10-14-2016 12:00 AM

Efficient Macromodeling Techniques of Distributed Networks Using Tabulated Data

Mohamed Sahouli, *The University of Western Ontario*

Supervisor: Dr. Anestis Dounavis, *The University of Western Ontario*

A thesis submitted in partial fulfillment of the requirements for the Master of Engineering Science degree in Electrical and Computer Engineering

Abstract

Efficient macromodeling techniques used to model multi-port distributed systems using tabulated data are presented. First a method to macromodel large multiport systems characterized by noisy frequency domain data is shown. The proposed method is based on the vector fitting algorithm and uses an instrumental variable approach and QR decomposition to formulate the least squares equations. The instrumental variable method minimizes the biasing effect of the least squares solution caused by the noise of the data samples while QR decomposition decouples the least squares equations of multiport systems described by common set of poles. It is illustrated, that the proposed approach can increase the accuracy of the pole-residue estimates with less iteration when compared to the traditional QR decomposition vector fitting method.

Second, a method to obtain delay rational macromodels of electrically long interconnects from tabulated frequency data, is presented. The proposed algorithm first extracts multiple propagation delays and splits the data into single delay regions using a time-frequency decomposition transform. Then, the attenuation losses of each region is approximated using the Loewner Matrix approach. The resulting macromodel is a combination of delay rational approximations. Numerical examples are presented to illustrate efficiency of the proposed method compared to traditional Loewner where the delays are not extracted beforehand.

Keywords: Delay extraction, High-speed interconnect, instrumental variable, macromodeling, noise, Loewner Matrix, rational approximation, time-frequency decomposition, transmission lines, vector fitting.

Acknowledgment

This thesis would not be possible without the invaluable help with my supervisor Dr. Anestis Dounavis, his open door policy and constant encouragement were a real help throughout my degree.

I would like to acknowledge my lab colleagues: Tarek, Sadia and Sara for their support and providing respite during the course of my Master's work.

A great thanks goes to my family who I love dearly. The constant encouragement of my sisters and brother Imane, Fardous and Ismail are always very important to me. Finally, the deepest thanks goes to my parents, for whom these few words can never do justice to how important they are to me, they always pushed me to strive to be the best I can be and supported me and believed in me.

Contents

List of Figures

List of Tables

List of Symbols

s	Laplace variable ($j\omega$).
ω	Angular frequency.
j	Imaginary number.
τ	Time Delay.
t	Time.
$\in$	Belongs to.
$\|.\|$	Absolute value.
$arg[]$	Principal argument.
$\mathcal{P}$	Cauchy principal value.

Abbreviations

IFFT	Inverse fast Fourier transform.
LM	Loewner matrix.
MNA	Modified nodal analysis.
ODE	Ordinary differential equations.
PDE	Partial differential equations.
RVF	Relaxed vector fitting.
SNR	Signal-to-noise ratio.
VF	Vector fitting.

Chapter 1

Introduction

1.1 Background and Motivation

As the operating frequencies of high-speed electrical networks continue to increase, the issue of signal propagation has taken a more prominent role in the design cycle. This increase in operating speed has made the analysis of interconnects a major part in the design cycle of electronic systems since previously neglected effects (Fig. 1.1) such as delay, crosstalk and attenuation can greatly affect the signal propagation on interconnects and the overall performance of electronic systems [1, 3]. The process of analysis, design, and validation of the interconnect necessary for the successful transmission of signals is called signal integrity [4]. Because, interconnects exist at various levels in the design of any electronic systems such as on-chip, packaging structures, printed circuit boards (PCB) and backplanes, interconnects are considered to be the cause of the majority of signal degradation in the high-speed electronic systems [1, 3, 5].

Simulating interconnect with circuit elements can be done using circuit simulators such as SPICE [6], however in order to use these simulators, electrical models of the interconnects

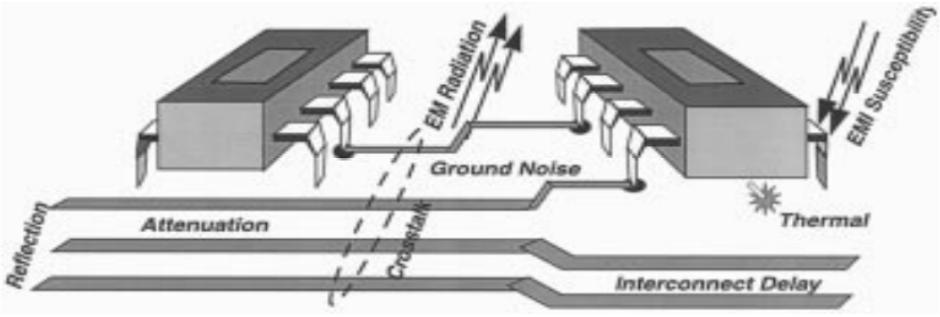


Figure 1.1: Interconnect effects [1]

need to be done. Depending on the physical interconnect structure, signal rise times, and the operating frequency of the circuit, different models can be used [1]. Developing analytical interconnect models for the case when there are process variations, non-uniformities and complex geometries is a challenging task, since analytical models require a full solution of partial differential equations (PDE) which are hard to solve by circuit simulators. Analysis of distributed transmission lines when nonlinear elements are present give rise to the so-called mixed frequency/time problem [1]. This problem arises from the fact that circuit simulators solve time-domain (transient) analysis using ordinary differential equations (ODE), while the PDE are traditionally solved in the frequency-domain. To overcome this problem, macromodeling techniques are used to convert the interconnect models into ODE.

Under these conditions, the behavior of interconnects lumped with other electromagnetic modules such as vias, connectors, and packages is often characterized by tabulated data, obtained by measurements or by electromagnetic simulations [1, 3, 7–9]. Using inverse fast Fourier transform (IFFT) [10] to convert the frequency-domain data into time domain data, analysis of port responses can be computed by using convolution [11, 12]. Using IFFT directly on the frequency-domain data can lead to inaccurate transient simulation [13, 14]. Another approach seeks to approximate the tabulated frequency data as a set of ODE, which can then be easily incorporated with circuit simulators directly or converted into equivalent circuit, using techniques referred as macromodel synthesis [1], the process of building mathematical models of dynamic systems from measured data is called system identification, and it represents an

important issue for the analysis of high-speed circuits.

Macromodeling of distributed networks characterized by frequency-domain data, is usually performed using rational curve fitting techniques [1,3,7–9,15,16]. Among these techniques, the Vector Fitting (VF) algorithms [9, 15–17] have emerged as a popular system identification tool since the rational approximation is formulated as a linear least squares problem and relies on an iterative pole relocation approach to improve the approximation. This leads to better numerical stability and robustness when compared to non-iterative or polynomial approaches. In recent years, Loewner Matrix (LM) [18, 19] has been proposed as an alternative to VF. Unlike, what is done in VF, which relies on multiple different order approximations to determine the best order to fit the data [20], LM provides a direct mechanism to identify the order based on the magnitudes of a Singular Value Decomposition (SVD) [18]. Furthermore, for the case of multi-port networks the time-domain macromodel can be realized with fewer state space equations when compared to VF [18, 19].

1.2 Objectives

Interconnect models derived from tabulated data are often used to obtain macromodels and interact with SPICE circuit solvers [1, 3, 7–9, 15, 16].

The objective of this thesis is to develop efficient macromodeling algorithms for high-speed distributed networks by expanding on the known techniques. These algorithms do not make any assumption about the underlying structure of the devices under study. Using only tabulated frequency-data such as Y or S parameters obtained using measurements or by electromagnetic simulations to characterize the networks. The methods developped in this work seek to address specifically two issues: noisy tabulated data and tabulated data obtained from long distributed interconnect.

1.3 Contributions

The main contributions of this thesis are:

1. A method is proposed to efficiently macromodel large multiport networks characterized by noisy frequency domain data. The method uses the concept of instrumental variable [21, 22] to minimize the biasing effect of the least squares caused by the noise present in the data samples leading to more accurate solutions in fewer iterations.

2. Delayed rational approximations from tabulated frequency data are derived using the LM method [18]. The method uses explicit delay extraction to extract propagation delays estimates and partitions the data into single delay regions using a time-frequency transform.

3. From the partitioned regions, a new approach to refine the delay estimates using the LM method is performed.

4. The delay extraction LM algorithm is extended to multi-port networks. Numerical examples are presented to illustrate efficiency of the proposed method compared to traditional Loewner where the delays of the transfer function are not extracted.

1.4 Organization of the Thesis

The organization of the thesis is as follows. Chapter 2 offers an overview of different macromodeling based on tabulated data, along with some of the issues related to these methods and some of the state-of-art proposed solutions. Chapter 3 presents an efficient method to macromodel large multiport systems characterized by noisy frequency domain data, using a modified VF algorithm. It is illustrated, that the proposed approach can increase the accuracy

of the pole-residue estimates with less iteration when compared to the traditional vector fitting method. Chapter 4 presents a method to obtain delay rational macromodels of electrically long interconnects from tabulated frequency data. Numerical examples are presented to illustrate efficiency of the proposed method compared to traditional Loewner where the delays are not extracted beforehand. A summary of the work presented along with suggestions of future related works are presented in the concluding Chapter 5.

Chapter 2

Literature Review

2.1 Overview

As mentioned in the previous chapter, due to system complexity, process variations and non-uniformities of electrical circuits, rational macromodel approximations from tabulated measured data are often used to model high speed interconnects.

Among these techniques, the Vector Fitting (VF) algorithms [9, 15–17] have emerged as a popular system identification tool since the rational approximation is formulated as a linear least squares problem and relies on an iterative pole relocation approach to improve the approximation. This leads to better numerical stability and robustness when compared to non-iterative or polynomial approaches.

Although the VF method works well in estimating rational transfer functions, this is not the case when dealing with large multi-port networks or when data samples are contaminated by noise. Over the years, several modifications have been proposed to improve computational efficiency and accuracy of this method. To efficiently calculate the transfer functions of multi-port

networks using a common set of poles, a QR decomposition method was proposed to decouple least squares equations of each transfer function [17]. The QR decomposition approach was also used to implement a parallel processing VF algorithm for large multi-port networks [2].

Another issue with VF is that it has difficulty estimating the poles of transfer functions when data sample measurements are contaminated by noise. This is due to the fact that the noise of the data causes the least squares solution to bias the location of the poles.

Various enhancements have been proposed to deal with noisy data, such as pole adding and skimming method [23], least squares weighted functions [24] and instrumental variable VF method [25], these method will be discussed in detail later in this chapter.

In recent years, Loewner Matrix (LM) [18, 19] has been proposed as an alternative to VF. Unlike, what is done in VF, which relies on multiple different order approximations to determine the best order to fit the data [20], LM provides a direct mechanism to identify the order based on the magnitudes of a Singular Value Decomposition (SVD) [18]. Furthermore, for the case of multi-port networks the time-domain macromodel can be realized with fewer state space equations when compared to VF [18, 19].

When dealing with long interconnects, attempting to approximate the tabulated data as rational functions, will typically require many poles to accurately approximate the data [13, 29–31]. For distributed networks with long delays, methods based on delayed rational functions can be used to provide accurate and efficient macromodels [13, 30, 31]. These techniques extract the propagation delays from the tabulated data, while the remaining attenuation losses are approximated using low order rational functions, leading to more compact macromodels with fewer poles when compared to using only rational functions.

In the following sections an overview of two popular rational curve techniques, vector fitting (VF) and Loewner matrix (LM) are presented, along with some of the issues that can be encountered when macromodeling networks characterized by tabulated data.

2.2 Vector Fitting

Using an iterative approach, the vector fitting algorithm allows to obtain a rational function to approximate a set of tabulated data obtained either by measurement or electromagnetic simulation. It was originally introduced in context of analysis of transmission lines and power systems, but was later extended to many fields, with signal integrity being among them.

Using the vector fitting algorithm, the objective is to approximate a set of tabulated data $(s_k, Y(s_k))_{k=1}^{K}$ to get a rational function of the form

$$f(s) = \sum_{n=1}^{N} \frac{c_n}{s - p_n} + d + se \tag{2.1}$$

where p_n and c_n correspond to poles and residues respectively, these quantities can either be real or complex conjugates, while the real variables d and e are optional; s is the Laplace variable and $Y(s_k)$ is the value of the data at the particular k frequency. N is the order the rational function.

VF is an iterative method that seeks to solve for the unknowns p_n, c_n, d and e. This nonlinear problem (because of the term p_n that appears in the denominator) is solved by making into a linear problem following a two stage process: 1) pole identification followed by 2) residue identification.

Pole and Residues Identification

In the first stage, the goal is to obtain an approximation for the poles p_n. This is done by introducing an unknown function $\sigma(s)$ and the following system.

$$\begin{bmatrix} \sigma(s)f(s) \\ \sigma(s) \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^{N} \frac{c_n}{s-\bar{p}_n} + d + se \\ \sum_{n=1}^{N} \frac{\tilde{c}_n}{s-\bar{p}_n} + 1 \end{bmatrix} \tag{2.2}$$

where the terms $\bar{p}_n$ are starting values for the poles and the rest of the remaining terms are unknown.

Multiplying the second row of (2.2) with $f(s)$, the following relation is obtained

$$\sum_{n=1}^{N} \frac{c_n}{s-\bar{p}_n} + d + se = (\sum_{n=1}^{N} \frac{\tilde{c}_n}{s-\bar{p}_n} + 1)Y(s) \tag{2.3}$$

Doing so, results in a linear problem where the unknowns are $c_n, \tilde{c}_n, d$ and e. Equation (2.3) for a frequency point s_k gives a system of equation of the form

$$A_k x = b_k$$

where

$$\begin{aligned} A_k &= \begin{bmatrix} Re(q_1^k) & \dots & Re(q_N^k) & 1 & 0 & Re(\bar{q}_1^k) & \dots & Re(\bar{q}_N^k) \\ Im(q_1^k) & \dots & Im(q_N^k) & 1 & 0 & Im(\bar{q}_1^k) & \dots & Im(\bar{q}_N^k) \end{bmatrix} \\ x &= [c_1 \quad \dots \quad c_N \quad d \quad e \quad \tilde{c}_1 \quad \dots \quad \tilde{c}_N]^T \\ b_k &= \begin{bmatrix} Re(Y(s_k)) \\ Im(Y(s_k)) \end{bmatrix} \end{aligned} \tag{2.4}$$

Depending on whether the poles are real or complex, the terms of (2.4) will have different forms, to make sure that the residues are either real or come in complex conjugate form. For the case of real poles, the coefficients of (2.4) will be

$$q_i^k = \frac{1}{s_k - \bar{p}_i}$$
$$\bar{q}_i^k = \frac{-Y(s_k)}{s_k - \bar{p}_i}$$

And for the case of complex poles the coefficients of (2.4) will become

$$q_i^k = \frac{1}{s_k - \bar{p}_i} + \frac{1}{s_k - \bar{p}_{i+1}}$$
$$q_{i+1}^k = \frac{j}{s_k - \bar{p}_i} - \frac{j}{s_k - \bar{p}_{i+1}}$$
$$\bar{q}_i^k = \frac{-Y(s_k)}{s_k - \bar{p}_i} + \frac{-Y(s_k)}{s_k - \bar{p}_{i+1}}$$
$$\bar{q}_{i+1}^k = \frac{-jY(s_k)}{s_k - \bar{p}_i} + \frac{jY(s_k)}{s_k - \bar{p}_{i+1}}$$
$$c_i = Re(c_i) \quad c_{i+1} = Re(c_i)$$
$$\tilde{c}_i = Re(\tilde{c}_i) \quad \tilde{c}_{i+1} = Re(\tilde{c}_i)$$

Expanding equation (2.4) for K frequency points gives an overdetermined system of equations

$$\mathbf{Ax} = \mathbf{b}$$

From there the solution for **x** can be obtained by doing

$$\mathbf{x} = (\mathbf{A^T A})^{-1}(\mathbf{A^T b}) \tag{2.5}$$

The least squares solution obtained from (2.5) can be used to get approximations for the $\sigma(s)$ and $\sigma(s)f(s)$ functions written as

$$\begin{aligned} \sigma(s)_{fit} &= \prod_{n=1}^{N} \frac{(s-\tilde{z}_n)}{(s-\bar{p}_n)} \\ (\sigma Y)_{fit}(s) &= \prod_{n=1}^{N} \frac{(s-z_n)}{(s-\bar{p}_n)} \end{aligned} \tag{2.6}$$

Finally from (2.6) an approximation for $f(s)$ can be obtained as

$$f(s) = \prod_{n=1}^{N} \frac{(s-z_n)}{(s-\tilde{z}_n)} \tag{2.7}$$

It can be seen from equation (2.7), that the poles of $f(s)$ become the zeros of $\sigma(s)_{fit}$. Therefore, by taking the newly calculated zeros of $\sigma(s)_{fit}$ as the new guess for the poles, this procedure can be repeated until the poles converge.

In the procedure described above, it can be seen that only the zeros of the function $\sigma(s)_{fit}$ are needed to first compute the poles. Once that is done, an additional least square solution, corresponding to the second stage, is needed to obtain the residues and the terms d and e if they are included in the model.

2.2.1 Fast Vector Fitting

Although the VF method works well in estimating rational transfer functions, this is not the case when dealing with large multi-port networks. Over the years, several modifications have been proposed to improve computational efficiency and accuracy of this method. To efficiently calculate the transfer functions of multi-port networks using a common set of poles, a QR decomposition method can be used to decouple least squares equations of each transfer function [17].

Assuming a set of data coming from a multiport system $\{s_k, Y^j(s_k)\}_{k=1}^K$, where $j = 1 \dots J$ corresponds to the number of transfer function in the transfer function matrix. For a system with P number of ports, $J = P^2$. The objective is to find the function

$$f^j(s) = \frac{(\sigma H^j)(s)}{\sigma(s)} = \frac{\sum_{n=1}^{N} \frac{c_n^j}{s-p_n} + d + se}{\sum_{n=1}^{N} \frac{\tilde{c}_n}{s-p_n} + 1} \tag{2.8}$$

such that $\mathbf{f}(\mathbf{s_k}) \approx \mathbf{Y}(\mathbf{s_k})$. The terms $c_n^j, d, e, \tilde{c}_n$ are unknown coefficients and $\bar{p}_n$ are chosen heuristically in the first iteration. Using the same reasoning that was applied in the previous section, the solution for the J number of unknown equations (2.8) corresponds to solving the following overdetermined set of equations

$$\begin{bmatrix} \mathbf{X} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{H}^1\mathbf{X} \\ \mathbf{0} & \mathbf{X} & \mathbf{0} & \mathbf{0} & -\mathbf{H}^2\mathbf{X} \\ & & \cdots & \cdots & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{X} & -\mathbf{H}^J\mathbf{X} \end{bmatrix} \begin{bmatrix} \mathbf{C}^1 \\ \mathbf{C}^2 \\ \cdots \\ \mathbf{C}^J \\ \mathbf{C}_p \end{bmatrix} = \begin{bmatrix} \mathbf{H}^1\hat{\mathbf{1}} \\ \mathbf{H}^2\hat{\mathbf{1}} \\ \cdots \\ \mathbf{H}^J\hat{\mathbf{1}} \end{bmatrix} \tag{2.9}$$

where

$$\begin{aligned} \mathbf{H'}^j &= [Y^j(s_1) \dots Y^j(s_K)], \\ \mathbf{H}^j &= diag([Re(\mathbf{H'}^j) Im(\mathbf{H'}^j)]), \\ \hat{\mathbf{1}} &= (2Kx1) \text{ column vector of one}, \\ \mathbf{X'} &= \begin{bmatrix} \frac{1}{s_1 - p_1} \cdots \frac{1}{s_1 - p_N} \\ \cdots\cdots\cdots \\ \frac{1}{s_K - p_1} \cdots \frac{1}{s_K - p_N} \end{bmatrix}, \\ \mathbf{X} &= \begin{bmatrix} Re(\mathbf{X'}) \\ Im(\mathbf{X'}) \end{bmatrix} \end{aligned}$$

and $\mathbf{C}^j$ contains the residues c_n^j and $\mathbf{C}_p$ contains the residues $\tilde{c}_n$. The procedure is the same as is done when dealing with a single transfer function, once the system is solved, the zeros of $\sigma(s)_{fit}$ denoted $z = \{z_1, \dots, z_N\}$ become the poles of $f^j(s)$. The process is repeated again by making $\{p_1, \dots, p_N\} = \{z_1, \dots, z_N\}$ until convergence.

It was noted in [17] that since the residues c_n^j are discarded while convergence is not reached, the algorithm could be made faster if there was a way to transform system of (2.9) in such a

way that only the residues $\tilde{c}_n$ are solved for, thus making the overdetermined set of equations smaller. This is done by using the QR decomposition as described next.

Each j-th transfer function can be expressed as

$$[\mathbf{X} \quad -\mathbf{H}^j\mathbf{X}]\begin{bmatrix}\mathbf{C}^j\\ \mathbf{C}_p\end{bmatrix} = \mathbf{Q}^j\begin{bmatrix}\mathbf{R}^j_{11} & \mathbf{R}^j_{12}\\ \mathbf{0} & \mathbf{R}^j_{22}\end{bmatrix}\begin{bmatrix}\mathbf{C}^j\\ \mathbf{C}_p\end{bmatrix}$$

Where the right hand side of the equation is the QR decomposition. Then by combining the the factorization of all the matrices, the reduced set of equations where only the coefficients $\mathbf{C}_p$ are the unknowns

$$\begin{bmatrix}\mathbf{R}^1_{22}\\ \dots\\ \mathbf{R}^J_{22}\end{bmatrix}\mathbf{C}_p = \begin{bmatrix}(\mathbf{Q}^1)^T\mathbf{H}^1\hat{\mathbf{1}}\\ \dots\\ (\mathbf{Q}^J)^T\mathbf{H}^J\hat{\mathbf{1}}\end{bmatrix}$$

Finally the solution is

$$\mathbf{C}_p = (\sum_{j=1}^{J}[(\mathbf{R}^j_{22})^T\mathbf{R}^j_{22}])^{-1}\sum_{j=1}^{J}(\mathbf{Q}^j\mathbf{R}^j_{22})^T\mathbf{H}^j\hat{\mathbf{1}} \tag{2.10}$$

Once the poles solution converges, the residues are solved by using what is done for the single transfer function case. A visualization summary of the fast vector fitting algorithm can be seen in the Fig. 2.1.

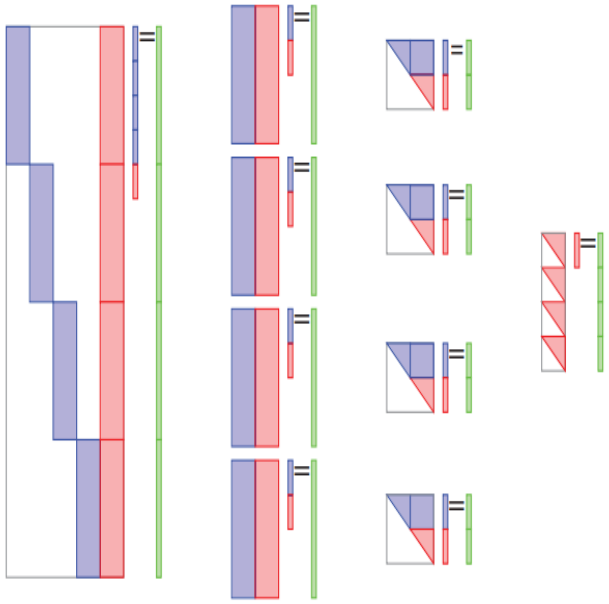


Figure 2.1: Application of QR on Multiport Networks [2].

2.2.2 Vector Fitting with Noisy Tabulated Data

One issue with VF is that it has difficulty estimating the poles of transfer functions when data sample measurements are contaminated by noise. This is due to the fact that the noise of the data causes the least squares solution to bias the location of the poles. Various enhancements have been proposed to deal with noisy data, such as relaxed vector fitting [15], pole adding and skimming method [23], least squares weighted functions [24] and instrumental variable VF method [25]. These techniques are briefly presented in the next sections.

Relaxed Vector Fitting

In [15], a change is made the original vector fitting algorithm, where the weight function $\sigma(s)$ is changed to

$$\sigma(s) = \sum_{n=1}^{N} \frac{\tilde{c}_n}{s - \bar{p}_n} + \tilde{c}_0 \tag{2.11}$$

using this new $\sigma(s)$, equation (2.2) becomes

$$\sum_{n=1}^{N} \frac{c_n}{s - \bar{p}_n} + d + se = (\sum_{n=1}^{N} \frac{\tilde{c}_n}{s - \bar{p}_n} + \tilde{c}_0)Y(s) \tag{2.12}$$

Using this new form, (2.4) is now expressed at each frequency sample s_k as

$$\begin{aligned} A_k &= \begin{bmatrix} Re(q_1^k) & \dots & Re(q_N^k) & 1 & 0 & Re(\bar{q}_1^k) & \dots & Re(\bar{q}_N^k) \\ Im(q_1^k) & \dots & Im(q_N^k) & 1 & 0 & Im(\bar{q}_1^k) & \dots & Im(\bar{q}_N^k) \end{bmatrix} \\ x &= [c_1 \quad \dots \quad c_N \quad d \quad e \quad \tilde{c}_0 \quad \tilde{c}_1 \quad \dots \quad \tilde{c}_N]^T \\ b_k &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{aligned} \tag{2.13}$$

where the terms are similar to what they were in the original form the vector fitting algorithm, however since now b_k is now equal, in order to avoid the trivial null solution, the following equation is added

$$Re\{\sum_{k=1}^{K}(\sum_{n=1}^{N} \frac{\tilde{c}_n}{s - \bar{p}_n} + \tilde{c}_0)\} = K \tag{2.14}$$

Using a similar approach as the previous section an overdetermined system of equation is obtained and the rest of the algorithm is the same as the original vector fitting. This new approach is often called relaxed vector fitting, and has been shown to have better properties when the tabulated data is contaminated by noise, since it can improve the reallocation of the poles [15]

Pole Adding and Skimming Method

In [20], another way to deal with rational approximation of data contaminated by noise using vector fitting is presented. The method identifies so-called spurious poles which are said to be responsible for the possible non-convergence of the standard vector fitting with noisy data. These spurious poles are dealt with in a two step process, first they are identified and then removed. In addition to dealing with noise, the method also presents a way to estimate the order of the underlying system by incrementally increasing the number of poles and applying relocation whenever it is necessary. The called is referred to in the paper as vector fitting with adding and skimming (VF-AS) by the authors. The spurious poles are said affect the convergence of the vector fitting method, since they will tend be stuck at a specific location and thus will never go to a better value at each subsequent iteration, indeed rather than trying to fit true data, these spurious poles try to fit the noise instead. Another way of looking at it, would be to consider that the constraint condition of (2.12) is not strong enough to for the spurious poles to converge to their expected location.

In order to address this issue and enhance the convergence of the poles, a hard relocation is proposed. This is a process through which an automatic detection of the spurious poles is done. Then, the poles are placed in a location of the complex plane that is closer to the true poles. Since VF is sensitive to the initial guess of the solution, it is expected that a better guess for the poles will also improve the behavior of VF in the presence of noise.

Least Squares Weighted Functions

In [24], another modification to the standard vector fitting algorithm, when fitting data contaminated by noise, is proposed. In this method, the noise is assumed to be colored additive with a zero-mean circular complex Gaussian distribution. The proposed method, uses information about the variance of the data samples obtained with the use of least-squares weighting functions. Estimation of the variance is done by performing measurements on a point-by-point basis and is incorporated in the weighting function of vector fitting algorithm as

$$w(s) = \frac{1}{\sigma^2(s)} \tag{2.15}$$

Quality information of the data samples is then given to the least-squares estimator using the weighting functions (2.15). This helps reduce the effect of the noise by improving the retrieval of the real behavior of system under study [24]

Instrumental Variable Vector Fitting

This section presents another modification to the vector fitting algorithm, when the tabulated data is contaminated by noise. First, a more detailed look at how the nosy data samples can bias the solution of the poles.

Consider the noisy tabulated data set defined as [25]

$$\hat{Y}(s) = Y(s) + \varepsilon \tag{2.16}$$

where $Y(s)$ is the exact transfer function and ε is a zero-mean complex noise. In the presence of noise, the system to be solved is modified and new terms are introduced due to the noise, A_k and b_k are replaced by

$$\begin{aligned} \hat{A}_k &= A_k + H_k^A \\ \hat{b}_k &= b_k + H_k^b \end{aligned} \tag{2.17}$$

where the extra terms H_k^A and H_k^B are due to ε and are defined as follows [25]:

$$\begin{aligned} H_k^A &= \begin{bmatrix} 0 & \cdots & 0 & Re(\tilde{e}_1^k) & \cdots & Re(\tilde{e}_N^k) \\ 0 & \cdots & 0 & Im(\tilde{e}_1^k) & \cdots & Im(\tilde{e}_N^k) \end{bmatrix} \\ H_k^b &= \begin{bmatrix} Re(\varepsilon_k) \\ Im(\varepsilon_k) \end{bmatrix} \end{aligned} \tag{2.18}$$

In the presence of noise the least square solution of (2.5) becomes:

$$\mathbf{x} = \left[\sum_k \hat{A}_k^T \hat{A}_k \right]^{-1} \left[\sum_k \hat{A}_k^T \hat{b}_k \right] \tag{2.19}$$

where

$$\hat{A}_k^T \hat{A}_k = A_k^T A_k + A_k^T H_k^A + (H_k^A)^T A_k + (H_k^A)^T H_k^A$$
$$\hat{A}_k^T \hat{b}_k = A_k^T b_k + A_k^T H_k^b + (H_k^A)^T b_k + (H_k^A)^T H_k^b \tag{2.20}$$

the first terms $A_k^T A_k$ and $A_k^T b_k$ of (2.20) would be the terms found if there was no noise. Since, as it was stated previously it is assumed that the disturbance ε is a zero-mean complex random noise (meaning the expected mean value $E[\varepsilon]$ is zero), the expected values of the second and third terms in (2.20) are

$$E[A_k^T H_k^A] = E[(H_k^A)^T A_k] = 0$$
$$E[A_k^T H_k^b] = E[(H_k^A)^T b_k] = 0 \tag{2.21}$$

the results in (2.21) mean that the second and third terms do not statistically bias the results of (2.19). The fourth terms are defined as [25]

$$(H_k^A)^T H_k^A = [h_{m,n}^a]$$
$$(H_k^A)^T H_k^b = [h_m^b] \tag{2.22}$$

where

$$h^a_{m,n} = \begin{cases} Re(\tilde{e}^k_{m'}) * Re(\tilde{e}^k_{n'}) + Im(\tilde{e}^k_{m'}) * Im(\tilde{e}^k_{n'}), \\ \qquad\qquad m, n > N + 2 \\ 0, \qquad\qquad \text{otherwise} \end{cases}$$
$$h^b_m = \begin{cases} Re(\tilde{e}^k_{m'}) * Re(\varepsilon_k) + Im(\tilde{e}^k_{m'}) * Im(\varepsilon_k) \\ \qquad\qquad m > N + 2 \\ 0, \qquad\qquad \text{otherwise} \end{cases} \tag{2.23}$$

with $m' = m - (N + 2)$ and $n' = n - (N + 2)$. Since the expected mean values of $E[Re(\varepsilon)^2]$ and $E[Im(\varepsilon)^2]$ are not equal to zero, we get the following

$$E[(H^A_k)^T H^A_k] \neq 0$$
$$E[(H^A_k)^T H^b_k] \neq 0 \tag{2.24}$$

this means that the nonzero $h^a_{m,n}$ terms will bias the $\hat{A}^T_k \hat{A}_k$ matrices, thus affecting the solution of all unknown variables in (2.19). The nonzero h^b_m terms bias the residues which are used to determine the poles of $H(s)$. This biasing effect of h^b_m is the main reason for the failure of vector fitting to capture the actual poles of the system in the presence of zero-mean noise. Next a technique to deal with least squares bias using the concept of instrumental variables is presented.

The least square solution using the instrumental variable is defined as

$$X = \left[\sum_k \Psi_k^T \hat{A}_k \right]^{-1} \left[\sum_k \Psi_k^T \hat{b}_k \right] \tag{2.25}$$

where Ψ_k is called the instrumental variable, just as $\hat{A}_k$ was defined from the noisy data $\hat{Y}(s) = Y(s)+\varepsilon$, $\Psi_k = A_k + H_k^{\Psi}$ is also obtained from a set of data $\hat{Y}(s) = Y(s)+\rho$. Note that in both sets, $Y(s)$ is the same theoretical noise free transfer function and ε and ρ are uncorrelated Gaussian noise.

Using the definition of Ψ, the terms in (2.19) become:

$$\begin{aligned} \Psi_k^T \hat{A}_k &= A_k^T A_k + A_k^T H_k^A + (H_k^{\Psi})^T A_k + (H_k^{\Psi})^T H_k^A \\ \Psi_k^T \hat{b}_k &= A_k^T b_k + A_k^T H_k^b + (H_k^{\Psi})^T b_k + (H_k^{\Psi})^T H_k^b \end{aligned} \tag{2.26}$$

The first three terms of (2.26) have the same values as those in (2.20), the only difference is in the fourth term. Since the values of the H_k terms come from the noise in the data sets and we are dealing with uncorrelated noise, the expected value of the fourth term also becomes zero when using the instrumental variable approach, leaving the least square solution with only the no-noise terms [21, 22, 26–28].

$$\begin{aligned} E[(H_k^{\Psi})^T H_k^A] &= 0 \\ E[(H_k^{\Psi})^T H_k^b] &= 0 \end{aligned} \tag{2.27}$$

2.3 Loewner Matrix

The Loewner Matrix method [18] seeks to macromodel the transfer function $Y(s)$ as

$$Y(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} + s\mathbf{Y}^{\infty} \tag{2.28}$$

where $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n\times n}, \mathbf{B} \in \mathbb{R}^{n\times 1}, \mathbf{C} \in \mathbb{R}^{1\times n}, \mathbf{D} \in \mathbb{R}, \mathbf{Y}^{\infty} \in \mathbb{R}$ describe the system of order n. The descriptor state space matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and $\mathbf{E}$ are obtained as follows.

First the given data is split into two groups, usually referred to right and left interpolation data points as

$$\begin{aligned}
&[s_1 \dots s_K] = [\mu_1 \dots \mu_{\underline{k}}] \cup [\lambda_1 \dots \lambda_{\overline{k}}] \\
&[Y(s_1) \dots Y(s_K)] = \\
&[Y(\mu_1) \dots Y(\mu_{\underline{k}})] \cup [Y(\lambda_1) \dots Y(\lambda_{\overline{k}})]
\end{aligned}$$

where $\underline{k} + \overline{k} = K$ and

$$\begin{cases}
\underline{k} = \overline{k} = K/2 & \text{if } K \text{ is even} \\
\underline{k} = \overline{k} + 1 = (K+1)/2 & \text{if } K \text{ is odd}
\end{cases}$$

There are many ways that the data could be split. In this work, the alternating splitting of the data is used since this will lead to better conditioning of the Loewner matrix as illustrated

in [18, 19].

$$\mathbf{V} = \begin{bmatrix} v_1 \\ \vdots \\ v_{\underline{k}} \end{bmatrix} = \begin{bmatrix} Y(\mu_1) \\ \vdots \\ Y(\mu_{\underline{k}}) \end{bmatrix},$$

$$\mathbf{W} = \begin{bmatrix} w_1 \dots w_{\overline{k}} \end{bmatrix} = \begin{bmatrix} Y(\lambda_1) \dots Y(\lambda_{\overline{k}}) \end{bmatrix}$$

With the left data set (μ_i, v_i) and right data set (λ_i, w_i), the $\underline{k} \times \overline{k}$ Loewner and Shifted Loewner matrices are computed as follows

$$\mathbb{L} = \begin{bmatrix} \frac{v_1 - w_1}{\mu_1 - \lambda_1} & \cdots & \frac{v_1 - w_{\overline{k}}}{\mu_1 - \lambda_{\overline{k}}} \\ \vdots & \ddots & \vdots \\ \frac{v_{\underline{k}} - w_1}{\mu_{\underline{k}} - \lambda_1} & \cdots & \frac{v_{\underline{k}} - w_{\overline{k}}}{\mu_{\underline{k}} - \lambda_{\overline{k}}} \end{bmatrix} \tag{2.29}$$

$$\sigma\mathbb{L} = \begin{bmatrix} \frac{\mu_1 v_1 - \lambda_1 w_1}{\mu_1 - \lambda_1} & \cdots & \frac{\mu_1 v_1 - \lambda_{\overline{k}} w_{\overline{k}}}{\mu_1 - \lambda_{\overline{k}}} \\ \vdots & \ddots & \vdots \\ \frac{\mu_{\underline{k}} v_{\underline{k}} - \lambda_1 w_1}{\mu_{\underline{k}} - \lambda_1} & \cdots & \frac{\mu_{\underline{k}} v_{\underline{k}} - \lambda_{\overline{k}} w_{\overline{k}}}{\mu_{\underline{k}} - \lambda_{\overline{k}}} \end{bmatrix} \tag{2.30}$$

Once the Loewner and shifted Loewner are computed, the next step is to determine the order of the approximation. In order to do that, a singular value decomposition (SVD) is performed on $(s\mathbb{L} - \sigma\mathbb{L})$. Any value of s can be chosen as long as it is not the eigenvalue of the $(\sigma\mathbb{L}, \mathbb{L})$ matrix pencil [18], resulting in the following expression

$$\text{SVD}(s\mathbb{L} - \sigma\mathbb{L}) = [\mathbf{Y}, \mathbf{\Sigma}, \mathbf{X}] \tag{2.31}$$

where $\mathbf{\Sigma}$ is a diagonal matrix containing the singular values. The order n of the approximation is chosen as the location where a large drop of the normalized singular value happens as described in [18, 19]. The descriptor system matrices are constructed as

$$\begin{aligned} \mathbf{A} &= -\mathbf{Y_n}^* \sigma\mathbb{L}\mathbf{X_n}, \quad \mathbf{B} = \mathbf{Y_n}^*\mathbf{V}, \\ \mathbf{C} &= \mathbf{W}\mathbf{X_n}, \quad \mathbf{E} = -\mathbf{Y_n}^*\mathbb{L}\mathbf{X_n} \end{aligned} \tag{2.32}$$

where $\mathbf{X_n} \in \mathbb{R}^{\bar{k}\times n}$ and $\mathbf{Y_n} \in \mathbb{R}^{\underline{k}\times n}$ are constructed from the first n columns of $\mathbf{X}$ and $\mathbf{Y}$ of (2.31) respectively [18, 19].

The method presented above leads to strictly proper rational approximations (i.e. $\mathbf{D}$ and $\mathbf{Y}^\infty$ is equal to zero). However, for the case when it is required to use $\mathbf{D}$ and $\mathbf{Y}^\infty$, setting these terms to zero may lead to unstable and inaccurate macromodels as illustrated in [19]. In this work, if the LM approximation produces unstable poles, the $\mathbf{D}$ and $\mathbf{Y}^\infty$ terms are extracted by first extracting the stable poles of the system, these will constitute the $\mathbf{A}, \mathbf{E}, \mathbf{B}, \mathbf{C}$ matrices. As for the $\mathbf{D}$ and $\mathbf{Y}^\infty$ terms, they are obtained by fitting the remaining unstable poles using a first order polynomial.

2.4 Macromodeling Data from Long Interconnects

Depending on the type of structure under study, different approach exists for developing macromodels of distributed networks. When dealing with long interconnects, attempting to approximate the tabulated data as rational functions, will typically require many poles to accurately approximate the data [13, 29–31]. For distributed networks with long delays, methods based on delayed rational functions can be used to provide accurate and efficient macromodels [13, 30, 31]. These techniques extract the propagation delays from the tabulated data, while the remaining attenuation losses are approximated using low order rational functions, leading to more compact macromodels with fewer poles when compared to using only rational functions. The next sections presents a summary of the different techniques that have been proposed to deal with tabulated data that describe the behavior of distributed networks with long delays.

2.4.1 Delay-Extraction Macromodeling using Hilbert Transforms

In [13, 14], a delay extraction based techniques using the Hilbert transform is presented. The method uses the concept of minimum phase functions for passive structures to obtain a delay estimate of the distributed networks. A function that has all its poles and zeros in the left-half plane is called a minimum phase function [32], in multi-port stable networks, this property is only present in diagonal elements. Considering a network of the following form

$$\mathbf{Y}(\mathbf{s}) = [Y_{ij}(s)] \quad (i, j \in 1, \ldots, P) \tag{2.33}$$

where the P is the number of ports. Unlike the diagonal elements, the off-diagonal elements $i \neq j$ are not minimum phase functions. Using the fact that any function from a stable distributed

system can be written as product of a minimum phase function and an all-pass function as

$$Y_{ij}(s) = Y_{ij}^{min}(s)Y_{ij}^{AP}(s) \tag{2.34}$$

where $Y_{ij}^{min}(s)$ is the minimum phase function part and $Y_{ij}^{AP}(s)$ is the all-pass portion, the Hilbert transform can be used to extract the delay and the attenuation losses part (function corresponding to the delay-free part of the function) as follows.

Equation (2.22) can be rewritten in following form

$$Y_{ij}(s) = \hat{Y}_{ij}(s)e^{-s\tau} \tag{2.35}$$

where $\hat{Y}_{ij}(s)$ is the delay-free portion and $e^{-s\tau}$ is the delay part portion corresponding to the extracted delay τ. The equivalence of (2.22) and (2.23) stems from the fact that $e^{-s\tau}$ acts as an all pass function since $|e^{-s\tau}| = 1$ (|.| corresponding to magnitude), leaving the delay-free term $\hat{Y}_{ij}(s)$ correspond to the minimum phase function. The first step of the delay-extraction based on Hilbert transform is to get an estimate for the delay term τ. This can be done by rearranging (2.23) to obtain an expression for the unknown τ term as follows [13, 14, 32]

$$\tau = -\text{Average Slope}\left(arg\left[\frac{Y_{ij}(s)}{\hat{Y}_{ij}(s)}\right]\right) \tag{2.36}$$

where $arg(z)$ refers to the principal argument of a complex number z. In order to compute τ, the attenuation losses needs to be known first. This is done using

$$|\hat{Y}_{ij}(s)| = |Y_{ij}(s)| \tag{2.37}$$

$$arg[\hat{Y}_{ij}(s)] = -HT\{ln|Y_{ij}(s)|\} \tag{2.38}$$

In (2.26) $HT\{.\}$ stands for the Hilbert transform [32], using the discrete Hilbert transform [32], (2.26) can be rewritten as [13, 14]

$$arg[\hat{Y}_{ij}(s)] = -\frac{1}{2\pi}\mathcal{P}\int_{\theta=-\pi}^{\pi} ln|Y_{ij}(\theta)| \cot\left(\frac{\omega-\theta}{2}\right)d\theta \tag{2.39}$$

where $\mathcal{P}$ denotes the Cauchy principal value of the integral that follows. Once (2.39) is solved, τ is obtained from (2.36). This allows to get all the off-diagonal terms as a product of minimum phase functions and an all-pass functions.

2.4.2 Compact Macromodeling of Electrically Long Interconnects

In [33], a method for macromodeling long interconnects is introduced. Starting from frequency-domain scattering data, the technique produces compact macromodels based on multiple delay extraction and rational approximations.

Using a set of measured frequency samples denoted as

$$H_k = H(s_k)$$

for $k = 1, \ldots, K$, the goal is to find a rational approximation of the form

$$H(s) = \sum_{m} Q_m(s) e^{-sT_m} \tag{2.40}$$

where T_m represents the signal propagation delays and $Q_m(s)$ represent effects such as attenuation losses and dispersion. Using the technique, first the number of delays is truncated to a finite number $\tilde{m}$ and second a rational approximation is applied to each $Q_m(s)$. The resulting delayed rational model is

$$H(s) \simeq \sum_{m=1}^{\tilde{m}} \frac{R_{m0} + \sum_{n=1}^{\tilde{n}} \frac{R_{mn}}{s-a_n}}{r_0 + \sum_{n=1}^{\tilde{n}} \frac{r_n}{s-a_n}} e^{-s\tau_m} \tag{2.41}$$

where $\tau_m \simeq T_m$ are suitable estimates of the dominant propagation delays, and a_n is a set of poles. It can be seen that when $\tilde{m} = 1, T_{\tilde{m}} = 0$, (4.21) becomes a normal rational approximation modeled like the regular vector fitting. The first stage of the identification of (2.41) is to identify estimates of dominant delay terms τ_m for the tabulated data. This is done with a Time-Frequency transform called the Gabor transform [34]. Details of how it is used to identify the delays will be presented in chapter 3. Once the set of dominant delays is known, what is left is the estimation of the coefficients R_{mn}, r_n of (2.41). This can be done in the same as the normal vector fitting is done, except in this case all the rational functions are fitted at once.

Chapter 3

Modeling Noisy Multiport Networks

In this chapter, the Instrumental Variable Vector Fitting method covered in section 2.1.2 is combined with the QR Decomposition technique of section 2.1.1 to efficiently macromodel large multiport networks characterized by noisy frequency domain data. The instrumental variable method is used to minimize the biasing effect of the least squares caused by the noise present in the data samples leading to more accurate solutions in fewer iterations. Furthermore, for the case of multiport networks described by common poles, the QR decomposition proposed in section 2.1.1 is used to decouple the equations which reduces the overall computation time and memory requirements for calculating the transfer functions. It is illustrated that the combination of the Instrumental Variable approach with QR decomposition leads to a lower errors and faster convergence of the overall macromodel when compared to using vector fitting with QR decomposition only (Fast vector fitting).

3.1 Problem formulation and review

As with it was done in section 2.1.2, consider tabulated frequency data from measurements

$$\hat{Y}_k^j = Y_k^j + \varepsilon_k^j$$

where Y_k^j is the k-th data sample of the j-th transfer function in the absence of noise and ε_k^j is the zero-mean complex random noise perturbing the k-th data sample of the j-th transfer function. The system of equations for the j-th transfer function can be expressed as

$$[\mathbf{X} \quad -\hat{\mathbf{H}}^j\mathbf{X}]\begin{bmatrix}\mathbf{C}^j\\ \mathbf{C}_p\end{bmatrix} = \hat{\mathbf{H}}^j\hat{\mathbf{1}} \tag{3.1}$$

where $\mathbf{C}^j$ corresponds to the unknown residues for the j-th transfer function, while $\mathbf{C}_p$ corresponds to the coefficients used to compute the unknown common poles shared among all the transfer functions describing the multiport network. A detailed description of the terms $\mathbf{X}$, $\hat{\mathbf{H}}^j$ and $\hat{\mathbf{1}}$ can be found in section 2.1.1 of the second chapter.

When trying to fit a multiport network with M transfer functions (i.e $j = 1, \ldots, M$) using common poles for all the transfer functions, the system of equations using (3.1) are coupled due to the shared coefficients of $\mathbf{C}_p$. To improve the efficiency of VF, a QR decomposition method is used to decouple the system of equations, which lead to simplified set of equations that depend only on $\mathbf{C}_p$ [17]. For this purpose, QR decomposition is applied to the left half side of (3.1) for each j-th transfer function

$$[\mathbf{X} \quad -\hat{\mathbf{H}}^j\mathbf{X}] = \hat{\mathbf{Q}}^j\begin{bmatrix}\hat{\mathbf{R}}_{11}^j & \hat{\mathbf{R}}_{12}^j\\ \mathbf{0} & \hat{\mathbf{R}}_{22}^j\end{bmatrix} \tag{3.2}$$

Next by combining the factorization of all the matrices, the following reduced set of equations is obtained, where only the coefficients $\mathbf{C}_p$ are the unknowns

$$\begin{bmatrix} \hat{\mathbf{R}}_{22}^{1} \\ \dots \\ \hat{\mathbf{R}}_{22}^{M} \end{bmatrix} \mathbf{C}_p = \begin{bmatrix} (\hat{\mathbf{Q}}^{1})^T \hat{\mathbf{H}}^{1} \hat{\mathbf{1}} \\ \dots \\ (\hat{\mathbf{Q}}^{M})^T \hat{\mathbf{H}}^{M} \hat{\mathbf{1}} \end{bmatrix} \tag{3.3}$$

Since (3.3) is an overdetermined system, its least squares solution for $\mathbf{C}_p$ is expressed as

$$(\sum_{j=1}^{M} [(\hat{\mathbf{R}}_{22}^{j})^T \hat{\mathbf{R}}_{22}^{j}]) \mathbf{C}_p = \sum_{j=1}^{M} (\hat{\mathbf{Q}}^{j} \hat{\mathbf{R}}_{22}^{j})^T \hat{\mathbf{H}}^{j} \hat{\mathbf{1}} \tag{3.4}$$

In order to examine how the noise biases the least squares solution of (3.4), the terms $\hat{\mathbf{H}}^{j}$, $\hat{\mathbf{R}}_{22}^{j}$ and $\hat{\mathbf{Q}}^{j} \hat{\mathbf{R}}_{22}^{j}$, are written as

$$\begin{aligned} \hat{\mathbf{H}}^{j} &= \mathbf{H}^{j} + \mathbf{N}_{h}^{j} \\ \hat{\mathbf{R}}_{22}^{j} &= \mathbf{R}_{22}^{j} + \mathbf{N}_{r\varepsilon}^{j} \\ \hat{\mathbf{Q}}^{j} \hat{\mathbf{R}}_{22}^{j} &= \mathbf{Q}^{j} \mathbf{R}_{22}^{j} + \mathbf{N}_{q\varepsilon}^{j} \end{aligned} \tag{3.5}$$

where $\mathbf{H}^{j}$, $\mathbf{R}_{22}^{j}$ and $\mathbf{Q}^{j}$ terms are derived from Y_k^j, if the data samples had no noise, and $\mathbf{N}_{h}^{j}$, $\mathbf{N}_{r\varepsilon}^{j}$ and $\mathbf{N}_{q\varepsilon}^{j}$ are due to the error ε_k^j in the data samples. Since, the noisy terms $\mathbf{N}_{h}^{j}$, $\mathbf{N}_{r\varepsilon}^{j}$ and $\mathbf{N}_{q\varepsilon}^{j}$ are from the same data values, the coefficients for the k-th data element are correlated. Using (3.5), the products of the least squares solutions for the j-th transfer function become

$$
\begin{aligned}
(\hat{\mathbf{Q}}^j\hat{\mathbf{R}}^j_{22})^T\hat{\mathbf{H}}^j = \quad & (\mathbf{Q}^j\mathbf{R}^j_{22})^T\mathbf{H}^j + (\mathbf{Q}^j\mathbf{R}^j_{22})^T\mathbf{N}^j_h \\
& + (\mathbf{N}^j_{q\varepsilon})^T\mathbf{H}^j + (\mathbf{N}^j_{q\varepsilon})^T\mathbf{N}^j_h \\
(\hat{\mathbf{R}}^j_{22})^T\hat{\mathbf{R}}^j_{22} = \quad & (\mathbf{R}^j_{22})^T\mathbf{R}^j_{22} + (\mathbf{R}^j_{22})^T\mathbf{N}^j_{r\varepsilon} \\
& + (\mathbf{N}^j_{r\varepsilon})^T\mathbf{R}^j_{22} + (\mathbf{N}^j_{r\varepsilon})^T\mathbf{N}^j_{r\varepsilon}
\end{aligned} \tag{3.6}
$$

Note that $(\mathbf{Q}^j\mathbf{R}^j_{22})^T\mathbf{H}^j$ and $(\mathbf{R}^j_{22})^T\mathbf{R}^j_{22}$ of (3.6) are the matrices obtained in the absence of noise. Since it is assumed that the biasing of the noise is zero (i.e. expected mean value, $E[\varepsilon] = 0$), the expected values of the second and third terms on the right-hand side of (3.6) are also zero. Thus, these terms do not statistically bias the results of the least squares approximation. For the fourth terms on the right hand side of (3.6), their expected values do not equal to zero, since they are the product of two correlated matrices. Therefore, it is due to the biasing effect of $(\mathbf{N}^j_{q\varepsilon})^T\mathbf{N}^j_h$ and $(\mathbf{N}^j_{r\varepsilon})^T\mathbf{N}^j_{r\varepsilon}$, that (3.4) fails to capture actual poles of the system in the presence of zero-mean noise.

For the implementation of the relaxed VF algorithm (RVF) [15], it can be shown that the noise also bias the least squares solution [25]. However, because the right hand side of (3.3) does not include $(\hat{\mathbf{Q}}^j)^T\hat{\mathbf{H}}^j\hat{\mathbf{1}}$ when implementing RVF, only the $(\hat{\mathbf{R}}^j_{22})^T\hat{\mathbf{R}}^j_{22}$ terms end up biasing the solution. This contributes to RVF being able to obtain more accurate results when compared to VF.

3.2 Proposed Algorithm

In order to minimize the biasing effect of the noise, the instrumental variable [22] method formulates the least squares solution as

$$(\sum_{j=1}^{M}[(\tilde{\mathbf{R}}_{IV22}^{j})^{T}\hat{\mathbf{R}}_{22}^{j}])\mathbf{C}_{p} = \sum_{j=1}^{M}(\tilde{\mathbf{Q}}_{IV}^{j}\tilde{\mathbf{R}}_{IV22}^{j})^{T}\hat{\mathbf{H}}^{j}\hat{\mathbf{1}} \tag{3.7}$$

where the terms $\tilde{\mathbf{Q}}_{IV}^{j}$ and $\tilde{\mathbf{R}}_{IV22}^{j}$ come from a QR decomposition performed on the j-th transfer function of the system $\mathbf{X} - \hat{\mathbf{\Psi}}^{j}\mathbf{X}$. The term $\hat{\mathbf{\Psi}}^{j}$ is constructed from different data estimates, $\hat{Y}_{IVk}^{j} = Y_{k}^{j} + \eta_{k}^{j}$, where η_{k}^{j} is the error of approximation of Y_{k}^{j} and is assumed to be a zero mean noise uncorrelated with ε_{k}^{j}. Since the terms $\tilde{\mathbf{R}}_{IV22}^{j}$ and $\tilde{\mathbf{Q}}_{IV}^{j}\tilde{\mathbf{R}}_{IV22}^{j}$ are formulated from a different set of noisy data samples, they can be expressed similar to (3.5), with $\mathbf{N}_{r\eta}^{j}$ and $\mathbf{N}_{q\eta}^{j}$ due to the error η_{k}^{j} in the data samples.

Next, to investigate the biasing effect of the noise, the product terms of (3.7) for the j-th transfer function are expressed as

$$\begin{aligned}
(\tilde{\mathbf{Q}}_{IV}^{j}\tilde{\mathbf{R}}_{IV22}^{j})^{T}\hat{\mathbf{H}}^{j} = \quad & (\mathbf{Q}^{j}\mathbf{R}_{22}^{j})^{T}\mathbf{H}^{j} + (\mathbf{Q}^{j}\mathbf{R}_{22}^{j})^{T}\mathbf{N}_{h}^{j} \\
& + (\mathbf{N}_{q\eta}^{j})^{T}\mathbf{H}^{j} + (\mathbf{N}_{q\eta}^{j})^{T}\mathbf{N}_{h}^{j} \\
(\tilde{\mathbf{R}}_{IV22}^{j})^{T}\hat{\mathbf{R}}_{22}^{j} = \quad & (\mathbf{R}_{22}^{j})^{T}\mathbf{R}_{22}^{j} + (\mathbf{R}_{22}^{j})^{T}\mathbf{N}_{r\varepsilon}^{j} \\
& + (\mathbf{N}_{r\eta}^{j})^{T}\mathbf{R}_{22}^{j} + (\mathbf{N}_{r\eta}^{j})^{T}\mathbf{N}_{r\varepsilon}^{j}
\end{aligned} \tag{3.8}$$

Since it is assumed that the biasing effect of ε and η are zero, the expected mean values of the second and third terms of the right hand side of (3.8) are zero. Furthermore, since it is assumed that ε and η are uncorrelated to each other (i.e. $E[\eta_{k}^{j}\varepsilon_{k}^{j}] = 0$), the expected mean values of the fourth terms are also zero. As a result, by using a different approximation uncorrelated to the original data, the instrumental variable provides a statistically unbiased result to the least squares solution of (3.7).

In this work, the previous rational approximation of the VF method is used to create the data samples of $\hat{Y}^j_{IVk} = Y^j_k + \eta^j_k$, similar to the approach described in [25]. Since the errors of the previous rational approximation is less correlated to the errors of the original data, it is demonstrated in Section IV that this yields better rational approximations without significantly increasing the complexity of the VF method.

3.3 Numerical Example

3.3.1 Synthetic Transfer Function

As a proof of concept, a transfer function (TF) with known poles and residues described in Table 3.1 with added noise is approximated. The TF is contaminated with a white gaussian noise with a signal to noise ration (SNR) of 35 dB. the TF is sampled using 2000 evenly spaced point that range from 0 to 10 GHz. A rational approximation is performed using the classic VF algorithm and the proposed method, the resulting approximations are compared with the original noiseless TF in Fig. 3.1. It can be seen that contrary to the VF algorithm, which cannot capture all the poles, the VF-IVQR method captures all 16 poles of the original TF. A second rational approximation is also performed with a SNR of 25 dB. Instead of using the classic VF, the algorithms are performed using the relaxed version of VF [15]. Once again, as it can be seen in Fig. 3.4, the RVF-QR fails to capture all the poles as accurately as RVF-IVQR. Along with the TF approximations, RMS Error versus the number of iterations plots are also provided for VF and RVF in Fig. 3.2 and Fig. 3.3 respectively. Table 3.2 provides a comparison between the RVF-QR and the RVF-QRIV poles, the proposed method has very low error when compared to the original poles of the TF, compared to the RVF-QR which has a relatively high percentage error in one of the pole, this error gets even greater when using the VF without the relaxed constraint.

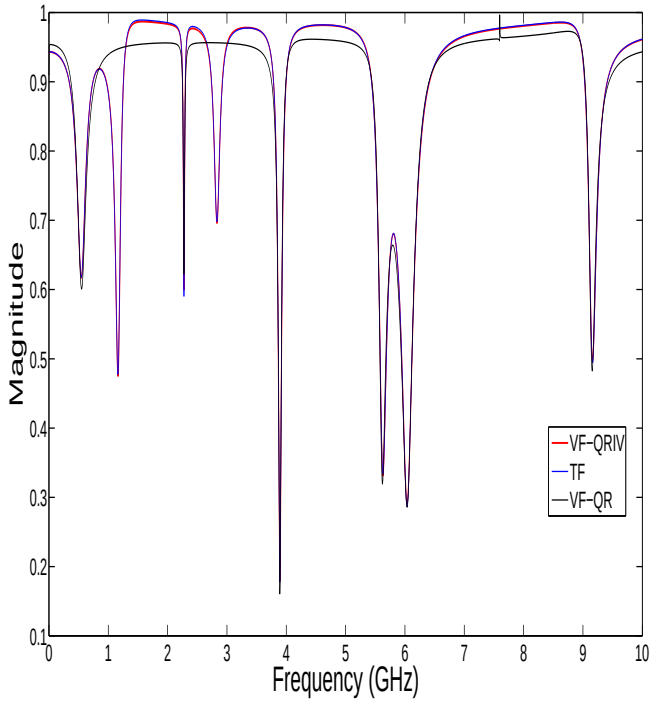


Figure 3.1: Rational approximations of the 35 dB SNR data.

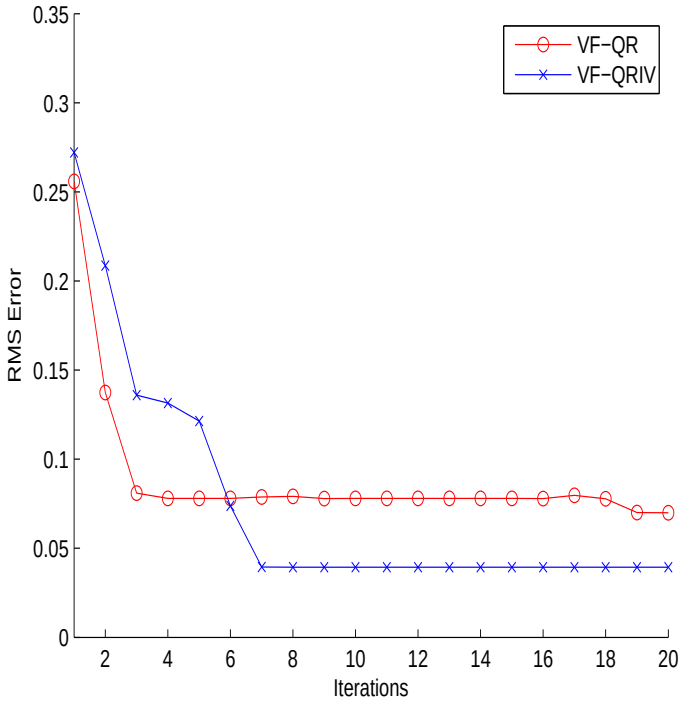


Figure 3.2: RMS Error versus iteration count for SNR = 35dB.

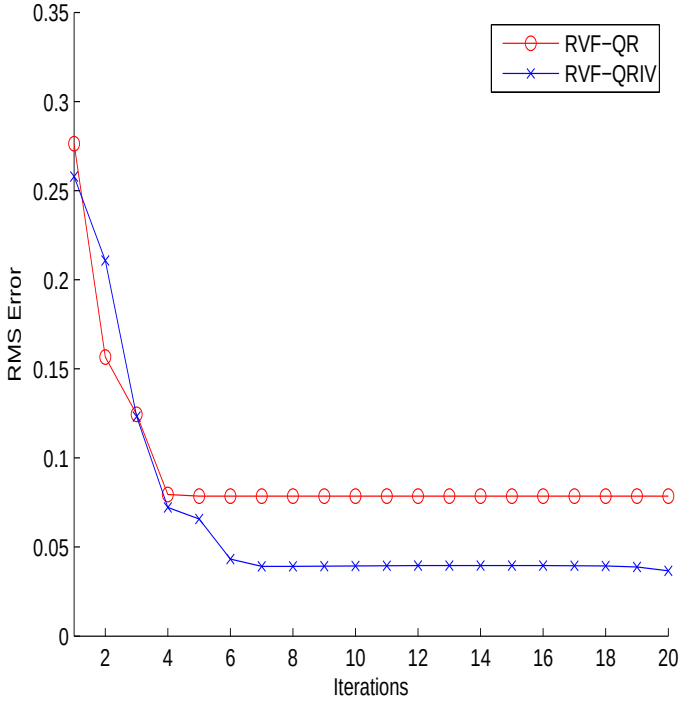


Figure 3.3: RMS Error versus iteration count for SNR = 25dB.

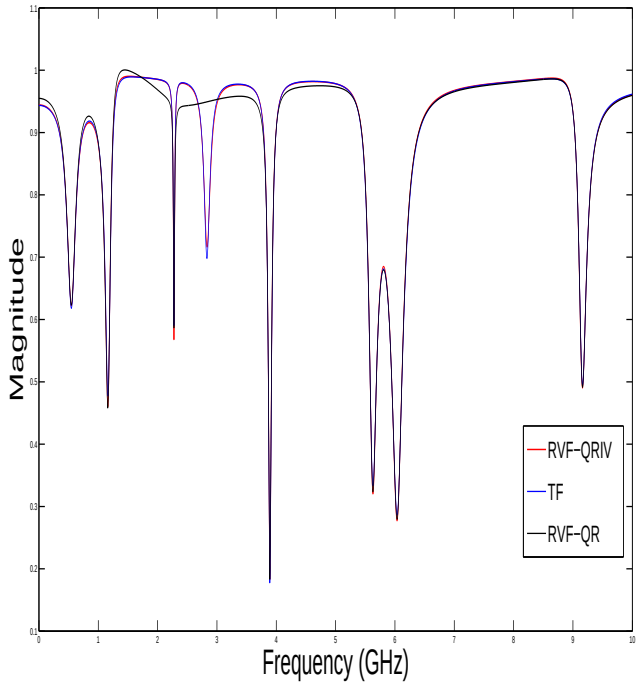


Figure 3.4: Rational approximations of the 25 dB SNR data.

Table 3.1: Poles and Residues of the TF for Example 1

Poles (GHz)	Residues (GHz)
$d = 0.98$	
$-0.6132 \pm j3.4551$	$-0.9877 \mp j0.0809$
$-0.3940 \pm j7.3758$	$-0.2067 \mp j0.0131$
$-0.0880 \pm j14.3024$	$-0.1382 \mp j0.0145$
$-0.4097 \pm j17.7864$	$-0.1182 \mp j0.0166$
$-0.2991 \pm j24.4622$	$-0.2426 \mp j0.0145$
$-0.6447 \pm j35.2669$	$-0.4043 \mp j0.0297$
$-1.0135 \pm j37.9655$	$-0.6787 \mp j0.1465$
$-0.5711 \pm j57.4748$	$-0.2626 \mp j0.1037$

Table 3.2: Comparison of poles RVF versus RVF-QRIV after the twentieth iteration

RVF-QR		RVF-QRIV	
Poles (GHz)	Error %	Poles (GHz)	Error %
$-0.6203 \pm j3.4514$	0.067	$-0.6143 \pm j3.4484$	0.18
$-0.4017 \pm j7.3789$	0.05	$-0.3916 \pm j7.3798$	0.05
$-0.0821 \pm j14.301$	0.01	$-0.08679 \pm j14.300$	0.02
$-2.8962 \pm j16.175$	7.65	$-0.4412 \pm j17.771$	0.08
$-0.2862 \pm j24.468$	0.02	$-0.2932 \pm j24.469$	0.03
$-0.6718 \pm j35.256$	0.03	$-0.6609 \pm j35.265$	0.003
$-1.0025 \pm j37.798$	0.03	$-1.0131 \pm j37.798$	0.03
$-0.6130 \pm j57.473$	0.002	$-0.6080 \pm j57.472$	0.005

3.3.2 Four Port Network

A four port network of a two differential pairs of Strada-Whisper connectors is characterized in terms of the S-parameters measured using a vector network analyzer. A circuit description of the four port network is provided in example 3 of [25]. Since this is a four port symmetric network, ten transfer functions are fitted simultaneously using 100 common poles. The data is fitted using VF [9], instrumental variable VF (VF-IV), relaxed VF (RVF) [15] and instrumental variable relaxed VF (RVF-IV), where all these methods are implemented using the

QR decomposition algorithm. The initial guess of the poles is distributed evenly among the imaginary axis as complex conjugate poles between 0 to 12 GHz. Fig. 3.5 shows normalized $\mathcal{H}_2$-norm [18] for 10 iterations which measures the error in the magnitude of all the entries of the S-parameter matrix. The instrumental variable was created after the first iteration for the IV algorithms. A sample of the rational approximation for S_{13} for RVF and RVF-IV, is shown in Fig. 2.

It should be noted that the difference between the macromodels obtained from VF, VF-IV, RVF and RVF-IV is dependent on the level of noise in the data. For high signal-to-noise ratio (SNR), all methods will give similar results. In the proposed VF-IV and RVF-IV, the biasing effect due to (8) is reduced since the noise matrices (using the instrumental variable approach) are less correlated. For this example, RVF, VF-IV and RVF-IV were able to get capture the transfer functions. Nonetheless, VF-IV and RVF-IV converged faster and achieved lower error (Fig. 3.5). As the noise of data sample increases, the instrumental variable will tend to outperform the VF and RVF due to the reduced biasing of the least squares solution. To illustrate this point, the noise obtained by $S_{ij}-S_{ji}$ is multiplied by five and ten and added to S_{ij}. Table 3.3 compares the error after the tenth iteration for original and modified data. In addition, zoomed regions of S_{13} (for noise multiplied by ten) are shown in Fig. 3 to show that the RVF-IV performs better than RVF. Since all four methods rely on QR decomposition to decouple the least squares equations, their CPU times were close to 9 seconds using an Intel Xeon dual-processor (3.16 GHz) and 4 GB of RAM. In comparison to [25], which does not decouple the equations of multiport network due to the common poles, the CPU time was 311 seconds.

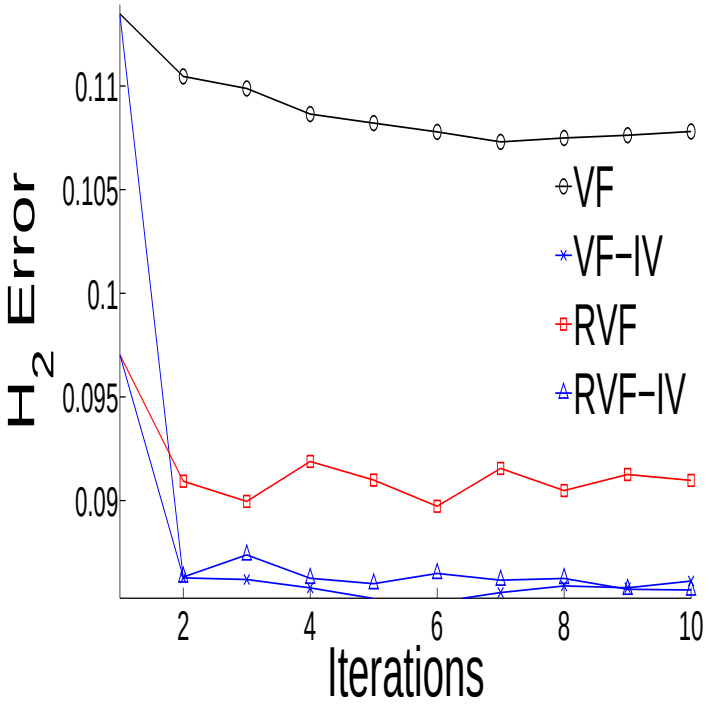


Figure 3.5: Normalized $\mathcal{H}_2$-norm versus number of iterations.

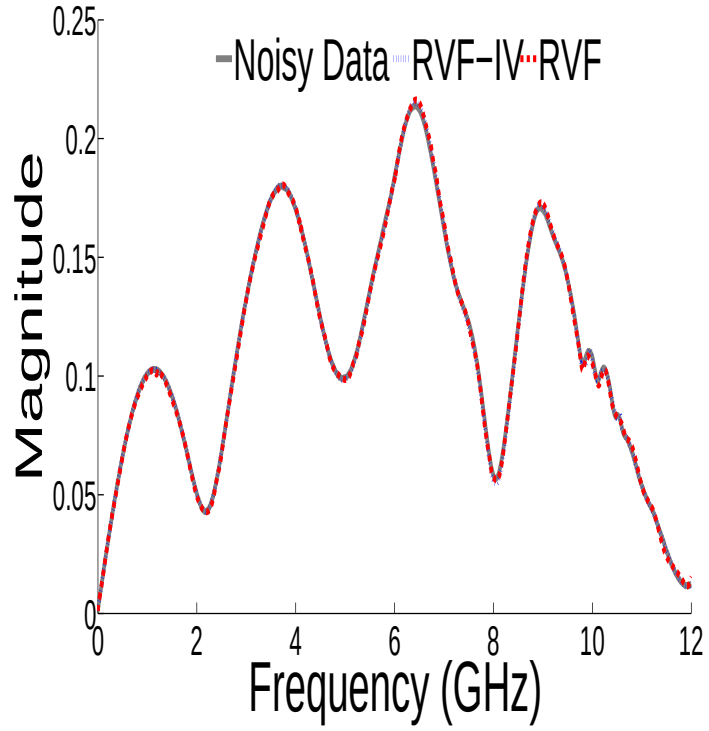
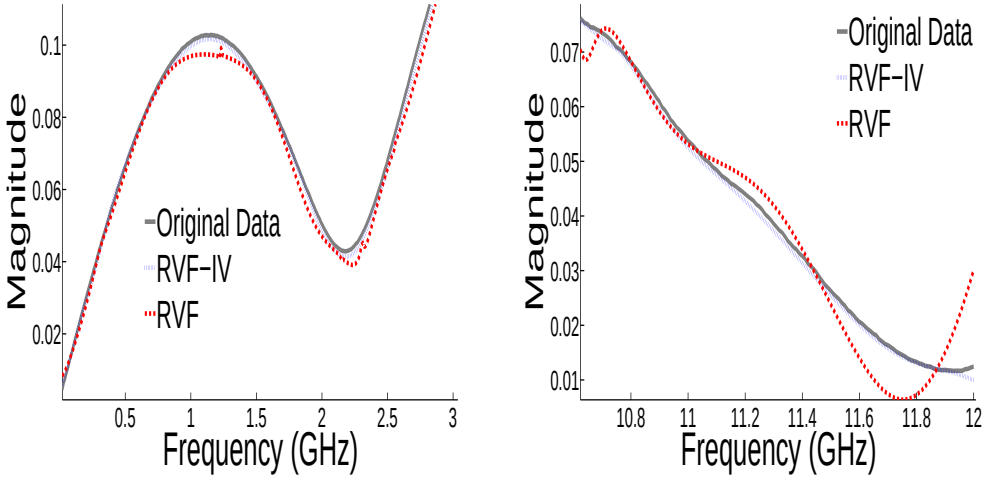


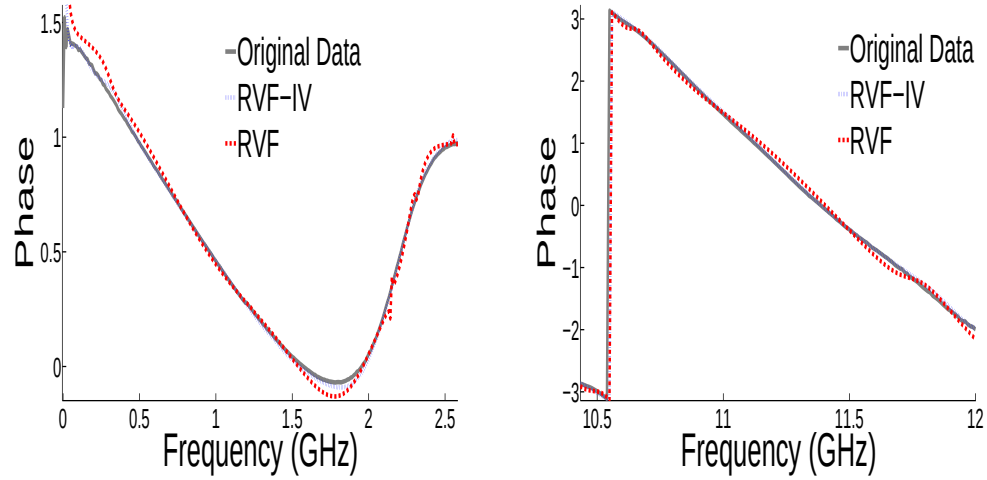
Figure 3.6: Magnitude of S_{13}.

Table 3.3: Error comparisons for original and modified data

	$\mathcal{H}_2$-norm		
Method	Original	5x	10x
VF	0.108	0.120	0.134
RVF	0.091	0.095	0.100
VF-IV	0.086	0.086	0.086
RVF-IV	0.086	0.086	0.086



(a) Magnitude of S_{13} Zoomed.



(b) Phase of S_{13} Zoomed.

Figure 3.7: Rational approximation of S_{13} (added noise) using RVF and RVF-IV.

Chapter 4

Delay Extraction Loewner Method

4.1 Introduction

Methodologies to obtain delayed rational functions have been proposed in [30, 35] using the VF approach for the attenuation losses approximations. However, these delayed rational function techniques have not been extended to the LM approach. In this chapter, delayed rational approximations from tabulated frequency data are derived using the LM method, based on the concepts developed in [36]. The method uses explicit delay extraction to extract propagation delays estimates and partitions the data into single delay regions using a time-frequency transform. A new approach to refine the delay estimates for each partitioned region is also proposed using the LM method. Once the best delay estimates are identified, the LM method is used to obtain rational approximations for each region. The developed delay extraction LM algorithm is also applied to multi-port networks. Numerical examples are presented to illustrate efficiency of the proposed method compared to traditional Loewner where the delays of the transfer function are not extracted.

4.2 Macormodels with Delays and Review of General Time-Frequency Decomposition

4.2.1 Theoretical Motivation

The main objective of the proposed method is to produce a delayed rational function of the following form:

$$H_{ij}(s) = \sum_{m=1}^{M} H_{ij}^{(m)}(s)\mathrm{e}^{-sT_m} \tag{4.1}$$

where T_m is the m^{th} propagation delay and $H_{ij}^{(m)}(s)$ is the delay free rational approximation corresponding to m^{th} delay. In practice, it is possible to approximate a long interconnect without the extraction of the delay terms, however this generally results in a very high number of poles, which makes the transient analysis computationally intensive. By extracting the delay, the attenuation losses can be approximated by low order rational function [13,14,29–31,35,37]. The next section presents an overview of estimating the delays when dealing with electrically long distributed networks characterized by measured or simulated data.

4.2.2 Time-Frequency Decomposition

The delay extraction is done using the concept of the time-frequency decomposition transforms. A time-frequency transform relates $H_{ij}(s)$ to $F_{ij}(\omega,\tau)$ with the following relation:

$$F_{ij}(\omega,\tau) = \int_{-\infty}^{\infty} H_{ij}(\zeta)W(\zeta-\omega)\mathrm{e}^{j\zeta\tau}\mathrm{d}\zeta \tag{4.2}$$

where $W(\zeta - \omega)$ is a window centred at $\zeta = \omega$ of specific width L [34, 38] It is observed from (4.2) that if $W = 1$, then the equation becomes the standard definition of the Inverse Fourier Transform (IFT). Therefore the time-frequency transform can be thought of as an IFT of $H_{ij}(s)$, that only retains the frequency components in the frequency band of the filtering window W. In this work, the Gabor transform [34, 38] is used, since it provides optimal support in both the time and frequency domain. The energy contents of $F_{ij}(\omega, \tau)$ over time is obtained by,

$$\eta_{ij}(t) = \int_{-\infty}^{\infty} |F_{ij}(\omega, \tau)|^2 \mathrm{d}\omega \tag{4.3}$$

where the propagation delays can be identified as the local maxima of the $\eta_{ij}(t)$ function [30].

The inverse of (4.2) is defined as [34, 38]:

$$H_{ij}(\zeta) = \frac{1}{2\pi} \iint_{-\infty}^{\infty} F_{ij}(\omega, \tau) W(\zeta - \omega) \mathrm{e}^{-j\omega\tau} \mathrm{d}\omega \mathrm{d}\tau \tag{4.4}$$

Using (4.3), the reconstruction of $H_{ij}(\zeta)$ can be done by splitting the time-frequency plane into separate regions Ω_m and performing the integral (4.4) over each region as follows [29, 30]

$$\begin{aligned} H_{ij}(\zeta) &= \sum_k \widetilde{H}_{ij}^{(m)}(\zeta) \\ \widetilde{H}_{ij}^{(m)}(\zeta) &= \frac{1}{2\pi} \iint_{\Omega_m} F_{ij}(\omega, \tau) W(\zeta - \omega) \mathrm{e}^{-j\omega\tau} \mathrm{d}\omega \mathrm{d}\tau \\ \bigcup_m \Omega_m &= \Re^2 \end{aligned} \tag{4.5}$$

The summation of each integral of (4.5) leads to the reconstruction of $H_{ij}(\zeta)$. The time-frequency transform discussed above, provides a way to extract delays from electrically long

distributed networks characterized by measured or simulated data.

4.3 Proposed Algorithm

Once the delays are determined from the measured data, the proposed work approximates the attenuation losses corresponding to each delay using a Loewner matrix approach. The steps involved are identifying of most significant propagation delays, partitioning of the time-frequency plane in regions and performing rational approximation of the attenuation losses using Loewner matrix.

4.3.1 Estimation of Propagation Delays and Partitioning Regions

The first step of the proposed algorithm is to estimate the propagation delays, given the tabulated data H_{ij}. The time-frequency representation $F_{ij}(\omega, \tau)$ is computed using (4.2). Once the time-frequency plane is obtained, evaluating the energy content over time $\eta_{ij}(t)$ using (4.3) provides estimates of the propagation delays, as the time values of the local maxima [30].

In order to extract the most relevant delays, all delay terms with relative energy contributions below a user-chosen tolerance ε are not taken into account

$$\frac{\hat{n}_{ij}^{(k)}}{\sum_{k} \hat{n}_{ij}^{(k)}} < \varepsilon \tag{4.6}$$

where $\hat{n}_{ij}^{(k)}$ is the energy content of a local delay evaluated as [30]

$$\hat{n}_{ij}^{(k)} = \frac{1}{2\pi} \int_{\tau_k - 1}^{\tau_k} \eta_{ij}(\tau) d\tau \tag{4.7}$$

where τ_{k-1} and τ_k correspond to local minimums between the kth local maximum of the function $\eta_{ij}(\tau)$. The value of ε is problem dependent and is chosen such that the energy contribution of the neglected delays does not significantly affect the accuracy of the model [30, 31].

Once the estimated delays are identified, the next step is to split the time-frequency plane in such a way as to get delay regions. The method used to split the plane is the same as the one proposed in [30, 31]. The partitioning for the (ω, τ) plane into Ω_m is done by choosing a point t_k between adjacent delays T_k and T_{k+1}, where the value of the energy content at that point is lower than a predefined value δ

$$\eta_{ij}(\tau = t_k) < \delta \tag{4.8}$$

Using (4.7), regions Ω_m are defined to be regions between two adjacent minima t_k and t_{k+1}, expressed as follows:

$$\Omega_m \in \{(\omega, \tau) : \ 0 \leq \omega \leq 2\pi F_{max}, \ t_k \leq \tau \leq t_{k+1}\} \tag{4.9}$$

Depending on the value of the estimated delay computed using the time-frequency transform, the order of the rational approximation that is chosen is not always optimal, therefore an extra step is also needed to optimize the delay. This step is important since an optimized delay can

reduce the order of the rational function of each region Ω_i. In this work, the delay optimization is performed using a similar method to what is done in [37], however, instead of using the error of the rational approximation for different delay values, the optimized delay is determined by considering the normalized singular values drop obtained from the SVD of the Loewner matrix. Details of how to select the optimized delay which leads to a low order rational approximation will be provided once LM method is presented.

4.3.2 Estimating Attenuation Losses $H_{ij}^{(m)}(s)$

Once the Ω_m regions are determined, the last step involves computing the attenuation losses rational function corresponding to each region. There are two cases that can arise when estimating attenuation losses; regions where there is only one identified delay peak in Ω_m and regions where there are more than one identified delay peak present in Ω_m.

For the case where there is only one identified delay peak, the goal is to evaluate

$$\tilde{H}_{ij}^{(m)}(s) \approx H_{ij}^{(m)}(s)e^{-sT_m} \tag{4.10}$$

where T_m is the known extracted delay and $\tilde{H}_{ij}^{(m)}(s)$ is obtained using (4.5).

To get the rational approximation $H_{ij}^{(m)}(s)$ for each region, the frequency domain data is expressed as

$$\{s_k, \tilde{H}_{ij}^{(m)}(s_k)e^{+s_kT_m}\} \tag{4.11}$$

where $k = 1, \ldots, K$, and K is the number of data points.

The Loewner Matrix method [18] seeks to macromodel the attenuation losses $H_{ij}^{(m)}(s) \approx \tilde{H}_{ij}^{(m)}(s)e^{+sT_m}$ as

$$H_{ij}^{(m)}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} + s\mathbf{Y}^{\infty} \tag{4.12}$$

where $\mathbf{A}, \mathbf{E} \in \mathbb{R}^{n\times n}, \mathbf{B} \in \mathbb{R}^{n\times 1}, \mathbf{C} \in \mathbb{R}^{1\times n}, \mathbf{D} \in \mathbb{R}, \mathbf{Y}^{\infty} \in \mathbb{R}$ describe the system of order n. The descriptor state space matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and $\mathbf{E}$ are obtain as follows.

The method to obtain these state space matrices are done using what is described in section 2.2 of the second chapter.

As mentioned in Section III.A, to get a low order n for each delay region Ω_m, an optimization step is performed. Different delay estimates over a small range centered around the estimated value T_m are used in (4.10). The optimized delay is determined by considering the normalized singular values $\mathbf{\Sigma}$ drop obtained from (2.31) that gives the lowest order n.

For the case when more than one identified delay exist in the region Ω_m, the goal is to evaluate

$$\begin{aligned} \tilde{H}_{ij}^{(m)}(s) &\approx H_{ij}^{(m)}(s)e^{-sT_m} \\ H_{ij}^{(m)}(s) &= \sum_{n=1}^{N} H_{ij}^{(n,m)}(s)e^{-s(T_{n,m}-T_m)} \end{aligned} \tag{4.13}$$

where N is the number of delays clustered together within a single region Ω_m. Due to the close proximity of the delays $T_{n,m}$, the attenuation losses $H_{ij}^{(n,m)}(s)$ cannot be evaluated individually,

however since the delays within the same region lie very close to each other, each $e^{-s(T_{n,m}-T_m)}$ will have a very small contribution to the approximation $H_{ij}^{(m)}(s)$. Therefore, by choosing the first value of the delay as $T_m = T_{1,m}$, $H_{ij}^{(m)}(s)$ can still be approximated with a low order rational function using the LM method. A summary of the steps of the overall proposed method can be found in algorithm 1.

Step 1: Obtain $H_{ij}(s)$ frequency tabulated data from EM simulation or measurements, for frequency points $s = \{s_1, \ldots, s_K\}$.
Step 2: Compute the Time-Frequency transform F_{ij} using (4.2).
Step 3: Compute energy function η_{ij} using F_{ij} from step 2, and identify propagation delays as local maxima. Select only the delay terms that satisfy the conditions of (4.6), (4.7), and ε to discard the delay terms that don't significantly affect the accuracy of the model.
Step 4: Partition the Time-Frequency plane, using (4.8),(4.9), and δ.
Step 5: Refine the delay value obtained from (3) for each region using (4.10) with different delays over a small range centered around the estimated value T_m. The optimized delay is determined by considering the normalized singular values drop that gives the lowest order n.
Step 6: Apply the LM algorithm on each delay-free data computed as

$$H_{Dij}^{(m)}(s) = H_{ij}^{(m)}(s)e^{+sT_m}$$

for $m = 0, \ldots, M-1$ regions.
Step 7: Obtain the final low-order model (4.1) as a sum of the M terms computed in step 6.

Algorithm 1: Proposed Delay Extraction LM method

4.4 Numerical Examples

In this section, three examples are presented to validate the proposed method. The work was done using MATLAB [39] The order of the macromodels are shown with and without delay extraction.

4.4.1 Synthetic Transfer Function

To validate the proposed method, a transfer function (TF) with known poles, residues and delays (described in Table 4.1) is approximated. The delay extraction is performed using a tolerance value of $\varepsilon = 1e-4$ in (4.6). The original estimates for the delays were $\{20.0, 70.3\}$ ns. Then, following the procedure of step 5 in algorithm 1, the estimates were refined to obtain values of $\{20.0, 70.0\}$ ns for delay region one and two respectively. An illustration of the normalized singular values of the second region using the original and optimized delays can be seen in Fig 4.1. The optimization chose 70 ns for the delay of the second region since it resulted in the lowest number of singular values above the selected threshold. Fig 4.2, shows the plot of the normalized singular values for the regions of the delayed rational approximations and for the LM without delay extraction. Since the most significant drop for the delayed rational function happens at the beginning, an order of 4 to 5 poles is enough to accurately approximate each region. For the LM without the delay, the significant drop in the singular values occurs in the 1400 poles range. Fig 4.3 shows the real part of the TF and is compared with the proposed method (4 poles for each region), LM with 1410 poles (corresponding to a threshold value of 1e-5 (the red dashed line in Fig 4.2(b))). In addition, an extra LM with 600 poles (corresponding to a threshold value of 1e-1 (the black dashed line in Fig 4.2(b))) with the same threshold as the one used for the proposed method was also computed. Both the proposed and the LM with 1410 poles show accurate results, as opposed to LM with 600 poles where it is less accurate. Using the proposed approach resulted in a macromodel with 8 poles, a significant improvement compared to LM without delay extraction. Table 4.2 shows the poles and residues calculated and the percentage error with respect to the real and imaginary parts. It should be noted, that VF can also approximate each delay region with the same order as the proposed method. However, with VF there is no direct way to determine the order of each region beforehand and requires fitting the data with varying order to obtain the best approximation.

Table 4.1: Poles, Residues and Delays of the TF

Poles (GHz)	Residues (GHz)
$Delay = 20ns$	
$-0.6132 \pm j3.4551$	$-0.9877 \mp j0.0809$
$-0.3940 \pm j7.3758$	$-0.2067 \mp j0.0131$
$Delay = 70ns$	
$-1.0135 \pm j37.9655$	$-0.6787 \mp j0.1465$
$-0.5711 \pm j57.4748$	$-0.2626 \mp j0.1037$

Table 4.2: Calculated poles and residues compared with theoretical values

Theoretical	Calculated	Error
Poles (GHz)		
$-0.6132 \pm j3.4551$	$-0.6133 \pm j3.4534$	0.05%
$-0.3940 \pm j7.3758$	$-0.3929 \pm j7.3748$	0.01%
$-1.0135 \pm j37.9655$	$-1.0131 \pm j37.9660$	0.00%
$-0.5711 \pm j57.4748$	$-0.5704 \pm j57.4760$	0.00%
Residues (GHz)		
$-0.9877 \mp j0.0809$	$-0.9880 \mp j0.0838$	0.06%
$-0.2067 \mp j0.0131$	$-0.2060 \mp j0.0136$	0.33%
$-0.6787 \mp j0.1465$	$-0.6783 \mp j0.1465$	0.05%
$-0.2626 \mp j0.1037$	$-0.2620 \mp j0.1035$	0.21%

4.4.2 PCB board interconnect data

The second example corresponds to a four-port network of an 8 inch PCB interconnect (courtesy of Broadcom). The network is characterized by actual measurements up to 20 GHz of the S-parameters using a vector network analyzer.

The time-frequency transform and the energy functions of all the S-parameters are evaluated

Table 4.3: Time Axis Partitioning of (ω, τ) Plane of Example 2

Regions	$\mathbf{S_{11}}$	$\mathbf{S_{12}}$	$\mathbf{S_{13}}$	$\mathbf{S_{14}}$	$\mathbf{S_{22}}$	$\mathbf{S_{23}}$	$\mathbf{S_{24}}$	$\mathbf{S_{33}}$	$\mathbf{S_{34}}$	$\mathbf{S_{44}}$
$\mathbf{\Omega_1}$	[0.00 2.90]	[0.00 41.50]	[0.00 2.1]	[0.00 2.45]	[0.00 11.00]	[0.00 2.40]	[0.00 1.65]	—	[0.00 6.45]	—
$\mathbf{\Omega_2}$	[2.90 4.95]	—	[2.1 5.00]	—	[2.40 5.55]	—	[1.65 4.80]	[2.10 6.00]	—	[2.30 5.30]

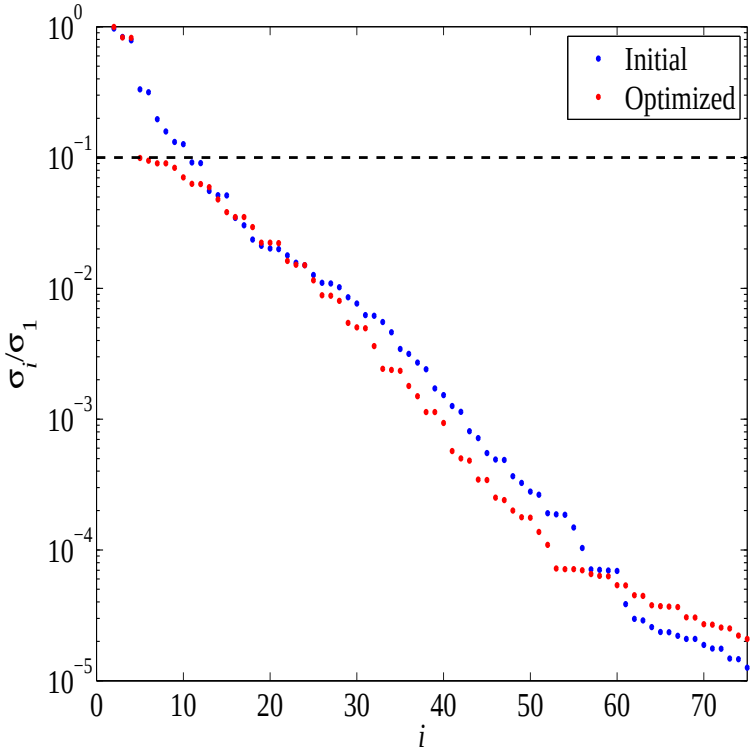


Figure 4.1: Normalized singular values of region 2.

Table 4.4: Estimated delays versus optimized delays for Example 2 (Times in *ns*)

Regions		S_{11}	S_{12}	S_{13}	S_{14}	S_{22}	S_{23}	S_{24}	S_{33}	S_{34}	S_{44}
Ω_1	**Est.**	—	1.35	0.220	3.386	—	1.350	0.200	—	1.300	—
	Opt.	—	1.285	0.198	3.029	—	1.264	0.184	—	1.284	—
Ω_2	**Est.**	2.403	—	2.508	—	2.403	—	2.550	2.560	—	2.545
	Opt.	2.225	—	2.390	—	2.225	—	2.315	2.230	—	2.220

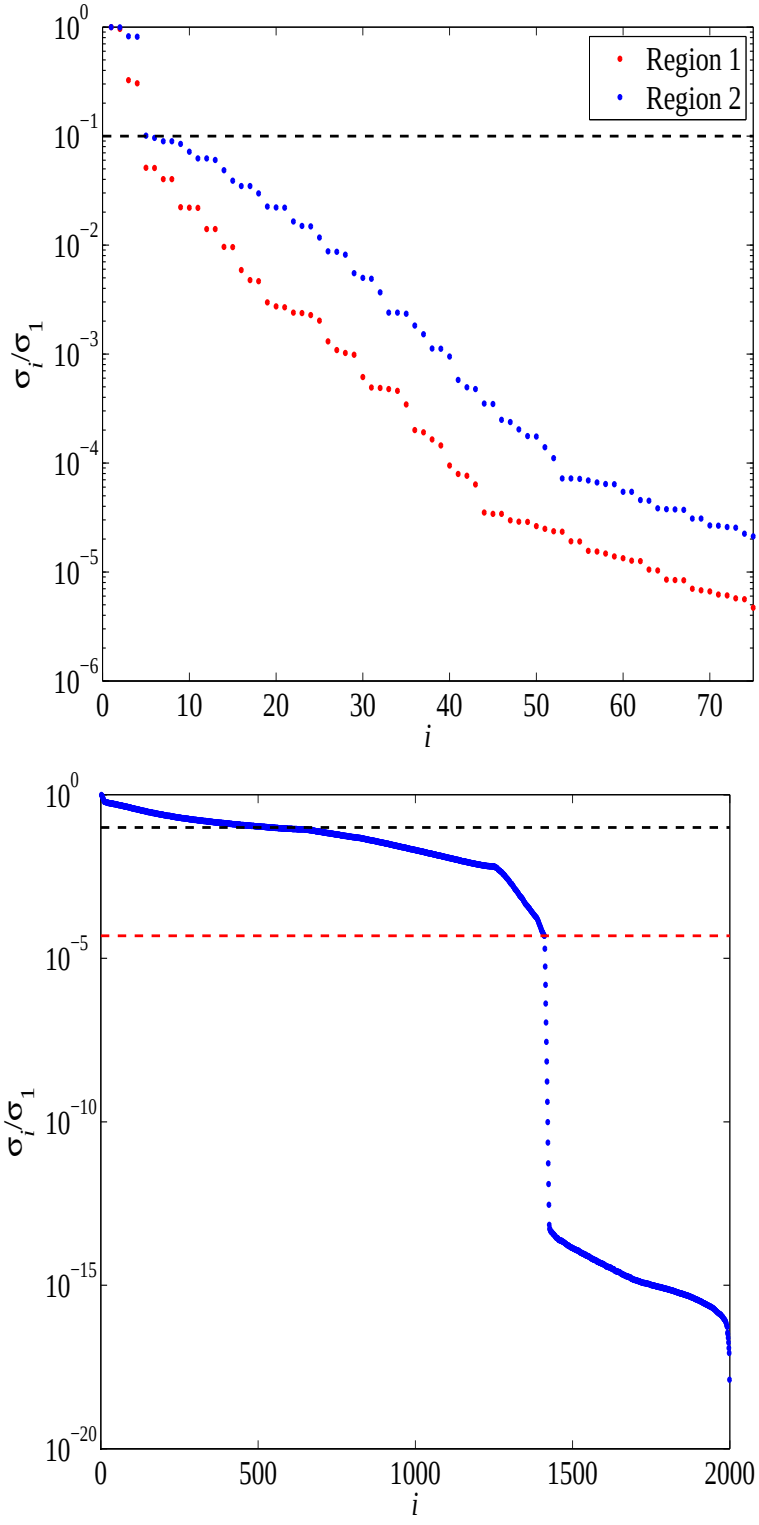


Figure 4.2: Normalized singular values (Top) delay LM (Bottom) original LM.

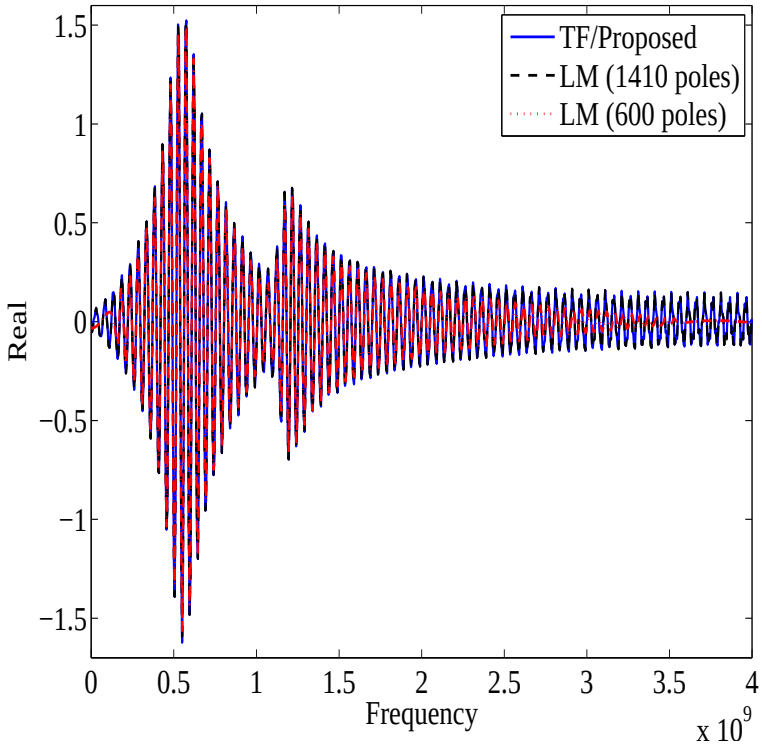


Figure 4.3: Frequency Response.

using (4.2) and (4.3). The results of the of energy spectra $\eta_{11}(\tau)$ and $\eta_{13}(\tau)$ for S_{11} and S_{13} respectively, are shown in Fig 4.4. Once the energy functions are determined, the delays of the systems are determined using (4.6) and (4.7) with a a threshold of $\varepsilon = 1e-5$. The partitioning of the time-frequency plane is done using (4.8) and (4.9), using a threshold of $\delta = 1e-6$ to identify the suitable partition points. Table 4.3 presents a summary of the identified partition points for all the transfer functions. Once, the regions are well defined, $\tilde{Y}_{ij}^{(m)}(s)$ are evaluated using (5). In order to ensure that the data for each region is fitted with low order rational functions, the delays are refined using step 5 of algorithm 1. Table 4.4 shows the original delays computed as well as the optimized ones. Fig. 4.5 shows the normalized singular values plots of S_{13} for the computed delay regions of the proposed method and LM method without delay extraction. Based on these plots, an order of $\{10, 21\}$ poles for regions one and two, respectively, were selected, while LM without delay extraction required 80 poles for similar accuracy. A summary of the selected order for each region for all the transfer functions along

Table 4.5: Results of the Rational Approximations of Example 2

	LM	**Proposed**		
		$\mathbf{\Omega_1}$	$\mathbf{\Omega_2}$	**Total**
$\mathbf{S_{11}}$	70	20	24	44
$\mathbf{S_{12}}$	60	8	—	8
$\mathbf{S_{13}}$	80	10	21	31
$\mathbf{S_{14}}$	50	10	—	10
$\mathbf{S_{22}}$	75	20	26	46
$\mathbf{S_{23}}$	48	9	—	9
$\mathbf{S_{24}}$	88	24	14	38
$\mathbf{S_{33}}$	70	22	25	47
$\mathbf{S_{34}}$	55	10	—	10
$\mathbf{S_{44}}$	72	24	20	44

with the order of the LM without delay extraction is presented in Table 4.5. The attenuation losses $S_{ij}^{(m)}(s)$ are evaluated as rational functions using the LM method on the data set of (10) for each region.

Fig 4.6 and 4.7 shows the frequency-response of S_{11} and S_{13} fitted using proposed method and the LM method without delay extraction. For this example, the proposed method resulted on on values ranging from 2 to 7 times fewer poles than LM without delay extraction.

4.4.3 Three port distributed network

A three-port distributed network is considered (as shown in Fig. 4.8) in this example. It consists of three subnetworks, containing two coupled transmission-lines. The per-unit-length parameters for subnetwork 1 are:

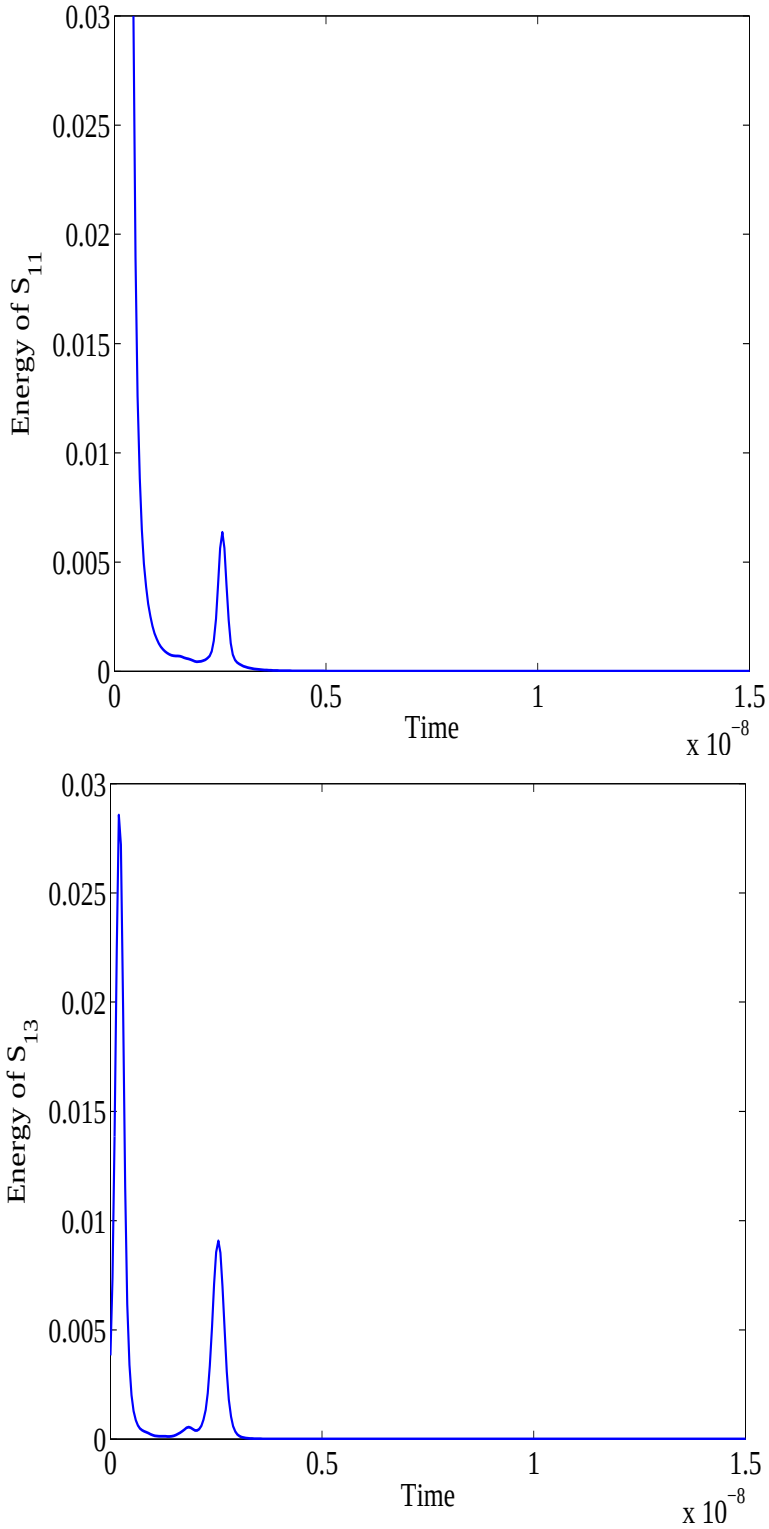


Figure 4.4: Energy functions of (top) S_{11} and (bottom) S_{13}

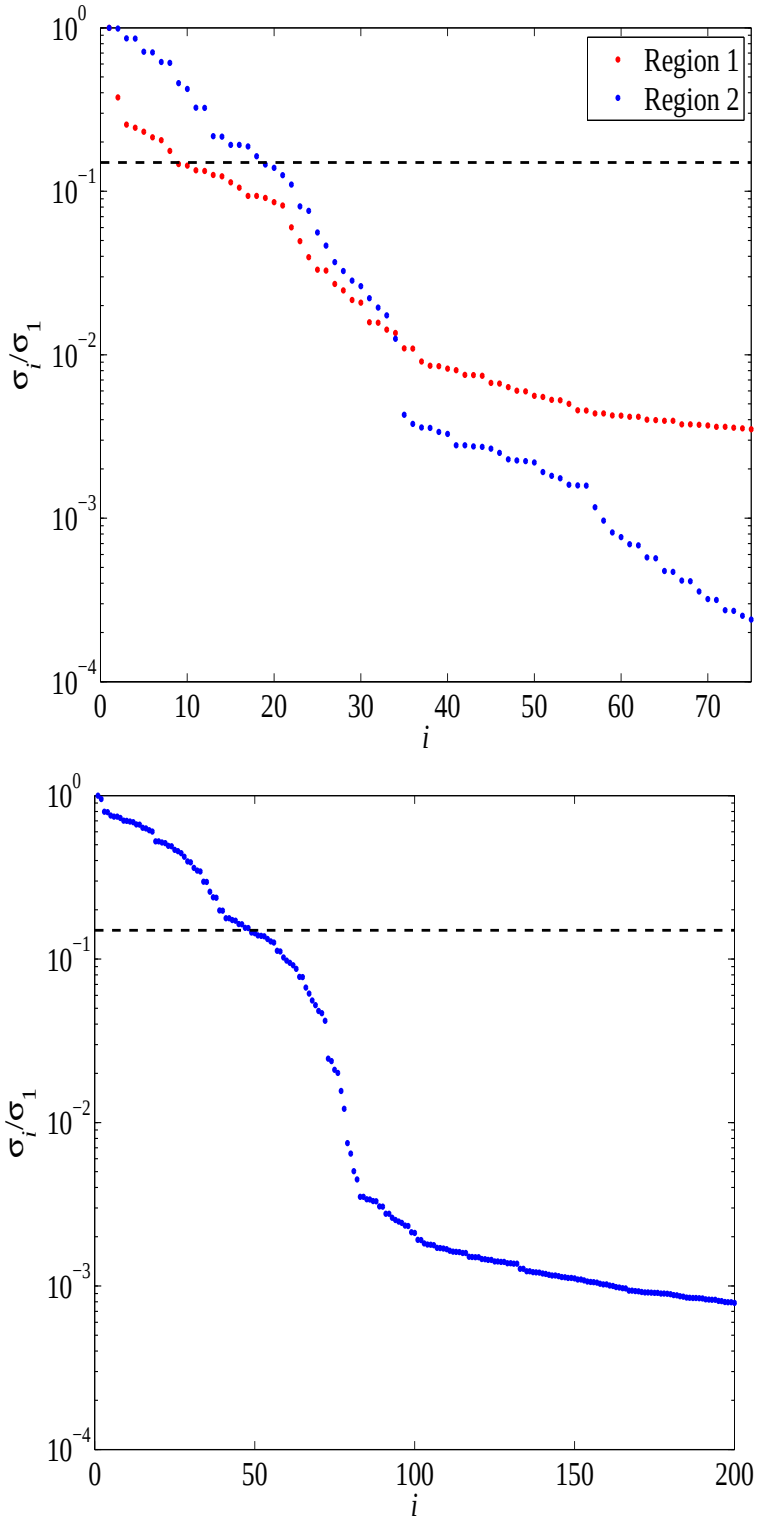


Figure 4.5: Normalized singular values (top) Proposed method (bottom) LM without delay extraction.

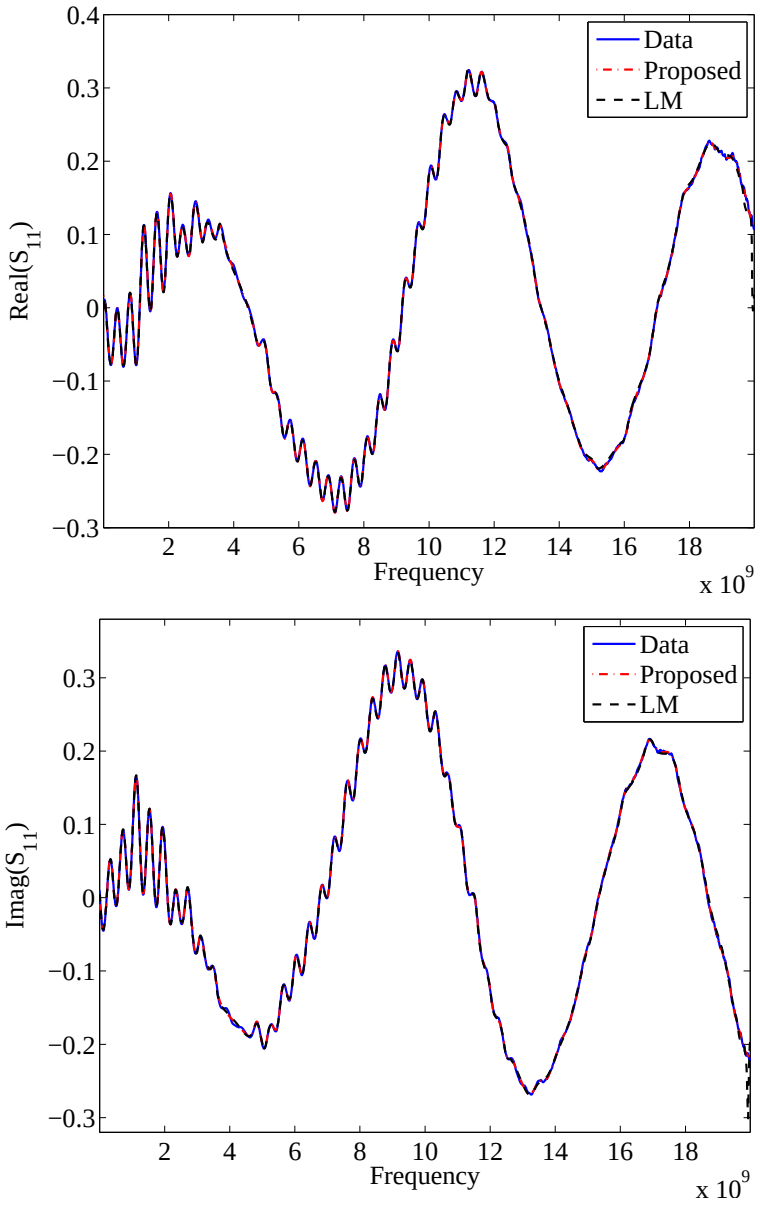


Figure 4.6: Comparaison of Proposed method and LM with the Data for S_{11}.

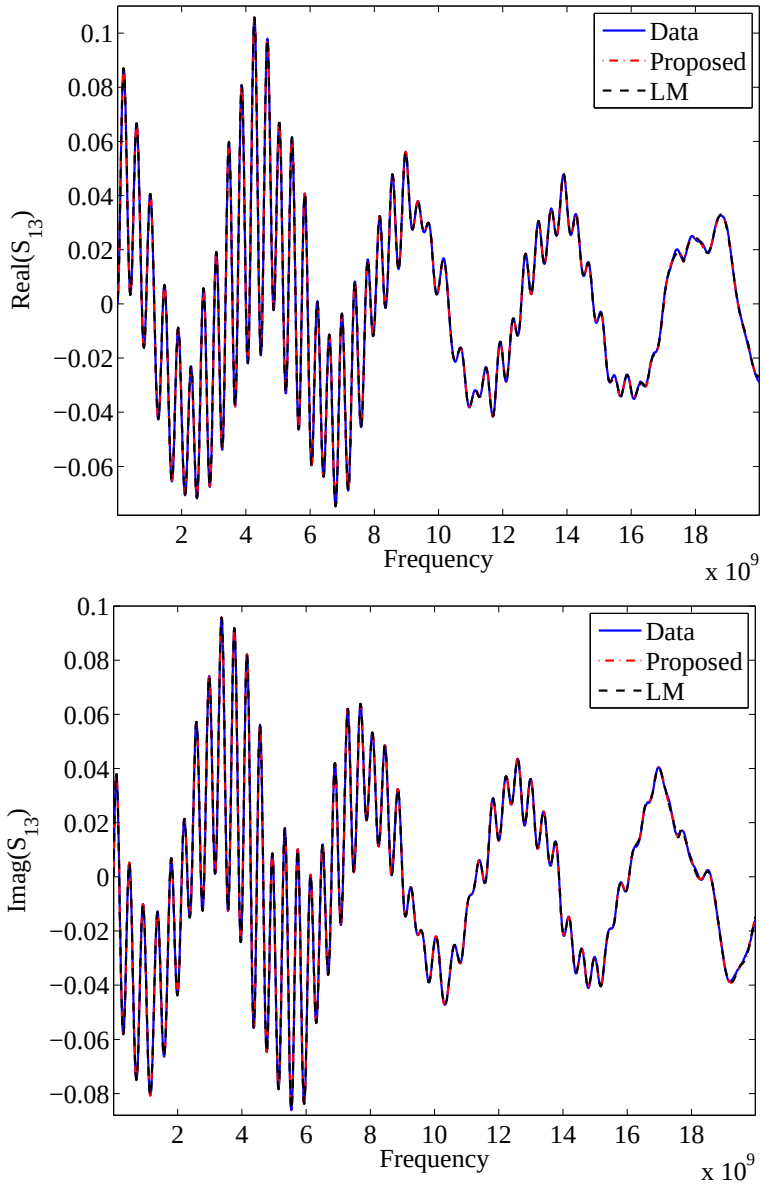


Figure 4.7: Comparaison of Proposed method and LM with the Data for S_{13}.

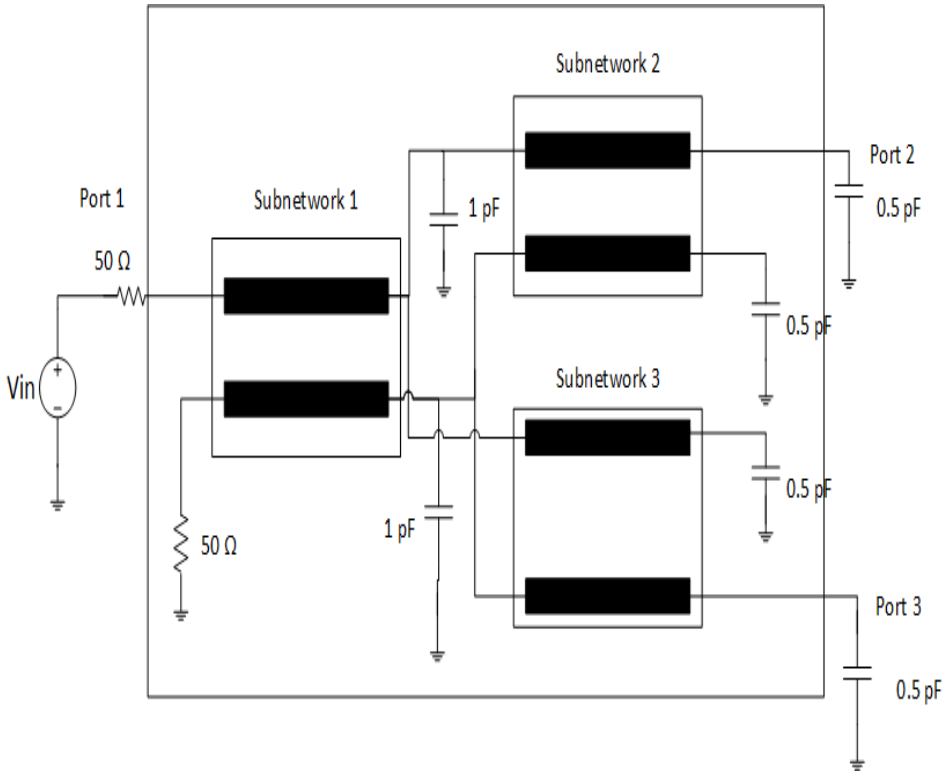


Figure 4.8: Three-port circuit of example 3

$$\mathbf{R} = \begin{bmatrix} 0.742 & 0 \\ 0 & 0.742 \end{bmatrix} \Omega/\text{cm}$$

$$\mathbf{L} = \begin{bmatrix} 3.36 & 0.865 \\ 0.865 & 3.36 \end{bmatrix} \text{nH/cm}$$

$$\mathbf{C} = \begin{bmatrix} 1.29 & -0.197 \\ -0.197 & 1.29 \end{bmatrix} \text{pF/cm}$$

$$\mathbf{G} = \begin{bmatrix} 0 \end{bmatrix} \text{mho/cm}$$

and the per-unit-length parameters for subnetwork 2 and 3 are:

Table 4.6: Time Axis Partitioning of (ω, τ) Plane of Example 3 (Times in ns)

Regions	$\mathbf{Y_{11}}$	$\mathbf{Y_{12}}$	$\mathbf{Y_{13}}$	$\mathbf{Y_{22}}$	$\mathbf{Y_{23}}$	$\mathbf{Y_{33}}$
$\mathbf{\Omega_1}$	[0.00 5.30]	[0.00 7.50]	[0.00 8.00]	[0.00 4.50]	[0.00 11.00]	[0.00 4.50]
$\mathbf{\Omega_2}$	[5.30 11.30]	[7.50 14.75]	[8.00 14.8]	[4.5 10.30]	[11.00 17.50]	[4.50 10.80]
$\mathbf{\Omega_3}$	[11.30 18.80]	[14.75 21.25]	[14.8 21.8]	[10.30 18.00]	[17.5 24.5]	[10.80 18.00]

$$\mathbf{R} = \begin{bmatrix} 0.803 & 0 \\ 0 & 0.803 \end{bmatrix} \Omega/\text{cm}$$

$$\mathbf{L} = \begin{bmatrix} 4.76 & 1.1 \\ 1.1 & 4.76 \end{bmatrix} \text{nH/cm}$$

$$\mathbf{C} = \begin{bmatrix} 2 & -0.58 \\ -0.58 & 2 \end{bmatrix} \text{pF/cm}$$

$$\mathbf{G} = \begin{bmatrix} 0 \end{bmatrix} \text{mho/cm}$$

The lines are of length $l1 = 5$cm and $l2 = 35$cm for the subnetwork one and subnetworks two and three, respectively. The three-port network is characterized by its Y-parameters response as tabulated data over a bandwidth of 0 to 3.5 GHz, generated using HSPICE. The time-frequency transform and the energy functions of all the Y-parameters are evaluated using (4.2) and (4.3). The results of the of energy spectra $\eta_{11}(\tau)$ and $\eta_{12}(\tau)$ for Y_{11} and Y_{23} respectively, are shown in Fig 4.9.

Once the energy functions are determined, the delays of the systems are determined using (4.6) and (4.7) with a a threshold of $\varepsilon = 1e-5$. The partitioning of the time-frequency plane is done using (4.8) and (4.9), using a threshold of $\delta = 1e-6$ to identify the suitable partition points, Table 4.6 presents a summary of the identified partition points for all the transfer functions. Once, the regions are well defined, $\tilde{Y}_{ij}^{(m)}(s)$ are evaluated using (4.5). In order to ensure that the data for each region is fitted with low order rational functions, the delays are refined using step

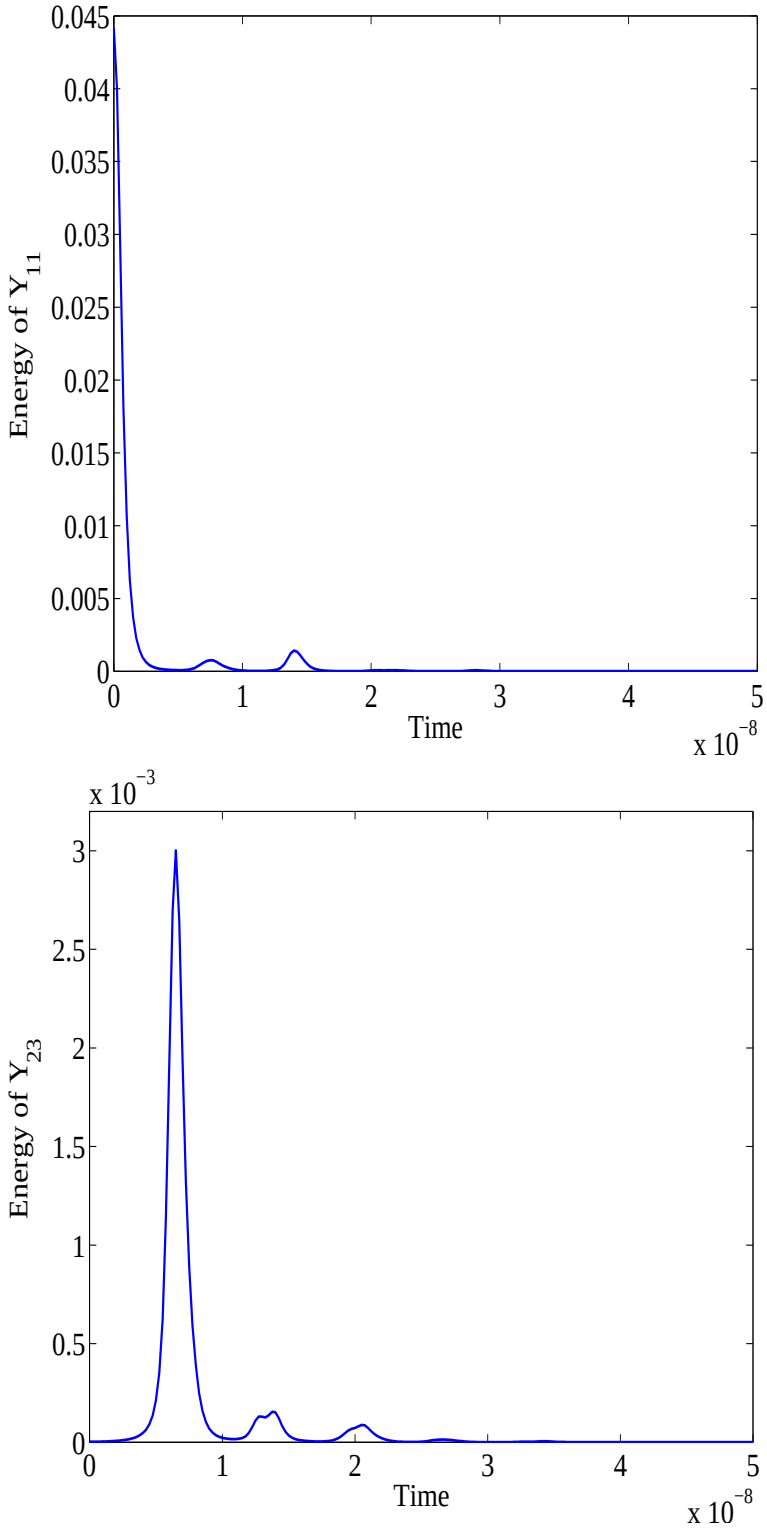


Figure 4.9: Energy functions of (top) Y_{11} and (bottom) Y_{23}

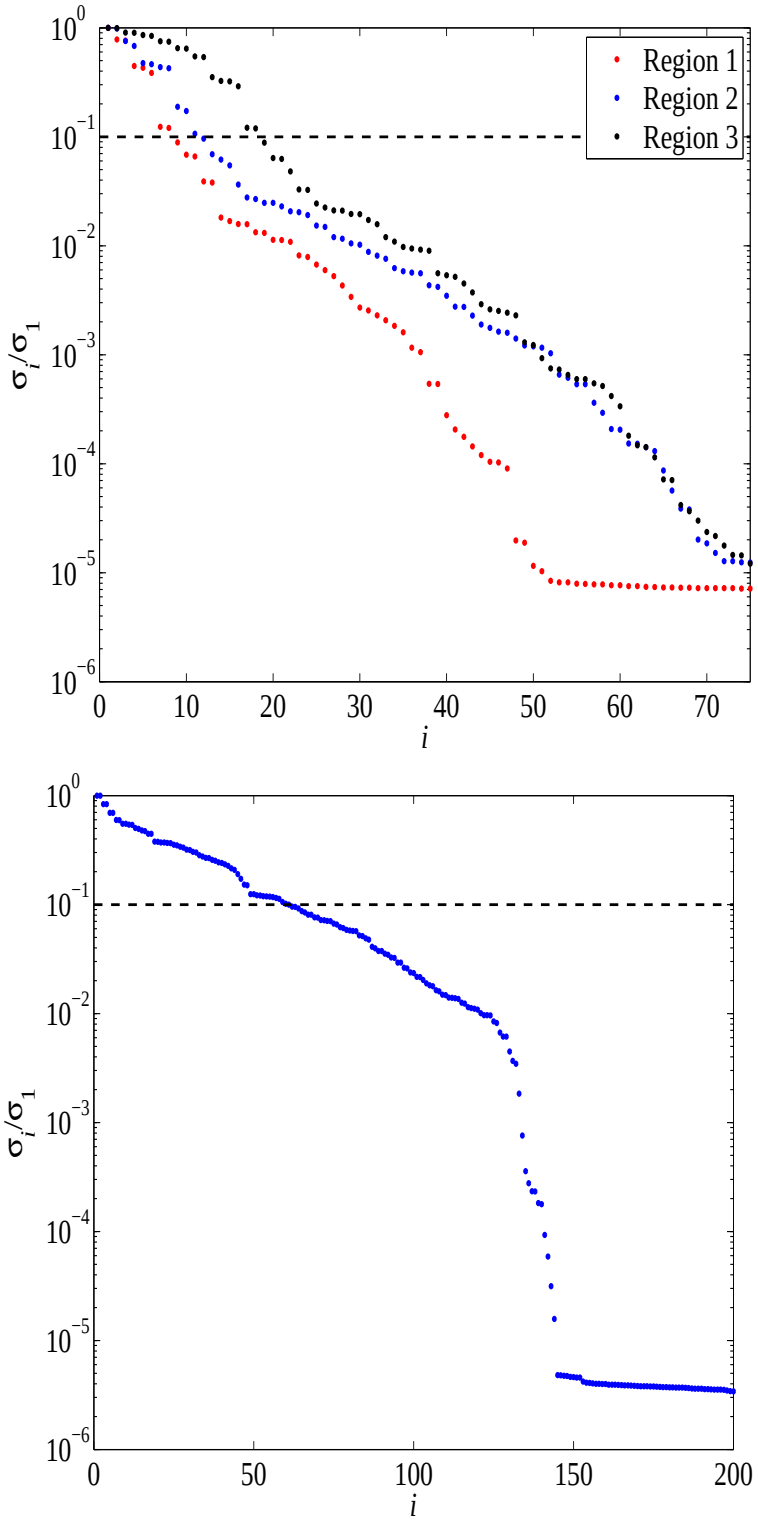


Figure 4.10: Normalized singular values (top) Proposed method (bottom) LM without delay extraction.

Table 4.7: Estimated delays versus optimized delays for Example 3 (Times in ns)

Regions		$\mathbf{Y_{11}}$	$\mathbf{Y_{12}}$	$\mathbf{Y_{13}}$	$\mathbf{Y_{22}}$	$\mathbf{Y_{23}}$	$\mathbf{Y_{33}}$
$\mathbf{\Omega_1}$	**Est.**	—	3.297	3.386	—	6.077	—
	Opt.	—	3.297	3.029	—	6.348	—
$\mathbf{\Omega_2}$	**Est.**	6.738	9.518	9.685	6.547	12.880	6.253
	Opt.	6.775	9.000	9.260	6.143	12.562	6.239
$\mathbf{\Omega_3}$	**Est.**	13.130	16.551	16.470	12.820	19.176	13.060
	Opt.	12.891	15.811	14.847	12.087	18.378	12.469

Table 4.8: Results of the Rational Approximations of Example 3

	LM	**Proposed**			
		$\mathbf{\Omega_1}$	$\mathbf{\Omega_2}$	$\mathbf{\Omega_3}$	**Total**
$\mathbf{Y_{11}}$	100	20	14	24	58
$\mathbf{Y_{12}}$	115	8	15	20	43
$\mathbf{Y_{13}}$	180	12	14	15	41
$\mathbf{Y_{22}}$	115	20	20	20	60
$\mathbf{Y_{23}}$	120	14	20	30	64
$\mathbf{Y_{33}}$	120	20	16	20	56

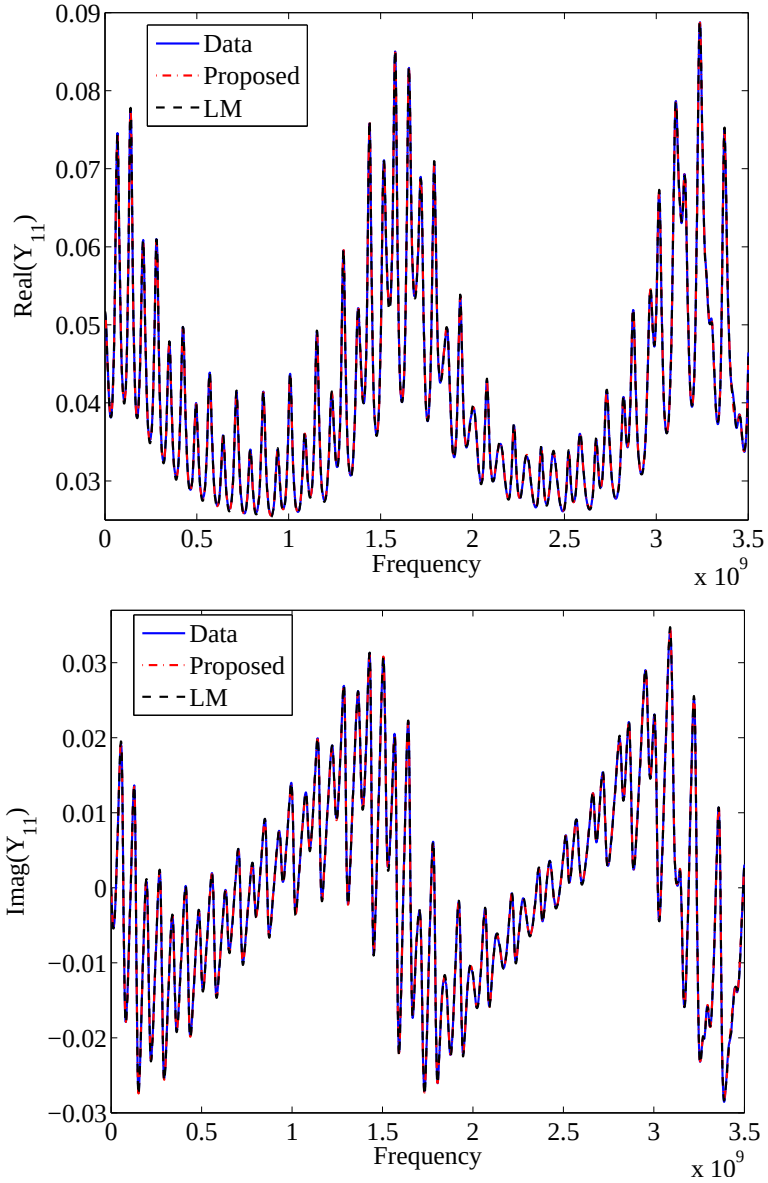


Figure 4.11: Comparaison of Proposed method and LM with the Data for Y_{11} (a) and (b); and Y_{23} (c) and (d)

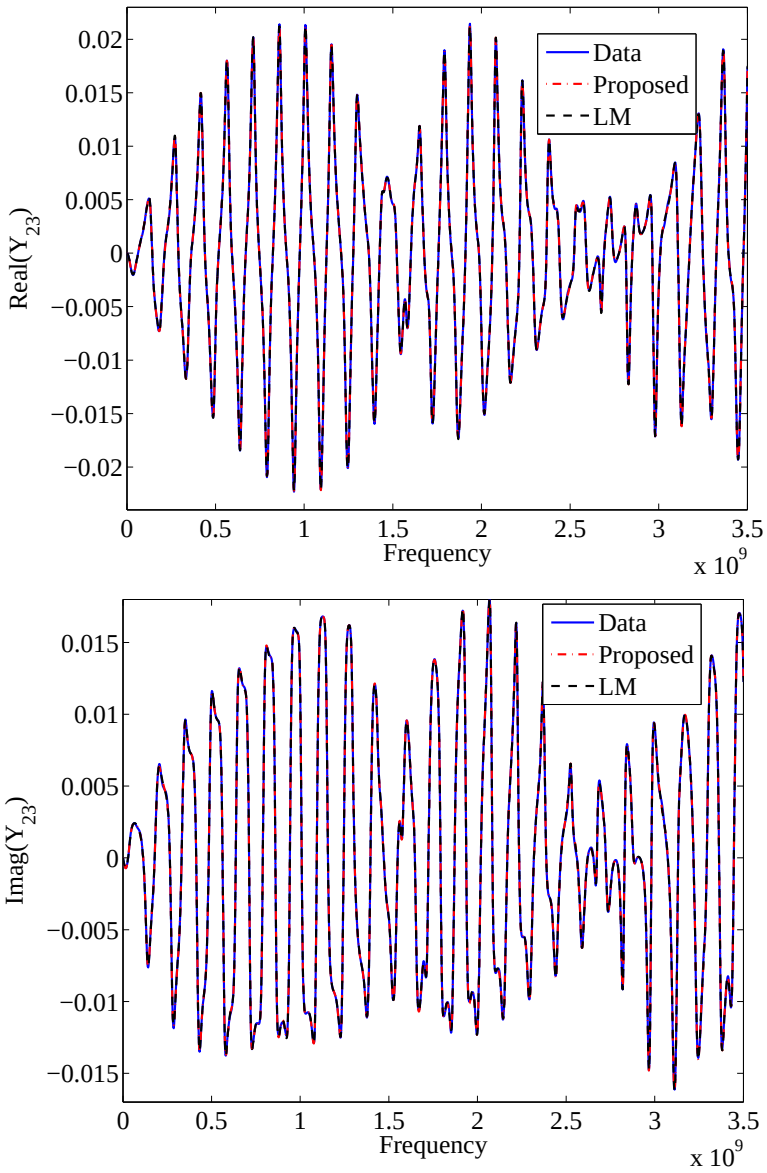


Figure 4.12: Comparaison of Proposed method and LM with the Data for Y_{11} (a) and (b); and Y_{23} (c) and (d)

5 of algorithm 1. Table 4.7 shows the original delays computed as well as the optimized ones. Fig. 4.10 shows the normalized singular values plots of Y_{23} for the computed delay regions of the proposed method and LM method without delay extraction. Based on these plots, an order of $\{14, 20, 30\}$ poles for regions one to three respectively was selected, while LM without delay extraction required 120 poles for similar accuracy. A summary of the selected order for each region for all the transfer functions along with the order of the LM without delay extraction is presented in Table 4.8. The attenuation losses $Y_{ij}^{(m)}(s)$ are evaluated as rational functions using the LM method on the data set of (4.10) for each region. Fig 4.11 and 4.12 shows the frequency-response of Y_{11} and Y_{23} fitted using proposed method and the LM method without delay extraction. For this example, the proposed method resulted on an average of 2.4 times fewer poles than LM without delay extraction.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this work, two efficient method to deal with tabulated data are presented, they seek to address two common problems when using curve fitting techniques to macromodel: data contaminated by noise and data that arise from long interconnect circuits. The first problem can cause the rational approximations to be inaccurate, while the second problem tends to produce rational approximations with higher order which can cause problems when using the models during transient analysis.

In chapter 2, an overview of the methods for the case when an analytic model is not easy to obtain is done. The methods presented fall under the rational curve fitting techniques category. The chapter focuses on two popular rational curve techniques, vector fitting (VF) and Loewner matrix (LM), along with some of the issues that can be encountered when macromodeling networks characterized by tabulated data.

In chapter 3, an instrumental variable QR decomposition approach is presented for vector fit-

ting and relaxed vector fitting to model multiport networks characterized by noisy frequency domain data. Using the concept of instrumental variables, wherein another set of tabulated is obtained, assuming that the noise of the two sets of noisy data are not very correlated, the bias due to the noise can be diminished.

In the proposed technique, it was shown that by taking the rational approximation of the previous iteration to create the instrumental variable, the biasing effect of the least squares solution caused by the noise of the data sample is minimized. This leads to an improved accuracy of the rational approximations.

In chapter 4, a modified version of the Loewner Method has been presented to obtain delay rational macromodel of distributed networks with electrically long interconnects characterized by frequency-domain tabulated data. Using a time-frequency transform, the propagation delays of the network are extracted and used to partition the time-frequency plane to create single atom, each containing a single delay. Then, the Loewner method is used to create low order rational approximations of each attenuation losses associated with the delay regions. Once all the region have been approximated, the final model is the combination of the delay rational approximations. Numerical examples were presented to validate the method and have shown that the method can be used to produced efficient and accurate methods that can be further used during transient analysis.

5.2 Suggestion for Future Research

There are many areas that can be explored further using the work presented in this thesis, some suggestions are presented in this section.

1. One area that could be pursed would to apply the Instrumental variable concept with the Loewner matrix method, indeed preliminary work done on the Loewner matrix frame-

work when dealing with noisy data shows that the method has trouble fitting the data. One way to deal with this problem would be to apply what was done with the instrumental variable vector fitting to the Loewner method. However, since the Loewner method is not iterative like vector fitting, a new instrumental varible data would need to be done. An early suggestion could be to mix and match the resulting matrices from two different Loewner matrices obtained from two different data sets. This would create a new descriptor state space system.

2. Another are to explore would be to expand on the method of delay macromodels using Loewner. However, instead of fitting the regions one by one and that combining them. A more efficient way would be to fit them globally, that way some errors introduce by the fit of each individual region would be reduced. This proposed method would be equivalent to what is done in the compact macromodeling technique described in chapter 2.

Bibliography

[1] R. Achar and M. S. Nakhla, "Simulation of high-speed interconnects," *Proceedings of the IEEE*, vol. 89, pp. 693–728, May 2001.

[2] A. Chinea and S. Grivet-Talocia, "On the parallelization of vector fitting algorithms," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 1, pp. 1761–1773, Nov 2011.

[3] C. R. Paul, *Analysis of Multiconductor Transmission Lines*. New York: Wiley, second ed., 2008.

[4] S. Thierauf, *Understanding Signal Integrity*. Artech House, 2010.

[5] C. R. Paul, *Introduction to Electromagnetic Compatibility*. New York: Wiley, 1992.

[6] Synopsys, Mountain View, California, *HSPICE Signal Integrity User Guide*, 2005.

[7] W. T. Beyene and J. Schutt-Aine, "Accurate frequency-domain modeling and efficient circuit simulation of high-speed packaging interconnects," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, pp. 1941–1947, Oct 1997.

[8] W. T. Beyene and J. E. Schutt-Aine, "Efficient transient simulation of high-speed interconnects characterized by sampled data," *IEEE Transactions on Components, Packaging, and Manufacturing Technology: Part B*, vol. 21, pp. 105–114, Feb 1998.

[9] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *Power Delivery, IEEE Transactions on*, vol. 14, pp. 1052–1061, Jul 1999.

[10] S. K. K. Mitra, *Digital Signal Processing: A Computer-Based Approach*. McGraw-Hill Higher Education, 2nd ed., 2000.

[11] A. R. Djordjevic, T. K. Sarkar, and R. F. Harrington, "Analysis of lossy transmission lines with arbitrary nonlinear terminal networks," *IEEE Transactions on Microwave Theory and Techniques*, vol. 34, pp. 660–666, Jun 1986.

[12] J. E. Schutt-Aine and R. Mittra, "Scattering parameter transient analysis of transmission lines loaded with nonlinear terminations," *IEEE Transactions on Microwave Theory and Techniques*, vol. 36, pp. 529–536, Mar 1988.

[13] R. Mandrekar and M. Swaminathan, "Causality enforcement in transient simulation of passive networks through delay extraction," in *Signal Propagation on Interconnects, 2005. Proceedings. 9th IEEE Workshop on*, pp. 25–28, May 2005.

[14] R. Mandrekar, K. Srinivasan, E. Engin, and M. Swaminathan, "Causal transient simulation of passive networks with fast convolution," in *Signal Propagation on Interconnects, 2006. IEEE Workshop on*, pp. 61–64, May 2006.

[15] B. Gustavsen, "Improving the pole relocating properties of vector fitting," *IEEE Transactions on Power Delivery*, vol. 21, pp. 1587–1592, July 2006.

[16] B. Gustavsen, "Relaxed vector fitting algorithm for rational approximation of frequency domain responses," in *Signal Propagation on Interconnects, 2006. IEEE Workshop on*, pp. 97–100, 2006.

[17] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter, "Macromodeling of multiport systems using a fast implementation of the vector fitting method," *Microwave and Wireless Components Letters, IEEE*, vol. 18, pp. 383–385, June 2008.

[18] S. Lefteriu and A. C. Antoulas, "A new approach to modeling multiport systems from frequency-domain data," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, pp. 14–27, Jan 2010.

[19] M. Kabir and R. Khazaka, "Macromodeling of distributed networks from frequency-domain data using the loewner matrix approach," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 60, pp. 3927–3938, Dec 2012.

[20] S. Grivet-Talocia and M. Bandinu, "Improving the convergence of vector fitting for equivalent circuit extraction from noisy frequency responses," *IEEE Transactions on Electromagnetic Compatibility*, vol. 48, pp. 104–120, Feb 2006.

[21] T. Soderstrom and P. Stoica, *Instrumental Variable Methods for System Identification*. Berlin, Germany: Springer-Verlag, 1983.

[22] L. Ljung, *System identification–Theory for the User*. Upper Saddle River, NJ: Prentice-Hall, second ed., 1999.

[23] S. Grivet-Talocia and M. Bandinu, "Improving the convergence of vector fitting for equivalent circuit extraction from noisy frequency responses," *Electromagnetic Compatibility, IEEE Transactions on*, vol. 48, pp. 104–120, Feb 2006.

[24] F. Ferranti, Y. Rolain, L. Knockaert, and T. Dhaene, "Variance weighted vector fitting for noisy frequency responses," *Microwave and Wireless Components Letters, IEEE*, vol. 20, pp. 187–189, April 2010.

[25] A. Beygi and A. Dounavis, "An instrumental variable vector-fitting approach for noisy frequency responses," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 60, pp. 2702–2712, Sept 2012.

[26] T. Soderstrom and P. Stoica, *System Identification*. London, UK: Prentice-Hall, 1989.

[27] P. Young, *Recursive Estimation and Time Series Analysis*. Berlin, Germany: Springer-Verlag, 1984.

[28] R. Pintelon and J. Schoukens, *System Identification–A Frequency Domain Approach*. Piscataway, NJ: IEEE, 2001.

[29] S. Grivet-Talocia, "Delay-based macromodels for long interconnects via time-frequency decompositions," in *Electrical Performance of Electronic Packaging, 2006 IEEE*, pp. 199–202, Oct 2006.

[30] A. Chinea, P. Triverio, and S. Grivet-Talocia, "Delay-based macromodeling of long interconnects from frequency-domain terminal responses," *Advanced Packaging, IEEE Transactions on*, vol. 33, pp. 246–256, Feb 2010.

[31] S. Roy and A. Dounavis, "Transient simulation of distributed networks using delay extraction based numerical convolution," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 30, pp. 364–373, March 2011.

[32] A. V. Oppenheim and R. W. Schafer, *Discrete-time Signal Processing (3rd Ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2010.

[33] A. Chinea, P. Triverio, and S. Grivet-Talocia, "Compact macromodeling of electrically long interconnects," in *2008 IEEE-EPEP Electrical Performance of Electronic Packaging*, pp. 199–202, Oct 2008.

[34] D. Gabor, *Theory of communication*. Institution of Electrical Engineering, 1946.

[35] A. Charest, M. Nakhla, R. Achar, D. Saraswat, N. Soveiko, and I. Erdin, "Time domain delay extraction-based macromodeling algorithm for long-delay networks," *Advanced Packaging, IEEE Transactions on*, vol. 33, pp. 219–235, Feb 2010.

[36] M. Sahouli and A. Dounavis, "Delay rational macromodels of long interconnects using loewner matrix," in *Electrical Performance of Electronic Packaging and Systems (EPEPS)*, p. in press.

[37] B. Gustavsen, "Time delay identification for transmission line modeling," in *Signal Propagation on Interconnects, 2004. Proceedings. 8th IEEE Workshop on*, pp. 103–106, May 2004.

[38] J. Yao, P. Krolak, and C. Steele, "The generalized gabor transform," *IEEE Transactions on Image Processing*, vol. 4, pp. 978–988, Jul 1995.

[39] The Mathworks, Inc., Natick, Massachusetts, *MATLAB version 8.5.0.197613 (R2014a)*, 2014.

Name:	Mohamed El Amine Sahouli
Post-Secondary Education and Degrees:	Carleton University Ottawa, ON 2010 - 2013 B.S.
Related Work Experience:	Teaching Assistant The University of Western Ontario 2014 - 2016 Research Assistant The University of Western Ontario 2014 - 2016

Publications:

- **An Instrumental-Variable QR Decomposition Vector-Fitting Method for Modeling Multiport Networks Characterized by Noisy Frequency Data**

  Published in: *IEEE Microwave and Wireless Components Letters* (Volume: 26, Issue: 9, Sept. 2016)

- **Delay Extraction Based Modeling Using Loewner Matrix Framework**

  Submitted to: *IEEE Transactions on Components, Packaging and Manufacturing Technology*

  Status: Accepted

- **Delay Rational Macromodels of Long Interconnects using Loewner Matrix**

  Submitted to: *25th Conference on Electrical Performance of Electronic Packages and Systems* Type: Poster