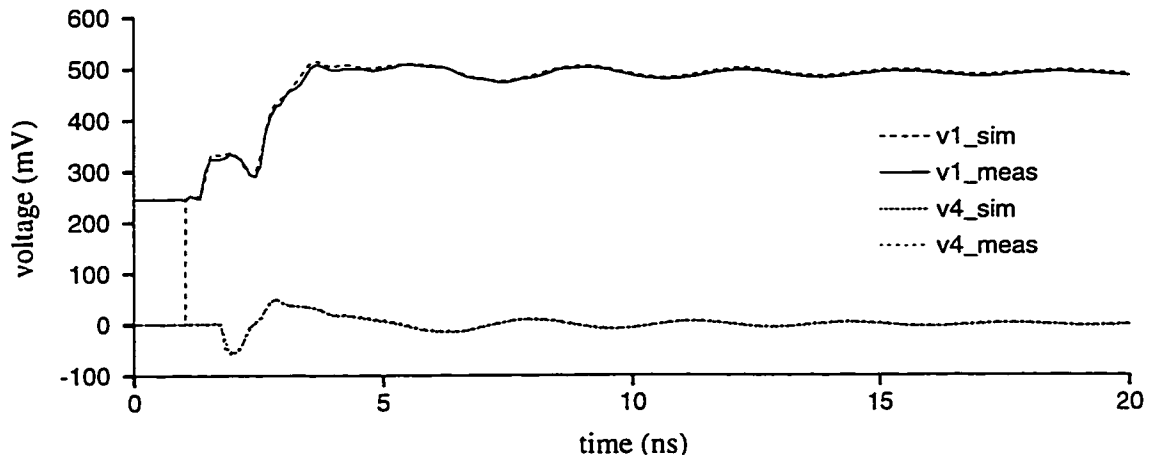


**Figure 7.10** Idealized setup for taking measurements at port 1 of the four-port circuit in Figure 7.9. The same setup, which ignores any fixturing issues, was also used to generate simulation data for model verification.

the 16 scattering parameters to within 2% accuracy up to 13 GHz. The majority of the nodes were spent in modeling the high-frequency behavior occurring early in the response, which is barely observable at the scale of Figure 7.11. To evaluate the accuracy of the model, the simulated step response waveforms with matched loads at every port



were compared to the measured step response waveforms and found to be in close agreement. To test the model under a termination scheme which was somewhat independent of the one under which the measurements were obtained, the response of the DUT was measured with matched loads at all ports except port 2, which was left unterminated, which is the setup in Figure 7.10 if  $R_2$  is removed. The corresponding simulation was performed with port 1 driven by a 500 mV, 35 ps step input through a  $50\Omega$  output impedance and port 2 terminated by  $50k\Omega$ , while ports 3 and 4 were terminated by  $50\Omega$  resistors. This is the setup shown in Figure 7.10 if  $R_2=50k\Omega$ . Figure 7.11 compares the measured responses for

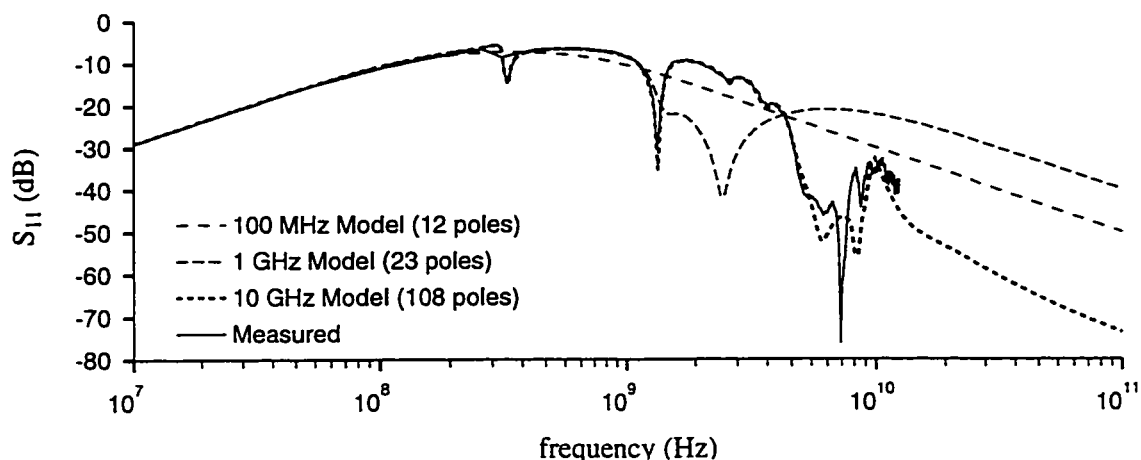


**Figure 7.11** Simulated and measured open-circuit reflection at port 1 and cross-talk at port 4 for the circuit in Figure 7.9. Simulations were performed using the 13 GHz model and a 35 ps input step.

reflection at port 1 and crosstalk at port 4 with the simulated responses at the same ports. The simulated and measured curves are nearly overlapping at the scale shown, and it is seen that the crosstalk waveform is accurately reproduced.

Reduced models were also extracted which were accurate to within 2 percent from DC up to 100 MHz, 1 GHz, and 10 GHz. These models contained 12, 23, and 103 nodes, respectively, which emphasizes that the majority of the nodes in the model of this particular circuit go to characterize its high-frequency behavior. The magnitude of the reflection

at port 1 for each of the models is shown in Figure 7.12, and the magnitude of the mea-



**Figure 7.12** Frequency-domain reflection magnitude comparison of measurement and the 100 MHz, 1 GHz, and 10 GHz models of the circuit in Figure 7.9.

sured reflection at port 1 is included for comparison. It can be seen that each model is accurate in its region of interest, and the usefulness of such a reduction step is again seen in the significantly lower complexity achieved by reducing the accurate bandwidth of the model from 10 GHz to 1 GHz.

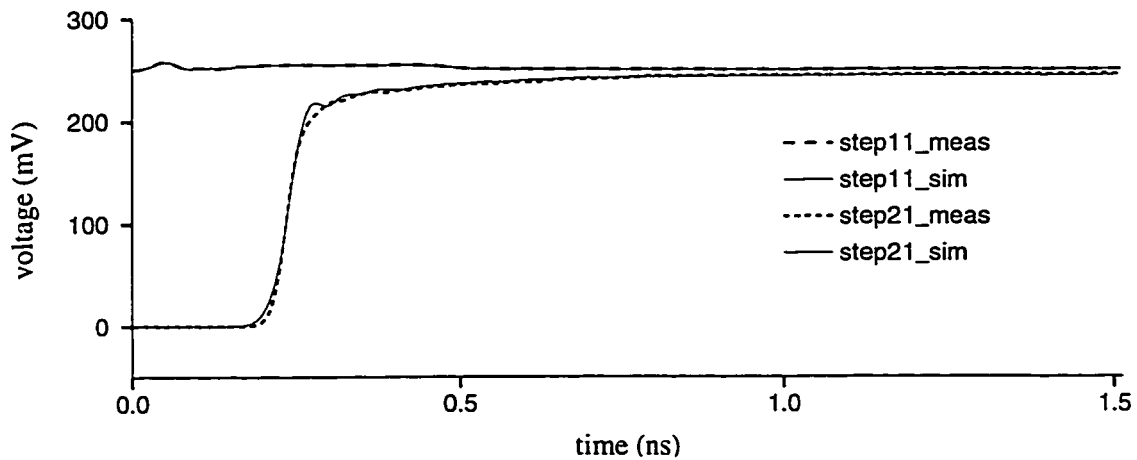
#### 7.4 Lossy MCM Line Characterization

A 4 cm lossy transmission line from an MCM (multichip module) substrate characterization coupon was also characterized. The coupon was originally fabricated by a digital design company to test the viability of its MCM process for use in fast digital applications. For the purposes of this research work, it was used to demonstrate the usefulness of the modeling approach for MCM applications, and specifically its capability to accurately characterize dispersion and skin effect. Skin effect, which causes series resistance of a transmission line to increase with frequency, becomes increasingly important in

narrow MCM traces, and is difficult to accurately model with non-measurement-based approaches.

The DUT was probed using Cascade Microtech 250  $\mu\text{m}$  coplanar probes, and approximate impulse-response scattering parameters were deconvolved out of the measured step-response waveforms. A model for the line was then extracted, and 88 poles were chosen by the algorithm to represent the four scattering parameters to within 2% accuracy up to 20 GHz. After adding the transformers to the two ports, the generated SPICE netlist representing the admittance of the DUT had 90 nodes.

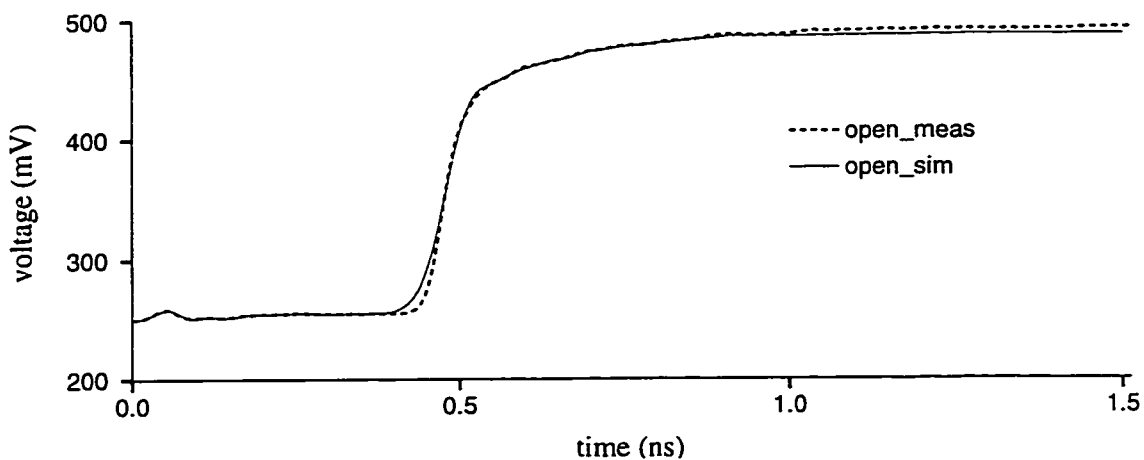
To evaluate the accuracy of the extracted model, it was first excited by a 35-ps step through a 50  $\Omega$  resistor while port 2 was also terminated by a matched load. This is effectively the setup shown in Figure 7.3 in which  $R_2=50\Omega$ . Figure 7.13 compares the results



**Figure 7.13** Simulated and measured waveforms for matched reflection and matched transmission for unreduced model.

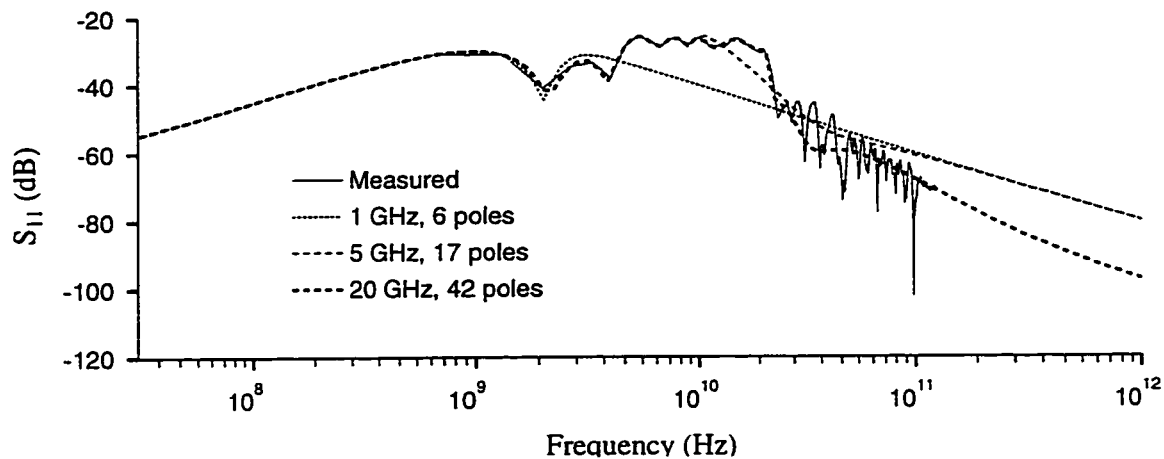
of this simulation with the original TDR data measured under a matched-load excitation and termination scheme. Both the reflection (step11) and transmission (step21) waveforms are seen to be accurately modeled, as well as dispersion and skin effect. Dispersion is seen in the slowing down of the transition in the transmitted step waveform, in that the

risetime of the original input step was 35 ps, whereas the transmitted step has a risetime of 100 ps. Skin effect is observed as the rounded upper edge of the same waveform, which is a result of increased reflection at high frequency due to the changing series resistance. Open-circuit measurements were also taken with port 2 left unterminated, and corresponding simulations were performed with port 2 terminated in a 5 M $\Omega$  resistor. Figure 7.14 compares the results of this simulation with measured open-circuit TDR data, and in both cases the delay and loss are seen to be accurately represented by the model.



**Figure 7.14** Comparison of open-circuit reflection waveforms for unreduced model.

In order to again test the model reduction step, the extracted model, which is accurate between DC and 20 GHz, was reduced in size to create two additional models which were accurate to within 5% between DC and cutoff frequencies of 1 GHz and 5 GHz. The models contained 6 and 17 poles, respectively, while the 20 GHz model had 42 poles. The magnitude of the  $S_{11}$  frequency-domain reflection waveform for each model is shown in Figure 7.15, in which each is seen to be accurate over its desired frequency range. This observation once again shows that the number of nodes may be significantly reduced if accuracy is only required over a specific band of frequencies.



**Figure 7.15** Magnitude plot of  $S_{11}$  for measured data as well as for 1 GHz, 5 GHz, and 20 GHz reduced models.

### 7.5 Conclusion

The three examples presented in this chapter serve to emphasize the significant features of the modeling algorithm presented in this dissertation. Most significantly, it is shown that the approach is valid for application to real-world interconnect design and modeling problems. From a mathematical and numerical perspective, the claims of Chapters 5 and 6 concerning the accuracy and robustness of the extraction algorithm are justified. The accuracy of the approach is seen in the many comparisons of simulation results against measurement. The fact that all examples were run using the same piece of software without user intervention strongly implies that the approach is in fact robust enough for automation. The compatibility of the model itself with existing CAD tools is evident in that all simulations were run using HSPICE, an industry standard commercial simulator. Finally, the effectiveness of the model reduction step to simplify a model for use over a narrower frequency range, as well as the utility of such a reduction capability in practical modeling situations, was seen in each of the characterization examples.

## Chapter 8

### *CONCLUSION*

The purpose of this research effort was to develop a robust algorithm and an associated CAD implementation to assist designers of high-speed digital circuits and radio-frequency (RF) analog circuits in developing models of the off-chip interconnections inherent in their designs. As might be expected, the process of accomplishing this goal has entailed a significantly broader effort than might be denoted by the above statement.

One very important aspect of the research is that the resultant CAD tool is intended to “assist designers.” In order to develop such a tool, it was first necessary to speak not only with those people familiar with the electronic design industry, but to speak with the designers themselves, as to how necessary it is to have an effective interconnect modeling tool. The invariant answer, whether the discussions took place at industry conferences, in various modeling and design labs, or at application-oriented research institutions, was that *the need was urgent*, not only for *interconnect modeling tools*, but for *application knowledge* in the measurement and simulation areas. The primary reason for the urgency of the need was increasing digital circuit speeds and data rates, as well as the dramatic growth of consumer RF products—and both areas continue to increase rapidly. There is no question that the need for effective interconnect modeling tools will continue to grow for as long as these trends continue, and they show no signs of abating in the near future.

Assuming, then, that accurate interconnect modeling is necessary for effective circuit design, a logical question is what the desired characteristics of an interconnect modeling tool would be. In talking with those involved in the design industry, the two most significant issues were found to be *compatibility* and *automation*, which are discussed in Sections 2.1.3 and 3.2, respectively, and these two issues have therefore been allotted a high priority in this research effort. Compatibility with existing CAD tools has been achieved by developing the extraction algorithm such that the resultant model is composed of standard SPICE elements and is written in SPICE netlist format. This allows the model to be evaluated by virtually any industry simulator, as SPICE is a well-known (if unofficial) standard across the industry. Automation has also been achieved to a large extent in the implementation of the modeling algorithm. Because of limitations on available resources, a fully automated tool spanning the entire characterization flow from measurement to verification was not developed, and the research was instead limited in scope to the flow shown in Figure 6.1.<sup>†</sup> Furthermore, in the implementation of the flow in software, a small amount of automation was withheld in deference to flexibility for the sake of continuing research.

Since a completely automated tool spanning the entire characterization flow was not developed, it was not possible to loan the tool to industry partners for evaluation purposes. However, we were able to obtain parts and data from several partners in order to test the algorithms ourselves at the level of automation which was achieved. The verification results presented in Chapter 7 show the algorithm to accurately and effectively model typical interconnect structures at varying levels of complexity as required by different applications. It is this contact with circuit designers and modelers, and the verification of the algorithm on industry-applicable problems, that lends credence to the interconnect modeling approach presented in this dissertation, and underscores its significant potential to be of practical use in the electronic design industry.

<sup>†</sup> The flow in Figure 6.1 explicitly excludes measurement and de-embedding, as these issues have already been researched to a high degree, and are better implemented in industry due to the expense involved. Furthermore, each one decouples cleanly from the modeling algorithm, and has already been automated to some extent in industry.

The other important goal of this research effort as stated above was the development of a *robust* algorithm. Robustness is discussed in Section 5.1, and in that sense can be seen as a corollary to automation which must be achieved in the mathematical and numerical domains. The relationship between robustness and automation exists because an unreliable CAD tool cannot be considered truly automated by an end user, since it must constantly be monitored and tweaked.

The development of a robust interconnect modeling algorithm was achieved by combining concepts from a variety of mathematical and engineering fields. Theoretical derivation of a multiport system identification formulation was based on network theory and linear algebra, which combine to form a portion of state-variable theory. And as is necessary for any algorithm being designed for computer implementation, every step in the derivation was tempered by an analysis of the numerical issues involved. This serves not only to maintain robustness and reliability, but also to ensure that the algorithm can be practically implemented as derived.

Concepts from various aspects of electrical engineering were also incorporated to ensure that the approach was not only robust, but would lead to a practically useful implementation. Insight into the internal workings and external interfaces of SPICE-based simulators was necessary to design a model which could be quickly and easily evaluated by existing industry simulation tools. Experience and advice in the area of high-frequency measurement was requisite for developing an algorithm whose inputs are practically obtainable. A knowledge of circuit design and interconnect design in particular was necessary to ensure that the approach was capable of modeling those circuits which need to be modeled. And as mentioned previously, this knowledge was often complemented by advice from those on the cutting edge of interconnect design.

As a result of this research effort, an automatic modeling capability exists for designers of off-chip interconnect, where such a tool is urgently needed, and for the most part did not previously exist. Furthermore, the inputs and outputs of the algorithm have been chosen to ensure the practicality and compatibility of the approach in the existing circuit design community. These two statements embody the contribution of this research to



the electronic design community at large, and it is intended that the modeling approach which has been developed will prove to be of significant practical use within that community, and possibly within others facing similar modeling challenges.

## Bibliography

1. Liang Chi Shen and Jin Au Kong, *Applied Electromagnetism*, Boston: PWS Engineering, 1987.
2. Akira Ishimaru, *Electromagnetic Wave Propagation, Radiation, and Scattering*, New Jersey: Prentice-Hall, 1991.
3. David Kahaner, Cleve Moler and Stephen Nash, *Numerical Methods and Software*, New Jersey: Prentice Hall, 1989.
4. Randall W. Jensen and Lawrence P. McNamee, eds., *Handbook of Circuit Analysis Languages and Techniques*, Englewood Cliffs, New Jersey: Prentice-Hall, 1976.
5. C.-W. Ho, A. E. Ruehli, and P. A. Brennan, "A Modified Nodal Approach to Network Analysis," *IEEE Transactions on Circuits and Systems*, vol. 22, no. 6, pp. 504-509, September, 1975.
6. Jiri Vlach and Kishore Singhal, *Computer Methods for Circuit Analysis and Design*, New York: Van Nostrand Reinhold, 1983.
7. A. Ruehli, "Equivalent Circuit Models for Three Dimensional Multiconductor Systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 22, no. 3, pp. 216-224, March, 1974.
8. A. E. Ruehli and H. Heeb, "Circuit Models for Three-dimensional Geometries Including Dielectrics," *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, no. 7, pp. 1507-1516, July, 1992.
9. V. Raghavan, R.A.Rohrer, L.T. Pillage, J.Y. Lee, J.E. Bracken, M.M. Alaybeyi, "AWE-Inspired," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 18.1.1-18.1.8, May, 1993.

10. T.J. Su and R. R. Craig, "Model Reduction and Control of Flexible Structures Using Krylov Vectors," *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pp. 260-267, January-February, 1991.
11. P. Feldmann and R.W. Freund, "Efficient Linear Circuit Analysis by Padé Approximation via the Lanczos process," *IEEE Transactions on Computer Aided Design*, vol. 14, no. 5, pp. 639-649, May 1995.
12. K. J. Kerns and A. T. Yang, "Preservation of Passivity During RLC Network Reduction via Split Congruence Transformations," *Proceedings of the 34th Design Automation Conference*, pp. 34-39, June, 1997.
13. Sidney Frankel, *Multiconductor Transmission Line Analysis*, Dedham, MA: Artech House, 1977.
14. J.A. Brandao Faria, *Multiconductor Transmission-Line Structures*, New York: Wiley, 1993.
15. F. H. Branin, "Transient Analysis of Lossless Transmission Lines," *Proceedings of the IEEE*, vol. 55, no. 11, pp. 2012-2013, November, 1967.
16. F. Y. Chang, "Transient Analysis of Lossless Coupled Transmission Lines in a Non-homogeneous Dielectric Medium," *IEEE Transactions on Microwave Theory and Techniques*, vol. 18, no. 9, pp. 616-626, September, 1970.
17. D. S. Gao, A. T. Yang, S. M. Kang, "Accurate Modeling and Simulation of Interconnection Delay and Crosstalks in High-speed Integrated Circuits," *IEEE Transactions on Circuits and Systems*, vol. 37, no. 1, pp. 1-9, January, 1990.
18. D. Winklestein, M. B. Steer, R. Pomerleau, "Simulation of Arbitrary Transmission Lines with Nonlinear Terminations," *IEEE Transactions on Circuits and Systems*, vol. 38, no. 4, pp. 418-422, April, 1991.

19. L. P. Vakanas, A. C. Cangellaris, O. A. Palusinski, "Scattering Parameter-based Simulation of Transients in Lossy Nonlinearly-terminated Packaging Interconnections," *IEEE Transactions Components, Packaging and Manufacturing Technology, Part B: Advanced Packaging*, vol. 17, no. 4, pp. 472-479, November, 1994.
20. Giri Devarayanadurg and Mani Soma, "An Interconnect Model for Arbitrary Terminations Based on Scattering Parameters," *Analog Integrated Circuits and Signal Processing*, vol. 5, no. 1, pp. 31-45, January, 1994.
21. S. D. Corey and A. T. Yang, "Interconnect Characterization Using Time Domain Reflectometry," *IEEE Transactions on Microwave Theory and Techniques*, vol. 43, no. 9, pp. 2151-2156, September, 1995.
22. R. Kipp, C. H. Chan, A. T. Yang, J. T. Yao, "Simulation of High-Frequency Integrated Circuits Incorporating Full-wave Analysis of Microstrip Discontinuities," *IEEE Transactions on Microwave Theory and Techniques*, vol. 41, no. 5, pp. 848-854, May, 1993.
23. A. Djordjevic and T. K. Sarkar, "Analysis of Time Response of Lossy Multiconductor Transmission Line Networks," *IEEE Transactions on Microwave Theory and Techniques*, vol. 35, no. 10, pp. 898-908, October, 1987.
24. B. J. Cooke, J. L. Prince, A. C. Cangellaris, "S-Parameter Analysis of Multiconductor Integrated Circuit Interconnect Systems," *IEEE Transactions on Computer Aided Design of Integrated Circuits*, vol. 11, no. 3, pp. 353-360, March 1992.
25. S. Corey and A. T. Yang, "Interconnect Characterization Using Time Domain Reflectometry," *Proceedings of the IEEE 3rd Topical Meeting on Electrical Performance of Electronic Packaging*, pp. 189-191, November, 1994.

26. S. D. Corey, K. J. Kerns, and A. T. Yang, "Automatic Measurement-Based Characterization of Lossy MCM Line Using Lumped Elements," *Proceedings of the IEEE 5th Topical Meeting on Electrical Performance of Electronic Packaging*, pp. 144-146, November, 1996.
27. S. D. Corey and A. T. Yang, "Automatic Netlist Extraction for Measurement-Based Characterization of Off-Chip Interconnect", *Proceedings of the IEEE/ACM International Conference on Computer Aided Design*, pp. 24-29, November, 1996.
28. S. D. Corey and A. T. Yang, "Automatic Netlist Extraction for Measurement-Based Characterization of Off-Chip Interconnect," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, no. 10, pp. 1934-1940, October, 1997.
29. *11801A Digital Sampling Oscilloscope User Manual*, Beaverton, OR: Tektronix, Inc., 1992.
30. L. A. Hayden and V. K. Tripathi, "Calibration Methods for Time Domain Network Analysis," *IEEE Transactions on Microwave Theory and Techniques*, vol. 41, no. 3, pp. 415-420, March, 1993.
31. D. F. Williams and R. B. Marks, "Accurate Transmission Line Characterization," *IEEE Microwave and Guided Wave Letters*, vol. 3, no. 8, pp. 247-249, August, 1993.
32. R. B. Marks and D. F. Williams, "Electrical Characterization Methods for High-Speed Interconnections," *The International Journal of Microcircuits and Electronic Packaging*, vol. 18, no. 3, pp. 207-216, Third Quarter, 1995.
33. *Circuit Analysis Reference Manual*, Irvine, CA: MicroSim Corporation, 1994, pp. 4-78.
34. Y. Hua and T. K. Sarkar, "Matrix Pencil Method for Estimating Parameters of Exponentially Damped/Undamped Sinusoids in Noise", *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 38, no. 5, pp. 814-824, May, 1990.

35. J. M. Jong and V. K. Tripathi, "Time Domain Characterization of Interconnect Discontinuities in High Speed Circuits," *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, vol. 15, pp. 497-504, August, 1992.
36. L. A. Hayden and V. K. Tripathi, "Characterization and Modeling of Multiple Line Interconnections from Time Domain Measurements," *IEEE Transactions on Microwave Theory and Techniques*, vol. 42, no. 9, pp. 1737-1743, September, 1994.
37. *IPAS10 Interconnect Parameter Analyzer Instruction Manual*, Beaverton, OR: Tektronix, Inc., 1996.
38. "The Z-Profile Algorithm," *Tektronix Application Note*, Beaverton, OR: Tektronix, Inc., 1993.
39. Edmund K. Miller, ed. *Time Domain Measurements in Electromagnetics*, New York: Van Nostrand Reinhold, 1986, pp. 87-93.
40. Edward J. Rothwell, Weimin Sun, "Time Domain Deconvolution of Transient Radar Data," *IEEE Transactions on Antennas and Propagation*, vol. 38, no. 4, pp. 470-475, April, 1990.
41. J. M. Jong, L. A. Hayden and V. K. Tripathi, "Impedance-Loss Profile Characterization of Interconnection Structures," *International Journal of Microcircuits and Electronic Packaging*, vol. 18, no. 3, pp. 193-199, Third Quarter, 1995.
42. J. M. Jong, B. Janko, and V. K. Tripathi, "Equivalent Circuit Modeling of Interconnects from Time-Domain Measurements," *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, vol. 16, no. 1, pp. 119-126, February, 1993.
43. "TDR Tools in Modeling Interconnects and Packages," *Tektronix Application Note*, Beaverton, OR: Tektronix, Inc., 1993.

44. Someshwar C. Gupta, *Transform and State Variable Methods in Linear Systems*, New York: John Wiley & Sons, 1966.
45. Athanasios Papoulis, *Circuits and Systems: A Modern Approach*, New York: Holt, Rinehart, and Winston, 1980.
46. Donald E. Scott, *An Introduction to Circuit Analysis: A Systems Approach*, New York: McGraw-Hill, 1987.
47. Alan V. Oppenheim and Alan S. Willsky, *Signals and Systems*, New Jersey: Prentice-Hall, 1983.
48. Gene H. Golub and Charles F. van Loan, *Matrix Computations*, Baltimore: Johns Hopkins, 1989.
49. Guillermo Gonzalez, *Microwave Transistor Amplifiers: Analysis and Design*, Englewood Cliffs, New Jersey: Prentice-Hall, 1984.
50. Kevin J. Kerns, *Accurate and Stable Reduction of RLC Networks Using Split Congruence Transformations*, Ph.D. dissertation, University of Washington, 1996.
51. Alan V. Oppenheim and Ronald W. Schaffer, *Discrete-Time Signal Processing*, New Jersey: Prentice Hall, 1989.
52. Francis B. Hildebrand, *Introduction to Numerical Analysis*, New York: McGraw-Hill, 1974, pp. 457-462.
53. G. J. Lastman and N. K. Sinha, "Infinite Series for Logarithm of Matrix, Applied to Identification of Linear Continuous-Time Multivariable Systems from Discrete-Time Models", *Electronics Letters*, vol. 27, no. 16, pp. 1468-1469, August, 1991.
54. *ADM User's Guide - Analytic Circuit Simulation*, Avant! Corporation, Fremont, CA, December, 1996.

55. *MATLAB High-Performance Numeric Computation and Visualization Software: Reference Guide*, Natick, MA: The Mathworks, Inc., 1994.
56. Al Kelley and Ira Pohl, *A Book on C: Programming in C*, Redwood City, CA: Benjamin-Cummings Publishing Company, 1995.
57. *Sun FORTRAN Programmer's Guide*, Mountain View, CA: Sun Microsystems, Inc., 1988.
58. Jack J. Dongarra, James R. Bunch, Cleve B. Moler, and G. W. Stewart, *LINPACK: Users' Guide*, Philadelphia: Society for Industrial and Applied Mathematics, 1979.
59. *HSPICE*, Meta-Software, Inc., Version H95.1, Campbell, CA, 1995.
60. D. C. DeGroot and R. B. Marks, "Optimizing Time-Domain Network Analysis," *46th ARFTG Conference Digest*, pp. 19-28, November, 1995.



## Appendix A: Underlying Circuit Theory of Multiport Measurements

The initial requirement for measurement-based modeling is a complete set of multiport measurements. It was stated in Section 3.1 that although the modeling approach presented in this dissertation is general enough to operate on any linearly independent set of measurements, it was chosen to use time-domain scattering parameters. This appendix reviews the circuit theory behind making any set of multiport impulse-response measurements, and some more specific guidelines for making scattering measurements in particular.<sup>†</sup> To begin our discussion, we refer to equation (4.25), restated here for convenience with a minor change in nomenclature as

$$\mathbf{X}(s) = \mathbf{H}(s) \mathbf{U}(s) \quad (\text{A.1})$$

The functions  $\mathbf{H}(s)$ ,  $\mathbf{U}(s)$ , and  $\mathbf{X}(s)$  are the frequency-domain representations of the system impulse response, its input signals, and its output signals, respectively. Equation (A.1) corresponds to a system represented by its transfer function  $\mathbf{H}(s)$  excited by a set of input vectors  $\mathbf{U}(s)$ , and this excitation dictates a set of output vectors  $\mathbf{X}(s)$ . Since we wish to measure the impulse response  $\mathbf{H}(s)$ , we set the input matrix  $\mathbf{U}(s)$  to the identity matrix as in equation (4.26). Having done so, we are assured that the outputs  $\mathbf{X}(s)$  are equal to the system impulse response  $\mathbf{H}(s)$ .

---

<sup>†</sup> Two primary (and related) *application* issues which often serve to muddle the clean network and circuit theory behind multiport measurement are fixturing and de-embedding. Discussion of these two issues is deferred until Appendices B and C, respectively, which allows us to assume for the remainder of this appendix that any necessary fixturing into the network in question is completely transparent, such that no de-embedding whatsoever is necessary. We furthermore assume that ideal impulse or step input waveforms are available for excitation.

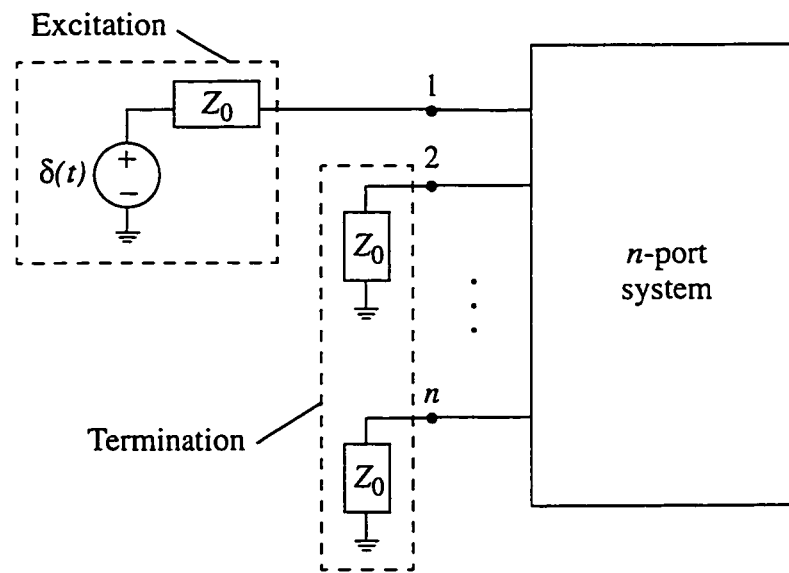
To understand the physical implications of the above discussion, we restate (A.1) in vector form as

$$\begin{bmatrix} \mathbf{x}_1(s) & \dots & \mathbf{x}_n(s) \end{bmatrix} = \mathbf{H}(s) \begin{bmatrix} \mathbf{e}_1 & \dots & \mathbf{e}_n \end{bmatrix} \quad (\text{A.2})$$

where  $\begin{bmatrix} \mathbf{e}_1 & \dots & \mathbf{e}_n \end{bmatrix}$  are the columns of the identity matrix. Each column of (A.2) represents a set of measurements taken under a specific termination scheme. For example, the first input vector  $\mathbf{e}_1$  represents a delta function introduced at port 1 while the inputs at all other ports are held to zero. The corresponding output vector  $\mathbf{x}_1(s)$  consists of the resultant outputs measured at each of the ports. If we are measuring the impedance parameters, for example,  $\mathbf{e}_1$  represents a delta function of current at port one, while all other port currents are held to zero. In the same way,  $\mathbf{x}_1(s)$  would represent the voltages measured at each of the ports in response to the input current vector  $\mathbf{e}_1$ . It should be apparent that the simplest way in theory to provide a delta function of current at a port is to connect an ideal current source to it, and that a port current is held to zero by open-circuiting that port. In a manner analogous to that described above for determination of  $\mathbf{x}_1(s)$ , each of the columns of  $\mathbf{X}(s)$ , and therefore  $\mathbf{H}(s)$ , is populated by implementing a termination scheme for each, and taking the associated set of measurements. If we are instead measuring admittance parameters, we apply termination and excitation schemes which are the duals of those outlined above. That is, a single port is excited by a delta function of *voltage*, and all other ports are *short-circuited*. One column of the admittance parameters may then be determined by measuring the injected *currents* at each of the ports.

In this work we are specifically measuring scattering parameters because of the physical difficulty of realizing ideal current and voltage sources or open and short circuits at high frequency. For this reason, the inputs to our system are incident voltage waves and its outputs are reflected voltage waves. An incident voltage wave delta function may be introduced at a port via an ideal voltage source with an output impedance equal to the characteristic impedance of the system,  $Z_0$  (or for that matter, the Norton equivalent of such a system). This more feasible in practice, since designing a high-frequency voltage

source with a controlled output impedance is far more reasonable than designing a voltage source with no output impedance (or infinite output impedance in the case of a current source). The input signal to a port may be set equal to zero by terminating that port with a load which is equal to the system characteristic impedance. Based on this capability to excite a port with a delta function and terminate a port such that it sees zero input, the measurements are made in the same fashion as was described above for impedance or admittance. Figure A.1 shows the setup for measuring  $\mathbf{x}_1(s)$ ,<sup>†</sup> the first column of the scattering impulse response, by exciting port 1 and terminating all others. The remaining col-



**Figure A.1** Excitation and termination scheme for measuring the first column of the transfer function of an  $n$ -port system.

umns are measured by analogous procedures. As an interesting sidelight, it may be observed that measurement of admittance parameters is a limiting case of the scattering measurement setup of Figure A.1 in which we have set  $Z_0 = 0$ . And to measure the

<sup>†</sup> Although the input delta function in Figure A.1 is in the time domain, whereas equation (A.2) is stated in the frequency domain, the two representations are mathematically equivalent.

impedance parameters, we replace the excitation shown in Figure A.1 with its Norton equivalent and allow  $Z_0 \rightarrow \infty$ .

As it turns out, the voltage source in conjunction with its output impedance, as seen in Figure A.1, is a very good description of the excitation circuitry of an ideal time domain reflectometer (TDR). The only difference is that the ideal TDR excitation signal is a voltage *step* rather than an *impulse*. Knowing this to be the case, the TDR may be used to directly measure a column of the time-domain scattering *step* response of a circuit by connecting the sampling head to each of the ports and applying an excitation signal. If a sampling head is simultaneously connected to each of the other ports, an entire column of the scattering response may be measured. As described above, a set of  $m$  termination/excitation schemes is necessary to measure all the columns of the  $m$ -port scattering step response. A quick check of linear network theory will validate that the impulse response is nothing more than the derivative of the step response.

The process of reviewing the idealized, theoretical approach to making a set of multiport measurements has likely raised as many questions as it has answered. The most important regards fixturing—how we will connect the TDR sampling heads to the DUT to make accurate measurements—and is the subject of Appendix B. Another important issue, how to deal with the fact that we have measured a step response when in fact we set out to measure the impulse response, is addressed in Appendix C.

## **Appendix B: Fixturing Issues in High-frequency Measurements**

The most basic requirement for effective measurement-based modeling is accurate measurement, and in the interconnect modeling arena, this implies high-frequency broadband measurements. Unfortunately, such measurements are not be obtained trivially; in fact, high-frequency measurement is considered to be as much an art as a science, to be learned and perfected under the watchful eye of an experienced mentor. In the more familiar arena of low-frequency electrical measurements, much of the electromagnetics present in the system can be approximated away, rendering the problem essentially scalar. At high frequency, however, a scalar conceptualization is altogether insufficient for an inherently vectorial problem, and a strong understanding of electromagnetic theory is therefore essential. For this reason, high-frequency measurements are seldom made by even a highly trained technician, but are more typically taken by an experienced high-frequency circuit designer.

At a very basic level, engineering represents the marriage of the two concepts of theory and application. Ideally application follows directly from complete understanding of the associated theory. However, the engineer soon discovers that in practice aseptic theoretical platitudes (such as those outlined in Appendix A) must often be significantly adjusted, and consequently blurred, in order to produce a workable application. High-frequency measurement is no exception to this principle, and typically the first question to arise is that of fixturing, or how to connect the measurement equipment to the device under test (DUT). A more general fixturing question, however, is how any measurements may be obtained without masking the behavior of the DUT.

Fixturing, then, is a question of how to effectively couple electrical energy into and out of the DUT. The problem is completely ignored by the representation in Figure A.1, which is meant to present a theoretically sound approach without confusing the theory with the application details. The fixturing difficulty typically lies in the connections from

the DUT to the normalizing impedances and the excitation source, since in interconnect modeling the size and shape of the DUT often do not correspond well to standard measurement configurations designed for more traditional measurement tasks. It is often necessary to design and build some sort of fixturing setup to couple the input and output signals without significant loss of energy in the process, since lost signal energy implies the loss of some information about the DUT.<sup>†</sup>

Since many measurement systems are designed to operate with a  $50\Omega$  reference impedance and coaxial transmission line connections, a high-bandwidth  $50\Omega$  coaxial line is typically used to bring the signal and load connections close to the DUT. At the end of the coaxial cable, a change is usually necessary in the form of the signal line to more closely match the form of the DUT. In some cases microwave probes may be used, and in other cases a coaxial line whose center conductor has been exposed may be more appropriate. These fixturing solutions are by far two of the most simple available, and in many cases a more complicated solution will be necessary for accurate measurement. It is at this point that the art of fixturing comes into play, since choosing the correct fixturing approach from the large number of possible solutions involves a tradeoff between simplicity of the design and accuracy of the subsequent measurements, and very few formal guidelines exist.

This aspect of fixturing is very important in that it is usually this change in the form of the signal lines that is responsible for the majority of the lost signal energy. This “lost” energy is actually reflected back toward the source for the most part; however, it is indeed lost in the sense that it is of no use in measuring the characteristics of the DUT—it has instead been used to characterize the fixturing setup! Since the high-frequency components of the input step are typically favored for reflection from the fixturing interface, the quality of the fixturing setup can determine whether or not the high-frequency characteristics of the DUT may be measured. Physically, the amount of reflection from the interface

<sup>†</sup> If the energy (and therefore information) loss is not catastrophic, deconvolution techniques may be applied to reclaim the partially lost information. Deconvolution discussed in Appendix C.

often depends on how drastic the changes are in form from the coax to the DUT. In most interconnect applications, this depends on how close electrically is the access to the ground plane, or some form of ground return. If this distance is electrically large in the frequency range over which the measurements are being taken, the signal transmission may occur across a very nonideal path from a transmission-line point of view. If this is the case, the electrical response of the fixture may mask or even dominate the response of the DUT, causing significant loss of information about the DUT. For well-designed fixturing solutions, these nonidealities can often be characterized by simple parasitic capacitance and inductance values, or even ignored; however, for more complex or difficult fixturing problems, the fixture itself must be rigorously characterized.<sup>†</sup>

Also of importance in building fixturing for high-frequency measurement is that the measurement setup closely recreate the situation in which the DUT is to be used in practice. Often this also refers to the presence and proximity of the ground plane or ground return, since in any circuitboard application the electrical characteristics of a device must be viewed as a function of its geometric relationship to the ground plane and other significant pieces of metal or other materials. If the various aspects of the ground return are not taken into account during fixturing and measurement, otherwise good measurements taken with great effort may be rendered inaccurate or useless.

In order to perform measurements in the real world, it is necessary to implement some level of fixturing to connect the measurement equipment to the DUT. However, due to the inherent nonideality of the real world, it is impossible to design a perfect fixture for any measurement application. Due to the large number of potential fixturing solutions and the apparent subjectivity involved in choosing one sufficient for the application at hand, fixturing and high-frequency measurement is said to be an art as well as a science. Because of the extreme complexity of the subject, this appendix achieves little more than to warn the would-be interconnect measurer of possible pitfalls, and point out the need to consult with experts. Fortunately a partial remedy exists for nonideal fixturing in the form

<sup>†</sup> In this sense the fixturing can be seen as creating an interconnect modeling problem of its own.

of software de-embedding, which is often useful for regaining some of the information lost due to nonideal fixturing solutions. Due to the complexity of that subject, it is not addressed in any detail in this dissertation, and the interested reader is instead referred to [30] as well as numerous publications referenced in [60].



## Appendix C: De-embedding High-Frequency Measurements

The reality of fixturing for measurement, as discussed in Appendix B, is that fixturing cannot be made ideal. This necessitates the de-embedding of the DUT response from the measured response, which consists of the combined response of the DUT and the fixture. However, even given *ideal* fixturing, the theoretical measurement approach outlined in Appendix A cannot be exactly implemented in practice. This is because a dirac delta function, which contains signal energy up to infinite frequencies, is physically unrealizable. The same is true for the unit step function, which is the integral of the delta function. When taking real-world measurements, the measurer must settle for an excitation signal which only approximates its ideal mathematical analog. Typically such a signal looks very similar to the ideal signal at low frequencies, but begins to deviate at higher frequencies. This is not necessarily problematic in light of the discussions in Sections 2.1.2, which states that a device generally needs to be modeled only over the frequency range under which it will be used. Since this fact precludes the necessity of exciting and measuring the DUT up to infinite frequencies, the problem of non-ideal excitation becomes more of an annoyance than an insurmountable obstacle. A problem will obviously arise, however, if a model is desired at frequencies above those contained in the excitation signal. If this is indeed the case, the only recourse is to invest in new measurement equipment.

An effective solution to the problem of nonideal excitation signals is to de-embed an approximate impulse response from the measured response. This operation is often referred to by the more specific system-theoretical terms “deconvolution” or “normalization”, although these concise terms belie the difficult numerical issues encountered in the implementation of such a scheme.<sup>†</sup> The rationale for deconvolution or normalization is

<sup>†</sup> Regardless, the terms from system theory will be used for the remainder of this discussion to avoid confusion with other meanings of the term “de-embedding”.

based on the fact that it is possible to compute a band-limited approximation to the impulse response, regardless of what excitation signal is used. The bandwidth of accuracy for such an approximate impulse response will always be less than or equal to that of the excitation signal. The theory behind this rationale is based on equation (4.25), stated here with a slight change in variables as

$$\mathbf{X}(s) = \mathbf{H}(s) \mathbf{U}(s) \quad (\text{C.1})$$

It may be seen that if  $\mathbf{U}(s)$  is a diagonal matrix, the entire relationship in (C.1) breaks down into a set of scalar relationships. That is,

$$x_{ij}(s) = h_{ij}(s) u_{jj}(s), \quad i, j = 1 \dots m \quad (\text{C.2})$$

for each of the  $m$  ports of the system. It is worth noting at this point that forcing  $\mathbf{U}(s)$  to be a diagonal matrix implies nothing more than implementation of the excitation/termination scheme outlined in Appendix A, in which one port is excited while the inputs to all other ports are set to zero. Computation of the frequency-domain transfer function, then, is simplified to inverting each of the scalar relations in (C.2), once the frequency-domain representations of the excitation and response measurements have been obtained. This process is stated mathematically as

$$h_{ij}(s) = \frac{x_{ij}(s)}{u_{jj}(s)}, \quad i, j = 1 \dots m \quad (\text{C.3})$$

Regardless of what approach is taken in the deconvolution of one measured signal from another, the deconvolution problem in general is known to be ill conditioned [39][40]. In the formulation of (C.3), the source of ill conditioning becomes apparent: at any frequencies  $s$  for which the energy present in the excitation signal  $u_{jj}(s)$  is zero, the transfer function  $h_{ij}(s)$  is undefined. Of course, in theoretical measurements of a linear system, the corresponding output  $x_{ij}(s)$  would also be zero, since a linear transfer function cannot con-

vert energy from one frequency to another. The ill-conditioning problem could then be theoretically solved by noting those frequencies at which the excitation  $u_{jj}(s)$  is zero, and artificially setting the corresponding  $h_{ij}(s)$  to zero or some other appropriate value.

However, when working with measured data there is generally noise present on all signals, in which case no signal is ever truly zero at any frequency. However, all signals will become very small, *i.e. below the noise floor*, above a certain frequency which is usually dependent on the measurement setup and equipment. So, the ill conditioning problem of deconvolution can be more accurately and realistically stated by saying that at frequencies where the excitation is below the noise floor, the transfer function as computed by (C.3) will be effectively meaningless.

A standard workaround for this problem is to apply a lowpass filter to the measured response  $x_{ij}(s)$  which effectively makes it significantly smaller than the excitation signal at frequencies above the cutoff frequency. This approach is stated in equation form as

$$h_{ij}(s) = \frac{x_{ij}(s)}{u_{jj}(s)} H_{LP}(s), \quad i, j = 1 \dots m \quad (\text{C.4})$$

where  $H_{LP}(s)$  is the transfer function of the lowpass filter whose cutoff frequency is below that of the excitation signal. More specifically, the transfer function of  $H_{LP}(s)$  is defined such that

$$H_{LP}(j\omega)x_{ij}(j\omega) \ll u_{jj}(j\omega) \quad \text{for } \omega > \omega_c \quad (\text{C.5})$$

where  $\omega_c$  is the frequency above which the input signal energy becomes insignificant. The approach used to obtain normalized data for the sake of this research work was quite similar to the one represented by (C.4), except that modifications are required if it is to be practically useful for sampled data.

To implement the modifications necessary to adapt the approach of (C.4) for use with sampled time-domain reflectometry (TDR) data, we note that TDR measures an inci-

dent voltage wave (the excitation)  $a_j[n]$  and a reflected voltage wave (the response)  $b_i[n]$ . The fast Fourier transform (FFT) [51] algorithm is used to convert the sampled-time data into sampled-frequency data, and an appropriate discrete filter is chosen. The normalization equation then becomes

$$h_{ij}[k] = \frac{FFT(diff(b_i[n]))}{FFT(diff(a_j[n]))} H_{LP}[k] \quad (C.6)$$

where  $b_i[n]$  and  $a_j[n]$  are the measured sampled-time waveforms, and  $h_{ij}[k]$  and  $H_{LP}[k]$  are computed sampled-frequency signals. Application of the difference function (the sampled version of analog differentiation) is necessary because the FFT algorithm computes an accurate representation of the spectrum of time-limited signals, whereas the step response is by nature not time-limited. Since the sampled-time representation of  $h_{ij}[k]$  is the desired result necessary for input into the modeling algorithm, the inverse FFT is applied to  $h_{ij}[k]$  as the final step of the normalization. This approach yields an impulse response approximation whose valid frequency range depends on the following parameters: the frequency content of the input step  $a_j[n]$ , the frequency response of the measurement system, and the sampling frequency of the acquired waveforms. The most restrictive of these three constraints determines the upper limit of the valid frequency range. For further information on normalization of measured time-domain data, the reader is referred to [39].

As was mentioned previously, nonideal fixturing can also be a source of error when approximating the impulse response of a device. The process of de-embedding the response of the DUT from the combined response of the DUT and the fixture may be performed in tandem with the normalization step described above. Due to the complexity of this step, it was not implemented as part of the system presented in Chapter 6. The system was instead designed to take as one of its inputs data which was previously de-embedded if necessary. Readers interested in the topic of removing the effects of nonideal excitation

and fixturing from measurement data are referred to [30] and the numerous publications referenced in [60].

## Vita

Steven D. Corey received the B.S.E.E. and M.S.E.E. degrees in 1991 and 1994, respectively, from the Department of Electrical Engineering at the University of Washington in Seattle. From 1994 to 1997 he worked part time at Tektronix, Inc., on interconnect modeling and measurement. Currently he is a candidate for the Ph.D. degree in electrical engineering at the University of Washington. His research interests include numerical methods, simulation and modeling of linear interconnect, and linear network reduction.

## Publications

- S. D. Corey and A. T. Yang, "Automatic Netlist Extraction for Measurement-Based Characterization of Off-Chip Interconnect," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, no. 10, pp. 1934-1940, October, 1997.
- S. D. Corey and A. T. Yang, "Automatic Netlist Extraction for Measurement-Based Characterization of Off-Chip Interconnect", *Proceedings of the IEEE/ACM International Conference on Computer Aided Design*, pp. 24-29, November, 1996.
- S. D. Corey, K. J. Kerns, and A. T. Yang, "Automatic Measurement-Based Characterization of Lossy MCM Line Using Lumped Elements," *Proceedings of the IEEE 5th Topical Meeting on Electrical Performance of Electronic Packaging*, pp. 144-146, November, 1996.
- S. D. Corey and A. T. Yang, "Interconnect Characterization Using Time Domain Reflectometry," *IEEE Transactions on Microwave Theory and Techniques*, vol. 43, no. 9, pp. 2151-2156, September, 1995.
- S. Corey and A. T. Yang, "Interconnect Characterization Using Time Domain Reflectometry," *Proceedings of the IEEE 3rd Topical Meeting on Electrical Performance of Electronic Packaging*, pp. 189-191, November, 1994.