Efficient Frequency-Domain Modeling and Circuit Simulation of Transmission Lines

L. Miguel Silveira Ibrahim M. Elfadel Jacob K. White

Research Laboratory of Electronics and the
Department of Electrical Engineering and Computer Science,
Massachusetts Institute of Technology
Cambridge, MA 02139.

Moni Chilukuri Kenneth S. Kundert

Cadence Design Systems, Inc.
555 River Oaks Parkway, MS 3B1
San Jose, CA 95134

Abstract

In this paper we describe an algorithm for efficient SPICE-level simulation of transmission lines with arbitrary scattering parameter descriptions. That is, the line can be represented in the form of a frequency-domain model or a table of measured frequency-domain data. Our approach initially uses a forced stable decade-by-decade ℓ_2 minimization approach to construct a sum of rational functions approximation, but the approximation has dozens of poles and zeros. This unnecessarily high-order model is then reduced using a guaranteed stable model order reduction scheme based on balanced realizations. Once the reduced-order model is derived, it can be combined with the transmission line's inherent delay to generate an impulse response. Finally, following what is now a standard approach, the impulse response can be efficiently incorporated in a circuit simulator using recursive convolution. An example of a transmission line with skin-effect is examined to both demonstrate the effectiveness of the approach and to show its generality.

1 Introduction

In the design of communication, high-speed digital, and microwave electronic systems, the behavior of transmission lines formed from packaging and interconnect can have an important impact on system performance. Stripline and microstrip printed circuit board traces, interchip connections on multi-chip modules, and coaxial cable connections all have nonidealities in their frequency response, many of which cannot be represented using a frequency-independent RLCG model. Since these nonidealities may or may not negatively impact signal integrity, depending on the driving and receiving circuitry, verification of system performance must involve circuit-level simulation that includes a transmission line model which faithfully represents the frequency-domain behavior.

The most straight-forward approach to including general frequency-domain transmission line models in a circuit simulator is to calculate the associated impulse response using an inverse fast Fourier transform [1]. Then, the response of the line at any given time can be determined by convolving the impulse response with an excitation waveform. Such an approach is too computationally inefficient for use in general circuit simulation, as it requires that at every simulator timestep, the impulse response be convolved with the entire computed excitation waveform.

An alternative approach is to approximate the frequency-domain representation with a rational function, in which case the associated convolution can be accelerated using a recursive algorithm [2, 3]. Very efficient circuit simulation programs which handle RLCG transmission lines have been developed using such an approach, where the rational function approximation was derived using Padé or moment-matching methods [2, 4, 5, 6]. In this paper we describe an algorithm for efficient SPICE-level simulation of transmission lines with arbitrary frequency-domain scattering parameter descriptions. The method is not as efficient as those intended specifically for RLCG transmission lines, but it is general enough to allow the use of any frequency-domain scattering parameter model or a table of measured data and it can be shown to have some important stability properties.

Our approach is a combination of several reasonably well-known techniques. First, a decade-by-decade ℓ_2 minimization approach is used to construct a collection of forced stable rational functions whose sum, after a final global ℓ_2 minimization, approximates the original frequency-domain data. This algorithm is described in Section 3, and it is shown that the resulting approximation, though extremely accurate, can have dozens of poles and zeros. Therefore, as described in Section 4, a second step is performed. The unnecessarily high-order model is reduced using a guaranteed stable model order reduction scheme based on balanced realizations [7, 8]. Once the reduced-order model is derived, it can be combined

with the transmission line's inherent delay to generate an impulse response. Then, following what is now a standard approach, the impulse response is efficiently incorporated into the circuit simulator SPICE using recursive convolution. In Section 5, we present results of the time-domain simulation of circuits containing a transmission line with skin-effect. The examples demonstrates both the efficiency of the approach and its generality, as there is no frequency-independent RLCG representation for transmission lines with skin effects.

2 Background

In general, a transmission line can be described in the frequency domain using scattering parameters, in which case

$$\begin{bmatrix} \boldsymbol{Y}_o(\boldsymbol{j}\omega)\boldsymbol{V}_a(\boldsymbol{j}\omega) + \boldsymbol{I}_a(\boldsymbol{j}\omega) \\ \boldsymbol{Y}_o(\boldsymbol{j}\omega)\boldsymbol{V}_b(\boldsymbol{j}\omega) + \boldsymbol{I}_b(\boldsymbol{j}\omega) \end{bmatrix} = \begin{bmatrix} 0 & \boldsymbol{S}_{12}(\boldsymbol{j}\omega) \\ \boldsymbol{S}_{12}(\boldsymbol{j}\omega) & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{Y}_o(\boldsymbol{j}\omega)\boldsymbol{V}_a(\boldsymbol{j}\omega) - \boldsymbol{I}_a(\boldsymbol{j}\omega) \\ \boldsymbol{Y}_o(\boldsymbol{j}\omega)\boldsymbol{V}_b(\boldsymbol{j}\omega) - \boldsymbol{I}_b(\boldsymbol{j}\omega) \end{bmatrix} \quad (1)$$

where $\boldsymbol{V}_a(\boldsymbol{j}\omega), \boldsymbol{I}_a(\boldsymbol{j}\omega)$ and $\boldsymbol{V}_b(\boldsymbol{j}\omega), \boldsymbol{I}_b(\boldsymbol{j}\omega)$ are the voltages and currents at terminals a and b of the transmission line, $\boldsymbol{Y}_o(\boldsymbol{j}\omega)$ is its characteristic admittance, and $\boldsymbol{S}_{12}(\boldsymbol{j}\omega)$ is the relation between the incident and reflected waves on opposite ends of the transmission line. Note, the nonstandard choice of $\boldsymbol{Y}_o(\boldsymbol{j}\omega)$ instead of $\boldsymbol{Z}_o(\boldsymbol{j}\omega) = 1/\boldsymbol{Y}_o(\boldsymbol{j}\omega)$ is that for a line with no shunt loss, $\boldsymbol{Z}_o(0) = \infty$, which may cause numerical difficulties in many situations. Any ideal delay resulting from propagation along the transmission line and which reflects itself on $\boldsymbol{S}_{12}(\boldsymbol{j}\omega)$ or $(\boldsymbol{Y}_o\boldsymbol{S}_{12})(\boldsymbol{j}\omega)$ is usually handled separately and cancelled from the above frequency dependent measurements or model before they are incorporated into the simulator. This is in general easily accomplished by multiplying by the associated exponentials [2, 5].

To incorporate such a general transmission line representation in a circuit simulator, it is necessary to compute the inverse Fourier transforms of $\boldsymbol{S}_{12}(\boldsymbol{j}\omega)$, $\boldsymbol{Y}_o(\boldsymbol{j}\omega)$, and $(\boldsymbol{Y}_o\boldsymbol{S}_{12})(\boldsymbol{j}\omega)$ so as to determine the impulse responses $s_{12}(t), y_o(t)$, and $(y_o s_{12})(t)$. Then (1) becomes

$$\begin{aligned} (y_o \star v_a)(t) + i_a(t) &= ((y_o s_{12}) \star v_b)(t - t_d) - (s_{12} \star i_b)(t - t_d) \\ (y_o \star v_b)(t) + i_b(t) &= ((y_o s_{12}) \star v_a)(t - t_d) - (s_{12} \star i_a)(t - t_d) \end{aligned} \quad (2)$$

where "$\star$" is used to denote convolution and t_d is the propagation delay which was extracted from the frequency dependent model and is now explicitly introduced into the time-domain equations.

As mentioned in the introduction, if $s_{12}(t), y_o(t)$ and $(y_o s_{12})(t)$ are derived by applying the inverse FFT to $\boldsymbol{S}_{12}(\boldsymbol{j}\omega)$, $\boldsymbol{Y}_o(\boldsymbol{j}\omega)$, and $(\boldsymbol{Y}_o\boldsymbol{S}_{12})(\boldsymbol{j}\omega)$ respectively, then the convolutions will be expensive to compute. If, however, $\boldsymbol{S}_{12}(\boldsymbol{j}\omega)$, $\boldsymbol{Y}_o(\boldsymbol{j}\omega)$, and $(\boldsymbol{Y}_o S_{12})(\boldsymbol{j}\omega)$ can be represented

using rational function approximations, then the convolution can be performed much faster, and deriving this rational functions is the subject of the subsequent sections.

3 Section-by-Section Approximations

The most commonly used approaches to fitting rational functions to frequency domain data are the Padé or moment-matching methods. These methods compute the coefficients of a rational function by matching that approximation to the value of the system function and its derivatives around $s = 0$.

In this section we describe a sectioned approach to the problem of approximating the transfer function of a system by a forced stable rational function. With this approach, we replace the problem of directly computing a low order rational function that is an accurate approximation over a wide frequency range with that of repeatedly computing local approximations over narrower ranges. These local approximations can then be summed to create an accurate approximation over the wide frequency range. This approach avoids, or at least minimizes, the ill-conditioning of the global approximation problem. This approach is similar in spirit to a generalization of the moment methods which is based upon multiple expansions around other values of s to gather more global information [9].

We will start in Section 3.1 by describing a standard constrained $\boldsymbol{\ell_2}$ minimization approach. The shortcomings of such an approach will be made clear, and in order to avoid these difficulties we describe, in Section 3.2, a section-by-section algorithm which is based on a local constrained $\boldsymbol{\ell_2}$ minimization procedure. Finally, in Section 3.3 we will present some results that show that this section-by-section algorithm can generate rational functions which match data very accurately.

3.1 Computing Global Approximants by Weighted $\boldsymbol{\ell_2}$ Minimization

One approach to generating a rational function which best matches a frequency response $\boldsymbol{F}(s)$ specified at a set of frequencies $\{s_1, s_2, \cdots, s_m\}$, is to set up and solve, as accurately as possible, the following set of equations:

$$\boldsymbol{H}(s_j) = \boldsymbol{F}(s_j) \qquad j = 1, 2, \cdots, m \tag{3}$$

where

$$\boldsymbol{H}(s) = \frac{\boldsymbol{U}(s)}{\boldsymbol{V}(s)} = \frac{u_p s^p + \cdots + u_1 s + u_0}{s^q + \cdots + v_1 s + v_0} \tag{4}$$

is the low-order approximation.

Typically, the system in (3) will be over-determined as the number of frequency points will exceed the number of unknown coefficients in the approximation (4), that is $m > p+q+1$. In this case there will generally be no exact solution, but the approximation error can be minimized in some appropriate norm. If the 2-norm of the error is minimized, then the coefficients of the polynomials $\boldsymbol{U}(s)$ and $\boldsymbol{V}(s)$ are chosen such that

$$\sqrt{|\boldsymbol{H}(s_1) - \boldsymbol{F}(s_1)|^2 + \cdots + |\boldsymbol{H}(s_m) - \boldsymbol{F}(s_m)|^2} = \|\boldsymbol{H}(\boldsymbol{s}) - \boldsymbol{F}(\boldsymbol{s})\|_2 = \left\| \frac{\boldsymbol{U}(\boldsymbol{s})}{\boldsymbol{V}(\boldsymbol{s})} - \boldsymbol{F}(\boldsymbol{s}) \right\|_2 \quad (5)$$

is minimized. However, this is a nonlinear optimization problem whose solution is difficult to compute. Instead, the problem can be made linear by weighting the 2-norm by $\boldsymbol{V}(s)$. Then, the minimization problem becomes

$$\min_{\boldsymbol{U},\boldsymbol{V}} \|\boldsymbol{U}(\boldsymbol{s}) - \boldsymbol{V}(\boldsymbol{s})\boldsymbol{F}(\boldsymbol{s})\|_2 \quad (6)$$

Note that the solution to (6) is not in general the same as the solution of (5), but is instead a weighted $\boldsymbol{\ell_2}$ minimization.

The above $\boldsymbol{\ell_2}$ minimizing solution of the over-determined system minimizes the *global* error in a weighted $\boldsymbol{\ell_2}$ sense instead of being very accurate at $s = 0$ or at any particular expansion point. However, to guarantee that the steady-state will be accurately computed when the rational function is used as a model in a circuit simulator, it is essential to constrain the minimization so that $\boldsymbol{U}(0) = \boldsymbol{V}(0)\boldsymbol{F}(0)$. Similar constraints can be imposed at high frequencies if necessary. The resulting constrained minimization can then be summarized as

$$\begin{cases} \dfrac{\boldsymbol{U}(0)}{\boldsymbol{V}(0)} = \boldsymbol{F}(0) \\ \displaystyle\min_{\boldsymbol{U},\boldsymbol{V}} \|\boldsymbol{U}(\boldsymbol{s}) - \boldsymbol{V}(\boldsymbol{s})\boldsymbol{F}(\boldsymbol{s})\|_2 \\ \displaystyle\lim_{s\to\infty} \frac{\boldsymbol{U}(s)}{\boldsymbol{V}(s)} = \lim_{s\to\infty} \boldsymbol{F}(s). \end{cases} \quad (7)$$

The global minimization in (7) has two major drawbacks, namely the large dynamic range of the numbers involved and the over-emphasizing of high-frequency errors. The dynamic range of the number in the equation presents a difficulty especially in the case when the natural frequencies of the problem span a wide range, as is usual in transmission line problems. In that situation, (7) can easily lead to extremely ill-conditioned matrix problems. To see this, consider the structure of the matrix one obtains from the minimization portion

of (7), which can be written as:

$$\begin{bmatrix} s_1^p & \cdots & s_1 & 1 & -F_1 s_1^{q-1} & \cdots & -F_1 s_1 & -F_1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\ s_j^p & \cdots & s_j & 1 & -F_j s_j^{q-1} & \cdots & -F_j s_j & -F_j \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\ s_m^p & \cdots & s_m & 1 & -F_m s_m^{q-1} & \cdots & -F_m s_m & -F_m \end{bmatrix} \begin{bmatrix} u_p \\ \vdots \\ u_1 \\ u_0 \\ v_{q-1} \\ \vdots \\ v_1 \\ v_0 \end{bmatrix} = \begin{bmatrix} F_1 s_1^q \\ \vdots \\ F_j s_j^q \\ \vdots \\ F_m s_m^q \end{bmatrix} \tag{8}$$

Each row of this matrix corresponds to computing $\boldsymbol{U}(s_j) - \boldsymbol{V}(s_j)\boldsymbol{F}(s_j)$ at some frequency value s_j. The matrix is therefore a transposed Vandermonde-like matrix in the sense that the entries along each row are simple powers of the corresponding frequency value. If the span of frequencies being considered is large, then the magnitude of the entries on some of those rows will be much larger than those in rows corresponding to low frequency values.

Even if the conditioning of the matrix in (8) is tolerable, the resulting solution will be skewed to minimizing high-frequency errors. To understand this problem, consider the case $p = q - 1$, and recall that an $\boldsymbol{\ell_2}$ minimization attempts to minimize the sums of the squares of the error at each point, that is:

$$e = \sum_{i=1}^{m} \sqrt{e_1^2 + e_2^2 + \cdots + e_m^2}, \tag{9}$$

where

$$\begin{aligned} e_j &= \|\boldsymbol{U}(\boldsymbol{s}) - \boldsymbol{V}(\boldsymbol{s})\boldsymbol{F}(\boldsymbol{s})\|_2 = \\ &= \left| u_{q-1} s_j^{q-1} + \cdots + u_1 s_j + u_0 - s_j^q F_j - \cdots + v_1 s_j F_j - v_0 F_j \right| = \\ &= \left| -s_j^q F_j + (u_{q-1} - v_{q-1} F_j)\, s_j^{q-1} + \cdots + (u_1 - v_1 F_j)\, s_j + (u_0 - v_0 F_j) \right| \end{aligned} \tag{10}$$

is the error for the j^{th} equation, corresponding to the frequency value s_j.

From Eqn. (11) one can immediately see that the sensitivity of the error for the j^{th} equation, e_j, with respect to any coefficient is a polynomial in s_j. Hence, the contribution of an error at s_j to the global cost function is a polynomial in s_j. This implies that for a high frequency value s_j, small changes in the values of the coefficients translate into large errors and e_j will be large. Therefore, minimizing the total error requires that the error components e_j corresponding to higher frequencies be carefully minimized, while those corresponding to lower frequencies, which have less impact on the global error, will not deserve so much attention. Though it is possible to introduce a weighting function that minimizes the high-frequency predominance effect, the precise weighting is difficult to determine a-priori.

3.2 Computing Section-by-Section Approximants

In order to avoid the numerical ill-conditioning and the uneven frequency weighting mentioned above, it is desirable to limit the frequency range for the $\boldsymbol{\ell}_2$ minimization. Computing a low-order local approximation has the added advantage that the orders of the polynomials in the rational function approximation may be chosen small without compromising the accuracy of the approximation for a small frequency range. Moreover, if unstable poles are obtained from the local minimization procedure it is likely that using some simple heuristic, such as simply discarding the unstable poles and associated residues, will not have a profound effect on the accuracy over the small range of frequencies involved. In other words, it is possible that a very low-order approximation is accurate enough to capture the local behavior of $\boldsymbol{F}(s)$ without instability, numerical or otherwise, playing a significant role.

The idea of computing local approximations leads to a **sectioning** algorithm in which only accurate local approximations are computed. The remaining problem is how to incorporate all the local information resulting from the various approximations into a global approximant.

Our proposed solution is to perform the local approximations in a repeated fashion using a constrained weighted local $\boldsymbol{\ell}_2$ minimization procedure. Initially, the frequency range of interest, $\boldsymbol{\Omega} = [\omega_{min}, \omega_{max}]$, is partitioned into small sections, $\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, \cdots, \boldsymbol{\Omega}_M$, such that $\boldsymbol{\Omega} = \bigcup_{i=1}^{M} \boldsymbol{\Omega}_i$, where each $\boldsymbol{\Omega}_i = [\omega_{i1}, \omega_{im_i}]$ is a decade or two long. Then, starting with the lowest frequency range $\boldsymbol{\Omega}_1$, with frequency values $\boldsymbol{F}(\omega_{11}), \boldsymbol{F}(\omega_{12}), \cdots, \boldsymbol{F}(\omega_{1m_1})$, a constrained $\boldsymbol{\ell}_2$ minimization is performed and a local approximant is computed. Once the first local approximation, $\boldsymbol{L}_1(s)$, is obtained in the form of a collection of poles and their corresponding residues, it is examined and the stable poles are retained while the unstable ones are discarded, leaving us with a *forced stable* approximation, $\boldsymbol{H}_1(s)$. Since the fit at the lower frequencies has captured the low frequency dynamics, $\boldsymbol{F}(s) - \boldsymbol{H}_1(s)$ will contain primarily the higher-frequency error information and is then approximated. To this end, frequency values in the second section $\boldsymbol{\Omega}_2$ are approximated. The value of $\boldsymbol{H}_1(s)$ at every point $\omega_{21}, \omega_{22}, \cdots, \omega_{2m_2}$ is computed, subtracted from the corresponding values $\boldsymbol{F}(\omega_{21}), \boldsymbol{F}(\omega_{22}), \cdots, \boldsymbol{F}(\omega_{2m_2})$ and the resulting data is again fit using a constrained weighted $\boldsymbol{\ell}_2$ minimization. This results in a new local approximant $\boldsymbol{L}_2(s)$, from which a stable approximation, $\boldsymbol{H}_2(s)$ can be obtained. $\boldsymbol{H}_2(s)$ is then a new approximant to $\boldsymbol{F}(s) - \boldsymbol{H}_1(s)$ on $\boldsymbol{\Omega}_1 \cup \boldsymbol{\Omega}_2$, and therefore $\boldsymbol{F}(s) \approx \boldsymbol{H}_1(s) + \boldsymbol{H}_2(s)$ on that frequency interval. The procedure is repeated until data in the last frequency section, $\boldsymbol{\Omega}_M$, is approximated. A simplified form of this sectioning algorithm is shown in pseudo-code form in Algorithm 3.1, and diagrammatically in Figure 1.

Algorithm 3.1 (Section-by-Section Approximations)

sectioned($\omega_{min}, \omega_{max}, \boldsymbol{F}$)

{

partition the frequency range into sections $\boldsymbol{\Omega}_1, \cdots, \boldsymbol{\Omega}_M$ with associated frequencies $\{\omega_{i1}, \cdots, \omega_{im_i}\}$, $i = 1, \cdots, M$, and function values $\{\boldsymbol{F}(\omega_{i1}), \cdots, \boldsymbol{F}(\omega_{im_i})\}$

for ($k = 1$; $k <= M$; $k++$) {

if (k > 1) {

subtract previous approximants from exact data:

$$\boldsymbol{F}_k(s_{kj}) = \boldsymbol{F}(s_{kj}) - \sum_{l=1}^{k-1} \boldsymbol{H}_l(s_{kj}) = \boldsymbol{F}(s_{kj}) - \boldsymbol{H}(s_{kj}),$$

$j = 1, \cdots, m_k, \;\; s_{kj} = \boldsymbol{j}\omega_{kj}$

} else {

$\boldsymbol{F}_1(s_{1,j}) = \boldsymbol{F}(s_{1,j})$

}

compute local approximant at the k-th section, $\boldsymbol{L}_k(s)$ using the corrected data $\boldsymbol{F}_k(s_{i,j})$

examine the approximation and keep the stable poles and residues of $\boldsymbol{L}_k(s)$ in $\boldsymbol{H}_k(s)$

add the new stable approximation to the current global approximant $\boldsymbol{H}(s) = \boldsymbol{H}(s) + \boldsymbol{H}_k(s)$

}

while keeping the locally computed dynamics, perform a final global constrained $\boldsymbol{\ell}_2$ minimization over the whole frequency range to recompute the residues

}

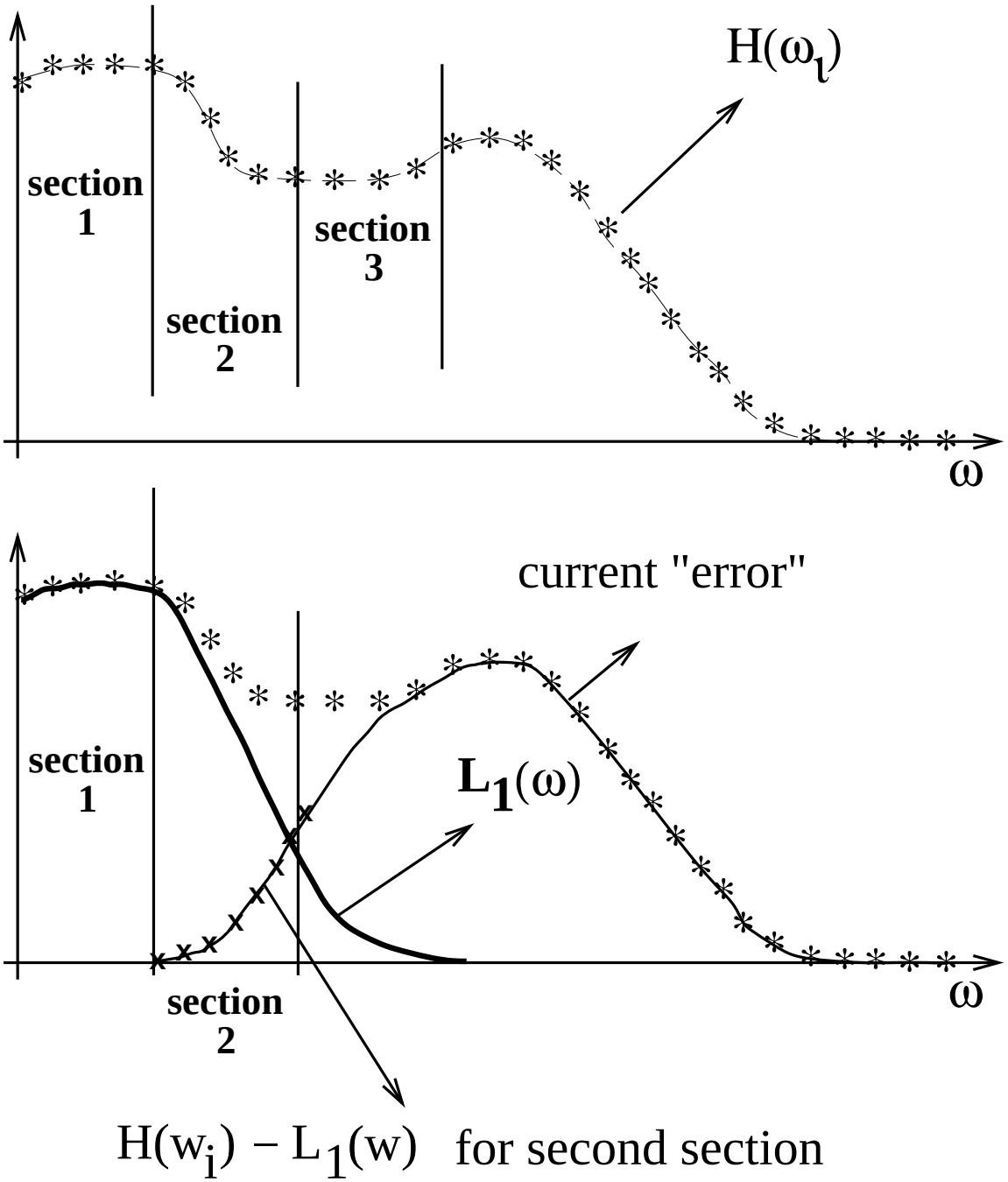

Figure 1: Applying the sectioning algorithm to measured or tabulated frequency data. The example illustrates the sequence of operations that are performed to compute a local approximation, add it to the current global approximation and recompute the current error function.

When the procedure terminates, the result is a forced stable global approximation which consists of all the stable poles and their corresponding residues obtained from the sequence of local minimizations. We should point out that the sectioning algorithm is aimed at computing approximations which match successively higher frequency ranges. However, while subtracting the already computed approximations from the exact data, some erroneous dynamics may be introduced at low frequency. To eliminate the associated errors, a final constrained global $\boldsymbol{\ell_2}$ minimization is performed in which the computed poles are used to recalculate their residues in order to match the exact data points. This final step does not suffer from the numerical problems mentioned in Section 3.1 regarding the global $\boldsymbol{\ell_2}$ minimization. In fact, the matrix one obtains in this case is better behaved because its (i, j) entries are of the form $(s_i - p_j)^{-1}$.

The algorithm just described reliably obtains a stable collection of pole-residue pairs which form an accurate approximation to $\boldsymbol{F}(s)$. Unfortunately, since $\boldsymbol{H}(s)$ is represented as a sum of local approximations, the approach introduces redundancies resulting in many more poles than necessary. With such a large number of terms, even fast recursive convolution may prove to be inefficient. However it is possible to further reduce the order of the approximation using robust model order reduction techniques, which are described in section 4.

3.3 Section-by-Section Approximant: numerical example

In order to test the accuracy of the approximant obtained with our section by section algorithm, consider the example of a transmission line where skin effects are significant, as shown in Figures 2 and 3. The approximations to $\boldsymbol{S}_{12}(\boldsymbol{j\omega})$ and $\boldsymbol{Y}_o(\boldsymbol{j\omega})$, after removing the ideal delay, have respectively 21 and 24 poles. In Figures 2 and 3, we compare the magnitude plots of the transfer functions of, respectively, $\boldsymbol{S}_{12}(\boldsymbol{j\omega})$ and $\boldsymbol{Y}_o(\boldsymbol{j\omega})$ with the transmission line data points.

As one can see, the match is almost perfect, and the error is smaller than 0.5%. Moreover the low-frequency error is nearly zero.

4 Model-Order Reduction by Truncated Balanced Realization

The frequency-domain data fitting method described in the previous section resulted in a stable transfer function $\boldsymbol{H}(s)$ with a large number of poles. Incorporating such a model (or equivalently its impulse response) directly in a circuit simulator will be computationally expensive. Instead, the model is reduced using an algorithm with three main steps. First,

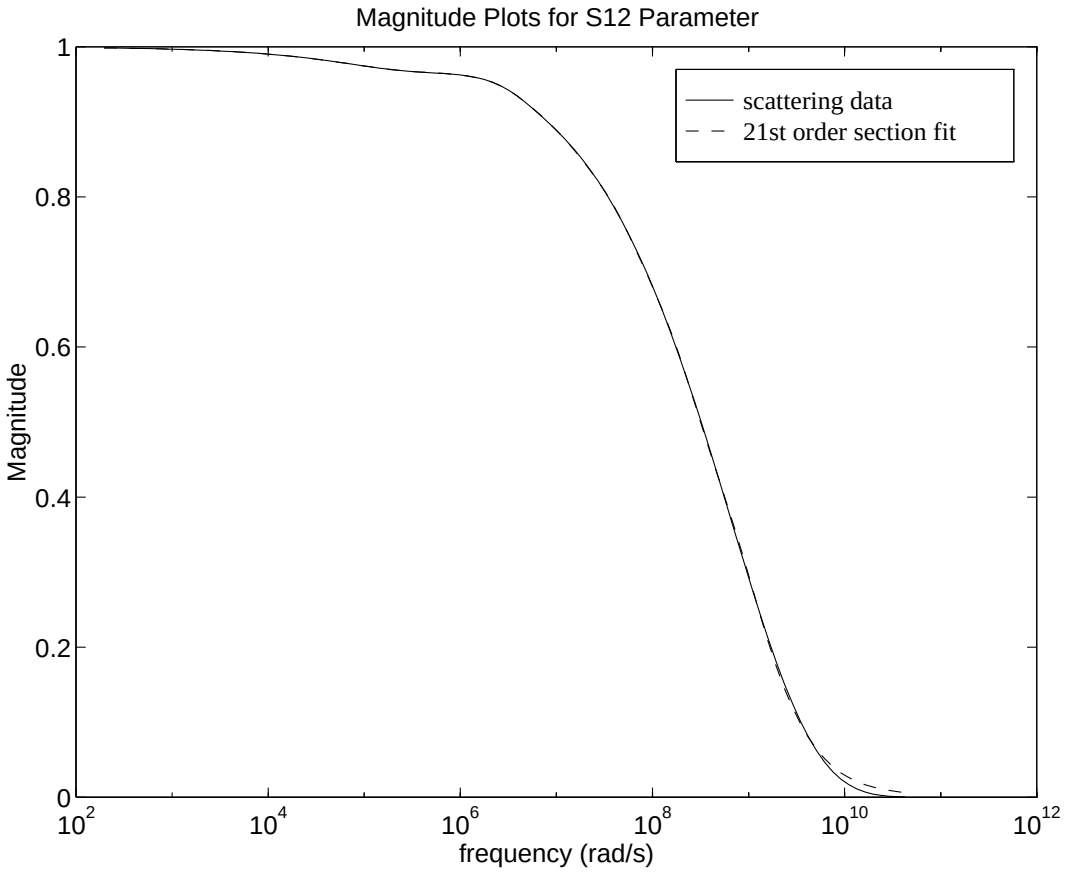

Figure 2: Accuracy of the section-by-section fit for the magnitude of the $\boldsymbol{S}_{12}$ transfer function with respect to the transmission line data points. The two curves are almost indistinguishable.

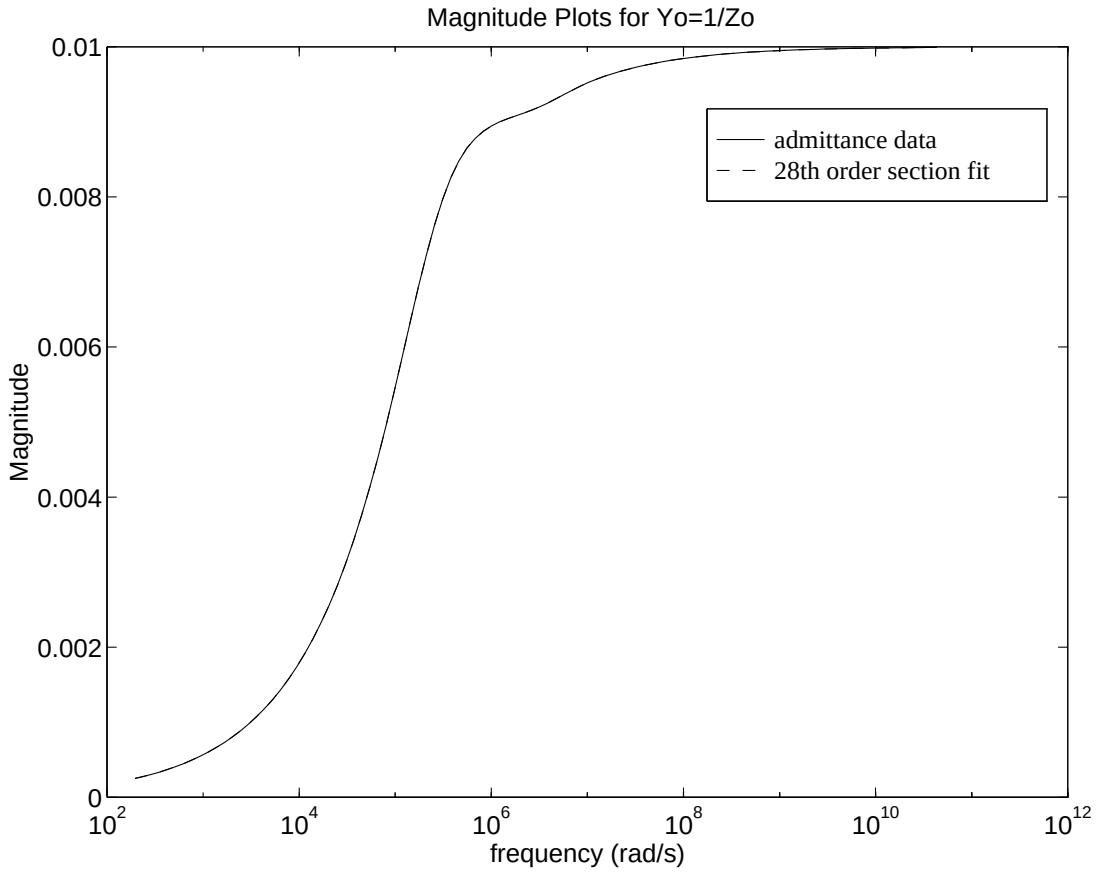

Figure 3: Accuracy of the section-by-section fit for the magnitude of the $\boldsymbol{Y}_o$ transfer function with respect to the transmission line data points. The two curves are almost indistinguishable.

the model is converted to a well-conditioned and robust state-space realization. Second, a state-space transformation is used to *balance* the state-space realization. Third, the balanced realization is truncated. Using this type of balanced realization approach has a key advantage: the resulting reduced $\boldsymbol{H}_k(s)$ is *guaranteed* stable if $\boldsymbol{H}(s)$ is stable.

4.1 State-Space Realization

To reduce the order of the transmission line model derived in the previous section, first we consider its state-space representation

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u, \qquad && \boldsymbol{x} \in \mathbb{R}^n,\ u \in \mathbb{R},\ \boldsymbol{A} \in \mathbb{R}^{n\times n},\ \boldsymbol{B} \in \mathbb{R}^n \\ y &= \boldsymbol{C}\boldsymbol{x}, && y \in \mathbb{R}, \boldsymbol{C} \in \mathbb{R}^n \end{aligned} \tag{11}$$

such that $\boldsymbol{H}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}$.

Converting $\boldsymbol{H}(s)$ in pole-residue form to state-space form is a standard problem [10], and it is tempting to use one of the common techniques (canonical controllability realization, canonical observability realization, etc.) to find the matrices $\boldsymbol{A}, \boldsymbol{B}$, and $\boldsymbol{C}$. However, these approaches can result in a system matrix $\boldsymbol{A}$ which is poorly scaled and therefore unsuitable for computations.

Instead, when all the poles are simple and real, the matrix $\boldsymbol{A}$ can be chosen equal to a diagonal matrix having the real poles as diagonal coefficients [10]. The control and observation matrices $\boldsymbol{B}$ and $\boldsymbol{C}$ can then be chosen based on the residues of the poles. More explicitly, given

$$\boldsymbol{H}(s) = \sum_{k=1}^{n} \frac{r_k}{s - p_k} \tag{12}$$

where all the poles are negative reals and all the residues are real,

$$\begin{aligned} \boldsymbol{A} &= diag(p_1, \ldots, p_n) \\ \boldsymbol{B} &= (\sqrt{|r_1|}, \ldots, \sqrt{|r_n|})^T \\ \boldsymbol{C} &= (\text{sign}(r_1)\sqrt{|r_1|}, \ldots, \text{sign}(r_n)\sqrt{|r_n|}) \end{aligned}$$

When $\boldsymbol{H}(s)$ has pairs of complex conjugate poles, a block diagonal matrix $\boldsymbol{A}$ can be constructed where the blocks are all 2×2 and correspond to pairing the complex conjugate poles in state-space realizations of order 2. It is also possible to find suitable state-space realizations when some of the poles are repeated. For transmission line examples there are only real, simple poles, and therefore the purely diagonal realization can be used.

4.2 Balanced Realizations

Once the state-space representation is adopted, it has to be internally balanced [7, 11]. That is, given $\boldsymbol{H}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}$, the choice of the triplet $[\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]$ is not unique. Indeed, a linear coordinate transformation $\tilde{\boldsymbol{x}} = \boldsymbol{T}\boldsymbol{x}$ modifies the triplet $[\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]$ to $[\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}]$ *without* modifying $\boldsymbol{H}(s)$.

For the specific purpose of extracting *stable* reduced-order models from the state-space representation, it is desirable that the new triplet $[\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}]$ be in a form that allows such an extraction using some simple operation on the new state $\tilde{\boldsymbol{x}} = \boldsymbol{T}\boldsymbol{x}$. The easiest conceivable such operation would be simple state truncation. Moore has shown [7] that such a transformation exists and he called the corresponding triplet $[\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}]$ a *balanced realization* of the transfer function $\boldsymbol{H}(s)$. The word "balanced" refers to the fact that the controllability and observability gramians of the triplet $[\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}]$ are both equal to the *same* diagonal matrix. The balancing transformation $\boldsymbol{T}$ can be computed explicitly for any triplet $[\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]$, and in particular for the diagonal realization that we have proposed in the previous paragraph. The numerical cost of such a computation is that of solving two matrix Lyapunov equations to obtain the controllability and observability gramians and one symmetric eigenvalue problem to diagonalize their product.

4.3 Truncated Realization

The triplet $[\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}}, \tilde{\boldsymbol{C}}]$ obtained by applying the balancing transformation $\boldsymbol{T}$ to the triplet $[\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]$ has the property that simple reordering and truncation of the state vector $\tilde{\boldsymbol{x}}$ with the corresponding reordering of the system matrices *necessarily* produce *stable* reduced-order models at any desirable order. Let k be this order, and let $[\tilde{\boldsymbol{A}}_k, \tilde{\boldsymbol{B}}_k, \tilde{\boldsymbol{C}}_k]$ be the reduced-order model with a transfer function $\boldsymbol{H}_k(s)$. It can then be shown [7, 8] that the error transfer function $\boldsymbol{E}_k(s) = \boldsymbol{H}(s) - \boldsymbol{H}_k(s)$ has an L_∞ norm that consistently decreases to zero as k is increased to n, the order of the original model. This L_∞ norm corresponds to the peak of the magnitude Bode plot of $\boldsymbol{E}_k(j\omega)$. Note that Padé approximation methods [4] do not enjoy such an error reduction property, and there is in fact ample experimental evidence that the Padé methods produce unstable reduced-order models.

Truncating the balanced-realization has the same flavor but is radically different from a spectral truncation, i.e., one that is based on neglecting the "fast" modes. Indeed, the latter method looks only at the state matrix $\boldsymbol{A}$ without taking into account how controllable or observable the neglected modes are. This is exactly what is achieved by truncating the balanced realization where the controllability and observability properties of the modes are taken into account through the gramian matrices.

4.4 Time-Domain Constraints

Judging the validity of the reduced-order model depends not only on meeting the L_∞ error criterion mentioned above but also on meeting the goals of the circuit simulation task for which this reduced model is used. Typically, in circuit simulations, it is essential that the reduced model match the original transfer function at $s = 0$ so that the steady-state behavior of both the reduced and full models are identical. Moreover, when the objective is to have a good match between the time-domain responses of the two models, it is essential that their transfer functions match at $s = \infty$ so that their initial behavior is the same [12]. To ensure the recovery of the steady-state behavior a final least-squares/collocation technique is used to match the reduced-order model with the full model at zero frequency [13].

4.5 Truncated Balanced Realization: numerical example

In order to test the accuracy of the order reduction algorithm, the method was applied to the transfer function obtained using the section-by-section procedure (see Section 3.3). It was found that reduced models with seven poles each were sufficient to approximate the full transfer functions of both $\boldsymbol{S}_{12}(j\omega)$ and $\boldsymbol{Y}_o(j\omega)$. In Figures 4 and 5, the magnitude plots of the reduced transfer functions of $\boldsymbol{S}_{12}(j\omega)$ and $\boldsymbol{Y}_o(j\omega)$ are compared with the transmission line data points. As is clear from the figures, the match is very accurate and the error is within 1%.

However in contrast to the section-by-section approximation the low-frequency error is more noticeable. In Figures 6 and 7, the magnitude plots of the frequency dependent fitting errors from the section-by-section approximation and the reduced-order model are shown for $\boldsymbol{S}_{12}(j\omega)$ and $\boldsymbol{Y}_o(j\omega)$, respectively.

5 Experimental Results

In this Section, we present results from an implementation of our algorithm for efficient time-domain simulation of transmission lines with arbitrary scattering parameter descriptions. The implementation is based on a modified version of SPICE3 [14], and uses a combination of sectioning, reduced-order modeling, and fast recursive convolution. We first show that the reduced-order model produces nearly the same time-domain waveforms as the more complete sectioning based model, but with many fewer poles. For completeness we will also apply a more traditional FFT-based method to this problem and compare the results in terms of accuracy and computational cost. Second, we show an example with realistic transistor drivers and receivers, to demonstrate the ability of the method to simulate complete

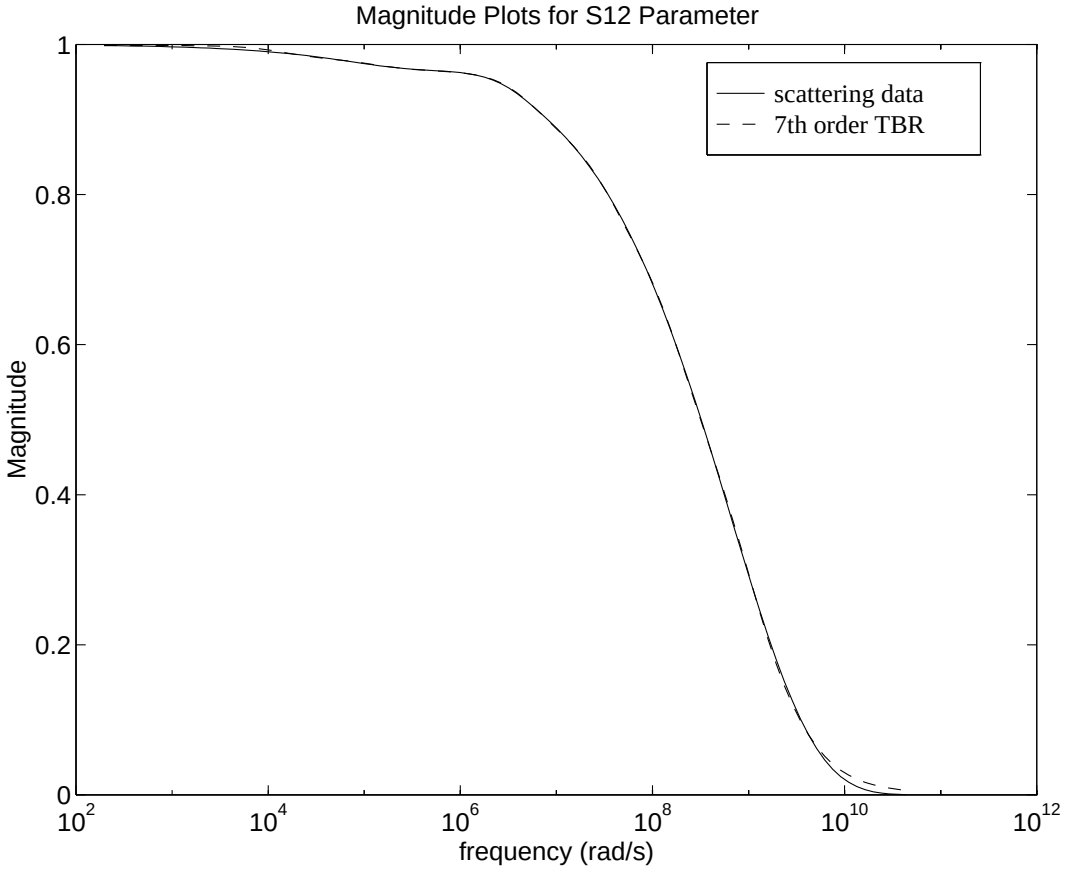

Figure 4: Accuracy of the reduced-order model fit for the magnitude of the $\boldsymbol{S}_{12}$ transfer function with respect to the transmission line data points.

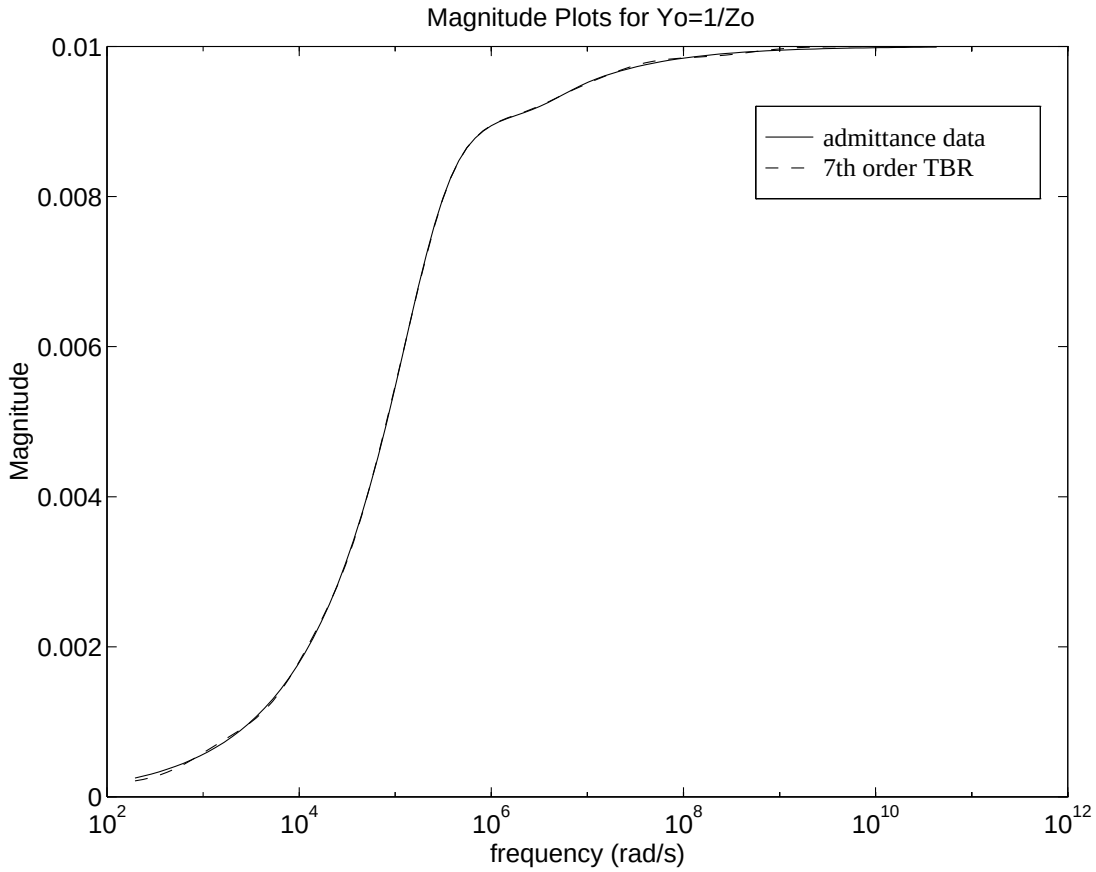

Figure 5: Accuracy of the reduced-order model fit for the magnitude of the $\boldsymbol{Y}_o$ transfer function with respect to the transmission line data points.

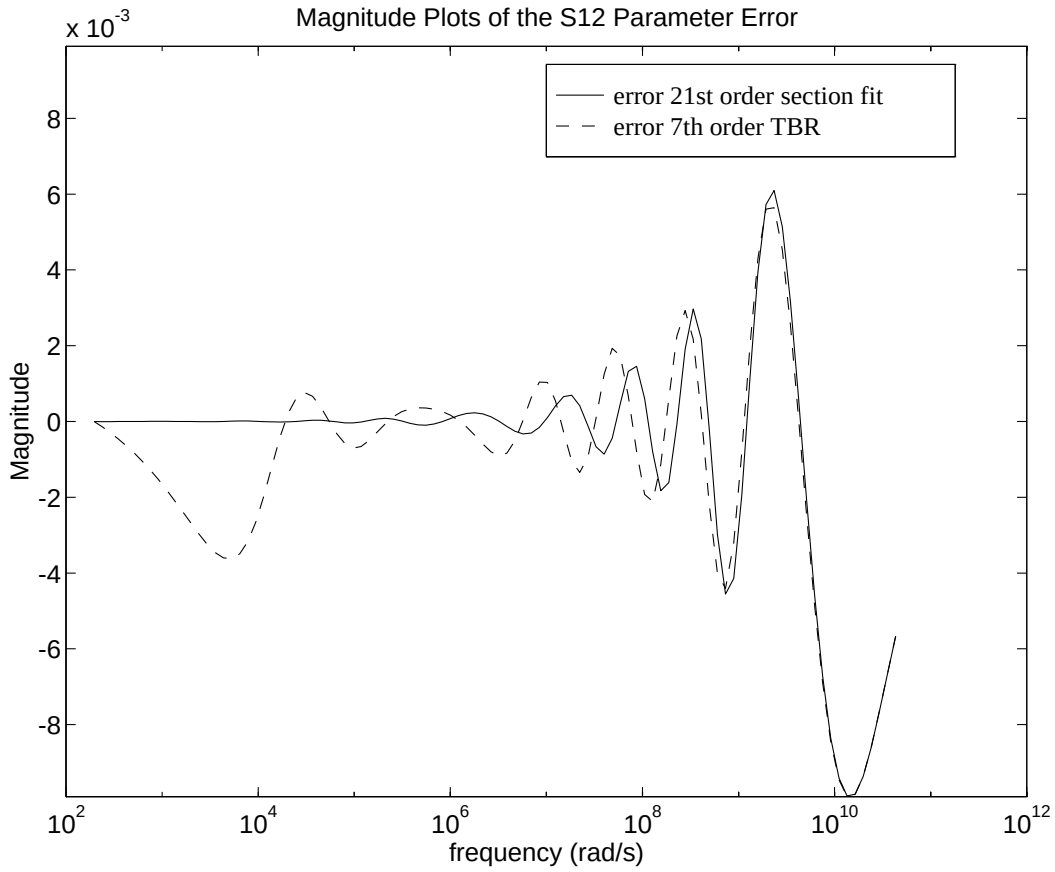

Figure 6: Magnitude plots of the errors with respect to the transmission line data points of the section-by-section approximant and the reduced-order transfer function for the $\boldsymbol{S}_{12}$ parameter.

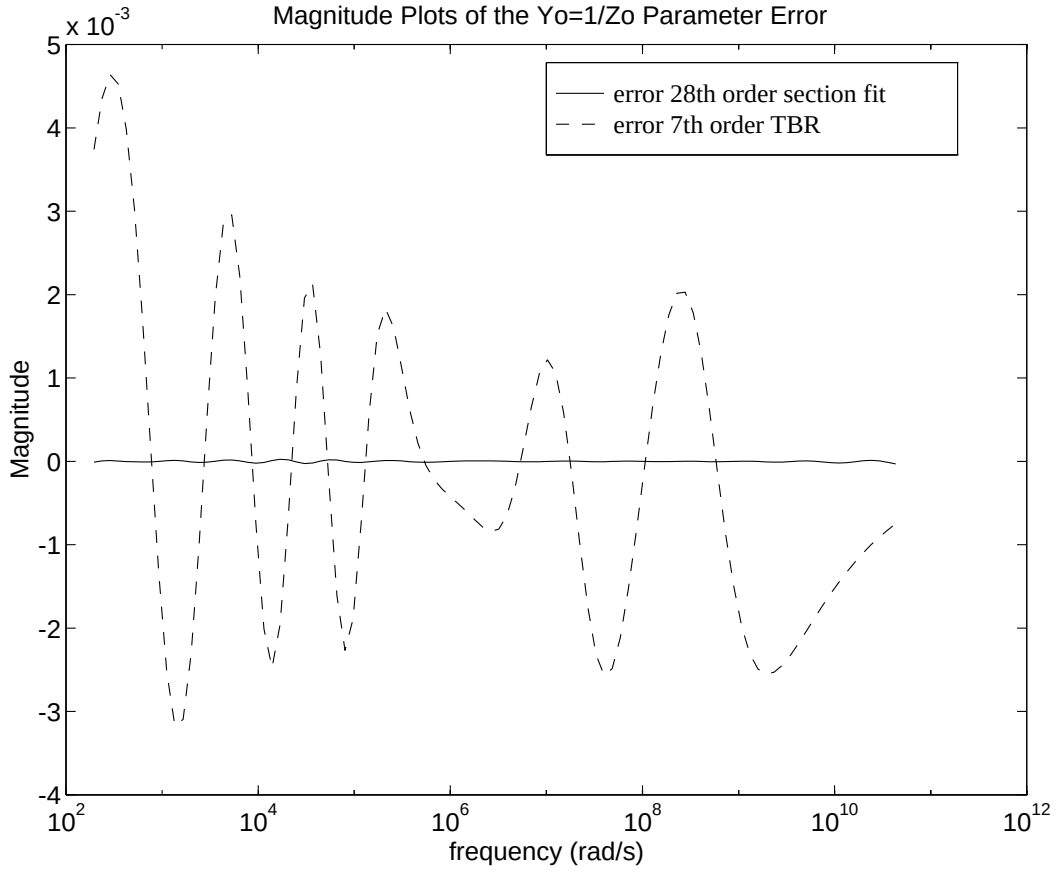

Figure 7: Magnitude plots of the errors with respect to the transmission line data points of the section-by-section approximant and the reduced-order transfer function for $\boldsymbol{Y}_o$.

circuit descriptions.

In Figure 8 we present the time-domain results of applying a 5 volt step to a 50Ω terminated transmission line with significant skin-effect. The pulse has a $1ns$ rise time, is applied at $t = 50ns$ and the delay of the line is $250ns$. In the figure, we compare the time response of the 7-th order reduced-order model with the time response obtained using the full sectioning based approximant, which has more than twenty poles. The fact that the two responses are indistinguishable in the figure shows that an excellent match has been obtained. In the same figure we show the time response obtained using a full convolution method applied to an impulse response obtained via inverse fast Fourier transform (iFFT) on 2048 frequency data points. As can be seen from the figure, the iFFT-derived response is equally accurate as expected since a fairly large number of frequency points were used. In Table 1 we show the CPU times required for obtaining the three time responses shown. The total number of timesteps required for obtaining the solution in the interval shown was 1004. From the results in the table, we can see that simulation of the reduced-order model is most efficient, as expected. Since the cost of recursive convolution is roughly proportional to the number of poles in the reduced-order model, the 7-th order model is over one and a half times more efficient than the sectioning approach. Both of these methods are over an order of magnitude faster than the full convolution method which shows that the recursive convolution procedure is extremely efficient. For a simulation on a longer interval, the difference in CPU times would tend to increase since, as we saw, the cost of a recursive convolution method is linear in the number of timesteps while the cost of a full convolution method is quadratic on the number of timesteps.

Algorithm	CPU time (s)
Full convolution	133
Section-by-section	13
Reduced-order model	8

Table 1: CPU time comparisons for full convolution versus recursive convolution methods. Times are in seconds on a SUN IPX.

In Figure 10 we present the time-domain results obtained from the circuit in Figure 9, using the transmission line from the previous example. The driver and the load are both CMOS inverters, where the transistors are described using SPICE3's default level 2 model with $W/L = 750$ for the p-type pullup devices and $W/L = 400$ for the n-type pull-down devices. The simulation results show clearly that the improper line termination causes reflections to transmit back and forth on the line.

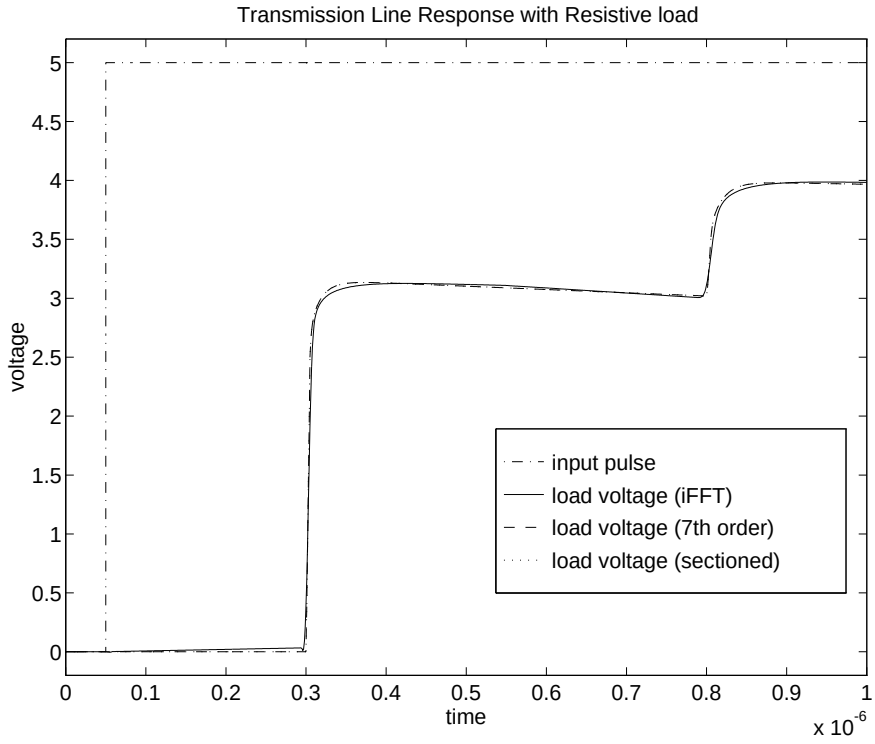

Figure 8: Time response obtained from applying a $5V$ pulse with a $1ns$ rise time at $t = 50ns$ to a resistively terminated transmission line. The figure shows the response of a line modeled with a 7 pole reduced-order model and that of a line modeled with the approximation resulting from our sectioning algorithm, which has more than 20 poles. The figure also shows the response of the line computed using full convolution with an impulse response obtained via inverse fast Fourier transform. For this example 2048 frequency points were used for the iFFT algorithm. The three waveforms are indistinguishable. The delay of the transmission line is $250ns$.

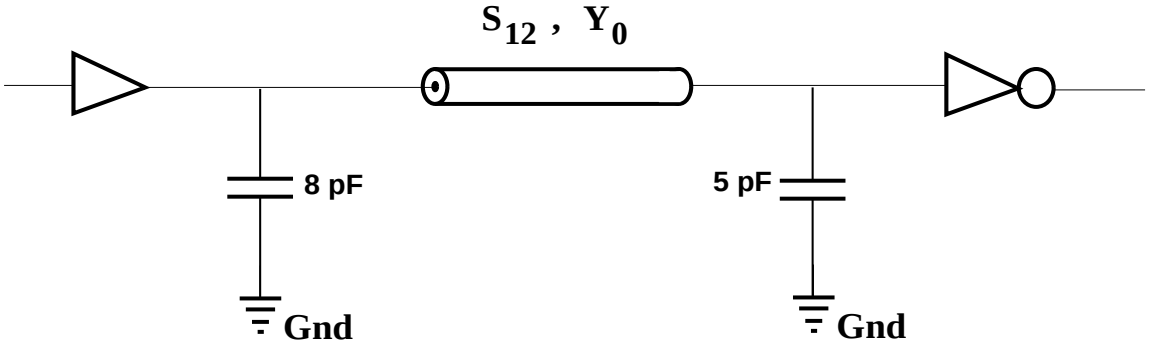

Figure 9: CMOS driver and load connected by a transmission line with skin-effect.

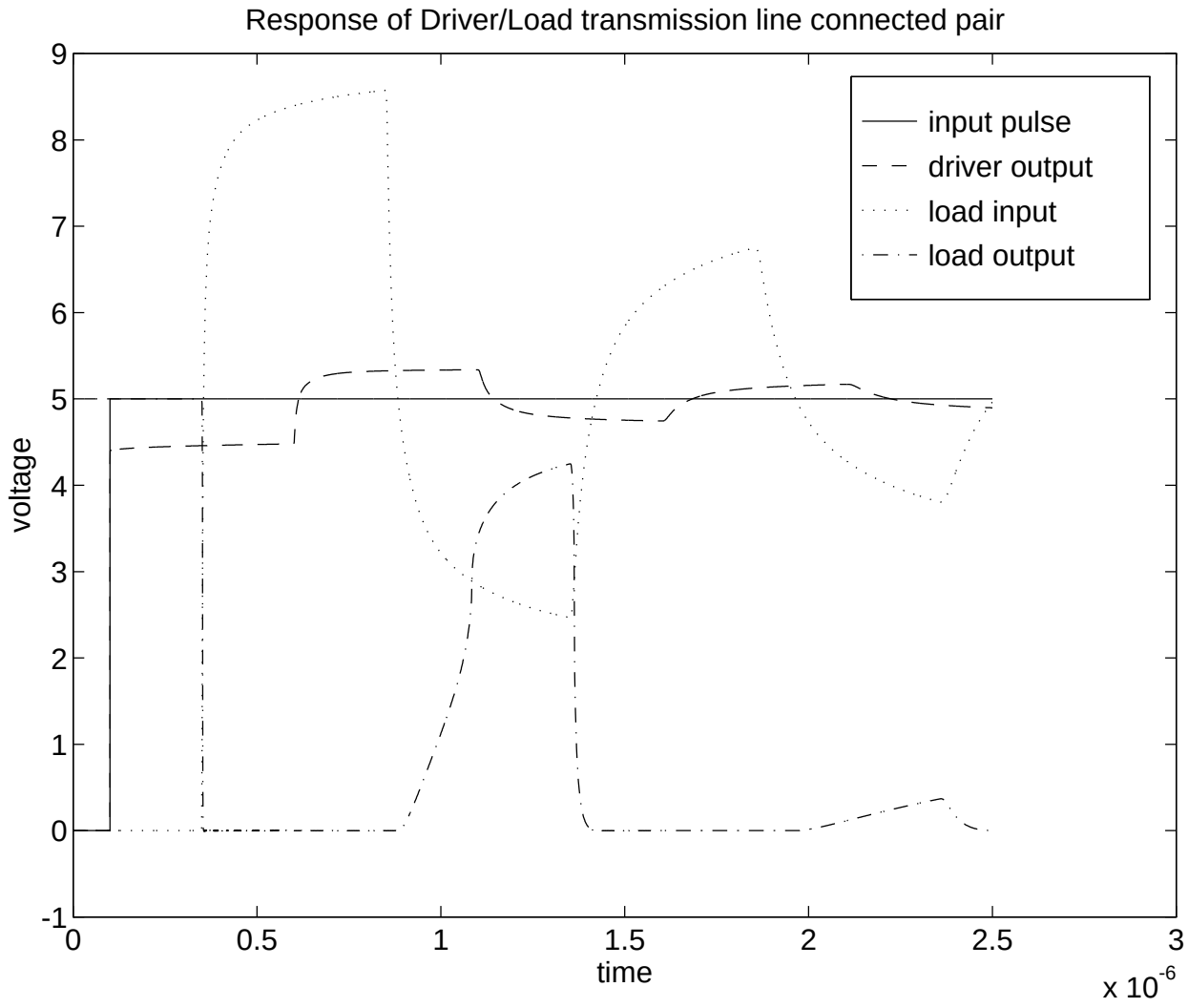

Figure 10: Time response obtained from a nonlinear circuit with a transmission line connecting driver and load. The transmission line is modeled with a 7 pole reduced-order model.

6 Conclusions

In this paper, we have proposed a robust algorithm for deriving stable, low-order, and accurate models for transmission lines based on realistic scattering data.

The main highlights of our algorithm are as follows: First, a stable, high-order transfer function is fitted to the scattering data using a two-step algorithm:

i) The frequency range is sectioned, and a section-by-section constrained $\boldsymbol{\ell_2}$, forced stable rational function approximation is fitted to the data in each frequency section.

ii) The section transfer functions are combined using a global $\boldsymbol{\ell_2}$ criterion to obtain a stable, accurate, high order model valid for the whole frequency range.

Second, a guaranteed stable, low-order model is obtained from the high-order model using the method of truncated balanced realizations.

Third, the DC gain of the low-order model is matched to that of the full model using a constrained $\boldsymbol{\ell_2}$ minimization scheme.

We have shown that our section by section approximation is very accurate and that the final stable low-order approximation derived using the truncated balanced realization has excellent match with the frequency response of the full model.

The resulting rational transfer function was incorporated in a circuit simulator, and the numerical experiments using a transmission line with skin-effects indicates that the time-domain responses match those obtained using the more computationally expensive convolution procedures currently in use for transmission line simulations. Moreover a reduction by over an order of magnitude in the computation time was observed.

Acknowledgments

This work was supported by the Advanced Research Projects Agency contract N00014-91-J-1698, the National Science Foundation contract MIP-8858764 A02, the NSF and ARPA contract 9117724-MIP, the Portuguese "Junta Nacional de Investigação Científica e Tecnológica" under project "Ciência" and grants from I.B.M. and Digital Equipment Corporation.

References

[1] J. E. Schutt-Aine and R. Mittra. Scattering Parameter Transient Analysis of Transmissions Lines loaded with Nonlinear Terminations. *IEEE Transactions on Microwave Theory and Techniques*, MTT-36:529–536, 1988.

[2] Shen Lin and Ernest S. Kuh. Transient Simulation of Lossy Interconnects Based on the Recursive Convolution Formulation. *IEEE Trans. Circuits Syst.*, 39(11):879–892, November 1992.

[3] C. Gordon, T. Blazeck, and R. Mittra. Time-domain simulation of multiconductor transmission lines with frequency-dependent losses. *IEEE Trans. CAD*, 11(11):1372–1387, November 1992.

[4] Lawrence T. Pillage, Xiaoli Huang, and Ronald A. Rohrer. AWEsim: Asymptotic Waveform Evaluation for Timing Analysis. In 26^{th} *ACM/IEEE Design Automation Conference*, pages 634–637, Las Vegas, Nevada, June 1989.

[5] J. E. Bracken, V. Raghavan, and R. A. Rohrer. Interconnect Simulation with Asymptotic Waveform Evaluation. *IEEE Trans. Circuits Syst.*, 39(11):869–878, November 1992.

[6] F.-Y. Chang. Transient Simulation of Nonuniform Coupled Lossy Transmission Lines Characterized with Frequency-Dependent Parameters – Part II: Discrete-Time Analysis. *IEEE Trans. Circuits Syst.*, 39(11):879–892, November 1992.

[7] Bruce Moore. Principal Component Analysis in Linear Systems: Controllability, Observability, and Model Reduction. *IEEE Transactions on Automatic Control*, AC-26(1):17–32, February 1981.

[8] Keith Glover. All optimal Hankel-norm approximations of linear multivariable systems and their $\mathbf{l}^{\infty}$-error bounds. *International Journal on Control*, 39(6):1115–1193, June 1984.

[9] Eli Chiprout and Michael Nakhla. Generalized Moment-Matching Methods for Transient Analysis of Interconnect Networks. In 29^{th} *ACM/IEEE Design Automation Conference*, pages 201–206, Anaheim, California, June 1992.

[10] Thomas Kailath. *Linear Systems*. Information and System Science Series. Prentice-Hall, Englewood Cliffs, New Jersey, First edition, 1980.

[11] L. Pernebo and L. M. Silverman. Model Reduction via Balanced State Space Representations. *IEEE Transactions on Automatic Control*, AC-27, April 1982.

[12] L. Miguel Silveira, Ibrahim M. Elfadel, and Jacob K. White. A Guaranteed Stable Order Reduction Algorithm for Packaging and Interconnect Simulation. In *Proceedings*

of the 2^{nd} *Topical Meeting on Electrical Performance of Electronic Packaging*, pages 165–168, Monterey, California, October 1993.

[13] L. Miguel Silveira, Ibrahim M. Elfadel, and Jacob K. White. Efficient Frequency-Domain Modeling and Circuit Simulation of Transmission Lines. In *Proceedings of the* 31^{st} *Design Automation Conference*, pages 634–639, San Diego, CA, June 1994.

[14] Thomas L. Quarles. The SPICE3 Implementation Guide. Technical Report ERL M89/44, Electronics Research Laboratory Report, University of California at Berkeley, Berkeley, California, April 1989.