

Advanced Macromodeling in the Loewner Framework

Yi Qing Xiao

Doctor of Philosophy

Department of Electrical and Computer Engineering

McGill University

Montreal, Quebec

2023-05-17

A thesis submitted in partial fulfilment
of the requirements for the degree of
Doctor of Philosophy

© Yi Qing Xiao 2023

DEDICATION

This document is dedicated to the graduate students of the McGill University

ACKNOWLEDGEMENTS

I would like to acknowledge my supervisor professor Roni Khazaka who has supported my over the years in shaping the project that I have worked on. His guidance and insight were invaluable in accomplishing what I have done for my thesis. I would like to thank professor Dennis Giannacopoulos and professor Benoit Boulet for providing advice and direction for the project. I would also like to thank professor Stefano Grivet-Talocia and his research group at the Politecnico di Torino with whom a large portion of the project was made and where a paper was able to be published. I would like to thank my colleagues Mr. Muhammad Kabir and Mr. Marco Kassis who have helped me ease into my project when I first started and helped me make great progress throughout it. I would finally like to thank my family who has been patiently supporting me throughout this endeavour.

ABSTRACT

Circuit modeling has become a key step in the design of circuit systems given the advantage of being able to evaluate a design before dedicating any physical resource towards it. Macromodeling is an important technique in circuit modeling where the frequency domain data such as Y or S -parameters describing the port-to-port relations of the system are used to generate a black-box model for simulating the system. Macromodeling naturally encourages and synergizes with modular system design leading up to a system wide simulation through combination of individual models, giving great importance to the circuit model property of passivity which guarantees stability to combination of sub-models if the sub-models were passive. Passivity enforcement thus became necessary to perturb non-passive models to become passive, and this thesis presents new flexible ways of error control for the enforcement process related to gramian based error control. Another direction of interest for macromodels is the parametric extension to the modeling process where variability with respect to one or more physical property of the system can be included in the macromodels, allowing adjustment of the model with respect to key parameters without having to generate fresh port data repeatedly. This thesis presents two novel parametric macromodeling methods in the Loewner Matrix framework.

ABRÉGÉ

La simulation par modèle est devenue une étape clé dans la conception de système de circuit à cause de l'avantage de pouvoir évaluer un système avant de consacrer de la resource physique. La macro-modélisation est une importante technique de modélisation des circuits qui utilise les données dans le domaine de la fréquence représentant les relation entre les ports du système pour créer un model de boîte noire du system utilisable dans les simulations. Naturellement, la macro-modélisation encourage et est en synergie avec l'utilisation de systèmes modulaire avec lesquels une simulation du système en entier est achevé en combinant ces modules. La conception modulaire crée de l'importance dans la propriété de passivité, laquelle garantit la propriété de stabilité pour la combinaison des sous-modèles qui sont passifs individuellement. Pour cette raison, l'enforcement de passivité devient nécessaire pour corriger les modèles non-passifs, et cette thèse présente une nouvelle façon flexible pour contrôler l'erreur de perturbation basée sur la méthode de contrôle d'erreur par gramien. Une autre direction intéressante sur le sujet de macro-modélisation est l'extension paramétrique dans le processus de modélisation dans lequel la variabilité par rapport à une ou plusieurs propriétés physiques du système peut-être inclue dans les macro-modèles. Cette inclusion permet l'ajustement du modèle par rapport aux propriété clés sans avoir besoin de générer de nouvelles données à plusieurs reprises. Cette thèse présente deux nouvelles méthodes de macro-modélisation paramétriques dans le cadre de la matrice de Loewner.

TABLE OF CONTENTS

DEDICATION	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
ABRÉGÉ	v
LIST OF TABLES	x
LIST OF FIGURES	xi
LIST OF ACRONYMS	xvi
NOMENCLATURE	xviii
1 Introduction	1
1.1 Motivation and Background	1
1.2 Contributions	10
1.3 Organisation of the Thesis	11
2 Current State of Art	14
2.1 Macromodeling System Definition and Key Properties	14
2.1.1 Frequency Data Representations	15
2.1.2 Linear Time-Invariant Systems	17
2.1.3 Stability Condition	19
2.1.4 Passivity Condition	20
2.1.5 After Note	23
2.2 Loewner Matrix Review	23
2.2.1 Loewner Matrix Definition	24
2.2.2 Barycentric Rational Loewner Matrix Interpolation Method	25
2.2.3 System Format Loewner Matrix Method	36
2.3 Parametric Method Review	47

2.3.1	Data-Driven Parametric Loewner Model	48
2.3.2	Parametrized Sanathana-Koerner Method	53
2.3.3	Other Parametric Methods and Proposed Methods Context	54
2.4	Passivity Enforcement Review	55
2.4.1	Hamiltonian Matrix Eigenvalue Passivity Checking	56
2.4.2	Hamiltonian Matrix Eigenvalue Perturbation	59
2.4.3	Singular Value Perturbation	64
2.4.4	Gramian Based Error Control	68
2.4.5	Filtered Gramian Error Control	72
3	Passivity Enforcement	76
3.1	Discrete Gramian Error Control	77
3.2	Loewner Matrix System Method Built-In Error Control	82
3.3	Simulation	87
3.3.1	Perturbation Methods Strategies	88
3.3.2	Example 1	92
3.3.3	Example 2	95
3.3.4	Example 3	98
3.3.5	Example 4	101
3.3.6	Example 5	104
3.3.7	Example 6	108
3.3.8	Example 7	111
3.3.9	Example 8	114
3.3.10	Overall Results Analysis	118
4	Polynomial Parametric Barycentric Interpolation Method	121
4.1	Formulation Set-Up	122
4.1.1	Parametric Data Definition	122
4.1.2	Parametric Data Partitioning	123
4.1.3	Parametric Data Fitting Objective	124
4.2	Formulation in Scalar Data Case	125
4.2.1	Polynomial Basis	125
4.2.2	Polynomial Barycentric Equation - Partition Λ_Θ Fitting . .	128
4.2.3	Polynomial Barycentric Equation - Partition \mathcal{M}_Θ Fitting .	131
4.2.4	Parametric Loewner Matrix Definition and Solution	134
4.2.5	Real Polynomial Parametric Loewner Matrix	136
4.3	Formulation in General Case and Optimization	139

4.3.1	Polynomial Barycentric Equation - MIMO Case	139
4.3.2	Efficient SVD for \mathbb{L}	143
4.4	PBRLM Simulations	145
4.4.1	Example 1	145
4.4.2	Example 2	148
4.4.3	Example 3	152
5	Parametric Loewner Matrix System Method	155
5.1	Formulation Set-Up	155
5.1.1	Parametric Data Definition	156
5.1.2	Parametric Data Partitioning	157
5.1.3	Specialized Layered Matrix Indexing	157
5.1.4	Parametric Opposite Lagrange Equation	158
5.2	Formulation	160
5.2.1	Parametric Loewner Matrix Definition	160
5.2.2	Parametric Loewner Matrix System Interpolation Equation	161
5.2.3	Parametric Loewner Matrix Real Transform	168
5.2.4	Parametric Loewner Matrix System Model Order Reduction	172
5.2.5	Parametric Loewner Matrix 4 Quadrant Formulation . . .	174
5.3	Simulation	179
5.3.1	Parametric Data Smoothing Through Polynomial Regression	181
5.3.2	Example 1	182
5.3.3	Example 2	185
5.3.4	Example 3	188
5.3.5	Example 4	191
5.3.6	Example 5	194
5.3.7	Example 6	197
5.3.8	Overall Results Analysis	201
6	Conclusion	204
6.1	Passivity Enforcement Error Control Scheme Contributions . . .	204
6.2	Parametric Loewner Matrix Methods Contributions	205
6.2.1	Polynomial Barycentric Rational Loewner Matrix Interpolation Method	206
6.2.2	Parametric System Format Loewner Matrix Interpolation Method	207
6.3	Future Works	209

6.3.1	Discrete Gramian Customization	209
6.3.2	AAA Algorithm for Polynomial LM	210
6.3.3	Identical Partition LM	211
6.3.4	PSFLM Method Computation Cost Improvement	211
A	Barycentric Rational to State-Space	213
A.1	Barycentric Rational to Pole-Zero Form	214
A.2	Pole-Zero to Pole-Residue Form	219
A.3	Pole-Residue to State-Space Form	221
A.4	MIMO Consideration	223
B	Special Real Matrix Transformation	228
B.1	Single Matrix Unit Case	228
B.2	Full Matrix Case	230
B.3	Descriptor System Consideration	232
B.4	Loewner Matrix Real Transform	233
B.4.1	Loewner Matrix Offshoot Cases	237
	References	241

LIST OF TABLES

<u>Table</u>		<u>page</u>
3.1 Ex. 1 Simulation Results	95	
3.2 Ex. 2 Simulation Results	98	
3.3 Ex. 3 Simulation Results	101	
3.4 Ex. 4 Simulation Results	104	
3.5 Ex. 5 Simulation Results	107	
3.6 Ex. 6 Simulation Results	110	
3.7 Ex. 7 Simulation Results	114	
3.8 Ex. 8 Simulation Results	117	
5.1 Ex. 1 Simulation Results	184	
5.2 Ex. 2 Simulation Results	187	
5.3 Ex. 3 Simulation Results	190	
5.4 Ex. 4 Simulation Results	193	
5.5 Ex. 5 Simulation Results	196	
5.6 Ex. 6 Simulation Results	199	

LIST OF FIGURES

<u>Figure</u>		<u>page</u>
1.1 Simple graphical representation of the macromodeling process.	6	
2.1 Generic 2-port black-box network voltage and current description.	15	
2.2 Generic 2-port black-box network incident and reflected power waves description.	16	
2.3 Pole stability area portrayal.	19	
2.4 Passivity violation frequency region ($[\omega_l, \omega_u]$) illustration.	21	
2.5 Comparison of Lagrange polynomial (1st degree) and its opposite.	37	
2.6 Special interpolation function illustration.	38	
2.7 Association of imaginary eigenvalues of $(\mathcal{J}, \mathcal{K})$ with the singular value curves of $\mathbf{G}(s)$. Any imaginary entry in $eig\{\mathcal{J}, \mathcal{K}\}$ (left) must have corresponding transfer function singular value crossing point with threshold 1 (right).	57	
2.8 Illustration of error control coverage between the standard and the filtered gramian schemes.	75	
3.1 Illustration of error control coverage between the standard, filtered, and discrete gramian schemes.	81	
3.2 Hamiltonian matrix pencil imaginary eigenvalue perturbation process.	89	
3.3 Imaginary eigenvalue grouping.	90	
3.4 Transfer function singular value perturbation process.	91	
3.5 Example 1 Schematic	92	
3.6 Example 1 Passivity Enforcement Progress Plot (M- β , Disc. G.)	93	

3.7	Ex. 1 before and after [M- β , Disc. G.] sample data plot	93
3.8	Example 1 Freq. Pt. Error Plot	94
3.9	Example 2 Schematic	95
3.10	Example 2 Passivity Enforcement Progress Plot (M- β , Disc. G.) . . .	96
3.11	Ex. 2 before and after [M- β , Disc. G.] sample data plot	97
3.12	Example 2 Freq. Pt. Error Plot	97
3.13	Example 3 Schematic	98
3.14	Example 3 Passivity Enforcement Progress Plot (M- β , Disc. G.) . . .	99
3.15	Ex. 3 before and after [M- β , Disc. G.] sample data plot	100
3.16	Example 3 Freq. Pt. Error Plot	100
3.17	Example 4 Schematic	101
3.18	Example 4 Passivity Enforcement Progress Plot (M- β , Disc. G.) . . .	102
3.19	Ex. 4 before and after [M- β , Disc. G.] sample data plot	103
3.20	Example 4 Freq. Pt. Error Plot	103
3.21	Example 5 Schematic	105
3.22	Example 5 Passivity Enforcement Progress Plot (M- β , Disc. G.) . . .	105
3.23	Ex. 5 before and after [M- β , Disc. G.] sample data plot	106
3.24	Example 5 Freq. Pt. Error Plot	106
3.25	Example 6 Schematic	108
3.26	Example 6 Passivity Enforcement Progress Plot (M- β , Disc. G.) . . .	109
3.27	Ex. 6 before and after [M- β , Disc. G.] sample data plot	109
3.28	Example 6 Freq. Pt. Error Plot	110
3.29	Example 7 Schematic	111

3.30 Example 7 Passivity Enforcement Progress Plot (M- β , Disc. G.)	112
3.31 Ex. 7 before and after [M- β , Disc. G.] sample data plot	113
3.32 Example 7 Freq. Pt. Error Plot	113
3.33 Example 8 Schematic	115
3.34 Example 8 Passivity Enforcement Progress Plot (M- β , Disc. G.)	115
3.35 Ex. 8 before and after [M- β , Disc. G.] sample data plot	116
3.36 Example 8 Freq. Pt. Error Plot	117
4.1 Magnitude and phase of $S_{1,1}$ for the Chebyshev filter example of Section 4.4.1, computed for different parameter values. Original data (thick dashed lines) are compared against macromodel responses (thin solid lines).	146
4.2 Magnitude and phase of $S_{1,1}$ for the Chebyshev filter example of Section 4.4.1, computed for different frequency values as a function of the parameter. Original data (thick dashed lines) are compared against macromodel responses (thin solid lines).	146
4.3 Magnitude of the error on $S_{1,1}$ between macromodel and original data for all the available parameter samples of the Chebyshev filter example.	147
4.4 Maximum and RMS approximation error against the singular value σ_* used to determine the model coefficients through the corresponding singular vector.	147
4.5 RMS error minimization path for the microstrip example of Section 4.4.2.148	
4.6 Magnitude of $S_{2,1}$ for the microstrip example of Section 4.4.2, computed for different parameter values. Original data (thick dashed lines) are compared against macromodel responses (thin solid lines).	149
4.7 Magnitude of model error over frequency, computed for all scattering matrix elements and available parameter samples for the microstrip example of Section 4.4.2.	150

4.8	Magnitude of model response computed over a fine sweep in the frequency-parameter space (thin solid blue lines) with the proposed method and the method of MLMRI.	151
4.9	Poles trajectories along the parameter space for the microstrip example from Section 4.4.2.	151
4.10	Normalized singular values of the parametric Loewner matrix (PLM) for the first example of Section 4.4.3 (parametric variation of antipad radius).	152
4.11	Comparison between model and original data for the magnitude and phase of the scattering response $S_{1,1}$ for the first example of Section 4.4.3 (parametric variation of antipad radius).	153
4.12	Magnitude of model error over frequency, computed for all scattering matrix elements and available parameter samples for the first example of Section 4.4.3 (parametric variation of antipad radius): (a) proposed Loewner approach with uniformly distributed points in the first partition j ; (b) proposed Loewner approach with a denser distribution of first partition points at high frequencies; (c) PSK algorithm.	154
5.1	Opposite two-variate 1st degree Lagrange polynomial interpolation illustration.	159
5.2	Dummy example illustration of the interpolation function (5.20) (left) and (5.21) (right).	162
5.3	Parametric data interleaving partitioning.	163
5.4	Example 3 $S_{1,2}$ data regression over the parameter domain. Each line is at a different sample frequency point.	181
5.5	Simple mstrip schematic	182
5.6	Ex. 1 Sample $S_{1,1}$ magnitude and phase plots	183
5.7	Ex. 1 poles maximum real part plot	184
5.8	Ex. 1 RMS error w.r.t. freq. domain plot.	185
5.9	Printed circuit board package merging section schematic	185

5.10 Ex. 2 Sample $S_{2,3}$ magnitude and phase plots	186
5.11 Ex. 2 poles maximum real part plot	187
5.12 Ex. 2 RMS error w.r.t. freq. domain plot.	188
5.13 Differential vias basic design schematic	188
5.14 Ex. 3 Sample $S_{2,2}$ magnitude and phase plots	189
5.15 Ex. 3 poles maximum real part plot	190
5.16 Ex. 3 RMS error w.r.t. freq. domain plot.	191
5.17 Package section schematic	191
5.18 Ex. 4 Sample $S_{1,3}$ magnitude and phase plots	192
5.19 Ex. 4 poles maximum real part plot	193
5.20 Ex. 4 RMS error w.r.t. freq. domain plot.	194
5.21 Package section schematic	195
5.22 Ex. 5 Sample $S_{1,1}$ magnitude and phase plots	196
5.23 Ex. 5 poles maximum real part plot	197
5.24 Ex. 5 RMS error w.r.t. freq. domain plot.	197
5.25 Package example schematic	198
5.26 Ex. 6 Sample $S_{1,4}$ magnitude and phase plots	199
5.27 Ex. 6 poles maximum real part plot	200
5.28 Ex. 6 RMS error w.r.t. freq. domain plot.	200
5.29 Poles maximum real part plots for all examples (divided by columns, ex. 1 to 6 from left to right) by both methods (divided by rows, method PBRLM to PSFLM from top to bottom).	201
5.30 RMS error over freq. domain plots at θ points for all examples (divided by columns, ex. 1 to 6 from left to right) by both methods (divided by rows, method PBRLM to PSFLM from top to bottom).	202

LIST OF ACRONYMS

<u>Acronyms</u>	<u>first case page</u>
AAA Adaptive-Antoulas Anderson	55
BIBO bounded-input bounded-output	19
BRLM barycentric rational Loewner matrix	12
DS descriptor system	17
EDA electronic design automation	1
EM electro-magnetic	1
FDTD finite-difference time-domain	3
FFT fast Fourier transform	148
I/O Input/Output	11
LM Loewner Matrix	6
LTI linear time-invariant	7
MIMO multiple inputs multiple outputs	29
MLMRI multivariate Loewner matrix rational interpolation	150
MNA modified nodal analysis	2
MOR model order reduction	4

ODE ordinary differential equations	1
PBRLM polynomial barycentric rational Loewner matrix	13
PCB printed circuit board	152
PDE partial differential equations	1
PEEC partial element equivalent circuit	4
PLM parametric Loewner matrix	xiv
PRIMA passive reduced-order interconnect macromodeling algorithm	5
PSFLM parametric system format Loewner matrix	13
PSK parameterized Sanathanan–Koerner	53
RMS root mean square	47
RS regular system	18
SFLM system format Loewner matrix	8
SISO single input single output	25
SK Sanathanan-Koerner	6
SPICE Simulation Program with Integrated Circuit Emphasis	2
SVD singular value decomposition	27
TL transmission lines	2
VF Vector Fitting	6

NOMENCLATURE

\mathbf{X}^*	Complex conjugate.
\mathbf{X}^H	Matrix Hermitian.
\mathbf{X}^\dagger	Matrix Moore-Penrose pseudo-inverse.
\mathbf{X}^{-H}	Inverse of matrix Hermitian.
$\mathbf{X}_{\{i,j\}}$	The scalar entry of matrix \mathbf{X} at its i 'th row and j 'th column.
$\langle \cdot, \cdot \rangle$	Inner product operator
$tr(\mathbf{X})$	Matrix trace.
$vec(\mathbf{X})$	Stacking of the columns of \mathbf{X} from left to right into a single column.
$\Re\{\mathbf{X}\}$	Matrix real part.
$\Im\{\mathbf{X}\}$	Matrix imaginary part.
$\max(\mathbf{X})$	Matrix maximum term.
$\min(\mathbf{X})$	Matrix minimum term.
$\sigma(\mathbf{X})$	Matrix's set of singular values.
$eig\{\mathbf{X}\}$	Matrix's set of eigenvalues.
$eig\{\mathbf{X}, \mathbf{Y}\}$	Set of generalized eigenvalues of matrix pair (\mathbf{X}, \mathbf{Y}) .
$\ \mathbf{X}(\cdot)\ _2$	H_2 -norm. If \mathbf{X} is not a function, assume P_2 -norm.
$\ \mathbf{X}\ _F$	Frobenius norm or P_2 -norm.
x'	Derivative with respect to variable given by context.
$\mathbf{0}_{x \times y}$	$x \times y$ zero matrix.

$\mathbf{I}_{x \times y}$ $x \times y$ identity matrix (1's only appear on main diagonal).

$\sup_{x \in \mathcal{D}}\{y(x)\}$ Supremum of scalar function $y(x)$. Least upper bound of $y(x)$ given any x within its defined domain \mathcal{D} .

Chapter 1

Introduction

1.1 Motivation and Background

Electronic design automation (EDA) tools have become a key step of designing a system as systems grow in size and complexity that cannot be manually analysed. EDA has had a great impact on the growth of the microelectronics industry as it reduces the need for physical prototype and general time from design to market. Over the EDA process, various properties of a circuit are simulated via the use of a mathematical representation. The representation is given the intuitive name of model, as it is its goal to model the behaviour of the original circuit.

One important class of models is based on Maxwell's equations whose solutions are obtained through full-wave electro-magnetic (EM) simulation. The solution to these partial differential equations (PDE) can fully describe the EM behaviours of a system through the technique of discretization of the solution space. The degree of solution accuracy is determined by the level of discretization of the solution space with denser discretization coming at the cost of increasing computation time as well as memory [1, 2].

The other important class of models is based on circuit theory which utilizes lumped circuit components (R, L, C, V/I sources, dependent sources, etc.). The use of lumped circuit components implies the abstraction of spatial variation within the problem, allowing the solving of the system through ordinary differential equations

(ODE) rather than PDE. The most prominent simulators using circuit components are the Simulation Program with Integrated Circuit Emphasis (SPICE) and SPICE like simulators, which model systems using such circuit elements listed on a netlist (text file) that is stamped into its modified nodal analysis (MNA) equation which is the non-linear ODE representation of the original circuit [3, 4]. The non-linear ODE representation provides SPICE simulators a distinct advantage of being able to include non-linear circuit elements in its transient simulation, contributing to the great popularity of SPICE simulators [5]. We utilize SPICE as the outlying goal for models in this Thesis: circuit models created should aim to be SPICE compatible so that they have access to non-linear transient simulation.

As system operating frequencies increased and design space decreased over time, an issue arose for the modeling of interconnects. The lumped circuit element based models previously thought to be sufficient for representing interconnects cannot characterize the rising EM effects such as reflections, crosstalk, ringing, signal distortion and delay [6–9]. The issue can be understood from the modeling of transmission lines (TL), which have the classic R, L, C, G lumped model [10] which leads to an easy integration into SPICE. However, as frequency of operation increases, dispersion effects such as conductor skin effect and substrate losses become more pronounced and eventually invalidate the model. Various techniques have been implemented to adjust the TL model to take into account various dispersion effects [5, 10–12], but ultimately, once the dimension of the conductor’s cross-section is no longer considered small compared to the minimum wavelength of operation, the TL model no longer

holds [13, 14]. This issue is further compounded by the breach of the other fundamental assumption of the uniformity of cross-section of the interconnect [13, 14]. Artefacts such as change in conductor dimensions, sudden change in conductor directions, change in the immediate environment around the conductor and others all contribute to invalidating the simple interconnect modeling. Although adjustments exist for TL modeling to deal with minor non-uniformities [15–17], more complex geometries of interconnects such as multi-conductor bends and vias make the simple TL model impractical. In order to obtain models that properly model dispersion effects, the most immediate solution would be to fall back unto EM models as they include EM effects by definition. However, EM models, which model electric and magnetic fields in the frequency domain, are incompatible with MNA format circuit models in SPICE that runs transient in the time-domain [5]. The situation prompted enormous effort in achieving models that take into account EM effects while being compatible with SPICE like simulators in order to have access to the convenience of simulating ODE.

One avenue of solution explored is a hybrid solution based on the finite-difference time-domain (FDTD) field solver method [18], which has the the speciality of solving for the EM response through a time-domain transient. This time-domain EM solution aspect opens the possibility to integrate SPICE non-linear models in the FDTD equations in order to solve for the transient simultaneously [19–21]. However, the complexity of non linear elements can easily become too prohibitive for proper interfacing with the EM solver [5]. Another line of solution starts in Maxwell

integral equation-based EM fields solving methods, whose most popular discretization method is the method of moments (MoM) [9]. With a specialized discretized electric field integral equation of the interconnects, one can create an EM approximation model which can be translated to an equivalent circuit description with capacitors and inductors, and the model is called a partial element equivalent circuit (PEEC) [22]. Though the PEEC models face time delay problems in the MNA formulation [23], it is flexible enough to allow omission of time delay if the structure is small enough relative to the wavelength [5]. Simplification can even go as far as to omit capacitance if switching analysis is the objective [24, 25].

Another solution goes back to the frequency domain EM models obtained through finite methods such as finite-element and finite difference methods which spatially discretize the Maxwell equations and can be expressed in state-space format [26, 27]. Such state-space model uses the discrete E-fields and M-fields components as state-variables and discrete E and M current sources as sources, but a specialized EM multiport version of the state-space model can be created if we limit the sources to only those that impose excitation at the defined ports [27]. Either way, the issue with such a state-space model is in its size, as the number of state-variables, which is also the order of the system, is proportional to the number of discretization units there are for the model. As such, such a state-space model can have prohibitively high order, making their simulation very expensive. Thus, extensive efforts have been made in model order reduction (MOR) techniques to extract usable reduced-order versions of said state-space models. The general model-order reduction development started based on the Padé approximation, which is a scalar approximation method

that uses a rational function of power series to match the moments of the original data function up to a certain degree [28–30]. The use of Arnoldi’s algorithm for MOR followed. The algorithm iteratively approximates the largest eigenvalues and their eigenvector of the poles matrix of the transfer function of the state-space model [31]. The Lanczos algorithm was created subsequently to improved the Arnoldi algorithm by further ensuring orthonormality between the approximated vectors throughout the Arnoldi iterations [32, 33]. Overall MOR development eventually produced the passive reduced-order interconnect macromodeling algorithm (PRIMA), which as the name suggests, goes a step beyond Arnoldi and Lanczos algorithms by preserving the passivity of the state-space system throughout the MOR if the system was passive to begin with [34]. Further development of MOR has followed [35–38], but as MOR is not a key part of this thesis, we go no further into details.

The final solution type presented here, which is the most relevant to this thesis, utilizes the tabulated port frequency data from the interconnects. Different from the method of MOR of EM models, we only utilize a finite set of frequency data from the original system which can be obtained by fullwave EM simulation [39] or direct measurement using vector network analyzer [40]. Any specific details of the original system or EM model of the system is discarded, which is why this type of methods is given the name of black-box macromodeling. The process is illustrated on Figure 1.1, which highlights the fact that the black-box macromodel is built upon only the frequency data of the original system, and as such, the black-box models tend to achieve significantly lower order since it is an approximation of a limited number of data. Using an appropriate macromodeling method based on rational functions,

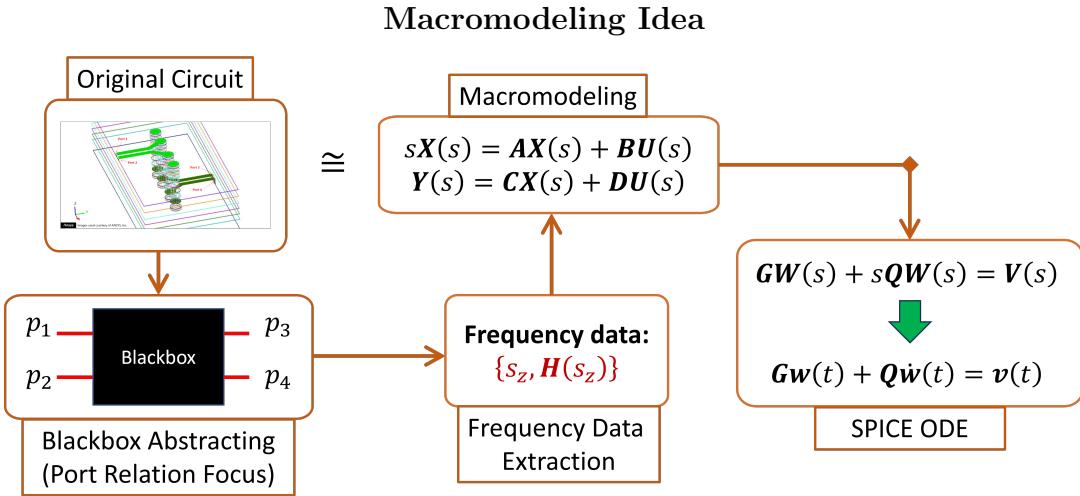


Figure 1.1: Simple graphical representation of the macromodeling process.

a macromodel can be generated typically represented as or can be translated to a state-space model. The macromodel can then be reformatted into a standard set of non-linear ODEs that is ready to be used for transient analysis in SPICE simulators, which is one of the key applications of black-box modeling.

Black-box macromodeling has been the subject of intense research over the past decades [41–65]. The current industry standard is the Vector Fitting (VF) method, an effective reformulation of the Sanathanan-Koerner (SK) iteration method [66], which have by now an extensive history of research and polishing [42, 46–48, 50] with a recent return to the SK formula for a parametric extension [67–70]. More recently, an interesting alternative macromodeling framework based on the Loewner Matrix (LM) has been introduced as an alternative to VF [49, 52–65, 71–85]. The LM methods tend to construct accurate models with low complexity and handles large number of ports efficiently, though computation cost vary from low to high depending on the approach. This thesis contributions are based on two points of interest regarding

black-box macromodeling. The contributions will be presented specifically using the black-box model form of linear time-invariant (LTI) state-space systems.

The first point is the important model property of system passivity which is directly associated to the key property of system stability. In general, a valid model needs to be stable in order to prevent transient simulation from diverging, especially if the model is simulating a passive component. The passivity property implies that of stability, but it further offers the useful feature of having combination of passive models result in a passive model and thus a stable model. Combination of stable models does not necessarily result in a stable model, which is why passivity is such an important property to have on macromodels as modeling tends to be performed piecewise with the intention of combining individual models at the end for a system wide simulation. The issue is that passivity isn't guaranteed with most macromodeling methods, which prompted great effort in developing passivity enforcement methods to patch the passivity property back to the macromodel or preemptive measures to ensure passivity by construction [60, 86–100]. The process of passivity enforcement for state-space models usually employs the Hamiltonian matrix as a way of robust passivity checking by checking the spectrum of the matrix [86]. The actual step of perturbing the system in order to correct passivity violations can be done by perturbing the spectrum of the Hamiltonian matrix [87]. Alternatively, we can directly perturb the singular values of the response data matrix at local passivity violation frequency points [99]. The enforcement methods naturally introduce perturbation error to the system, which is why it is normally performed while using an error control

scheme. The popular error control scheme that accompanies perturbation methods is the use of controllability or observability gramians to alter the perturbation equations so that minimization of system response error aligns with the least square minimization of the pertubant [87, 88]. The gramian-based error control scheme has been improved upon with the use of filters to adjust the frequency band over which the response's perturbation error control is prioritized [99]. This thesis will introduce an alternate modification to the gramian based error control under the name of discrete gramian which allows even further customization of error control over the frequency band. Furthermore, the thesis also introduces a perturbation error control scheme unique to the system format Loewner matrix (SFLM) macromodels which achieves a structure that provides a built-in error resistance towards perturbation of its residual matrices.

This thesis' second point of interest is parametric macromodeling, which is modeling with respect to additional variables other than frequency. A parametric macro-model does not simply interpolate a single frequency response, but a whole series of frequency responses that vary with respect to one or multiple variables. The additional variables can be real physical parameters such as wire thickness, substrate relative permittivity, separation length of strips, etc. The advantage offered by parametric models is that they provide a complete set of responses with respect to some chosen key design parameters of the system without having to generate the data through repeated measuring or costly simulation. The industry standard VF method has already been given much advances towards parametric extensions [67, 101–111] and its parent SK method was also given some recent breakthroughs [67–70] . This

thesis contribution towards parametric macromodeling is done within the LM framework. A number of developments have already been made towards the LM parametric extension [56, 84, 85, 112–114]. In [56], a Loewner Matrix parametric method was proposed by using principal component analysis to obtain continuous macromodels that are suitable for interpolation. However, this method can only deal with parametric challenge where the order remains relatively constant to the parameters [113]. [112, 114] offer a more direct method for obtaining Loewner Matrix based parametric macromodels by using the multivariate Loewner Matrix, with one variable being the frequency and the others being the desired circuit parameters. The main issue with this method is that it induces non-smooth approximations due to strict interpolation conditions that are enforced in the parametric data set. In this thesis, two new approaches to parametric Loewner Matrix macromodeling are presented. The first approach borrows the idea from [102] where the use of an orthogonal polynomial basis provided parametric extension to the VF method. The same strategy is applied to the barycentric form of the Loewner Matrix framework, and the specifics of the implementation as well as simulation results are presented in this thesis. This parametric method has been already published in the paper [65]. The second approach also utilizes the multivariate Loewner Matrix like the methods in [112, 114], but the manner in which the parametric model is formulated is different. The parametric LM method proposed directly extends the univariate SFLM method from [52] to a multivariate version.

1.2 Contributions

The first contribution is related to the issue of passivity enforcement error control, which consists of the Loewner matrix built-in error control scheme specifically applicable to the SFLM models. The contribution can be found at Section 3.2. This error control scheme relies on the special structure of the Loewner matrices which introduces a built-in error minimization effect when attempting to perturb the residue matrices of the state-space system created by the SFLM method.

The second contribution follows the passivity enforcement error control idea and is dubbed the discrete gramian error control and can be found in Section 3.1. The novelty comes from the fact that the standard gramian error control [115] attempts to minimize perturbation error on the frequency domain response of the LTI system over the entirety of the frequency band while the discrete gramian error control allows customizable error control over selective portions of the frequency band. Of course, this idea of selective error control is not new given the existence of filtered gramian error control [99], but the discrete gramian forgoes the need of designing a filter and is more cost efficient as well.

The third contribution is related to parametric macromodeling specifically within the LM framework and is found in Chapter 4. The proposed parametric Loewner Matrix method is created through introduction of polynomials to the Loewner Matrix barycentric interpolation method in order to tackle variations of the system response to variables other than the frequency. The merger of polynomials to the barycentric Loewner Matrix equation result in a macromodel that can vary with respect to multiple variables.

The fourth contribution is also related to parametric macromodeling within the Loewner Matrix framework and is found in Chapter 5. Different from the polynomial Loewner Matrix method, rather than utilizing polynomials, we directly turn the univariate SFLM method to its two-variate equivalent. It should be noted that the proposed Loewner Matrix method is different from existing two-variate Loewner Matrix method [112, 114] in that the interpolation principle is much more tied to the univariate SFLM method.

As for contributors towards the present thesis, I was the only student involved in this work where I implemented the algorithms under the supervision of professor Khazaka. The work is also performed in collaboration with Professor Stefano Grivet-Talocia and Dr. Paolo Manfredi at the Politecnico di Torino where I have also spent time as visiting student.

1.3 Organisation of the Thesis

Following the present introduction section, we begin first by defining the types of frequency data used to represent the electric network Input/Output (I/O) behaviour and used to construct the macromodels that simulate such behaviour (Section 2.1.1). Then, the form of macromodels used throughout the thesis, the LTI state-space systems, is defined in details (Section 2.1.2). Once the model format is defined, we define the model properties of stability (Section 2.1.3) and passivity (Section 2.1.4) necessary for presenting the passivity enforcement contributions.

Once the macromodeling base theories are presented, the univariate Loewner Matrix macromodeling methods are reviewed in order to be able to develop the multi-variate extensions (Section 2.2). The two methods reviewed are the barycentric rational Loewner matrix (BRLM) interpolation method (Section 2.2.2) and the system format Loewner matrix (SFLM) interpolation method (Section 2.2.3). Some existing parametric macromodeling methods are then reviewed (Section 2.3) in order to provide a bit of context for the parametric methods that are proposed. Following the parametric macromodeling review is the passivity enforcement review (Section 2.4), where we review the standard passivity check methods, the passivity perturbation methods and the existing error control schemes for the perturbation methods. The passivity checking method reviewed is the Hamiltonian eigenvalue passivity check required for robust passivity checking (Section 2.4.1). The perturbation methods reviewed are the Hamiltonian eigenvalue perturbation method (Section 2.4.2) and the transfer function singular value perturbation method (Section 2.4.3). The error control schemes reviewed are the standard gramian based error control (Section 2.4.4) and the filtered gramian error control (Section 2.4.5).

Once the reviews are finished, we first present the passivity enforcement contribution chapter (Chapter 3). The contributions are the two error control schemes of the discrete gramian scheme (Section 3.1) and the Loewner Matrix system method built-in error control scheme (Section 3.2). Once both schemes are presented, with the two reviewed perturbation methods, a set passivity enforcement simulations is performed using both the reviewed and proposed error control schemes and a comparison analysis is performed to validate the contribution (Section 3.3).

Following the presentation of the passivity enforcement contribution chapter, we continue with presenting the parametric macromodeling contributions (Chapter 4, Chapter 5). The first method presented is the polynomial barycentric rational Loewner matrix (PBRLM) interpolation method which is a direct parametric extension of the BRLM method (Chapter 4). The algorithm is presented first from set-up (Section 4.1) to derivations and definitions (Section 4.2, Section 4.3) to simulation (Section 4.4). Once the showcase of PBRLM is done, the second method by the name of parametric system format Loewner matrix (PSFLM) method is presented as the direct parametric extension of the SFLM method (Section 5). The algorithm is presented first from set-up (Section 5.1) to derivations and definitions (Section 5.2) to simulation (Section 5.3).

Chapter 2

Current State of Art

In this chapter, the various required definitions and theories to present the contributions of this thesis are reviewed. First, the concept of frequency data to describe a circuit system behaviour is presented and the type of frequency data used is decided. Afterwards, the concept of frequency data interpolation by macromodels is presented and the macromodel form used throughout the thesis is also defined. With the models described, the key system property of passivity is presented alongside the property of stability from which the concept of passivity is built upon. Once the base system models and its properties are defined, we review the specific univariate macromodeling techniques of the Loewner Matrix methods. A few parametric macromodeling of relevance are presented to give a sense how parametric modeling is conducted. Finally, the process of passivity enforcement is reviewed from passivity checking method to perturbation methods to error control schemes.

2.1 Macromodeling System Definition and Key Properties

This section reviews and defines key points of macromodels necessary to this thesis: the type of frequency points representation, the format of system representing the macromodel proper, and the important properties of stability and passivity of the macromodels.

2.1.1 Frequency Data Representations

Linear electric networks can be described through frequency domain data which characterizes the I/O behaviour of the networks. The two types of frequency data that occur in the thesis are the admittance/impedance (Y/Z) parameters and the scattering (S) parameters, though the greater majority of occurrences are of the latter type. Due to the likeliness between Y and Z parameters, we use Y parameters to refer both.

Standard 2-Port Description

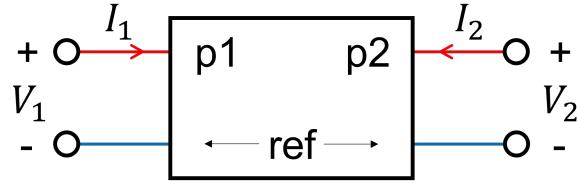


Figure 2.1: Generic 2-port black-box network voltage and current description.

The first type of frequency domain data we look at is the Y -parameter, and we utilize the simple case of a generic 2-port black-box portrayed on Figure 2.1 as template for the explanation. The Y -parameters relate the port voltages with the port currents as follows [116]:

$$\begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} \quad (2.1)$$

The Y -parameters are obtained by shorting one of the 2 terminations at a time and measuring the corresponding voltage to current relations:

$$\begin{aligned} Y_{11} &= \frac{I_1}{V_1} \Big|_{V_2=0} & Y_{12} &= \frac{I_1}{V_2} \Big|_{V_1=0} \\ Y_{21} &= \frac{I_2}{V_1} \Big|_{V_2=0} & Y_{22} &= \frac{I_2}{V_2} \Big|_{V_1=0} \end{aligned} \quad (2.2)$$

Standard 2-Port Description

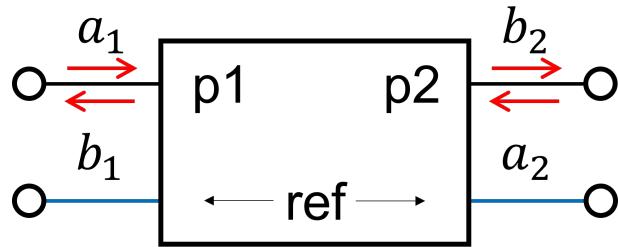


Figure 2.2: Generic 2-port black-box network incident and reflected power waves description.

We now look at the S -parameter, and we utilize again a generic 2-port black-box network portrayed on Figure 2.2. The variables a_1 and a_2 are the incident power waves at port 1 and 2, respectively, and the variables b_1 and b_2 are the reflected power waves at port 1 and 2, respectively. Instead of voltages and currents, S -parameters relate the incident and reflected waves as follows [116,117]:

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \quad (2.3)$$

The S -parameters are computed by matching termination impedance at one port at a time. When the termination impedance matches at p2, we would have no reflection

at p2 ($a_2 = 0$) and can compute:

$$S_{11} = \frac{b_1}{a_1}, \quad S_{21} = \frac{b_2}{a_1} \quad (2.4)$$

When the termination impedance matches at p1, we would have no reflection at p1 ($a_1 = 0$) and can compute:

$$S_{12} = \frac{b_1}{a_2}, \quad S_{22} = \frac{b_2}{a_2} \quad (2.5)$$

The great majority of examples used in the thesis are S-parameter based due to practicality of using S -parameters rather than Y/Z -parameters to describe network behaviour at broadband range. This is because at high frequency, it is impractical to expect to be able to short terminations to collect measurements for Y -parameters while you only need to perform impedance matching to obtain S -parameters.

2.1.2 Linear Time-Invariant Systems

Macromodels are generated using frequency data sampled over the bandwidth of a system. Such a system is assumed to be a linear passive \bar{p} -network modeling Input/Output (I/O) behaviour of an original design using its scattering (S) parameter data or admittance/impedance (Y) parameter data. In this thesis, the macromodels take the form of a causal LTI continuous time system which can be described under the descriptor system (DS) form [118]:

$$\begin{aligned} \mathbf{E}_{ds} \dot{\mathbf{x}}(t) &= \mathbf{A}_{ds} \mathbf{x}(t) + \mathbf{B}_{ds} \mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}_{ds} \mathbf{x}(t) + \mathbf{D}_{ds} \mathbf{u}(t) \end{aligned} \quad (2.6)$$

where $\mathbf{x}(t) \in \mathbb{R}^{n \times 1}$ is the state vector, $\mathbf{u}(t) \in \mathbb{R}^{\bar{m} \times 1}$ is the incident power vector at the ports, and $\mathbf{y}(t) \in \mathbb{R}^{\bar{p} \times 1}$ is the reflected power vector at the ports. Unless

specified otherwise, all examples in this thesis have matching number \bar{m} of input ports and number \bar{p} of output ports, but we derive the equations in the general sense with the two numbers considered distinct. The system is described by the matrices $\mathbf{E}_{ds} \in \mathbb{R}^{n \times n}$, $\mathbf{A}_{ds} \in \mathbb{R}^{n \times n}$, $\mathbf{B}_{ds} \in \mathbb{R}^{n \times \bar{m}}$, $\mathbf{C}_{ds} \in \mathbb{R}^{\bar{p} \times n}$, and $\mathbf{D}_{ds} \in \mathbb{R}^{\bar{p} \times \bar{m}}$, where n is the order of the system. The I/O relation of the system (2.6) can be expressed in the frequency domain by its transfer function:

$$\mathbf{G}(s) = \mathbf{C}_{ds}(s\mathbf{E}_{ds} - \mathbf{A}_{ds})^{-1}\mathbf{B}_{ds} + \mathbf{D}_{ds} \quad (2.7)$$

where $\mathbf{G}(s)$ can be either the Y/Z or S -parameter matrix of the system evaluated at frequency s .

In various contexts such as for passivity enforcement, a system in the form (2.6) with singular \mathbf{E}_{ds} must be treated differently from a system of the same form with non-singular \mathbf{E}_{ds} , but in this thesis, the focus is solely on the latter case. This means that the DS we consider in this thesis can be easily converted into a state-space or regular system (RS) having the form [119, 120]:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}_{rs}\mathbf{x}(t) + \mathbf{B}_{rs}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}_{rs}\mathbf{x}(t) + \mathbf{D}_{rs}\mathbf{u}(t) \end{aligned} \quad (2.8)$$

with the transfer function:

$$\mathbf{G}(s) = \mathbf{C}_{rs}(s\mathbf{I} - \mathbf{A}_{rs})^{-1}\mathbf{B}_{rs} + \mathbf{D}_{rs} \quad (2.9)$$

The translation of DS matrices to RS matrices is performed as follow:

$$\mathbf{A}_{rs} = \mathbf{E}_{ds}^{-1}\mathbf{A}_{ds}, \quad \mathbf{B}_{rs} = \mathbf{E}_{ds}^{-1}\mathbf{B}_{ds}, \quad \mathbf{C}_{rs} = \mathbf{C}_{ds}, \quad \mathbf{D}_{rs} = \mathbf{D}_{ds} \quad (2.10)$$

Throughout the remaining part of the thesis, all instances of LTI systems are represented by the general system matrices $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ for the RS form and $\{\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ for the DS form. The thesis presents its content using either the DS or RS form, but with the assumption that \mathbf{E}^{-1} always exists. The above assumption does mean that the theories presented in the thesis are only truly applicable to RS, but due to circumstances of certain theories presented, it is more convenient to write the LTI systems in DS form at times.

2.1.3 Stability Condition

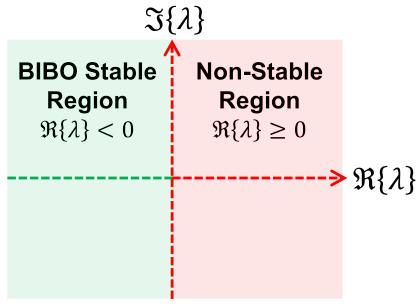


Figure 2.3: Pole stability area portrayal.

A property of critical importance in an LTI system is its stability. In an informal way, one may view it as the property of a system being unable to create an infinite gain. Within this thesis, the bounded-input bounded-output (BIBO) definition for stability is used to describe the property [121], so whenever we refer to system stability, the BIBO definition is implied. For our continuous-time causal LTI systems, the requirements for stability is to have its frequency domain transfer function (2.7) or (2.9) being finite in the right half-plane of the complex frequency s Cartesian plane (real part x-axis, imaginary part y-axis), including the imaginary axis. This translates to the requirement that all poles of (2.7) or (2.9) must lie strictly in the open

left half-plane of the s domain, which turns into the concrete stability conditions:

$$\text{DS case: } \Re\{\lambda\} < 0, \quad \forall \lambda \in \text{eig}\{\mathbf{A}, \mathbf{E}\} \quad (2.11)$$

$$\text{RS case: } \Re\{\lambda\} < 0, \quad \forall \lambda \in \text{eig}\{\mathbf{A}\} \quad (2.12)$$

where the eigenvalue sets $\text{eig}\{\mathbf{A}, \mathbf{E}\}$ and $\text{eig}\{\mathbf{A}\}$ are the system poles for DS and RS, respectively. The stability condition can be observed on Figure 2.3. Stability is not a primary point in the contributions of this thesis, but the need to achieve component wise passivity is largely for the purpose of general system stability and many passivity related conditions are based on the assumption of system stability.

2.1.4 Passivity Condition

In the most general sense, a system that is passive is a system which does not generate energy: a passive system does not output more energy than it is given. As such, the property of passivity strictly implies that of stability, though the reverse implication isn't true. More concrete formulation of the passivity condition can be given through systems represented by a macromodel of the form (2.6) or (2.8). Note that in this section, we concentrate on the formulations for the RS format LTI system (2.8), though these formulations are equally applicable to the descriptor system (2.6) in the case when \mathbf{E} is non-singular.

S-parameter System Passivity

Here, we consider a system (2.8) built upon S -parameter data, where the transfer function (2.9) evaluates the S -parameter itself, that is, $\mathbf{S}(s) = \mathbf{H}(s)$.

Theorem 2.1.1. *For the LTI system in the form given in the regular system format (2.8), we have passivity of the system if and only if its S-parameters $\mathbf{S}(s)$ satisfy all of the following conditions [122, 123]:*

1. $\mathbf{S}(s)$ has no poles $\forall \Re\{s\} > 0$
2. $\mathbf{I}_{n \times n} - \mathbf{S}(j\omega)^H \mathbf{S}(j\omega)$ is positive semi-definite, $\forall \omega \in \mathbb{R}$.
3. $\mathbf{S}^*(j\omega) = \mathbf{S}(-j\omega)$, $\forall \omega \in \mathbb{R}$

where \mathbf{S}^* denotes the complex conjugate of \mathbf{S} .

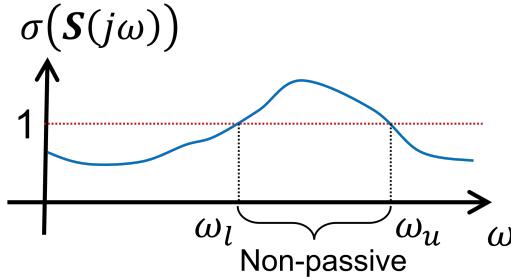


Figure 2.4: Passivity violation frequency region ($[\omega_l, \omega_u]$) illustration.

The fulfillment of all 3 conditions in Theorem 2.1.1 is equivalent to the condition that the transfer function is bounded-real, that is, the condition of passivity is equivalent to the condition of bounded-realness of $\mathbf{S}(s)$ [86]. For system (2.8) that is real and stable, conditions 1 and 3 are necessarily satisfied and we only need to fulfil condition 2 in order to ascertain the passivity of the system. A simple passivity check method based on condition 2 can be derived by observing the fact that condition 2 is equivalent to the condition:

$$\sup_{\omega \in \mathbb{R}} \{\sigma_{max}(\mathbf{S}(j\omega))\} \leq 1 \quad (2.13)$$

where $\sigma_{max}(\mathbf{S}(j\omega))$ denotes the largest singular value of $\mathbf{S}(j\omega)$. This condition, which is illustrated by a simple example on Figure 2.4, indicates that for a stable system (2.8) to be passive, the singular values of $\mathbf{S}(j\omega)$ must be below 1 at all frequency ω , so a simple frequency scan can be used to provide fast preliminary passivity check.

An alternative method for determining the passivity of the system is given by the Linear Matrix Inequality (LMI) [86]:

$$\mathbf{P} > 0, \quad \begin{bmatrix} \mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} + \mathbf{C}^T \mathbf{C} & \mathbf{P} \mathbf{B} + \mathbf{C}^T \mathbf{D} \\ \mathbf{B}^T \mathbf{P} + \mathbf{D}^T \mathbf{C} & \mathbf{D}^T \mathbf{D} - \mathbf{I}_{\bar{p} \times \bar{m}} \end{bmatrix} \leq 0 \quad (2.14)$$

There exists a solution $\mathbf{P}^T = \mathbf{P} \in \mathbb{R}^{n \times n}$ if and only if system (2.8) is passive.

\mathbf{Y} -parameter System Passivity

Here, we consider a system (2.8) built upon Y -parameter data, where the transfer function (2.9) evaluates the Y -parameter itself, that is, $\mathbf{Y}(s) = \mathbf{H}(s)$.

Theorem 2.1.2. *Let $\mathbf{Y}(s)$ be a square matrix rational function of complex variable s . Then $\mathbf{Y}(s)$ is positive-real if and only if all following conditions are met: [122, 124]:*

1. $\mathbf{Y}(s)$ has no poles ρ where $\Re\{\rho\} > 0$
2. $\mathbf{Y}^*(s) = \mathbf{Y}(s^*)$, $\forall \Re\{s\} > 0$
3. $\mathbf{G}(j\omega) = \mathbf{Y}(j\omega) + \mathbf{Y}^H(j\omega) \geq 0$, if $j\omega$ is not a pole for $\omega \in \mathbb{R}$.
4. All imaginary poles $\rho = j\omega$ of the rational function $\mathbf{Y}(s)$ are simple poles and each has a corresponding residual matrix that is positive-semi definite.

2.1.5 After Note

One of the key reason for the importance of system passivity is the fact that combination of passive systems result in a passive (and thus stable) system [125], whereas combination of stable systems does not necessarily result in a stable system. As such, passivity enforcement becomes an important step of the macromodeling process because even when a physical system is passive, the macromodels generated from its extracted frequency data is not guaranteed to be passive due to various factors such as numerical errors or the nature of how macromodeling methods select the poles of the system.

This thesis exclusively concentrates on S-parameter data, so we focus on the passivity condition given by the 3 conditions of Theorem 2.1.1. As only the passivity property is directly relevant to the contributions of the thesis rather than the stability property, and because many steps of the passivity enforcement process require the system to be initially stable, all LTI systems in this paper are implied stable unless specified otherwise. As such, only the 2nd condition of Theorem 2.1.1 is fallible, which means the passivity condition (2.13) is the only one that needs to be checked and fulfilled.

2.2 Loewner Matrix Review

In this section, we shall review the fundamentals of the LM theory and a number of its relevant applications. The theories given here are limited to portions that we need to present the contributions of the work done, and thus is not exhaustive. Readers are referred to [49,52,71–73] for the more complete theory of Loewner Matrix methods. The overall objective of the LM based methods presented in this section is

to create an approximation system which properly approximates an initial data set:

$$\mathcal{S} = \{s_1, \dots, s_{\bar{t}}\} \quad \mathcal{H} = \{\mathbf{H}(s_t) \in \mathbb{C}^{\bar{p} \times \bar{m}}; s_t \in \mathcal{S}\}, \quad (2.15)$$

where \bar{p} and \bar{m} are the number of outputs and inputs, respectively.

2.2.1 Loewner Matrix Definition

At the base of the LM definition is the partitioning of the initial data set \mathcal{S} (2.15) into two data partitions, which we define as frequency partitions Λ and \mathcal{M} with their associated data partitions \mathcal{H}_Λ and $\mathcal{H}_{\mathcal{M}}$, respectively:

$$\begin{aligned} \Lambda &= \{\lambda_1, \dots, \lambda_{\bar{j}}\} & \mathcal{H}_\Lambda &= \{\mathbf{H}(\lambda_j); \lambda_j \in \Lambda\}, \\ \mathcal{M} &= \{\mu_1, \dots, \mu_{\bar{i}}\} & \mathcal{H}_{\mathcal{M}} &= \{\mathbf{H}(\mu_i); \mu_i \in \mathcal{M}\}. \end{aligned} \quad (2.16)$$

where

- $\mathcal{S} = \Lambda \cup \mathcal{M}$, $\Lambda \cap \mathcal{M} = \emptyset$
- $\mathcal{H} = \mathcal{H}_\Lambda \cup \mathcal{H}_{\mathcal{M}}$
- \bar{j} and \bar{i} are the size of partitions Λ and \mathcal{M} , respectively.
- $\mathbf{H}(s) \in \mathbb{C}^{\bar{p} \times \bar{m}}$
- \bar{p} and \bar{m} are the number of outputs and inputs, respectively.

The Loewner Matrix is defined as:

$$\mathbb{L} = \begin{bmatrix} \mathbb{L}_{\{1,1\}} & \mathbb{L}_{\{1,2\}} & \cdots & \mathbb{L}_{\{1,\bar{j}\}} \\ \mathbb{L}_{\{2,1\}} & \mathbb{L}_{\{2,2\}} & \cdots & \mathbb{L}_{\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{\{\bar{i},1\}} & \mathbb{L}_{\{\bar{i},2\}} & \cdots & \mathbb{L}_{\{\bar{i},\bar{j}\}} \end{bmatrix} \quad (2.17)$$

where

$$\mathbb{L}_{\{i,j\}} = \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \quad (2.18)$$

where we have defined a shorthand for the data of partitions Λ and \mathcal{M} :

$$\mathbf{H}_{\lambda_j} = \mathbf{H}(\lambda_j) \quad (2.19)$$

$$\mathbf{H}_{\mu_i} = \mathbf{H}(\mu_i) \quad (2.20)$$

The meaning and practicality of \mathbb{L} will become apparent as we present the methods which use \mathbb{L} to interpolate the data sets Λ and \mathcal{M} in the subsequent sections.

2.2.2 Barycentric Rational Loewner Matrix Interpolation Method

In this section, we describe the BRLM method. The method is first presented with the single input single output (SISO) case ($\bar{p} = 1, \bar{m} = 1$) and then extended towards the general data case as well as adding complex conjugate set of poles to ensure the system is physical.

Scalar BRLM Equation

For the span of this subsection, we shall assume that $\bar{p} = 1$ and $\bar{m} = 1$ in order to provide an easier to follow derivation process. Suppose we use frequency data of partition Λ to define the barycentric interpolation equation, an offshoot of the Lagrange polynomial interpolation formula [35, 71, 73, 126]:

$$\mathbf{G}(s) = \frac{\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\lambda_j}}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{s - \lambda_j}} \quad (2.21)$$

which, by definition, achieves perfect interpolation of all points in Λ for any non-trivial set of coefficients $\{c_j\}_{j=1}^{\bar{j}}$:

$$\mathbf{G}(\lambda_j) = \mathbf{H}_{\lambda_j}, \quad \forall \lambda_j \in \Lambda, \quad \{c_j\}_{j=1}^{\bar{j}} \neq \emptyset \quad (2.22)$$

The Loewner Matrix can be derived when attempting to determine a suitable coefficient set $\{c_j\}_{j=1}^{\bar{j}}$ such that

$$\mathbf{G}(\mu_i) = \mathbf{H}_{\mu_i}, \quad \forall \mu_i \in \mathcal{M} \quad (2.23)$$

We manipulate equation (2.23) as follows:

$$\begin{aligned} \mathbf{G}(\mu_i) &= \frac{\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{\mu_i - \lambda_j}} = \mathbf{H}_{\mu_i} \\ &\frac{\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{\mu_i - \lambda_j}} = \mathbf{H}_{\mu_i} \cdot \frac{\sum_{j=1}^{\bar{j}} \frac{c_j}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{\mu_i - \lambda_j}} \\ &\frac{\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{\mu_i - \lambda_j}} = \frac{\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\mu_i}}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{\mu_i - \lambda_j}} \\ &\frac{\sum_{j=1}^{\bar{j}} \frac{c_j (\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j})}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{\mu_i - \lambda_j}} = 0 \end{aligned}$$

$$\sum_{j=1}^{\bar{J}} \frac{c_j(\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j})}{\mu_i - \lambda_j} = 0 \quad (2.24)$$

The condition (2.24) is the fitting condition for the coefficient set $\{c_j\}_{j=1}^{\bar{J}}$ to properly approximate frequency point μ_i 's data, which can be written in matrix equivalent form:

$$\begin{bmatrix} (\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_1}) & (\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_2}) & \dots & (\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_{\bar{J}}}) \\ \mu_i - \lambda_1 & \mu_i - \lambda_2 & & \mu_i - \lambda_{\bar{J}} \end{bmatrix} \mathbf{c} = 0, \quad \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_{\bar{J}} \end{bmatrix} \quad (2.25)$$

If we assemble all cases of condition (2.25) for $\mu_i \in \mathcal{M}$, we obtain the following system of equations:

$$\begin{bmatrix} (\mathbf{H}_{\mu_1} - \mathbf{H}_{\lambda_1}) & (\mathbf{H}_{\mu_1} - \mathbf{H}_{\lambda_2}) & \dots & (\mathbf{H}_{\mu_1} - \mathbf{H}_{\lambda_{\bar{J}}}) \\ \mu_1 - \lambda_1 & \mu_1 - \lambda_2 & & \mu_1 - \lambda_{\bar{J}} \\ (\mathbf{H}_{\mu_2} - \mathbf{H}_{\lambda_1}) & (\mathbf{H}_{\mu_2} - \mathbf{H}_{\lambda_2}) & \dots & (\mathbf{H}_{\mu_2} - \mathbf{H}_{\lambda_{\bar{J}}}) \\ \mu_2 - \lambda_1 & \mu_2 - \lambda_2 & & \mu_2 - \lambda_{\bar{J}} \\ \vdots & \vdots & \ddots & \vdots \\ (\mathbf{H}_{\mu_{\bar{i}}} - \mathbf{H}_{\lambda_1}) & (\mathbf{H}_{\mu_{\bar{i}}} - \mathbf{H}_{\lambda_2}) & \dots & (\mathbf{H}_{\mu_{\bar{i}}} - \mathbf{H}_{\lambda_{\bar{J}}}) \\ \mu_{\bar{i}} - \lambda_1 & \mu_{\bar{i}} - \lambda_2 & & \mu_{\bar{i}} - \lambda_{\bar{J}} \end{bmatrix} \mathbf{c} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (2.26)$$

We recognize the left-hand side matrix of the linear system of equations (2.26) as the Loewner Matrix defined by (2.17). A solution \mathbf{c} to (2.26) can be obtained through simple singular value decomposition (SVD) of \mathbb{L} :

$$\mathbb{L} = \mathbf{U} \mathbf{S} \mathbf{V}^H \quad (2.27)$$

where \mathbf{U} is the left singular vector matrix, \mathbf{S} is the diagonal matrix with the singular values of \mathbb{L} spread on the diagonal from largest to smallest, and \mathbf{V} is the right singular

vector matrix. Any column vector \mathbf{v}_z of \mathbf{V} can potentially replace \mathbf{c} in (2.26), and produces the matrix product

$$\mathbb{L} \cdot \mathbf{v}_z = \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_i \end{bmatrix} \quad (2.28)$$

Ideally, the error $\boldsymbol{\epsilon}$ is the null vector as prescribed by our objective function (2.26), but normally, $\boldsymbol{\epsilon}$ has a non-trivial norm. The norm of $\boldsymbol{\epsilon}$ can be directly assessed because \mathbf{v}_z is a right singular vector of \mathbb{L} , and as such, the following property is true [35]:

$$\|\sigma_z\|_F = \|\mathbb{L}\mathbf{v}_z\|_F = \|\boldsymbol{\epsilon}\|_F \quad (2.29)$$

where σ_z is the singular value associated to \mathbf{v}_z . Indeed, by virtue of the singular value property (2.29), the smaller the singular value associated to the chosen \mathbf{v}_z to replace \mathbf{c} , the smaller the overall approximation error of points in partition \mathcal{M} , and potentially the better the final barycentric approximation equation (2.21) at approximating the original system. The final barycentric equation is able to perfectly approximate points in Λ while approximating points in \mathcal{M} to a degree of accuracy indicated by singular values of \mathbb{L} . Of course, there is no guarantee of stability to the system resulting from the BRLM method, and enforcement methods are required to remedy this, though that is not discussed in this Thesis.

General BRLM Equation

Having defined the SISO case of the barycentric rational Loewner matrix (BRLM) equation, we now extend the equation to the multiple inputs multiple outputs (MIMO) case. The reason why a separate version of the Loewner Matrix is needed when we have $\bar{m} \geq 1$ is because the matrix \mathbb{L} 's dimension becomes $\bar{p}\bar{\imath} \times \bar{m}\bar{\jmath}$. If we follow the barycentric method's procedure presented in the SISO case, we would end up trying to replace the coefficient vector $\mathbf{c} \in \mathbb{C}^{\bar{j} \times 1}$ with a mismatched size right singular vector $\mathbf{v}_z \in \mathbb{C}^{\bar{m}\bar{\jmath} \times 1}$. The full size $\mathbb{L} \in \mathbb{C}^{\bar{p}\bar{\imath} \times \bar{m}\bar{\jmath}}$ basically allocates one coefficient to not only each data matrix, but one for each of the data matrix columns. Incidentally, the number of outputs \bar{p} does not cause any such problem because we are trying to use only the right singular vectors of \mathbb{L} .

To adjust the matrix \mathbb{L} for the general case, we use the simple trick of transforming the data matrices $\mathbf{H}(s)$ into a single column vector by stacking its column vectors one on top of each other from leftmost to rightmost:

$$\mathbf{H}(s) \in \mathbb{C}^{\bar{p} \times \bar{m}} \rightarrow \text{vec}(\mathbf{H}(s)) \in \mathbb{C}^{\bar{p}\bar{m} \times 1} \quad (2.30)$$

The $\text{vec}(\cdot)$ operator then changes the barycentric equation (2.21) to

$$\text{vec}(\mathbf{G}(s)) = \frac{\sum_{j=1}^{\bar{j}} \frac{c_j \cdot \text{vec}(\mathbf{H}_{\lambda_j})}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{s - \lambda_j}} \quad (2.31)$$

which, following the same derivations steps taken in the scalar case derivation subsection, would result in the LM of the following form:

$$\mathbb{L} = \begin{bmatrix} \frac{\text{vec}(\mathbf{H}_{\mu_1} - \mathbf{H}_{\lambda_1})}{\mu_1 - \lambda_1} & \frac{\text{vec}(\mathbf{H}_{\mu_1} - \mathbf{H}_{\lambda_2})}{\mu_1 - \lambda_2} & \dots & \frac{\text{vec}(\mathbf{H}_{\mu_1} - \mathbf{H}_{\lambda_{\bar{j}}})}{\mu_1 - \lambda_{\bar{j}}} \\ \frac{\text{vec}(\mathbf{H}_{\mu_2} - \mathbf{H}_{\lambda_1})}{\mu_2 - \lambda_1} & \frac{\text{vec}(\mathbf{H}_{\mu_2} - \mathbf{H}_{\lambda_2})}{\mu_2 - \lambda_2} & \dots & \frac{\text{vec}(\mathbf{H}_{\mu_2} - \mathbf{H}_{\lambda_{\bar{j}}})}{\mu_2 - \lambda_{\bar{j}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\text{vec}(\mathbf{H}_{\mu_{\bar{i}}} - \mathbf{H}_{\lambda_1})}{\mu_{\bar{i}} - \lambda_1} & \frac{\text{vec}(\mathbf{H}_{\mu_{\bar{i}}} - \mathbf{H}_{\lambda_2})}{\mu_{\bar{i}} - \lambda_2} & \dots & \frac{\text{vec}(\mathbf{H}_{\mu_{\bar{i}}} - \mathbf{H}_{\lambda_{\bar{j}}})}{\mu_{\bar{i}} - \lambda_{\bar{j}}} \end{bmatrix} \in \mathbb{C}^{\bar{p}\bar{m} \times \bar{j}} \quad (2.32)$$

The vectorized \mathbb{L} as defined by (2.32) allows us to use one of its right-singular vector $\mathbf{v}_z \in \mathbb{C}^{\bar{j} \times 1}$ as the coefficient vector with number of coefficients \bar{j} matching the number of data matrices defining the barycentric equation (2.21). Note that such a coefficient set can be used in either of the barycentric equation forms (2.21) or (2.31) because the difference between the two is purely in formatting.

Real Loewner Matrix

The transfer function $\mathbf{G}(s)$ needs to achieve the realness condition:

$$\mathbf{G}(s)^* = \mathbf{G}(s^*) \quad (2.33)$$

in order to be able to result in real valued impulse-response. In order to achieve this, we can make sure that all complex poles of the system come as complex conjugate pairs. Given the structure of the barycentric (2.21), we see that it is the denominator

$$\sum_{j=1}^{\bar{j}} \frac{c_j}{s - \lambda_j} \quad (2.34)$$

which dictates the poles of the system, which are equivalent to the zeros of (2.34). If we design (2.34) to be complex conjugate, its complex zeros would have to be complex

conjugate, and in turn renders the complex poles of (2.21) complex conjugate. To achieve complex conjugacy of (2.34), we can pad the frequency data partitions with their respective complex conjugates in the following manner [51]:

$$\begin{aligned}\Lambda &= \{\lambda_1, \lambda_1^*, \lambda_2, \lambda_2^*, \dots, \lambda_{\bar{j}}, \lambda_{\bar{j}}^*\} & \mathcal{H}_\Lambda &= \{\mathbf{H}(\lambda_j); \lambda_j \in \Lambda\} & \mathbf{H}(\lambda_j)^* &= \mathbf{H}(\lambda_j^*) \\ \mathcal{M} &= \{\mu_1, \mu_1^*, \mu_2, \mu_2^* \dots, \mu_{\bar{i}}, \mu_{\bar{i}}^*\} & \mathcal{H}_\mathcal{M} &= \{\mathbf{H}(\mu_i); \mu_i \in \mathcal{M}\} & \mathbf{H}(\mu_i)^* &= \mathbf{H}(\mu_i^*)\end{aligned}\quad (2.35)$$

As a first direct result of the padded frequency partitions, the denominator (2.34) now has the pattern:

$$\sum_{j=1}^{\bar{j}} \left[\frac{c_{ja}}{s - \lambda_j} + \frac{c_{jb}}{s - \lambda_j^*} \right] \quad (2.36)$$

Incidentally, the numerator of (2.21) also obtains a similar pattern through this process:

$$\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \rightarrow \sum_{j=1}^{\bar{j}} \left[\frac{c_{ja} \mathbf{H}_{\lambda_j}}{s - \lambda_j} + \frac{c_{jb} \mathbf{H}_{\lambda_j}^*}{s - \lambda_j^*} \right] \quad (2.37)$$

Thus, the next step is ensuring complex conjugacy between the coefficient pairs $\{(c_{ja}, c_{jb})\}_{j=1}^{j=\bar{j}}$. The problem in directly performing SVD on the matrix \mathbb{L} , currently complex, is that the right singular vectors \mathbf{v}_z 's are not guaranteed to have complex conjugate terms due to numerical limitations, and replacing the coefficient vector \mathbf{c} with one such \mathbf{v}_z will not yield complex conjugate (2.36).

The solution is to transform \mathbb{L} into a real matrix using unitary transformation matrices. The detailed process of real LM alteration is presented in Appendix B and specifically in the appendix Section B.4, so we shall quickly go over the key points. Given the structure of the general $\mathbb{L} \in \mathbb{C}^{2\bar{i}\bar{p}\bar{m} \times 2\bar{j}}$ as defined by (2.32), it can be turned

real through the following transformation

$$\mathbb{L}_{re} = \mathbf{T}_{(\bar{p}\bar{m}\langle\bar{i}\rangle)}^H \cdot \mathbb{L} \cdot \mathbf{T}_{(1\langle\bar{j}\rangle)} \in \mathbb{R}^{i\bar{p}\bar{m} \times \bar{j}} \quad (2.38)$$

using the general transformation matrix defined at (B.10) that we rewrite here:

$$\mathbf{T}_{(x\langle y\rangle)} \in \mathbb{C}^{2yx \times 2yx} = \\ \frac{1}{\sqrt{2}} \begin{bmatrix} \mathcal{T}_{(x)} & & & \\ & \mathcal{T}_{(x)} & & \\ & & \ddots & \\ & & & \mathcal{T}_{(x)} \end{bmatrix}, \quad \mathcal{T}_{(x)} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{x \times x} & -1i \cdot \mathbf{I}_{x \times x} \\ \mathbf{I}_{x \times x} & 1i \cdot \mathbf{I}_{x \times x} \end{bmatrix} \in \mathbb{C}^{2x \times 2x} \quad (2.39)$$

for the variable positive integers x and y . The resulting real LM matrix \mathbb{L}_{re} has the more direct formula:

$$\mathbb{L}_{re} = \begin{bmatrix} \mathbb{L}_{re\{1,1\}} & \mathbb{L}_{re\{1,2\}} & \cdots & \mathbb{L}_{re\{1,\bar{j}\}} \\ \mathbb{L}_{re\{2,1\}} & \mathbb{L}_{re\{2,2\}} & \cdots & \mathbb{L}_{re\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{re\{\bar{i},1\}} & \mathbb{L}_{re\{\bar{i},2\}} & \cdots & \mathbb{L}_{re\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{2i\bar{p}\bar{m} \times 2\bar{j}} \quad (2.40)$$

$$\mathbb{L}_{re\{i,j\}} = \begin{bmatrix} \Re\{\mathbb{L}_{\alpha\{i,j\}}\} + \Re\{\mathbb{L}_{\beta\{i,j\}}\} & \Im\{\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\mathbb{L}_{\beta\{i,j\}}\} \\ -\Im\{\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\mathbb{L}_{\beta\{i,j\}}\} & \Re\{\mathbb{L}_{\alpha\{i,j\}}\} - \Re\{\mathbb{L}_{\beta\{i,j\}}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p}\bar{m} \times 2}$$

$$\mathbb{L}_{\alpha\{i,j\}} = \frac{\text{vec}(\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j})}{\mu_i - \lambda_j}, \quad \mathbb{L}_{\beta\{i,j\}} = \frac{\text{vec}(\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}^*)}{\mu_i - \lambda_j^*}$$

\mathbb{L}_{re} would then undergo SVD, from where the selected right singular vector \mathbf{v}_{rez} would be purely real. Now map the reverse transformation of $\mathbf{T}_{(\bar{p}\bar{m}\langle\bar{i}\rangle)}$ and $\mathbf{T}_{(1\langle\bar{j}\rangle)}$ for

the SVD equation:

$$\begin{aligned}
 \mathbb{L}_{re} &= \mathbf{U}_{re} \cdot \mathbf{S} \cdot \mathbf{V}_{re}^H \\
 \mathbf{U}_{re}^H \cdot \mathbb{L}_{re} \cdot \mathbf{V}_{re} &= \mathbf{S} \\
 \mathbf{U}_{re}^H \cdot \mathbf{T}_{(\bar{p}\bar{m}\langle\bar{i}\rangle)}^H \cdot \mathbb{L} \cdot \mathbf{T}_{(1\langle\bar{j}\rangle)} \cdot \mathbf{V}_{re} &= \mathbf{S} \\
 \mathbf{U}^H \cdot \mathbb{L} \cdot \mathbf{V} &= \mathbf{S}
 \end{aligned} \tag{2.41}$$

where

$$\mathbf{U} = \mathbf{T}_{(\bar{p}\bar{m}\langle\bar{i}\rangle)} \cdot \mathbf{U}_{re}, \quad \mathbf{V} = \mathbf{T}_{(1\langle\bar{j}\rangle)} \cdot \mathbf{V}_{re} \tag{2.42}$$

Thus, to revert the chosen right singular vector \mathbf{v}_{re_z} of \mathbb{L}_{re} to the right singular vector \mathbf{v}_z of \mathbb{L} , we simply perform the derived transformation:

$$\mathbf{v}_z = \mathbf{T}_{(1\langle\bar{j}\rangle)} \cdot \mathbf{v}_{re_z} \tag{2.43}$$

The resulting \mathbf{v}_z would then replace the coefficient vector with the following structure:

$$\mathbf{c}^T = \left[c_1 \ c_1^* \ c_2 \ c_2^* \ \cdots \ c_{\bar{j}} \ c_{\bar{j}}^* \right]^T \tag{2.44}$$

which would ensure that the denominator (2.36) of the barycentric equation (2.21) would be complex conjugate, in turn ensuring the complex zeros of the denominator come in complex conjugate pairs, which means the complex poles of (2.21) also come in complex conjugate pairs.

Pole-Zero to State-Space Translation

In this thesis, macromodels were introduced in the descriptor system format or the state-space system format in Section 2.1.2. The use of barycentric rational

equation format (2.21) does not contradict the state-space format as a direct translation exists. The barycentric rational equation translation process to its real matrix state-space transfer function form equivalent is detailed in Appendix A, and we will only go through the outcome. Essentially, the main purpose of the barycentric form is to determine a suitable set of zeros and poles of the approximation system which can then be translated from the pole-zero rational equation form to its state-space equivalent form. The final result is the transformation of (2.21) rewritten here:

$$\mathbf{G}(s) = \frac{\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\lambda_j}}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{s - \lambda_j}} \quad (2.45)$$

to its state-space equivalent form given by:

$$\mathbf{G}(s) = \mathbf{C}_r (s \mathbf{I}_{(\bar{a}+2\bar{b})\bar{m} \times (\bar{a}+2\bar{b})\bar{m}} - \mathbf{A}_r)^{-1} \mathbf{B}_r + \mathbf{D} \quad (2.46)$$

where \bar{a} is the number of real poles, \bar{b} is the number of pair of complex conjugate poles, $\bar{a} + 2\bar{b} = \bar{j} - 1$ is the total number of poles, and:

$$\begin{aligned} \mathbf{A}_r &\in \mathbb{R}^{(\bar{a}+2\bar{b})\bar{m} \times (\bar{a}+2\bar{b})\bar{m}} & \mathbf{B}_r &\in \mathbb{R}^{(\bar{a}+2\bar{b})\bar{m} \times \bar{m}} \\ \mathbf{C}_r &\in \mathbb{R}^{\bar{p} \times (\bar{a}+2\bar{b})\bar{m}} & \mathbf{D} = \mathbf{r}_0 &\in \mathbb{R}^{\bar{p} \times \bar{m}} \end{aligned} \quad (2.47)$$

$$\begin{aligned}
 \mathbf{B}_r &= \left[\begin{array}{c} \mathbf{I}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{I}_{\bar{m} \times \bar{m}} \\ \hline \sqrt{2} \cdot \mathbf{I}_{\bar{m} \times \bar{m}} \\ \mathbf{0}_{\bar{m} \times \bar{m}} \\ \vdots \\ \sqrt{2} \cdot \mathbf{I}_{\bar{m} \times \bar{m}} \\ \mathbf{0}_{\bar{m} \times \bar{m}} \end{array} \right], \quad \mathbf{A}_r = \left[\begin{array}{c|c} \boldsymbol{\rho}_{r1} & \\ \ddots & \boldsymbol{\rho}_{r\bar{a}} \\ \hline & \mathbf{p}_{c1} \\ & \ddots & \mathbf{p}_{c\bar{b}} \end{array} \right] \\
 \mathbf{C}_r &= \left[\begin{array}{ccc|cc} \boldsymbol{\tau}_{r1} & \cdots & \boldsymbol{\tau}_{r\bar{a}} & \sqrt{2} \cdot \mathbf{r}_{c1} & \cdots & \sqrt{2} \cdot \mathbf{r}_{c\bar{b}} \end{array} \right] \quad (2.48) \\
 \mathbf{p}_{cb} &= \begin{bmatrix} \Re\{\boldsymbol{\rho}_{cb}\} & \Im\{\boldsymbol{\rho}_{cb}\} \\ -\Im\{\boldsymbol{\rho}_{cb}\} & \Re\{\boldsymbol{\rho}_{cb}\} \end{bmatrix}, \quad \mathbf{r}_{cb} = \begin{bmatrix} \Re\{\boldsymbol{\tau}_{cb}\} & \Im\{\boldsymbol{\tau}_{cb}\} \end{bmatrix} \\
 \boldsymbol{\rho}_{ra} &= \rho_{ra} \cdot \mathbf{I}_{\bar{m} \times \bar{m}}, \quad \boldsymbol{\rho}_{cb} = \rho_{cb} \cdot \mathbf{I}_{\bar{m} \times \bar{m}}
 \end{aligned}$$

- $\bar{j} - 1 = \bar{a} + 2\bar{b}$: Total number of poles and residues.
- $\begin{cases} \bar{a} : \text{Total number of real poles/residues.} \\ a : \text{Real pole/residue index.} \end{cases}$
- $\begin{cases} \bar{b} : \text{Total number of complex conjugate pole/residue pairs.} \\ b : \text{Complex conjugate pole/residue pair index.} \end{cases}$
- $\{\rho_{ra}\}_{a=1}^{a=\bar{a}}$: Set of real poles.
- $\{(\rho_{cb}, \rho_{cb}^*)\}_{b=1}^{b=\bar{b}}$: Set of complex conjugate pole pairs.
- $\{\boldsymbol{\tau}_{ra}\}_{a=1}^{a=\bar{a}}$: Set of real residue matrices.
- $\{(\boldsymbol{\tau}_{cb}, \boldsymbol{\tau}_{cb}^*)\}_{b=1}^{b=\bar{b}}$: Set of complex conjugate residue matrix pairs.
- $\boldsymbol{\tau}_0$: The transfer function constant term or the constant residue matrix.

The effective order of the system given by the transfer function (2.46) is $(\bar{j} - 1)\bar{m}$ according to the translation formula given in Appendix A, where all details of the process is found.

2.2.3 System Format Loewner Matrix Method

In this section, we shall derive the system format Loewner Matrix (SFLM) interpolation method, where we shall use the matrix \mathbb{L} along with its offshoot matrices to directly replace the matrices $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ of a LTI system transfer function and achieve the intended interpolation of the frequency data with which the LM matrices were created with.

Opposite Lagrange Equation

We build upon the initial frequency partitions Λ and \mathcal{M} defined by (2.16) that we rewrite here:

$$\begin{aligned} \Lambda &= \{\lambda_1, \dots, \lambda_{\bar{j}}\} & \mathcal{H}_\Lambda &= \{\mathbf{H}(\lambda_j); \lambda_j \in \Lambda\}, \\ \mathcal{M} &= \{\mu_1, \dots, \mu_{\bar{i}}\} & \mathcal{H}_\mathcal{M} &= \{\mathbf{H}(\mu_i); \mu_i \in \mathcal{M}\}. \end{aligned} \quad (2.49)$$

From each of the partitions, we extract a point to form a cross-partition pair of data points $(\lambda_j, \mathbf{H}_{\lambda_j})$ and $(\mu_i, \mathbf{H}_{\mu_i})$. The basic 1st degree Lagrange polynomial linear interpolation between the point pair would be defined as:

$$\begin{aligned} \mathbf{L}_{\{i,j\}}(s) &= \frac{s - \lambda_j}{\mu_i - \lambda_j} \mathbf{H}_{\mu_i} + \frac{s - \mu_i}{\lambda_j - \mu_i} \mathbf{H}_{\lambda_j} \\ \mathbf{L}_{\{i,j\}}(s) &= \frac{(s - \lambda_j)\mathbf{H}_{\mu_i} + (\mu_i - s)\mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \end{aligned} \quad (2.50)$$

which achieves the easily verifiable properties:

$$\mathbf{L}_{\{i,j\}}(\lambda_j) = \mathbf{H}_{\lambda_j}, \quad \mathbf{L}_{\{i,j\}}(\mu_i) = \mathbf{H}_{\mu_i} \quad (2.51)$$

We perform a simple modification on $\mathbf{L}_{\{i,j\}}(s)$ such that instead of achieving exact point evaluation, the modified Lagrange equation would actually perform opposite

evaluation:

$$\gamma_{\{i,j\}}(s) = \frac{(s - \lambda_j)\mathbf{H}_{\lambda_j} + (\mu_i - s)\mathbf{H}_{\mu_i}}{\mu_i - \lambda_j} \quad (2.52)$$

which we shall call the opposite Lagrange equation since it achieves the opposite of (2.51):

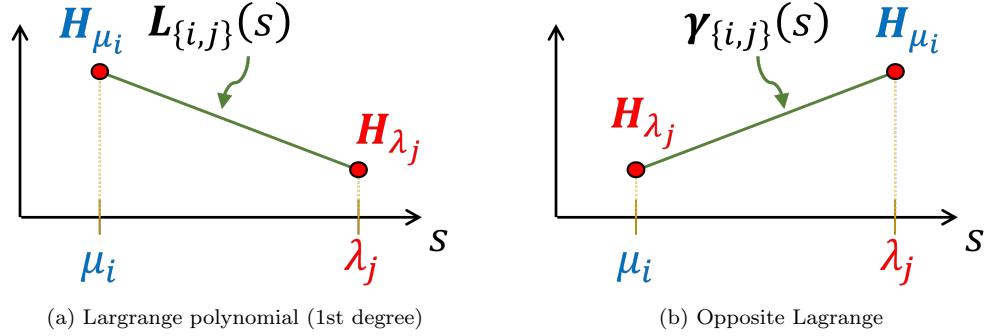


Figure 2.5: Comparison of Lagrange polynomial (1st degree) and its opposite.

$$\gamma_{\{i,j\}}(\lambda_j) = \mathbf{H}_{\mu_i}, \quad \gamma_{\{i,j\}}(\mu_i) = \mathbf{H}_{\lambda_j} \quad (2.53)$$

The property (2.53) is illustrated on Figure 2.5b where it is shown doing the exact opposite interpolation as the Lagrange 1st degree polynomial on Figure 2.5a. We notice that we can manipulate (2.52) in a way to reveal a LM expression:

$$\begin{aligned} \gamma_{\{i,j\}}(s) &= \frac{(s - \lambda_j)\mathbf{H}_{\lambda_j} + (\mu_i - s)\mathbf{H}_{\mu_i}}{\mu_i - \lambda_j} \\ \gamma_{\{i,j\}}(s) &= \frac{s\mathbf{H}_{\lambda_j} - \lambda_j\mathbf{H}_{\lambda_j} + \mu_i\mathbf{H}_{\mu_i} - s\mathbf{H}_{\mu_i}}{\mu_i - \lambda_j} \\ &= \frac{s\mathbf{H}_{\lambda_j} - s\mathbf{H}_{\mu_i} + \mu_i\mathbf{H}_{\mu_i} - \lambda_j\mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \\ &= s \frac{-\mathbf{H}_{\mu_i} + \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} + \frac{\mu_i\mathbf{H}_{\mu_i} - \lambda_j\mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \end{aligned} \quad (2.54)$$

$$\gamma_{\{i,j\}}(s) = -s\mathbb{L}_{\{i,j\}} + \sigma\mathbb{L}_{\{i,j\}} \quad (2.55)$$

where $\mathbb{L}_{\{i,j\}}$ is exactly the Loewner Matrix unit cell defined at (2.18) and where we have defined the frequency scaled LM unit cell term $\sigma\mathbb{L}_{\{i,j\}}$ from the matrix $\sigma\mathbb{L}$ which is called the shifted-Loewner Matrix [51]. It is now clear that opposite Lagrange function $\gamma_{\{i,j\}}$ is closely related to the LM, and we shall proceed to derive the LM based interpolation function using it.

Loewner Matrix System Interpolation

To follow up on the definition (2.55) of $\gamma_{\{i,j\}}(s)$, we define the function:

$$\mathbf{G}_{\{i,j\}}(s) = \mathbf{H}_{\lambda_j} \cdot [\gamma_{\{i,j\}}(s)]^{-1} \cdot \mathbf{H}_{\mu_i} \quad (2.56)$$

which one can quickly check to achieve exact interpolation again:

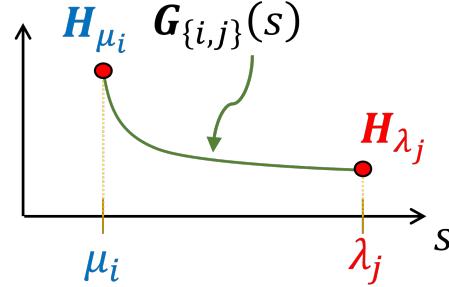


Figure 2.6: Special interpolation function illustration.

$$\mathbf{G}_{\{i,j\}}(\lambda_j) = \mathbf{H}_{\lambda_j}, \quad \mathbf{G}_{\{i,j\}}(\mu_i) = \mathbf{H}_{\mu_i} \quad (2.57)$$

Now, define the full matrix version of $\mathbf{G}_{\{i,j\}}(s)$:

$$\mathbf{G}(s) = \mathbf{W}[\gamma(s)]^{-1}\mathbf{F} \quad (2.58)$$

where

$$\boldsymbol{\gamma}(s) = \begin{bmatrix} \boldsymbol{\gamma}_{\{1,1\}}(s) & \boldsymbol{\gamma}_{\{1,2\}}(s) & \cdots & \boldsymbol{\gamma}_{\{1,\bar{j}\}}(s) \\ \boldsymbol{\gamma}_{\{2,1\}}(s) & \boldsymbol{\gamma}_{\{2,2\}}(s) & \cdots & \boldsymbol{\gamma}_{\{2,\bar{j}\}}(s) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\gamma}_{\{\bar{i},1\}}(s) & \boldsymbol{\gamma}_{\{\bar{i},2\}}(s) & \cdots & \boldsymbol{\gamma}_{\{\bar{i},\bar{j}\}}(s) \end{bmatrix} \in \mathbb{C}^{\bar{p}\bar{i} \times \bar{m}\bar{j}} \quad (2.59)$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{H}_{\lambda_1} & \mathbf{H}_{\lambda_2} & \cdots & \mathbf{H}_{\lambda_j} \end{bmatrix} \in \mathbb{C}^{\bar{p}\bar{i} \times \bar{m}\bar{j}} \quad (2.60)$$

$$\mathbf{F} = \begin{bmatrix} \mathbf{H}_{\mu_1}^T & \mathbf{H}_{\mu_2}^T & \cdots & \mathbf{H}_{\mu_{\bar{i}}}^T \end{bmatrix}^T \in \mathbb{C}^{\bar{p}\bar{i} \times \bar{j}} \quad (2.61)$$

Of course, for the matrix equation (2.58) to be possible, we require that $\boldsymbol{\gamma}(s)$ to be a square matrix for it to have an inverse defined. Because the problem of non-square $\boldsymbol{\gamma}(s)$ will be completely circumvented due to requirement of using MOR later on, we shall assume that $\boldsymbol{\gamma}(s)$ is square and invertible for the sake of maintaining the convenient definition of the matrix inverse throughout the derivations.

We now state the theorem of Loewner Matrix system method [49] :

Theorem 2.2.1 (Loewner Matrix System Interpolation). *The $\mathbf{G}(s)$ function as defined by (2.58) achieves the perfect interpolation of all points in Λ and \mathcal{M} , that is:*

$$\mathbf{G}(\lambda_j) = \mathbf{H}_{\lambda_j}, \quad \forall \lambda_j \in \Lambda \quad (2.62)$$

$$\mathbf{G}(\mu_i) = \mathbf{H}_{\mu_i}, \quad \forall \mu_i \in \mathcal{M} \quad (2.63)$$

Proof. Suppose we evaluate $\mathbf{G}(s)$ for any frequency $\lambda_j \in \Lambda$:

$$\mathbf{G}(\lambda_j) = \mathbf{W}[\boldsymbol{\gamma}(\lambda_j)]^{-1} \mathbf{F} \quad (2.64)$$

We take a closer look at the term $[\boldsymbol{\gamma}(\lambda_j)]^{-1} \mathbf{F}$, which we replace with the solution \mathbf{x} to the linear system of equations:

$$\boldsymbol{\gamma}(\lambda_j) \mathbf{x} = \mathbf{F} \quad (2.65)$$

to create the alternative form of (2.64):

$$\mathbf{G}(\lambda_j) = \mathbf{W}\mathbf{x} \quad (2.66)$$

Now perform some manipulation on (2.65):

$$\begin{aligned} \boldsymbol{\gamma}(\lambda_j) \mathbf{x} = \mathbf{F} \rightarrow \\ \begin{bmatrix} \boldsymbol{\gamma}_{\{1,1\}}(\lambda_j) & \boldsymbol{\gamma}_{\{1,2\}}(\lambda_j) & \cdots & \boldsymbol{\gamma}_{\{1,j\}}(\lambda_j) & \cdots & \boldsymbol{\gamma}_{\{1,\bar{j}\}}(\lambda_j) \\ \boldsymbol{\gamma}_{\{2,1\}}(\lambda_j) & \boldsymbol{\gamma}_{\{2,2\}}(\lambda_j) & \cdots & \boldsymbol{\gamma}_{\{2,j\}}(\lambda_j) & \cdots & \boldsymbol{\gamma}_{\{2,\bar{j}\}}(\lambda_j) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{\gamma}_{\{\bar{i},1\}}(\lambda_j) & \boldsymbol{\gamma}_{\{\bar{i},2\}}(\lambda_j) & \cdots & \boldsymbol{\gamma}_{\{\bar{i},j\}}(\lambda_j) & \cdots & \boldsymbol{\gamma}_{\{\bar{i},\bar{j}\}}(\lambda_j) \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_j \\ \vdots \\ \mathbf{x}_{\bar{j}} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{\mu_1} \\ \mathbf{H}_{\mu_2} \\ \cdots \\ \vdots \\ \mathbf{H}_{\mu_{\bar{i}}} \end{bmatrix} \\ \begin{bmatrix} \boldsymbol{\gamma}_{\{1,1\}}(\lambda_j) & \boldsymbol{\gamma}_{\{1,2\}}(\lambda_j) & \cdots & \mathbf{H}_{\mu_1} & \cdots & \boldsymbol{\gamma}_{\{1,\bar{j}\}}(\lambda_j) \\ \boldsymbol{\gamma}_{\{2,1\}}(\lambda_j) & \boldsymbol{\gamma}_{\{2,2\}}(\lambda_j) & \cdots & \mathbf{H}_{\mu_2} & \cdots & \boldsymbol{\gamma}_{\{2,\bar{j}\}}(\lambda_j) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{\gamma}_{\{\bar{i},1\}}(\lambda_j) & \boldsymbol{\gamma}_{\{\bar{i},2\}}(\lambda_j) & \cdots & \mathbf{H}_{\mu_{\bar{i}}} & \cdots & \boldsymbol{\gamma}_{\{\bar{i},\bar{j}\}}(\lambda_j) \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_j \\ \vdots \\ \mathbf{x}_{\bar{j}} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{\mu_1} \\ \mathbf{H}_{\mu_2} \\ \cdots \\ \vdots \\ \mathbf{H}_{\mu_{\bar{i}}} \end{bmatrix} \end{aligned} \quad (2.67)$$

By assessing (2.67), we can clearly see that direct solution \mathbf{x} must be a block column vector of zeros except for the row block at index j , where we have the identity matrix:

$$\mathbf{x} = \begin{bmatrix} \mathbf{0}_{\bar{m} \times \bar{m}} \\ \mathbf{0}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{I}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{0}_{\bar{m} \times \bar{m}} \end{bmatrix} \quad (2.68)$$

which means that the result of (2.66) is:

$$\mathbf{G}(\lambda_j) = \mathbf{W}\mathbf{x} = \left[\mathbf{H}_{\lambda_1} \ \mathbf{H}_{\lambda_2} \ \dots \ \mathbf{H}_{\lambda_j} \ \dots \ \mathbf{H}_{\lambda_{\bar{j}}} \right] \mathbf{x} = \mathbf{H}_{\lambda_j} \quad (2.69)$$

Since λ_j was an arbitrary entry of Λ , we have that

$$\mathbf{G}(\lambda_j) = \mathbf{H}_{\lambda_j}, \quad \forall \lambda_j \in \Lambda \quad (2.70)$$

Note that the same proof process can be done on $\mathbf{G}(\mu_i)^T$, and we would arrive at the conclusion that

$$\begin{aligned} \mathbf{G}(\mu_i)^T &= \mathbf{H}_{\mu_i}^T, \quad \forall \mu_i \in \mathcal{M} \\ \rightarrow \mathbf{G}(\mu_i) &= \mathbf{H}_{\mu_i}, \quad \forall \mu_i \in \mathcal{M} \end{aligned} \quad (2.71)$$

□

It should be noted that the proof given above for Theorem 2.2.1 only serves to demonstrate the perfect interpolation of points in Λ and \mathcal{M} . The process shown

in [49] is more theoretically robust where the Hankel matrix property of the LM is explained and how the LM is broken down into the product of a controllability matrix and an observability matrix. Given Theorem 2.2.1, we have that our function $\mathbf{G}(s)$ defined as (2.58) achieves perfect interpolation of all points in Λ and \mathcal{M} . Recalling that $\gamma_{\{i,j\}}(s) = -s\mathbb{L}_{\{i,j\}} + \sigma\mathbb{L}_{\{i,j\}}$ as mentioned by (2.55), we have:

$$\mathbf{G}(s) = \mathbf{W}[-s\mathbb{L} + \sigma\mathbb{L}]^{-1}\mathbf{F} \quad (2.72)$$

If we refer to the template transfer function (2.7) of a LTI descriptor system, equation (2.72) clearly insinuates that the Loewner Matrices \mathbb{L} and $\sigma\mathbb{L}$ with its auxiliary matrices \mathbf{W} and \mathbf{F} can be directly used as the matrices of a descriptor system as follow:

$$\mathbf{E}_{\mathbb{L}} = -\mathbb{L}, \quad \mathbf{A}_{\mathbb{L}} = -\sigma\mathbb{L}, \quad \mathbf{B}_{\mathbb{L}} = \mathbf{F}, \quad \mathbf{C}_{\mathbb{L}} = \mathbf{W} \quad (2.73)$$

At this point, the base Loewner Matrix system method has been described in full, but a couple of issues need to be resolved. First, because the Loewner Matrices are generated using complex frequency data, they are complex matrices, and the system built using them would be non-physical. Second, using \mathbb{L} and $\sigma\mathbb{L}$ directly as system matrices often results in numerically unsound systems because it is highly likely that \mathbb{L} and $\sigma\mathbb{L}$ are severely ill-conditioned. We present the solutions already established to deal with the above problems in the two following sub-subsections.

Real Loewner Matrices

Because the intention is to use the LM group $\{\mathbb{L}, \sigma\mathbb{L}, \mathbf{W}, \mathbf{F}\}$ directly as the matrices of the LTI descriptor system as described by (2.73), we need to turn these matrices into real matrices for the resulting system to be physical. The detailed

process of real LM alteration is presented over Appendix B and specifically in the appendix Section B.4, so we shall quickly go over the key points. Just like the case for the BRLM method, we modify the initial partitions Λ and \mathcal{M} to include the complete set of complex conjugate data resulting in the partition sets (2.35), which we rewrite here:

$$\begin{aligned}\Lambda &= \{\lambda_1, \lambda_1^*, \lambda_2, \lambda_2^*, \dots, \lambda_{\bar{j}}, \lambda_{\bar{j}}^*\} & \mathcal{H}_\Lambda &= \{\mathbf{H}(\lambda_j); \lambda_j \in \Lambda\} & \mathbf{H}(\lambda_j)^* &= \mathbf{H}(\lambda_j^*) \\ \mathcal{M} &= \{\mu_1, \mu_1^*, \mu_2, \mu_2^*, \dots, \mu_{\bar{i}}, \mu_{\bar{i}}^*\} & \mathcal{H}_\mathcal{M} &= \{\mathbf{H}(\mu_i); \mu_i \in \mathcal{M}\} & \mathbf{H}(\mu_i)^* &= \mathbf{H}(\mu_i^*)\end{aligned}\quad (2.74)$$

Now define the unitary transformation matrices $\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}$ and $\mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)}$ based on the general matrix definition given by (B.10) that we rewrite here:

$$\mathbf{T}_{(x\langle y\rangle)} \in \mathbb{C}^{2yx \times 2yx} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathcal{T}_{(x)} & & & \\ & \mathcal{T}_{(x)} & & \\ & & \ddots & \\ & & & \mathcal{T}_{(x)} \end{bmatrix}, \quad \mathcal{T}_{(x)} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{x \times x} & -1i \cdot \mathbf{I}_{x \times x} \\ \mathbf{I}_{x \times x} & 1i \cdot \mathbf{I}_{x \times x} \end{bmatrix} \in \mathbb{C}^{2x \times 2x} \quad (2.75)$$

The matrices in the Loewner Matrix group $\{\mathbb{L}, \sigma\mathbb{L}, \mathbf{W}, \mathbf{F}\}$ can be turned real through the following transformations:

$$\begin{aligned}\mathbb{L}_{re} &= \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \mathbb{L} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} \in \mathbb{R}^{2\bar{p}\bar{i} \times 2\bar{j}\bar{m}}, & \sigma\mathbb{L}_{re} &= \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \sigma\mathbb{L} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} \in \mathbb{R}^{2\bar{p}\bar{i} \times 2\bar{j}\bar{m}}, \\ \mathbf{W}_{re} &= \mathbf{W} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} \in \mathbb{R}^{\bar{p} \times 2\bar{j}\bar{m}}, & \mathbf{F}_{re} &= \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \mathbf{F} \in \mathbb{R}^{2\bar{i}\bar{m} \times \bar{p}}\end{aligned}\quad (2.76)$$

The direct formula in obtaining the real matrices are defined as follow:

$$\begin{aligned}
 \mathbb{L}_{re} &= \begin{bmatrix} \mathbb{L}_{re\{1,1\}} & \mathbb{L}_{re\{1,2\}} & \cdots & \mathbb{L}_{re\{1,\bar{j}\}} \\ \mathbb{L}_{re\{2,1\}} & \mathbb{L}_{re\{2,2\}} & \cdots & \mathbb{L}_{re\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{re\{\bar{i},1\}} & \mathbb{L}_{re\{\bar{i},2\}} & \cdots & \mathbb{L}_{re\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{2ip \times 2j\bar{m}} \\
 \mathbb{L}_{re\{i,j\}} &= \begin{bmatrix} \Re\{\mathbb{L}_{\alpha\{i,j\}}\} + \Re\{\mathbb{L}_{\beta\{i,j\}}\} & \Im\{\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\mathbb{L}_{\beta\{i,j\}}\} \\ -\Im\{\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\mathbb{L}_{\beta\{i,j\}}\} & \Re\{\mathbb{L}_{\alpha\{i,j\}}\} - \Re\{\mathbb{L}_{\beta\{i,j\}}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times 2\bar{m}} \\
 \mathbb{L}_{\alpha\{i,j\}} &= \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j}, \quad \mathbb{L}_{\beta\{i,j\}} = \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}^*}{\mu_i - \lambda_j^*} \\
 \sigma\mathbb{L}_{re} &= \begin{bmatrix} \sigma\mathbb{L}_{re\{1,1\}} & \sigma\mathbb{L}_{re\{1,2\}} & \cdots & \sigma\mathbb{L}_{re\{1,\bar{j}\}} \\ \sigma\mathbb{L}_{re\{2,1\}} & \sigma\mathbb{L}_{re\{2,2\}} & \cdots & \sigma\mathbb{L}_{re\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma\mathbb{L}_{re\{\bar{i},1\}} & \sigma\mathbb{L}_{re\{\bar{i},2\}} & \cdots & \sigma\mathbb{L}_{re\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{2ip \times 2j\bar{m}} \\
 \sigma\mathbb{L}_{re\{i,j\}} &= \begin{bmatrix} \Re\{\sigma\mathbb{L}_{\alpha\{i,j\}}\} + \Re\{\sigma\mathbb{L}_{\beta\{i,j\}}\} & \Im\{\sigma\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\sigma\mathbb{L}_{\beta\{i,j\}}\} \\ -\Im\{\sigma\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\sigma\mathbb{L}_{\beta\{i,j\}}\} & \Re\{\sigma\mathbb{L}_{\alpha\{i,j\}}\} - \Re\{\sigma\mathbb{L}_{\beta\{i,j\}}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times 2\bar{m}} \\
 \sigma\mathbb{L}_{\alpha\{i,j\}} &= \frac{\mu_i \mathbf{H}_{\mu_i} - \lambda_j \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j}, \quad \sigma\mathbb{L}_{\beta\{i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i} - \lambda_j^* \mathbf{H}_{\lambda_j}^*}{\mu_i - \lambda_j^*}
 \end{aligned} \tag{2.77}$$

$$\mathbf{W}_{re} \in \mathbb{R}^{\bar{p} \times 2j\bar{m}} =$$

$$\sqrt{2} \begin{bmatrix} \Re\{\mathbf{H}_{\lambda_1}\} & \Im\{\mathbf{H}_{\lambda_1}\} & \Re\{\mathbf{H}_{\lambda_2}\} & \Im\{\mathbf{H}_{\lambda_2}\} & \cdots & \Re\{\mathbf{H}_{\lambda_{\bar{j}}}\} & \Im\{\mathbf{H}_{\lambda_{\bar{j}}}\} \end{bmatrix}$$

$$\mathbf{F}_{re} \in \mathbb{R}^{2\bar{i}\bar{p} \times \bar{m}} = \sqrt{2} \begin{bmatrix} \Re\{\mathbf{H}_{\mu_1}\}^T & \Im\{\mathbf{H}_{\mu_1}\}^T & \Re\{\mathbf{H}_{\mu_2}\}^T & \Im\{\mathbf{H}_{\mu_2}\}^T & \cdots & \Re\{\mathbf{H}_{\mu_j}\}^T & \Im\{\mathbf{H}_{\mu_j}\}^T \end{bmatrix}^T$$

Note from this point on, when we refer to any matrix of the Loewner Matrix set $\{\mathbb{L}, \sigma\mathbb{L}, \mathbf{W}, \mathbf{F}\}$, we shall assume that the real transformation is applied and the number of frequency partitions entries \bar{j} and \bar{i} include the added complex conjugate data entries, so that our Loewner Matrices imply the following characteristics:

$$\begin{aligned} \mathbb{L} &\in \mathbb{R}^{\bar{i}\bar{p} \times \bar{j}\bar{m}}, & \sigma\mathbb{L} &\in \mathbb{R}^{\bar{i}\bar{p} \times \bar{j}\bar{m}}, \\ \mathbf{W} &\in \mathbb{R}^{\bar{p} \times \bar{j}\bar{p}}, & \mathbf{F} &\in \mathbb{R}^{\bar{i}\bar{m} \times \bar{m}} \end{aligned} \quad (2.79)$$

Loewner Matrix System Model Order Reduction

The model order reduction procedure is based on the procedure described in [49], with a few minor tweaks. First, define the LM pencil:

$$\boldsymbol{\gamma} = s_{ref} \cdot \mathbb{L} - \sigma\mathbb{L} \in \mathbb{C}^{\bar{i}\bar{p} \times \bar{j}\bar{m}} \quad (2.80)$$

where $s_{ref} \in \Lambda \cup \mathcal{M}$ is the reference frequency. However, s_{ref} is a complex frequency, and would turn $\boldsymbol{\gamma}$ to a complex matrix when \mathbb{L} and $\sigma\mathbb{L}$ are real, preventing us from obtaining a real matrix pencil. As a work around, we instead use the alternative reference variable $f_{ref} = |s_{ref}|$, which allows us to maintain real LMs while allocating the same magnitude of contribution of \mathbb{L} in $\boldsymbol{\gamma}$:

$$\boldsymbol{\gamma} = f_{ref} \cdot \mathbb{L} - \sigma\mathbb{L} \in \mathbb{R}^{\bar{i}\bar{p} \times \bar{j}\bar{m}} \quad (2.81)$$

We then perform SVD on $\boldsymbol{\gamma}$:

$$\boldsymbol{\gamma} = \mathbf{U} \mathbf{S} \mathbf{V}^H \quad (2.82)$$

Now choose a system order $n \leq \min(\bar{p}, \bar{m})$ as the cut-off index of the singular values. The singular value cut-off at index n is performed by only keeping the n largest singular values and their respective right and left singular vectors. Doing so results in the reduced singular vector matrices:

$$\mathbf{U}_n = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_n \end{bmatrix} \in \mathbb{R}^{\bar{p} \times n}, \quad \mathbf{V}_n = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \end{bmatrix} \in \mathbb{R}^{\bar{m} \times n} \quad (2.83)$$

We can then apply back the reduced singular vector matrices \mathbf{U}_n and \mathbf{V}_n to the individual LMs \mathbb{L} , $\sigma\mathbb{L}$ and the auxiliary matrices \mathbf{W} , \mathbf{F} :

$$\begin{aligned} \mathbb{L}_n &= \mathbf{U}_n^H \mathbb{L} \mathbf{V}_n \in \mathbb{R}^{n \times n} \\ \sigma\mathbb{L}_n &= \mathbf{U}_n^H \sigma\mathbb{L} \mathbf{V}_n \in \mathbb{R}^{n \times n} \\ \mathbf{W}_n &= \mathbf{W} \mathbf{V}_n \in \mathbb{R}^{\bar{p} \times n} \\ \mathbf{F}_n &= \mathbf{U}_n^H \mathbf{F} \in \mathbb{R}^{n \times \bar{m}} \end{aligned} \quad (2.84)$$

The order reduced matrices in (2.84) can then be assigned to the transfer function matrices of an LTI descriptor system:

$$\mathbf{E}_n = -\mathbb{L}_n, \quad \mathbf{A}_n = -\sigma\mathbb{L}_n, \quad \mathbf{B}_n = \mathbf{F}_n, \quad \mathbf{C}_n = \mathbf{W}_n \quad (2.85)$$

The descriptor system consisting of matrices in (2.85) leads to a transfer function $\mathbf{G}(s)$ which, given a good choice of n , can approximate closely the initial data sets:

$$\mathbf{G}(\lambda_j) \approx \mathbf{H}(\lambda_j), \quad \forall \lambda_j \in \Lambda$$

$$\mathbf{G}(\mu_i) \approx \mathbf{H}(\mu_i), \quad \forall \mu_i \in \mathcal{M}$$

Unfortunately, there is no solve-all methodology in selecting n because n is affected by a number of factors such as the nature of the system, the number of data utilized, the range of data, noise in the data, and the partitioning method used on the data. In general, a good approach to automatic order n selection is in the form of a binary search using as measure the combination of root mean square (RMS) error and system stability, though no rigorous proof can attest this.

2.3 Parametric Method Review

In this section, we review some of the current state-of-the art parametric macromodeling methods. Two methods are presented, one as a significant representative of parametric macromodeling in the LM framework and one as a significant representative of the VF framework. The LM framework parametric method is named as the Data-Driven Parametric Loewner Model method and the VF framework parametric method is named the parameterized Sanathanan–Koerner (PSK) iteration method. The section ends with a small mention of other parametric macromodeling methods as well as a small contextualization of the proposed parametric methods of this Thesis.

For the quick review of the parametric methods, we define the objective of the methods as the fitting or approximation of a SISO system with an unknown frequency response $\mathbf{H}(s, \theta)$ that depends on both frequency s and some external parameter θ . The system is known through a set of measurements

$$\mathcal{H} = \{\mathbf{H}(s_t, \theta_q); s_t \in \mathcal{S}, \theta_q \in \Theta\} \quad (2.86)$$

where

- $\mathbf{H}(s_t, \theta_q) \in \mathbb{C}^{1 \times 1}$ is the transfer function of the system evaluated at s_t and θ_q ;
- $\Theta = \{\theta_q\}_{q=1}^{\bar{q}}$ is a set of single parameter samples with no repeated entries spanning the parameter range $[\theta_1, \theta_{\bar{q}}]$. We assume real parameter values.
- \mathcal{S} is defined as in (2.15) as the set of frequency sample points and s_t is a random entry of the set \mathcal{S} :

$$\mathcal{S} = \{s_1, \dots, s_{\bar{t}}\} \quad s_t \in \mathcal{S} \quad (2.87)$$

2.3.1 Data-Driven Parametric Loewner Model

The data-driven parametric Loewner model method presented here is primarily based on [114] which presented the base SISO version of the method, but development into the more general MIMO version translated to descriptor system form with lower order realization can be found at [85]

The method first partitions the frequency and parameter sets into pairs of partitions as follow:

$$\begin{aligned} \Lambda &= \{\lambda_1, \dots, \lambda_{\bar{j}}\}, \quad \mathcal{M} = \{\mu_1, \dots, \mu_{\bar{i}}\} \\ \Lambda \cup \mathcal{M} &= \mathcal{S}, \quad \Lambda \cap \mathcal{M} = \emptyset \end{aligned} \quad (2.88)$$

$$\begin{aligned} \Pi &= \{\pi_1, \dots, \pi_{\bar{l}}\}, \quad \mathcal{E} = \{\varepsilon_1, \dots, \varepsilon_{\bar{k}}\} \\ \Pi \cup \mathcal{E} &= \Theta, \quad \Pi \cap \mathcal{E} = \emptyset \end{aligned} \quad (2.89)$$

Combination of one frequency partition with one parameter partition results in 4 distinct data partitions:

$$\begin{aligned} \Phi_{\Lambda, \Pi} &= \{\mathbf{H}_{\lambda_j, \pi_l} = \mathbf{H}(\lambda_j, \pi_l); \lambda_j \in \mathcal{S}_\Lambda, \pi_l \in \Theta_\Pi\} \\ \Phi_{\Lambda, \mathcal{E}} &= \{\mathbf{H}_{\lambda_j, \varepsilon_k} = \mathbf{H}(\lambda_j, \varepsilon_k); \lambda_j \in \mathcal{S}_\Lambda, \varepsilon_k \in \Theta_\mathcal{E}\} \end{aligned}$$

$$\begin{aligned}\Phi_{\mathcal{M},\Pi} &= \{\mathbf{H}_{\mu_i,\pi_l} = \mathbf{H}(\mu_i, \pi_l); \mu_i \in \mathcal{S}_{\mathcal{M}}, \pi_l \in \Theta_{\Pi}\} \\ \Phi_{\mathcal{M},\mathcal{E}} &= \{\mathbf{H}_{\mu_i,\varepsilon_k} = \mathbf{H}(\mu_i, \varepsilon_k); \mu_i \in \mathcal{S}_{\mathcal{M}}, \varepsilon_k \in \Theta_{\mathcal{E}}\}\end{aligned}\quad (2.90)$$

The base fitting equation is the standard two-variate barycentric equation defined using entries of the parametric data set $\Phi_{\Lambda,\Pi}$:

$$\mathbf{G}(s, \theta) = \frac{\sum_{j=1}^{\bar{j}} \sum_{l=1}^{\bar{l}} \frac{c_{j,l} \mathbf{H}_{\lambda_j, \pi_l}}{(s - \lambda_j)(\theta - \pi_l)}}{\sum_{j=1}^{\bar{j}} \sum_{l=1}^{\bar{l}} \frac{c_{j,l}}{(s - \lambda_j)(\theta - \pi_l)}} \quad (2.91)$$

In a similar logic to how the uni-variate barycentric equation (2.21) interpolates perfectly the entries of the \mathcal{H}_{Λ} set by definition, given a non-trivial coefficient set $\{c_{j,l}; j = 1, \dots, \bar{j}, l = 1, \dots, \bar{l}\}$, $\mathbf{G}(s, \theta)$ as defined by (2.91) interpolates all entries of set $\Phi_{\Lambda,\Pi}$ perfectly as well:

$$\mathbf{G}(\lambda_j, \pi_l) = \mathbf{H}_{\lambda_j, \pi_l}, \quad \forall \lambda_j \in \Lambda, \pi_l \in \Pi \quad (2.92)$$

Define the coefficient set compactly as the entries of the coefficient vector:

$$\mathbf{C} = \begin{bmatrix} \mathbf{c}_{\lambda_1} \\ \mathbf{c}_{\lambda_2} \\ \vdots \\ \mathbf{c}_{\lambda_{\bar{j}}} \end{bmatrix} \in \mathbb{C}^{\bar{j} \bar{l} \times 1}, \quad \mathbf{c}_{\lambda_j} = \begin{bmatrix} c_{j,1} \\ c_{j,2} \\ \vdots \\ c_{j,\bar{l}} \end{bmatrix} \in \mathbb{C}^{\bar{l} \times 1} \quad (2.93)$$

The parametric Loewner Matrix is formulated through minimization of the approximation error of $\mathbf{G}(s, \theta)$ when evaluated at entries of partitions $\Phi_{\Lambda,\mathcal{E}}$, $\Phi_{\mathcal{M},\Pi}$, and $\Phi_{\mathcal{M},\mathcal{E}}$. The evaluation error of an entry $(\lambda_j, \varepsilon_k)$ from the set $\Phi_{\Lambda,\mathcal{E}}$ can be determined

as:

$$\mathbf{H}(\lambda_j, \varepsilon_k) - \mathbf{G}(\lambda_j, \varepsilon_k) = \frac{\sum_{l=1}^{\bar{l}} \frac{(\mathbf{H}_{\lambda_j, \varepsilon_k} - \mathbf{H}_{\lambda_j, \pi_l})c_{j,l}}{\varepsilon_k - \pi_l}}{\sum_{l=1}^{\bar{l}} \frac{c_{j,l}}{\varepsilon_k - \pi_l}} \quad (2.94)$$

Setting the error equation (2.94) to equate zero and assembling all index k cases of the equation and reformatting into a matrix equation, we obtain:

$$\widehat{\mathbb{L}}_{\Lambda} \cdot \mathbf{C} = \mathbf{0}_{\bar{j} \cdot \bar{k} \times 1}, \quad \widehat{\mathbb{L}}_{\Lambda} = \begin{bmatrix} \mathbb{L}_{\lambda_1} \\ & \mathbb{L}_{\lambda_2} \\ & & \ddots \\ & & & \mathbb{L}_{\lambda_{\bar{j}}} \end{bmatrix} \in \mathbb{C}^{\bar{j} \cdot \bar{k} \times \bar{j} \cdot \bar{l}} \quad (2.95)$$

$$\mathbb{L}_{\lambda_j} = \begin{bmatrix} \frac{\mathbf{H}_{\lambda_j, \varepsilon_1} - \mathbf{H}_{\lambda_j, \pi_1}}{\varepsilon_1 - \pi_1} & \frac{\mathbf{H}_{\lambda_j, \varepsilon_1} - \mathbf{H}_{\lambda_j, \pi_2}}{\varepsilon_1 - \pi_2} & \dots & \frac{\mathbf{H}_{\lambda_j, \varepsilon_1} - \mathbf{H}_{\lambda_j, \pi_{\bar{l}}}}{\varepsilon_1 - \pi_{\bar{l}}} \\ \frac{\mathbf{H}_{\lambda_j, \varepsilon_2} - \mathbf{H}_{\lambda_j, \pi_1}}{\varepsilon_2 - \pi_1} & \frac{\mathbf{H}_{\lambda_j, \varepsilon_2} - \mathbf{H}_{\lambda_j, \pi_2}}{\varepsilon_2 - \pi_2} & \dots & \frac{\mathbf{H}_{\lambda_j, \varepsilon_2} - \mathbf{H}_{\lambda_j, \pi_{\bar{l}}}}{\varepsilon_2 - \pi_{\bar{l}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\mathbf{H}_{\lambda_j, \varepsilon_{\bar{k}}} - \mathbf{H}_{\lambda_j, \pi_1}}{\varepsilon_{\bar{k}} - \pi_1} & \frac{\mathbf{H}_{\lambda_j, \varepsilon_{\bar{k}}} - \mathbf{H}_{\lambda_j, \pi_2}}{\varepsilon_{\bar{k}} - \pi_2} & \dots & \frac{\mathbf{H}_{\lambda_j, \varepsilon_{\bar{k}}} - \mathbf{H}_{\lambda_j, \pi_{\bar{l}}}}{\varepsilon_{\bar{k}} - \pi_{\bar{l}}} \end{bmatrix} \quad (2.96)$$

The evaluation error of an entry (μ_i, π_l) from the set $\Phi_{\mathcal{M}, \Pi}$ can be determined

as:

$$\mathbf{H}(\mu_i, \pi_l) - \mathbf{G}(\mu_i, \pi_l) = \frac{\sum_{j=1}^{\bar{j}} \frac{(\mathbf{H}_{\mu_i, \pi_l} - \mathbf{H}_{\lambda_j, \pi_l})c_{j,l}}{\mu_i - \lambda_j}}{\sum_{l=1}^{\bar{l}} \frac{c_{j,l}}{\mu_i - \lambda_j}} \quad (2.97)$$

Setting the error equation (2.97) to equate zero and assembling all index l cases of the equation and reformatting into a matrix equation, we obtain:

$$\widehat{\mathbb{L}}_{\Pi} \cdot \mathbf{C} = \mathbf{0}_{\bar{l} \cdot \bar{i} \times 1} \quad (2.98)$$

$$\widehat{\mathbb{L}}_{\Pi} = \begin{bmatrix} \mathbb{L}_{\pi_1}(:, 1) & & & \\ & \ddots & & \\ & & \mathbb{L}_{\pi_{\bar{l}}}(:, 1) & \\ & & & \end{bmatrix} \in \mathbb{C}^{\bar{l} \cdot \bar{i} \times \bar{j} \cdot \bar{l}} \quad (2.99)$$

where $\mathbb{L}_{\pi_l}(:, j)$ is the j 'th column of matrix \mathbb{L}_{π_l} and \mathbb{L}_{π_l} is defined as:

$$\mathbb{L}_{\pi_l} = \begin{bmatrix} \frac{\mathbf{H}_{\mu_1, \pi_l} - \mathbf{H}_{\lambda_1, \pi_l}}{\mu_1 - \lambda_1} & \frac{\mathbf{H}_{\mu_1, \pi_l} - \mathbf{H}_{\lambda_2, \pi_l}}{\mu_1 - \lambda_2} & \dots & \frac{\mathbf{H}_{\mu_1, \pi_l} - \mathbf{H}_{\lambda_j, \pi_l}}{\mu_1 - \lambda_j} \\ \frac{\mathbf{H}_{\mu_2, \pi_l} - \mathbf{H}_{\lambda_1, \pi_l}}{\mu_2 - \lambda_1} & \frac{\mathbf{H}_{\mu_2, \pi_l} - \mathbf{H}_{\lambda_2, \pi_l}}{\mu_2 - \lambda_2} & \dots & \frac{\mathbf{H}_{\mu_2, \pi_l} - \mathbf{H}_{\lambda_j, \pi_l}}{\mu_2 - \lambda_j} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\mathbf{H}_{\mu_{\bar{i}}, \pi_l} - \mathbf{H}_{\lambda_1, \pi_l}}{\mu_{\bar{i}} - \lambda_1} & \frac{\mathbf{H}_{\mu_{\bar{i}}, \pi_l} - \mathbf{H}_{\lambda_2, \pi_l}}{\mu_{\bar{i}} - \lambda_2} & \dots & \frac{\mathbf{H}_{\mu_{\bar{i}}, \pi_l} - \mathbf{H}_{\lambda_j, \pi_l}}{\mu_{\bar{i}} - \lambda_j} \end{bmatrix} \quad (2.100)$$

The evaluation error of an entry (μ_i, ε_k) from the set $\Phi_{\mathcal{M}, \mathcal{E}}$ can be determined as:

$$\mathbf{H}(\mu_i, \varepsilon_k) - \mathbf{G}(\mu_i, \varepsilon_k) = \frac{\sum_{j=1}^{\bar{j}} \sum_{l=1}^{\bar{l}} \frac{(\mathbf{H}_{\mu_i, \varepsilon_k} - \mathbf{H}_{\lambda_j, \pi_l}) c_{j,l}}{(\mu_i - \lambda_j)(\varepsilon_k - \pi_l)}}{\sum_{j=1}^{\bar{j}} \sum_{l=1}^{\bar{l}} \frac{c_{j,l}}{(\mu_i - \lambda_j)(\varepsilon_k - \pi_l)}} \quad (2.101)$$

Setting the error equation (2.101) to equate zero and assembling all index l cases of the equation and reformatting into a matrix equation, we obtain:

$$\mathbb{L}_2 \cdot \mathbf{C} = \mathbf{0}_{\bar{i} \cdot \bar{k} \times 1} \quad (2.102)$$

$$\mathbb{L}_2 = \begin{bmatrix} \mathbb{L}_{\mu_1, \lambda_1} & \mathbb{L}_{\mu_1, \lambda_2} & \dots & \mathbb{L}_{\mu_1, \lambda_{\bar{j}}} \\ \mathbb{L}_{\mu_2, \lambda_1} & \mathbb{L}_{\mu_2, \lambda_2} & \dots & \mathbb{L}_{\mu_2, \lambda_{\bar{j}}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{\mu_{\bar{i}}, \lambda_1} & \mathbb{L}_{\mu_{\bar{i}}, \lambda_2} & \dots & \mathbb{L}_{\mu_{\bar{i}}, \lambda_{\bar{j}}} \end{bmatrix} \in \mathbb{C}^{\bar{i} \cdot \bar{k} \times \bar{j} \cdot \bar{l}} \quad (2.103)$$

$$\mathbb{L}_{\mu_i, \lambda_j} = \begin{bmatrix} \frac{\mathbf{H}_{\mu_i, \varepsilon_1} - \mathbf{H}_{\lambda_j, \pi_1}}{(\mu_i - \lambda_j)(\varepsilon_1 - \pi_1)} & \frac{\mathbf{H}_{\mu_i, \varepsilon_1} - \mathbf{H}_{\lambda_j, \pi_2}}{(\mu_i - \lambda_j)(\varepsilon_1 - \pi_2)} & \dots & \frac{\mathbf{H}_{\mu_i, \varepsilon_1} - \mathbf{H}_{\lambda_j, \pi_{\bar{l}}}}{(\mu_i - \lambda_j)(\varepsilon_1 - \pi_{\bar{l}})} \\ \frac{\mathbf{H}_{\mu_i, \varepsilon_2} - \mathbf{H}_{\lambda_j, \pi_1}}{(\mu_i - \lambda_j)(\varepsilon_2 - \pi_1)} & \frac{\mathbf{H}_{\mu_i, \varepsilon_2} - \mathbf{H}_{\lambda_j, \pi_2}}{(\mu_i - \lambda_j)(\varepsilon_2 - \pi_2)} & \dots & \frac{\mathbf{H}_{\mu_i, \varepsilon_2} - \mathbf{H}_{\lambda_j, \pi_{\bar{l}}}}{(\mu_i - \lambda_j)(\varepsilon_2 - \pi_{\bar{l}})} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\mathbf{H}_{\mu_i, \varepsilon_{\bar{k}}} - \mathbf{H}_{\lambda_j, \pi_1}}{(\mu_i - \lambda_j)(\varepsilon_{\bar{k}} - \pi_1)} & \frac{\mathbf{H}_{\mu_i, \varepsilon_{\bar{k}}} - \mathbf{H}_{\lambda_j, \pi_2}}{(\mu_i - \lambda_j)(\varepsilon_{\bar{k}} - \pi_2)} & \dots & \frac{\mathbf{H}_{\mu_i, \varepsilon_{\bar{k}}} - \mathbf{H}_{\lambda_j, \pi_{\bar{l}}}}{(\mu_i - \lambda_j)(\varepsilon_{\bar{k}} - \pi_{\bar{l}})} \end{bmatrix} \in \mathbb{C}^{\bar{l} \times \bar{k}} \quad (2.104)$$

By stacking the sub-Loewner Matrices $\widehat{\mathbb{L}}_\Lambda$, $\widehat{\mathbb{L}}_\Pi$, and \mathbb{L}_2 , we obtain a complete parametric Loewner Matrix:

$$\widehat{\mathbb{L}}_2 = \begin{bmatrix} \widehat{\mathbb{L}}_\Lambda \\ \widehat{\mathbb{L}}_\Pi \\ \mathbb{L}_2 \end{bmatrix} \in \mathbb{C}^{(\bar{j} \cdot \bar{k} + \bar{l} \cdot \bar{i} + \bar{i} \cdot \bar{k}) \times \bar{j} \cdot \bar{l}} \quad (2.105)$$

An ideal solution \mathbf{C} would satisfy the zero error condition:

$$\widehat{\mathbb{L}}_2 \cdot \mathbf{C} = \mathbf{0}_{(\bar{j} \cdot \bar{k} + \bar{l} \cdot \bar{i} + \bar{i} \cdot \bar{k}) \times 1} \quad (2.106)$$

A practical solution \mathbf{C} can be generated by selecting the right singular vector of $\widehat{\mathbb{L}}_2$ associated to a sufficiently small singular value. A small singular value insinuates an overall small approximation error when $\mathbf{G}(s, \theta)$ is evaluated at entries of the partitions $\Phi_{\Lambda, \mathcal{E}}$, $\Phi_{\mathcal{M}, \Pi}$, and $\Phi_{\mathcal{M}, \mathcal{E}}$ if $\mathbf{G}(s, \theta)$ adopts the associated right singular vector as \mathbf{C} . More detailed presentation of this algorithm is given in [114] where we have a deeper discussion regarding the Loewner Matrix ranks considerations as well as translation into descriptor system form.

A strong point of the algorithm which guarantees perfect interpolation of points in $\Phi_{\Lambda,\Pi}$ by $\mathbf{G}(s, \theta)$ by definition is also a source of problem. Such a strict interpolation condition often promotes extreme pole behaviour which may cause erratic approximation profile in-between points belonging to the set $\Phi_{\Lambda,\Pi}$. This algorithm is to be directly compared to the proposed PBRLM method presented in Chapter 4, where use of polynomials to deal with parametric variations allow a more lax interpolation condition without significantly deteriorating the overall model accuracy

2.3.2 Parametrized Sanathana-Koerner Method

The SK iteration method [127] development towards its parametric extension, the parameterized Sanathanan–Koerner (PSK) iteration method, was incited by the fact that the popular VF method, a known efficient reformulation of SK iteration method [66], was not easily extendible in a parametric sense. The PSK was first introduced in [128] with the base form of the approximation equation defined as the rational function

$$\mathbf{G}(s, \theta) = \frac{\mathbf{N}(s, \theta)}{\mathbf{D}(s, \theta)} = \frac{\sum_{n=0}^{\bar{n}} \sum_{m=0}^{\bar{m}} R_{n,m} \rho_m(\theta) \varphi_n(s)}{\sum_{n=0}^{\bar{n}} \sum_{m=0}^{\bar{m}} r_{n,m} \rho_m(\theta) \varphi_n(s)} \quad (2.107)$$

where $R_{n,m}$ and $r_{n,m}$ are the coefficients to be determined, $\varphi_n(s)$ is the VF basis dictating the poles of the model:

$$\varphi_n(s) = (s - a_n)^{-1}, \quad \varphi_0(s) = 1 \quad (2.108)$$

and $\rho_m(\theta)$ is the selected polynomial basis that approximates the parametric variation of the model.

The coefficients are solved through an iterative process where we try to minimize the modeling error in the least-square sense. At iteration index i , the error minimization equation is defined as:

$$\varepsilon_{(i)}^2 = \left\| \mathbf{D}^{(i-1)}(s_t, \theta_q) [\mathbf{N}^{(i)}(s_t, \theta_q) - \mathbf{D}^{(i)}(s_t, \theta_q) \cdot \mathbf{H}_{t,q}] \right\| \quad (2.109)$$

In [128], the polynomial basis $\rho_m(\theta)$ was defined simply as $\rho_m(\theta) = \theta^m$. The formula was improved in [129] with the use of Chebychev polynomials and further improved in [104] with optimization and adjustment for multi-parameter cases. Later, stability can be imposed by introducing constraints in the fitting process [68, 130, 131] and most recently, passivity can also be imposed [70]. It is from this parametric method that the proposed PBRLM method in Chapter 4 took inspiration from, where we introduce polynomials to the barycentric equation to deal with parametric variation of the model.

2.3.3 Other Parametric Methods and Proposed Methods Context

We briefly present a non-exhaustive set of parametric methods that are relevant to the LM or the VF framework. An alternative take to LM parametric macromodeling can be found in [132] and [56], which both utilize tangential interpolation that involves ‘connecting’ a set of non-parametric models defined at a set of θ points, unlike the method presented in Section 2.3.1 which formulates a parametric model as a single entity proper. Another avenue for parametric macromodeling involves a form of

combination of the LM and the VF approaches. The method is called the Adaptive-Antoulas Anderson (AAA) algorithm originating from [63]. The Adaptive-Antoulas Anderson (AAA) method is an iterative method that is based on the barycentric formula and solves for its coefficients at select points while leaving the approximation of the remaining data through least-square optimization. The method recently was given a parametric extension in [84].

Within the general field of parametric macromodeling, the proposed PBRLM method presented in Chapter 4 is fully within the LM framework. It is similar to the Data-Driven Parametric Loewner Model presented in Section 2.3.1 in that both extends the uni-variate barycentric equation (2.21), but whereas the method in Section 2.3.1 extends in the sense of two-variate equation, the PBRLM method, borrowing the idea of the PSK method, inserts polynomials in the barycentric equation to deal with parametric variation. The second proposed PSFLM method presented in Chapter 5 is also within the LM framework. Rather than interpolating frequency data using the barycentric equation directly, this method is a parametric extension of the non-parametric method in [49] and presented in Section 2.2.3 where the Loewner matrix and its auxiliary matrices can be directly used as the descriptor system matrices $\{\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$. The resulting parametric system generated by the PSFLM method is a parametric descriptor system $\{\mathbf{E}(\theta), \mathbf{A}(\theta), \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ whose matrices are the parametric Loewner Matrix and its auxiliary matrices.

2.4 Passivity Enforcement Review

In this section, we review the general process of passivity enforcement on state-space models. The review starts with the robust Hamiltonian matrix eigenvalues

passivity checking method followed by the perturbation method of said eigenvalues in order to eliminate non-passive behaviours of the system. The review also presents the alternative perturbation method of direct perturbation of the transfer function's singular values which can also achieve elimination of passivity violations. The review ends with presenting the gramian based error control scheme along with the more advanced filtered gramian error control scheme.

2.4.1 Hamiltonian Matrix Eigenvalue Passivity Checking

The passivity condition for S -parameter data has been associated with the bounded-real condition of the LTI system expressed by Theorem 2.1.1 described in section 2.1.4. Passivity enforcement typically aims to deal with condition 2 of the bounded-real state, which is expressed in a practical manner as (2.13) that we rewrite here:

$$\sup_{\omega \in \mathbb{R}} \{\sigma_{max}(\mathbf{S}(j\omega))\} \leq 1 \quad (2.110)$$

Condition (2.110) leads to the most natural and direct form of passivity check consisting of sampling of the transfer function maximum singular values for a given sample set $\mathcal{S}_{samp} = \{s_1, s_2, \dots, s_t\}$ to determine if the threshold is respected at all points:

$$\max_{s_t \in \mathcal{S}_{samp}} \{\sigma(\mathbf{G}(s_t))\} \leq 1 \quad (2.111)$$

Of course, the problem with frequency sampling passivity check is that it can never truly guarantee full system passivity given we cannot check an infinite number of sample points.

For a comprehensive passivity check, the Hamiltonian Matrix pencil $(\mathcal{J}, \mathcal{K})$ is the standard method used instead of the sampling method. The pencil is defined

as [86, 133, 134] :

$$\mathcal{J} = \begin{bmatrix} \mathbf{A} - \mathbf{B}\mathbf{D}^T\mathbf{S}^{-1}\mathbf{C} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ \mathbf{C}^T\mathbf{S}^{-1}\mathbf{C} & -\mathbf{A}^T + \mathbf{C}^T\mathbf{D}\mathbf{R}^{-1}\mathbf{B}^T \end{bmatrix} \in \mathbb{R}^{2n \times 2n} \quad (2.112)$$

$$\mathcal{K} = \begin{bmatrix} \mathbf{E} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}^T \end{bmatrix} \in \mathbb{R}^{2n \times 2n} \quad (2.113)$$

where $\mathbf{S} = \mathbf{D}\mathbf{D}^T - \mathbf{I} \in \mathbb{R}^{\bar{p} \times \bar{p}}$ and $\mathbf{R} = \mathbf{D}^T\mathbf{D} - \mathbf{I} \in \mathbb{R}^{\bar{m} \times \bar{m}}$. The special property of matrix pencil $(\mathcal{J}, \mathcal{K})$ is that if the imaginary value $j\psi_z$ for $\psi_z \in \mathbb{R}$ is an eigenvalue of matrix pencil $(\mathcal{J}, \mathcal{K})$, then at least one of the singular value of $\mathbf{G}(j\psi)$ must be 1 [134]. More concretely:

$$j\psi_z \in \text{eig}\{\mathcal{J}, \mathcal{K}\} \rightarrow 1 \in \sigma(\mathbf{G}(j\psi_z)), \quad \psi_z \in \mathbb{R} \quad (2.114)$$

Given the passivity condition defined as (2.111), we have that existence of imaginary

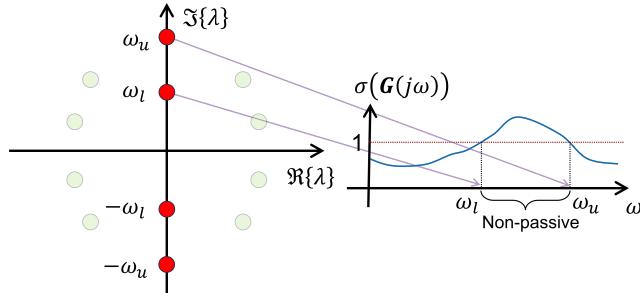


Figure 2.7: Association of imaginary eigenvalues of $(\mathcal{J}, \mathcal{K})$ with the singular value curves of $\mathbf{G}(s)$. Any imaginary entry in $\text{eig}\{\mathcal{J}, \mathcal{K}\}$ (left) must have corresponding transfer function singular value crossing point with threshold 1 (right).

eigenvalues in $\text{eig}\{\mathcal{J}, \mathcal{K}\}$ is sufficient proof that a system is not passive. Furthermore, these very imaginary eigenvalues pinpoint exactly where we have passivity violation over the frequency band, as demonstrated by the illustration on Figure 2.7 where

we see two imaginary eigenvalues are associated to their respective transfer function singular value crossing point with threshold 1.

We list a few additional properties to the eigenvalues of $(\mathcal{J}, \mathcal{K})$ that are of use in later derivations:

1. if $\alpha + j\beta \in \text{eig}\{\mathcal{J}, \mathcal{K}\}$, then the following is true [95]:

$$\pm\alpha \pm j\beta \in \text{eig}\{\mathcal{J}, \mathcal{K}\} \quad (2.115)$$

which means complex eigenvalues of $(\mathcal{J}, \mathcal{K})$ come in groups of four distributed symmetrically across both real and imaginary axis. Incidentally, purely real or imaginary eigenvalues come in pairs.

2. If \mathbf{y} and \mathbf{x} are the left and right eigenvectors, respectively, of a purely imaginary eigenvalue $j\beta \in \text{eig}\{\mathcal{J}, \mathcal{K}\}$, then we have the following relation [95]:

$$\mathbf{y} = \mathbf{J}_0 \mathbf{x} \quad (2.116)$$

where the skew-symmetric matrix \mathbf{J}_0 is defined as:

$$\mathbf{J}_0 = \begin{bmatrix} \mathbf{0}_{n \times n} & \mathbf{I}_{n \times n} \\ -\mathbf{I}_{n \times n} & \mathbf{0}_{n \times n} \end{bmatrix} \quad (2.117)$$

3. Given (2.114), we have that an imaginary eigenvalue $j\psi_z$ of $(\mathcal{J}, \mathcal{K})$ indicates the crossing point of a singular value curve with threshold 1. However, the imaginary eigenvalue does not indicate whether the curve goes above 1 or dive below 1 at the crossing point. The direction of the crossing can be identified by assessing the slope/derivative of the singular value curve at the crossing point

which is quantified as [135]:

$$d_z = \frac{1i \cdot \mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z}{\mathbf{x}_z^H \mathbf{J}_0 \mathcal{J}' \mathbf{x}_z} \in \mathbb{R} \quad (2.118)$$

where \mathbf{x}_z is the right eigenvector associated to $j\psi_z$ and:

$$\mathcal{J}' = \begin{bmatrix} -2\mathbf{B}\mathbf{R}^{-2}\mathbf{D}^T\mathbf{C} & -\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T - 2\mathbf{B}\mathbf{R}^{-2}\mathbf{B}^T \\ \mathbf{C}^T\mathbf{S}^{-1}\mathbf{C} + 2\mathbf{C}^T\mathbf{S}^{-2}\mathbf{C} & 2\mathbf{C}^T\mathbf{D}\mathbf{R}^{-2}\mathbf{B}^T \end{bmatrix} \quad (2.119)$$

Take note that not having any imaginary eigenvalue in $eig\{\mathcal{J}, \mathcal{K}\}$ still requires an additional step of a single sampling of singular value before being sure if the system is passive, as it is still possible we simply have a scenario where all singular values are above 1.

2.4.2 Hamiltonian Matrix Eigenvalue Perturbation

The method of passivity enforcement using the Hamiltonian Matrix (2.112) involves perturbation of the imaginary eigenvalues until they disappear through merging. The final form of the perturbation equation depends on which part of the transfer function (2.9) is to be modified. Naturally, we choose to perturb matrices that will not severely impact on the performance of the system in approximating the original system's behaviour. To ensure system poles are not perturbed, \mathbf{E} and \mathbf{A} must not be modified. Perturbing \mathbf{D} would modify the system response over the entire frequency band, which is why it is also not to be modified. The only viable choice for perturbation are the residue matrices \mathbf{B} and \mathbf{C} . To avoid unnecessarily inflating this document, we focus on perturbing \mathbf{C} only, though perturbation of \mathbf{B} is practically a mirror image version through transpose of the transfer function $\mathbf{G}(s)$.

Let our target of perturbation be $\lambda \in \text{eig}\{\mathcal{J}, \mathcal{K}\}$ with associated left eigenvector \mathbf{y} and right eigenvector \mathbf{x} . Starting with the definition of generalized eigenvalue:

$$\lambda = \frac{\mathbf{y}^H \mathcal{J} \mathbf{x}}{\mathbf{y}^H \mathcal{K} \mathbf{x}} \quad (2.120)$$

If the only source of change in the transfer function is from matrix \mathbf{C} , then only \mathcal{J} is modified and we define the change in \mathcal{J} as $d\mathcal{J}$. The first order eigenvalue perturbation equation due to $d\mathcal{J}$ is defined as [136]:

$$d\lambda = \frac{\mathbf{y}^H d\mathcal{J} \mathbf{x}}{\mathbf{y}^H \mathcal{K} \mathbf{x}} \quad (2.121)$$

$$\tilde{\lambda} - \lambda = \frac{\mathbf{y}^H d\mathcal{J} \mathbf{x}}{\mathbf{y}^H \mathcal{K} \mathbf{x}} \quad (2.122)$$

where $\tilde{\lambda}$ is the resulting modified eigenvalue and

$$d\mathcal{J} = \begin{bmatrix} -\mathbf{B} \mathbf{D}^T \mathbf{S}^{-1} d\mathbf{C} & \mathbf{0} \\ d\mathbf{C}^T \mathbf{S}^{-1} \mathbf{C} + \mathbf{C}^T \mathbf{S}^{-1} d\mathbf{C} & d\mathbf{C}^T \mathbf{D} \mathbf{R}^{-1} \mathbf{B}^T \end{bmatrix} \quad (2.123)$$

Within the context of passivity enforcement, we are only trying to perturb the imaginary eigenvalues $j\psi_z$ of $(\mathcal{J}, \mathcal{K})$, which is why we modify eq. (2.122) into the following form:

$$j\tilde{\psi}_z - j\psi_z = \frac{\mathbf{y}_z^H d\mathcal{J} \mathbf{x}_z}{\mathbf{y}_z^H \mathcal{K} \mathbf{x}_z} \quad (2.124)$$

where \mathbf{y}_z and \mathbf{x}_z are the left and right eigenvectors of $j\psi_z$, respectively. Given property (2.116), we can change (2.124) into:

$$j\tilde{\psi}_z - j\psi_z = \frac{\mathbf{x}_z^H \mathbf{J}_0 d\mathcal{J} \mathbf{x}_z}{\mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z} \quad (2.125)$$

Given that $\mathbf{J}_0 d\mathcal{J}$ is real symmetric and $\mathbf{J}_0 \mathcal{K}$ is real skew-symmetric, one can show that

$$\mathbf{x}_z^H \mathbf{J}_0 d\mathcal{J} \mathbf{x}_z \in \mathbb{R}^{1 \times 1}$$

$$\mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z \in \mathbb{I}^{1 \times 1}$$

which allows us to write (2.125) as follow:

$$\begin{aligned} j\tilde{\psi}_z - j\psi_z &= \frac{\Re\{\mathbf{x}_z^H \mathbf{J}_0 d\mathcal{J} \mathbf{x}_z\}}{j \cdot \Im\{\mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z\}} \\ (j\tilde{\psi}_z - j\psi_z) \cdot j \cdot \Im\{\mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z\} &= \Re\{\mathbf{x}_z^H \mathbf{J}_0 d\mathcal{J} \mathbf{x}_z\} \\ (-\tilde{\psi}_z + \psi_z) \cdot \Im\{\mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z\} &= \Re\{\mathbf{x}_z^H \mathbf{J}_0 d\mathcal{J} \mathbf{x}_z\} \end{aligned} \quad (2.126)$$

Dissect the matrices in equation (2.126) into half-and-half by defining equal subdivision in the eigenvectors:

$$\mathbf{x}_z = \begin{bmatrix} \mathbf{x}_{za} \\ \mathbf{x}_{zb} \end{bmatrix} \in \mathbb{C}^{2n \times 1}, \quad \mathbf{x}_{za}, \mathbf{x}_{zb} \in \mathbb{C}^{n \times 1} \quad (2.127)$$

which allows us to write the term $\mathbf{x}_z^H \mathbf{J}_0 d\mathcal{J} \mathbf{x}_z$ as follow:

$$\begin{aligned} \mathbf{x}_z^H \mathbf{J}_0 d\mathcal{J} \mathbf{x}_z &= \begin{bmatrix} \mathbf{x}_{za}^H & \mathbf{x}_{zb}^H \end{bmatrix} \begin{bmatrix} d\mathbf{C}^T \mathbf{S}^{-1} \mathbf{C} + \mathbf{C}^T \mathbf{S}^{-1} d\mathbf{C} & d\mathbf{C}^T \mathbf{D} \mathbf{R}^{-1} \mathbf{B}^T \\ \mathbf{B} \mathbf{D}^T \mathbf{S}^{-1} d\mathbf{C} & \mathbf{0}_{n \times n} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{za} \\ \mathbf{x}_{zb} \end{bmatrix} \\ &= \mathbf{x}_{za}^H d\mathbf{C}^T \mathbf{S}^{-1} \mathbf{C} \mathbf{x}_{za} + \mathbf{x}_{za}^H \mathbf{C}^T \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za} + \\ &\quad \mathbf{x}_{zb}^H \mathbf{B} \mathbf{D}^T \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za} + \mathbf{x}_{za}^H d\mathbf{C}^T \mathbf{D} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{x}_{zb} \end{aligned} \quad (2.128)$$

and to write the term $\mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z$ as follow:

$$\begin{aligned}\mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z &= \begin{bmatrix} \mathbf{x}_{za}^H & \mathbf{x}_{zb}^H \end{bmatrix} \begin{bmatrix} \mathbf{0}_{n \times n} & \mathbf{E}^T \\ -\mathbf{E} & \mathbf{0}_{n \times n} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{za} \\ \mathbf{x}_{zb} \end{bmatrix} \\ &= -\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za} + \mathbf{x}_{za}^H \mathbf{E}^T \mathbf{x}_{zb}\end{aligned}\quad (2.129)$$

The left-hand side of (2.126) can now be rewritten as follow:

$$\begin{aligned}(-\tilde{\psi}_z + \psi_z) \Im\{\mathbf{x}_z^H \mathbf{J}_0 \mathcal{K} \mathbf{x}_z\} &= \\ (-\tilde{\psi}_z + \psi_z) \Im\{-\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za} + \mathbf{x}_{za}^H \mathbf{E}^T \mathbf{x}_{zb}\} &= \\ (-\tilde{\psi}_z + \psi_z) \Im\{-\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za} - \mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} &= \\ 2(\tilde{\psi}_z - \psi_z) \Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\}\end{aligned}\quad (2.130)$$

The right-hand side of (2.126) can be also be rewritten:

$$\begin{aligned}\Re\{\mathbf{x}_z^H \mathbf{J}_0 d \mathcal{J} \mathbf{x}_z\} &= \\ \Re\{\mathbf{x}_{za}^H d \mathbf{C}^T \mathbf{S}^{-1} \mathbf{C} \mathbf{x}_{za} + \mathbf{x}_{za}^H \mathbf{C}^T \mathbf{S}^{-1} d \mathbf{C} \mathbf{x}_{za} + \\ \mathbf{x}_{zb}^H \mathbf{B} \mathbf{D}^T \mathbf{S}^{-1} d \mathbf{C} \mathbf{x}_{za} + \mathbf{x}_{za}^H d \mathbf{C}^T \mathbf{D} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{x}_{zb}\} &= \\ \Re\{2 \cdot \mathbf{x}_{za}^H \mathbf{C}^T \mathbf{S}^{-1} d \mathbf{C} \mathbf{x}_{za} + \mathbf{x}_{zb}^H \mathbf{B} \mathbf{D}^T \mathbf{S}^{-1} d \mathbf{C} \mathbf{x}_{za} + \mathbf{x}_{zb}^H \mathbf{B} \mathbf{R}^{-1} \mathbf{D}^T d \mathbf{C} \mathbf{x}_{za}\}\end{aligned}\quad (2.131)$$

To further transform equation (2.131), we take note of the following relation when \mathbf{D} is square and not singular:

$$\begin{aligned}\mathbf{D}^T \mathbf{S}^{-1} &= \mathbf{D}^T (\mathbf{D} \mathbf{D}^T - \mathbf{I})^{-1} \\ &= ((\mathbf{D} \mathbf{D}^T - \mathbf{I}) \mathbf{D}^{-T})^{-1} \\ &= (\mathbf{D} - \mathbf{D}^{-T})^{-1}\end{aligned}$$

$$\begin{aligned}
 &= (\mathbf{D}^{-T}(\mathbf{D}^T \mathbf{D} - \mathbf{I}))^{-1} \\
 &= (\mathbf{D}^T \mathbf{D} - \mathbf{I})^{-1} \mathbf{D}^T = \mathbf{R}^{-1} \mathbf{D}^T
 \end{aligned} \tag{2.132}$$

The relation (2.132) allows us to write (2.131) as:

$$\begin{aligned}
 \Re\{\mathbf{x}_z^H \mathbf{J}_0 d\mathcal{J} \mathbf{x}_z\} &= \Re\{2\mathbf{x}_{za}^H \mathbf{C}^T \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za} + 2\mathbf{x}_{zb}^H \mathbf{B} \mathbf{D}^T \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za}\} \\
 &= 2 \cdot \Re\{\mathbf{x}_{za}^H \mathbf{C}^T \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za} + \mathbf{x}_{zb}^H \mathbf{B} \mathbf{D}^T \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za}\}
 \end{aligned} \tag{2.133}$$

Using the updated left-hand side of (2.126) transformed into (2.130) and the right-hand side of (2.126) transformed into (2.133), we rewrite (2.126) into the following form:

$$\begin{aligned}
 2(\tilde{\psi}_z - \psi_z) \Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} &= 2 \cdot \Re\{\mathbf{x}_{za}^H \mathbf{C}^T \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za} + \mathbf{x}_{zb}^H \mathbf{B} \mathbf{D}^T \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za}\} \\
 (\tilde{\psi}_z - \psi_z) \Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} &= \Re\{(\mathbf{x}_{za}^H \mathbf{C}^T + \mathbf{x}_{zb}^H \mathbf{B} \mathbf{D}^T) \mathbf{S}^{-1} d\mathbf{C} \mathbf{x}_{za}\} \\
 (\tilde{\psi}_z - \psi_z) \Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} &= \Re\{\mathbf{u}_z^H d\mathbf{C} \mathbf{x}_{za}\}
 \end{aligned} \tag{2.134}$$

where

$$\mathbf{u}_z = \mathbf{S}^{-1}(\mathbf{C} \mathbf{x}_{za} + \mathbf{D} \mathbf{B}^T \mathbf{x}_{zb}) \tag{2.135}$$

The derivation continues with the use of the Kronecker product property:

$$\begin{aligned}
 \mathbf{P} \mathbf{X} \mathbf{Q} &= \mathbf{Y} \rightarrow \\
 \text{vec}(\mathbf{P} \mathbf{X} \mathbf{Q}) &= \text{vec}(\mathbf{Y}) \rightarrow \\
 (\mathbf{Q}^T \otimes \mathbf{P}) \text{vec}(\mathbf{X}) &= \text{vec}(\mathbf{Y})
 \end{aligned} \tag{2.136}$$

where \otimes is the Kronecker product sign and $vec(\mathbf{X})$ is an operator which transforms a matrix \mathbf{X} into a column vector resulting from stacking the columns of \mathbf{X} one on top of each other starting from leftmost column towards the rightmost one. The matrices \mathbf{P} , \mathbf{X} , \mathbf{Q} , and \mathbf{Y} are token matrices. The property (2.136) allows us to write (2.134) in the following form:

$$(\tilde{\psi}_z - \psi_z)\Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} = \Re\{(\mathbf{x}_{za}^T \otimes \mathbf{u}_z^H)\} vec(d\mathbf{C}) \quad (2.137)$$

We can solve for $d\mathbf{C}$ by assembling all index cases $z = 1, \dots, \bar{z}$ of (2.137) and perform simple regression on the linear system of equations:

$$\begin{bmatrix} \Re\{\mathbf{x}_{1a}^T \otimes \mathbf{u}_1^H\} \\ \Re\{\mathbf{x}_{2a}^T \otimes \mathbf{u}_2^H\} \\ \vdots \\ \Re\{\mathbf{x}_{\bar{z}a}^T \otimes \mathbf{u}_{\bar{z}}^H\} \end{bmatrix} vec(d\mathbf{C}) = \begin{bmatrix} (\tilde{\psi}_1 - \psi_1)\Im\{\mathbf{x}_{1b}^H \mathbf{E} \mathbf{x}_{1a}\} \\ (\tilde{\psi}_2 - \psi_2)\Im\{\mathbf{x}_{2b}^H \mathbf{E} \mathbf{x}_{2a}\} \\ \vdots \\ (\tilde{\psi}_{\bar{z}} - \psi_{\bar{z}})\Im\{\mathbf{x}_{\bar{z}b}^H \mathbf{E} \mathbf{x}_{\bar{z}a}\} \end{bmatrix} \quad (2.138)$$

2.4.3 Singular Value Perturbation

The full description and derivation of the process can be found in [99]. The target of perturbation are $\sigma(\mathbf{G}(s))$, the singular values of $\mathbf{G}(s)$, and specifically those that are ≥ 1 and thus violate the passivity condition. To achieve this, we desire the following perturbation:

$$\tilde{\mathbf{G}}(s) = \mathbf{G}(s) + d\mathbf{G}(s) \quad (2.139)$$

where we once again define $d\mathbf{G}(s)$ with the perturbation restricted onto the \mathbf{C} matrix for consistency with the Hamiltonian eigenvalue perturbation method presented in

Section 2.4.2:

$$d\mathbf{G}(s) = d\mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} \quad (2.140)$$

Rather than directly perturbing singular values, we perturb the eigenvalue equivalent by defining the following analogous form of the system response [99]:

$$\mathbf{Z}(s) = \mathbf{I}_{n \times n} - \mathbf{G}(s)^H \mathbf{G}(s) \quad (2.141)$$

Matrix function $\mathbf{Z}(s)$ has the key properties of having only real eigenvalues by virtue of being an hermitian matrix and these eigenvalues become negative only if their corresponding singular value in $\mathbf{G}(s)$ becomes larger than 1 [99]. Effectively, the passivity condition for $\mathbf{G}(j\omega)$ becomes

$$\text{eig}\{\mathbf{Z}(j\omega)\} > 0, \quad \forall \omega \in \mathbb{R} \quad (2.142)$$

for $\mathbf{Z}(s)$. Thus, the goal of the perturbation is to turn any negative eigenvalues of $\mathbf{Z}(s)$ into positive ones, which is equivalent to pushing the singular values of $\mathbf{G}(s)$ that are ≥ 1 below 1.

Without changing the initial perturbation form of the system response defined by (2.139) and (2.140), we need to translate these changes for the case of the matrix function $\mathbf{Z}(s)$ defined as (2.141):

$$\begin{aligned} \tilde{\mathbf{Z}}(s) &= \mathbf{I}_{n \times n} - \tilde{\mathbf{G}}(s)^H \tilde{\mathbf{G}}(s) \\ \tilde{\mathbf{Z}}(s) &= \mathbf{I}_{n \times n} - [\mathbf{G}(s) + d\mathbf{G}(s)]^H [\mathbf{G}(s) + d\mathbf{G}(s)] \\ \tilde{\mathbf{Z}}(s) &= \mathbf{I}_{n \times n} - \mathbf{G}(s)^H \mathbf{G}(s) - \mathbf{G}(s)^H d\mathbf{G}(s) - d\mathbf{G}(s)^H \mathbf{G}(s) - d\mathbf{G}(s)^H d\mathbf{G}(s) \end{aligned} \quad (2.143)$$

We eliminate the 2nd order term $d\mathbf{G}(s)^H d\mathbf{G}(s)$ from (2.143) to allow the use of regression:

$$\begin{aligned}\tilde{\mathbf{Z}}(s) &\cong \mathbf{I}_{n \times n} - \mathbf{G}(s)^H \mathbf{G}(s) - \mathbf{G}(s)^H d\mathbf{G}(s) - d\mathbf{G}(s)^H \mathbf{G}(s) \\ \tilde{\mathbf{Z}}(s) &\cong \mathbf{Z}(s) - \mathbf{G}(s)^H d\mathbf{G}(s) - d\mathbf{G}(s)^H \mathbf{G}(s)\end{aligned}\quad (2.144)$$

From (2.144), the change matrix $d\mathbf{Z}(s)$ applied to $\mathbf{Z}(s)$ can be clearly defined as:

$$d\mathbf{Z}(s) = -\mathbf{G}(s)^H d\mathbf{G}(s) - d\mathbf{G}(s)^H \mathbf{G}(s) \quad (2.145)$$

Now, in order to expose the term $d\mathbf{C}$ in order to perturb it, define extra variable:

$$\mathbf{Q}(s) = (s\mathbf{E} - \mathbf{A})^{-1} \mathbf{B} \quad (2.146)$$

which allows us to rewrite (2.145) as follow:

$$d\mathbf{Z}(s) = -\mathbf{G}(s)^H d\mathbf{C} \mathbf{Q}(s) - \mathbf{Q}(s)^H d\mathbf{C}^H \mathbf{G}(s) \quad (2.147)$$

Equation (2.147) clearly defines the change to $d\mathbf{Z}(s)$ as a function of $d\mathbf{C}$. In order to proceed further with the derivation, we need to deal with the variability with s in equation (2.147). To perform perturbation over the entire frequency band is an overreach of the scope of the perturbation, so we limit the definition (2.147) to only frequency points s_r of interest, namely, where we have $\text{eig}\{\mathbf{Z}(s_r)\} \leq 0$, which is sign of passivity violation. From this point on, we use the frequency independent form of (2.147) that focuses on only a set of frequency evaluation points:

$$d\mathbf{Z}(s_r) = d\mathbf{Z}_r = -\mathbf{G}_r^H d\mathbf{C} \mathbf{Q}_r - \mathbf{Q}_r^H d\mathbf{C}^H \mathbf{G}_r, \quad r = 1, 2, \dots, \bar{r} \quad (2.148)$$

for:

$$\mathbf{G}_r = \mathbf{G}(s_r), \quad \mathbf{Z}_r = \mathbf{Z}(s_r)$$

We can now proceed to define the eigenvalue perturbation equation for $\mathbf{Z}(s)$. The eigenvalue perturbation equation of 1st degree can be readily borrowed from (2.122) with some small adjustments:

$$\tilde{\lambda}_r - \lambda_r = \frac{\mathbf{x}_r^H d\mathbf{Z}_r \mathbf{x}_r}{\mathbf{x}_r^H \mathbf{x}_r} \quad (2.149)$$

where λ_r is an eigenvalue of \mathbf{Z}_r that is to be perturbed into its new value $\tilde{\lambda}_r$ and \mathbf{x}_r is both the right and left eigenvectors associated to λ_r since \mathbf{Z} is hermitian. The derivation continues as follow:

$$\begin{aligned} (\tilde{\lambda}_r - \lambda_r) \mathbf{x}_r^H \mathbf{x}_r &= \mathbf{x}_r^H d\mathbf{Z}_r \mathbf{x}_r \\ (\tilde{\lambda}_r - \lambda_r) \mathbf{x}_r^H \mathbf{x}_r &= -\mathbf{x}_r^H \mathbf{G}_r^H d\mathbf{C} \mathbf{Q}_r \mathbf{x}_r - \mathbf{x}_r^H \mathbf{Q}_r^H d\mathbf{C}^H \mathbf{G}_r \mathbf{x}_r \end{aligned} \quad (2.150)$$

Noting the fact that the right hand side of (2.150) is merely a scalar c added to its hermitian c^H which results in $2 \cdot \Re\{c\}$, we have

$$(\tilde{\lambda}_r - \lambda_r) \mathbf{x}_r^H \mathbf{x}_r = -2 \cdot \Re\{\mathbf{x}_r^H \mathbf{G}_r^H d\mathbf{C} \mathbf{Q}_r \mathbf{x}_r\} \quad (2.151)$$

$$(\tilde{\lambda}_r - \lambda_r) \mathbf{x}_r^H \mathbf{x}_r = -2 \cdot \Re\{(\mathbf{Q}_r \mathbf{x}_r)^T \otimes (\mathbf{x}_r^H \mathbf{G}_r^H)\} \text{vec}(d\mathbf{C}) \quad (2.152)$$

In general, there might be more than one eigenvalue to perturb at each frequency s_r of passivity violation, so we need to add an additional eigenvalue index z to the perturbation equation (2.152):

$$(\tilde{\lambda}_{r,z} - \lambda_{r,z}) \mathbf{x}_{r,z}^H \mathbf{x}_{r,z} = -2 \cdot \Re\{(\mathbf{Q}_r \mathbf{x}_{r,z})^T \otimes (\mathbf{x}_{r,z}^H \mathbf{G}_r^H)\} \text{vec}(d\mathbf{C}) \quad (2.153)$$

where

- $\lambda_{r,z}$ is the eigenvalue of index z from \mathbf{Z}_r .
- $\tilde{\lambda}_{r,z}$ is the intended perturbation result for $\lambda_{r,z}$.
- $\mathbf{x}_{r,z}$ is the right/left eigenvector associated to $\lambda_{r,z}$ from \mathbf{Z}_r .

$\text{vec}(d\mathbf{C})$ can be solved by stacking all cases of (2.153) together to form a linear system of equation and perform simple regression on it. It should be noted that the number of eigenvalues perturbed at each frequency point s_r can differ from point to point on a case by case basis.

2.4.4 Gramian Based Error Control

The perturbation equations (2.137) and (2.153) can provide a solution $d\mathbf{C}$ through solving a linear system of equations, but this approach has no explicit error control other than the minimization of the norm of $d\mathbf{C}$. The more focused error control solution relies on the controllability gramian when perturbing \mathbf{C} [88] or the observability gramian when perturbing \mathbf{B} . We will only demonstrate error control using the controllability gramian for perturbation of \mathbf{C} , as the method of perturbing \mathbf{B} is practically a mirror image through transpose of the transfer function $\mathbf{G}(s)$.

We start with error quantification which can be done using quantification of the system response through the use of H_2 -norm of the transfer function [115]:

$$\begin{aligned}\|\mathbf{G}(\cdot)\|_2 &= \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{tr} \left(\mathbf{G}(j\omega) \mathbf{G}(j\omega)^H \right) d\omega} \\ \|\mathbf{G}(\cdot)\|_2^2 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{tr} \left(\mathbf{G}(j\omega) \mathbf{G}(j\omega)^H \right) d\omega \\ \|\mathbf{G}(\cdot)\|_2^2 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{tr} \left(\mathbf{C}(j\omega \mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \mathbf{B}^T (j\omega \mathbf{I} - \mathbf{A})^{-H} \mathbf{C}^T \right) d\omega\end{aligned}$$

$$\Omega = \frac{1}{2\pi} \int_{-\infty}^{\infty} \text{tr} (\mathbf{C}(j\omega \mathbf{I} - \mathbf{A})^{-1} \mathbf{B} \mathbf{B}^T (j\omega \mathbf{I} - \mathbf{A})^{-H} \mathbf{C}^T) d\omega \quad (2.154)$$

where $\text{tr}(\cdot)$ is the matrix trace function and where we have defined the norm value Ω as an intermediate variable to alleviate the notation. To proceed with the derivation, we alter Ω as follow:

$$\begin{aligned} \Omega &= \text{tr} \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \mathbf{C}(j\omega \mathbf{E} - \mathbf{A})^{-1} \mathbf{B} \mathbf{B}^T (j\omega \mathbf{E} - \mathbf{A})^{-H} \mathbf{C}^T d\omega \right) \\ &= \text{tr} \left(\mathbf{C} \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} (j\omega \mathbf{E} - \mathbf{A})^{-1} \mathbf{B} \mathbf{B}^T (j\omega \mathbf{E} - \mathbf{A})^{-H} d\omega \right) \mathbf{C}^T \right) \end{aligned} \quad (2.155)$$

The controllability gramian \mathcal{G} of our descriptor system can be defined as [137]:

$$\mathcal{G} = \frac{1}{2\pi} \int_{-\infty}^{\infty} (j\omega \mathbf{E} - \mathbf{A})^{-1} \mathbf{B} \mathbf{B}^T (j\omega \mathbf{E} - \mathbf{A})^{-H} d\omega \quad (2.156)$$

which allows us to write Ω definition at (2.155) in the more compact form:

$$\Omega = \text{tr} (\mathbf{C} \mathcal{G} \mathbf{C}^T) \quad (2.157)$$

Equation (2.157) provides a mean to quantify the system perturbation error in the specific scenario when the only matrix being perturbed is \mathbf{C} . If \mathbf{C} is perturbed with $d\mathbf{C}$ as follow:

$$\tilde{\mathbf{C}} = \mathbf{C} + d\mathbf{C}, \quad (2.158)$$

we can quantify the perturbation error as:

$$\begin{aligned} \Omega + d\Omega &= \text{tr} ((\mathbf{C} + d\mathbf{C}) \mathcal{G} (\mathbf{C} + d\mathbf{C})^T) \\ \Omega + d\Omega &= \text{tr} (\mathbf{C} \mathcal{G} \mathbf{C}^T + \mathbf{C} \mathcal{G} d\mathbf{C}^T + d\mathbf{C} \mathcal{G} \mathbf{C}^T + d\mathbf{C} \mathcal{G} d\mathbf{C}^T) \\ d\Omega &= \text{tr} (\mathbf{C} \mathcal{G} d\mathbf{C}^T + d\mathbf{C} \mathcal{G} \mathbf{C}^T + d\mathbf{C} \mathcal{G} d\mathbf{C}^T) \end{aligned}$$

$$d\Omega = 2 \cdot \text{tr} (dC\mathbf{G}C^T) + \text{tr} (dC\mathbf{G}dC^T) = \Delta \quad (2.159)$$

where we denote the system perturbation error as Δ . An important property of a continuous time LTI descriptor system (with non-singular \mathbf{E}) is that its gramian \mathbf{G} can be solved more directly as the solution to the following generalized Lyapunov equation [137]:

$$\mathbf{E}\mathbf{G}\mathbf{A}^T + \mathbf{A}\mathbf{G}\mathbf{E}^T = -\mathbf{B}\mathbf{B}^T \quad (2.160)$$

and if our system is stable, which we have already assumed, \mathbf{G} is positive definite and can undergo Choleski factorization:

$$\mathbf{G} = \mathbf{L}^T \mathbf{L} \quad (2.161)$$

which allows us to write (2.159) in the updated form:

$$\Delta = 2 \cdot \text{tr} (dC\mathbf{L}^T \mathbf{L}C^T) + \text{tr} (dC\mathbf{L}^T \mathbf{L}dC^T) \quad (2.162)$$

We define here a property of the Frobenius-norm (F-norm) related to matrix trace:

$$\|\mathbf{X}\|_F^2 = \text{tr} (\mathbf{X} \mathbf{X}^H) \quad (2.163)$$

on a token matrix \mathbf{X} . The property (2.163) allows us to write the second order term of perturbation error (2.162) as a F-norm:

$$\Delta = 2 \cdot \text{tr} (dC\mathbf{L}^T \mathbf{L}C^T) + \|dC\mathbf{L}^T\|_F^2 \quad (2.164)$$

Considering the second order term in the perturbation error equation (2.164) as the dominant error term, a change of variable by direct substitution is now possible

and beneficial because the norm minimization of $d\mathbf{C}$ is not as effective as the norm minimization of $d\mathbf{C}\mathbf{L}^T$. The latter option of minimization is more aligned with the norm minimization of Δ as can be assessed with equation (2.164). Thus, we use the following change of variable:

$$\begin{aligned} d\mathbf{C}\mathbf{L}^T &= d\mathbf{C}_L \\ d\mathbf{C} &= d\mathbf{C}_L \mathbf{L}^{-T} \end{aligned} \quad (2.165)$$

The change of variable (2.165) can be applied on the Hamiltonian eigenvalue perturbation equation starting at equation (2.134):

$$\begin{aligned} (\tilde{\psi}_z - \psi_z) \Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} &= 2 \cdot \Re\{\mathbf{u}_z^H d\mathbf{C} \mathbf{x}_{za}\} \\ (\tilde{\psi}_z - \psi_z) \Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} &= 2 \cdot \Re\{\mathbf{u}_z^H d\mathbf{C}_L \mathbf{L}^{-T} \mathbf{x}_{za}\} \\ (\tilde{\psi}_z - \psi_z) \Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} &= 2 \cdot \Re\{((\mathbf{x}_{za}^T \mathbf{L}^{-1}) \otimes \mathbf{u}_z^H)\} \text{vec}(d\mathbf{C}_L) \end{aligned} \quad (2.166)$$

The change of variable (2.165) can also be applied on the singular value perturbation equation starting at equation (2.151):

$$\begin{aligned} (\tilde{\lambda}_r - \lambda_r) \mathbf{x}_r^H \mathbf{x}_r &= -2 \cdot \Re\{\mathbf{x}_r^H \mathbf{G}_r^H d\mathbf{C} \mathbf{Q}_r \mathbf{x}_r\} \\ (\tilde{\lambda}_r - \lambda_r) \mathbf{x}_r^H \mathbf{x}_r &= -2 \cdot \Re\{\mathbf{x}_r^H \mathbf{G}_r^H d\mathbf{C}_L \mathbf{L}^{-T} \mathbf{Q}_r \mathbf{x}_r\} \\ (\tilde{\lambda}_r - \lambda_r) \mathbf{x}_r^H \mathbf{x}_r &= -2 \cdot \Re\{(\mathbf{L}^{-T} \mathbf{Q}_r \mathbf{x}_r)^T \otimes (\mathbf{x}_r^H \mathbf{G}_r^H)\} \text{vec}(d\mathbf{C}_L) \end{aligned} \quad (2.167)$$

Adding the eigenvalue indexing z for each frequency point that we used for (2.153), the final form of (2.167) becomes:

$$(\tilde{\lambda}_{r,z} - \lambda_{r,z}) \mathbf{x}_{r,z}^H \mathbf{x}_{r,z} = -2 \cdot \Re\{(\mathbf{L}^{-T} \mathbf{Q}_r \mathbf{x}_{r,z})^T \otimes (\mathbf{x}_{r,z}^H \mathbf{G}_r^H)\} \text{vec}(d\mathbf{C}_L) \quad (2.168)$$

Perturbation equations (2.166) and (2.168) are equivalent to their respective original versions (2.137) and (2.153), but we have switched the variable of the linear system of equations from $d\mathbf{C}$ to $d\mathbf{C}_L$, which means applying linear regression to solve for $d\mathbf{C}_L$ would more directly minimize the perturbation error Δ .

2.4.5 Filtered Gramian Error Control

In order to concentrate error control on a select frequency band, we can apply filtering on the initial non-passive system which suppresses the system's response outside the relevant frequency band of operation. From this filtered system, we then extract a controllability gramian and conduct the passivity enforcement as per usual. The entirety of this error control method can be referred to [99].

Suppose that our filter is represented by a descriptor system (non-singular \mathbf{E}_Y):

$$\mathbf{F}(s) = \mathbf{C}_Y(s\mathbf{E}_Y - \mathbf{A}_Y)^{-1}\mathbf{B}_Y + \mathbf{D}_Y \in \mathbb{C}^{\bar{p} \times \bar{m}} \quad (2.169)$$

of system order \bar{f} which we shall assume is stable. Usually, $\mathbf{E}_Y = \mathbf{I}_{\bar{f} \times \bar{f}}$, but we keep the definition general with \mathbf{E}_Y . The system from which we are to extract our ‘filtered’ controllability gramian is given as the combination of our target system $\mathbf{G}(s)$ with the filter system $\mathbf{F}(s)$. The filter system can be combined either at the input or output of our target system. We proceed with the derivations with the filter assigned to the input of the target system. As such, the transfer function of the combined LTI systems can be directly obtained as the multiplication between $\mathbf{G}(s)$ and $\mathbf{F}(s)$:

$$\mathbf{G}_F(s) = \mathbf{G}(s) \cdot \mathbf{F}(s) \in \mathbb{C}^{\bar{p} \times \bar{m}} \quad (2.170)$$

We dub the system represented by the transfer function (2.170) as our weighted system. A direct definition of the matrices of $\mathbf{G}_F(s)$ can be derived by considering the frequency domain state-space models of $\mathbf{G}(s)$ and $\mathbf{F}(s)$:

$$\begin{aligned} s\mathbf{E}\mathbf{x}(s) &= \mathbf{A}\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s) & s\mathbf{E}_T\mathbf{x}_T(s) &= \mathbf{A}_T\mathbf{x}_T(s) + \mathbf{B}_T\mathbf{u}_T(s) \\ \mathbf{y}(s) &= \mathbf{C}\mathbf{x}(s) + \mathbf{D}\mathbf{u}(s) & \mathbf{y}_T(s) &= \mathbf{C}_T\mathbf{x}_T(s) + \mathbf{D}_T\mathbf{u}_T(s) \end{aligned} \quad (2.171)$$

By assigning the $\mathbf{F}(s)$ system at the input of $\mathbf{G}(s)$, we effectively force the input of the $\mathbf{G}(s)$ system as the output of the $\mathbf{F}(s)$ system, i.e. $\mathbf{u}(s) = \mathbf{y}_T(s)$. The system of equation (2.171) can then be updated as follow:

$$\begin{aligned} s\mathbf{E}\mathbf{x}(s) &= \mathbf{A}\mathbf{x}(s) + \mathbf{B}\mathbf{C}_T\mathbf{x}_T(s) + \mathbf{B}\mathbf{D}_T\mathbf{u}_T(s) \\ s\mathbf{E}_T\mathbf{x}_T(s) &= \mathbf{A}_T\mathbf{x}_T(s) + \mathbf{B}_T\mathbf{u}_T(s) \\ \mathbf{y}(s) &= \mathbf{C}\mathbf{x}(s) + \mathbf{D}\mathbf{C}_T\mathbf{x}_T(s) + \mathbf{D}\mathbf{D}_T\mathbf{u}_T(s) \end{aligned} \quad (2.172)$$

We can then combine the two first state equations in (2.172) if we cascade the state vectors $\mathbf{x}(s)$ and $\mathbf{x}_T(s)$ into a unified state vector $\mathbf{x}_F(s)$

$$\begin{aligned} s \begin{bmatrix} \mathbf{E} \\ \mathbf{E}_T \end{bmatrix} \begin{bmatrix} \mathbf{x}(s) \\ \mathbf{x}_T(s) \end{bmatrix} &= \begin{bmatrix} \mathbf{A} & \mathbf{B}\mathbf{C}_T \\ & \mathbf{A}_T \end{bmatrix} \begin{bmatrix} \mathbf{x}(s) \\ \mathbf{x}_T(s) \end{bmatrix} + \begin{bmatrix} \mathbf{B}\mathbf{D}_T \\ \mathbf{B}_T \end{bmatrix} \mathbf{u}_T(s) \\ \mathbf{y}(s) &= \begin{bmatrix} \mathbf{C} & \mathbf{D}\mathbf{C}_T \end{bmatrix} \begin{bmatrix} \mathbf{x}(s) \\ \mathbf{x}_T(s) \end{bmatrix} + \mathbf{D}\mathbf{D}_T\mathbf{u}_T(s) \end{aligned} \quad (2.173)$$

From (2.173), we can directly define the filtered system $\mathbf{G}_F(s)$ as:

$$\begin{aligned} s\mathbf{E}_F\mathbf{x}_F(s) &= \mathbf{A}_F\mathbf{x}_F(s) + \mathbf{B}_F\mathbf{u}(s) \\ \mathbf{y}_F(s) &= \mathbf{C}_F\mathbf{x}_F(s) + \mathbf{D}_F\mathbf{u}(s) \end{aligned} \quad (2.174)$$

where:

$$\begin{aligned}
 \mathbf{E}_F &= \begin{bmatrix} \mathbf{E} \\ \mathbf{E}_\Upsilon \end{bmatrix} \in \mathbb{R}^{(n+\bar{f}) \times (n+\bar{f})} & \mathbf{A}_F &= \begin{bmatrix} \mathbf{A} & \mathbf{B}\mathbf{C}_\Upsilon \\ \mathbf{0} & \mathbf{A}_\Upsilon \end{bmatrix} \in \mathbb{R}^{(n+\bar{f}) \times (n+\bar{f})} \\
 \mathbf{B}_F &= \begin{bmatrix} \mathbf{B}\mathbf{D}_\Upsilon \\ \mathbf{B}_\Upsilon \end{bmatrix} \in \mathbb{R}^{(n+\bar{f}) \times \bar{m}} & \mathbf{C}_F &= \begin{bmatrix} \mathbf{C} & \mathbf{D}\mathbf{C}_\Upsilon \end{bmatrix} \in \mathbb{R}^{\bar{p} \times (n+\bar{f})} \\
 \mathbf{D}_F &= \begin{bmatrix} \mathbf{D}\mathbf{D}_\Upsilon \end{bmatrix} \in \mathbb{R}^{\bar{p} \times \bar{m}} & \mathbf{x}_F &= \begin{bmatrix} \mathbf{x}(s) \\ \mathbf{x}_F(s) \end{bmatrix} \in \mathbb{R}^{(n+\bar{f}) \times 1}
 \end{aligned} \tag{2.175}$$

Note that we use the input vector $\mathbf{u}(s)$ in (2.174) because the addition of the filter simply reassigned the input to the filter's input without any modification.

The controllability gramian of system (2.174) can be found as the solution \mathbf{G}_F of the Lyapunov equation:

$$\mathbf{A}_F \mathbf{G}_F \mathbf{E}_F^T + \mathbf{E}_F \mathbf{G}_F \mathbf{A}_F^T = -\mathbf{B}_F \mathbf{B}_F^T \tag{2.176}$$

The actual portion of \mathbf{G}_F that is used for the error control scheme is the upper-left quadrant $\mathbf{G}_w \in \mathbb{R}^{n \times n}$ of the following dissection of \mathbf{G}_F :

$$\mathbf{G}_F = \begin{bmatrix} \mathbf{G}_w & \mathbf{G}_{F12} \\ \mathbf{G}_{F21} & \mathbf{G}_{F22} \end{bmatrix} \in \mathbb{R}^{(n+\bar{f}) \times (n+\bar{f})} \tag{2.177}$$

\mathbf{G}_w is selected because it is the only portion relevant to matrix \mathbf{C} in \mathbf{C}_F as defined in (2.175), and we are only perturbing \mathbf{C} . From this point on, the error control scheme is identical to the original gramian method presented in (2.4.4). We would proceed with Choleski factorization on the gramian \mathbf{G}_w and then change of variable through direct substitution, with the only difference being \mathbf{G} being substituted with

\mathcal{G}_w . The rational behind the use of filtered gramian \mathcal{G}_F compared to the standard

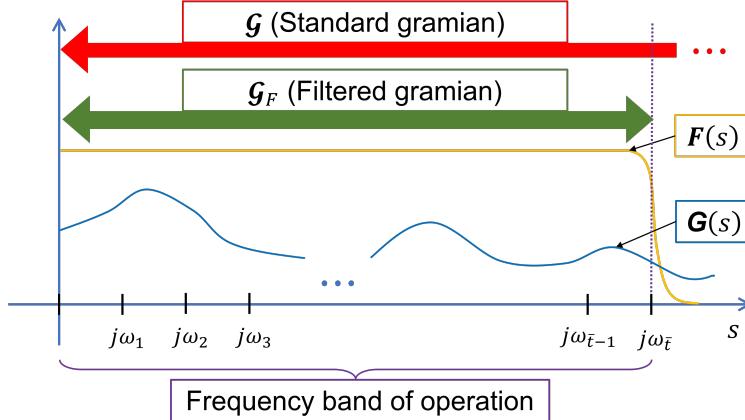


Figure 2.8: Illustration of error control coverage between the standard and the filtered gramian schemes.

gramian \mathcal{G} is illustrated on Figure 3.1, where we observe the error control coverage by \mathcal{G}_F is only the region non-suppressed by the filter function $F(s)$ whereas the error control by \mathcal{G} is the entirety of the frequency band. As a result, for initial systems having non-passive behaviours outside a frequency band of interest, the filtered gramian \mathcal{G}_F would be more efficient than \mathcal{G} in error control as it shifts the perturbation error outside the band of interest.

Chapter 3

Passivity Enforcement

In this chapter, we present the passivity enforcement error control schemes proposed by the thesis. In this thesis, passivity enforcement is applied to S -parameter systems under the descriptor systems representation having the transfer function (2.7) that we rewrite here:

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \quad (3.1)$$

We maintain the assumption made in Section 2.1.2 that \mathbf{E} is non-singular, which effectively means we are working with state-space systems. The general idea of passivity enforcement is the modification of the system matrix set $\{\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ to maintain the singular values of transfer function $\mathbf{G}(s)$ below the maximum threshold 1 at all s , which is the required passivity condition as described in Section 2.1.4. As was first explained in Section 2.4.2, practically speaking, the only suitable matrices for modification are the residue matrices \mathbf{B} and \mathbf{C} , and due to the similarity between the two choices, the thesis concentrates on explaining the passivity enforcement applied on \mathbf{C} . In general, enforcement involves generating a modification $d\mathbf{C}$ to remedy a non-passive system and the natural error control is to simply minimize the norm of $d\mathbf{C}$ in hope of minimizing perturbation error introduced to $\mathbf{G}(s)$ during the process. As was explained in Section 2.4.4, a stronger error control option is to minimize the matrix norm of $d\mathbf{C}\mathbf{L}^T$ instead of the norm of $d\mathbf{C}$ where \mathbf{L} is the

Choleski factor of the system controllability gramian. $d\mathbf{C}\mathbf{L}^T$ being the better option is due to the fact its matrix norm minimization is significantly more aligned with the minimization of the perturbation error introduced to the system. The work of this thesis with regard to passivity enforcement is focused on this dynamic of gramian error control by offering modifications such as more customized error control as well as lower computation cost.

This chapter can be viewed as a direct continuation of Section 2.4 where we first presented the passivity checking method, perturbation methods, and existing error control schemes. This chapter directly adds two more error control schemes on top of the standard gramian scheme of Section 2.4.4 and the filtered gramian scheme of 2.4.5. Once the proposed schemes are presented, simulations are conducted on various test cases where we compare the results of the two proposed schemes against the results of the standard and filtered gramian schemes in order to validate the former two.

3.1 Discrete Gramian Error Control

The derivation of the discrete gramian starts on the common ground with the standard gramian at the definition of the system response squared H_2 -norm quantification Ω defined as (2.157):

$$\Omega = \text{tr}(\mathbf{C}\mathcal{G}\mathbf{C}^T) \quad (3.2)$$

where the controllability gramian \mathcal{G} was defined as (2.156), which we rewrite here:

$$\mathcal{G} = \frac{1}{2\pi} \int_{-\infty}^{\infty} (j\omega\mathbf{E} - \mathbf{A})^{-1} \mathbf{B} \mathbf{B}^T (j\omega\mathbf{E} - \mathbf{A})^{-H} d\omega \quad (3.3)$$

Further derivations from this point on are made easier by using the definition of the transfer function without \mathbf{C} as defined by (2.146) that we rewrite here:

$$\mathbf{Q}(s) = (s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} \quad (3.4)$$

The derivation continues from (3.3) as follow:

$$\mathcal{G} = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathbf{Q}(j\omega) \mathbf{Q}(j\omega)^H d\omega \quad (3.5)$$

We start with limiting the integral to a finite region of evaluation:

$$\mathcal{G}_{\omega_t} = \frac{1}{2\pi} \int_{-\omega_t}^{\omega_t} \mathbf{Q}(j\omega) \mathbf{Q}(j\omega)^H d\omega \in \mathbb{R}^{n \times n} \quad (3.6)$$

The finite integral can then be approximated using simple discretization where we set a linearly distributed sample points set $\mathcal{S}_\Psi = \{\omega_t\}_{t=1,2,\dots,\bar{t}}$ and add the contribution of the integrand evaluated at these sample points scaled by the distance $d\omega$ between each consecutive sample point:

$$\mathcal{Q} = \frac{1}{2\pi} d\omega \sum_{t=1}^{\bar{t}} [\mathbf{Q}(j\omega_t) \mathbf{Q}(j\omega_t)^H + \mathbf{Q}(-j\omega_t) \mathbf{Q}(-j\omega_t)^H] \in \mathbb{R}^{n \times n} \quad (3.7)$$

The summation term in (3.7) has form:

$$\mathbf{X}^H \mathbf{X} + \mathbf{X}^{H*} \mathbf{X}^* \quad (3.8)$$

where we have used the temporary matrix variable $\mathbf{X} = \mathbf{Q}(j\omega_t)^H$. We can simplify the expression into purely real form as follow:

$$\mathbf{X}^H \mathbf{X} + \mathbf{X}^{H*} \mathbf{X}^*$$

$$\begin{aligned}
 &= \mathbf{X}^H \mathbf{X} + (\mathbf{X}^H \mathbf{X})^* \\
 &= 2 \cdot \Re\{\mathbf{X}^H \mathbf{X}\} \\
 &= 2 \cdot \Re\{(\Re\{\mathbf{X}\} + 1i \cdot \Im\{\mathbf{X}\})^H (\Re\{\mathbf{X}\} + 1i \cdot \Im\{\mathbf{X}\})\} \\
 &= 2 \cdot \Re\{\Re\{\mathbf{X}\}^T \Re\{\mathbf{X}\} + 1i \cdot \Re\{\mathbf{X}\}^T \Im\{\mathbf{X}\} - 1i \cdot \Im\{\mathbf{X}\}^T \Re\{\mathbf{X}\} + \Im\{\mathbf{X}\}^T \Im\{\mathbf{X}\}\} \\
 &= 2(\Re\{\mathbf{X}\}^T \Re\{\mathbf{X}\} + \Im\{\mathbf{X}\}^T \Im\{\mathbf{X}\})
 \end{aligned} \tag{3.9}$$

With the alternative form (3.9) of the summation term (3.8), we can write (3.7) as follow:

$$\mathbf{Q} = \frac{1}{\pi} d\omega \sum_{t=1}^{\bar{t}} [\Re\{\mathbf{Q}(j\omega_t)^H\}^T \Re\{\mathbf{Q}(j\omega_t)^H\} + \Im\{\mathbf{Q}(j\omega_t)^H\}^T \Im\{\mathbf{Q}(j\omega_t)^H\}] \tag{3.10}$$

which we then write in matrix form as follow:

$$\mathbf{Q} = \frac{1}{\pi} d\omega (\Re\{\Psi_c\}^T \Re\{\Psi_c\} + \Im\{\Psi_c\}^T \Im\{\Psi_c\}) \tag{3.11}$$

where

$$\Psi_c = \begin{bmatrix} \mathbf{Q}(j\omega_1)^H \\ \mathbf{Q}(j\omega_2)^H \\ \vdots \\ \mathbf{Q}(j\omega_{\bar{t}})^H \end{bmatrix} \in \mathbb{C}^{\bar{m}\bar{t} \times n} \tag{3.12}$$

which is the hermitian of the controllability matrix defined in [49], but in the context of an existing approximate LTI descriptor system not necessarily related to Loewner Matrices. A final step of compact rearrangement for \mathbf{Q} gives us the equation:

$$\mathbf{Q} = \frac{1}{\pi} d\omega \cdot \Psi^T \Psi \tag{3.13}$$

where

$$\Psi = \begin{bmatrix} \Re\{\Psi_c\} \\ \Im\{\Psi_c\} \end{bmatrix} \in \mathbb{R}^{2\bar{m}\bar{t} \times n} \quad (3.14)$$

The matrix \mathbf{Q} can already be formally called the discretized controllability gramian. However, the discrete frequency interval magnitude $d\omega$ is problem dependent, and because the term $\frac{1}{\pi}d\omega$ is merely a scaling factor, the discretized gramian can be defined in a more general way through normalization:

$$\mathcal{G}_\Psi = \Psi^T \Psi \quad (3.15)$$

Now, the system response norm Ω defined as (3.2) can be defined in discretized form by replacing \mathcal{G} with its discretized form \mathcal{G}_Ψ :

$$\Omega_\Psi = \text{tr} (\mathbf{C} \mathcal{G}_\Psi \mathbf{C}^T) \quad (3.16)$$

We call the matrix \mathcal{G}_Ψ the discretized controllability gramian, and because it is generated through multiplication of a matrix that is unlikely rank deficient with its own transpose, it can undergo Choleski factorization:

$$\mathcal{G}_\Psi = \mathbf{L}_\Psi^T \mathbf{L}_\Psi \quad (3.17)$$

Now, the perturbation error Δ defined as (2.164) can be written using the discrete gramian Choleski factors:

$$\Delta = 2 \cdot \text{tr} (d\mathbf{C} \mathbf{L}_\Psi^T \mathbf{L}_\Psi \mathbf{C}^T) + \|d\mathbf{C} \mathbf{L}_\Psi^T\|_F^2 \quad (3.18)$$

Again, just like in the case of the true gramian \mathbf{G} , we perform a change of variable such that minimization of the variable aligns with error minimization under the discrete gramian framework:

$$\begin{aligned} d\mathbf{C}_\Psi &= d\mathbf{C}\mathbf{L}_\Psi^T \\ d\mathbf{C} &= d\mathbf{C}_\Psi\mathbf{L}_\Psi^{-T} \end{aligned} \quad (3.19)$$

resulting in the modified version of the Hamiltonian eigenvalue perturbation equation (2.166):

$$(\tilde{\psi}_z - \psi_z) \Im\{\mathbf{x}_{zb}^H \mathbf{E} \mathbf{x}_{za}\} = 2 \cdot \Re\{(\mathbf{x}_{za}^T \mathbf{L}_\Psi^{-1}) \otimes \mathbf{u}_z^H\} \text{vec}(d\mathbf{C}_\Psi) \quad (3.20)$$

and the modified version of the singular value perturbation equation (2.153):

$$(\tilde{\lambda}_{r,z} - \lambda_{r,z}) \mathbf{x}_{r,z}^H \mathbf{x}_{r,z} = -2 \cdot \Re\{(\mathbf{L}_\Psi^{-T} \mathbf{Q}_r \mathbf{x}_{r,z})^T \otimes (\mathbf{x}_{r,z}^H \mathbf{G}_r^H)\} \text{vec}(d\mathbf{C}_\Psi) \quad (3.21)$$

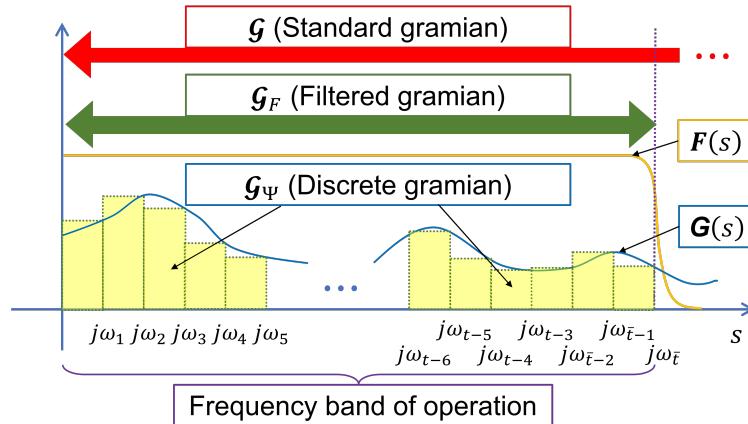


Figure 3.1: Illustration of error control coverage between the standard, filtered, and discrete gramian schemes.

The rational behind the use of \mathcal{G}_Ψ compared to its standard (\mathcal{G}) and filtered (\mathcal{G}_F) counterparts is illustrated on Figure 3.1. The figure illustrates the error control coverage of all three error control schemes, where \mathcal{G} covers the entire frequency band, \mathcal{G}_F covers the frequency band not suppressed by the related filter function $F(s)$ which is low-pass for this example, and \mathcal{G}_Ψ covers through discrete sampling of the system response at regular frequency intervals. Given the above set-up, the minimization of $d\mathbf{C}_\Psi$ mimics the filtered gramian effect and limits the constraint on a frequency band of interest, allowing more liberty to perturb the system response outside the frequency band of interest. As a result, for initial systems having non-passive behaviours outside a frequency band of interest, the discrete-gramian \mathcal{G}_Ψ would be more efficient than \mathcal{G} in error control as it shifts the perturbation error outside the band of interest, but would be similar in efficiency to \mathcal{G}_F . It should be noted that lack of error control outside the frequency band of interest does not mean lack of passivity enforcement outside that band. On the contrary, the perturbation has more leeway to force passivity outside the frequency band of interest given no error control restriction there.

3.2 Loewner Matrix System Method Built-In Error Control

The perturbation error control presented in this section is uniquely applicable to system generated using the LM system method presented in section 2.2.3. We recall that theorem 2.2.1 led to the fact that the LMs can be directly used as the descriptor system matrices as follow:

$$\mathbf{E}_L = -\mathbb{L}, \quad \mathbf{A}_L = -\sigma\mathbb{L}, \quad \mathbf{B}_L = \mathbf{F}, \quad \mathbf{C}_L = \mathbf{W}, \quad \mathbf{D}_L = \mathbf{0}_{\bar{p} \times \bar{m}} \quad (3.22)$$

Assuming $\mathbf{E}_{\mathbb{L}}$ and $\mathbf{A}_{\mathbb{L}}$ are square and the matrix pencil $(\mathbf{E}_{\mathbb{L}}, \mathbf{A}_{\mathbb{L}})$ is regular, the system (3.22) has a transfer function which perfectly interpolates the frequency data partitions Λ and \mathcal{M} :

$$\mathbf{G}_{\mathbb{L}}(s) = \mathbf{C}_{\mathbb{L}}(s\mathbf{E}_{\mathbb{L}} - \mathbf{A}_{\mathbb{L}})^{-1} \mathbf{B}_{\mathbb{L}} \quad (3.23)$$

As we have seen in the proof of theorem 2.2.1, part of the reason why perfect interpolation of frequency data in data partitions \mathcal{H}_{Λ} and $\mathcal{H}_{\mathcal{M}}$ is achieved by (3.23) is because $\mathbf{C}_{\mathbb{L}}$ and $\mathbf{B}_{\mathbb{L}}$ are the literal vectors of matrices in \mathcal{H}_{Λ} and $\mathcal{H}_{\mathcal{M}}$, respectively.

Rewrite the transfer function:

$$\mathbf{G}_{\mathbb{L}}(s) = \begin{bmatrix} \mathbf{H}_{\lambda_1} & \mathbf{H}_{\lambda_2} & \cdots & \mathbf{H}_{\lambda_j} & \cdots & \mathbf{H}_{\lambda_{\bar{j}}} \end{bmatrix} \boldsymbol{\gamma}(s)^{-1} \begin{bmatrix} \mathbf{H}_{\mu_1} \\ \mathbf{H}_{\mu_2} \\ \vdots \\ \mathbf{H}_{\mu_i} \\ \vdots \\ \mathbf{H}_{\mu_{\bar{i}}} \end{bmatrix} \quad (3.24)$$

where $\boldsymbol{\gamma}(s)$ was defined as (2.59). Using the steps shown in proving theorem 2.2.1, we have that

$$\mathbf{G}_{\mathbb{L}}(\lambda_j) = \begin{bmatrix} \mathbf{H}_{\lambda_1} & \mathbf{H}_{\lambda_2} & \cdots & \mathbf{H}_{\lambda_j} & \cdots & \mathbf{H}_{\lambda_{\bar{j}}} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{0}_{\bar{m} \times \bar{m}} \\ \mathbf{0}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{I}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{0}_{\bar{m} \times \bar{m}} \end{bmatrix} \quad (3.25)$$

where the block column vector of zeros has the only identity matrix located at the j 'th block row. The solution to (3.25) is $\mathbf{G}_{\mathbb{L}}(\lambda_j) = \mathbf{H}_{\lambda_j}$, but we can more easily explore the scenario of perturbing $\mathbf{C}_{\mathbb{L}}$ using (3.25). Suppose we perturb $\mathbf{C}_{\mathbb{L}}$ by

$$d\mathbf{C}_{\mathbb{L}} = \begin{bmatrix} d\mathbf{H}_{\lambda_1} & d\mathbf{H}_{\lambda_2} & \cdots & d\mathbf{H}_{\lambda_{\bar{m}}} \end{bmatrix} \quad (3.26)$$

with the ensuing modified transfer function $\tilde{\mathbf{G}}_{\mathbb{L}}(s)$ defined as:

$$\tilde{\mathbf{G}}_{\mathbb{L}}(s) = \tilde{\mathbf{C}}_{\mathbb{L}} \gamma(s)^{-1} \mathbf{B}_{\mathbb{L}}, \quad \tilde{\mathbf{C}}_{\mathbb{L}} = \mathbf{C}_{\mathbb{L}} + d\mathbf{C}_{\mathbb{L}} \quad (3.27)$$

Updating (3.25) with the perturbation by $d\mathbf{C}_{\mathbb{L}}$, we have:

$$\tilde{\mathbf{G}}_{\mathbb{L}}(\lambda_j) = \begin{bmatrix} \mathbf{H}_{\lambda_1} + d\mathbf{H}_{\lambda_1} & \mathbf{H}_{\lambda_2} + d\mathbf{H}_{\lambda_2} & \cdots & \mathbf{H}_{\lambda_j} + d\mathbf{H}_{\lambda_j} & \cdots & \mathbf{H}_{\lambda_{\bar{m}}} + d\mathbf{H}_{\lambda_{\bar{m}}} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{0}_{\bar{m} \times \bar{m}} \\ \mathbf{0}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{I}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{0}_{\bar{m} \times \bar{m}} \end{bmatrix} \quad (3.28)$$

whose solution can be directly assessed as $\tilde{\mathbf{G}}_{\mathbb{L}}(\lambda_j) = \mathbf{H}_{\lambda_j} + d\mathbf{H}_{\lambda_j}$. As the evaluation input was an arbitrary $\lambda_j \in \Lambda$, it means the evaluation of all points in Λ by $\tilde{\mathbf{G}}_{\mathbb{L}}(s)$ leads to the result of evaluation of $\mathbf{G}_{\mathbb{L}}(s)$ directly added with the corresponding perturbation $d\mathbf{H}_{\lambda_j}$. As such, $d\mathbf{C}_{\mathbb{L}}$ is the literal perturbation error vector, so the perturbation error norm of frequency points only in Λ can be defined under F-norm as:

$$\Delta_{\Lambda} = \|d\mathbf{C}_{\mathbb{L}}\|_F^2 \quad (3.29)$$

Note that we have chosen the F-norm to maintain consistency with how we defined the error norm Δ by (2.159). Although Δ_Λ only takes into account of perturbation error in partition Λ and not the error in partition \mathcal{M} , one can design Δ_Λ to be a satisfactory approximate of Δ by making sure sufficient number of sample points are in Λ and that they are well spread out throughout the frequency domain of operation of the original system. As such, $\mathbf{C}_\mathbb{L}$ can be directly perturbed for the sake of passivity enforcement as it has built-in error control.

The problem with the above error control formulation is that the direct LM system (3.22) is almost never used as the final approximation LTI system. As was shown in section 2.2.3, a MOR is performed on the raw LM system (3.22) in order to obtain a reduced order system that is numerically stable. Recall that the MOR defined a LM pencil:

$$\boldsymbol{\gamma} = s_{ref} \cdot \mathbb{L} - \sigma \mathbb{L} \quad (3.30)$$

which underwent SVD:

$$\boldsymbol{\gamma} = \mathbf{U} \mathbf{S} \mathbf{V}^H \quad (3.31)$$

A reduced system order n is chosen and only the first n columns of $\mathbf{U} \in \mathbb{R}^{\bar{p} \times \bar{p}}$ and $\mathbf{V} \in \mathbb{R}^{\bar{m} \bar{j} \times \bar{m} \bar{j}}$ are kept resulting in the reduced singular vector matrices $\mathbf{U}_n \in \mathbb{R}^{\bar{p} \times n}$ and $\mathbf{V}_n \in \mathbb{R}^{\bar{m} \bar{j} \times n}$. The final reduced order LM system is generated following (2.84), but we now focus specifically on the $\mathbf{C}_\mathbb{L}$ modification in (2.84):

$$\mathbf{C}_n = \mathbf{C}_\mathbb{L} \mathbf{V}_n \quad (3.32)$$

It is clear that the reduced order \mathbf{C}_n defined by (3.32) no longer holds the structure of $\mathbf{C}_\mathbb{L}$, which in turn means that the reduced order system $\{\mathbf{E}_n, \mathbf{A}_n, \mathbf{B}_n, \mathbf{C}_n\}$ no

longer perfectly maintains the special built-in error control property. In order to reinstate the error control property of $\mathbf{C}_{\mathbb{L}}$, we can attempt a similar change of variable to (2.165) utilized by the gramian error control method:

$$\tilde{\mathbf{C}}_{\mathbb{L}} = \mathbf{C}_n \mathbf{V}_n^\dagger \quad (3.33)$$

where we use the Moore-Penrose pseudo-inverse to retract, as best as we can, the MOR. However, given the Moore-Penrose pseudo-inverse definition and the fact that \mathbf{V}_n is a matrix of singular vectors, we have:

$$\begin{aligned} \mathbf{V}_n^\dagger &= (\mathbf{V}_n^H \mathbf{V}_n)^{-1} \mathbf{V}_n^H \\ \mathbf{V}_n^\dagger &= (\mathbf{I}_{n \times n})^{-1} \mathbf{V}_n^H \\ \mathbf{V}_n^\dagger &= \mathbf{V}_n^H \end{aligned} \quad (3.34)$$

which means the pseudo-inverse of \mathbf{V}_n is simply the hermitian of \mathbf{V}_n . Observing the F-norm of $\tilde{\mathbf{C}}_{\mathbb{L}}$, we then have:

$$\begin{aligned} \|\tilde{\mathbf{C}}_{\mathbb{L}}\|_F^2 &= \text{tr} \left(\tilde{\mathbf{C}}_{\mathbb{L}} \cdot \tilde{\mathbf{C}}_{\mathbb{L}}^H \right), \quad \text{by property (2.163)} \\ &= \text{tr} \left(\mathbf{C}_n \mathbf{V}_n^\dagger \cdot (\mathbf{C}_n \mathbf{V}_n^\dagger)^H \right) \\ &= \text{tr} \left(\mathbf{C}_n \mathbf{V}_n^H \cdot (\mathbf{C}_n \mathbf{V}_n^H)^H \right) \\ &= \text{tr} \left(\mathbf{C}_n \mathbf{V}_n^H \cdot \mathbf{V}_n \mathbf{C}_n^H \right) \\ &= \text{tr} \left(\mathbf{C}_n \mathbf{I}_{n \times n} \mathbf{C}_n^H \right) \\ \|\tilde{\mathbf{C}}_{\mathbb{L}}\|_F^2 &= \text{tr} \left(\mathbf{C}_n \cdot \mathbf{C}_n^H \right) = \|\mathbf{C}_n\|_F^2 \end{aligned} \quad (3.35)$$

Indeed, as demonstrated by (3.35), the F-norm of the changed variable $\tilde{\mathbf{C}}_{\mathbb{L}}$ is identical to the F-norm of \mathbf{C}_n due to property (3.34). In turn, this means that \mathbf{C}_n is already the optimal order reduced version of $\mathbf{C}_{\mathbb{L}}$ at order n , and the change of variable (3.33) is unnecessary.

One way to interpret property (3.35) is that SVD model order reduction produces reduced order systems with \mathbf{C}_n with remnant error control with respect to the data in \mathcal{H}_{Λ} rather than to the data approximated by system $\{\mathbf{E}_n, \mathbf{A}_n, \mathbf{B}_n, \mathbf{C}_n\}$ at the frequency points in Λ . This remnant error control effect can still be very potent if the order n system retains enough characteristics of its original full sized system. As a matter of fact, the remnant error control effect can be so significant that it can achieve comparable or better error control during passivity enforcement when perturbing \mathbf{C}_n when compared to the gramian based methods presented in sections 2.4.4, 2.4.5, and 3.1.

3.3 Simulation

This section presents a series of examples to illustrate the passivity enforcement algorithms' capability and how they fare with respect to each other. Each example is a state-space system generated either by the SFLM method or the VF method. Each such example system is pre-designed to be stable under the BIBO definition explained in Section 2.1.3 while non-passive according to the passivity condition defined at Section 2.1.4.

Each example system undergoes passivity enforcement through a passivity enforcement algorithm, which combines a perturbation method with an error control

scheme. The participating perturbation methods are the Hamiltonian matrix eigenvalue perturbation method (Section 2.4.2) and the transfer function singular value perturbation method (Section 2.4.3) which, for the sake of convenience, shall be dubbed as M- α and M- β , respectively. The perturbation equations target only the \mathbf{C} matrix of the transfer function because the \mathbf{B} matrix perturbation may be problematic in case of special structure in VF systems. The participating error control schemes are the standard controllability gramian scheme (Section 2.4.4), the filter gramian scheme (Section 2.4.5), the discrete gramian scheme (Section 3.1), and the special LM system built-in error control scheme (Section 3.2) which, for the sake of convenience once again, shall be dubbed as Std. G., Filt. G., Disc. G., and LM E.C., respectively. Due to its formulation, the LM E.C. scheme will be only applied to LM systems, as it isn't applicable to the VF systems.

3.3.1 Perturbation Methods Strategies

Although we have defined the perturbation equations, the strategies applied when using the equations are not properly defined yet. In this sub-section, the perturbation strategies employed by both the Hamiltonian matrix eigenvalue perturbation method (Section 2.4.2) and the transfer function singular value perturbation method (Section 2.4.3) are defined.

Hamiltonian Matrix Eigenvalue Perturbation Strategy

The guiding principle behind concrete definition of the Hamiltonian matrix eigenvalue perturbation equation (2.137) is illustrated on Figure 3.2. From the figure we see how an imaginary eigenvalue pair of the Hamiltonian matrix pencil $(\mathcal{J}, \mathcal{K})$ associated to the starting and ending points of a singular value curve going above

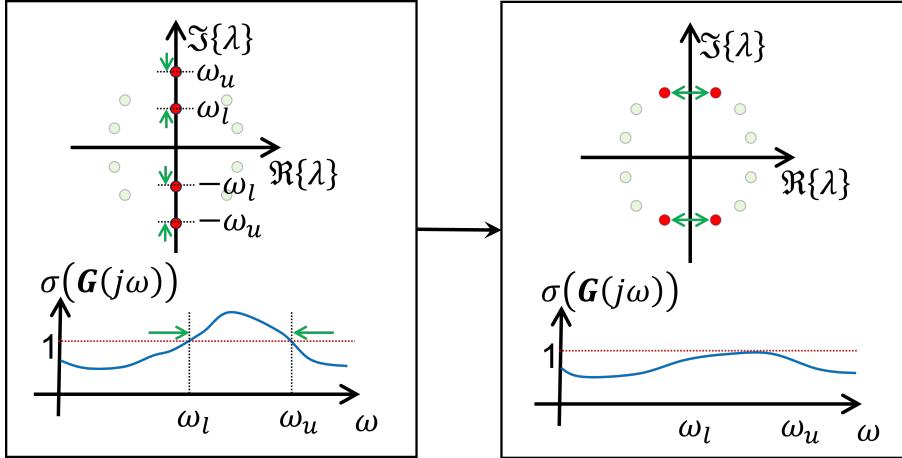


Figure 3.2: Hamiltonian matrix pencil imaginary eigenvalue perturbation process.

threshold 1 can be perturbed towards each other to merge and then split into a complex pair of eigenvalues that are no-longer purely imaginary, reflected by the fact that the associated segment of singular value curve above threshold 1 is pushed below the threshold. This dynamic strongly promotes perturbing upwards the imaginary eigenvalues associated to positive singular value curve slopes and downwards the imaginary eigenvalues associated to negative slopes. Doing so would promote merger and then split of imaginary eigenvalue value pairs unto the complex plane. Note that often more than 1 iteration of perturbation is required to fully suppress the non-passive regions.

Having determined in which direction to perturb the imaginary eigenvalues, the perturbation amount is still undetermined. Perturbation amount is highly case dependent, so we shall use a general procedure developed empirically. The procedure is illustrated on Figure 3.3. We simply scan the non-passive region from left to right and group the foremost non-grouped positive slope imaginary eigenvalue detected

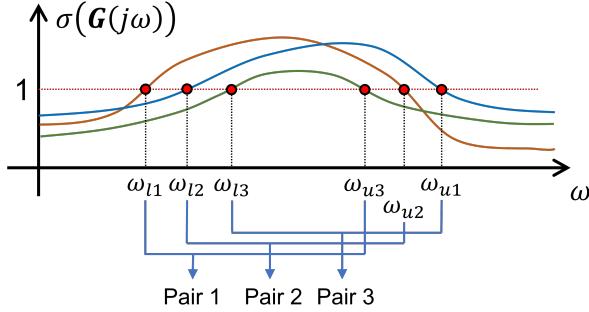


Figure 3.3: Imaginary eigenvalue grouping.

with the first non-grouped negative slope imaginary eigenvalue we come upon. The reason for this interlaced eigenvalue grouping is the fact that it is difficult to determine the exact pair of imaginary eigenvalues marking the starting and ending points of the same singular value curve going above threshold 1 due to singular value curves crossing each other in an unpredictable manner. The interlaced pairing allows a more measured perturbation magnitude that isn't too small as to slow the process too much and isn't too large as to significantly increase the chance of divergence of the process. We define the perturbation rate for $M-\alpha$ as ζ . When multiplied with the distance between the eigenvalue pair, ζ provides the amount of perturbation for the lower eigenvalue to increase by and the upper eigenvalue to decrease by.

For the special case of non-passive regions including the DC point ($s = 0$), we inevitably have unpaired imaginary eigenvalues of negative singular value slope. For those cases, one can simply perturb them towards a negative value, which is symbolically pairing them with their complex conjugate and which typically results in a merger and split into a pair of purely real eigenvalues.

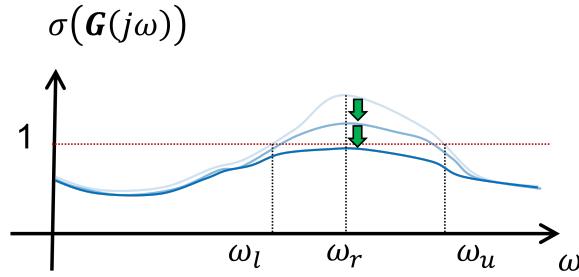


Figure 3.4: Transfer function singular value perturbation process.

Singular Value Perturbation Strategy

The guiding principle of passivity enforcement through singular value perturbation is illustrated on Figure 3.4, where a singular value curve having a segment $[\omega_l, \omega_u]$ above threshold 1 has this very segment gradually pushed below the threshold. The general idea is to simply select a ω_r point within the segment and turn the negative eigenvalues of $Z(j\omega_r)$ positive possibly through several iterations, resulting in the transformation illustrated by Figure 3.4. There is no absolute guideline regarding which and how many such sample points $j\omega_r$ per non-passive region should one take and how many negative eigenvalues to perturb by how much at each point. For the perturbation method M- β used in the simulation within the following sections, we shall adopt the simple strategy of selecting only the frequency point where the highest singular value occur within each non-passive region. At each such point $j\omega_r$, we perturb all negative eigenvalues of $Z(j\omega_r)$ as defined by (2.141) upward to become positive by a magnitude equal the perturbation factor ε multiplied by the magnitude of these negative eigenvalues.

Example 1 Schematic

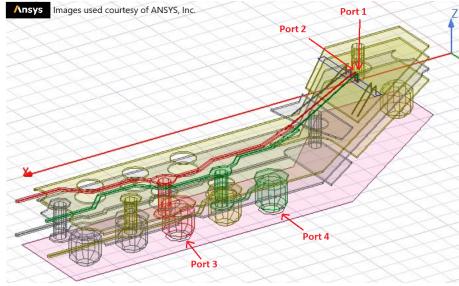


Figure 3.5: Example 1 Schematic

3.3.2 Example 1

The example is illustrated on Figure 3.5, where we have the schematic of a package section design originating from Ansys HFSS software as a template design project. The design defines 4 independent ports, with port 1 and 2 located at the top of the package defined between the copper wires and the perfect conductor plate supporting the copper wires. The two remaining ports are defined between the legs of two ball shaped vias and the floor level perfect conductor plate. The initial data consists of 199 s points linearly distributed over the range of [0.04, 8]GHz. Only 100 frequency points linearly sampled from the initial set, are used to construct the macromodel using the LM system method with a final system order of 50. The perturbation rate used for M- α is $\zeta = 0.25$ and the one used for M- β is $\varepsilon = 0.4$ and are applied to all four error control schemes. Similarly, for each individual error control scheme, the same gramian is applied for both methods M- α and M- β .

On Figure 3.6 is plotted the singular value data progression through each iteration with passive and non-passive band highlight on the 0 singular value plane. As can be observed on Figure 3.6, the initial non-passive region at the 1st iteration is

Passivity Enforcement Progress Plot

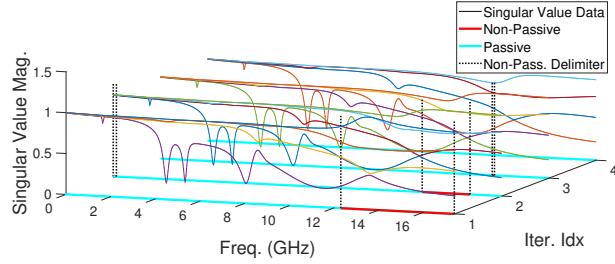


Figure 3.6: Example 1 Passivity Enforcement Progress Plot ($M-\beta$, Disc. G.)

above the 8 GHz upper bound of our initial frequency data range, which matches the intended non-passivity problem that the Filt. G. and Disc. G. error control schemes are designed to deal with. On Figure 3.7 are plotted the magnitude and phases of

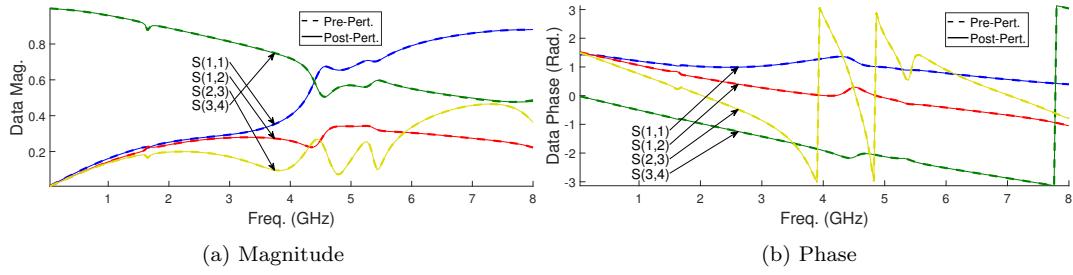


Figure 3.7: Ex. 1 before and after [$M-\beta$, Disc. G.] sample data plot

select S-parameters before and after the perturbation by method $M-\beta$ using scheme Disc. G.. We only show the before-after results of algorithm [$M-\beta$, Disc. G.] due to redundancy of showing the very graphically similar results of the other perturbation algorithms. From Figure 3.7, we see the perturbed system is very accurate. Accuracy for each algorithm can also be observed on Figure 3.8, where the RMS error at each frequency is plotted. A number of important observations can be made:

1. The Filt. G. and Disc. G. error control schemes achieved significantly better accuracy than the Std. G. scheme, demonstrating the value of frequency band

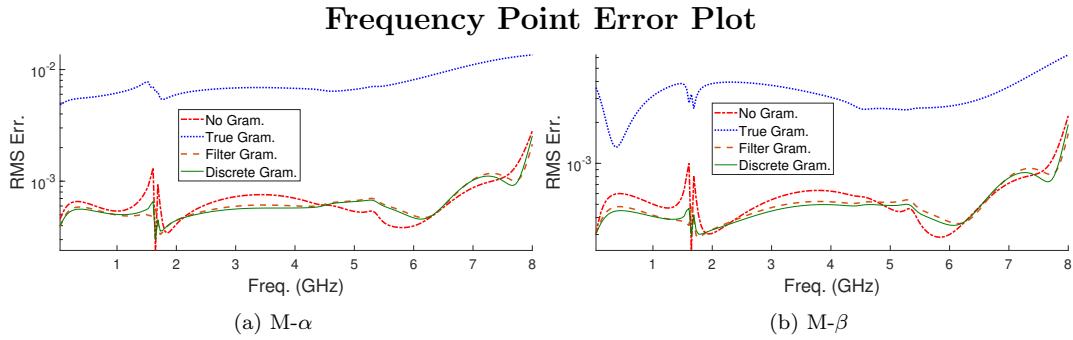


Figure 3.8: Example 1 Freq. Pt. Error Plot

targeted perturbation error control for a system which has non-passivity outside its frequency band of operation.

2. Disc. G. error control scheme achieved similar accuracy to the Filt. G. scheme, demonstrating validity of Disc. G. scheme as an alternative to the Filt. G. scheme for frequency band targeted error control.
3. The LM E.C. scheme, which is technically a perturbation scheme using no gramian, preserved the accuracy of the model better than the use of the standard gramian and the accuracy is very similar to the ones achieved by the Filt. G. and Disc. G. schemes. This result align with the theory that LM system macromodels have built-in error control similar in execution to the Disc. G. scheme.

More specific numerical results of the perturbation run are given in Table 3.1 where the iteration, computation times, and absolute RMS errors are given for all algorithms. Of note, the absolute RMS errors confirm the accuracy results shown on Figure 3.8, where we have the Filt. G., Disc. G., and LM E.C. schemes achieving significant accuracy gain over the Std. G. scheme. However, due to the low order

Table 3.1: Ex. 1 Simulation Results

		Std. G.	Filt. G.	Disc. G.	LM E.C.
Iter. Cnt	M- α	5	2	3	2
	M- β	2	2	2	2
Time (s)	M- α	0.06	0.05	0.05	0.02
	M- β	0.06	0.2	0.09	0.08
G. Time (s)	M- α	0.003	0.02	0.01	n/a
	M- β	0.01	0.08	0.02	n/a
RMS Error	M- α	7.975e-3	7.011e-4	6.994e-4	7.681e-4
	M- β	3.399e-3	5.551e-4	5.450e-4	6.221e-4

of the system, the computation times for all algorithms are too small to derive any meaningful comparison between the computation efficient of the algorithms.

3.3.3 Example 2

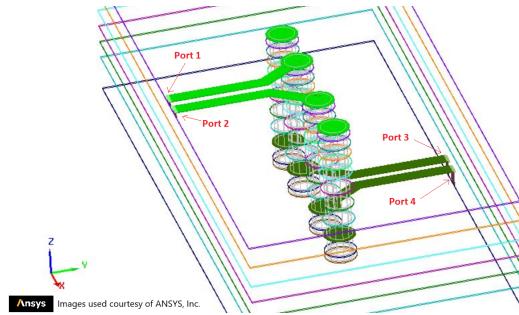
Example 2 Schematic


Figure 3.9: Example 2 Schematic

The example is illustrated on Figure 3.11 where we have the schematic of a differential via design originating from the Ansys HFSS 3D Layout software as a template design project. The via design comprises of interleaving layers of copper and FR4 epoxy substrate. Although the two pairs of ports (port1, port2) and (port3, port4) are supposed to be defined as differential pairs, the simulation was run with 4

independent ports. The initial data consists of 199 s points linearly distributed over the range of [0.05, 10]GHz. A VF model is generated with 20 initial poles, resulting in a final stable but non-passive model of order 80. The perturbation rate used for M- α is $\zeta = 0.4$ and the one used for M- β is $\varepsilon = 0.4$ and are applied to all four error control schemes. Similarly, for each individual error control scheme, the same gramian is applied for both methods M- α and M- β .

Passivity Enforcement Progress Plot

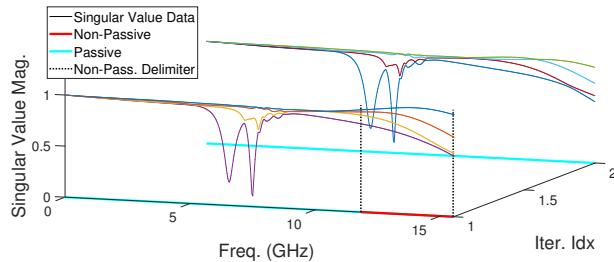


Figure 3.10: Example 2 Passivity Enforcement Progress Plot (M- β , Disc. G.)

On Figure 3.10 is plotted the singular value data progression through each iteration with passive and non-passive band highlight on the 0 singular value plane. As can be observed on Figure 3.10, the initial non-passive region at the 1st iteration is above the 10 GHz upper bound of our initial frequency data range, which matches the intended non-passivity problem that the Filt. G. and Disc. G. error control schemes are designed to deal with. On Figure 3.11 are plotted the magnitude and phases of select S-parameters before and after the perturbation by method M- β using scheme Disc. G.. We only show the before-after results of algorithm [M- β , Disc. G.] due to redundancy of showing the very graphically similar results of the other perturbation algorithms. From Figure 3.11, we see the perturbed system is very accurate. Accuracy for each algorithm can also be observed on Figure 3.12, where the

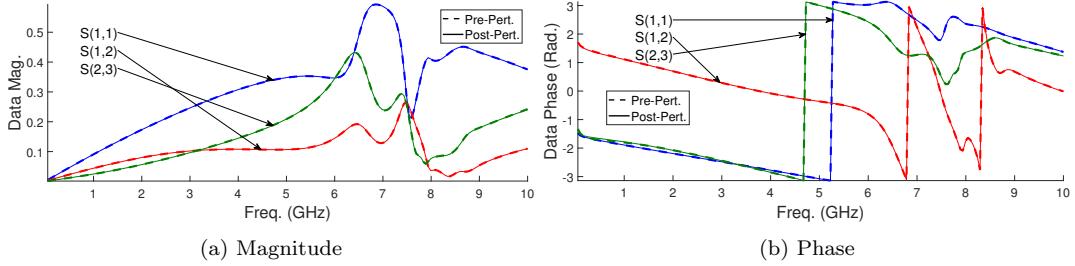
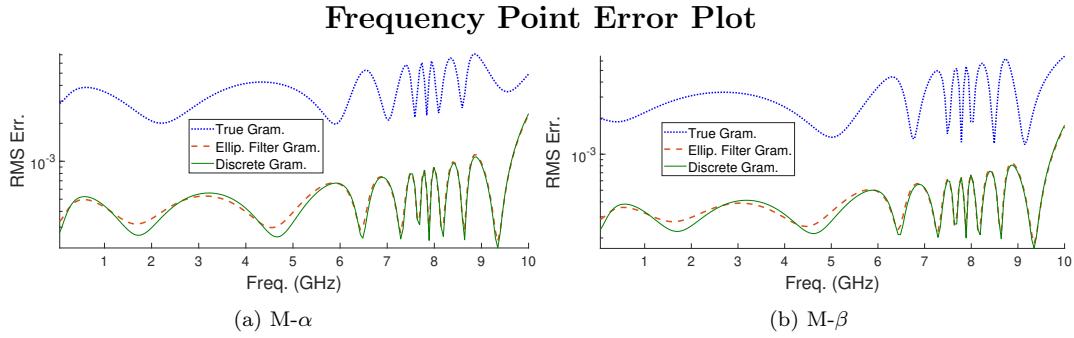

 Figure 3.11: Ex. 2 before and after $[M-\beta, \text{Disc. G.}]$ sample data plot


Figure 3.12: Example 2 Freq. Pt. Error Plot

RMS error at each frequency is plotted. We observe the similar result to example 1 with Filt. G. and Disc. G. error control schemes obtaining better accuracy than the Std. G. scheme while having similar accuracy with respect to each other. This demonstrate once again that Disc. G. is a valid alternative to the Filt. G. scheme in enacting frequency band targeted error control even when we apply the scheme on a VF model rather than a LM model. Unfortunately, we cannot use the LM E.C. scheme because this example is not a LM system model.

More specific numerical results of the perturbation runs are given on Table 3.2 where the iteration, computation times, and absolute RMS errors are given for all algorithms. Of note, the absolute RMS errors confirm the accuracy results shown on Figure 3.12, where we have the Filt. G. and Disc. G. schemes achieving significant

Table 3.2: Ex. 2 Simulation Results

		Std. G.	Filt. G.	Disc. G.	LM E.C.
Iter. Cnt	M- α	2	1	1	n/a
	M- β	2	1	1	n/a
Time (s)	M- α	0.09	0.06	0.07	n/a
	M- β	0.1	0.06	0.07	n/a
G. Time (s)	M- α	0.009	0.02	0.01	n/a
	M- β	0.006	0.01	0.01	n/a
RMS Error	M- α	3.842e-3	6.279e-4	6.233e-4	n/a
	M- β	3.211e-3	4.714e-4	4.673e-4	n/a

accuracy gain over the Std. G. scheme. However, due to the low order of the system, the computation times for all algorithms are too small to derive any meaningful comparison between the computation efficient of the algorithms. As well, example 2 is a VF model which means there is no data for LM E.C. scheme on Table 3.2 due to incompatibility of the scheme.

3.3.4 Example 3

Example 3 Schematic

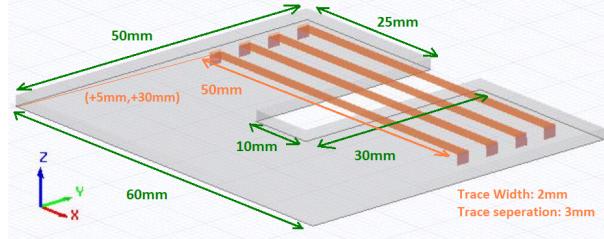


Figure 3.13: Example 3 Schematic

The system is illustrated on Figure 3.13 where we have a simple microstripline model with a gap in the substrate and ground layers. The three layers are the copper ground of 0mm thickness, the FR4-epoxy substrate ($\epsilon_r = 4.4$) of 1.6mm thickness,

and the copper trace layer of 0mm thickness. A total of 8 edge ports of 50Ω are defined on the edge of the traces. The S-parameter data is generated using HFSS simulation on ANSYS Electronics Desktop 2020 R1. The initial data set consists of 499 frequency points linearly distributed over the frequency range from [0, 10]GHz. Only 100 frequency points, linearly sampled from the initial set, are used to construct the macromodel using the LM system method with a final system order of 200. The perturbation rate used for $M-\alpha$ is $\zeta = 0.25$ and the one used for $M-\beta$ is $\varepsilon = 0.4$ and are applied to all four error control schemes. Similarly, for each individual error control scheme, the same gramian is applied for both methods $M-\alpha$ and $M-\beta$.

Passivity Enforcement Progress Plot

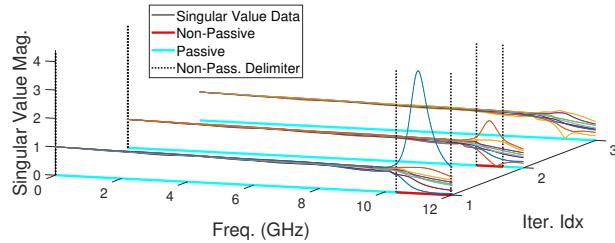
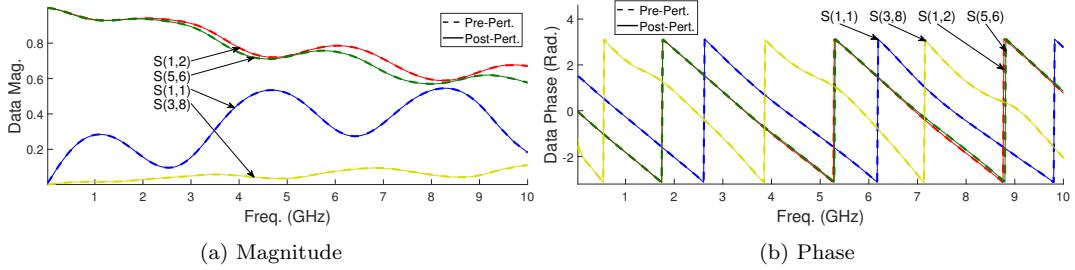


Figure 3.14: Example 3 Passivity Enforcement Progress Plot (M- β , Disc. G.)

On Figure 3.14 is plotted the singular value data progression through each iteration with passive and non-passive band highlight on the 0 singular value plane. As can be observed on Figure 3.14, the initial non-passive region at the 1st iteration is above the 10 GHz upper bound of our initial frequency data range, which matches the intended non-passivity problem that the Filt. G. and Disc. G. error control schemes are designed to deal with. On Figure 3.15 are plotted the magnitude and phases of select S-parameters before and after the perturbation by method M- β using scheme Disc. G.. We only show the before-after results of algorithm [M- β , Disc. G.] due to


 Figure 3.15: Ex. 3 before and after $[M-\beta, \text{Disc. G.}]$ sample data plot

redundancy of showing the very graphically similar results of the other perturbation algorithms. From Figure 3.15, we see the perturbed system is very accurate. Accu-

Frequency Point Error Plot

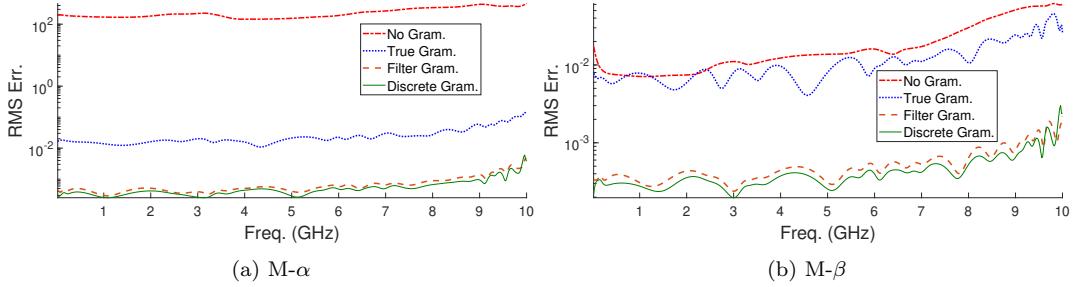


Figure 3.16: Example 3 Freq. Pt. Error Plot

racy for each algorithm can also be observed on Figure 3.16, where the RMS error at each frequency is plotted. For the Filt. G. and Disc. G. error control schemes, we observe similar results to the previous examples where we have better accuracy than the Std. G. scheme while having similar accuracy with respect to each other. However, the LM E.C. scheme is performing poorly on this LM system by having the worse overall error, especially with perturbation method $M-\alpha$ where the high error is due to the perturbation process timing out without reaching convergence.

Table 3.3: Ex. 3 Simulation Results

		Std. G.	Filt. G.	Disc. G.	LM E.C.
Iter. Cnt	M- α	13	5	6	30 (Max.)
	M- β	8	2	2	10
Time (s)	M- α	2.6	1.1	1.3	13.4
	M- β	2.5	0.7	0.6	4.0
G. Time (s)	M- α	0.05	0.03	0.05	n/a
	M- β	0.07	0.1	0.05	n/a
RMS Error	M- α	3.396e-2	1.018e-3	9.022e-4	253.6
	M- β	1.416e-2	6.242e-4	5.540e-4	2.542e-2

More specific numerical results of the perturbation runs are given on Table 3.3 where the iteration, computation times, and absolute RMS errors are given for all algorithms. As expected, the absolute RMS errors confirm the accuracy results shown on Figure 3.16, where we have the Filt. G. and Disc. G. schemes achieving significant accuracy gain over the Std. G. scheme. However, we also see that the LM E.C. scheme used relatively high number of iterations with M- β while reaching the maximum number of iterations with M- α , which lead to the conclusion that the LM E.C. scheme is more volatile in terms of error control and may falter when facing more severe non-passivity violations that require more substantial perturbation.

3.3.5 Example 4

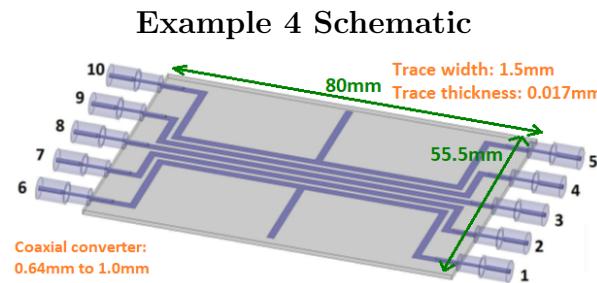


Figure 3.17: Example 4 Schematic

The model is illustrated on Figure 3.17 where we have a stripline structure terminated with a total of 10 coaxial terminations. The conductors are made of copper and the substrate chosen is Rogers TMM 10 ($\epsilon_r = 9.2$). All traces are terminated by coaxial converters with the 0.64 mm end in contact with the trace and the 1.00 mm end serving as the port terminals. Each coaxial converter has 1 mm length of transition region between the two wires of different diameters. The S-parameter data is generated using HFSS simulation on ANSYS HFSS 2014. The initial data set consists of 200 frequency points distributed in the range of [1 MHz, 10 GHz]. Only 100 frequency points, linearly sampled from the initial set, are used to construct the macromodel using the LM system method with a final system order of 300.

Passivity Enforcement Progress Plot

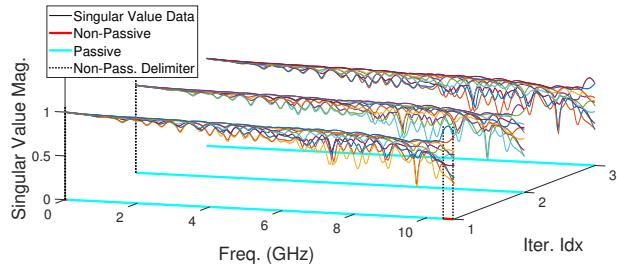
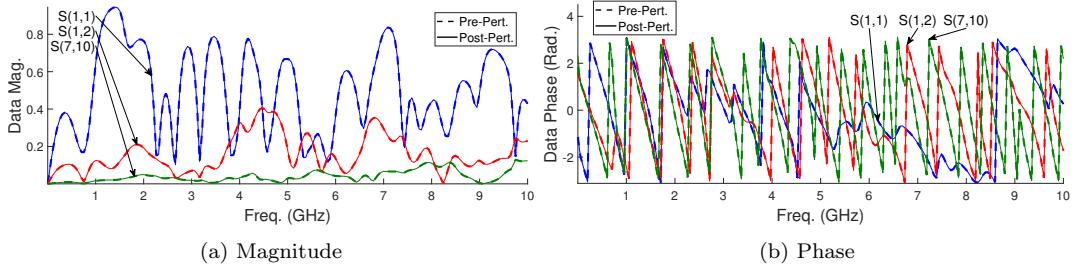


Figure 3.18: Example 4 Passivity Enforcement Progress Plot (M- β , Disc. G.)

On Figure 3.18 is plotted the singular value data progression through each iteration with passive and non-passive band highlight on the 0 singular value plane. As can be observed on Figure 3.18, the initial non-passive region at the 1st iteration is above the 10 GHz upper bound of our initial frequency data range, which matches the intended non-passivity problem that the Filt. G. and Disc. G. error control schemes are designed to deal with. On Figure 3.19 are plotted the magnitude


 Figure 3.19: Ex. 4 before and after [M- β , Disc. G.] sample data plot

and phases of select S-parameters before and after the perturbation by method M- β using scheme Disc. G.. We only show the before-after results of algorithm [M- β , Disc. G.] due to redundancy of showing the very graphically similar results of the other perturbation algorithms. From Figure 3.19, we see the perturbed system is very accurate. Accuracy for each algorithm can also be observed on Figure 3.20,

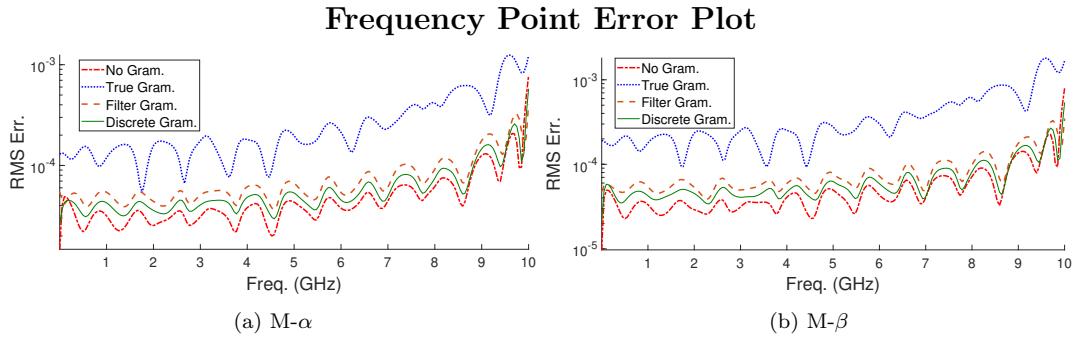


Figure 3.20: Example 4 Freq. Pt. Error Plot

where the RMS error at each frequency is plotted. For the Filt. G., Disc. G., and LM E.C. error control schemes, we observe similar results to example 1 where we have better accuracy than the Std. G. scheme while having similar accuracy with respect to each other, demonstrating the advantage of frequency band targeted error control. Interestingly, LM E.C. managed to achieve the greatest accuracy out of all

Table 3.4: Ex. 4 Simulation Results

		Std. G.	Filt. G.	Disc. G.	LM E.C.
Iter. Cnt	M- α	1	1	2	2
	M- β	1	1	1	1
Time (s)	M- α	1.0	1.0	1.4	1.3
	M- β	0.9	1.1	0.9	1.1
G. Time (s)	M- α	0.2	0.3	0.1	n/a
	M- β	0.1	0.2	0.1	n/a
RMS Error	M- α	3.775e-4	9.629e-5	8.557e-5	8.847e-5
	M- β	5.365e-4	9.994e-5	8.807e-5	9.269e-5

error control schemes, in complete contrast to example 3 where the LM E.C. scheme can cause the enforcement to fail. One should thus be wary of the LM E.C. scheme due to its efficiency being heavily dependent on the passivity enforcement problem being tackled.

More specific numerical results of the perturbation run are given on Table 3.4 where the iteration, computation times, and absolute RMS errors are given for all algorithms. Of note, the absolute RMS errors confirm the accuracy results shown on Figure 3.20, where we have the Filt. G., Disc. G., and LM E.C. schemes achieving significant accuracy gain over the Std. G. scheme. However, due to the low order of the system, the computation times for all algorithms are too small to derive any meaningful comparison between the computation efficient of the algorithms.

3.3.6 Example 5

The model is illustrated on Figure 3.21 where we have a short section of a stripline structure with alternating pattern of two types of group of striplines A and B. The detailed specifications of the design on given on Figure 3.21 itself, and edge ports of 50Ω are defined at the terminating edges of all traces to a total of 56 ports.

Example 5 Schematic

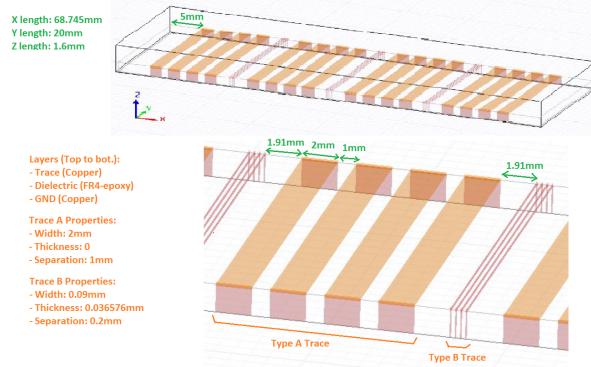


Figure 3.21: Example 5 Schematic

The S-parameter data is generated using HFSS simulation on ANSYS Electronics Desktop 2020 R1. The initial data set consists of 599 frequency points linearly distributed over the frequency range from [0.02, 12]GHz. A VF model is generated with 22 initial poles, resulting in a final stable but non-passive model of order 1232. The perturbation rate used for M- α is $\zeta = 0.25$ and the one used for M- β is $\varepsilon = 0.4$ and are applied to all four error control schemes. Similarly, for each individual error control scheme, the same gramian is applied for both methods M- α and M- β .

Passivity Enforcement Progress Plot

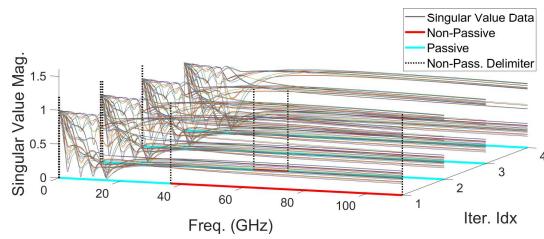


Figure 3.22: Example 5 Passivity Enforcement Progress Plot (M- β , Disc. G.)

On Figure 3.22 is plotted the singular value data progression through each iteration with passive and non-passive band highlight on the 0 singular value plane. As

can be observed on Figure 3.22, the initial non-passive region at the 1st iteration is above the 12 GHz upper bound of our initial frequency data range, which matches the intended non-passivity problem that the Filt. G. and Disc. G. error control schemes are designed to deal with. Of note, the initial non-passive region is immense compared to the previous example and is a multitude times larger than the frequency band of operation. On Figure 3.23 are plotted the magnitude and phases of select

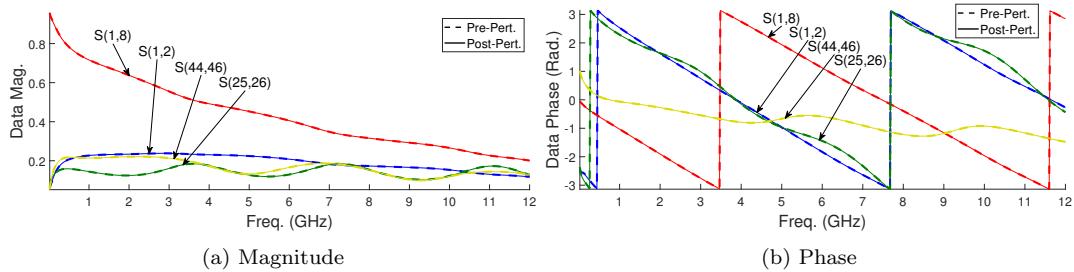


Figure 3.23: Ex. 5 before and after [M- β , Disc. G.] sample data plot

S-parameters before and after the perturbation by method M- β using scheme Disc. G.. We only show the before-after results of algorithm [M- β , Disc. G.] due to redundancy of showing the very graphically similar results of the other perturbation algorithms. From Figure 3.23, we see the perturbed system is very accurate. Accu-

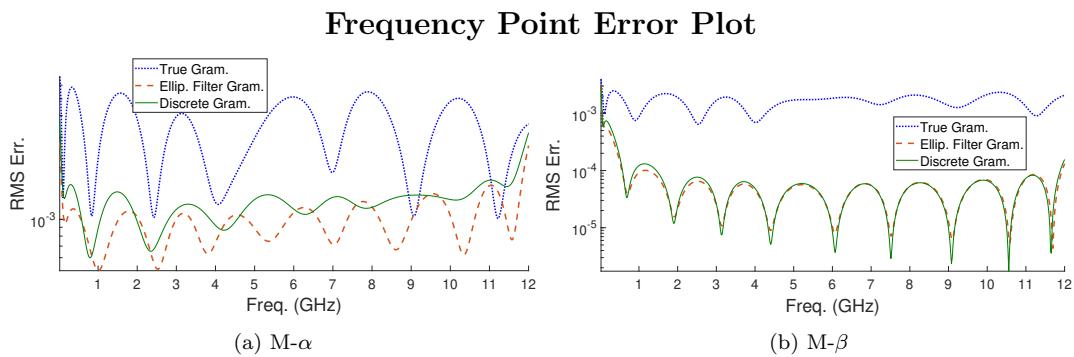


Figure 3.24: Example 5 Freq. Pt. Error Plot

Table 3.5: Ex. 5 Simulation Results

		Std. G.	Filt. G.	Disc. G.	LM E.C.
Iter. Cnt	M- α	22	30 (max)	30 (max)	n/a
	M- β	5	4	3	n/a
Time (s)	M- α	450.9	599.7	576.8	n/a
	M- β	131.0	124.0	83.3	n/a
G. Time (s)	M- α	15.5	27.3	4.8	n/a
	M- β	12.9	27.3	4.7	n/a
RMS Error	M- α	3.619e-3	1.034e-3	1.313e-3	n/a
	M- β	1.680e-3	1.705e-4	1.755e-4	n/a

racy for each algorithm can also be observed on Figure 3.24, where the RMS error at each frequency is plotted. For the Filt. G. and Disc. G. error control schemes, we once again observe better accuracy than the Std. G. scheme while having similar accuracy with respect to each other, demonstrating the advantage of frequency band targeted error control.

More specific numerical results of the perturbation runs are given on Table 3.5 where the iteration, computation times, and absolute RMS errors are given for all algorithms. The reoccurring result is that the absolute RMS errors confirm the accuracy results shown on Figure 3.12, where we have the Filt. G. and Disc. G. schemes achieving significant accuracy gain over the Std. G. scheme. However, different from the previous examples is that the Filt. G. and Disc. G. error control schemes both failed to converge when used by the M- α perturbation method and the Std. G. took a relatively large number of iteration for convergence for M- α compared to M- β . One of the possible reason for perturbation method M- α 's difficulty is revealed on Figure 3.22, where we see the initial non-passive region is clearly immense compared

to the initial frequency band of data. M- α 's perturbation aims to push the boundaries of the non-passive region towards each other whereas M- β 's perturbation aims to push the highest singular value peaks of the region downward, making the former method's task much more laborious. Another important result from Table 3.5 is the difference in computation times that are now notable given the much larger system order of the example compared to the previous ones. The gramian computation time of Disc. G. is significantly lower than that of the Std. G. and especially the Filt. G. schemes. Indeed, the larger order of the current example exposes the better scalability of Disc. G. error control scheme which can achieve similar results to Filt. G. at a lesser computation cost.

3.3.7 Example 6

Example 6 Schematic

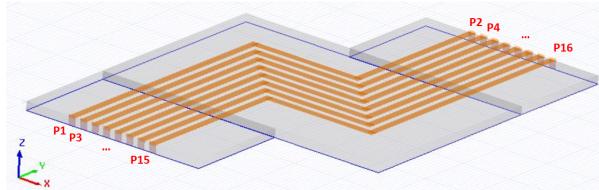


Figure 3.25: Example 6 Schematic

The model is illustrated on Figure 3.25, where we have a section of microstriplines. The FR4-epoxy substrate has thickness of 1.6mm, with flat ground plane made with a perfect conductor. The traces at the top of the substrate are made of copper and have 1.5 mm width, 20 μm thickness, and 1 mm gap in-between them. Two 90 degrees turns are located at the mid-section of the traces to provide some variations of the data. A total of 16 edge ports of 50Ω are defined on the edge of the traces. The S-parameter data is generated using HFSS simulation on ANSYS Electronics Desktop

2021 R1. The initial data set consists of 500 frequency points linearly distributed over the frequency range from [0, 10]GHz. A stable but non-passive 90 poles VF model is generated with an RMS error of 8.936e-5 with respect to the HFSS data, with an order of 1440. On Figure 3.26 is plotted the singular value data progression

Passivity Enforcement Progress Plot

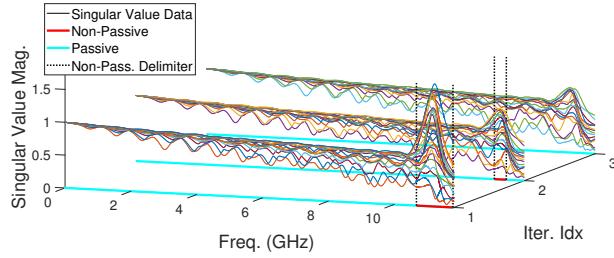


Figure 3.26: Example 6 Passivity Enforcement Progress Plot (M- β , Disc. G.)

through each iteration with passive and non-passive band highlight on the 0 singular value plane. As can be observed on Figure 3.26, the initial non-passive region at the 1st iteration is above the 10 GHz upper bound of our initial frequency data range, which matches the intended non-passivity problem that the Filt. G. and Disc. G. error control schemes are designed to deal with.

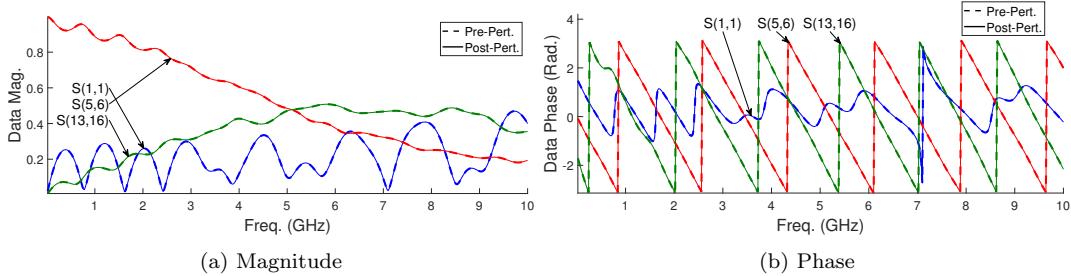


Figure 3.27: Ex. 6 before and after [M- β , Disc. G.] sample data plot

On Figure 3.27 are plotted the magnitude and phases of select S-parameters before and after the perturbation by method M- β using scheme Disc. G.. We only

Table 3.6: Ex. 6 Simulation Results

		Std. G.	Filt. G.	Disc. G.	LM E.C.
Iter. Cnt	M- α	6	3	3	n/a
	M- β	3	2	2	n/a
Time (s)	M- α	243.8	154.0	135.3	n/a
	M- β	149.8	122.6	102.4	n/a
G. Time (s)	M- α	21.5	25.7	5.7	n/a
	M- β	22.0	26.1	5.8	n/a
RMS Error	M- α	3.588e-3	1.226e-4	3.169e-5	n/a
	M- β	5.136e-3	9.369e-5	2.575e-5	n/a

show the before-after results of algorithm [M- β , Disc. G.] due to redundancy of showing the very graphically similar results of the other perturbation algorithms. From Figure 3.27, we see the perturbed system is very accurate. Accuracy for each

Frequency Point Error Plot

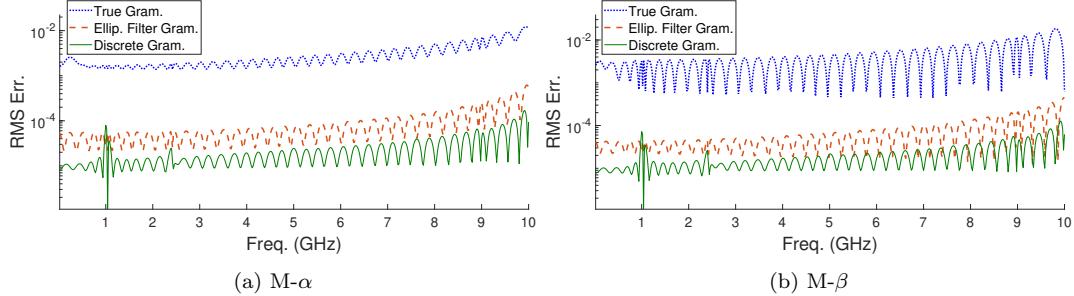


Figure 3.28: Example 6 Freq. Pt. Error Plot

algorithm can also be observed on Figure 3.28, where the RMS error at each frequency is plotted. For the Filt. G. and Disc. G. error control schemes, we once again observe better accuracy than the Std. G. scheme while having similar accuracy with respect to each other, demonstrating the advantage of frequency band targeted error control.

More specific numerical results of the perturbation runs are given on Table 3.6 where the iteration, computation times, and absolute RMS errors are given for all

algorithms. As explained in previous examples, the current example is a VF model which means there is no data for LM E.C. scheme on Table 3.6 due to incompatibility of the scheme. Once again, the absolute RMS errors confirm the accuracy results shown on Figure 3.28, where we have the Filt. G. and Disc. G. schemes achieving significant accuracy gain over the Std. G. scheme. Given the large order of the current example, we observe once again from Table 3.6 the difference in computation times between the different schemes. The gramian computation time of Disc. G. is significantly lower than that of the Std. G. and especially the Filt. G. schemes. Indeed, the larger order of the current example exposes the better scalability of Disc. G. error control scheme which can achieve similar results to Filt. G. at a lesser computation cost.

3.3.8 Example 7

Example 7 Schematic

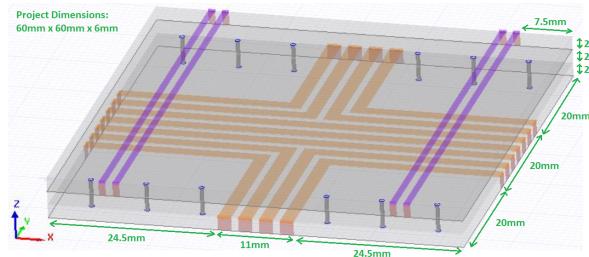


Figure 3.29: Example 7 Schematic

The system is illustrated on 3.29 where we have a $60\text{ mm} \times 60\text{ mm} \times 6\text{ mm}$ multilayer microstripline system with vias connecting the two reference layers. From bottom to top, we have the copper reference plane A, the copper trace plane A, the copper reference plane B, and the copper trace plane B, where each plane has

0 thickness and is separated from adjacent planes by 2 mm thickness layer of FR4-epoxy substrate with modified relative permittivity of 10 F/m. The trace plane A and B contain traces with widths of 2 mm and 1 mm, respectively, and both with trace separation of 1 mm. Edge ports with 50Ω impedance are placed at all trace edges with a total of 30 ports. The S-parameter data is generated using HFSS simulation on ANSYS Electronics Desktop 2020 R1. The initial data set consists of 200 frequency points linearly distributed over the frequency range from [0, 8]GHz. A stable but non-passive 60 poles VF model is generated with an RMS error of 1.167e-3 with respect to the HFSS data, with an order of 1800. The perturbation rate used for $M-\alpha$ is $\zeta = 0.25$ and the one used for $M-\beta$ is $\varepsilon = 0.4$ and are applied to all three error control schemes. Similarly, the same gramian is used for both method $M-\alpha$ and $M-\beta$ for each of the error control schemes.

Passivity Enforcement Progress Plot

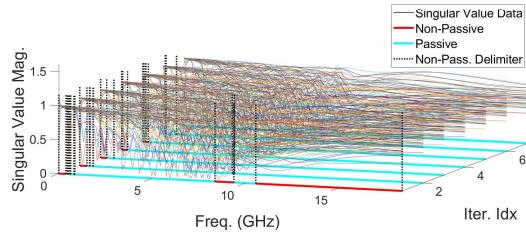


Figure 3.30: Example 7 Passivity Enforcement Progress Plot (M- β , Disc. G.)

On Figure 3.30 is plotted the singular value data progression through each iteration with passive and non-passive band highlight on the 0 singular value plane. As can be observed on Figure 3.30, the initial non-passive region at the 1st iteration is above the 8 GHz upper bound of our initial frequency data range, which matches the intended non-passivity problem that the Filt. G. and Disc. G. error control schemes

are designed to deal with. On Figure 3.31 are plotted the magnitude and phases of

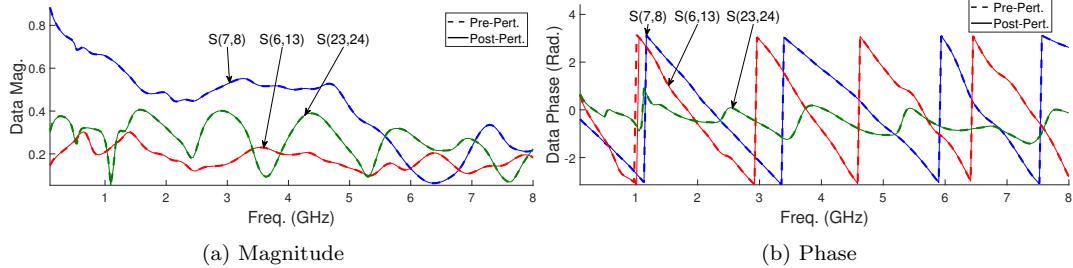


Figure 3.31: Ex. 7 before and after [M- β , Disc. G.] sample data plot

select S-parameters before and after the perturbation by method M- β using scheme Disc. G.. We only show the before-after results of algorithm [M- β , Disc. G.] due to redundancy of showing the very graphically similar results of the other perturbation algorithms. From Figure 3.31, we see the perturbed system is very accurate. Accu-

Frequency Point Error Plot

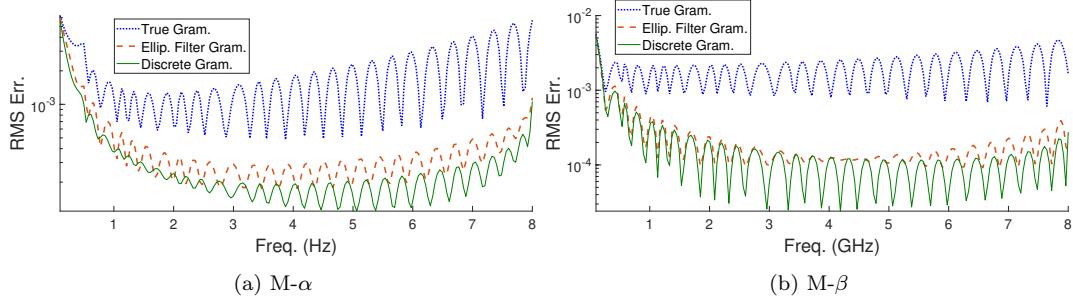


Figure 3.32: Example 7 Freq. Pt. Error Plot

racy for each algorithm can also be observed on Figure 3.32, where the RMS error at each frequency is plotted. For the Filt. G. and Disc. G. error control schemes, we once again observe better accuracy than the Std. G. scheme while having similar accuracy with respect to each other, demonstrating the advantage of frequency band targeted error control.

Table 3.7: Ex. 7 Simulation Results

		Std. G.	Filt. G.	Disc. G.	LM E.C.
Iter. Cnt	M- α	9	10	12	n/a
	M- β	7	6	6	n/a
Time (s)	M- α	609.2	671.8	733.9	n/a
	M- β	508.9	461.8	415.4	n/a
G. Time (s)	M- α	42.7	58.3	10.7	n/a
	M- β	42.6	58.4	10.7	n/a
RMS Error	M- α	2.177e-3	8.310e-4	6.956e-4	n/a
	M- β	2.185e-3	5.450e-4	5.313e-4	n/a

More specific numerical results of the perturbation runs are given on Table 3.7 where the iteration, computation times, and absolute RMS errors are given for all algorithms. As explained in previous examples, the current example is a VF model which means there is no data for LM E.C. scheme on Table 3.7 due to incompatibility of the scheme. Once again, the absolute RMS errors confirm the accuracy results shown on Figure 3.32, where we have the Filt. G. and Disc. G. schemes achieving significant accuracy gain over the Std. G. scheme. Given the large order of the current example, we observe once again from Table 3.7 the difference in computation times between the different schemes. The gramian computation time of Disc. G. is significantly lower than that of the Std. G. and especially the Filt. G. schemes. Indeed, the larger order of the current example exposes the better scalability of Disc. G. error control scheme which can achieve similar results to Filt. G. at a lesser computation cost.

3.3.9 Example 8

The model's cross-section is described by Figure 3.33 where we have simply lined up 32 coupled-traces in parallel, which results in a total of 128 ports. The data is

Example 8 Schematic

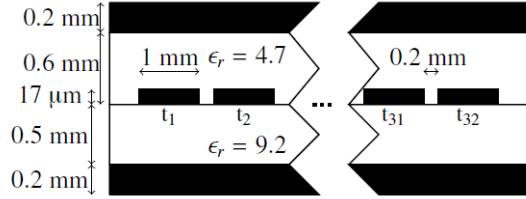


Figure 3.33: Example 8 Schematic

generated by first obtaining the per unit length parameters of the transmission lines followed by generation of Y-parameter data using the matrix exponential method [9]. Y-parameters from the range of [1 MHz, 10 GHz] are obtained and translated to S-parameters with reference port impedance of 50Ω . The S-parameter data is used to generate a LM system model of order 1900. The perturbation rate used for M- α is $\zeta = 0.6$ and the one used for M- β is $\varepsilon = 1.0$ and are applied to all four error control schemes. Similarly, the same gramian is used for both method M- α and M- β for each of the error control schemes.

Passivity Enforcement Progress Plot

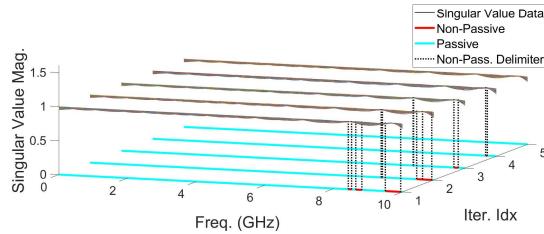


Figure 3.34: Example 8 Passivity Enforcement Progress Plot (M- β , Disc. G.)

On Figure 3.34 is plotted the singular value data progression through each iteration with passive and non-passive band highlight on the 0 singular value plane. As can be observed on Figure 3.34, the initial non-passive regions at the 1st iteration

is below the 10 GHz upper bound of our initial frequency data range, which is a breakaway characteristic with respect to the previous examples. Although this non-passivity pattern may seem like a mismatch with the intended non-passivity problem that the Filt. G. and Disc. G. error control schemes are designed to deal with, the non-passivities are close enough to the upper bound to have an impact on the outcome of the different error control schemes. On Figure 3.35 are plotted the magnitude

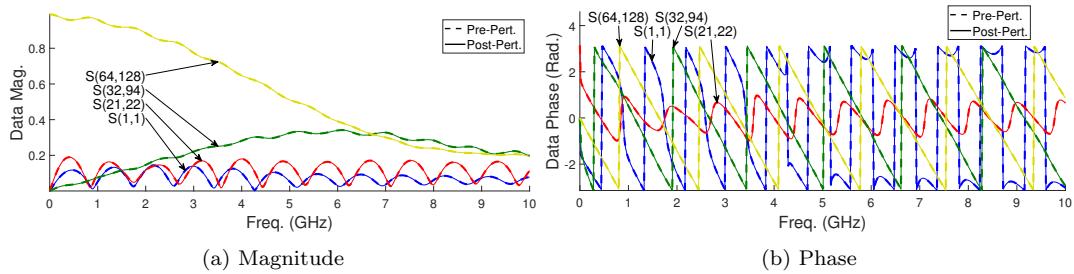


Figure 3.35: Ex. 8 before and after [M- β , Disc. G.] sample data plot

and phases of select S-parameters before and after the perturbation by method M- β using scheme Disc. G.. We only show the before-after results of algorithm [M- β , Disc. G.] due to redundancy of showing the very graphically similar results of the other perturbation algorithms. From Figure 3.35, we see the perturbed system is very accurate. Accuracy for each algorithm can also be observed on Figure 3.36, where the RMS error at each frequency is plotted. For the Filt. G., Disc. G., and LM E.C. error control schemes, we once again observe better accuracy than the Std. G. scheme while having similar accuracy with respect to each other, demonstrating the advantage of frequency band targeted error control. Interestingly, the accuracy improvements for the Filt. G., Disc. G. schemes are not as visually significant than

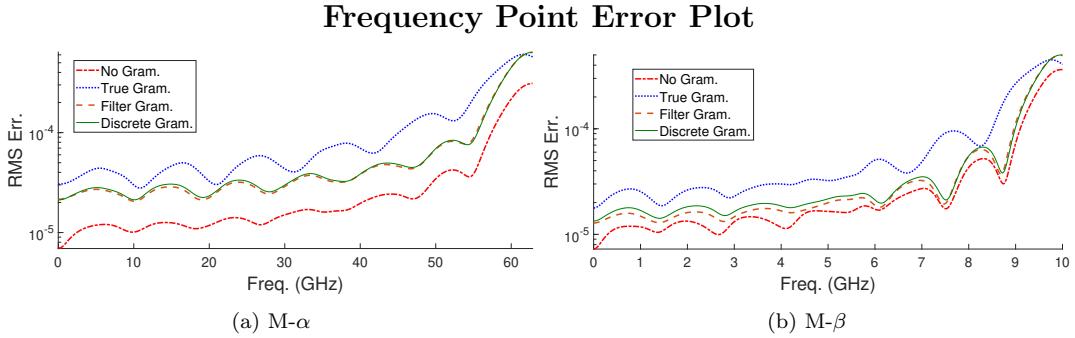


Figure 3.36: Example 8 Freq. Pt. Error Plot

Table 3.8: Ex. 8 Simulation Results

		Std. G.	Filt. G.	Disc. G.	LM E.C.
Iter. Cnt	M- α	4	2	2	10
	M- β	5	4	4	5
Time (s)	M- α	410.1	414.7	235.0	865.4
	M- β	524.6	594.3	410.6	501.7
G. Time (s)	M- α	49.9	198.2	19.0	n/a
	M- β	51.6	207.1	20.1	n/a
RMS Error	M- α	1.823e-4	1.477e-4	1.476e-4	7.150e-5
	M- β	1.370e-4	1.152e-4	1.150e-4	8.354e-5

the previous examples, which can be attributed to the fact that the non-passive regions are within the initial frequency data range. Furthermore, LM E.C. managed to achieve the greatest accuracy by a large margin, in complete contrast to example 3 where the LM E.C. scheme can cause the enforcement to fail. One can deduce that the LM E.C. scheme has high potential in error control, but is still somewhat volatile.

More specific numerical results of the perturbation run are given on Table 3.8 where the iteration, computation times, and absolute RMS errors are given for all algorithms. Of note, the absolute RMS errors confirm the accuracy results shown on

Figure 3.36, where we have the Filt. G., Disc. G., and LM E.C. schemes achieving significant accuracy gain over the Std. G. scheme. Given the large order of the current example, we observe once again from Table 3.8 the difference in computation times between the different schemes. The gramian computation time of Disc. G. is significantly lower than that of the Std. G. and especially the Filt. G. schemes. For this example in particular, the gramian computation time of Filt. G. is especially high due to the large number of ports of the example which significantly increases the order of the filtered system. Indeed, the larger order and number of ports of the current example exposes the better scalability of Disc. G. error control scheme which can achieve similar results to Filt. G. at a lesser computation cost.

3.3.10 Overall Results Analysis

From the series of example simulations presented over this section, several conclusions can be made. We restate the proposed objects by this thesis, with regard to passivity enforcement of LTI system in general, are the error control schemes of discrete gramian error control (Disc. G.) and the Loewner Matrix system method built-in error control (LM E.C.). As such, analysis and discussion focus on how well the two proposed schemes fair and how they fair compared to the standard gramian error control scheme (Std. G.) and the filtered gramian control scheme (Filt. G.).

1. The Disc. G. and LM E.C. schemes are generally functional error control schemes that worked with all examples when applied with perturbation method $M-\beta$. The two schemes also worked with perturbation method $M-\alpha$ for the majority of examples, but had convergence complications with examples 3 and 5. The overall assessment is that the proposed error control schemes are slightly

less stable than the standard gramian, but usually convergent if the right perturbation method is used.

2. Examples 5 to 8, which had relatively large system orders, demonstrated the cost of computing the gramian is significantly reduced for the Disc. G. scheme compared to the Std. G. and Filt. G. schemes, especially the latter one. Of course the LM E.C. scheme is the most computational efficient scheme because it skips the gramian computation entirely, but the overall assessment is that the Disc. G. scheme has a much better cost scaling than the Std. G. and Filt. G. schemes.
3. The LM E.C. scheme achieved similar accuracy to the Disc. G. and Filt. G. schemes for example 1 and overall best accuracy in examples 4 and 8. However, although it reached convergence when used with perturbation method M- α in example 3, it had the worse accuracy overall. The overall assessment is that the LM E.C. scheme has strong potential as frequency band targeted error control, but has a non-negligible instability that can become problematic depending on the example.
4. Whenever the Disc. G. scheme converged, it achieved similar if not better accuracy than all other schemes. If we further consider the fact that the LM E.C. scheme cannot be used outside of LM systems, we have Disc. G. as the most broadly applicable targeted frequency band error control.

The overall conclusion regarding the effectiveness of the Disc. G. and LM E.C. schemes is that both schemes can serve as alternative to the Filt. G. in targeted frequency band error control with similar error control capacity but with significantly

lower computation cost. Between the two schemes, LM E.C. is somewhat limited due to being only applicable to LM systems and has shown convergence issues that can occur depending on the system being perturbed, which makes the Disc. G. scheme to be the more generally reliable one. We should further note that the Disc. G. scheme's characteristic of selectable frequency sample points in its error control scheme lend itself readily for more customized error control with perhaps even weighted frequency points, but that is a point not dealt with in this thesis and is left for further exploration.

Chapter 4

Polynomial Parametric Barycentric Interpolation Method

In this chapter, we look at the Loewner Matrix (LM) parametric method which expands the univariate BRLM interpolation method presented in section 2.2.2 by embedding a parameter dependence in the coefficients through a polynomial expansion. The key feature that differentiates this parametric LM method from existing multi-variate LM ones such as those in [112, 114] is the use of polynomials to deal with parametric variations. This method avoids the significant increase in computation cost of solving multi-variate LMs while efficiently covering parametric variations with a relatively small number of polynomials. The present parametric method also alleviates the problem of hard-set interpolation of the primary frequency partition of the LM-based rational function interpolation presented in section 2.2.2. Though perfect interpolation of a set of data points sounds appealing, this property tends to promote unstable poles in the system.

Throughout this chapter, we will define the partitioning of the parametric data, present the set of polynomials that is going to represent the parametric dependence, and then show the combination of the polynomial set with the BRLM equation to produce the final parametric LM method. Due to this combination, we dub the method as the polynomial barycentric rational Loewner matrix (PBRLM) interpolation method.

4.1 Formulation Set-Up

In this section, we concretely define the parametric data set to be fitted as well as how they are partitioned in the LM framework. The section ends with defining the parametric version of the LM barycentric rational fitting equation.

4.1.1 Parametric Data Definition

Assume an underlying system with \bar{m} inputs and \bar{p} outputs, with an unknown frequency response $\mathbf{H}(s, \theta)$ that depends on both frequency s and some external parameter θ , which in the following we assume to be a scalar variable defined in a finite range $[\theta_1, \theta_{\bar{q}}]$. The system is known through a set of measurements

$$\mathcal{H} = \{\mathbf{H}(s_t, \theta_q); s_t \in \mathcal{S}, \theta_q \in \Theta\} \quad (4.1)$$

where

- $\mathbf{H}(s_t, \theta_q) \in \mathbb{C}^{\bar{m} \times \bar{p}}$ is the transfer function of the system evaluated at s_t and θ_q ;
- $\Theta = \{\theta_q\}_{q=1}^{\bar{q}}$ is a set of single parameter samples with no repeated entries spanning the parameter range $[\theta_1, \theta_{\bar{q}}]$, with corresponding index set $\{1, \dots, \bar{q}\}$;
- \mathcal{S} is defined as in (2.15) as the set of frequency sample points and s_t is a random entry of the set \mathcal{S} :

$$\mathcal{S} = \{s_1, \dots, s_{\bar{t}}\} \quad s_t \in \mathcal{S} \quad (4.2)$$

The goal is to compute a parameterized model whose frequency response approximates the data set (4.1) accurately. Note that the presentation of the PBRLM method goes under the assumption of real $\theta \in \mathbb{R}$ which would allow a more clean explanation of the algorithm.

4.1.2 Parametric Data Partitioning

From the standard LM procedure of Section 2.2.1, we have partitioned the data set \mathcal{H} by dividing the frequency samples \mathcal{S} into two mutually exclusive partitions

$$\Lambda = \{\lambda_1, \dots, \lambda_{\bar{j}}\}, \quad \mathcal{M} = \{\mu_1, \dots, \mu_{\bar{i}}\} \quad (4.3)$$

$$\Lambda \cup \mathcal{M} = \mathcal{S}, \quad \Lambda \cap \mathcal{M} = \emptyset$$

In the context of parametric data set, we can go a step further by including the parameter Θ directly in the frequency partitioning definition since the Θ set itself does not undergo any specific manipulation:

$$\begin{aligned} \Lambda_\Theta &= \{(\lambda_j, \theta_q); \lambda_j \in \Lambda, \theta_q \in \Theta\} \\ \mathcal{M}_\Theta &= \{(\mu_i, \theta_q); \mu_i \in \mathcal{M}, \theta_q \in \Theta\} \end{aligned} \quad (4.4)$$

The respective data partitions matching the frequency partitioning (4.3) are defined as:

$$\begin{aligned} \mathcal{H}_{\Lambda_\Theta} &= \{\mathbf{H}(\lambda_j, \theta_q); \lambda_j \in \Lambda, \theta_q \in \Theta\}, \\ \mathcal{H}_{\mathcal{M}_\Theta} &= \{\mathbf{H}(\mu_i, \theta_q); \mu_i \in \mathcal{M}, \theta_q \in \Theta\}. \end{aligned} \quad (4.5)$$

Note that, differently from [114], we only perform a partition of the frequency samples, whereas all parameter samples are considered as a single set.

4.1.3 Parametric Data Fitting Objective

We want to approximate the data set \mathcal{H} by extending the standard Loewner framework to the parametric case. To this end, using (2.21) rewritten here as:

$$\mathbf{G}(s) = \frac{\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}(\lambda_j)}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{s - \lambda_j}} \quad (4.6)$$

as the template, we add parametric variation to the barycentric rational function:

$$\mathbf{G}(s, \theta) = \frac{\sum_{j=1}^{\bar{j}} \frac{c_j(\theta) \mathbf{H}(\lambda_j, \theta)}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j(\theta)}{s - \lambda_j}}, \quad \forall \theta. \quad (4.7)$$

The addition of dependence on θ does not change the fact that by construction, no matter the choice of non-trivial functions $\{c_j(\theta)\}_{j=1}^{\bar{j}}$, the function $\mathbf{G}(s, \theta)$ in (4.7) provides an exact interpolation at all points in the first frequency partition Λ , but it does so now uniformly in the parameter space:

$$\mathbf{G}(\lambda_j, \theta) \equiv \mathbf{H}(\lambda_j, \theta), \quad \forall \lambda_j \in \Lambda, \forall \theta. \quad (4.8)$$

The determination of an appropriate set of functions $\{c_j(\theta)\}_{j=1}^{\bar{j}}$ in (4.7) will be guided by additional interpolation conditions, to be enforced at the second frequency partition points \mathcal{M} and at the available discrete parameter points Θ through

$$\mathbf{G}(s_t, \theta_q) \equiv \mathbf{H}(s_t, \theta_q), \quad \forall s_t \in \mathcal{S}, \forall \theta_q \in \Theta \quad (4.9)$$

4.2 Formulation in Scalar Data Case

This section goes through the derivation for the PBRLM method based on the specific SISO case: $\bar{m} = 1, \bar{p} = 1$. The SISO case makes understanding of the derivation much easier, and we will simply build upon the scalar data derivations to provide the MIMO case later. We first define the polynomial basis that is used to introduce parametric dependence to the LM barycentric rational equation. We then derive the interpolation condition equations for the data partitions Λ and \mathcal{M} , following which the PLM is defined.

4.2.1 Polynomial Basis

A key part of the PBRLM method consists of using a polynomial basis to characterize and approximate the transfer function dependence on the parameter variable θ . We define a polynomial basis as $\{\phi_y(\theta)\}_{y=1}^{\bar{y}}$, where $\phi_y(\theta)$ is a polynomial of degree $y - 1$. The bare minimum requirement for polynomials of the basis is for them to be linearly independent, but we opt for the more optimized orthonormal polynomials to construct our basis. The condition for a polynomial pair $(\phi_i(\theta), \phi_j(\theta))$ to be orthonormal is defined by the inner product:

$$\langle \phi_i(\theta), \phi_j(\theta) \rangle = \int_{\alpha}^{\beta} w(\theta) \phi_i(\theta) \phi_j(\theta) d\theta \quad \begin{cases} 0 & \text{for } i \neq j \\ 1 & \text{for } i = j \end{cases} \quad (4.10)$$

where:

- $\langle \cdot, \cdot \rangle$ is the inner product operator.
- $w(\theta)$ is the weight function associated with the polynomial basis.
- $[\alpha, \beta]$ is the range of the function variable θ under which orthogonality of the polynomials is defined.

In choosing the polynomial to serve as the basis, the baseline condition is the polynomial orthogonality defined under finite parameter space such that the integral can be rescaled to fit the Θ range. For the presentation of the parametric algorithm in this thesis, we shall use the Legendre polynomial, though any other polynomials with finite space orthogonality can be used as well such as Chebyshev polynomials. From this point on, $\phi_y(\theta)$ will be assumed to be a Legendre polynomial of degree $y - 1$. Legendre polynomials can be defined recursively:

$$(y + 1) \cdot \phi_{y+2}(\theta) = (2y + 1) \cdot \theta \cdot \phi_{y+1}(\theta) - y \cdot \phi_y(\theta) \quad (4.11)$$

with initial terms:

$$\phi_1(\theta) = 1, \quad \phi_2(\theta) = \theta \quad (4.12)$$

Alternatively, Legendre polynomials can be defined using the more direct formula derived from Rodrigues' formula [138]:

$$\varphi_y(\theta) = \frac{1}{2^{y-1}} \sum_{k=0}^{y-1} \binom{y-1}{k}^2 (\theta - 1)^{y-1-k} (\theta + 1)^k \quad (4.13)$$

For reference, here is the list of the first few degrees of Legendre polynomials:

$$\begin{aligned} \phi_2(\theta) &= \frac{1}{2}(3 \cdot \theta^2 - 1) \\ \phi_3(\theta) &= \frac{1}{2}(5 \cdot \theta^3 - 3 \cdot \theta) \\ \phi_4(\theta) &= \frac{1}{8}(35 \cdot \theta^4 - 30 \cdot \theta^2 + 3) \\ \phi_5(\theta) &= \frac{1}{8}(63 \cdot \theta^5 - 70 \cdot \theta^3 + 15 \cdot \theta) \\ &\dots \end{aligned}$$

The orthogonality of the Legendre polynomial is defined with neutral weight function $w(s) = 1$ over the parameter interval $[-1, 1]$:

$$\langle \phi_i(\theta), \phi_j(\theta) \rangle = \int_{-1}^1 \phi_i(\theta) \phi_j(\theta) d\theta \quad \begin{cases} 0 & \text{for } i \neq j \\ \frac{2}{2*(i-1)+1} & \text{for } i = j \end{cases} \quad (4.14)$$

The condition (4.14) needs to be rescaled with an added offset such that the orthogonality is defined over the range $[\theta_1, \theta_{\bar{q}}]$:

$$\vartheta = \frac{2}{\theta_{\bar{q}} - \theta_1} \cdot (\theta - \theta_1) - 1 \quad (4.15)$$

Furthermore, to establish orthonormal relations between the polynomials, we need to add the normalization factor

$$\left(\frac{2 * (y - 1) + 1}{2} \right)^{0.5} \quad (4.16)$$

Define the alternative Legendre polynomials $\varphi_y(\theta)$ that is a modified form of the Legendre polynomials $\phi_y(\theta)$ created using change of variable (4.15) and normalizing factor (4.16):

$$\varphi_y(\theta) = \left(\frac{2 * (y - 1) + 1}{2} \right)^{0.5} \cdot \phi_y(\vartheta) \quad (4.17)$$

The alternative Legendre polynomials $\varphi_y(\theta)$ now fulfil the orthonormal condition:

$$\langle \varphi_i(\theta), \varphi_j(\theta) \rangle = \int_{\theta_1}^{\theta_{\bar{q}}} \varphi_i(\theta) \varphi_j(\theta) d\theta \quad \begin{cases} 0 & \text{for } i \neq j \\ 1 & \text{for } i = j \end{cases} \quad (4.18)$$

From this point on, when we use a Legendre polynomial, we shall refer to $\varphi_i(\theta)$ exclusively, which is simply $\phi_i(\theta)$ with variable offset and rescaled on a case-by-case basis of the example in which the polynomials are used.

4.2.2 Polynomial Barycentric Equation - Partition Λ_Θ Fitting

We start the derivation by rewriting the parametric barycentric equation (4.7):

$$\mathbf{G}(s, \theta) = \frac{\sum_{j=1}^{\bar{j}} \frac{c_j(\theta) \mathbf{H}(\lambda_j, \theta)}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j(\theta)}{s - \lambda_j}} \quad (4.19)$$

The linear combination of entries within polynomial basis $\{\varphi_y(\theta)\}_{y=1}^{\bar{y}}$ is used to approximate the unknown functions $\{c_j(\theta)\}_{j=1}^{\bar{j}}$ in the denominator of (4.19)

$$c_j(\theta) \approx \sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta). \quad (4.20)$$

where the coefficients $c_{y,j}$ can be collected in a matrix $\mathbf{C} \in \mathbb{C}^{\bar{y} \times \bar{j}}$. The same approximation method is applied to the numerator of (4.19), but with a different set of unknown coefficients $d_{y,j}$ collected in a matrix $\mathbf{D} \in \mathbb{C}^{\bar{y} \times \bar{j}}$:

$$c_j(\theta) \mathbf{H}(\lambda_j, \theta) \approx \sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta). \quad (4.21)$$

Applying the polynomial approximation equations (4.20) and (4.21) to (4.19), we obtain the equation:

$$\mathbf{G}(s, \theta) \approx \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta)}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta)}{s - \lambda_j}} \quad (4.22)$$

Because of polynomial approximation of the numerator and denominator, the parametric barycentric equation (4.22) no longer holds the property of perfect interpolation of points in $\mathcal{H}_{\Lambda_\Theta}$ that equation (4.19) had. To reinforce the partition Λ_Θ fitting,

we analyse the following evaluation at a specific point (λ_x, θ_q) from the partition Λ_Θ :

$$\begin{aligned} \mathbf{G}(\lambda_x, \theta_q) &= \lim_{s \rightarrow \lambda_x} \mathbf{G}(s, \theta_q) = \lim_{s \rightarrow \lambda_x} \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q)}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{s - \lambda_j}} \\ &= \lim_{s \rightarrow \lambda_x} \frac{\sum_{y=1}^{\bar{y}} d_{y,x} \varphi_y(\theta_q)}{\sum_{y=1}^{\bar{y}} c_{y,x} \varphi_y(\theta_q)} = \frac{\sum_{y=1}^{\bar{y}} d_{y,x} \varphi_y(\theta_q)}{\sum_{y=1}^{\bar{y}} c_{y,x} \varphi_y(\theta_q)} \end{aligned} \quad (4.23)$$

Thus, from the limit derivation (4.23), we have the general limit result:

$$\mathbf{G}(\lambda_j, \theta_q) = \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q)}{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)} \quad (4.24)$$

for which the ideal outcome of evaluation becomes:

$$\begin{aligned} \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q)}{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)} &= \mathbf{H}(\lambda_j, \theta_q) = \mathbf{H}_{\lambda_j, \theta_q} \\ \sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q) &= \sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q) \cdot \mathbf{H}_{\lambda_j, \theta_q} \end{aligned} \quad (4.25)$$

With the relation (4.25), the condition for (4.22) to fit all frequency data in $\mathcal{H}_{\Lambda_\Theta}$ at $\theta = \theta_q$ now becomes:

$$\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q) = \sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q) \cdot \mathbf{H}_{\lambda_j, \theta_q}, \quad \forall \lambda_j \in \Lambda, \theta_q \in \Theta \quad (4.26)$$

The fitting condition (4.26) can be written in matrix form:

$$\boldsymbol{\varphi}_q^T \mathbf{d}_j = \mathbf{H}_{\lambda_j, \theta_q} \boldsymbol{\varphi}_q^T \mathbf{c}_j \quad (4.27)$$

where

$$\boldsymbol{\varphi}_q^T = \begin{bmatrix} \varphi_1(\theta_q) & \varphi_2(\theta_q) & \cdots & \varphi_{\bar{y}}(\theta_q) \end{bmatrix} \in \mathbb{R}^{1 \times \bar{y}}, \quad \mathbf{c}_j = \begin{bmatrix} c_{1,j} \\ \vdots \\ c_{\bar{y},j} \end{bmatrix}, \quad \mathbf{d}_j = \begin{bmatrix} d_{1,j} \\ \vdots \\ d_{\bar{y},j} \end{bmatrix} \quad (4.28)$$

(4.27) serves as the fitting condition of data of partition Λ_Θ only at the single parameter point θ_q . In order to satisfy interpolation of all points in $\mathcal{H}_{\Lambda_\Theta}$, we assemble all $\theta_q \in \Theta$ cases of (4.27) to produce the full matrix equation:

$$\boldsymbol{\Phi} \mathbf{d}_j = \mathbf{W}_j \boldsymbol{\Phi} \mathbf{c}_j \in \mathbb{C}^{\bar{y} \times 1} \quad (4.29)$$

where

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\varphi}_1^T \\ \vdots \\ \boldsymbol{\varphi}_{\bar{q}}^T \end{bmatrix} \in \mathbb{R}^{\bar{q} \times \bar{y}}, \quad \mathbf{W}_j = \begin{bmatrix} \mathbf{H}_{\lambda_j, \theta_1} & 0 & \cdots & 0 \\ 0 & \mathbf{H}_{\lambda_j, \theta_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{H}_{\lambda_j, \theta_{\bar{q}}} \end{bmatrix} \in \mathbb{C}^{\bar{q} \times \bar{y}} \quad (4.30)$$

Equation (4.29) is the final form of fitting condition for (4.22) to perfectly interpolate all points in $\mathcal{H}_{\Lambda_\Theta}$. However, because the $\boldsymbol{\Phi}$ matrix has dimension \bar{q} (number of parameters) by \bar{y} (number of polynomials) and is thus non-square, the relation between the $c_{y,j}$ and $d_{y,j}$ coefficients are at best established through the pseudo-inverse of $\boldsymbol{\Phi}$:

$$\mathbf{d}_j = \boldsymbol{\Phi}^\dagger \mathbf{W}_j \boldsymbol{\Phi} \mathbf{c}_j \in \mathbb{C}^{\bar{y} \times 1} \quad (4.31)$$

Three distinct outcomes appear from choosing the number of parameter sample points \bar{q} and the number of polynomials \bar{y} :

1. $\bar{q} < \bar{y}$: This is the case where we have more polynomials than parameter sample points, which is undesirable as we would be dealing with an under-determined system with more variables than equations.
2. $\bar{q} = \bar{y}$: This is the unique case when Φ is square and possibly has a true inverse. If an inverse is defined, we would achieve true perfect interpolation of all points in $\mathcal{H}_{\Lambda_\Theta}$:

$$\mathbf{G}(\lambda_j, \theta_q) = \mathbf{H}_{\lambda_j, \theta_q} \quad (4.32)$$

3. $\bar{q} > \bar{y}$: This is the most useful and realistic case, where we have more parameter sample points than polynomials. Though perfect interpolation cannot be achieved at all (λ_j, θ_q) , we can generally attain good approximation:

$$\mathbf{G}(\lambda_j, \theta_q) \approx \mathbf{H}_{\lambda_j, \theta_q} \quad (4.33)$$

Although option 2 seems ideal, it most likely results in the final system poorly interpolating the evaluation point in-between the points (λ_j, θ_q) . In general, having more parameter samples points \bar{q} than number of polynomials \bar{y} result in more numerically sound final systems. Unless specified otherwise, option 3 is assumed being chosen from this point on, including for the simulation examples in Section 4.4.

4.2.3 Polynomial Barycentric Equation - Partition \mathcal{M}_Θ Fitting

For this subsection, we maintain the assumption if SISO case ($\bar{p} = \bar{m} = 1$) for cleaner explanation of the PBRLM algorithm. In section 4.2.2, we established the condition required to fit the parametric frequency data in $\mathcal{H}_{\Lambda_\Theta}$ using the polynomial

infused barycentric equation (4.22) that we rewrite here:

$$\mathbf{G}(s, \theta) = \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta)}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta)}{s - \lambda_j}} \quad (4.34)$$

The next step is to fit the remainder of data found in partition $\mathcal{H}_{\mathcal{M}_\Theta}$. We start with the individual fitting problem for a specific point (μ_i, θ_q) from \mathcal{M}_Θ :

$$\mathbf{G}(\mu_i, \theta_q) = \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}} = \mathbf{H}_{\mu_i, \theta_q} \quad (4.35)$$

from which we perform the following derivations:

$$\begin{aligned} \mathbf{H}_{\mu_i, \theta_q} &= \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}} \\ \mathbf{H}_{\mu_i, \theta_q} \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}} &= \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}} \\ \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \mathbf{H}_{\mu_i, \theta_q} \cdot \varphi_y(\theta_q)}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}} &= \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}} \end{aligned}$$

$$\begin{aligned} & \frac{\sum_{j=1}^{\bar{y}} \frac{\sum_{y=1}^{\bar{y}} [(c_{y,j} \mathbf{H}_{\mu_i, \theta_q} - d_{y,j}) \varphi_y(\theta_q)]}{\mu_i - \lambda_j}}{\sum_{j=1}^{\bar{y}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta_q)}{\mu_i - \lambda_j}} = 0 \\ & \sum_{j=1}^{\bar{y}} \frac{\sum_{y=1}^{\bar{y}} [(c_{y,j} \mathbf{H}_{\mu_i, \theta_q} - d_{y,j}) \varphi_y(\theta_q)]}{\mu_i - \lambda_j} = 0 \end{aligned} \quad (4.36)$$

The derivation can proceed by using the same translation to matrix operation equivalent used on (4.26) to produce (4.27):

$$\sum_{j=1}^{\bar{y}} \frac{\mathbf{H}_{\mu_i, \theta_q} \boldsymbol{\varphi}_q^T \mathbf{c}_j - \boldsymbol{\varphi}_q^T \mathbf{d}_j}{\mu_i - \lambda_j} = 0 \quad (4.37)$$

where $\boldsymbol{\varphi}_q, \mathbf{c}_j \in \mathbb{C}^{\bar{y} \times 1}$ and $\mathbf{d}_j \in \mathbb{C}^{\bar{y} \times 1}$ were defined by (4.28). We can proceed to stack all parameter $\theta_q \in \Theta$ cases of (4.37) to build the full matrix version of the fitting condition:

$$\sum_{j=1}^{\bar{y}} \frac{\mathbf{V}_i \Phi \mathbf{c}_j - \Phi \mathbf{d}_j}{\mu_i - \lambda_j} = \mathbf{0}_{\bar{q} \times 1} \quad (4.38)$$

where $\Phi \in \mathbb{C}^{\bar{q} \times \bar{y}}$ was defined by (4.30), and

$$\mathbf{V}_i = \begin{bmatrix} \mathbf{H}_{\mu_i, \theta_1} & 0 & \cdots & 0 \\ 0 & \mathbf{H}_{\mu_i, \theta_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{H}_{\mu_i, \theta_{\bar{q}}} \end{bmatrix} \in \mathbb{C}^{\bar{q} \times \bar{q}} \quad (4.39)$$

Starting from (4.38), we can finally insert the fitting condition (4.31) which is the main result of section 4.2.2:

$$\sum_{j=1}^{\bar{y}} \frac{\mathbf{V}_i \Phi \mathbf{c}_j - \Phi \mathbf{d}_j}{\mu_i - \lambda_j} = \mathbf{0}_{\bar{q} \times 1}$$

$$\begin{aligned} \sum_{j=1}^{\bar{J}} \frac{\mathbf{V}_i \Phi \mathbf{c}_j - \Phi \Phi^\dagger \mathbf{W}_j \Phi \mathbf{c}_j}{\mu_i - \lambda_j} &= \mathbf{0}_{\bar{q} \times 1} \\ \sum_{j=1}^{\bar{J}} \frac{(\mathbf{V}_i - \Phi \Phi^\dagger \mathbf{W}_j) \Phi \mathbf{c}_j}{\mu_i - \lambda_j} &= \mathbf{0}_{\bar{q} \times 1} \end{aligned} \quad (4.40)$$

4.2.4 Parametric Loewner Matrix Definition and Solution

The parametric Loewner Matrix itself can now be derived based on the combined fitting condition equation (4.40). We start by writing equation (4.40) in expanded matrix form:

$$\left[\frac{(\mathbf{V}_i - \Phi \Phi^\dagger \mathbf{W}_1) \Phi}{\mu_i - \lambda_1} \quad \frac{(\mathbf{V}_i - \Phi \Phi^\dagger \mathbf{W}_2) \Phi}{\mu_i - \lambda_2} \quad \dots \quad \frac{(\mathbf{V}_i - \Phi \Phi^\dagger \mathbf{W}_{\bar{J}}) \Phi}{\mu_i - \lambda_{\bar{J}}} \right] \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \vdots \\ \mathbf{c}_{\bar{J}} \end{bmatrix} = \mathbf{0}_{\bar{q} \times 1} \quad (4.41)$$

If we stack all $\mu_i \in \mathcal{M}$ cases of the matrix equation (4.41), we would obtain the full matrix equation:

$$\left[\begin{array}{cccc} \frac{(\mathbf{V}_1 - \Phi \Phi^\dagger \mathbf{W}_1) \Phi}{\mu_1 - \lambda_1} & \frac{(\mathbf{V}_1 - \Phi \Phi^\dagger \mathbf{W}_2) \Phi}{\mu_1 - \lambda_2} & \dots & \frac{(\mathbf{V}_1 - \Phi \Phi^\dagger \mathbf{W}_{\bar{J}}) \Phi}{\mu_1 - \lambda_{\bar{J}}} \\ \frac{(\mathbf{V}_2 - \Phi \Phi^\dagger \mathbf{W}_1) \Phi}{\mu_2 - \lambda_1} & \frac{(\mathbf{V}_2 - \Phi \Phi^\dagger \mathbf{W}_2) \Phi}{\mu_2 - \lambda_2} & \dots & \frac{(\mathbf{V}_2 - \Phi \Phi^\dagger \mathbf{W}_{\bar{J}}) \Phi}{\mu_2 - \lambda_{\bar{J}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{(\mathbf{V}_{\bar{i}} - \Phi \Phi^\dagger \mathbf{W}_1) \Phi}{\mu_{\bar{i}} - \lambda_1} & \frac{(\mathbf{V}_{\bar{i}} - \Phi \Phi^\dagger \mathbf{W}_2) \Phi}{\mu_{\bar{i}} - \lambda_2} & \dots & \frac{(\mathbf{V}_{\bar{i}} - \Phi \Phi^\dagger \mathbf{W}_{\bar{J}}) \Phi}{\mu_{\bar{i}} - \lambda_{\bar{J}}} \end{array} \right] \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \vdots \\ \mathbf{c}_{\bar{J}} \end{bmatrix} = \mathbf{0}_{\bar{i} \bar{q} \times 1} \quad (4.42)$$

The matrix on the left of equation (4.42) is the PLM of the PBRLM method, which we more formally define as follow:

$$\begin{aligned} \mathbb{L} &= \begin{bmatrix} \mathbb{L}_{\{1,1\}} & \mathbb{L}_{\{1,2\}} & \cdots & \mathbb{L}_{\{1,\bar{j}\}} \\ \mathbb{L}_{\{2,1\}} & \mathbb{L}_{\{2,2\}} & \cdots & \mathbb{L}_{\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{\{\bar{i},1\}} & \mathbb{L}_{\{\bar{i},2\}} & \cdots & \mathbb{L}_{\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{\bar{q} \times \bar{j}\bar{y}} \quad (4.43) \\ \mathbb{L}_{\{i,j\}} &= \frac{(\mathbf{V}_i - \Phi \Phi^\dagger \mathbf{W}_j) \Phi}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{q} \times \bar{y}} \end{aligned}$$

If we also define the vector of coefficients of the denominator of (4.34) in (4.42) as:

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}_1^T & \mathbf{c}_2^T & \cdots & \mathbf{c}_{\bar{j}}^T \end{bmatrix}^T \in \mathbb{C}^{\bar{j}\bar{y} \times 1}, \quad (4.44)$$

we can rewrite (4.42) in the more compact form:

$$\mathbb{L}\mathbf{c} = \mathbf{0}_{\bar{q} \times 1} \quad (4.45)$$

A solution to \mathbf{c} can be found using the exact SVD method used in section 2.2.2 starting at equation (2.27). Ideally, we would like a solution \mathbf{c} which achieves the equality (4.45), but because the matrix \mathbb{L} is most likely not squared, the more realistic solution $\tilde{\mathbf{c}}$ would have the following result:

$$\mathbb{L}\tilde{\mathbf{c}} = \boldsymbol{\epsilon} \quad (4.46)$$

where $\boldsymbol{\epsilon} \in \mathbb{C}^{\bar{q} \times 1}$ is the approximation error vector of all entries of data partition $\mathcal{H}_{\mathcal{M}_\Theta}$. A major difference between $\boldsymbol{\epsilon}$ defined in (4.46) and the one defined for the non-parametric case at (2.28) is that the latter case $\boldsymbol{\epsilon}$ contains the true interpolation

error of data in the non-parametric data set $\mathcal{H}_{\mathcal{M}}$ whereas the former case ϵ contains only approximate interpolation error of data in the parametric data set $\mathcal{H}_{\mathcal{M}_{\Theta}}$. Of course, the difference is caused by the necessity of using the pseudo-inverse term Φ^{\dagger} in (4.40) due to the lack of a true inverse. A solution $\tilde{\mathbf{c}}$ can be obtained through SVD of matrix \mathbb{L} :

$$\mathbb{L} = \mathbf{U} \mathbf{S} \mathbf{V}^H \quad (4.47)$$

Any right singular vector \mathbf{v}_z within \mathbf{V} can be chosen as $\tilde{\mathbf{c}}$, though each affects the F-norm of ϵ as follow:

$$\|\sigma_z\|_F = \|\mathbb{L}\mathbf{v}_z\|_F = \|\epsilon\|_F \quad (4.48)$$

where σ_z is the singular value of index z along the diagonal of \mathbf{S} and which is associated to singular vector \mathbf{v}_z . As the magnitude of singular value σ_z directly dictates the expected norm of ϵ in (4.46), we would naturally choose the right singular vector \mathbf{v}_* associated to a small singular value σ_* in order to minimize the approximate system interpolation error of data in $\mathcal{H}_{\mathcal{M}_{\Theta}}$. Once a solution to \mathbf{c} is found, the coefficients $d_{y,q}$ can be obtained through translation of coefficients in \mathbf{c} through equation (4.31) that we rewrite here:

$$\mathbf{d}_j = \Phi^{\dagger} \mathbf{W}_j \Phi \mathbf{c}_j \in \mathbb{C}^{\bar{y} \times 1} \quad (4.49)$$

4.2.5 Real Polynomial Parametric Loewner Matrix

As was explained in section 2.2.3 for the non-parametric case of the barycentric LM method, we need to add complex conjugate frequency data to the system to achieve a real physical system. With the complex conjugate data set added, we can transform the matrix \mathbb{L} into a real matrix \mathbb{L}_{re} of the same spectrum and obtain a

solution to the coefficient vector \mathbf{c} by applying SVD on the real \mathbb{L}_{re} rather than the complex \mathbb{L} .

The first step is no different from the non-parametric \mathbb{L} , where we add a complete set of complex conjugate data:

$$\Lambda = \{\lambda_1, \lambda_1^*, \lambda_2, \lambda_2^*, \dots, \lambda_{\bar{j}}, \lambda_{\bar{j}}^*\} \quad (4.50a)$$

$$\mathcal{H}_{\Lambda_\Theta} = \{\mathbf{H}(\lambda_j, \theta_q); \lambda_j \in \Lambda, \theta_q \in \Theta\} \quad \mathbf{H}(\lambda_j, \theta_q)^* = \mathbf{H}(\lambda_j^*, \theta_q) \quad (4.50b)$$

$$\mathcal{M} = \{\mu_1, \mu_1^*, \mu_2, \mu_2^*, \dots, \mu_{\bar{i}}, \mu_{\bar{i}}^*\} \quad (4.50c)$$

$$\mathcal{H}_{\mathcal{M}_\Theta} = \{\mathbf{H}(\mu_i, \theta_q); \mu_i \in \mathcal{M}, \theta_q \in \Theta\} \quad \mathbf{H}(\mu_i, \theta_q)^* = \mathbf{H}(\mu_i^*, \theta_q) \quad (4.50d)$$

If we build the polynomial barycentric rational equation (4.22) using the data sets (4.50), the equation will have the following format:

$$\mathbf{H}(s, \theta) = \frac{\sum_{j=1}^{\bar{j}} \left[\frac{\sum_{y=1}^{\bar{y}} d_{j,y} \varphi_y(\theta)}{s - \lambda_j} + \frac{\sum_{y=1}^{\bar{y}} d_{j,y}^* \varphi_y(\theta)}{s - \lambda_j^*} \right]}{\sum_{j=1}^{\bar{j}} \left[\frac{\sum_{y=1}^{\bar{y}} c_{j,y} \varphi_y(\theta)}{s - \lambda_j} + \frac{\sum_{y=1}^{\bar{y}} c_{j,y}^* \varphi_y(\theta)}{s - \lambda_j^*} \right]}. \quad (4.51)$$

This means the polynomial parametric \mathbb{L} takes on the following structure:

$$\mathbb{L} = \begin{bmatrix} \mathbb{L}_{\{1,1\}} & \mathbb{L}_{\{1,2\}} & \cdots & \mathbb{L}_{\{1,\bar{j}\}} \\ \mathbb{L}_{\{2,1\}} & \mathbb{L}_{\{2,2\}} & \cdots & \mathbb{L}_{\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{\{\bar{i},1\}} & \mathbb{L}_{\{\bar{i},2\}} & \cdots & \mathbb{L}_{\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{2\bar{q}\bar{i} \times 2\bar{y}\bar{j}}, \quad \mathbb{L}_{\{i,j\}} \in \mathbb{C}^{2\bar{q} \times 2\bar{y}} = \begin{bmatrix} \alpha_{\{i,j\}} & \beta_{\{i,j\}} \\ \alpha^*_{\{i,j\}} & \beta^*_{\{i,j\}} \end{bmatrix} \quad (4.52)$$

$$\alpha_{\{i,j\}} = \frac{(\mathbf{V}_i - \Phi \Phi^\dagger \mathbf{W}_j) \Phi}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{q} \times \bar{y}}, \quad \beta_{\{i,j\}} = \frac{(\mathbf{V}_i - \Phi \Phi^\dagger \mathbf{W}_j^*) \Phi}{\mu_i - \lambda_j^*} \in \mathbb{C}^{\bar{q} \times \bar{y}}$$

which matches the matrix structure necessary to undergo the real matrix transformation described in detail in Appendix B and specifically in the appendix Section B.4, whose key points we shall go over quickly. The real matrix transform is performed using the unitary transformation matrices $\mathbf{T}_{(\bar{q}\langle\bar{i}\rangle)}$ and $\mathbf{T}_{(\bar{y}\langle\bar{j}\rangle)}$ based on the general matrix definition given by (B.10) that we rewrite here:

$$\mathbf{T}_{(x\langle y\rangle)} \in \mathbb{C}^{2yx \times 2yx} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathcal{T}_{(x)} & & & \\ & \mathcal{T}_{(x)} & & \\ & & \ddots & \\ & & & \mathcal{T}_{(x)} \end{bmatrix}, \quad \mathcal{T}_{(x)} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{x \times x} & -1i \cdot \mathbf{I}_{x \times x} \\ \mathbf{I}_{x \times x} & 1i \cdot \mathbf{I}_{x \times x} \end{bmatrix} \in \mathbb{C}^{2x \times 2x} \quad (4.53)$$

where x and y are variable positive integers. The real matrix transformation equation is:

$$\mathbb{L}_{re} = \mathbf{T}_{(\bar{q}\langle\bar{i}\rangle)}^H \cdot \mathbb{L} \cdot \mathbf{T}_{(\bar{y}\langle\bar{j}\rangle)} \in \mathbb{R}^{2\bar{q}\bar{i} \times 2\bar{y}\bar{j}} \quad (4.54)$$

Alternatively, the real matrix \mathbb{L}_{re} can be constructed using the direct formula:

$$\mathbb{L}_{re} = \begin{bmatrix} \mathbb{L}_{re\{1,1\}} & \mathbb{L}_{re\{1,2\}} & \cdots & \mathbb{L}_{re\{1,\bar{j}\}} \\ \mathbb{L}_{re\{2,1\}} & \mathbb{L}_{re\{2,2\}} & \cdots & \mathbb{L}_{re\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{re\{\bar{i},1\}} & \mathbb{L}_{re\{\bar{i},2\}} & \cdots & \mathbb{L}_{re\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{2\bar{q}\bar{i} \times 2\bar{y}\bar{j}} \quad (4.55)$$

$$\mathbb{L}_{re\{i,j\}} = \begin{bmatrix} \Re\{\alpha_{i,j}\} + \Re\{\beta_{i,j}\} & \Im\{\alpha_{i,j}\} - \Im\{\beta_{i,j}\} \\ -\Im\{\alpha_{i,j}\} - \Im\{\beta_{i,j}\} & \Re\{\alpha_{i,j}\} - \Re\{\beta_{i,j}\} \end{bmatrix} \in \mathbb{R}^{2\bar{q} \times 2\bar{y}} \quad (4.56)$$

$$\alpha_{i,j} = \frac{(\mathbf{V}_i - \Phi\Phi^\dagger \mathbf{W}_j)\Phi}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{q} \times \bar{y}}, \quad \beta_{i,j} = \frac{(\mathbf{V}_i - \Phi\Phi^\dagger \mathbf{W}_j^*)\Phi}{\mu_i - \lambda_j^*} \in \mathbb{C}^{\bar{q} \times \bar{y}} \quad (4.57)$$

Performing SVD on \mathbb{L}_{re} and choosing one of its right singular vector $\hat{\mathbf{v}}_z$ as the solution vector \mathbf{c} , we can then translate $\hat{\mathbf{v}}_z$ back to \mathbf{v}_z , the singular vector of \mathbb{L} , through the simple reverse transform $\mathbf{v}_z = \mathbf{T}_{(\bar{y}(\bar{j}))} \cdot \hat{\mathbf{v}}_z$, which has the final form with complex conjugate coefficients:

$$\mathbf{v}_z = \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_1^* & \mathbf{c}_2 & \mathbf{c}_2^* & \cdots & \mathbf{c}_{\bar{j}} & \mathbf{c}_{\bar{j}}^* \end{bmatrix}^T \quad (4.58)$$

The corresponding translation to the numerator coefficients is no different from (4.49):

$$\mathbf{d}_j = \Phi^\dagger \mathbf{W}_j \Phi \mathbf{c}_j \in \mathbb{C}^{\bar{y} \times 1} \quad (4.59)$$

4.3 Formulation in General Case and Optimization

In this section, we present the translation of the SISO case parametric Loewner Matrix from Section 4.2 to its MIMO case. This section also includes an efficient way for computing the SVD of the parametric Loewner Matrix.

4.3.1 Polynomial Barycentric Equation - MIMO Case

We consider now how the PBRLM method can deal with the MIMO case ($\bar{p} \geq 1$ and $\bar{m} \geq 1$). The starting point would be equation (4.22), which we rewrite here:

$$\mathbf{G}(s, \theta) = \frac{\sum_{j=1}^{\bar{j}} \sum_{y=1}^{\bar{y}} d_{y,j} \varphi_y(\theta)}{\sum_{j=1}^{\bar{j}} \sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta)} \quad (4.60)$$

Define the single scalar entry of our matrix $\mathbf{H}(s, \theta)$ and transfer function $\mathbf{G}(s, \theta)$ at coordinate (p, m) .

$$\mathbf{H}^{\{p,m\}}(s, \theta), \quad \mathbf{G}^{\{p,m\}}(s, \theta) \quad (4.61)$$

Using notation (4.61), we can rewrite (4.60) to generalize to the MIMO case:

$$\mathbf{G}^{\{p,m\}}(s, \theta) = \frac{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} \mathbf{d}_{y,j}^{\{p,m\}} \varphi_y(\theta)}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{\sum_{y=1}^{\bar{y}} c_{y,j} \varphi_y(\theta)}{s - \lambda_j}} \quad (4.62)$$

where the coefficient $\mathbf{d}_{y,j} \in \mathbb{C}^{\bar{p} \times \bar{m}}$ is the matrix version of scalar coefficient $d_{y,j}$ and $\mathbf{d}_{y,j}^{\{p,m\}}$ is defined the exact same way as (4.21), but with the addition of notation (4.61) :

$$c_j(\theta) \mathbf{H}^{\{p,m\}}(\lambda_j, \theta) \approx \sum_{y=1}^{\bar{y}} \mathbf{d}_{y,j}^{\{p,m\}} \varphi_y(\theta) \quad (4.63)$$

The resulting PLMs based on (4.62) constructed following derivation steps of subsections 4.2.2 and 4.2.3 is given as:

$$\mathbb{L}^{\{p,m\}} = \begin{bmatrix} \mathbb{L}^{\{p,m\}}_{\{1,1\}} & \mathbb{L}^{\{p,m\}}_{\{1,2\}} & \dots & \mathbb{L}^{\{p,m\}}_{\{1,\bar{j}\}} \\ \mathbb{L}^{\{p,m\}}_{\{2,1\}} & \mathbb{L}^{\{p,m\}}_{\{2,2\}} & \dots & \mathbb{L}^{\{p,m\}}_{\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}^{\{p,m\}}_{\{\bar{i},1\}} & \mathbb{L}^{\{p,m\}}_{\{\bar{i},2\}} & \dots & \mathbb{L}^{\{p,m\}}_{\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{\bar{q} \times \bar{y}\bar{j}} \quad (4.64)$$

where

$$\mathbb{L}^{\{p,m\}}_{\{i,j\}} = \frac{\left(\mathbf{V}_i^{\{p,m\}} - \Phi \Phi^\dagger \mathbf{W}_j^{\{p,m\}} \right) \Phi}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{q} \times \bar{y}} \quad (4.65)$$

$$\mathbf{W}_j^{\{p,m\}} = \begin{bmatrix} \mathbf{H}_{\lambda_j, \theta_1}^{\{p,m\}} & 0 & \cdots & 0 \\ 0 & \mathbf{H}_{\lambda_j, \theta_2}^{\{p,m\}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{H}_{\lambda_j, \theta_{\bar{q}}}^{\{p,m\}} \end{bmatrix} \in \mathbb{C}^{\bar{q} \times \bar{q}}, \quad \mathbf{H}_{\lambda_j, \theta_q}^{\{p,m\}} = \mathbf{H}^{\{p,m\}}(\lambda_j, \theta_q)$$
(4.66)

$$\mathbf{V}_i^{\{p,m\}} = \begin{bmatrix} \mathbf{H}_{\mu_i, \theta_1}^{\{p,m\}} & 0 & \cdots & 0 \\ 0 & \mathbf{H}_{\mu_i, \theta_2}^{\{p,m\}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{H}_{\mu_i, \theta_{\bar{q}}}^{\{p,m\}} \end{bmatrix} \in \mathbb{C}^{\bar{q} \times \bar{q}}, \quad \mathbf{H}_{\mu_i, \theta_q}^{\{p,m\}} = \mathbf{H}^{\{p,m\}}(\mu_i, \theta_q)$$
(4.67)

The matrix $\mathbb{L}^{\{p,m\}}$ is simply the PLM constructed using the scalar data of $\mathbf{H}(s, \theta)$ at matrix coordinate (p, m) . From (4.62), we note that the same denominator coefficient $c_{y,j}$ is paired with all permutations of $\mathbf{d}_{y,j}^{\{p,m\}}$ with respect to matrix indices p and m . Thus, in attempting to find a solution vector \mathbf{c} containing the coefficients $c_{y,j}$'s through SVD of the PLM, we can stack all permutations of $\mathbb{L}^{\{p,m\}}$ with respect to indices p and m to create a merged PLM \mathbf{L} :

$$\mathbf{L} = \begin{bmatrix} \mathbb{L}^{\{1,1\}} \\ \mathbb{L}^{\{1,2\}} \\ \vdots \\ \mathbb{L}^{\{p,m\}} \\ \vdots \\ \mathbb{L}^{\{\bar{p},\bar{m}\}} \end{bmatrix} \in \mathbb{C}^{\mathbb{C}^{\bar{p}\bar{m}\bar{q}\bar{i} \times \bar{y}\bar{j}}}$$
(4.68)

As was the case for the SISO \mathbb{L} , a left singular vector \mathbf{v}_z of \mathbf{L} associated with a sufficiently small singular value σ_z can be chosen as the coefficient \mathbf{c} vector, but this time, the coefficients are generated taking into account all entries of the original matrix system data $\mathbf{H}(s, \theta)$.

If a real PLM is required, instead of stacking the SISO PLMs $\mathbb{L}^{\{p,m\}}$, we would stack the real SISO PLMs $\mathbb{L}_{re}^{\{p,m\}}$ defined in subsection 4.2.5.

$$\widehat{\mathbf{L}} = \begin{bmatrix} \mathbb{L}_{re}^{\{1,1\}} \\ \mathbb{L}_{re}^{\{1,2\}} \\ \vdots \\ \mathbb{L}_{re}^{\{p,m\}} \\ \vdots \\ \mathbb{L}_{re}^{\{\bar{p},\bar{m}\}} \end{bmatrix} \in \mathbb{R}^{2\bar{p}\bar{m}\bar{q}\bar{i} \times 2\bar{y}\bar{j}} \quad (4.69)$$

where

$$\mathbb{L}_{re}^{\{p,m\}} = \begin{bmatrix} \mathbb{L}_{re}^{\{p,m\}}_{\{1,1\}} & \mathbb{L}_{re}^{\{p,m\}}_{\{1,2\}} & \cdots & \mathbb{L}_{re}^{\{p,m\}}_{\{1,\bar{j}\}} \\ \mathbb{L}_{re}^{\{p,m\}}_{\{2,1\}} & \mathbb{L}_{re}^{\{p,m\}}_{\{2,2\}} & \cdots & \mathbb{L}_{re}^{\{p,m\}}_{\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{re}^{\{p,m\}}_{\{\bar{i},1\}} & \mathbb{L}_{re}^{\{p,m\}}_{\{\bar{i},2\}} & \cdots & \mathbb{L}_{re}^{\{p,m\}}_{\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{2\bar{q}\bar{i} \times 2\bar{y}\bar{j}} \quad (4.70)$$

$$\mathbb{L}_{re}^{\{p,m\}}_{\{i,j\}} = \begin{bmatrix} \Re\{\alpha_{i,j}\} + \Re\{\beta_{i,j}\} & \Im\{\alpha_{i,j}\} - \Im\{\beta_{i,j}\} \\ -\Im\{\alpha_{i,j}\} - \Im\{\beta_{i,j}\} & \Re\{\alpha_{i,j}\} - \Re\{\beta_{i,j}\} \end{bmatrix} \in \mathbb{R}^{2\bar{q} \times 2\bar{y}} \quad (4.71)$$

$$\alpha_{i,j} = \frac{(\mathbf{V}_i^{\{p,m\}} - \Phi\Phi^\dagger \mathbf{W}_j^{\{p,m\}})\Phi}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{q} \times \bar{y}}, \quad \beta_{i,j} = \frac{(\mathbf{V}_i^{\{p,m\}} - \Phi\Phi^\dagger \mathbf{W}_j^{\{p,m\}}^*)\Phi}{\mu_i - \lambda_j^*} \in \mathbb{C}^{\bar{q} \times \bar{y}} \quad (4.72)$$

4.3.2 Efficient SVD for \mathbb{L}

The MIMO Loewner Matrix $\mathbf{L} \in \mathbb{C}^{\bar{p}\bar{m}\bar{q}\bar{i} \times \bar{y}\bar{j}}$ as defined by (4.68) can incur significant computation cost when undergoing SVD if the system has high input count (\bar{m}) and output count (\bar{p}). However, \mathbf{L} typically has significantly more rows than columns since we generally use greater number of θ points than number of polynomials in the basis ($\bar{q} > \bar{y}$) and allocate far greater number of s points in partition \mathcal{M} than partition Λ ($\bar{i} \gg \bar{j}$). The ‘slim’ shape of \mathbf{L} and its structure which consists of vertically stacked SISO PLMs as depicted by (4.68) open the way for efficient computation of a coefficient vector solution.

Consider the approximation error equation (4.46), which we now update to case of the full PLM \mathbf{L} :

$$\mathbf{L}\tilde{\mathbf{c}} = \boldsymbol{\epsilon} \quad (4.73)$$

where $\boldsymbol{\epsilon}$ is still the approximation error for data of partition \mathcal{M}_Θ that we are trying to minimize. We start by casting (4.73) as a norm minimization problem, and we expand it in the equivalent form:

$$\min_{\tilde{\mathbf{c}}} \|\mathbf{L}\tilde{\mathbf{c}}\|_F^2 = \min_{\tilde{\mathbf{c}}} \left\{ \sum_{p,m} \|\mathbb{L}^{\{p,m\}} \tilde{\mathbf{c}}\|_F^2 \right\} \quad (4.74)$$

where $\mathbb{L}^{\{p,m\}} \in \mathbb{C}^{\bar{q}\bar{i} \times \bar{y}\bar{j}}$ is the individual scalar LM constructed using the scalar data of $\mathbf{H}(s, \theta)$ at matrix coordinate (p, m) . $\mathbb{L}^{\{p,m\}}$ is itself a tall and slim matrix ($\bar{q}\bar{i} \gg \bar{y}\bar{j}$), which suggests computing the QR factorization of each block:

$$\mathbb{L}^{\{p,m\}} = \mathbf{Q}^{\{p,m\}} \mathbf{R}^{\{p,m\}} \quad (4.75)$$

where the columns of $\mathbf{Q}^{\{p,m\}} \in \mathbb{C}^{\bar{q}\bar{\iota} \times \bar{y}\bar{\jmath}}$ are orthogonal ($(\mathbf{Q}^{\{p,m\}})^H \mathbf{Q}^{\{p,m\}} = \mathbf{I}_{\bar{y}\bar{\jmath} \times \bar{y}\bar{\jmath}}$) and $\mathbf{R}^{\{p,m\}} \in \mathbb{C}^{\bar{y}\bar{\jmath} \times \bar{y}\bar{\jmath}}$ is square and upper-triangular. We are now cleared to manipulate the term $\|\mathbb{L}^{\{p,m\}} \tilde{\mathbf{c}}\|_F^2$ as follow:

$$\begin{aligned} \|\mathbb{L}^{\{p,m\}} \tilde{\mathbf{c}}\|_F^2 &= \tilde{\mathbf{c}}^H \mathbb{L}^{\{p,m\}}^H \mathbb{L}^{\{p,m\}} \tilde{\mathbf{c}} \\ &= \tilde{\mathbf{c}}^H \mathbf{R}^{\{p,m\}}^H \mathbf{Q}^{\{p,m\}}^H \mathbf{Q}^{\{p,m\}} \mathbf{R}^{\{p,m\}} \tilde{\mathbf{c}} \\ &= \tilde{\mathbf{c}}^H \mathbf{R}^{\{p,m\}}^H \mathbf{R}^{\{p,m\}} \tilde{\mathbf{c}} \\ \|\mathbb{L}^{\{p,m\}} \tilde{\mathbf{c}}\|_F^2 &= \|\mathbf{R}^{\{p,m\}} \tilde{\mathbf{c}}\|_F^2 \end{aligned} \quad (4.76)$$

Using property (4.76), we can write (4.74) as:

$$\min_{\tilde{\mathbf{c}}} \|\mathbf{L} \tilde{\mathbf{c}}\|_F^2 = \min_{\tilde{\mathbf{c}}} \left\{ \sum_{p,m} \|\mathbb{L}^{\{p,m\}} \tilde{\mathbf{c}}\|_F^2 \right\} = \min_{\tilde{\mathbf{c}}} \left\{ \sum_{p,m} \|\mathbf{R}^{\{p,m\}} \tilde{\mathbf{c}}\|_F^2 \right\} = \min_{\tilde{\mathbf{c}}} \|\mathbf{R} \tilde{\mathbf{c}}\|_F^2 \quad (4.77)$$

where we use the newly defined \mathbf{R} matrix which is simply the vertical stack of $\mathbf{R}^{\{p,m\}}$ matrices:

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}^{\{1,1\}} \\ \mathbf{R}^{\{1,2\}} \\ \vdots \\ \mathbf{R}^{\{p,m\}} \\ \vdots \\ \mathbf{R}^{\{\bar{p},\bar{m}\}} \end{bmatrix} \in \mathbb{R}^{\mathbb{C}^{\bar{p}\bar{m}\bar{y}\bar{\jmath} \times \bar{y}\bar{\jmath}}} \quad (4.78)$$

When comparing $\mathbf{L} \in \mathbb{R}^{\mathbb{C}^{\bar{p}\bar{m}\bar{q}\bar{\iota} \times \bar{y}\bar{\jmath}}}$ with $\mathbf{R} \in \mathbb{R}^{\mathbb{C}^{\bar{p}\bar{m}\bar{y}\bar{\jmath} \times \bar{y}\bar{\jmath}}}$, we see that \mathbf{R} has significantly less number of rows ($\bar{q} > \bar{y}$, $\bar{\iota} \gg \bar{\jmath}$, $\bar{q}\bar{\iota} \gg \bar{y}\bar{\jmath}$), which allows SVD applied on \mathbf{R} to be at a considerably lower cost than the cost of SVD applied on \mathbf{L} . Another

advantage of using \mathbf{R} is the possibility of computing \mathbf{R} one sub-matrix $\mathbf{R}^{\{p,m\}}$ at a time. This means we can apply QR-decomposition on individual $\mathbb{L}^{\{p,m\}}$ blocks one at a time, which incur relatively small memory footprint and opens possibility for parallel computing.

It should be noted that the entire process described in this sub-section can be readily applied to the real PLM (4.69). The only modification needed is using $\mathbb{L}_{re}^{\{p,m\}}$ instead of $\mathbb{L}^{\{p,m\}}$ sub-blocks when applying the QR-decomposition.

4.4 PBRLM Simulations

In this section, the proposed PBRLM method is applied to several test cases to demonstrate its capability. The examples used here are directly extracted from [65], where the parametric macromodeling method is officially published.

4.4.1 Example 1

The first example considers a seventh-order Chebyshev low-pass filter, designed to have nominal cut-off frequency $s_0 = 2$ GHz, passband ripple $\varepsilon = 0.5$ dB, and impedance $Z_0 = 50 \Omega$. The ideal component values for an optimal design are $C_1 = C_4 = 2.765$ pF, $L_1 = L_3 = 5.006$ nH, $C_2 = C_4 = 4.199$ pF, $L_2 = 5.349$ nH. However, the following component values, available from the market, are considered: $C_{1,4} = 2.7$ pF, $C_{2,4} = 4.3$ pF, $L_2 = 5.8$ nH. In addition, a parasitic dc resistance of 0.12Ω is included in the inductor models. The parametric variable θ is here the cut-off frequency, which is swept in the range $\theta = [1.5, 2.5]$ GHz. This corresponds to rescaling each component value by a factor s_0/θ .

The two-port S -parameters of the filter are computed for $\bar{q} = 101$ parameter values at $\bar{t} = 501$ frequency points from dc to 4 GHz (both uniformly spaced).

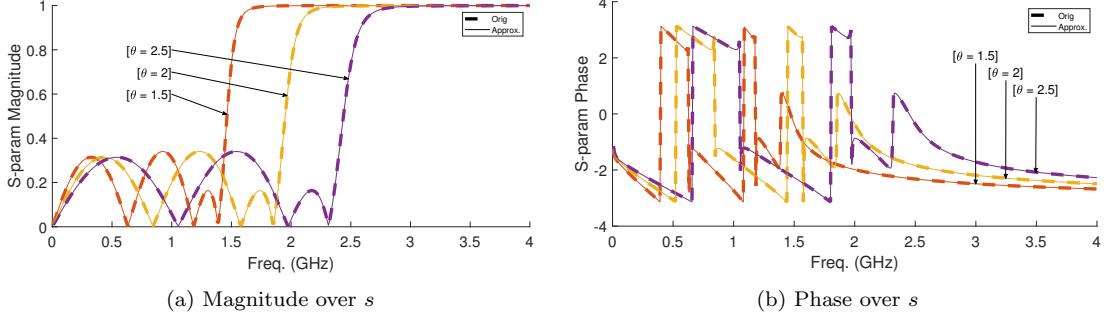


Figure 4.1: Magnitude and phase of $S_{1,1}$ for the Chebyshev filter example of Section 4.4.1, computed for different parameter values. Original data (thick dashed lines) are compared against macromodel responses (thin solid lines).

All these samples are used to build a macromodel using the proposed black-box multivariate LM interpolation. The frequency samples are further divided into the two partitions, $\bar{J} = 4$ in partition Λ , and the remaining $\bar{I} = 496$ samples in partition \mathcal{M} . Based on the real-valued formulation, this corresponds to a model order of 7, matching the one of the actual system. Finally, $\bar{y} = 6$ polynomials (thus, up to order five) are used to approximate the parameter variation.

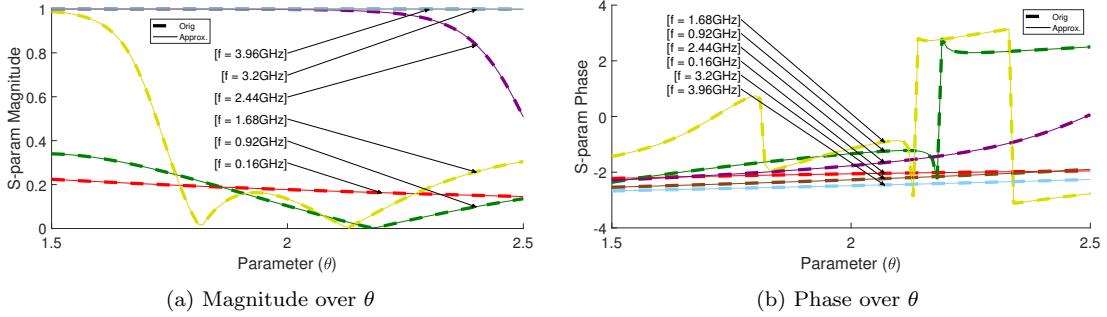


Figure 4.2: Magnitude and phase of $S_{1,1}$ for the Chebyshev filter example of Section 4.4.1, computed for different frequency values as a function of the parameter. Original data (thick dashed lines) are compared against macromodel responses (thin solid lines).

Figure 4.1 shows the magnitude and phase of $S_{1,1}$ over frequency for the minimum, central, and maximum value of the parameter. Figure 4.2 shows the variation

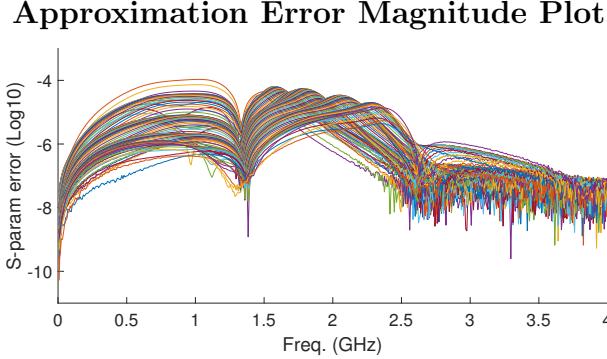


Figure 4.3: Magnitude of the error on $S_{1,1}$ between macromodel and original data for all the available parameter samples of the Chebyshev filter example.

of $S_{1,1}$ with respect to parameter θ at various frequency points. In both cases, the original data are compared against the macromodel, highlighting excellent accuracy. The accuracy is further demonstrated by the errors on $S_{1,1}$ for each of the 101 available parameter samples, provided in Figure 4.3. The maximum absolute error is below 10^{-4} .

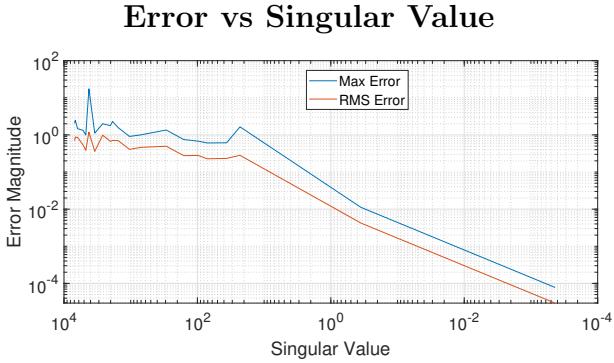


Figure 4.4: Maximum and RMS approximation error against the singular value σ_* used to determine the model coefficients through the corresponding singular vector.

Figure 4.4 depicts the error in $S_{1,1}$ at $\theta = 1.52$ as a function of the singular value σ_* chosen to identify the model coefficients. Both the RMS error and the maximum error over 500 frequency points are shown. As expected, a smaller σ_* leads to a

more accurate multivariate macromodel. This figure shows that the choice of model coefficients is in fact not unique given a desired model accuracy level. The model turns out to be uniformly stable over the entire parameter range. This was verified by a fine parameter sweep.

4.4.2 Example 2

This example considers a microstrip (trace width $w = 0.035$ mm) running from end to end over a slotted ground plane (size: 10 cm \times 10 cm, substrate with relative permittivity $\varepsilon_r = 4.7$ and thickness $t = 0.3$ mm). The slit (total length: 20 mm, width: 0.12 mm) is orthogonal to the microstrip with equal sidelength to the left and right of the microstrip longitudinal axis. The free parameter that we consider is the slit offset from the center of the microstrip, which ranges in the interval $\theta = [1, 15]$ mm. For a more complete description of this geometry, see [131].

RMS Error Minimization Plot

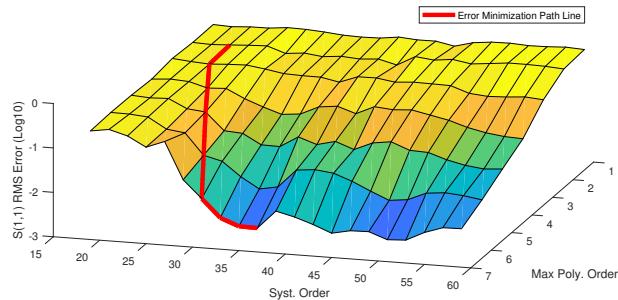


Figure 4.5: RMS error minimization path for the microstrip example of Section 4.4.2.

The initial parameterized frequency responses are computed using a time-domain field solver combined with a fast Fourier transform (FFT) postprocessing, resulting

in 1858 frequency samples of the 2×2 scattering matrix, linearly spaced over the frequency band $[0, 10]$ GHz. A total of $\bar{q} = 15$ sets of scattering responses are available, corresponding to 1-mm steps in θ .

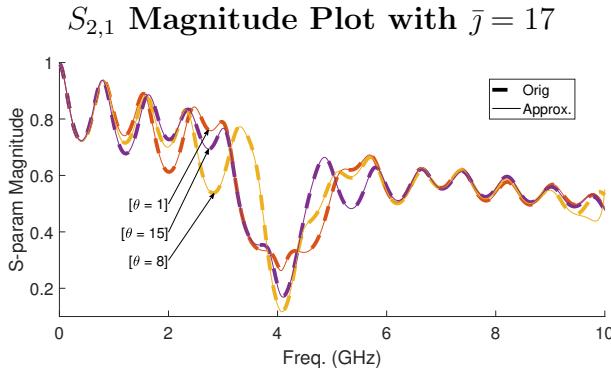


Figure 4.6: Magnitude of $S_{2,1}$ for the microstrip example of Section 4.4.2, computed for different parameter values. Original data (thick dashed lines) are compared against macromodel responses (thin solid lines).

Figure 4.5 reports the evolution of the proposed greedy order estimation process, by depicting the RMS error minimization path leading to the final frequency and parameter orders $\bar{j} = 17$ and $\bar{y} = 7$, respectively. The figure shows that, as expected, the RMS error decreases both along frequency and parameter directions almost monotonically. Figure 4.6 compares the model responses to the original data for the magnitude of $S_{2,1}$ and three parameter values, showing excellent agreement. The accuracy is confirmed by Figure 4.7, which provides the frequency-dependent model-vs-data error for all scattering matrix elements and for all parameter values. We see that this error has local minima at those frequencies that correspond to the frequency points in the first partition Λ . This behaviour is expected and is a

common feature of Loewner-based methods because of the exact interpolation conditions (4.9). The error however does not fall to zero exactly owing to our proposed polynomial approximation of these exact interpolation constraints.

Approximation Error Magnitude Plot with $\bar{J} = 17$

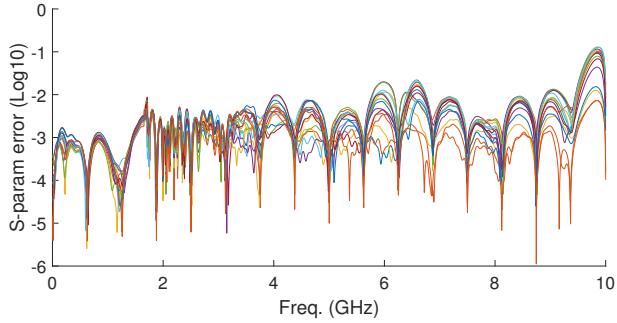


Figure 4.7: Magnitude of model error over frequency, computed for all scattering matrix elements and available parameter samples for the microstrip example of Section 4.4.2.

The performance of the proposed scheme is compared in Figure 4.8 to the multivariate Loewner matrix rational interpolation (MLMRI) scheme of [114]. This figure reports the parameterized model response computed over a very fine sweep in the frequency and parameter plane, in order to check the smoothness of the interpolation/approximation at points that are not in the initial dataset, and for which we cannot therefore enforce accuracy. We see that our method (top panel) provides a uniformly smooth response, whereas the method of [114] (bottom panel) is affected by spurious oscillations and non-smooth behavior. This is readily explained by noting that [114] is based on a double rational barycentric approximation in both frequency and parameter, which is enforced to interpolate exactly the initial responses at selected points in the frequency-parameter space. Nothing in that method prevents the occurrence of model poles in-between raw data samples, and this is exactly what

appears from Figure 4.8. Conversely, our proposed method guarantees smoothness thanks to the overdetermined nature of the parameter fitting problem (4.31), whose least-squares formulation relaxes the exact interpolation constraints and results to be more robust.

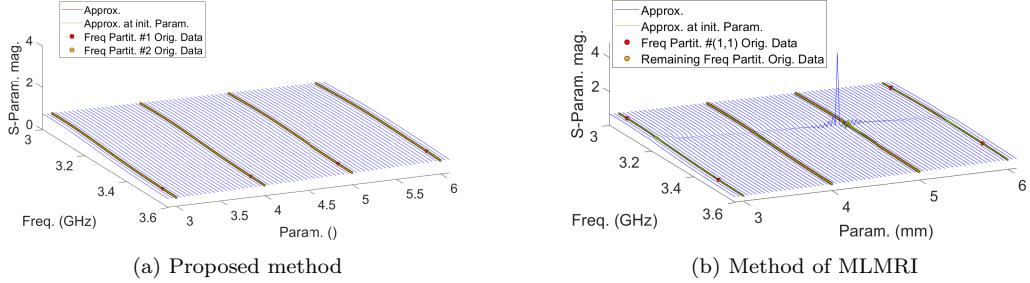


Figure 4.8: Magnitude of model response computed over a fine sweep in the frequency-parameter space (thin solid blue lines) with the proposed method and the method of MLMRI.

Poles Trajectories Over Parameter Range

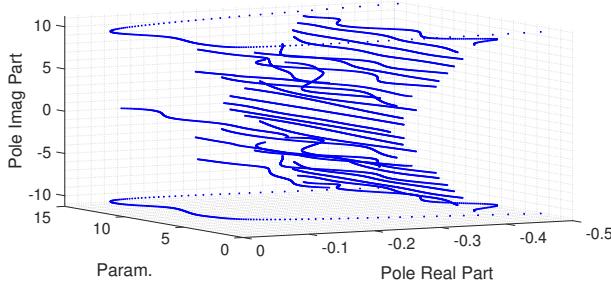


Figure 4.9: Poles trajectories along the parameter space for the microstrip example from Section 4.4.2.

Figure 4.9 depicts the trajectories of the model poles along the parameter space obtained by a fine parameter sweep, showing continuous and smooth poles shifts as the parameter changes. Incidentally, the plot shows that the parametric system is stable throughout the parameter range, as none of the poles have real poles crossing the imaginary axis.

4.4.3 Example 3

This example refers to a high-speed interconnect routed through two multilayer printed circuit boards (PCBs) [139] and one connector, with the inclusion of the corresponding via fields. The via diameter is fixed to $100 \mu\text{m}$, whereas the antipad radius is a free parameter in the range $\theta = [400, 600] \mu\text{m}$. The number of ports of this structure is $\bar{m} = \bar{p} = 2$. A parameter sweep with $\bar{q} = 9$ sets of scattering responses, each with $\bar{t} = 500$ linearly distributed frequency samples spanning the band $[0, 10]$ GHz, were computed with a field solver and used for model extraction.

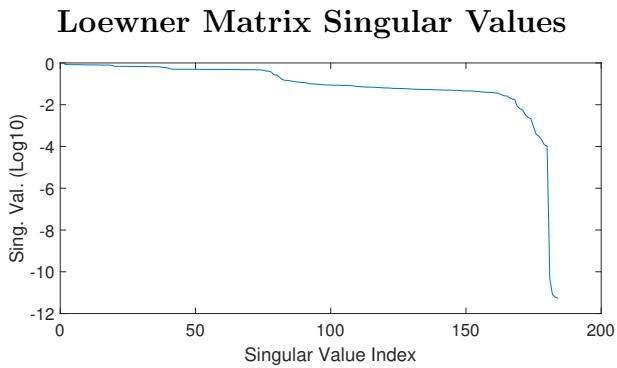


Figure 4.10: Normalized singular values of the PLM for the first example of Section 4.4.3 (parametric variation of antipad radius).

The proposed algorithm led to $\bar{j} = 23$ points in the first frequency partition Λ (resulting in a dynamical order of the resulting parameterized macromodel equal to 45) and $\bar{y} = 4$ polynomial basis functions, corresponding to a third-order polynomial approximation of the model coefficients.

The normalized singular value plot of the Loewner matrix is depicted in Figure 4.10, showing that the least (normalized) singular value is practically down to machine precision. The corresponding singular vector is thus a good approximation

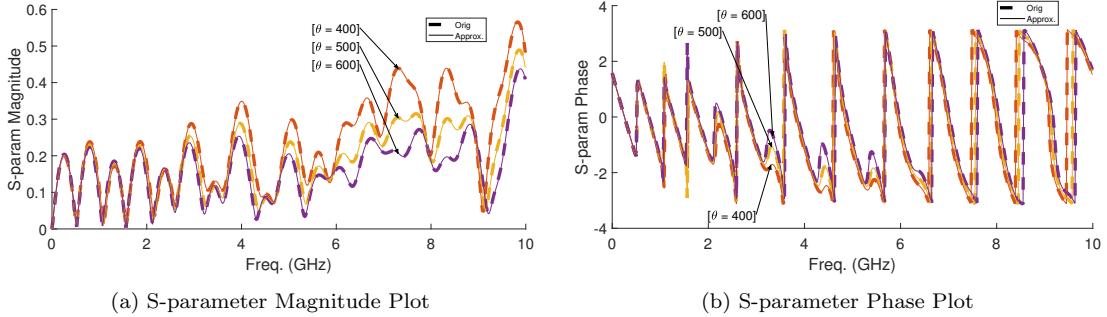


Figure 4.11: Comparison between model and original data for the magnitude and phase of the scattering response $S_{1,1}$ for the first example of Section 4.4.3 (parametric variation of antipad radius).

of the LM null space and is used to define the model coefficients. Figure 4.11 compares the magnitude and phase of $S_{1,1}$ obtained from the model versus original data, for three selected parameter values. These figures show an excellent agreement between model and data. The accuracy is further confirmed by Figure 4.12a, reporting the absolute RMS model error for all available parameter data samples. We see that the error is uniformly below 10^{-3} , except for the upper frequencies. A better control of local model accuracy can be obtained by a dedicated selection of points in the first partition Λ . For instance, Figure 4.12b shows that increasing the density of the first partition points at high frequencies significantly improves accuracy with respect to a uniform sampling (cfr. top panel). Note that for this example, after checking for stability, we determined that the macromodel is stable over the entire parameter range.

For this example, a parameterized model was also generated by applying the PSK scheme of [131]. The resulting errors on all responses are depicted in Figure 4.12c. Since the latter method is based on an iterative least squares process, without explicit exact interpolation constraints as in the proposed Loewner approach,

the error is more uniformly distributed along frequency, without well-defined “deeps”.

In any case, both methods provide parameterized models of excellent accuracy.

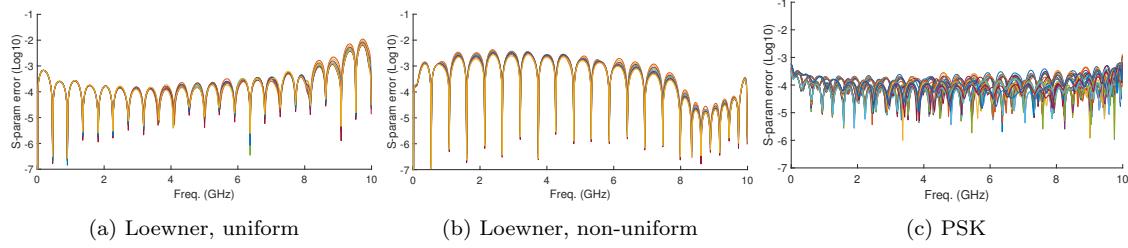


Figure 4.12: Magnitude of model error over frequency, computed for all scattering matrix elements and available parameter samples for the first example of Section 4.4.3 (parametric variation of antipad radius): (a) proposed Loewner approach with uniformly distributed points in the first partition j ; (b) proposed Loewner approach with a denser distribution of first partition points at high frequencies; (c) PSK algorithm.

Chapter 5

Parametric Loewner Matrix System Method

In this chapter, we present the PLM method to which we give the short-hand name parametric system format Loewner matrix (PSFLM). Different from the PBRLM method presented in Chapter 4, the present PSFLM method utilizes the two-variate LM. Different from existing two-variate LM methods [112, 114], the novelty in PSFLM is in the fact that it utilizes the interpolation property of the SFLM method of Section 2.2.3 which has a tendency to result in accurate and numerically sound macromodels that typically achieve stability and stands a good chance to be enforced for passivity if not outright passive already. The PSFLM method is created through direct two-variate extension of the SFLM method. We first present the parametric data partitioning required for the method, following which we define the two variate LM equation which serves as the basis of the parametric method and present how the equation is used to derive a parametric LTI descriptor system. We then present the matrix transformation to transform the LMs into real matrices followed by MOR to provide a more numerically stable lower order system. The chapter closes with a special version of the PSFLM method which closes a theoretical gap of the standard PSFLM method, though at a greater cost.

5.1 Formulation Set-Up

This section sets the preliminary tools that are required in formulating the PSFLM method. The initial parametric data set is once again defined, followed by

the necessary data partitioning required by the LM method. A specialized matrix indexing scheme is then defined to ease readability of the derivation process. The section ends with the definition of the parametric version of the opposite Lagrange interpolation equation.

5.1.1 Parametric Data Definition

Assume an underlying system with \bar{m} inputs and \bar{p} outputs, with an unknown frequency response $\mathbf{H}(s, \theta)$ that depends on both frequency s and some external parameter θ , which in the following we assume to be a scalar variable defined in a finite range $[\theta_1, \theta_{\bar{q}}]$. The system is known through a set of measurements

$$\mathcal{H} = \{\mathbf{H}(s_t, \theta_q); s_t \in \mathcal{S}, \theta_q \in \Theta\} \quad (5.1)$$

where

- $\mathbf{H}(s_t, \theta_q) \in \mathbb{C}^{\bar{m} \times \bar{p}}$ is the transfer function of the system evaluated at s_t and θ_q ;
- $\Theta = \{\theta_q\}_{q=1}^{\bar{q}}$ is a set of single parameter samples with no repeated entries spanning the parameter range $[\theta_1, \theta_{\bar{q}}]$, with corresponding index set $\{1, \dots, \bar{q}\}$;

The parameters are assumed real in this thesis: $\theta_q \in \mathbb{R}$.

- \mathcal{S} is defined as in (2.15) as the set of frequency sample points and s_t is a random entry of the set \mathcal{S} :

$$\mathcal{S} = \{s_1, \dots, s_{\bar{t}}\} \quad s_t \in \mathcal{S} \quad (5.2)$$

The goal is to compute a parameterized model whose frequency response approximates the data set (5.1) accurately.

5.1.2 Parametric Data Partitioning

As was the case for the non-parametric SFLM method in section 2.2.3, the initial parametric frequency data (5.1) is partitioned with respect to the frequency domain into two partitions:

$$\Lambda = \{\lambda_1, \dots, \lambda_{\bar{j}}\}, \quad \mathcal{M} = \{\mu_1, \dots, \mu_{\bar{i}}\} \quad (5.3)$$

$$\Lambda \cup \mathcal{M} = \mathcal{S}, \quad \Lambda \cap \mathcal{M} = \emptyset$$

For the PSFLM method, we additionally perform the same form of partitioning with respect to the parameter domain Θ :

$$\Pi = \{\pi_1, \dots, \pi_{\bar{l}}\}, \quad \mathcal{E} = \{\varepsilon_1, \dots, \varepsilon_{\bar{k}}\} \quad (5.4)$$

$$\Pi \cup \mathcal{E} = \Theta, \quad \Pi \cap \mathcal{E} = \emptyset$$

In combination with the frequency partitioning (5.3), we end up with effectively 4 distinct data partitions:

$$\begin{aligned} \mathcal{H}_{\Lambda, \Pi} &= \{\mathbf{H}(\lambda_j, \pi_l); \lambda_j \in \mathcal{S}_\Lambda, \pi_l \in \Theta_\Pi\} \\ \mathcal{H}_{\Lambda, \mathcal{E}} &= \{\mathbf{H}(\lambda_j, \varepsilon_k); \lambda_j \in \mathcal{S}_\Lambda, \varepsilon_k \in \Theta_\mathcal{E}\} \\ \mathcal{H}_{\mathcal{M}, \Pi} &= \{\mathbf{H}(\mu_i, \pi_l); \mu_i \in \mathcal{S}_\mathcal{M}, \pi_l \in \Theta_\Pi\} \\ \mathcal{H}_{\mathcal{M}, \mathcal{E}} &= \{\mathbf{H}(\mu_i, \varepsilon_k); \mu_i \in \mathcal{S}_\mathcal{M}, \varepsilon_k \in \Theta_\mathcal{E}\} \end{aligned} \quad (5.5)$$

5.1.3 Specialized Layered Matrix Indexing

The presentation of the PLM involves two layers of indexing, one for the s points and one for the θ points. For this reason, a specialized matrix indexing is required and

is explained in this subsection in order to make the reading of subsequent subsections more clear.

We begin with a generic matrix \mathbf{X} to which we give the sub-indexing form $\mathbf{X}_{\{\cdot\}:\cdot\}$ to indicate it has a two-tier sub-matrix division. The upper tier subdivision of the matrix is defined as follow:

$$\mathbf{X} = \mathbf{X}_{\{\cdot\}:\cdot\} = \begin{bmatrix} \mathbf{X}_{\{1,1|\cdot\}} & \mathbf{X}_{\{1,2|\cdot\}} & \cdots & \mathbf{X}_{\{1,\bar{l}|\cdot\}} \\ \mathbf{X}_{\{2,1|\cdot\}} & \mathbf{X}_{\{2,2|\cdot\}} & \cdots & \mathbf{X}_{\{2,\bar{l}|\cdot\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{X}_{\{\bar{k},1|\cdot\}} & \mathbf{X}_{\{\bar{k},2|\cdot\}} & \cdots & \mathbf{X}_{\{\bar{k},\bar{l}|\cdot\}} \end{bmatrix} \in \mathbb{C}^{\bar{k}\bar{i}\bar{p} \times \bar{l}\bar{j}\bar{m}} \quad (5.6)$$

where each matrix $\mathbf{X}_{\{k,l|\cdot\}}$ is a upper tier sub-matrix of $\mathbf{X}_{\{\cdot\}:\cdot\}$. In turn, each upper tier sub-matrix $\mathbf{X}_{\{k,l|\cdot\}}$ has its own subdivision by the corresponding lower tier sub-matrices defined as follow:

$$\mathbf{X}_{\{k,l|\cdot\}} = \begin{bmatrix} \mathbf{X}_{\{k,l|1,1\}} & \mathbf{X}_{\{k,l|1,2\}} & \cdots & \mathbf{X}_{\{k,l|1,\bar{j}\}} \\ \mathbf{X}_{\{k,l|2,1\}} & \mathbf{X}_{\{k,l|2,2\}} & \cdots & \mathbf{X}_{\{k,l|2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{X}_{\{k,l|\bar{i},1\}} & \mathbf{X}_{\{k,l|\bar{i},2\}} & \cdots & \mathbf{X}_{\{k,l|\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{\bar{i}\bar{p} \times \bar{j}\bar{m}} \quad (5.7)$$

where each $\mathbf{X}_{\{k,l|i,j\}} \in \mathbb{C}^{\bar{p} \times \bar{m}}$ matrix is a lower tier sub-matrix. The lower tier sub-matrices are the smallest building blocks of the overall $\mathbf{X}_{\{\cdot\}:\cdot\}$ matrix.

5.1.4 Parametric Opposite Lagrange Equation

The starting point of the equation derivation is with the non-parametric opposite Lagrange equation (2.52) that we rewrite here:

$$\gamma_{\{i,j\}}(s) = \frac{(s - \lambda_j)\mathbf{H}_{\lambda_j} + (\mu_i - s)\mathbf{H}_{\mu_i}}{\mu_i - \lambda_j} \quad (5.8)$$

$$\gamma_{\{i,j\}}(\lambda_j) = \mathbf{H}_{\mu_i}, \quad \gamma_{\{i,j\}}(\mu_i) = \mathbf{H}_{\lambda_j}$$

The two-variate version of (5.8) is given as:

$$\begin{aligned} \Gamma_{\{k,l|i,j\}}(s, \theta) &= \frac{(\mu_i - s)(\varepsilon_k - \theta)\mathbf{H}_{\mu_i, \varepsilon_k} + (\mu_i - s)(\theta - \pi_l)\mathbf{H}_{\mu_i, \pi_l} +}{(\mu_i - \lambda_j)(\varepsilon_k - \pi_l)} \\ &\quad \frac{(s - \lambda_j)(\varepsilon_k - \theta)\mathbf{H}_{\lambda_j, \varepsilon_k} + (s - \lambda_j)(\theta - \pi_l)\mathbf{H}_{\lambda_j, \pi_l}}{(\mu_i - \lambda_j)(\varepsilon_k - \pi_l)} \end{aligned} \quad (5.9)$$

which achieves the opposite point approximation but in a four point quadrant sense as illustrated on Figure 5.1:

$$\Gamma_{\{k,l|i,j\}}(\lambda_j, \pi_l) = \mathbf{H}_{\mu_i, \varepsilon_k} = \mathbf{H}(\mu_i, \varepsilon_k)$$

$$\Gamma_{\{k,l|i,j\}}(\lambda_j, \varepsilon_k) = \mathbf{H}_{\mu_i, \pi_l} = \mathbf{H}(\mu_i, \pi_l)$$

$$\Gamma_{\{k,l|i,j\}}(\mu_i, \pi_l) = \mathbf{H}_{\lambda_j, \varepsilon_k} = \mathbf{H}(\lambda_j, \varepsilon_k)$$

$$\Gamma_{\{k,l|i,j\}}(\mu_i, \varepsilon_k) = \mathbf{H}_{\lambda_j, \pi_l} = \mathbf{H}(\lambda_j, \pi_l)$$

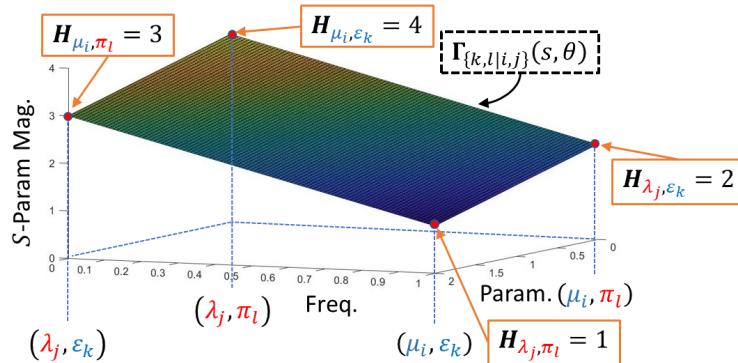


Figure 5.1: Opposite two-variate 1st degree Lagrange polynomial interpolation illustration.

5.2 Formulation

This section formulates the PSFLM method in its standard form. The set of PLMs are defined first and are then used to define the interpolating transfer function equation. The real matrix transformation for the PLMs is then given, followed by MOR of the PLM based transfer function. This section ends with a special definition of the PLM based transfer function that theoretically perfectly interpolates all parametric data, but require derivatives of the data with respect to s .

5.2.1 Parametric Loewner Matrix Definition

The derivation of the PLMs begin with the two-variate opposite Lagrange equation (5.9), which can be conveniently broken down into the summation of four distinct entities scaled by s and/or θ or nothing:

$$\mathbf{\Gamma}_{\{k,l|i,j\}}(s, \theta) = \theta \cdot s \cdot \mathbb{T}_{\{k,l|i,j\}} - \theta \cdot \sigma \mathbb{T}_{\{k,l|i,j\}} - s \cdot \tau \mathbb{T}_{\{k,l|i,j\}} + \sigma \tau \mathbb{T}_{\{k,l|i,j\}} \in \mathbb{C}^{\bar{p} \times \bar{m}} \quad (5.10)$$

where

$$\mathbb{T}_{\{k,l|i,j\}} = \frac{\mathbf{H}_{\mu_i, \varepsilon_k} - \mathbf{H}_{\lambda_j, \varepsilon_k} - \mathbf{H}_{\mu_i, \pi_l} + \mathbf{H}_{\lambda_j, \pi_l}}{(\mu_i - \lambda_j)(\varepsilon_k - \pi_l)} \quad (5.11)$$

$$\sigma \mathbb{T}_{\{k,l|i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i, \varepsilon_k} - \lambda_j \mathbf{H}_{\lambda_j, \varepsilon_k} - \mu_i \mathbf{H}_{\mu_i, \pi_l} + \lambda_j \mathbf{H}_{\lambda_j, \pi_l}}{(\mu_i - \lambda_j)(\varepsilon_k - \pi_l)} \quad (5.12)$$

$$\tau \mathbb{T}_{\{k,l|i,j\}} = \frac{\varepsilon_k \mathbf{H}_{\mu_i, \varepsilon_k} - \varepsilon_k \mathbf{H}_{\lambda_j, \varepsilon_k} - \pi_l \mathbf{H}_{\mu_i, \pi_l} + \pi_l \mathbf{H}_{\lambda_j, \pi_l}}{(\mu_i - \lambda_j)(\varepsilon_k - \pi_l)} \quad (5.13)$$

$$\sigma \tau \mathbb{T}_{\{k,l|i,j\}} = \frac{\mu_i \varepsilon_k \mathbf{H}_{\mu_i, \varepsilon_k} - \lambda_j \varepsilon_k \mathbf{H}_{\lambda_j, \varepsilon_k} - \mu_i \pi_l \mathbf{H}_{\mu_i, \pi_l} + \lambda_j \pi_l \mathbf{H}_{\lambda_j, \pi_l}}{(\mu_i - \lambda_j)(\varepsilon_k - \pi_l)} \quad (5.14)$$

The matrices \mathbb{T} , $\sigma \mathbb{T}$, $\tau \mathbb{T}$, and $\sigma \tau \mathbb{T}$ are the set of PLMs which serve as direct extension to the non-parametric LMs \mathbb{L} and $\sigma \mathbb{L}$. As a matter of fact, the PLMs can be directly

built using the non-parametric set:

$$\mathbb{T}_{\{k,l|i,j\}} = \frac{\mathbb{L}_{\{\varepsilon_k|i,j\}} - \mathbb{L}_{\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \quad (5.15)$$

$$\sigma\mathbb{T}_{\{k,l|i,j\}} = \frac{\sigma\mathbb{L}_{\{\varepsilon_k|i,j\}} - \sigma\mathbb{L}_{\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \quad (5.16)$$

$$\tau\mathbb{T}_{\{k,l|i,j\}} = \frac{\varepsilon_k\mathbb{L}_{\{\varepsilon_k|i,j\}} - \pi_l\mathbb{L}_{\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \quad (5.17)$$

$$\sigma\tau\mathbb{T}_{\{k,l|i,j\}} = \frac{\varepsilon_k\sigma\mathbb{L}_{\{\varepsilon_k|i,j\}} - \pi_l\sigma\mathbb{L}_{\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \quad (5.18)$$

where $\mathbb{L}_{\{\theta_q|i,j\}}$ and $\sigma\mathbb{L}_{\{\theta_q|i,j\}}$ are the non-parametric frequency \mathbb{L} and $\sigma\mathbb{L}$ constructed using frequency data at parameter θ_q . The full parametric matrices $\mathbb{T}, \sigma\mathbb{T}, \tau\mathbb{T}$, and $\sigma\tau\mathbb{T}$ are constructed using the base building blocks (5.11), (5.12), (5.13), and (5.14), respectively, following the two-tiered matrix indexing structure presented in Section 5.1.3.

5.2.2 Parametric Loewner Matrix System Interpolation Equation

Following the definition of the parametric Lagrange equation (5.9), we attempt at defining the same equation as (2.56) which we rewrite here:

$$\mathbf{G}_{\{i,j\}}(s) = \mathbf{H}_{\lambda_j} \cdot [\boldsymbol{\gamma}_{\{i,j\}}(s)]^{-1} \cdot \mathbf{H}_{\mu_i} \quad (5.19)$$

where we used the inverse of $\boldsymbol{\gamma}_{\{i,j\}}(s)$ to create the unit function $\mathbf{G}_{\{i,j\}}(s)$. However, given the two-variate structure of $\mathbf{\Gamma}_{\{k,l|i,j\}}(s, \theta)$ as opposed to the one-variate structure of $\boldsymbol{\gamma}_{\{i,j\}}(s)$, we have two options in terms of base building block of the transfer function:

$$(A) \quad \mathbf{G}_{\{k,l|i,j\}}(s, \theta) = \mathbf{H}_{\lambda_j, \pi_l} [\mathbf{\Gamma}_{\{k,l|i,j\}}(s, \theta)]^{-1} \mathbf{H}_{\mu_i, \varepsilon_k} \quad (5.20)$$

$$\begin{aligned}
 & \rightarrow \begin{cases} \mathbf{G}_{\{k,l|i,j\}}(\lambda_j, \pi_l) = \mathbf{H}_{\lambda_j, \pi_l} \\ \mathbf{G}_{\{k,l|i,j\}}(\mu_i, \varepsilon_k) = \mathbf{H}_{\mu_i, \varepsilon_k} \end{cases} \\
 (B) \quad \mathbf{G}_{\{k,l|i,j\}}(s, \theta) &= \mathbf{H}_{\lambda_j, \varepsilon_k} [\Gamma_{\{k,l|i,j\}}(s, \theta)]^{-1} \mathbf{H}_{\mu_i, \pi_l} \quad (5.21) \\
 & \rightarrow \begin{cases} \mathbf{G}_{\{k,l|i,j\}}(\lambda_j, \varepsilon_k) = \mathbf{H}_{\lambda_j, \varepsilon_k} \\ \mathbf{G}_{\{k,l|i,j\}}(\mu_i, \pi_l) = \mathbf{H}_{\mu_i, \pi_l} \end{cases}
 \end{aligned}$$

The interpolation property can be observed on a make-shift example illustrated on

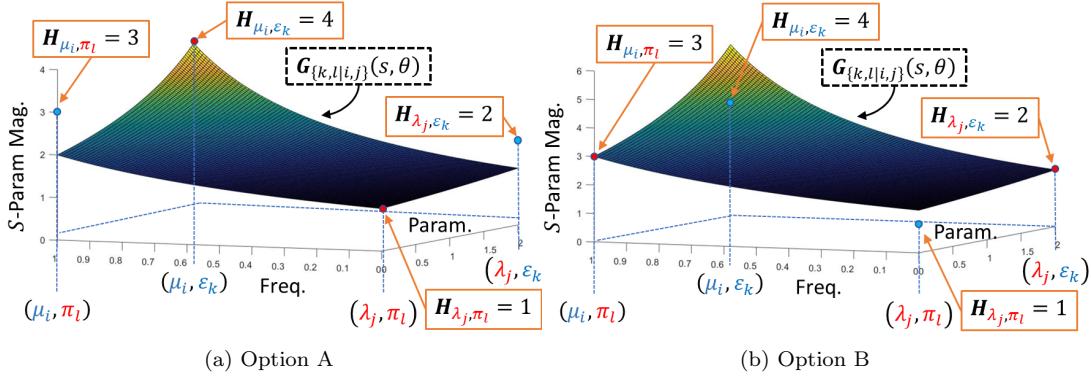


Figure 5.2: Dummy example illustration of the interpolation function (5.20) (left) and (5.21) (right).

Figure 5.2. Using option (A), we would eventually achieve perfect interpolation of all data in partitions $\mathcal{H}_{\Lambda, \Pi}$ and $\mathcal{H}_{\mathcal{M}, \varepsilon}$. Using option (B), we would achieve interpolation of the other two partitions $\mathcal{H}_{\Lambda, \varepsilon}$ and $\mathcal{H}_{\mathcal{M}, \Pi}$. The problem is the two options are mutually exclusive, and no more than two partitions can reach perfect interpolation at the same time following the structure of the LM system. The proposed solution is simply pick one of the two options and forfeit perfect interpolation of the two remaining data partitions. To mitigate the interpolation coverage gap left by this choice, we make sure the two interpolated partitions are spread as evenly as possible over the two-variate approximation domain. The most straightforward approach is

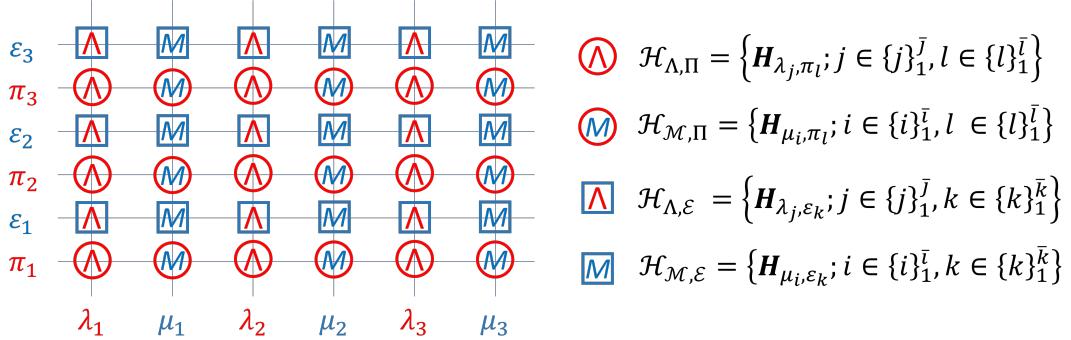


Figure 5.3: Parametric data interleaving partitioning.

to simply use the interleaving partitioning scheme illustrated on Figure 5.3, which involves selecting every other point in one partition when assigning frequency points between frequency partitions Λ and \mathcal{M} and when assigning parameter points between parameter partitions Π and \mathcal{E} .

For all derivation from this point on, we shall assume option (A) was chosen, in which case our unit parametric transfer function is officially defined as:

$$\mathbf{G}_{\{k,l|i,j\}}(s, \theta) = \mathbf{H}_{\lambda_j, \pi_l} [\boldsymbol{\Gamma}_{\{k,l|i,j\}}(s, \theta)]^{-1} \mathbf{H}_{\mu_i, \varepsilon_k} \quad (5.22)$$

Following the same steps as the non-parametric case, we present the full matrix version of (5.22):

$$\mathbf{G}(s, \theta) = \mathbf{W}_{\Lambda,\Pi} [\boldsymbol{\Gamma}(s, \theta)]^{-1} \mathbf{F}_{\mathcal{M},\mathcal{E}} \quad (5.23)$$

and lay claim that $\mathbf{G}(s, \theta)$ perfectly interpolates all data in Λ and \mathcal{M} and where

$$\mathbf{\Gamma}(s, \theta) = \begin{bmatrix} \mathbf{\Gamma}_{\{1,1|\cdot\}}(s, \theta) & \mathbf{\Gamma}_{\{1,2|\cdot\}}(s, \theta) & \cdots & \mathbf{\Gamma}_{\{1,\bar{l}|\cdot\}}(s, \theta) \\ \mathbf{\Gamma}_{\{2,1|\cdot\}}(s, \theta) & \mathbf{\Gamma}_{\{2,2|\cdot\}}(s, \theta) & \cdots & \mathbf{\Gamma}_{\{2,\bar{l}|\cdot\}}(s, \theta) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{\Gamma}_{\{\bar{k},1|\cdot\}}(s, \theta) & \mathbf{\Gamma}_{\{\bar{k},2|\cdot\}}(s, \theta) & \cdots & \mathbf{\Gamma}_{\{\bar{k},\bar{l}|\cdot\}}(s, \theta) \end{bmatrix} \quad (5.24)$$

$$\rightarrow \mathbf{\Gamma}_{\{k,l|\cdot\}}(s, \theta) = \begin{bmatrix} \mathbf{\Gamma}_{\{k,l|1,1\}}(s, \theta) & \mathbf{\Gamma}_{\{k,l|1,2\}}(s, \theta) & \cdots & \mathbf{\Gamma}_{\{k,l|1,\bar{j}\}}(s, \theta) \\ \mathbf{\Gamma}_{\{k,l|2,1\}}(s, \theta) & \mathbf{\Gamma}_{\{k,l|2,2\}}(s, \theta) & \cdots & \mathbf{\Gamma}_{\{k,l|2,\bar{j}\}}(s, \theta) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{\Gamma}_{\{k,l|\bar{i},1\}}(s, \theta) & \mathbf{\Gamma}_{\{k,l|\bar{i},2\}}(s, \theta) & \cdots & \mathbf{\Gamma}_{\{k,l|\bar{i},\bar{j}\}}(s, \theta) \end{bmatrix} \quad (5.25)$$

$$\mathbf{W}_{\Lambda, \Pi} = \begin{bmatrix} \mathbf{W}_{\Lambda, \pi_1} & \mathbf{W}_{\Lambda, \pi_2} & \cdots & \mathbf{W}_{\Lambda, \pi_{\bar{l}}} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times \bar{m} \bar{l}} \quad (5.26)$$

$$\rightarrow \mathbf{W}_{\Lambda, \pi_l} = \begin{bmatrix} \mathbf{H}_{\lambda_1, \pi_l} & \mathbf{H}_{\lambda_2, \pi_l} & \cdots & \mathbf{H}_{\lambda_{\bar{j}}, \pi_l} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times \bar{m} \bar{j}} \quad (5.27)$$

$$\mathbf{F}_{\mathcal{M}, \mathcal{E}} = \begin{bmatrix} \mathbf{F}_{\mathcal{M}, \varepsilon_1}^T & \mathbf{F}_{\mathcal{M}, \varepsilon_2}^T & \cdots & \mathbf{F}_{\mathcal{M}, \varepsilon_{\bar{k}}}^T \end{bmatrix}^T \in \mathbb{C}^{\bar{p} \bar{k} \bar{i} \times \bar{m}} \quad (5.28)$$

$$\rightarrow \mathbf{F}_{\mathcal{M}, \varepsilon_k} = \begin{bmatrix} \mathbf{F}_{\mu_1, \varepsilon_k}^T & \mathbf{F}_{\mu_2, \varepsilon_k}^T & \cdots & \mathbf{F}_{\mu_{\bar{i}}, \varepsilon_k}^T \end{bmatrix}^T \in \mathbb{C}^{\bar{p} \bar{i} \times \bar{m}} \quad (5.29)$$

Theorem 5.2.1 (Parametric System Format Loewner Matrix Interpolation). *The $\mathbf{G}(s, \theta)$ function as defined by (5.22) achieves the perfect interpolation of all data in $\mathcal{H}_{\Lambda, \Pi}$ and $\mathcal{H}_{\mathcal{M}, \mathcal{E}}$, that is:*

$$\begin{aligned} \mathbf{G}(\lambda_j, \pi_l) &= \mathbf{H}_{\lambda_j, \pi_l}, \quad \forall \lambda_j \in \Lambda, \quad \forall \pi_l \in \Pi \\ \mathbf{G}(\mu_i, \varepsilon_k) &= \mathbf{H}_{\mu_i, \varepsilon_k}, \quad \forall \mu_i \in \mathcal{M}, \quad \forall \varepsilon_k \in \mathcal{E} \end{aligned} \quad (5.30)$$

Proof. Suppose we evaluate $\mathbf{G}(s, \theta)$ at any pair of frequency $\lambda_j \in \Lambda$ and parameter $\pi_l \in \Pi$:

$$\mathbf{G}(\lambda_j, \pi_l) = \mathbf{W}_{\Lambda, \Pi} [\Gamma(\lambda_j, \pi_l)]^{-1} \mathbf{F}_{\mathcal{M}, \mathcal{E}} \quad (5.31)$$

We take a closer look at $\Gamma(\lambda_j, \pi_l)$, which has the following structure:

$$\Gamma(\lambda_j, \pi_l) = \begin{bmatrix} \mathbf{z}_{1,1} & \mathbf{z}_{1,2} & \cdots & \Gamma_{\{1,l|:\}}(\lambda_j, \pi_l) & \cdots & \mathbf{z}_{1,\bar{l}} \\ \mathbf{z}_{2,1} & \mathbf{z}_{2,2} & \cdots & \Gamma_{\{2,l|:\}}(\lambda_j, \pi_l) & \cdots & \mathbf{z}_{2,\bar{l}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{z}_{k,1} & \mathbf{z}_{k,2} & \cdots & \Gamma_{\{k,l|:\}}(\lambda_j, \pi_l) & \cdots & \mathbf{z}_{k,\bar{l}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{z}_{\bar{k},1} & \mathbf{z}_{\bar{k},2} & \cdots & \Gamma_{\{\bar{k},l|:\}}(\lambda_j, \pi_l) & \cdots & \mathbf{z}_{\bar{k},\bar{l}} \end{bmatrix} \quad (5.32)$$

where $\mathbf{z}_{k,l}$ is a place-holder matrix for blocks of the matrix that are of no interest. Note that we have purposely highlighted the block column of index l . In turn, observe the sample block matrix $\Gamma_{\{k,l|:\}}(\lambda_j, \pi_l)$:

$$\Gamma_{\{k,l|:\}}(\lambda_j, \pi_l) = \begin{bmatrix} \boldsymbol{\eta}_{\{k,l|1,1\}} & \boldsymbol{\eta}_{\{k,l|1,2\}} & \cdots & \mathbf{H}_{\mu_1, \varepsilon_k} & \cdots & \boldsymbol{\eta}_{\{k,l|1,\bar{j}\}} \\ \boldsymbol{\eta}_{\{k,l|2,1\}} & \boldsymbol{\eta}_{\{k,l|2,2\}} & \cdots & \mathbf{H}_{\mu_2, \varepsilon_k} & \cdots & \boldsymbol{\eta}_{\{k,l|2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{\eta}_{\{k,l|\bar{i},1\}} & \boldsymbol{\eta}_{\{k,l|\bar{i},2\}} & \cdots & \mathbf{H}_{\mu_{\bar{i}}, \varepsilon_k} & \cdots & \boldsymbol{\eta}_{\{k,l|\bar{i},\bar{j}\}} \end{bmatrix} \quad (5.33)$$

where $\boldsymbol{\eta}_{\{k,l|\bar{i},\bar{j}\}}$ is a place-holder matrix for blocks of the matrix $\Gamma_{\{k,l|:\}}(\lambda_j, \pi_l)$ that are of no interest, and where we have purposely highlighted the block column of index j with the matrix vector of frequency partition \mathcal{M} at parameter ε_k . Making use of the definitions (5.29), we can write (5.33) in the more compact form:

$$\Gamma_{\{k,l|:\}}(\lambda_j, \pi_l) = \begin{bmatrix} \boldsymbol{\eta}_{\{k,l|L\}} & \mathbf{F}_{\mathcal{M}, \varepsilon_k} & \boldsymbol{\eta}_{\{k,l|R\}} \end{bmatrix} \quad (5.34)$$

where $\boldsymbol{\eta}_{\{k,l|L\}}$ and $\boldsymbol{\eta}_{\{k,l|R\}}$ are just the matrix subsections on the left and right of block column $\mathbf{F}_{\mathcal{M},\varepsilon_k}$ in $\Gamma_{\{k,l|\cdot\}}(\lambda_j, \pi_l)$. Using (5.34), we can now write (5.32) as follow:

$$\Gamma(\lambda_j, \pi_l) = \begin{bmatrix} \mathbf{z}_{1,1} & \mathbf{z}_{1,2} & \cdots & \left[\boldsymbol{\eta}_{\{1,l|L\}} \quad \mathbf{F}_{\mathcal{M},\varepsilon_1} \quad \boldsymbol{\eta}_{\{1,l|R\}} \right] & \cdots & \mathbf{z}_{1,\bar{l}} \\ \mathbf{z}_{2,1} & \mathbf{z}_{2,2} & \cdots & \left[\boldsymbol{\eta}_{\{2,l|L\}} \quad \mathbf{F}_{\mathcal{M},\varepsilon_2} \quad \boldsymbol{\eta}_{\{2,l|R\}} \right] & \cdots & \mathbf{z}_{2,\bar{l}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{z}_{k,1} & \mathbf{z}_{k,2} & \cdots & \left[\boldsymbol{\eta}_{\{k,l|L\}} \quad \mathbf{F}_{\mathcal{M},\varepsilon_k} \quad \boldsymbol{\eta}_{\{k,l|R\}} \right] & \cdots & \mathbf{z}_{k,\bar{l}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{z}_{\bar{k},1} & \mathbf{z}_{\bar{k},2} & \cdots & \left[\boldsymbol{\eta}_{\{\bar{k},l|L\}} \quad \mathbf{F}_{\mathcal{M},\varepsilon_{\bar{k}}} \quad \boldsymbol{\eta}_{\{\bar{k},l|R\}} \right] & \cdots & \mathbf{z}_{\bar{k},\bar{l}} \end{bmatrix} \quad (5.35)$$

In the special case when $\Gamma(\lambda_j, \pi_l)$ is square and non-singular, the following linear equation solution follows:

$$[\Gamma(\lambda_j, \pi_l)]^{-1} \mathbf{F}_{\mathcal{M},\varepsilon} =$$

$$\begin{bmatrix} \mathbf{z}_{1,1} & \mathbf{z}_{1,2} & \cdots & \left[\boldsymbol{\eta}_{\{1,l|L\}} \quad \mathbf{F}_{\mathcal{M},\varepsilon_1} \quad \boldsymbol{\eta}_{\{1,l|R\}} \right] & \cdots & \mathbf{z}_{1,\bar{l}} \\ \mathbf{z}_{2,1} & \mathbf{z}_{2,2} & \cdots & \left[\boldsymbol{\eta}_{\{2,l|L\}} \quad \mathbf{F}_{\mathcal{M},\varepsilon_2} \quad \boldsymbol{\eta}_{\{2,l|R\}} \right] & \cdots & \mathbf{z}_{2,\bar{l}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{z}_{k,1} & \mathbf{z}_{k,2} & \cdots & \left[\boldsymbol{\eta}_{\{k,l|L\}} \quad \mathbf{F}_{\mathcal{M},\varepsilon_k} \quad \boldsymbol{\eta}_{\{k,l|R\}} \right] & \cdots & \mathbf{z}_{k,\bar{l}} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{z}_{\bar{k},1} & \mathbf{z}_{\bar{k},2} & \cdots & \left[\boldsymbol{\eta}_{\{\bar{k},l|L\}} \quad \mathbf{F}_{\mathcal{M},\varepsilon_{\bar{k}}} \quad \boldsymbol{\eta}_{\{\bar{k},l|R\}} \right] & \cdots & \mathbf{z}_{\bar{k},\bar{l}} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{F}_{\mathcal{M},\varepsilon_1} \\ \mathbf{F}_{\mathcal{M},\varepsilon_2} \\ \vdots \\ \mathbf{F}_{\mathcal{M},\varepsilon_k} \\ \vdots \\ \mathbf{F}_{\mathcal{M},\varepsilon_{\bar{k}}} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{\bar{p} \times \bar{m}} \\ \mathbf{0}_{\bar{p} \times \bar{m}} \\ \vdots \\ \mathbf{I}_{\bar{p} \times \bar{m}} \\ \vdots \\ \mathbf{0}_{\bar{p} \times \bar{m}} \end{bmatrix} \quad (5.36)$$

and one can determine through correct indexing that:

$$\begin{aligned} \mathbf{G}(\lambda_j, \pi_l) &= \mathbf{W}_{\Lambda, \Pi} [\Gamma(\lambda_j, \pi_l)]^{-1} \mathbf{F}_{\mathcal{M}, \mathcal{E}} \\ \mathbf{G}(\lambda_j, \pi_l) &= \begin{bmatrix} \mathbf{H}_{\lambda_1, \pi_1} & \mathbf{H}_{\lambda_2, \pi_1} & \cdots & \mathbf{H}_{\lambda_j, \pi_l} & \cdots & \mathbf{H}_{\lambda_{\bar{j}}, \pi_{\bar{l}}} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{\bar{p} \times \bar{m}} \\ \mathbf{0}_{\bar{p} \times \bar{m}} \\ \vdots \\ \mathbf{I}_{\bar{p} \times \bar{m}} \\ \vdots \\ \mathbf{0}_{\bar{p} \times \bar{m}} \end{bmatrix} \\ \mathbf{G}(\lambda_j, \pi_l) &= \mathbf{H}_{\lambda_j, \pi_l} \end{aligned} \tag{5.37}$$

Since λ_j and π_l are arbitrary entries of Λ and Π , respectively, we have that

$$\mathbf{G}(\lambda_j, \pi_l) = \mathbf{H}_{\lambda_j, \pi_l}, \quad \forall \lambda_j \in \Lambda, \pi_l \in \Pi \tag{5.38}$$

Note that the same proof process can be done on $\mathbf{G}(\mu_i, \varepsilon_k)^T$, and we would arrive at the conclusion that

$$\mathbf{G}(\mu_i, \varepsilon_k)^T = \mathbf{H}_{\mu_i, \varepsilon_k}^T, \quad \forall \mu_i \in \mathcal{M}, \varepsilon_k \in \mathcal{E} \tag{5.39}$$

□

To summarize, perfect approximation of data in $\mathcal{H}_{\Lambda, \Pi}$ and $\mathcal{H}_{\mathcal{M}, \mathcal{E}}$ is achieved by the parametric Loewner Matrix based transfer function:

$$\begin{aligned} \mathbf{G}(s, \theta) &= \mathbf{W}_{\Lambda, \Pi} [\Gamma(s, \theta)]^{-1} \mathbf{F}_{\mathcal{M}, \mathcal{E}} \\ \mathbf{G}(s, \theta) &= \mathbf{W}_{\Lambda, \Pi} [\theta \cdot s \cdot \mathbb{T} - \theta \cdot \sigma \mathbb{T} - s \cdot \tau \mathbb{T} + \sigma \tau \mathbb{T}]^{-1} \mathbf{F}_{\mathcal{M}, \mathcal{E}} \end{aligned} \tag{5.40}$$

5.2.3 Parametric Loewner Matrix Real Transform

Just like in the non-parametric case, we need to ensure physical system with real transfer function matrices. The first step towards real PLMs is no different from the first step taken towards real non-parametric LMs in section 2.2.2 where we added a complete set of complex conjugate data in the frequency domain.

$$\Lambda = \{\lambda_1, \lambda_1^*, \lambda_2, \lambda_2^*, \dots, \lambda_{\bar{j}}, \lambda_{\bar{j}}^*\}, \quad \mathcal{M} = \{\mu_1, \mu_1^*, \lambda_2, \mu_2^*, \dots, \mu_{\bar{i}}, \mu_{\bar{i}}^*\} \quad (5.41a)$$

$$\Pi = \{\pi_1, \dots, \pi_{\bar{l}}\}, \quad \mathcal{E} = \{\varepsilon_1, \dots, \varepsilon_{\bar{k}}\} \quad (5.41b)$$

$$\mathcal{H}_{\Lambda, \Pi} = \{\mathbf{H}(\lambda_j, \pi_l); \lambda_j \in \mathcal{S}_\Lambda, \pi_l \in \Theta_\Pi\}, \quad \mathbf{H}(\lambda_j^*, \pi_l) = \mathbf{H}(\lambda_j, \pi_l)^* \quad (5.41c)$$

$$\mathcal{H}_{\Lambda, \mathcal{E}} = \{\mathbf{H}(\lambda_j, \varepsilon_k); \lambda_j \in \mathcal{S}_\Lambda, \varepsilon_k \in \Theta_\mathcal{E}\}, \quad \mathbf{H}(\lambda_j^*, \varepsilon_k) = \mathbf{H}(\lambda_j, \varepsilon_k)^* \quad (5.41d)$$

$$\mathcal{H}_{\mathcal{M}, \Pi} = \{\mathbf{H}(\mu_i, \pi_l); \mu_i \in \mathcal{S}_\mathcal{M}, \pi_l \in \Theta_\Pi\}, \quad \mathbf{H}(\mu_i^*, \pi_l) = \mathbf{H}(\mu_i, \pi_l)^* \quad (5.41e)$$

$$\mathcal{H}_{\mathcal{M}, \mathcal{E}} = \{\mathbf{H}(\mu_i, \varepsilon_k); \mu_i \in \mathcal{S}_\mathcal{M}, \varepsilon_k \in \Theta_\mathcal{E}\}, \quad \mathbf{H}(\mu_i^*, \varepsilon_k) = \mathbf{H}(\mu_i, \varepsilon_k)^* \quad (5.41f)$$

Because the PLMs are essentially variable θ LMs of variable s LMs as described by (5.15), (5.16), (5.17), and (5.18), they inherit the matrix structure necessary for the real matrix transform process described in detail in Appendix B and specifically in the appendix Section B.4. As such, we can perform the transform using the real transformation matrices in the form of the general matrix $\mathbf{T}_{(x(y))}$ by simply adjusting the number of unit transformation sub-matrices $\mathcal{T}_{(x)}$ that it holds. $\mathbf{T}_{(x(y))}$ and $\mathcal{T}_{(x)}$

are defined by (B.10) which we rewrite here:

$$\mathbf{T}_{(x\langle y \rangle)} \in \mathbb{C}^{2yx \times 2yx} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathcal{T}_{(x)} & & & \\ & \mathcal{T}_{(x)} & & \\ & & \ddots & \\ & & & \mathcal{T}_{(x)} \end{bmatrix}, \quad \mathcal{T}_{(x)} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{x \times x} & -1i \cdot \mathbf{I}_{x \times x} \\ \mathbf{I}_{x \times x} & 1i \cdot \mathbf{I}_{x \times x} \end{bmatrix} \in \mathbb{C}^{2x \times 2x} \quad (5.42)$$

$\mathcal{T}_{(x)}$ is repeated in $\mathbf{T}_{(x\langle y \rangle)}$ a number of time taking into account of the number of parameters in Π and \mathcal{E} . The final full transformation matrices are defined as:

$$\begin{aligned} \text{Left Transfom Matrix} &\rightarrow \mathbf{T}_{(\bar{p}\langle \bar{k}\bar{i} \rangle)} \in \mathbb{C}^{2\bar{k}\bar{i}\bar{p} \times 2\bar{k}\bar{i}\bar{p}} \\ \text{Right Transfom Matrix} &\rightarrow \mathbf{T}_{(\bar{m}\langle \bar{l}\bar{j} \rangle)} \in \mathbb{C}^{2\bar{l}\bar{j}\bar{m} \times 2\bar{l}\bar{j}\bar{m}} \end{aligned} \quad (5.43)$$

The real matrix transformations utilizing $\mathbf{T}_{(\bar{p}\langle \bar{k}\bar{i} \rangle)}$ and $\mathbf{T}_{(\bar{m}\langle \bar{l}\bar{j} \rangle)}$ are defined as follow:

$$\begin{aligned} \mathbb{T}_{re} &= \mathbf{T}_{(\bar{p}\langle \bar{k}\bar{i} \rangle)}^H \cdot \mathbb{T} \cdot \mathbf{T}_{(\bar{m}\langle \bar{l}\bar{j} \rangle)}, & \sigma\mathbb{T}_{re} &= \mathbf{T}_{(\bar{p}\langle \bar{k}\bar{i} \rangle)}^H \cdot \sigma\mathbb{T} \cdot \mathbf{T}_{(\bar{m}\langle \bar{l}\bar{j} \rangle)}, \\ \tau\mathbb{T}_{re} &= \mathbf{T}_{(\bar{p}\langle \bar{k}\bar{i} \rangle)}^H \cdot \tau\mathbb{T} \cdot \mathbf{T}_{(\bar{m}\langle \bar{l}\bar{j} \rangle)}, & \sigma\tau\mathbb{T}_{re} &= \mathbf{T}_{(\bar{p}\langle \bar{k}\bar{i} \rangle)}^H \cdot \sigma\tau\mathbb{T} \cdot \mathbf{T}_{(\bar{m}\langle \bar{l}\bar{j} \rangle)}, \\ \mathbf{W}_{\Lambda,\Pi re} &= \mathbf{W}_{\Lambda,\Pi} \cdot \mathbf{T}_{(\bar{m}\langle \bar{l}\bar{j} \rangle)} \in \mathbb{R}^{\bar{p} \times 2\bar{l}\bar{j}\bar{m}}, & \mathbf{F}_{\mathcal{M},\mathcal{E} re} &= \mathbf{T}_{(\bar{p}\langle \bar{k}\bar{i} \rangle)}^H \cdot \mathbf{F}_{\mathcal{M},\mathcal{E}} \in \mathbb{R}^{2\bar{k}\bar{i}\bar{p} \times \bar{m}} \\ \mathbb{T}_{re}, \sigma\mathbb{T}_{re}, \tau\mathbb{T}_{re}, \sigma\tau\mathbb{T}_{re} &\in \mathbb{R}^{2\bar{p}\bar{k}\bar{i} \times 2\bar{m}\bar{l}\bar{j}} \end{aligned} \quad (5.44)$$

The real $\mathbf{W}_{\Lambda,\Pi re}$ and $\mathbf{F}_{\Lambda,\Pi re}$ matrices can be constructed using the direct formula:

$$\begin{aligned} \mathbf{W}_{\Lambda,\Pi re} &= \begin{bmatrix} \mathbf{W}_{\Lambda,\pi_1 re} & \mathbf{W}_{\Lambda,\pi_2 re} & \dots & \mathbf{W}_{\Lambda,\pi_{\bar{l}} re} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times 2\bar{m}\bar{l}\bar{j}} \\ \mathbf{W}_{\Lambda,\pi_l re} &= \sqrt{2} \begin{bmatrix} \Re\{\mathbf{H}_{\lambda_1,\pi_l}\} & \Im\{\mathbf{H}_{\lambda_1,\pi_l}\} & \Re\{\mathbf{H}_{\lambda_2,\pi_l}\} & \Im\{\mathbf{H}_{\lambda_2,\pi_l}\} & \dots \\ \Re\{\mathbf{H}_{\lambda_{\bar{j}},\pi_l}\} & \Im\{\mathbf{H}_{\lambda_{\bar{j}},\pi_l}\} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times 2\bar{m}\bar{j}} \end{aligned} \quad (5.45)$$

$$\begin{aligned}\mathbf{F}_{\mathcal{M},\mathcal{E}} &= \left[\mathbf{F}_{\mathcal{M},\varepsilon_1}^T \quad \mathbf{F}_{\mathcal{M},\varepsilon_2}^T \quad \cdots \quad \mathbf{F}_{\mathcal{M},\varepsilon_{\bar{k}}}^T \right]^T \in \mathbb{C}^{2\bar{p}\bar{k}\bar{i} \times \bar{m}} \\ \mathbf{F}_{\mathcal{M},\varepsilon_k re} &= \sqrt{2} \begin{bmatrix} \Re\{\mathbf{H}_{\mu_1,\varepsilon_k}\}^T & \Im\{\mathbf{H}_{\mu_1,\varepsilon_k}\}^T & \Re\{\mathbf{H}_{\mu_1,\varepsilon_k}\}^T & \Im\{\mathbf{H}_{\mu_1,\varepsilon_k}\}^T & \cdots \\ \Re\{\mathbf{H}_{\mu_{\bar{i}},\varepsilon_k}\}^T & \Im\{\mathbf{H}_{\mu_{\bar{i}},\varepsilon_k}\}^T \end{bmatrix}^T \in \mathbb{C}^{2\bar{p}\bar{i} \times \bar{m}}\end{aligned}\quad (5.46)$$

The real \mathbb{T}_{re} can be constructed using the direct formula:

$$\mathbb{T}_{re} = \begin{bmatrix} \mathbb{T}_{re\{1,1|\cdot\}} & \mathbb{T}_{re\{1,2|\cdot\}} & \cdots & \mathbb{T}_{re\{1,\bar{l}|\cdot\}} \\ \mathbb{T}_{re\{2,1|\cdot\}} & \mathbb{T}_{re\{2,2|\cdot\}} & \cdots & \mathbb{T}_{re\{2,\bar{l}|\cdot\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{T}_{re\{\bar{k},1|\cdot\}} & \mathbb{T}_{re\{\bar{k},2|\cdot\}} & \cdots & \mathbb{T}_{re\{\bar{k},\bar{l}|\cdot\}} \end{bmatrix} \in \mathbb{C}^{2\bar{k}\bar{i}\bar{p} \times 2\bar{l}\bar{j}\bar{m}} \quad (5.47)$$

$$\mathbb{T}_{re\{k,l|\cdot\}} = \begin{bmatrix} \mathbb{T}_{re\{k,l|1,1\}} & \mathbb{T}_{re\{k,l|1,2\}} & \cdots & \mathbb{T}_{re\{k,l|1,\bar{j}\}} \\ \mathbb{T}_{re\{k,l|2,1\}} & \mathbb{T}_{re\{k,l|2,2\}} & \cdots & \mathbb{T}_{re\{k,l|2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{T}_{re\{k,l|\bar{i},1\}} & \mathbb{T}_{re\{k,l|\bar{i},2\}} & \cdots & \mathbb{T}_{re\{k,l|\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{2\bar{i}\bar{p} \times 2\bar{j}\bar{m}} \quad (5.48)$$

$$\mathbb{T}_{re\{k,l|i,j\}} = \frac{\mathbb{L}_{re\{\varepsilon_k|i,j\}} - \mathbb{L}_{re\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \in \mathbb{R}^{2\bar{p} \times 2\bar{m}} \quad (5.49)$$

$$\mathbb{L}_{re\{\theta|i,j\}} = \begin{bmatrix} \Re\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} + \Re\{\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Im\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\mathbb{L}_{\beta\{\theta|i,j\}}\} \\ -\Im\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Re\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Re\{\mathbb{L}_{\beta\{\theta|i,j\}}\} \end{bmatrix} \quad (5.50)$$

$$\mathbb{L}_{\alpha\{\theta|i,j\}} = \frac{\mathbf{H}_{\mu_i,\theta} - \mathbf{H}_{\lambda_j,\theta}}{\mu_i - \lambda_j}, \quad \mathbb{L}_{\beta\{\theta|i,j\}} = \frac{\mathbf{H}_{\mu_i,\theta} - \mathbf{H}_{\lambda_j,\theta}^*}{\mu_i - \lambda_j^*} \quad (5.51)$$

The matrix structure hierarchy of \mathbb{T}_{re} can be summarized through the short expression:

$$\mathbb{T}_{re} \leftarrow \mathbb{T}_{re\{k,l|i,j\}} = \frac{\mathbb{L}_{re\{\varepsilon_k|i,j\}} - \mathbb{L}_{re\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \begin{cases} \mathbb{L}_{\alpha\{\theta|i,j\}} = \frac{\mathbf{H}_{\mu_i,\theta} - \mathbf{H}_{\lambda_j,\theta}}{\mu_i - \lambda_j} \\ \mathbb{L}_{\beta\{\theta|i,j\}} = \frac{\mathbf{H}_{\mu_i,\theta} - \mathbf{H}_{\lambda_j,\theta}^*}{\mu_i - \lambda_j^*} \end{cases} \quad (5.52)$$

$$\mathbb{L}_{re\{\theta|i,j\}} = \begin{bmatrix} \Re\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} + \Re\{\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Im\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\mathbb{L}_{\beta\{\theta|i,j\}}\} \\ -\Im\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Re\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Re\{\mathbb{L}_{\beta\{\theta|i,j\}}\} \end{bmatrix}$$

Since the PLMs $\sigma\mathbb{T}, \tau\mathbb{T}$ and $\sigma\tau\mathbb{T}$ have the same structure as \mathbb{T} in the context of real matrix transform, we avoid fully describing the direct formula of their real transformation and simply state their short expression using (5.52) as template:

$$\sigma\mathbb{T}_{re} \leftarrow \sigma\mathbb{T}_{re\{k,l|i,j\}} = \frac{\sigma\mathbb{L}_{re\{\varepsilon_k|i,j\}} - \sigma\mathbb{L}_{re\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \begin{cases} \sigma\mathbb{L}_{\alpha\{\theta|i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i,\theta} - \lambda_j \mathbf{H}_{\lambda_j,\theta}}{\mu_i - \lambda_j} \\ \sigma\mathbb{L}_{\beta\{\theta|i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i,\theta} - \lambda_j^* \mathbf{H}_{\lambda_j,\theta}^*}{\mu_i - \lambda_j^*} \end{cases}$$

$$\sigma\mathbb{L}_{re\{\theta|i,j\}} = \begin{bmatrix} \Re\{\sigma\mathbb{L}_{\alpha\{\theta|i,j\}}\} + \Re\{\sigma\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Im\{\sigma\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\sigma\mathbb{L}_{\beta\{\theta|i,j\}}\} \\ -\Im\{\sigma\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\sigma\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Re\{\sigma\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Re\{\sigma\mathbb{L}_{\beta\{\theta|i,j\}}\} \end{bmatrix} \quad (5.53)$$

$$\tau\mathbb{T}_{re} \leftarrow \tau\mathbb{T}_{re\{k,l|i,j\}} = \frac{\varepsilon_k \cdot \mathbb{L}_{re\{\varepsilon_k|i,j\}} - \pi_l \cdot \mathbb{L}_{re\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \begin{cases} \mathbb{L}_{\alpha\{\theta|i,j\}} = \frac{\mathbf{H}_{\mu_i,\theta} - \mathbf{H}_{\lambda_j,\theta}}{\mu_i - \lambda_j} \\ \mathbb{L}_{\beta\{\theta|i,j\}} = \frac{\mathbf{H}_{\mu_i,\theta} - \mathbf{H}_{\lambda_j,\theta}^*}{\mu_i - \lambda_j^*} \end{cases}$$

$$\mathbb{L}_{re\{\theta|i,j\}} = \begin{bmatrix} \Re\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} + \Re\{\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Im\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\mathbb{L}_{\beta\{\theta|i,j\}}\} \\ -\Im\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Re\{\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Re\{\mathbb{L}_{\beta\{\theta|i,j\}}\} \end{bmatrix} \quad (5.54)$$

$$\begin{aligned}
 \sigma\tau\mathbb{T}_{re} &\leftarrow \sigma\tau\mathbb{T}_{re\{k,l|i,j\}} \\
 \sigma\tau\mathbb{T}_{re\{k,l|i,j\}} &= \frac{\varepsilon_k \cdot \sigma\mathbb{L}_{re\{\varepsilon_k|i,j\}} - \pi_l \cdot \sigma\mathbb{L}_{re\{\pi_l|i,j\}}}{\varepsilon_k - \pi_l} \quad \left\{ \begin{array}{l} \sigma\mathbb{L}_{\alpha\{\theta|i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i,\theta} - \lambda_j \mathbf{H}_{\lambda_j,\theta}}{\mu_i - \lambda_j} \\ \sigma\mathbb{L}_{\beta\{\theta|i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i,\theta} - \lambda_j^* \mathbf{H}_{\lambda_j,\theta}^*}{\mu_i - \lambda_j^*} \end{array} \right. \\
 \sigma\mathbb{L}_{re\{\theta|i,j\}} &= \begin{bmatrix} \Re\{\sigma\mathbb{L}_{\alpha\{\theta|i,j\}}\} + \Re\{\sigma\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Im\{\sigma\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\sigma\mathbb{L}_{\beta\{\theta|i,j\}}\} \\ -\Im\{\sigma\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Im\{\sigma\mathbb{L}_{\beta\{\theta|i,j\}}\} & \Re\{\sigma\mathbb{L}_{\alpha\{\theta|i,j\}}\} - \Re\{\sigma\mathbb{L}_{\beta\{\theta|i,j\}}\} \end{bmatrix} \quad (5.55)
 \end{aligned}$$

5.2.4 Parametric Loewner Matrix System Model Order Reduction

As was the case for the non-parametric system, a model order reduction is required on the PLM system in order to attain a numerically stable system. A similar model order reduction process to the one presented in Section 2.2.3 is utilized to create the order reduced system, which starts with the definition of the PLM pencil:

$$\Gamma(x, y) = |x| \cdot y \cdot \mathbb{T} - y \cdot \sigma\mathbb{T} - |x| \cdot \tau\mathbb{T} + \sigma\tau\mathbb{T} \quad (5.56)$$

where $y \in \Theta$, $x \in \mathcal{S}$, and we have reused the modification made for the non-parametric LM pencil (2.81) by using the magnitude of the imaginary frequency in order to keep the PLM pencil real as well. We perform singular value decomposition on matrix $\Gamma(x, y)$:

$$\Gamma(x, y) = \mathbf{U} \mathbf{S} \mathbf{V}^H \quad (5.57)$$

Given an ideal system order of n , we perform a singular value cut-off at singular value index n , resulting in:

- \mathbf{U}_n : \mathbf{U} with only its first n columns.
- \mathbf{U}_n : \mathbf{V} with only its first n columns.

The order reduced system matrices can then be defined as follow:

$$\mathbf{E}(\theta) = \mathbf{U}_n^H (\theta \cdot \mathbb{T} - \tau \mathbb{T}) \mathbf{V}_n \quad (5.58)$$

$$\mathbf{A}(\theta) = \mathbf{U}_n^H (\theta \cdot \sigma \mathbb{T} - \sigma \tau \mathbb{T}) \mathbf{V}_n \quad (5.59)$$

$$\mathbf{B} = \mathbf{U}_n^H \mathbf{F}_{\mathcal{M}, \mathcal{E}} \quad (5.60)$$

$$\mathbf{C} = \mathbf{W}_{\Lambda, \Pi} \mathbf{V}_n \quad (5.61)$$

with the final approximation transfer function being defined as:

$$\mathbf{G}(s, \theta) = \mathbf{C} [s \mathbf{E}(\theta) - \mathbf{A}(\theta)]^{-1} \mathbf{B} \quad (5.62)$$

Given that the system was reduced to an order n , equation (5.62) does not possess the perfect interpolation property of equation (5.40). However, if n is well chosen, (5.62) can achieve both satisfactory accuracy and system stability, whereas equation (5.40) could only achieve perfect accuracy at designated points and unlikely achieve system stability or even numerical stability (ill-conditioned \mathbf{E} and \mathbf{A} matrices). Unfortunately, just like the non-parametric method SFLM, there is no solve-all methodology in selecting n because n is affected by a number of factors such as the nature of the system, the number of data utilized, the range of data, noise in the data, and the partitioning method used on the data. In the non-parametric case presented in Section 2.2.3, it was still semi-plausible to have automatic order selection process in the form of binary search using as measure the combination of RMS error and system stability. In the PSFLM case, such a binary search method is unreliable as the stability profile in general evolves far more erratically as a function of n .

5.2.5 Parametric Loewner Matrix 4 Quadrant Formulation

When defining the matrices of the PSFLM method in subsection 5.2.2, we had to make a critical choice between (5.20) and (5.21) which decided which partition data pairs $\{\mathcal{H}_{\Lambda,\Pi}, \mathcal{H}_{\mathcal{M},\mathcal{E}}\}$ or $\{\mathcal{H}_{\Lambda,\mathcal{E}}, \mathcal{H}_{\mathcal{M},\Pi}\}$ is going to be perfectly interpolated by definition while leaving the two other partitions with no concrete interpolation scheme. In this subsection, we present an alternative definition for the PLMs which allows perfect interpolation of all four data partitions.

So far, the base equation $\Gamma(s, \theta)$ which originated the LMs was defined with frequency partition Λ serving as the first partition and \mathcal{M} as the second partition. To denote this ordering of frequency partitions, we use the parametric Lagrange inverse matrix equation $\Gamma(s, \theta)$ and add the notation:

$$\Gamma(s, \theta) = \Gamma^{[\Lambda, \mathcal{M}]}(s, \theta) \quad (5.63)$$

This implies that $\Gamma(s, \theta)$ can be defined in three other ways:

- $\Gamma^{[\mathcal{M}, \Lambda]}$: \mathcal{M} is frequency partition 1, Λ is frequency partition 2
- $\Gamma^{[\Lambda, \Lambda]}$: Identical frequency partitions based on Λ
- $\Gamma^{[\mathcal{M}, \mathcal{M}]}$: Identical frequency partitions based on \mathcal{M}

As expected, $\Gamma^{[\Lambda, \Lambda]}$ and $\Gamma^{[\mathcal{M}, \mathcal{M}]}$ creates the special problem of having to deal with division by 0. We use $\Gamma^{[\Lambda, \Lambda]}$ as template to illustrate the problem. According to (5.15) to (5.18), the PLMs are built using the frequency LMs:

$$\begin{aligned} \Gamma_{\{k,l|i,j\}}^{[\Lambda, \Lambda]}(s, \theta) &= \theta \cdot s \cdot \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda, \Lambda]} - \theta \cdot \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda, \Lambda]} - s \cdot \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda, \Lambda]} + \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda, \Lambda]} \quad (5.64) \\ \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda, \Lambda]} &= \frac{\mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda, \Lambda]} - \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda, \Lambda]}}{\varepsilon_k - \pi_l}, \quad \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda, \Lambda]} = \frac{\sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda, \Lambda]} - \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda, \Lambda]}}{\varepsilon_k - \pi_l} \end{aligned}$$

$$p\mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} = \frac{\varepsilon_k \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\Lambda]} - \pi_l \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\Lambda]}}{\varepsilon_k - \pi_l}, \quad \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} \frac{\varepsilon_k \sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\Lambda]} - \pi_l \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\Lambda]}}{\varepsilon_k - \pi_l}$$

where the non-parametric LMs are generated using the inverse Legendre equation (2.52) that we rewrite here with the parameter specific indexing:

$$\gamma_{\{\theta_q|i,j\}}^{[\Lambda,\Lambda]}(s) = -s \mathbb{L}_{\{\theta_q|i,j\}}^{[\Lambda,\Lambda]} + \sigma \mathbb{L}_{\{\theta_q|i,j\}}^{[\Lambda,\Lambda]} \quad (5.65)$$

$$\gamma_{\{\theta_q|i,j\}}^{[\Lambda,\Lambda]}(s) = \frac{(s - \lambda_j) \mathbf{H}_{\lambda_j, \theta_q} + (\lambda_i - s) \mathbf{H}_{\lambda_i, \theta_q}}{\lambda_i - \lambda_j} \quad (5.66)$$

The immediate issue with (5.66) is when $j = i$, in which case we have a 0/0 situation. This allows a path forward through limit analysis by using the Hospital's rule, and we perform this limit analysis on a more general non-parametric LM pencil $\gamma_{\{y,y\}}$:

$$\begin{aligned} \gamma_{\{y,y\}}^{[\Lambda,\Lambda]}(s) &= \lim_{\lambda_x \rightarrow \lambda_y} \frac{(s - \lambda_y) \mathbf{H}(\lambda_y) + (\lambda_x - s) \mathbf{H}(\lambda_x)}{\lambda_x - \lambda_y} \\ &= \lim_{\lambda_x \rightarrow \lambda_y} \frac{d[(s - \lambda_y) \mathbf{H}(\lambda_y) + (\lambda_x - s) \mathbf{H}(\lambda_x)]/d\lambda_x}{d[\lambda_x - \lambda_y]/d\lambda_x} \quad < \text{Hospital's rule} > \\ &= \lim_{\lambda_x \rightarrow \lambda_y} \frac{d[(s - \lambda_y) \mathbf{H}(\lambda_y)]/d\lambda_x + d[(\lambda_x - s) \mathbf{H}(\lambda_x)]/d\lambda_x}{d\lambda_x/d\lambda_x - d\lambda_y/d\lambda_x} \\ &= \lim_{\lambda_x \rightarrow \lambda_y} \frac{d[(\lambda_x - s) \mathbf{H}(\lambda_x)]/d\lambda_x}{d\lambda_x/d\lambda_x} \\ &= \lim_{\lambda_x \rightarrow \lambda_y} \frac{d[\lambda_x \mathbf{H}(\lambda_x)]/d\lambda_x - d[s \mathbf{H}(\lambda_x)]/d\lambda_x}{d\lambda_x/d\lambda_x} \\ \gamma_{\{y,y\}}^{[\Lambda,\Lambda]}(s) &= \lim_{\lambda_x \rightarrow \lambda_y} \left[\mathbf{H}(\lambda_x) + \lambda_x \cdot \frac{d\mathbf{H}}{d\lambda_x}(\lambda_x) - s \cdot \frac{d\mathbf{H}}{ds}(\lambda_x) \right] \end{aligned}$$

Because λ_x is an arbitrary frequency value, a derivative with respect to it is technically no different from a derivative with respect to s , which is by definition an arbitrary frequency value. Thus, we can continue the derivation as follow:

$$\gamma_{\{y,y\}}^{[\Lambda,\Lambda]}(s) = \lim_{\lambda_x \rightarrow \lambda_y} \left[\mathbf{H}(\lambda_x) + \lambda_x \cdot \frac{d\mathbf{H}}{ds}(\lambda_x) - s \cdot \frac{d\mathbf{H}}{ds}(\lambda_x) \right]$$

$$\begin{aligned}\gamma_{\{y,y\}}^{[\Lambda,\Lambda]}(s) &= \mathbf{H}(\lambda_y) + \lambda_y \cdot \frac{d\mathbf{H}}{ds}(\lambda_y) - s \cdot \frac{d\mathbf{H}}{ds}(\lambda_y) \\ \gamma_{\{y,y\}}^{[\Lambda,\Lambda]}(s) &= \mathbf{H}_{\lambda_y} + \lambda_y \cdot \mathbf{H}'_{\lambda_y} - s \cdot \mathbf{H}'_{\lambda_y}\end{aligned}\quad (5.67)$$

Given the result (5.67), we have that the inverse Legendre equation has the more complete definition:

$$\gamma_{\{i,j\}}^{[\Lambda,\Lambda]}(s) = \begin{cases} \frac{(s - \lambda_j)\mathbf{H}_{\lambda_j} + (\lambda_i - s)\mathbf{H}_{\lambda_i}}{\lambda_i - \lambda_j} & \text{if } i \neq j \\ \mathbf{H}_{\lambda_z} + (\lambda_z - s) \cdot \mathbf{H}'_{\lambda_z} & \text{if } z = i = j \end{cases} \quad (5.68)$$

Thus, if we have identical entries between the two LM partitions Λ and \mathcal{M} , we would need the derivative of the data with respect to s . Expressing (5.68) in terms of general LMs, we have:

$$\gamma_{\{i,j\}}(s) = -s\mathbb{L}_{\{i,j\}} + \sigma\mathbb{L}_{\{i,j\}} \quad (5.69)$$

$$\mathbb{L}_{\{i,j\}} = \begin{cases} \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} & \text{if } \mu_i \neq \lambda_j \\ \mathbf{H}'_{s_z} & \text{if } s_z = \mu_i = \lambda_j \end{cases} \quad (5.70)$$

$$\sigma\mathbb{L}_{\{i,j\}} = \begin{cases} \frac{\mu_i\mathbf{H}_{\mu_i} - \lambda_j\mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} & \text{if } \mu_i \neq \lambda_j \\ \mathbf{H}_{s_z} + s_z\mathbf{H}'_{s_z} & \text{if } s_z = \mu_i = \lambda_j \end{cases} \quad (5.71)$$

With the definitions (5.70) and (5.71), we can go ahead and define the full PLMs $\Gamma^{[\Lambda,\Lambda]}$ and $\Gamma^{[\mathcal{M},\mathcal{M}]}$, though we do so under the assumption that we possess the derivative data of all our sample points.

Define the extended \mathbf{W} data vector matrix:

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{\Lambda,\Pi} & \mathbf{W}_{\mathcal{M},\Pi} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times \bar{m}\bar{l}(\bar{j}+\bar{i})} \quad (5.72)$$

$$\begin{aligned}
 \mathbf{W}_{\Lambda,\Pi} &= \begin{bmatrix} \mathbf{W}_{\Lambda,\pi_1} & \cdots & \mathbf{W}_{\Lambda,\pi_{\bar{l}}} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times \bar{m}\bar{l}\bar{j}} \\
 \mathbf{W}_{\Lambda,\pi_l} &= \begin{bmatrix} \mathbf{H}_{\lambda_1,\pi_l} & \mathbf{H}_{\lambda_2,\pi_l} & \cdots & \mathbf{H}_{\lambda_{\bar{j}},\pi_l} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times \bar{m}_j} \\
 \mathbf{W}_{\mathcal{M},\Pi} &= \begin{bmatrix} \mathbf{W}_{\mathcal{M},\pi_1} & \cdots & \mathbf{W}_{\mathcal{M},\pi_{\bar{l}}} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times \bar{m}\bar{l}\bar{i}} \\
 \mathbf{W}_{\mathcal{M},\pi_l} &= \begin{bmatrix} \mathbf{H}_{\mu_1,\pi_l} & \mathbf{H}_{\mu_2,\pi_l} & \cdots & \mathbf{H}_{\mu_{\bar{i}},\pi_l} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times \bar{m}_i}
 \end{aligned}$$

and the extended \mathbf{F} data vector matrix:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{\Lambda,\mathcal{E}} \\ \mathbf{F}_{\mathcal{M},\mathcal{E}} \end{bmatrix} \left\{ \begin{array}{l} \mathbf{F}_{\Lambda,\mathcal{E}} = \begin{bmatrix} \mathbf{F}_{\Lambda,\varepsilon_1} \\ \vdots \\ \mathbf{F}_{\Lambda,\varepsilon_{\bar{k}}} \end{bmatrix} \rightarrow \mathbf{F}_{\Lambda,\varepsilon_k} = \begin{bmatrix} \mathbf{H}_{\lambda_1,\varepsilon_k} \\ \vdots \\ \mathbf{H}_{\lambda_{\bar{j}},\varepsilon_k} \end{bmatrix} \\ \mathbf{F}_{\mathcal{M},\mathcal{E}} = \begin{bmatrix} \mathbf{F}_{\mathcal{M},\varepsilon_1} \\ \vdots \\ \mathbf{F}_{\mathcal{M},\varepsilon_{\bar{k}}} \end{bmatrix} \rightarrow \mathbf{F}_{\mathcal{M},\varepsilon_k} = \begin{bmatrix} \mathbf{H}_{\mu_1,\varepsilon_k} \\ \vdots \\ \mathbf{H}_{\mu_{\bar{i}},\varepsilon_k} \end{bmatrix} \end{array} \right. \quad (5.73)$$

Finally, define the full 4-quadrant PLM:

$$\boldsymbol{\Gamma}(s, \theta) = \begin{bmatrix} \boldsymbol{\Gamma}^{[\Lambda,\mathcal{M}]}(s, \theta) & \boldsymbol{\Gamma}^{[\mathcal{M},\mathcal{M}]}(s, \theta) \\ \boldsymbol{\Gamma}^{[\Lambda,\Lambda]}(s, \theta) & \boldsymbol{\Gamma}^{[\mathcal{M},\Lambda]}(s, \theta) \end{bmatrix} \quad (5.74)$$

where the base building block of each quadrant are defined as:

$$\begin{aligned}
 \boldsymbol{\Gamma}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]}(s, \theta) &= \theta \cdot s \cdot \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]} - \theta \cdot \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]} - s \cdot \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]} + \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]} \quad (5.75) \\
 \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]} &= \frac{\mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\mathcal{M}]} - \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\mathcal{M}]}}{\varepsilon_k - \pi_l}, \quad \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]} = \frac{\sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\mathcal{M}]} - \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\mathcal{M}]}}{\varepsilon_k - \pi_l} \\
 p \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]} &= \frac{\varepsilon_k \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\mathcal{M}]} - \pi_l \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\mathcal{M}]}}{\varepsilon_k - \pi_l}, \quad \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\mathcal{M}]} = \frac{\varepsilon_k \sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\mathcal{M}]} - \pi_l \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\mathcal{M}]}}{\varepsilon_k - \pi_l}
 \end{aligned}$$

$$\mathbb{L}_{\{\theta_q|i,j\}}^{[\Lambda,\mathcal{M}]} = \frac{\mathbf{H}_{\mu_i,\theta_q} - \mathbf{H}_{\lambda_j,\theta_q}}{\mu_i - \lambda_j}, \quad \sigma \mathbb{L}_{\{\theta_q|i,j\}}^{[\Lambda,\mathcal{M}]} = \frac{\mu_i \mathbf{H}_{\mu_i,\theta_q} - \lambda_j \mathbf{H}_{\lambda_j,\theta_q}}{\mu_i - \lambda_j}$$

$$\begin{aligned} \mathbf{\Gamma}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]}(s,\theta) &= \theta \cdot s \cdot \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]} - \theta \cdot \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]} - s \cdot \tau \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]} + \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]} \quad (5.76) \\ \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]} &= \frac{\mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\mathcal{M},\Lambda]} - \mathbb{L}_{\{\pi_l|i,j\}}^{[\mathcal{M},\Lambda]}}{\varepsilon_k - \pi_l}, \quad \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]} = \frac{\sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\mathcal{M},\Lambda]} - \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\mathcal{M},\Lambda]}}{\varepsilon_k - \pi_l} \\ p \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]} &= \frac{\varepsilon_k \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\mathcal{M},\Lambda]} - \pi_l \mathbb{L}_{\{\pi_l|i,j\}}^{[\mathcal{M},\Lambda]}}{\varepsilon_k - \pi_l}, \quad \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\Lambda]} = \frac{\varepsilon_k \sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\mathcal{M},\Lambda]} - \pi_l \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\mathcal{M},\Lambda]}}{\varepsilon_k - \pi_l} \\ \mathbb{L}_{\{\theta_q|i,j\}}^{[\mathcal{M},\Lambda]} &= \frac{\mathbf{H}_{\lambda_i,\theta_q} - \mathbf{H}_{\mu_j,\theta_q}}{\lambda_i - \mu_j}, \quad \sigma \mathbb{L}_{\{\theta_q|i,j\}}^{[\mathcal{M},\Lambda]} = \frac{\lambda_i \mathbf{H}_{\lambda_i,\theta_q} - \mu_j \mathbf{H}_{\mu_j,\theta_q}}{\lambda_i - \mu_j} \end{aligned}$$

$$\begin{aligned} \mathbf{\Gamma}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]}(s,\theta) &= \theta \cdot s \cdot \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]} - \theta \cdot \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]} - s \cdot \tau \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]} + \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]} \quad (5.77) \\ \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]} &= \frac{\mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\mathcal{M},\mathcal{M}]} - \mathbb{L}_{\{\pi_l|i,j\}}^{[\mathcal{M},\mathcal{M}]}}{\varepsilon_k - \pi_l}, \quad \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]} = \frac{\sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\mathcal{M},\mathcal{M}]} - \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\mathcal{M},\mathcal{M}]}}{\varepsilon_k - \pi_l} \\ p \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]} &= \frac{\varepsilon_k \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\mathcal{M},\mathcal{M}]} - \pi_l \mathbb{L}_{\{\pi_l|i,j\}}^{[\mathcal{M},\mathcal{M}]}}{\varepsilon_k - \pi_l}, \quad \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\mathcal{M},\mathcal{M}]} = \frac{\varepsilon_k \sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\mathcal{M},\mathcal{M}]} - \pi_l \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\mathcal{M},\mathcal{M}]}}{\varepsilon_k - \pi_l} \\ \mathbb{L}_{\{\theta_q|i,j\}}^{[\mathcal{M},\mathcal{M}]} &= \left\{ \begin{array}{ll} \frac{\mathbf{H}_{\mu_i,\theta_q} - \mathbf{H}_{\mu_j,\theta_q}}{\mu_i - \mu_j} & \text{if } i \neq j \quad \frac{\mu_i \mathbf{H}_{\mu_i,\theta_q} - \mu_j \mathbf{H}_{\mu_j,\theta_q}}{\mu_i - \mu_j} \\ \mathbf{H}'_{\mu_i,\theta_q} & \text{if } i = j \quad \mathbf{H}_{\mu_i,\theta_q} + \mu_i \mathbf{H}'_{\mu_i,\theta_q} \end{array} \right\} = \sigma \mathbb{L}_{\{\theta_q|i,j\}}^{[\mathcal{M},\mathcal{M}]} \end{aligned}$$

$$\begin{aligned} \mathbf{\Gamma}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]}(s,\theta) &= \theta \cdot s \cdot \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} - \theta \cdot \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} - s \cdot \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} + \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} \quad (5.78) \\ \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} &= \frac{\mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\Lambda]} - \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\Lambda]}}{\varepsilon_k - \pi_l}, \quad \sigma \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} = \frac{\sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\Lambda]} - \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\Lambda]}}{\varepsilon_k - \pi_l} \\ p \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} &= \frac{\varepsilon_k \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\Lambda]} - \pi_l \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\Lambda]}}{\varepsilon_k - \pi_l}, \quad \sigma \tau \mathbb{T}_{\{k,l|i,j\}}^{[\Lambda,\Lambda]} = \frac{\varepsilon_k \sigma \mathbb{L}_{\{\varepsilon_k|i,j\}}^{[\Lambda,\Lambda]} - \pi_l \sigma \mathbb{L}_{\{\pi_l|i,j\}}^{[\Lambda,\Lambda]}}{\varepsilon_k - \pi_l} \\ \mathbb{L}_{\{\theta_q|i,j\}}^{[\Lambda,\Lambda]} &= \left\{ \begin{array}{ll} \frac{\mathbf{H}_{\lambda_i,\theta_q} - \mathbf{H}_{\lambda_j,\theta_q}}{\lambda_i - \lambda_j} & \text{if } i \neq j \quad \frac{\lambda_i \mathbf{H}_{\lambda_i,\theta_q} - \lambda_j \mathbf{H}_{\lambda_j,\theta_q}}{\lambda_i - \lambda_j} \\ \mathbf{H}'_{\lambda_j,\theta_q} & \text{if } i = j \quad \mathbf{H}_{\lambda_j,\theta_q} + \lambda_j \mathbf{H}'_{\lambda_j,\theta_q} \end{array} \right\} = \sigma \mathbb{L}_{\{\theta_q|i,j\}}^{[\Lambda,\Lambda]} \end{aligned}$$

Note that the derivative terms for (5.77) and (5.78) are partial derivatives with respect to s only. If we use the same proof process utilized for Theorem 2.2.1 and 5.2.1, we can show that the transfer function

$$\mathbf{G}(s, \theta) = \mathbf{W} [\boldsymbol{\Gamma}(s, \theta)]^{-1} \mathbf{F} \quad (5.79)$$

achieves complete initial data interpolation coverage:

$$\begin{aligned} \mathbf{G}(\lambda_j, \pi_l) &= \mathbf{H}_{\lambda_j, \pi_l} \quad \forall \lambda_j \in \Lambda, \forall \pi_l \in \Pi \\ \mathbf{G}(\mu_i, \pi_l) &= \mathbf{H}_{\mu_i, \pi_l} \quad \forall \mu_i \in \mathcal{M}, \forall \pi_l \in \Pi \\ \mathbf{G}(\lambda_j, \varepsilon_k) &= \mathbf{H}_{\lambda_j, \varepsilon_k} \quad \forall \lambda_j \in \Lambda, \forall \varepsilon_k \in \mathcal{E} \\ \mathbf{G}(\mu_i, \varepsilon_k) &= \mathbf{H}_{\mu_i, \varepsilon_k} \quad \forall \mu_i \in \mathcal{M}, \forall \varepsilon_k \in \mathcal{E} \end{aligned} \quad (5.80)$$

Although the 4-quadrant PLM transfer function (5.79) achieves perfect initial data interpolation, it also will go through the same MOR process described in Section 5.2.4 of the 1-quadrant PLM transfer function, effectively nullifying the perfect interpolation property. As the post-model order reduction systems from either the 1-quadrant or 4-quadrant PLM transfer function do not achieve significantly different results and because the 4-quadrant PLMs double the initial system size as well as require highly accurate derivative data, we will not utilize the 4-quadrant formulation in the simulation section and keep it as a proof of concept only.

5.3 Simulation

Here is the simulation section dedicated to demonstrate the capability of the PSFLM method in parametric modeling. As the PSFLM method was developed after establishing the PBRLM method, the simulations performed in this section

focuses on comparing PSFLM against PBRLM in terms of performance of parametric macromodeling. A number of properties that can be used for comparison are:

1. Computation cost of the macromodels.
2. Accuracy of the macromodels.
3. Stability of the macromodels.

Even without going into the simulation results, one can reliably predict that the PBRLM method holds a tremendous advantage over the PSFLM method in terms of the computational cost of the macromodels. This is because the bottleneck for the PBRLM method is the SVD of a $\mathbb{C}^{\bar{p}\bar{m}\bar{q}\bar{l}\times\bar{y}\bar{l}}$ matrix while the bottleneck for the PSFLM method is the SVD of a $\mathbb{C}^{\bar{p}\bar{k}\bar{l}\times\bar{m}\bar{l}\bar{l}}$ matrix. Given that the number of polynomials \bar{y} for the PBRLM method is generally far smaller than the product of the size \bar{l} of parameter partition Π with the number of outputs \bar{m} , the LM for PBRLM is far slimmer than that of PSFLM, incurring a far better cost scaling. If we then take into account the efficient SVD technique from Section 4.3.2 available to the PBRLM method due to the fact we only need the right singular vectors for the PBRLM method, it should be clear that the PSFLM method poses no contest in terms of computation cost. As consequence, the present simulation section focuses on comparing accuracy and stability of the parametric models generated.

Given the stated objective of accuracy and stability comparisons, the parametric models generated from the examples of this section are optimized for both the PBRLM and PSFLM methods towards achieving highest accuracy while maintaining as wide a stability band over the θ domain as possible.

5.3.1 Parametric Data Smoothing Through Polynomial Regression

For black-box macromodeling in general, and in the simulation section 5.3 in particular, the data are generated using full-wave simulators, which implies some form of discretization of Maxwell's equations in space. During a sweep of a geometric parameter a new mesh is often required for each parameter value $\theta_q \in \Theta$. This

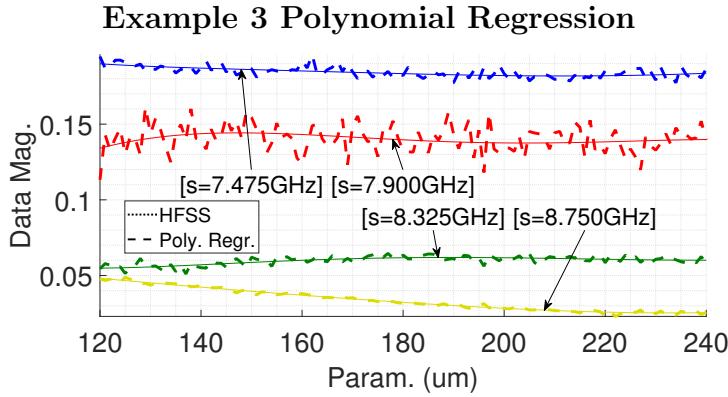


Figure 5.4: Example 3 $S_{1,2}$ data regression over the parameter domain. Each line is at a different sample frequency point.

typically results in non-smooth behavior if we plot the results as a function of the parameter at a given frequency value as shown in Figure 5.4. Despite the fact that these discretization errors are relatively small, the PSFLM method is sensitive to non-smooth variations over the parameter domain. As a result, linear regression is first used to smooth the data over the parameter domain and then pass the post-processed data to the PLM methods. The regression method used for examples in Section 5.3 is the basic polynomial regression method. The polynomial regression is applied on the data over the θ domain at one frequency point at a time with the number of polynomials locked within the same example. The polynomial basis specifically used for Figure 5.4 utilized a polynomial of order up to 4.

If polynomial regression is applied to an example in Section 5.3, the approximation error of the parametric models will be calculated with respect to the polynomial regression data rather than the original full-wave simulation data because the parametric model is based on the former rather than the latter.

5.3.2 Example 1

The example is a simple microstripline example which is illustrated in Figure 5.5. The dielectric substrate has the dimensions: 50 mm \times 10 mm \times 1 mm. The copper

Example 1 Schematic

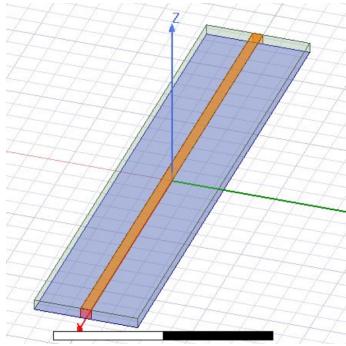


Figure 5.5: Simple mstrip schematic

trace has a width of 1 mm and a 20 μm thickness. The two ports are defined at the two ends of the trace with reference to a perfect conductor ground covering the bottom surface of the substrate. The parameter of variation is the permittivity of the substrate. Polynomial regression is used to process the HFSS raw data with a polynomial basis of degrees up to 3.

The initial parametric data set consists of 401 s points and 121 θ points linearly distributed over the ranges of [0, 8]GHz and [3.8, 4.4] F/m , respectively. Only 100 s and 30 θ points, linearly sampled from the initial set, are used to construct the

PLMs. When applying method- α , 8 linearly distributed s points over the set are selected to be in partition Λ , which dictates the system order to be 30 at all θ , and the remaining 92 s points are left to partition \mathcal{M} . The polynomial basis consists of Legendre polynomials with maximum polynomial order 3. When applying method- β , we have equal s partitions of 50 points and equal θ partitions of 15 points created by interleaving index sampling. The SVD cut-off index of the pencil $\Gamma(\theta, s)$, which is the final system order, is 62.

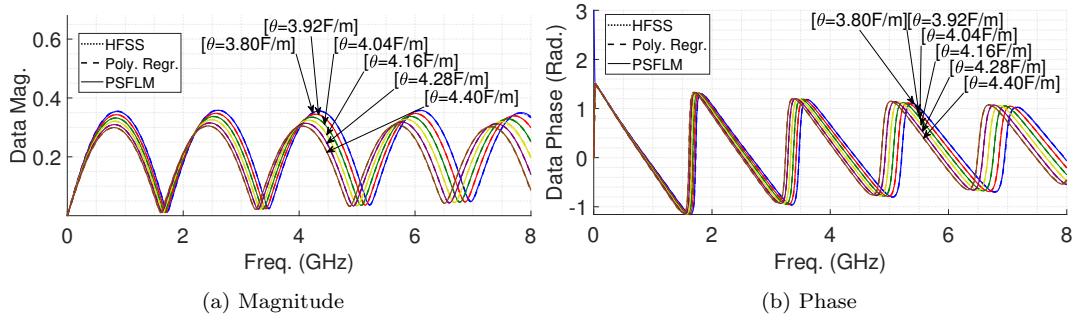


Figure 5.6: Ex. 1 Sample $S_{1,1}$ magnitude and phase plots

On Figure 5.6a and Figure 5.6b are plotted the $S_{1,1}$ magnitudes and phases at select number of parameter points. The sample parameters points are selected using linear distribution over the continuous range of parameters where the PSFLM method achieved stability. In both plots, we can see accurate matching of data between the three data origins: HFSS, polynomial regression, and PSFLM approximation.

The overall numerical results of the parametric models are given on Table 5.1. The accuracy, represented by the absolute RMS error, reaches 5.01e-4 by the PSFLM model and only 9.55e-4 by the PBRLM model. Furthermore, the continuous stability band achieved by PSFLM is far wider than the one achieved by PBRLM. The stability profiles can be better visualized on Figure 5.7 where the maximum poles

Table 5.1: Ex. 1 Simulation Results

	PBRLM	PSFLM
Comp. Time (SVD)	0.04s	13s
Longest Stab. Band	[4.02, 4.40]	[3.80, 4.40]
Poly. Regr. Error	7.520e-4	
Absolute RMS error	9.55e-4	5.01e-4
Abs. RMS err. over longest stab. band	8.81e-4	5.01e-4

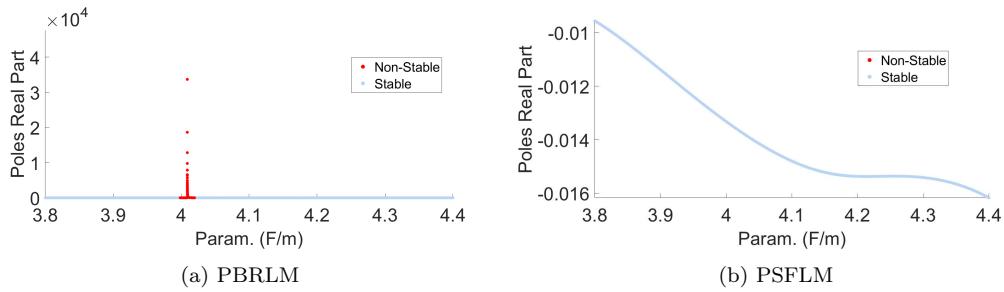


Figure 5.7: Ex. 1 poles maximum real part plot

real part are plotted over the θ domain. We note the asymptotic non-stable zone in Figure 5.7a, which highlights one of the key problems of the PBRLM method: severe non-stability may appear in a non-predictable way well within the parameter region of approximation. On the other hand, Figure 5.7b demonstrate full initial parameter band stability by the PSFLM method. The stability profiles can also be observed in Figure 5.8 where the RMS error with respect to s domain at each individual θ point are plotted for both methods. As can be seen from Figure 5.8a, the RMS error peaks at where the asymptotic non-stability occurs in the PBRLM model whereas the error more gradually degrade as we approach the extremities of the θ domain of approximation on Figure 5.8b for the PSFLM model.

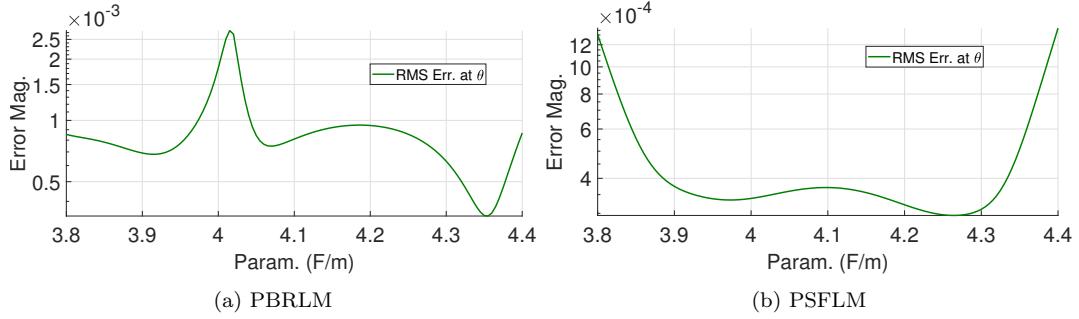


Figure 5.8: Ex. 1 RMS error w.r.t. freq. domain plot.

5.3.3 Example 2

The example, illustrated in Figure 5.9, is a section of a PCB package where merger of two differential pairs of traces are merged into one pair. The schematic is found as an example design project of the Ansys HFSS 3D Layout software. The defined parameter of the parametric data extraction is the width of the traces connecting the vias throughout the PCB. Two pairs of ports (port1, port2) and (port3, port4) are defined as differential port pairs located at the ends of the board. The schematic is shown in Figure 5.9.

Example 2 Schematic

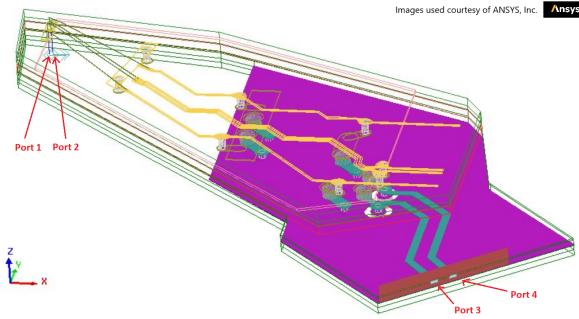


Figure 5.9: Printed circuit board package merging section schematic

port4) are defined as differential port pairs located at the ends of the board. Polynomial regression is used to process the HFSS raw data with a polynomial basis of degrees up to 5.

The initial parametric data set consists of 500 s points and 61 θ points linearly distributed over the range of [0, 10]GHz and [20, 80]μm, respectively. Only 100 s and 30 θ points, linearly sampled from the initial set, are used to construct the PLMs.

When applying method- α , 7 linearly distributed s points over the set are selected to be in partition Λ , which dictates the system order to be 52 at all θ , and the remaining 93 s points are left to partition \mathcal{M} . The polynomial basis consists of Legendre polynomials with maximum polynomial order 4. When applying method- β , we have equal s partitions of 50 points and equal θ partitions of 15 points created by interleaving index sampling. The SVD cut-off index of the pencil $\mathbf{\Gamma}(\theta, s)$, which is the final system order, is 64.

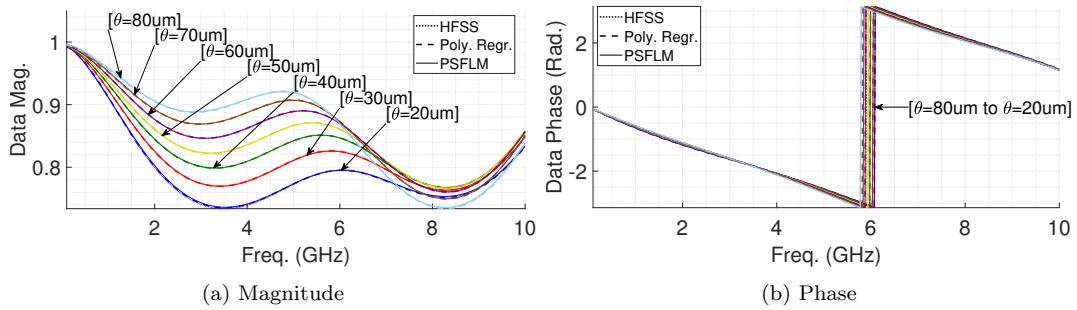


Figure 5.10: Ex. 2 Sample $S_{2,3}$ magnitude and phase plots

On Figure 5.10a and Figure 5.10b are plotted the $S_{2,3}$ magnitudes and phases at select number of parameter points. The sample parameters points are selected using linear distribution over the continuous range of parameters where the PSFLM method achieved stability. In both plots, we can see accurate matching of data between the three data origins: HFSS, polynomial regression, and PSFLM approximation.

The overall numerical results of the parametric models are given on Table 5.2. The accuracy, represented by the absolute RMS error, reaches 6.48e-4 by the PBRLM

Table 5.2: Ex. 2 Simulation Results

	PBRLM	PSFLM
Comp. Time (SVD)	0.01s	111s
Longest Stab. Band	[20, 80]	[20, 80]
Poly. Regr. Error	1.661e-3	
Absolute RMS error	6.48e-4	2.32e-4
Abs. RMS err. over longest stab. band	6.48e-4	2.32e-4

model and the lower error of 2.32e-4 by the PSFLM model. For this example, the continuous stability bands achieved by both PSFLM and PBRLM cover the full initial θ range. The stability profiles can be better visualized on Figure 5.11 where

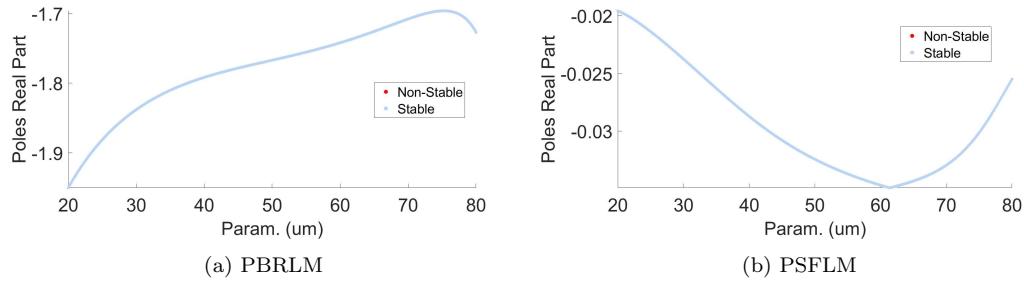


Figure 5.11: Ex. 2 poles maximum real part plot

the maximum poles real part are plotted over the θ domain. In echo to Table 5.2 results, we see complete stability band achieved by both methods. On Figure 5.8 is plotted the RMS error with respect to s domain at each individual θ point are plotted for both methods. From the figures, we can see similar pattern where the error is the lowest at the center of the θ band while gradually rising as we approach the limits of the initial band. All-in-all, this example demonstrate how both methods can be valid parametric macromodeling methods in terms of achieving accuracy and stability.

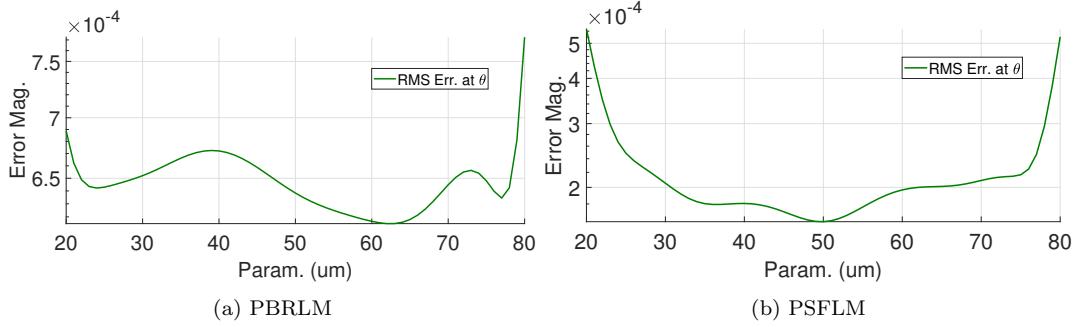


Figure 5.12: Ex. 2 RMS error w.r.t. freq. domain plot.

5.3.4 Example 3

The example, illustrated in Figure 5.13, is a differential via design found in the Ansys HFSS 3D Layout software as a template design project. The via design comprises of interleaving layers of copper and FR4 epoxy substrate. The defined parameter of the parametric data extraction is the width of the 4 traces bridging the two center vias to the 4 ports located at the edges of the design area. In this

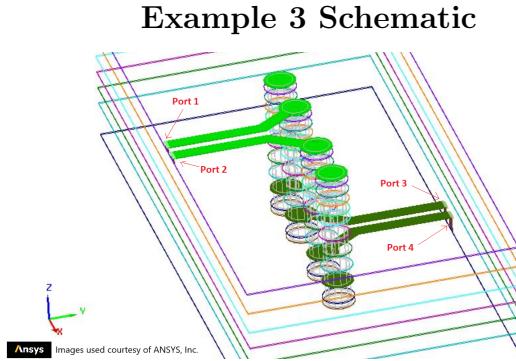


Figure 5.13: Differential vias basic design schematic

example we define port 1, port 2, port 3, and port 4 as four independent ports rather than two differential pairs. Polynomial regression is used to pre-process the HFSS raw data with a polynomial basis of degrees up to 4.

The initial parametric data set consists of 401 frequency points s and 121 parameter points θ linearly distributed over the range of [0, 10]GHz and [120, 240]μm, respectively. Only 100 s points and 30 θ points, linearly sampled from the initial set, are used to construct the PBRLM and the proposed PSFLM models.

When constructing the PBRLM model, 8 linearly distributed s points over the set are selected to be in partition Λ , which dictates the system order to be 60 at all θ , and the remaining 92 s points are left to partition \mathcal{M} . The polynomial basis consists of Legendre polynomials with maximum polynomial order 5. When constructing the PSFLM model, we have equal s partitions of 50 points and equal θ partitions of 15 points created by interleaving index sampling. The SVD cut-off index of the pencil $\Gamma(\theta, s)$, which is the final system order, is 120.

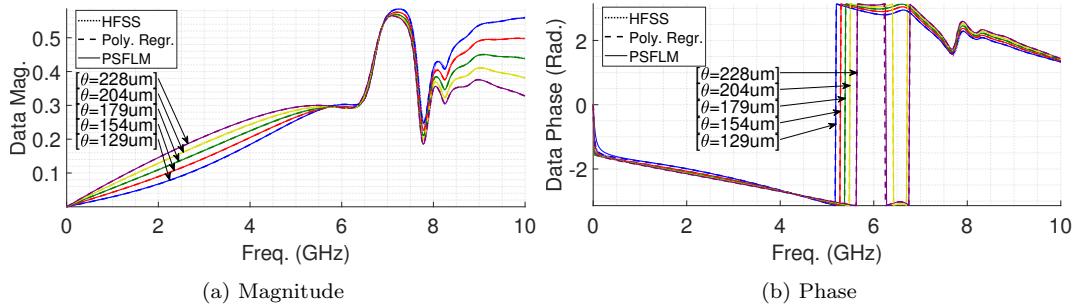


Figure 5.14: Ex. 3 Sample $S_{2,2}$ magnitude and phase plots

On Figure 5.14 are plotted the $S_{2,2}$ magnitudes and phases at select number of parameter points. The sample parameters points are selected using linear distribution over the continuous range of parameters where the PSFLM method achieved stability. In both plots, we can see accurate matching of data between the three data origins: HFSS, polynomial regression, and PSFLM approximation.

Table 5.3: Ex. 3 Simulation Results

	PBRLM	PSFLM
Comp. Time (SVD)	0.01s	124s
Longest Stab. Band	[178.5, 240.0] μm	[128.4, 228.5] μm
Poly. Regr. Error	3.567e-3	
Absolute RMS error	1.44e-2	6.14e-4
Abs. RMS err. over longest stab. band	1.44e-2	4.76e-4

The overall numerical results of the parametric models are given on Table 5.3. The accuracy, represented by the absolute RMS error, reaches 6.14e-4 by the PSFLM model and only 1.44e-2 by the PBRLM model. Furthermore, the continuous stability band achieved by PSFLM is far wider than the one achieved by PBRLM. The

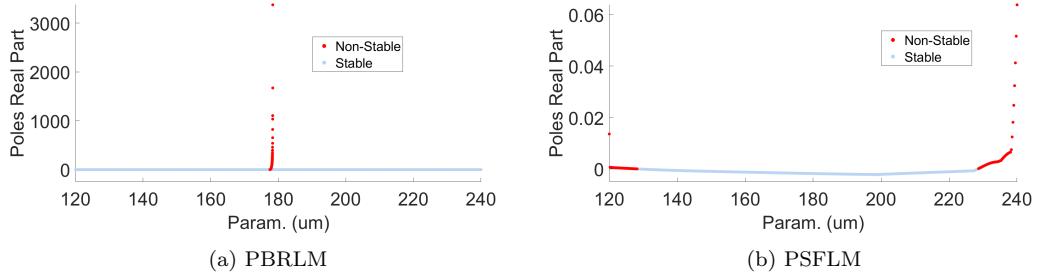


Figure 5.15: Ex. 3 poles maximum real part plot

stability profiles can be better visualized on Figure 5.15 where the maximum poles real part are plotted over the θ domain. We note the asymptotic non-stable zone in Figure 5.15a, which once again highlights the problems of severe non-stability that may be produced by the PBRLM method well within the parameter region of approximation. On the other hand, Figure 5.15b demonstrate a more gradual lost of stability as we approach the border of the θ domain, demonstrating a more controlled stability profile by the PSFLM method. The more controlled profile is also observed

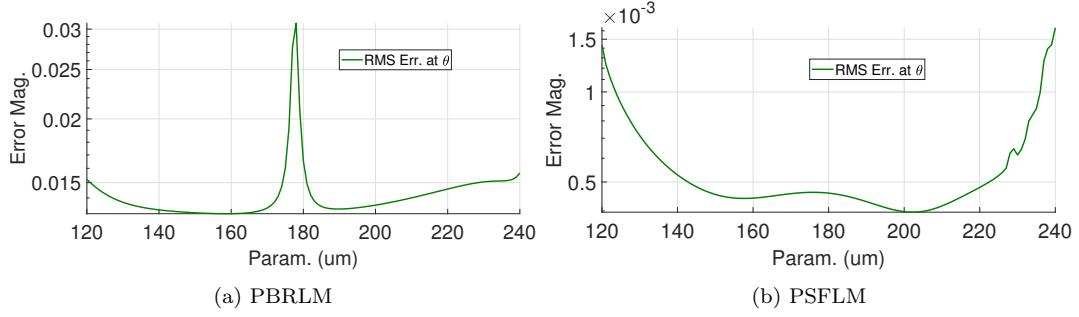


Figure 5.16: Ex. 3 RMS error w.r.t. freq. domain plot.

in Figure 5.16 where the RMS error with respect to s domain at each individual θ point are plotted for both methods. As can be seen from Figure 5.16a, the RMS error peaks at where the asymptotic non-stability occurs in the PBRLM model whereas the error more gradually degrade as we approach the extremities of the θ domain of approximation on Figure 5.16b for the PSFLM model.

5.3.5 Example 4

The example, illustrated in Figure 5.17, is a package section design coming from the Ansys HFSS software as a template design project. The defined parameter of the parametric data extraction is the permittivity of the FR4 epoxy substrate encasing the whole package. The design defines 4 independent ports, with port 1 and 2 located

Example 4 Schematic

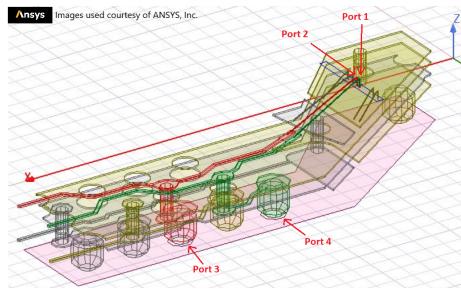


Figure 5.17: Package section schematic

at the top of the package defined between the copper wires and the perfect conductor plate supporting the copper wires. The two remaining ports are defined between the legs of two ball shaped vias and the floor level perfect conductor plate. Polynomial regression is used to process the HFSS raw data with a polynomial basis of degrees up to 5.

The initial parametric data set consists of 401 s points and 81 θ points linearly distributed over the range of [0, 8]GHz and [4.2, 5.0]F/m, respectively. The PLMs are constructed with a subset of data sampled from the initial set, with 32 θ points linearly sampled and 401 s points linearly sampled with higher density over the s range: [1.48GHz, 1.78GHz].

When applying method- α , 9 linearly distributed s points over the set are selected to be in partition Λ , which dictates the system order to be 68 at all θ , and the remaining 106 s points are left to partition \mathcal{M} . The polynomial basis consists of Legendre polynomials with maximum polynomial order 6. When applying method- β , we have s partitions of 58 and 57 points and equal θ partitions of 16 points created by interleaving index sampling. The SVD cut-off index of the pencil $\Gamma(\theta, s)$, which is the final system order, is 143.

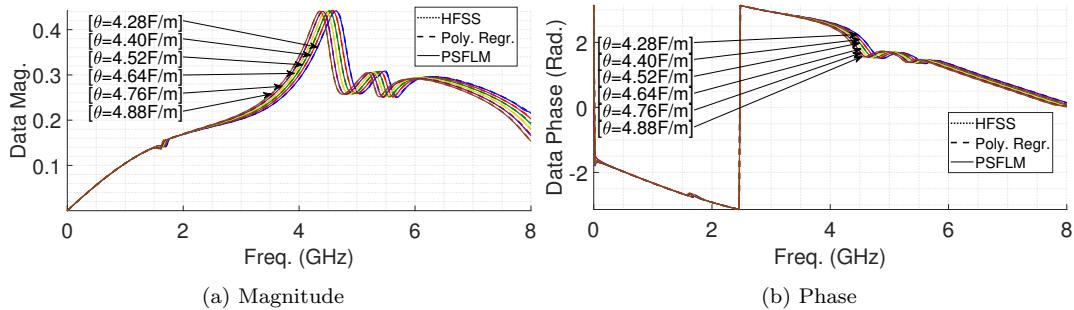


Figure 5.18: Ex. 4 Sample $S_{1,3}$ magnitude and phase plots

Table 5.4: Ex. 4 Simulation Results

	PBRLM	PSFLM
Comp. Time (SVD)	0.02s	140s
Longest Stab. Band	[4.38, 4.82]	[4.28, 4.89]
Poly. Regr. Error		1.936e-3
Absolute RMS error	2.11e-3	1.32e-3
Abs. RMS err. over longest stab. band	1.33e-3	8.14e-4

On Figure 5.18 are plotted the $S_{1,3}$ magnitudes and phases at select number of parameter points. The sample parameters points are selected using linear distribution over the continuous range of parameters where the PSFLM method achieved stability. In both plots, we can see accurate matching of data between the three data origins: HFSS, polynomial regression, and PSFLM approximation.

The overall numerical results of the parametric models are given on Table 5.4. The accuracy, represented by the absolute RMS error, reaches 1.32e-3 by the PSFLM model and only 2.11e-3 by the PBRLM model. Furthermore, the continuous stability band achieved by PSFLM is far wider than the one achieved by PBRLM. The

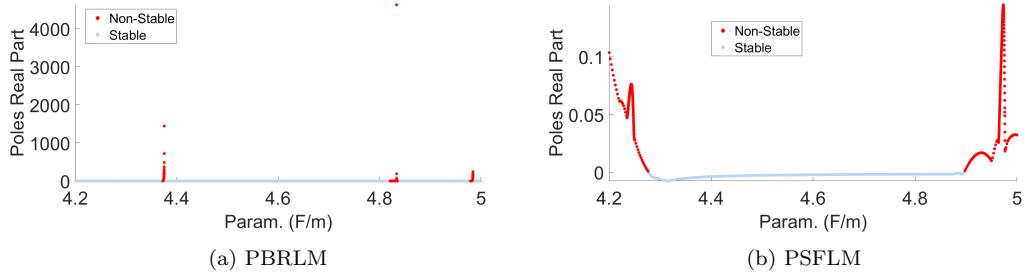


Figure 5.19: Ex. 4 poles maximum real part plot

stability profiles can be better visualized on Figure 5.19 where the maximum poles real part are plotted over the θ domain. We note the asymptotic non-stable zones

in Figure 5.19a, which once again highlight the problem of method PBRLM introducing severe instability. On the other hand, Figure 5.19b demonstrate a more gradual lost of stability as we approach the border of the θ domain, demonstrating a more controlled stability profile by the PSFLM method. The more controlled profile is

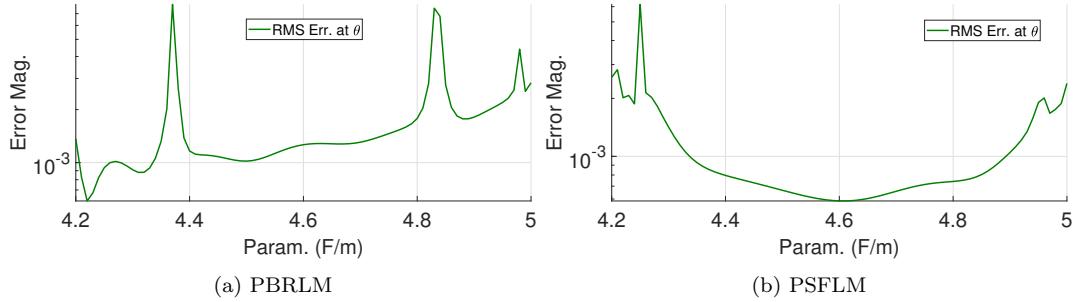


Figure 5.20: Ex. 4 RMS error w.r.t. freq. domain plot.

also observed in Figure 5.20 where the RMS error with respect to s domain at each individual θ point are plotted for both methods. As can be seen from Figure 5.20a, the RMS error peaks at where the asymptotic non-stability occurs in the PBRLM model whereas the error more gradually degrade as we approach the extremities of the θ domain of approximation on Figure 5.20b for the PSFLM model.

5.3.6 Example 5

The example, illustrated in Figure 5.21, is a simple microstripline design consisting of three parallel traces with a 90 degree turn at the middle point. The defined parameter of the parametric data extraction is the width of the traces. The design defines 6 ports, 1 at each ends of each trace with the reference being the perfect conductor plane at the bottom of the substrate. Polynomial regression is used to process the HFSS raw data with a polynomial basis of degrees up to 4.

Example 5 Schematic

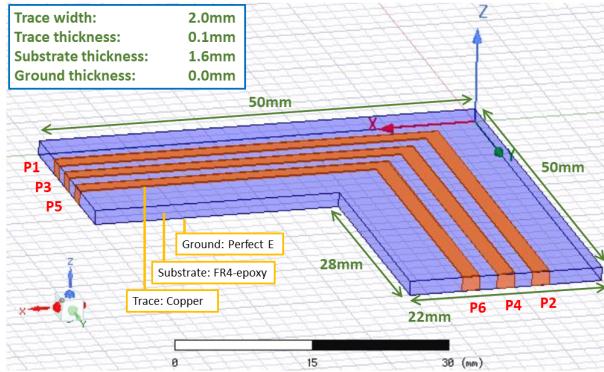


Figure 5.21: Package section schematic

The initial parametric data set consists of 500 frequency points s points and 81 parameter points θ points linearly distributed over the range of [0, 10]GHz and [1.8, 2.2]mm, respectively. Only 100 s points and 30 θ points, linearly sampled from the initial set, are used to construct the PBRLM and the proposed PSFLM models.

When applying the PBRLM method, 21 linearly distributed s points over the set are selected to be in partition Λ , which dictates the system order to be 246 at all θ , and the remaining 79 s points are left to partition \mathcal{M} . The polynomial basis consists of Legendre polynomials with maximum polynomial order 4. When applying the PSFLM method, we have equal s partitions of 50 points and equal θ partitions of 15 points created by interleaving index sampling. The SVD cut-off index of the pencil $\Gamma(\theta, s)$, which is the final system order, is 225.

On Figure 5.22 are plotted the $S_{1,1}$ magnitudes and phases at select number of parameter points. The sample parameters points are selected using linear distribution over initial parameter range. In both plots, we can see accurate matching

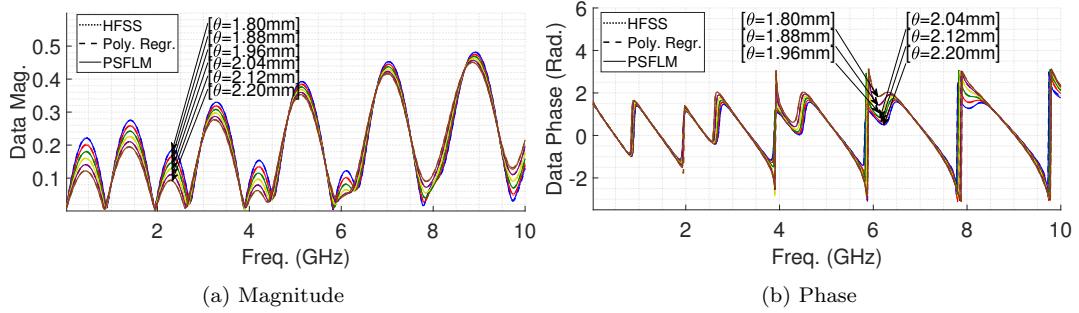

 Figure 5.22: Ex. 5 Sample $S_{1,1}$ magnitude and phase plots

Table 5.5: Ex. 5 Simulation Results

	PBRLM	PSFLM
Comp. Time (SVD)	0.2s	332s
Longest Stab. Band	[1.810, 2.105]mm	[1.800, 2.200]mm
Poly. Regr. Error	3.335e-3	
Absolute RMS error	1.39e-3	7.14e-4
Abs. RMS err. over longest stab. band	1.28e-3	7.14e-4

of data between the three data origins: HFSS, polynomial regression, and PSFLM approximation.

The overall numerical results of the parametric models are given on Table 5.5. The accuracy, represented as the absolute RMS error, reaches 7.14e-4 by the PSFLM method and reaches the relatively higher value of 1.39e-3 by the PBRLM method. Furthermore, the PSFLM model achieves full θ band stability as can be observed on Figure 5.23b whereas the PBRLM model contains a couple of asymptotic non-stable regions over the θ band as can be observed on Figure 5.23a. As can be seen, the PBRLM method introduced severe non-stability in a seemingly arbitrary way over the θ range of approximation. In plotting the RMS error with respect to s domain at the θ points in Figure 5.24, we observe a similar pattern in which we

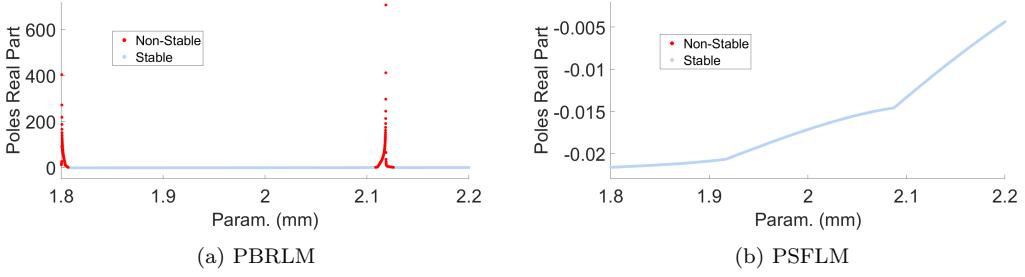


Figure 5.23: Ex. 5 poles maximum real part plot

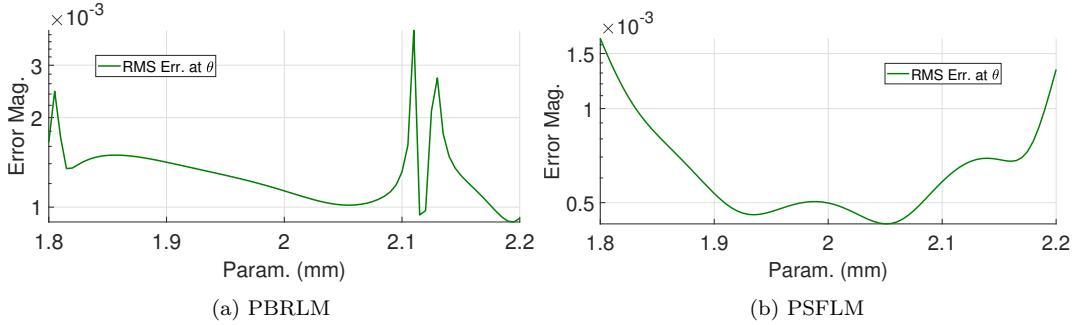


Figure 5.24: Ex. 5 RMS error w.r.t. freq. domain plot.

have for the PBRLM model error spikes located at where the asymptotic non-stable regions occurred and for the PSFLM model a gradual degradation of accuracy at the extremities of the θ band.

5.3.7 Example 6

The example, illustrated in Figure 5.25, is a package example that can be found as an example design project of the Ansys HFSS 3D Layout software. The defined parameter of the parametric data extraction is the permittivity of the substrate block. The design defines 8 independent ports, with ports 1 to 4 located at the left end of the package and the ports 5 to 8 in contact to the legs of ball shaped vias on the right side of the package. As a breakaway case, the HFSS data is used directly by

Example 6 Schematic

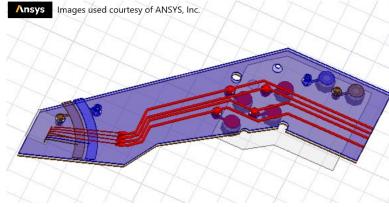


Figure 5.25: Package example schematic

both methods without going through polynomial regression. This was made possible because the parameter does not change the physical dimensions of the design which permits the software to maintain the same meshing throughout the parametric data generation.

The initial parametric data set consists of 401 frequency points s points and parameter points 65 θ points linearly distributed over the range of [0, 10]GHz and [4.8, 5.6] F/m , respectively. The PBRLM and the PSFLM models are constructed with a subset of data sampled from the initial set, with 33 θ points linearly sampled and 128 s points linearly sampled with higher density over the s range: [1.225, 1.675]GHz.

When applying the PBRLM method, 15 linearly distributed s points over the set are selected to be in partition Λ , which dictates the system order to be 232 at all θ , and the remaining 113 s points are left to partition \mathcal{M} . The polynomial basis consists of Legendre polynomials with maximum polynomial order 8. When applying the PSFLM method, we have equal s partitions of 64 points and θ partitions of 17 and 16 points created by interleaving index sampling. The SVD cut-off index of the pencil $\Gamma(\theta, s)$, which is the final system order, is 150.

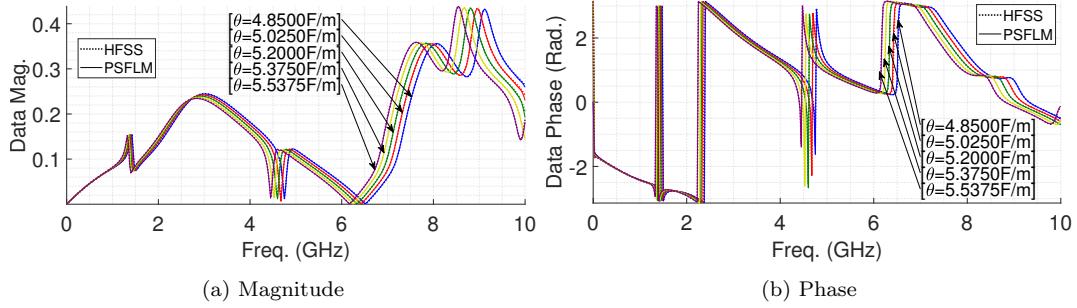

 Figure 5.26: Ex. 6 Sample $S_{1,4}$ magnitude and phase plots

Table 5.6: Ex. 6 Simulation Results

	PBRLM	PSFLM
Comp. Time (SVD)	2s	1835s
Longest Stab. Band	[4.8625, 5.0375]F/m	[4.8500, 5.5375]F/m
Poly. Regr. Error	n/a	
Absolute RMS error	5.72e-3	1.73e-3
Abs. RMS err. over longest stab. band	5.77e-3	1.57e-3

On Figure 5.26 are plotted the $S_{1,4}$ magnitudes and phases at select number of parameter points. The sample parameters points are selected using linear distribution over the continuous range of parameters where the PSFLM method achieved stability. In both plots, we can see accurate matching of data between the two data origins: HFSS and PSFLM approximation.

The overall numerical results of the parametric models are given on Table 5.6. The accuracy, represented as the absolute RMS error, reaches 1.73e-3 by the PSFLM model and reaches the higher value of 5.72e-3 by the PBRLM model. Furthermore, the continuous stability band achieved by the PSFLM model is far wider than the one achieved by the PBRLM model. The stability profiles can be visualized on the poles maximum real part plot on Figure 5.27. We note the multiple asymptotic

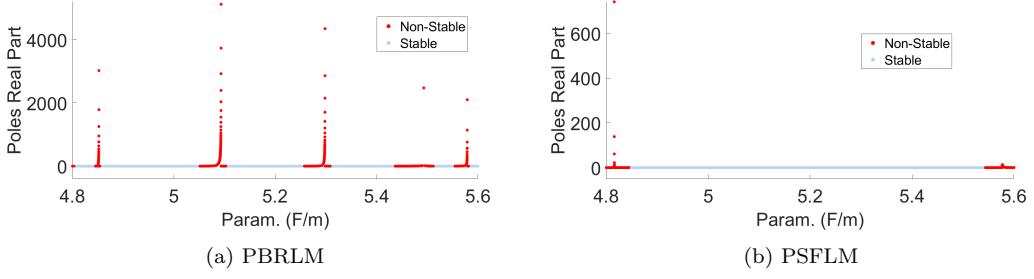


Figure 5.27: Ex. 6 poles maximum real part plot

non-stable regions in Figure 5.27a, which once again highlight the main problem with the PBRLM method which arbitrarily creates severe instabilities. As for the stability profile of the PSFLM model on Figure 5.27b, we also observe an asymptotic non-stability, but it is consigned to the extremities of the θ region of approximation, resulting in a clean stability zone within the core of the θ domain. Figure 5.28

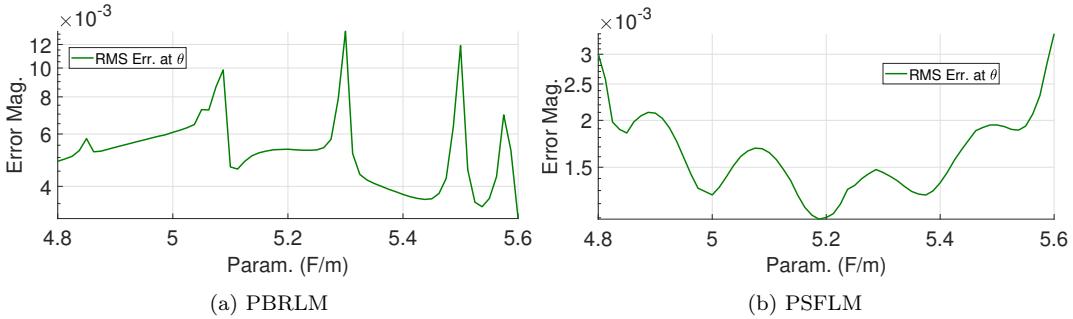


Figure 5.28: Ex. 6 RMS error w.r.t. freq. domain plot.

provides further insight into how accuracy is distributed over the θ domain, with Figure 5.28a showing error spikes matching the asymptotic spikes of the pole plot for the PBRLM model and Figure 5.28b showing a more gradual degradation of accuracy as we approach the extremities of the θ domain for the PSFLM model.

5.3.8 Overall Results Analysis

We start the analysis of the simulation results with the computation time, which came as no surprise with the PBRLM method being far more efficient than the PSFLM method. The cost scaling of performing SVD on the full parametric Loewner Matrix inflates the overall computation cost greatly, making the PSFLM method undeniably expensive.

Continuing the analysis with the accuracy results, we can see that the PSFLM method achieved better RMS error values than the PBRLM method in all cases. However, one should note that the difference in accuracy is usually not ground-breaking as the RMS errors are on similar scales. What's more interesting is the

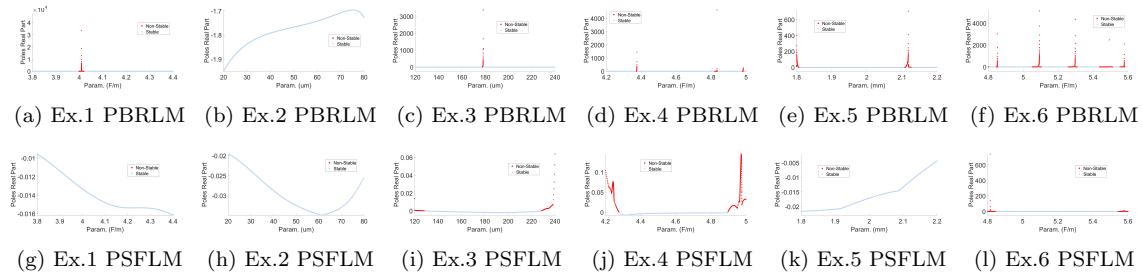


Figure 5.29: Poles maximum real part plots for all examples (divided by columns, ex. 1 to 6 from left to right) by both methods (divided by rows, method PBRLM to PSFLM from top to bottom).

stability results from the examples, which is summarized on Figure 5.29, where we have grouped together all poles maximum real part plots from each example and method. From the first row of Figure 5.29, the key issue with the PBRLM method is made visible: asymptotic unstable regions may appear in an unpredictable manner over the θ band of interpolation. With the exception of example 2, this problem can be observed in all examples, which makes it difficult to properly define an effective θ region for the PBRLM models. On the other hand, the PSFLM method provides

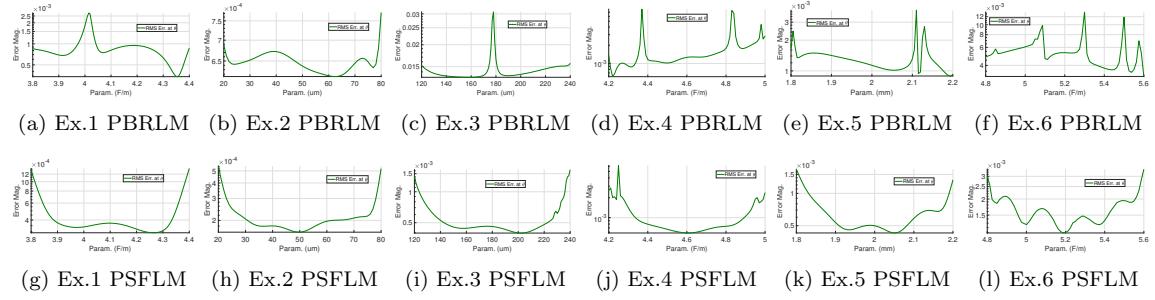


Figure 5.30: RMS error over freq. domain plots at θ points for all examples (divided by columns, ex. 1 to 6 from left to right) by both methods (divided by rows, method PBRLM to PSFLM from top to bottom).

generally better stability patterns with examples 1, 2, and 5 achieving full initial θ band stability. However, it is more interesting to observe the stability band result of example 3, 4, and 6, which demonstrate a gradual lost of stability as we approach the limits of the θ band.

On Figure 5.30 is plotted all examples' frequency domain RMS error at each θ . If we compare the first row of sub-figures in Figure 5.29 with the first row of sub-figures in Figure 5.30, we can see the error peaks where we have asymptotic non-stability in the PBRLM models. On the other-hand, the second row of sub-figures in Figure 5.29 all demonstrate a profile of strong accuracy near the center of the θ band with gradually degrading accuracy as we approach the border of the band.

The overall assessment is that though the PBRLM method can achieve parametric data interpolation with good accuracy at efficient cost, the stability profile is easily corrupted with unpredictable asymptotic non-stable θ areas that also directly cause error spikes over the same θ areas. On the other hand, the PSFLM method produces parametric macromodels with a consistent profile of strong accuracy and chance of stability for the inner portion of the θ band being approximated, though it does so at

a much higher cost than the PBRLM method. Given the above assessments, the ideal procedure when utilizing the PSFLM method should be first dissecting the θ region of approximation into sufficiently small and slightly over-lapping θ sub-bands and then create individual parametric macromodels for each sub-band so that the combined parametric models create an overall stable and accurate interpolation solution over the full θ region.

Chapter 6

Conclusion

In this chapter, we summarize the results of the contributions presented throughout this thesis.

6.1 Passivity Enforcement Error Control Scheme Contributions

The first contributions presented in the thesis is with regard to error control over the passivity enforcement process presented over Chapter 3. The specific passivity enforcement process used to present the subject is the perturbation of the \mathbf{C} matrix from the S -parameter descriptor system whose transfer function is defined as:

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \quad (6.1)$$

with the assumption that \mathbf{E} is non-singular and the perturbant defined as $d\mathbf{C}$. The existing method of minimizing the perturbation error is to switch the minimization of the matrix norm of the perturbant $d\mathbf{C}$ to the minimization of the matrix norm of $d\mathbf{C}\mathbf{L}^T$, where \mathbf{L} is the Choleski factor of the system's controllability gramian \mathbf{G} whose inclusion in the minimization cause strong alignment with the minimization of perturbation error.

The first contribution with regard to this error minimization is the introduction of the discrete gramian \mathbf{G}_Ψ as a replacement to \mathbf{G} in Section 3.1. Unlike the computation for \mathbf{G} , \mathbf{G}_Ψ is defined with a select number of frequency data over the s domain, and thus allows the user to customize the areas over the s domain where

error control is focused on. This effect is also achieved with the filtered gramian \mathcal{G}_F , but the generation of \mathcal{G}_{Ψ} does not require the design of a filter and avoids the added cost of computing and factoring a larger gramian due to the addition of the filter to the system. In the simulation Section 3.3, the discrete gramian has shown similar if not better error control over the passivity enforcement at low cost compared to its counterparts.

The second contribution with regard to the error minimization of passivity enforcement process is the SFLM method built-in error control scheme presented in Section 3.2. This scheme leverages on the special structure of the SFLM descriptor system which has the perturbant dC matrix norm minimization being strongly aligned with perturbation error minimization by construction. Although this scheme is uniquely applicable to the SFLM systems, it holds the advantage of avoiding the computational cost of the Choleski factorization of the gramian entirely while being able to achieve strong error control as was demonstrated in the simulation Section 3.3. However, we have also seen from the simulations in Section 3.3 that this error control has the highest risk of divergence during the enforcement process, so it has a reliability issue which cannot be ignored.

6.2 Parametric Loewner Matrix Methods Contributions

The second set of contributions are related to parametric macromodeling in the LM framework. The specific goal of parametric modeling in this thesis is to achieve a LTI continuous time descriptor system constructed through a LM macromodeling method that integrates a dependence with one or more physical variables θ , resulting

in a parametric transfer function $\mathbf{G}(s, \theta)$ that can provide the system frequency response at the desired parameter setting.

6.2.1 Polynomial Barycentric Rational Loewner Matrix Interpolation Method

The first contribution with regard to LM parametric macromodeling is the PBRLM method presented over Chapter 4. The PBRLM method uses Legendre polynomials to introduce parametric dependence in the original BRLM interpolation method. The addition of polynomials in the BRLM equation indirectly mitigates the issue of strict interpolation condition of the partition Λ data, which was a factor which increased risk of non-stable poles in the final system. The use of polynomials for parametric extension in the BRLM equation also results in a slim LM

$$\mathbb{L} \in \mathbb{C}^{\bar{p}\bar{m}\bar{q}\bar{\iota} \times \bar{y}\bar{\jmath}}$$

$[\bar{p} = \text{output count}] [\bar{m} = \text{input count}] [\bar{q} = \theta \text{ count}]$

$[\bar{\iota} = \mathcal{M} \text{ freq. partition count}] [\bar{j} = \Lambda \text{ freq. partition count}] [\bar{y} = \text{Polynomial count}]$

Given \bar{j} , which should be significantly smaller than $\bar{\iota}$, decides the number of poles of the system which is naturally limited for most examples and \bar{y} decides the number of polynomials which is generally comparable or lower than \bar{j} , the product $\bar{y} \cdot \bar{j}$ is typically significantly smaller than the product $\bar{p}\bar{m}\bar{q}\bar{\iota}$, which results in a slim \mathbb{L} . Coupled with the fact that we only need the right singular vectors of \mathbb{L} for the LM method, the efficient SVD method presented in Section 4.3.2 makes extraction of left singular vectors of \mathbb{L} exceptionally cost efficient.

The approximation capability of the PBRLM method is presented in both simulation Sections 4.4 and 5.3. From the simulation results in Section 4.4, we have seen that PBRLM can achieve highly accurate interpolation of parametric examples with low number of ports. We have also seen how the PBRLM method faired better than the method of [114] in terms of smoothness of frequency response over the parameter domain due to the latter using double rational barycentric interpolation whereas the PBRLM uses the barycentric equation with polynomials. From the simulation results in Section 5.3, we have seen that the PBRLM method is highly computationally efficient, achieving SVD time no more than 2s for even the largest example shown whereas the PSFLM method used 1835s for its SVD computation. However, simulation results in Section 5.3 also revealed how the PBRLM method struggles to maintain stability of the parametric model over the parameter domain, with sporadic asymptotic unstable protrusions over the pole plot.

All in all, the PBRLM interpolation method is accurate, highly computationally efficient PLM interpolation method that struggles somewhat in maintaining stability over the parameter domain.

6.2.2 Parametric System Format Loewner Matrix Interpolation Method

The second contribution presented with regard to LM parametric macromodeling is the PSFLM method presented over Chapter 5. The method is a direct two variate extension to the SFLM interpolation method, where the LMs \mathbb{L} and $\sigma\mathbb{L}$ along with their auxiliary data matrices \mathbf{W} and \mathbf{F} are directly used as the matrices $\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}$ of a descriptor system with \mathbf{D} set to null:

$$\mathbf{G}(s) = \mathbf{W} [-s \cdot \mathbb{L} + \sigma\mathbb{L}]^{-1} \mathbf{F} \quad (6.2)$$

As was shown in from Section 5.2, the two-variate extension to (6.2) is:

$$\mathbf{G}(s, \theta) = \mathbf{W}_{\Lambda, \Pi} [\theta \cdot s \cdot \mathbb{T} - \theta \cdot \sigma \mathbb{T} - s \cdot \tau \mathbb{T} + \sigma \tau \mathbb{T}]^{-1} \mathbf{F}_{\mathcal{M}, \mathcal{E}} \quad (6.3)$$

which at least interpolated two out of four parametric data partitions correctly while leaving the two other partitions being covered through interleaving partitioning scheme on both the frequency and parameter partitions with balanced sizes between the partition pairs. Real matrix transformation and model order reduction processes for the PLMs were also detailed through Section 5.2.3 and Section 5.2.4, respectively.

The final complete PSFLM interpolation method is tested in Section 5.3, where multiple HFSS examples were used with polynomial post-process smoothing as described in Section 5.3.1. Simulations of PSFLM were conducted in direct comparison with the PBRLM interpolation method, and we have seen that the PBRLM method had far superior computation cost efficiency. This is due to the fact that the PSFLM method requires performing the SVD on the PLM pencil $\Gamma(x, y) \in \mathbb{C}^{\bar{p}\bar{k}\bar{i} \times \bar{m}\bar{l}\bar{j}}$ defined as (5.56), where:

- \bar{p}, \bar{m} = number of outputs and inputs, respectively.
- \bar{k}, \bar{l} = parameter partition B and A sizes, respectively.
- \bar{i}, \bar{j} = frequency partition B and A sizes, respectively.

Generally, both the s and θ partition pairs are made as equal in size as possible ($\bar{k} \approx \bar{l}$, $\bar{i} \approx \bar{j}$), which results in $\Gamma(x, y)$ having a square profile as opposed to the slim profile of the \mathbb{L} matrix from the PBRLM method. Furthermore, both left and right singular vector matrices of $\Gamma(x, y)$ are needed in the MOR process shown in Section 5.2.4, so the QR-decomposition trick used in Section 4.3.2 is not applicable

to $\Gamma(x, y)$. However, what is lost in terms of computation efficiency is gained through the consistent accuracy and stability of the resulting models created by the PSFLM method for the examples of Section 5.3, where we have seen that all examples PSFLM models exhibit the consistent pattern of strong accuracy and stability of the parametric models over the center of θ domain and a gradual degradation towards the edge of the θ domain. In contrast, the PBRLM method, though achieving similar accuracy as the PSFLM method for the examples of Section 5.3, tends to generate erratic stability profile where asymptotic non-stable poles occur in a haphazard way over the θ domain which also cause dip in accuracy over the θ area near these pole spikes.

In summary, the PSFLM method is shown as an interesting alternative parametric LM macromodeling method to the PBRLM method with its much more consistent and predictable accuracy and stability profile, but which comes at a comparatively severe computational cost that needs to be addressed if it is to be used on a realistic time-frame.

6.3 Future Works

Based on the contributions presented in this thesis, a number of possible research extensions are presented in this section.

6.3.1 Discrete Gramian Customization

The discrete gramian defined in Section 3.1 was formulated to mimic what the filter gramian was defined to achieve: focused error control on passivity enforcement over the frequency band relevant to the system. However, the discrete gramian is defined by chosen sample data points, and thus can easily be manipulated to focus

error control wherever is needed over the frequency domain by simply including the related sample points near the focus points at whichever required level of sample density. This level of freedom in shifting error control focus is not easily achievable by filtered gramians. Furthermore, the formulation of the discrete gramian can be modified to put weight on individual sample points when constructing the discrete gramian as an alternative way to put higher error control focus on specific sample points. These customization options are interesting subject for further research for a complete picture of the potentials of the discrete gramian.

6.3.2 AAA Algorithm for Polynomial LM

For the standard Barycentric rational Loewner Matrix interpolation method reviewed in Section 2.2.2, an issue that was not mentioned was that we cannot generally predict the exact number of sample points to allocate in Λ , which decides the number of poles of the system. We can only assess if our partitioning of Λ and \mathcal{M} is satisfactory after constructing the \mathbb{L} matrix and performing SVD on it to extract one of its right singular vector to serve as the coefficient vector of the barycentric equation (2.21) that we rewrite here:

$$\mathbf{G}(s) = \frac{\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\lambda_j}}{s - \lambda_j}}{\sum_{j=1}^{\bar{j}} \frac{c_j}{s - \lambda_j}} \quad (6.4)$$

The existing solution is the AAA algorithm [63] which iteratively transfers the highest error entry in partition \mathcal{M} into partition Λ and solving for $\{c_j\}_{j=1}^{j=\bar{j}}$ in (6.4). The process is made efficient by simply deleting the rows in \mathbb{L} related to the transferring

entry from \mathcal{M} and add the subsequent columns in \mathbb{L} required for the new entry of Λ . The process repeats until the desired accuracy threshold is reached. A faster version of the algorithm under the name of Fast AAA can be used to significantly reduce the computation cost of the iterative process. Rather than performing SVD after each transfer, the SVD can be updated for the row deletion and column addition in the \mathbb{L} matrix [62], avoiding the full cost of SVD.

Naturally, such a process would be useful for the PBRLM method presented in Chapter 4, as the equation of the method is built upon (6.4) with parametric extension through the addition of polynomials. If an AAA process can be established for the PBRLM method, we can then more systematically determine good PBRLM models.

6.3.3 Identical Partition LM

In Section 5.2.5 was presented the 4-quadrant PSFLM method, the process of which included the introduction of non-parametric LM utilizing identical data partition pairs which ended up requiring the data derivative with respect to s . The identical partition LM itself should be explored a little better as it can provide an interesting option in constructing LMs in general. At the very least, use of derivative in constructing LM should be more properly examined before attempting to use it on a higher level such as the 4-quadrant PSFLM method.

6.3.4 PSFLM Method Computation Cost Improvement

The PSFLM method presented over Chapter 5 has been shown to provide relatively consistent parametric macromodels in terms of accuracy and stability, but it suffers from a high computation cost that scales badly. Naturally, remedying the

poor cost scaling is of high interest for this method. One possible way towards this goal is to use uneven partition pair (Λ, \mathcal{M}) or/and (Π, \mathcal{E}) to create a relatively slimmer profile for the parametric Loewner Matrices $\mathbb{T}, \sigma\mathbb{T}, \tau\mathbb{T}$, and $\sigma\tau\mathbb{T}$ which would result in significant cut in SVD cost. If we observe the examples from Section 5.3, the cut-off indices after the SVD are significantly smaller than the total width/height of the PLM pencil, which means there is generally ample room for the partition pairs (Λ, \mathcal{M}) or (Π, \mathcal{E}) to transfer a good number of entries from one partition to its counterpart. However, unbalanced partitions for the SFLM and PSFLM methods are not thoroughly tested and would require further studies if they are to be proven valid.

Another line of solution is simply utilizing the model order reduction techniques. Unlike the BRLM and PBRLM methods where we seek the smallest singular values of the LM and its corresponding right singular vector, the PSFLM method seeks the largest singular values and their related singular vectors. As such, a process similar to the Arnoldi [31] or Lanczos [32, 33] algorithms could potentially serve as alternative to the brute force SVD method, especially for the PSFLM method where we can observe in the examples of Section 5.3 that the SVD cut-off point is generally extremely small compared to the total number of singular values available.

Appendix A

Barycentric Rational to State-Space

In this appendix, we present the process of state-space format translation from the barycentric rational equation (2.21) which is rewritten here:

$$\mathbf{G}_{br}(s) = \frac{\sum_{j=1}^{\bar{J}} \frac{c_j \mathbf{H}_{\lambda_j}}{s - \lambda_j}}{\sum_{j=1}^{\bar{J}} \frac{c_j}{s - \lambda_j}} \quad (\text{A.1})$$

where the subscript ‘br’ means ‘barycentric rational’. The process is performed in steps:

- From barycentric rational form to pole-zero form.
- From pole-zero form to pole-residue form.
- From pole-residue form to state-space form.

The overall process is loosely based on the translation process of VF equation to its state-space equivalent given in [42], but adjusted for the barycentric equation case.

The translation process is first presented for the SISO case ($\bar{p} = 1$, $\bar{m} = 1$) and then the general MIMO ($\bar{p} \geq 1$, $\bar{m} \geq 1$) case is built upon the SISO case. Note that the translation formula presented in this appendix is by no mean the cutting-edge formula. The present formula is meant to concretely show that the barycentric rational equation form is directly linked to the state-space representation form.

A.1 Barycentric Rational to Pole-Zero Form

The derivation is done under the SISO assumption ($\bar{p} = 1$, $\bar{m} = 1$). The zeros of (A.1) are the zeros of its numerator:

$$\sum_{j=1}^{\bar{j}} \frac{c_j \mathbf{H}_{\lambda_j}}{s - \lambda_j} \quad (\text{A.2})$$

The poles of (A.1) are the zeros of its denominator:

$$\sum_{j=1}^{\bar{j}} \frac{c_j}{s - \lambda_j} \quad (\text{A.3})$$

Both (A.2) and (A.3) are functions falling under the general function format:

$$v(s) = \sum_{j=1}^{\bar{j}} \frac{k_j}{s - g_j} + \mathbf{D} \quad (\text{A.4})$$

In the context of generating physical systems, the set of variables $\{k_j\}_{j=1}^{j=\bar{j}}$ and $\{g_j\}_{j=1}^{j=\bar{j}}$ must contain either real (Namely, the 0 frequency) or complex conjugate pairs of entries. To meet this requirement, define subsets of the initial variables sets:

$$\begin{aligned} \{k_{r\alpha}\}_{\alpha=1}^{\alpha=\bar{\alpha}}, \ k_{r\alpha} \in \mathbb{R} &\quad \{k_{c\beta}\}_{\beta=1}^{\beta=\bar{\beta}}, \ k_{c\beta} \in \mathbb{C} \\ \{g_{r\alpha}\}_{\alpha=1}^{\alpha=\bar{\alpha}}, \ g_{r\alpha} \in \mathbb{R} &\quad \{g_{c\beta}\}_{\beta=1}^{\beta=\bar{\beta}}, \ g_{c\beta} \in \mathbb{C} \end{aligned} \quad (\text{A.5})$$

where

- $\bar{\alpha}$, α : real frequency count and index, respectively.
- $\bar{\beta}$, β : complex conjugate frequency pair count and index, respectively.
- $\bar{\alpha} + 2\beta = \bar{j}$

With the subset definitions, equation (A.4) now has the format:

$$v(s) = \sum_{\alpha=1}^{\bar{\alpha}} \left(\frac{k_{r\alpha}}{s - g_{r\alpha}} \right) + \sum_{\beta=1}^{\bar{\beta}} \left(\frac{k_{c\beta}}{s - g_{c\beta}} + \frac{k_{c\beta}^*}{s - g_{c\beta}^*} \right) + \mathbf{D} \quad (\text{A.6})$$

If we regard equation (A.6) as a transfer function, we can write it in the form of a standard state-space transfer function:

$$v(s) = \mathbf{C}_c(s\mathbf{I}_{\bar{J} \times \bar{J}} - \mathbf{A}_c)^{-1} \mathbf{B}_c + \mathbf{D} \quad (\text{A.7})$$

where the matrix definitions are:

$$\begin{aligned} \mathbf{B}_c &\in \mathbb{C}^{\bar{J} \times 1}, \quad \mathbf{A}_c \in \mathbb{C}^{\bar{J} \times \bar{J}}, \quad \mathbf{C}_c \in \mathbb{C}^{1 \times \bar{J}} \\ \mathbf{B}_c &= \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \hline \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ \vdots \\ \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ 1 \end{bmatrix}, \quad \mathbf{A}_c = \begin{bmatrix} g_{r1} & & & & \\ & \ddots & & & \\ & & g_{r\bar{\alpha}} & & \\ & & & g_{c1} & \\ & & & & g_{c1}^* \\ & & & & \ddots \\ & & & & g_{c\bar{\beta}} \\ & & & & g_{c\bar{\beta}}^* \end{bmatrix} \quad (\text{A.8}) \\ \mathbf{C}_c &= \left[\begin{array}{ccc|ccccc} k_{r1} & \cdots & k_{r\bar{\alpha}} & k_{c1} & k_{c1}^* & \cdots & k_{c\bar{\beta}} & k_{c\bar{\beta}}^* \end{array} \right] \end{aligned}$$

We notice that the matrices in (A.8) have the structure necessary to undergo the real matrix transformation process as presented over Appendix B and specifically in the appendix Section B.4. We perform the transform to turn the state-space matrices

real, but with added portion for the entries that are already real:

$$\mathfrak{T}_L = \begin{bmatrix} \mathbf{I}_{\bar{x} \times \bar{x}} \\ \mathbf{T}_{(1\langle y \rangle)} \end{bmatrix} \in \mathbb{C}^{\bar{j} \times \bar{j}}, \quad \mathfrak{T}_R = \begin{bmatrix} \mathbf{I}_{\bar{x} \times \bar{x}} \\ \mathbf{T}_{(1\langle y \rangle)} \end{bmatrix} \in \mathbb{C}^{\bar{j} \times \bar{j}} \quad (\text{A.9})$$

where $\mathbf{T}_{(1\langle y \rangle)} \in \mathbb{C}^{2y \times 2y}$ is based on the general matrix definition defined as (B.10) which we rewrite here:

$$\mathbf{T}_{(x\langle y \rangle)} \in \mathbb{C}^{2yx \times 2yx} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathcal{T}_{(x)} & & & \\ & \mathcal{T}_{(x)} & & \\ & & \ddots & \\ & & & \mathcal{T}_{(x)} \end{bmatrix}, \quad \mathcal{T}_{(x)} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{x \times x} & -1i \cdot \mathbf{I}_{x \times x} \\ \mathbf{I}_{x \times x} & 1i \cdot \mathbf{I}_{x \times x} \end{bmatrix} \in \mathbb{C}^{2x \times 2x} \quad (\text{A.10})$$

Applying the transformation would result in real system matrices:

$$\mathbf{A}_r = \mathfrak{T}_L^H \cdot \mathbf{A}_c \cdot \mathfrak{T}_R \in \mathbb{R}^{\bar{j} \times \bar{j}} \quad \mathbf{B}_r = \mathfrak{T}_L^H \cdot \mathbf{B}_c \in \mathbb{R}^{\bar{j} \times 1} \quad \mathbf{C}_r = \mathbf{C}_c \cdot \mathfrak{T}_R \in \mathbb{R}^{1 \times \bar{j}} \quad (\text{A.11})$$

where:

$$\mathbf{B}_r = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \hline \begin{bmatrix} \sqrt{2} \\ 0 \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \sqrt{2} \\ 0 \end{bmatrix} \end{bmatrix}, \quad \mathbf{A}_r = \left[\begin{array}{c|c} g_{r1} & \\ \ddots & \\ \hline g_{r\bar{\alpha}} & \\ \hline \mathbf{g}_{c1} & \\ & \ddots \\ & \mathbf{g}_{c\bar{\beta}} \end{array} \right]$$

$$\begin{aligned} \mathbf{C}_r &= \left[\begin{array}{ccc|cc} k_{r1} & \cdots & k_{r\bar{\alpha}} & \sqrt{2} \cdot \mathbf{k}_{c1} & \cdots & \sqrt{2} \cdot \mathbf{k}_{c\bar{\beta}} \end{array} \right] & (A.12) \\ \mathbf{g}_{c\beta} &= \begin{bmatrix} \Re\{g_{c\beta}\} & \Im\{g_{c\beta}\} \\ -\Im\{g_{c\beta}\} & \Re\{g_{c\beta}\} \end{bmatrix}, \quad \mathbf{k}_{c\beta} = \begin{bmatrix} \Re\{k_{c\beta}\} & \Im\{k_{c\beta}\} \end{bmatrix} \end{aligned}$$

The transfer function (A.7) can now be written in real matrix equivalent form:

$$v(s) = \mathbf{C}_r(s\mathbf{I} - \mathbf{A}_r)^{-1}\mathbf{B}_r + \mathbf{D} \quad (A.13)$$

for which the zeros of the transfer function are equivalent to the poles of the following state transition equation [140]:

$$\begin{bmatrix} \mathbf{I}_{\bar{\jmath} \times \bar{\jmath}} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{u} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_r & \mathbf{B}_r \\ -\mathbf{C}_r & -\mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ u \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} y \quad (A.14)$$

The poles in (A.14) are the eigenvalues of the matrix pencil:

$$\left(\begin{bmatrix} \mathbf{A}_r & \mathbf{B}_r \\ -\mathbf{C}_r & -\mathbf{D} \end{bmatrix}, \begin{bmatrix} \mathbf{I}_{\bar{\jmath} \times \bar{\jmath}} & 0 \\ 0 & 0 \end{bmatrix} \right) \quad (A.15)$$

Suppose we have computed the poles of (A.2) and (A.3) which are the equivalent of the zeros set \mathcal{Z} and the poles set \mathcal{P} , respectively, of (A.1):

$$\mathcal{Z} = \{\chi_j; j = 1, \dots, \bar{\jmath} - 1\} \quad (A.16)$$

$$\mathcal{P} = \{\rho_j; j = 1, \dots, \bar{\jmath} - 1\} \quad (A.17)$$

If there is no error in the poles and zeros, then we have the base pole-zero function:

$$\tilde{\mathbf{G}}_{pz}(s) = \frac{\prod_{j=1}^{\bar{\jmath}-1} (s - \chi_j)}{\prod_{j=1}^{\bar{\jmath}-1} (s - \rho_j)} \quad (A.18)$$

being different from $\mathbf{G}_{br}(s)$ by the constant factor γ , that is

$$\mathbf{G}_{br}(s) = \gamma \cdot \tilde{\mathbf{G}}_{pz}(s) \quad (\text{A.19})$$

It should be noted that because the matrices in (A.15) are real, the finite non-zero poles are either real or complex conjugate pairs. This means that both the \mathcal{Z} and \mathcal{P} sets are composed exclusively of either real or complex conjugate pair terms, which can be expressed as follow:

$$\mathcal{Z} = \{\chi_{r\mathfrak{c}}; \mathfrak{c} = 1, \dots, \bar{\mathfrak{c}}; \chi_{r\mathfrak{c}} \in \mathbb{R}\} \cup \{\chi_{c\mathfrak{d}}, \chi_{c\mathfrak{d}}^*; \mathfrak{d} = 1, \dots, \bar{\mathfrak{d}}; \chi_{c\mathfrak{d}} \in \mathbb{C}\} \quad (\text{A.20})$$

$$\mathcal{P} = \{\rho_{r\mathfrak{a}}; \mathfrak{a} = 1, \dots, \bar{\mathfrak{a}}; \rho_{r\mathfrak{a}} \in \mathbb{R}\} \cup \{\rho_{c\mathfrak{b}}, \rho_{c\mathfrak{b}}^*; \mathfrak{b} = 1, \dots, \bar{\mathfrak{b}}; \rho_{c\mathfrak{b}} \in \mathbb{C}\} \quad (\text{A.21})$$

where

- $\bar{\mathfrak{c}}$, \mathfrak{c} : real zeros count and index, respectively.
- $\bar{\mathfrak{d}}$, \mathfrak{d} : complex conjugate zero pair count and index, respectively.
- $\bar{\mathfrak{a}}$, \mathfrak{a} : real pole count and index, respectively.
- $\bar{\mathfrak{b}}$, \mathfrak{b} : complex conjugate pole pair count and index, respectively.
- $\bar{\mathfrak{c}} + 2\mathfrak{d} = \bar{\mathfrak{a}} + 2\mathfrak{b} = \bar{j} - 1$

Using the \mathcal{Z} and \mathcal{P} subset breakdown of (A.20) and (A.21), respectively, (A.22) takes the form of:

$$\tilde{\mathbf{G}}_{pz}(s) = \frac{\prod_{\mathfrak{c}=1}^{\bar{\mathfrak{c}}}(s - \chi_{\mathfrak{c}}) \prod_{\mathfrak{d}=1}^{\bar{\mathfrak{d}}}(s - \chi_{\mathfrak{d}})(s - \chi_{\mathfrak{d}}^*)}{\prod_{\mathfrak{a}=1}^{\bar{\mathfrak{a}}}(s - \rho_{\mathfrak{a}}) \prod_{\mathfrak{b}=1}^{\bar{\mathfrak{b}}}(s - \rho_{\mathfrak{b}})(s - \rho_{\mathfrak{b}}^*)} \quad (\text{A.22})$$

Given that the sets \mathcal{Z} , \mathcal{P} , $\{k_j\}_{j=1}^{j=\bar{j}}$, and $\{g_j\}_{j=1}^{j=\bar{j}}$ all only contain either real or complex conjugate pair entries, we have the necessary evaluations $\mathbf{G}_{br}(0) \in \mathbb{R}$ and $\tilde{\mathbf{G}}_{pz}(0) \in \mathbb{R}$ turning real. As such, γ in (A.19) becomes necessarily real as well. In order to determine γ , we have to use a test point s_{test} to evaluate the $\mathbf{G}_{br}(s)$ and solve for γ

in the equality:

$$\mathbf{G}_{br}(s_{test}) = \mathbf{G}_{pz}(s_{test}) = \gamma \frac{\prod_{j=1}^{\bar{j}-1} (s_{test} - \chi_j)}{\prod_{j=1}^{\bar{j}-1} (s_{test} - \rho_j)} \quad (\text{A.23})$$

s_{test} can be any value that does not belong to the sets \mathcal{Z} and \mathcal{P} . The final pole-zero form equation is defined as:

$$\mathbf{G}_{pz}(s) = \gamma \frac{\prod_{j=1}^{\bar{j}-1} (s - \chi_j)}{\prod_{j=1}^{\bar{j}-1} (s - \rho_j)} \quad (\text{A.24})$$

which is equivalent to $\mathbf{G}_{br}(s)$.

A.2 Pole-Zero to Pole-Residue Form

The pole-residue form can be generally represented by the following equation:

$$\mathbf{G}_{pr}(s) = \tau_0 + \sum_{j=1}^{\bar{j}-1} \frac{\tau_j}{s - \rho_j} \quad (\text{A.25})$$

With the poles set \mathcal{P} already determined through the pole-zero formulation (A.24), it is the residue set

$$\mathcal{R} = \{\tau_j; j = 1, \dots, \bar{j} - 1\} \quad (\text{A.26})$$

that is to be determined. The process involves solving the equality equation between the pole-zero and pole-residue formulations:

$$\gamma \frac{\prod_{j=1}^{\bar{j}-1} (s - \chi_j)}{\prod_{j=1}^{\bar{j}-1} (s - \rho_j)} = \tau_0 + \sum_{j=1}^{\bar{j}-1} \frac{\tau_j}{s - \rho_j} \quad (\text{A.27})$$

To solve for any $\tau_j \in \mathcal{R}$ specifically, simply multiply both sides of the equality in (A.27) by $(s - \rho_x)$ which turns (A.27) into the form:

$$\gamma \frac{\prod_{j=1, j \neq x}^{\bar{j}-1} (s - \chi_j)}{\prod_{j=1, j \neq x}^{\bar{j}-1} (s - \rho_j)} = \tau_x + (s - \rho_x) \left[\tau_0 + \sum_{j=1, j \neq x}^{\bar{j}-1} \frac{\tau_j}{s - \rho_j} \right] \quad (\text{A.28})$$

Setting $s = \rho_x$, equation (A.28) reduces to:

$$\gamma \frac{\prod_{j=1}^{\bar{j}-1} (\rho_x - \chi_j)}{\prod_{j=1, j \neq x}^{\bar{j}-1} (\rho_x - \rho_j)} = \tau_x \quad (\text{A.29})$$

Equation (A.29) can directly solve for all $\tau_x \in \mathcal{R}$. It should be also noted that because all $\chi_j \in \mathcal{Z}$ and $\rho_j \in \mathcal{P}$ come as either real or as complex conjugate pairs, we naturally have the complex conjugacy property for (A.29):

$$\gamma \left\{ \frac{\prod_{j=1}^{\bar{j}-1} (\rho_x - \chi_j)}{\prod_{j=1, j \neq x}^{\bar{j}-1} (\rho_x - \rho_j)} \right\}^* = \gamma \frac{\prod_{j=1}^{\bar{j}-1} (\rho_x^* - \chi_j^*)}{\prod_{j=1, j \neq x}^{\bar{j}-1} (\rho_x^* - \rho_j^*)} = \tau_x^* \quad (\text{A.30})$$

Because both $\rho_j, \rho_j^* \in \mathcal{P}$, then both $\tau_j, \tau_j^* \in \mathcal{R}$. This means the set \mathcal{R} only contains real or complex conjugate pair residue values with matching number of real and complex conjugate pairs as the entries in \mathcal{P} :

$$\mathcal{R} = \{\tau_{r\bar{a}}; \bar{a} = 1, \dots, \bar{\bar{a}}; \tau_{r\bar{a}} \in \mathbb{R}\} \cup \{\tau_{c\bar{b}}, \tau_{c\bar{b}}^*; \bar{b} = 1, \dots, \bar{\bar{b}}; \tau_{c\bar{b}} \in \mathbb{C}\} \quad (\text{A.31})$$

Once all residues beside τ_0 are computed, τ_0 itself can be solved as follow:

$$\mathbf{G}_{pz}(s) - \sum_{j=1}^{\bar{j}-1} \frac{\tau_j}{s - p_j} = \tau_0 \quad (\text{A.32})$$

where τ_0 is a constant. τ_0 is necessarily real because $\mathbf{G}_{pz}(0) \in \mathbb{R}$ and

$$\sum_{j=1}^{\bar{j}-1} \frac{\tau_j}{0 - \rho_j} \in \mathbb{R} \quad (\text{A.33})$$

by virtue of the fact that the sets \mathcal{Z} , \mathcal{R} , and \mathcal{P} have only purely real or complex conjugate pair entries. Once we have obtained all $\{\tau_j\}_{j=1}^{\bar{j}-1}$, τ_0 can be solved by

evaluating $\mathbf{G}_{pr}(s)$ at any zeros χ_y for $y \in 1, \dots, \bar{j} - 1$:

$$\begin{aligned}\mathbf{G}_{pr}(\chi_y) &= \tau_0 + \sum_{j=1}^{\bar{j}-1} \frac{\tau_j}{\chi_y - \rho_j} = 0 \\ \tau_0 &= - \sum_{j=1}^{\bar{j}-1} \frac{\tau_j}{\chi_y - \rho_j}\end{aligned}\tag{A.34}$$

A.3 Pole-Residue to State-Space Form

At the stage of pole-residue form, the state-space form conversion is already immediately available just like how (A.4) can be directly turned to its state-space equivalent (A.7). First, the pole-residue equation (A.25) rewritten here

$$\mathbf{G}_{pr}(s) = \tau_0 + \sum_{j=1}^{\bar{j}-1} \frac{\tau_j}{s - \rho_j}\tag{A.35}$$

can be turned into the alternate form with its residues and poles clearly sub-divided into the group of purely real entries and the group of complex conjugate pair entries following the sub-division established by (A.31):

$$\begin{aligned}\{\tau_{r\mathfrak{a}}\}_{\mathfrak{a}=1}^{\mathfrak{a}=\bar{\mathfrak{a}}}, \tau_{r\mathfrak{a}} &\in \mathbb{R} & \{\tau_{c\mathfrak{b}}\}_{\mathfrak{b}=1}^{\mathfrak{b}=\bar{\mathfrak{b}}}, \tau_{c\mathfrak{b}} &\in \mathbb{C} \\ \{\rho_{r\mathfrak{a}}\}_{\mathfrak{a}=1}^{\mathfrak{a}=\bar{\mathfrak{a}}}, \rho_{r\mathfrak{a}} &\in \mathbb{R} & \{\rho_{c\mathfrak{b}}\}_{\mathfrak{b}=1}^{\mathfrak{b}=\bar{\mathfrak{b}}}, \rho_{c\mathfrak{b}} &\in \mathbb{C}\end{aligned}\tag{A.36}$$

where $\bar{\mathfrak{a}} + 2\mathfrak{b} = \bar{j} - 1$. The resulting alternative form of (A.35) is defined as:

$$\mathbf{G}_{pr}(s) = \tau_0 + \sum_{\mathfrak{a}=1}^{\bar{\mathfrak{a}}} \frac{\tau_{r\mathfrak{a}}}{s - \rho_{r\mathfrak{a}}} + \sum_{\mathfrak{b}=1}^{\bar{\mathfrak{b}}} \frac{\tau_{c\mathfrak{b}}}{s - \rho_{c\mathfrak{b}}}\tag{A.37}$$

Following the same process which turned (A.6) into its real matrix state-space transfer function equivalent (A.13), we can turn (A.37) into its own real matrix state-space

transfer function equivalent:

$$\mathbf{G}_{pr}(s) = \mathbf{C}_{pr}(s\mathbf{I}_{(\bar{\alpha}+2\bar{\mathfrak{b}})\times(\bar{\alpha}+2\bar{\mathfrak{b}})} - \mathbf{A}_{pr})^{-1}\mathbf{B}_{pr} + \mathbf{D}_{pr} \quad (\text{A.38})$$

where

$$\mathbf{B}_{pr} \in \mathbb{R}^{(\bar{\alpha}+2\bar{\mathfrak{b}})\times 1}, \quad \mathbf{A}_{pr} \in \mathbb{R}^{(\bar{\alpha}+2\bar{\mathfrak{b}})\times(\bar{\alpha}+2\bar{\mathfrak{b}})}, \quad \mathbf{C}_{pr} \in \mathbb{R}^{1\times(\bar{\alpha}+2\bar{\mathfrak{b}})}, \quad \mathbf{D}_{pr} \in \mathbb{R}^{1\times 1}$$

$$\mathbf{B}_{pr} = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \hline \begin{bmatrix} \sqrt{2} \\ 0 \end{bmatrix} \\ \vdots \\ \begin{bmatrix} \sqrt{2} \\ 0 \end{bmatrix} \end{bmatrix}, \quad \mathbf{A}_{pr} = \left[\begin{array}{c|c} \rho_{r1} & \cdot \\ \cdot & \rho_{r\bar{\alpha}} \\ \hline & \mathbf{p}_{c1} \\ & \cdot \\ & \mathbf{p}_{c\bar{\mathfrak{b}}} \end{array} \right], \quad \mathbf{C}_{pr} = \left[\begin{array}{ccc|cc} \tau_{r1} & \cdots & \tau_{r\bar{\alpha}} & \sqrt{2} \cdot \mathbf{r}_{c1} & \cdots & \sqrt{2} \cdot \mathbf{r}_{c\bar{\mathfrak{b}}} \end{array} \right], \quad \mathbf{D}_{pr} = \tau_0 \quad (\text{A.39})$$

$$\mathbf{p}_{c\mathfrak{b}} = \begin{bmatrix} \Re\{\rho_{c\mathfrak{b}}\} & \Im\{\rho_{c\mathfrak{b}}\} \\ -\Im\{\rho_{c\mathfrak{b}}\} & \Re\{\rho_{c\mathfrak{b}}\} \end{bmatrix}, \quad \mathbf{r}_{c\mathfrak{b}} = \begin{bmatrix} \Re\{\tau_{c\mathfrak{b}}\} & \Im\{\tau_{c\mathfrak{b}}\} \end{bmatrix}$$

Note the demarcation lines in matrices \mathbf{A}_{pr} , \mathbf{B}_{pr} , and \mathbf{C}_{pr} which subdivide the matrices into two sub-sections are located between the row or/and column of index $\bar{\alpha}$ and $\bar{\alpha} + 1$.

A.4 MIMO Consideration

In the MIMO case ($\bar{p} \geq 1, \bar{m} \geq 1$), only the numerator of (A.1) is modified with the \mathbf{H}_{λ_j} entries being data matrices rather than scalars. We rewrite (A.1) here:

$$\mathbf{G}_{br}(s) = \frac{\sum_{j=1}^{\bar{J}} \frac{c_j \mathbf{H}_{\lambda_j}}{s - \lambda_j}}{\sum_{j=1}^{\bar{J}} \frac{c_j}{s - \lambda_j}} \quad (\text{A.40})$$

Writing the data matrices with each scalar entry emphasized:

$$\mathbf{H}_{\lambda_j} = \begin{bmatrix} \mathbf{H}_{\lambda_j \{1,1\}} & \mathbf{H}_{\lambda_j \{1,2\}} & \cdots & \mathbf{H}_{\lambda_j \{1,\bar{m}\}} \\ \mathbf{H}_{\lambda_j \{2,1\}} & \mathbf{H}_{\lambda_j \{2,2\}} & \cdots & \mathbf{H}_{\lambda_j \{2,\bar{m}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_{\lambda_j \{\bar{p},1\}} & \mathbf{H}_{\lambda_j \{\bar{p},2\}} & \cdots & \mathbf{H}_{\lambda_j \{\bar{p},\bar{m}\}} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times \bar{m}} \quad (\text{A.41})$$

The formulation of the BRLM equation (A.40) has the matrix data $\mathbf{H}_{\lambda_j} \in \mathbb{C}^{\bar{p} \times \bar{m}}$ multiplied by a scalar coefficient c_j . This allows us to translate the barycentric form to the pole-residue form via pole-zero form with the zeros and residuals computed for each scalar entry of (A.41) separately. This means we run the translation process for each scalar data entry $\mathbf{H}_{\lambda_j \{p,m\}}$ one at a time to obtain the individual zeros and residues of each scalar data entry. As for the computation of poles in the MIMO case, nothing is changed because the denominator of (A.40) did not change in the MIMO case though it is implied that the poles are now repeated by a number matching the number of ports. The overall translation results can be assembled into the zero

matrix and the residue matrix:

$$\boldsymbol{\chi}_j \in \mathbb{C}^{\bar{p} \times \bar{m}} = \begin{bmatrix} \boldsymbol{\chi}_{j\{1,1\}} & \boldsymbol{\chi}_{j\{1,2\}} & \cdots & \boldsymbol{\chi}_{j\{1,\bar{m}\}} \\ \boldsymbol{\chi}_{j\{2,1\}} & \boldsymbol{\chi}_{j\{2,2\}} & \cdots & \boldsymbol{\chi}_{j\{2,\bar{m}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\chi}_{j\{\bar{p},1\}} & \boldsymbol{\chi}_{j\{\bar{p},2\}} & \cdots & \boldsymbol{\chi}_{j\{\bar{p},\bar{m}\}} \end{bmatrix}, \quad \boldsymbol{\tau}_j \in \mathbb{C}^{\bar{p} \times \bar{m}} = \begin{bmatrix} \boldsymbol{\tau}_{j\{1,1\}} & \boldsymbol{\tau}_{j\{1,2\}} & \cdots & \boldsymbol{\tau}_{j\{1,\bar{m}\}} \\ \boldsymbol{\tau}_{j\{2,1\}} & \boldsymbol{\tau}_{j\{2,2\}} & \cdots & \boldsymbol{\tau}_{j\{2,\bar{m}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\tau}_{j\{\bar{p},1\}} & \boldsymbol{\tau}_{j\{\bar{p},2\}} & \cdots & \boldsymbol{\tau}_{j\{\bar{p},\bar{m}\}} \end{bmatrix} \quad (A.42)$$

$$j = 1, \dots, \bar{j} - 1$$

We rewrite the pole-residue equation form (A.37) but which takes into account that we have only real and complex conjugate entries like in (A.6):

$$\mathbf{G}_{pr}(s) = \tau_0 + \sum_{x=1}^{\bar{a}} \left(\frac{\tau_{r\mathfrak{a}}}{s - \rho_{r\mathfrak{a}}} \right) + \sum_{y=1}^{\bar{b}} \left(\frac{\tau_{c\mathfrak{b}}}{s - \rho_{c\mathfrak{b}}} + \frac{\tau_{c\mathfrak{b}}^*}{s - \rho_{c\mathfrak{b}}^*} \right) \quad (A.43)$$

where \bar{a} is the number of real poles, \bar{b} is the number of pairs of complex conjugate poles, and $\bar{a} + 2\bar{b} = \bar{j} - 1$. In the MIMO case, if we follow the individual computation process to obtain the residue matrices defined in (A.42), the only change to (A.43) is to simply replace the scalar residues with the matrix residues:

$$\mathbf{G}_{pr}(s) = \tau_0 + \sum_{\mathfrak{a}=1}^{\bar{a}} \left(\frac{\boldsymbol{\tau}_{r\mathfrak{a}}}{s - \rho_{r\mathfrak{a}}} \right) + \sum_{\mathfrak{b}=1}^{\bar{b}} \left(\frac{\boldsymbol{\tau}_{c\mathfrak{b}}}{s - \rho_{c\mathfrak{b}}} + \frac{\boldsymbol{\tau}_{c\mathfrak{b}}^*}{s - \rho_{c\mathfrak{b}}^*} \right) \quad (A.44)$$

$$\begin{aligned} \tau_0 &\in \mathbb{R}^{\bar{p} \times \bar{m}} & \{\boldsymbol{\tau}_{r\mathfrak{a}}\}_{\mathfrak{a}=1}^{\mathfrak{a}=\bar{a}}, \boldsymbol{\tau}_{r\mathfrak{a}} &\in \mathbb{R}^{\bar{p} \times \bar{m}} & \{\boldsymbol{\tau}_{c\mathfrak{b}}\}_{\mathfrak{b}=1}^{\mathfrak{b}=\bar{b}}, \boldsymbol{\tau}_{c\mathfrak{b}} &\in \mathbb{C}^{\bar{p} \times \bar{m}} \\ \{\rho_{rx}\}_{x=1}^{x=\bar{x}}, \rho_{rx} &\in \mathbb{R} & \{\rho_{cy}\}_{y=1}^{y=\bar{y}}, \rho_{cy} &\in \mathbb{C} \end{aligned} \quad (A.45)$$

which once again can be turned into an equivalent transfer function

$$\mathbf{G}_{pr}(s) = \mathbf{C}_c(s\mathbf{I}_{(\bar{a}+2\bar{b})\bar{m} \times (\bar{a}+2\bar{b})\bar{m}} - \mathbf{A}_c)^{-1} \mathbf{B}_c + \mathbf{D} \quad (A.46)$$

with the matrices defined as:

$$\begin{aligned} \mathbf{B}_c &\in \mathbb{C}^{(\bar{a}+2\bar{b})\bar{m} \times \bar{m}}, & \mathbf{A}_c &\in \mathbb{C}^{(\bar{a}+2\bar{b})\bar{m} \times (\bar{a}+2\bar{b})\bar{m}} \\ \mathbf{C}_c &\in \mathbb{C}^{\bar{p} \times (\bar{a}+2\bar{b})\bar{m}}, & \mathbf{D} &\in \mathbb{R}^{\bar{p} \times \bar{m}} \end{aligned} \quad (\text{A.47})$$

$$\mathbf{B}_c = \left[\begin{array}{c} \mathbf{I}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{I}_{\bar{m} \times \bar{m}} \\ \hline \left[\begin{array}{c} \mathbf{I}_{\bar{m} \times \bar{m}} \\ \mathbf{I}_{\bar{m} \times \bar{m}} \end{array} \right] \\ \vdots \\ \left[\begin{array}{c} \mathbf{I}_{\bar{m} \times \bar{m}} \\ \mathbf{I}_{\bar{m} \times \bar{m}} \end{array} \right] \end{array} \right], \quad \mathbf{A}_c = \left[\begin{array}{c|c} \boldsymbol{\rho}_{r1} & \\ \ddots & \\ \hline & \boldsymbol{\rho}_{r\bar{a}} \\ & \boldsymbol{\rho}_{c1} \\ & \boldsymbol{\rho}_{c1}^* \\ & \ddots \\ & \boldsymbol{\rho}_{c\bar{b}} \\ & \boldsymbol{\rho}_{c\bar{b}}^* \end{array} \right] \quad (\text{A.48})$$

$$\mathbf{C}_c = \left[\begin{array}{ccc|cccc} \boldsymbol{\tau}_{r1} & \cdots & \boldsymbol{\tau}_{r\bar{a}} & \boldsymbol{\tau}_{c1} & \boldsymbol{\tau}_{c1}^* & \cdots & \boldsymbol{\tau}_{c\bar{b}} & \boldsymbol{\tau}_{c\bar{b}}^* \end{array} \right], \quad \mathbf{D} = \boldsymbol{\tau}_0$$

$$\boldsymbol{\rho}_{ra} = \rho_{ra} \cdot \mathbf{I}_{\bar{m} \times \bar{m}}, \quad \boldsymbol{\rho}_{cb} = \rho_{cb} \cdot \mathbf{I}_{\bar{m} \times \bar{m}}$$

where we defined the repeated pole matrices \mathbf{p}_{rx} and \mathbf{p}_{cy} . We notice that the matrices in (A.48) have the structure necessary to undergo the real matrix transformation process as presented over Appendix B. We perform the transform to turn the state-space matrices real, but with added portion for the entries that are already real

without breaking the unitary property of the transformation matrices:

$$\begin{aligned}\mathfrak{T}_L &= \begin{bmatrix} \mathbf{I}_{\bar{a}\bar{m} \times \bar{a}\bar{m}} \\ \mathbf{T}_{(\bar{m}\langle\bar{b}\rangle)} \end{bmatrix}, \quad \mathfrak{T}_R = \begin{bmatrix} \mathbf{I}_{\bar{a}\bar{m} \times \bar{a}\bar{m}} \\ \mathbf{T}_{(\bar{m}\langle\bar{b}\rangle)} \end{bmatrix} \\ \mathfrak{T}_L &\in \mathbb{C}^{(\bar{a}+2\bar{b})\bar{m} \times (\bar{a}+2\bar{b})\bar{m}} \quad \mathfrak{T}_R \in \mathbb{C}^{(\bar{a}+2\bar{b})\bar{m} \times (\bar{a}+2\bar{b})\bar{m}}\end{aligned}\quad (\text{A.49})$$

where $\mathbf{T}_{(\bar{m}\langle\bar{b}\rangle)} \in \mathbb{C}^{2\bar{b}\bar{m} \times 2\bar{b}\bar{m}}$ is based on the general matrix definition defined as (B.10) which we rewrite here:

$$\begin{aligned}\mathbf{T}_{(x\langle y\rangle)} &\in \mathbb{C}^{2yx \times 2yx} = \\ \frac{1}{\sqrt{2}} \begin{bmatrix} \mathcal{T}_{(x)} & & & \\ & \mathcal{T}_{(x)} & & \\ & & \ddots & \\ & & & \mathcal{T}_{(x)} \end{bmatrix}, \quad \mathcal{T}_{(x)} &= \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{x \times x} & -1i \cdot \mathbf{I}_{x \times x} \\ \mathbf{I}_{x \times x} & 1i \cdot \mathbf{I}_{x \times x} \end{bmatrix} \in \mathbb{C}^{2x \times 2x}\end{aligned}\quad (\text{A.50})$$

Applying the transformation would result in real system matrices:

$$\begin{aligned}\mathbf{A}_r &= \mathfrak{T}_L^H \cdot \mathbf{A}_c \cdot \mathfrak{T}_R \in \mathbb{R}^{(\bar{a}+2\bar{b})\bar{m} \times (\bar{a}+2\bar{b})\bar{m}} & \mathbf{B}_r &= \mathfrak{T}_L^H \cdot \mathbf{B}_c \in \mathbb{R}^{(\bar{a}+2\bar{b})\bar{m} \times \bar{m}} \\ \mathbf{C}_r &= \mathbf{C}_c \cdot \mathfrak{T}_R \in \mathbb{R}^{\bar{p} \times (\bar{a}+2\bar{b})\bar{m}} & \mathbf{D} &= \mathbf{r}_0 \in \mathbb{R}^{\bar{p} \times \bar{m}}\end{aligned}\quad (\text{A.51})$$

which can also be computed using the direct formula:

$$\begin{aligned}
 \mathbf{B}_r &= \left[\begin{array}{c} \mathbf{I}_{\bar{m} \times \bar{m}} \\ \vdots \\ \mathbf{I}_{\bar{m} \times \bar{m}} \\ \hline \left[\begin{array}{c} \sqrt{2} \cdot \mathbf{I}_{\bar{m} \times \bar{m}} \\ 0 \end{array} \right] \\ \vdots \\ \left[\begin{array}{c} \sqrt{2} \cdot \mathbf{I}_{\bar{m} \times \bar{m}} \\ 0 \end{array} \right] \end{array} \right], \quad \mathbf{A}_r = \left[\begin{array}{c|c} \boldsymbol{\rho}_{r1} & \cdots \\ \hline & \boldsymbol{\rho}_{r\bar{a}} \\ & \mathbf{p}_{c1} \\ & \ddots \\ & \mathbf{p}_{c\bar{b}} \end{array} \right] \\
 \mathbf{C}_r &= \left[\begin{array}{ccc|cc} \boldsymbol{\tau}_{r1} & \cdots & \boldsymbol{\tau}_{r\bar{a}} & \sqrt{2} \cdot \mathbf{r}_{c1} & \cdots & \sqrt{2} \cdot \mathbf{r}_{c\bar{b}} \end{array} \right] \tag{A.52} \\
 \mathbf{p}_{cb} &= \begin{bmatrix} \Re\{\boldsymbol{\rho}_{cb}\} & \Im\{\boldsymbol{\rho}_{cb}\} \\ -\Im\{\boldsymbol{\rho}_{cb}\} & \Re\{\boldsymbol{\rho}_{cb}\} \end{bmatrix}, \quad \mathbf{r}_{cb} = \begin{bmatrix} \Re\{\boldsymbol{\tau}_{cb}\} & \Im\{\boldsymbol{\tau}_{cb}\} \end{bmatrix}
 \end{aligned}$$

The matrix set $\{\mathbf{A}_r, \mathbf{B}_r, \mathbf{C}_r, \mathbf{D}_r\}$ forms the state-space transfer function equivalent of the initial barycentric rational equation (A.40)

Appendix B

Special Real Matrix Transformation

This chapter explains the process of transforming complex matrices to real matrices in the context of LM interpolation methods. The real matrix transformation process is based on [51], but elaborated in this appendix for a more complete description.

B.1 Single Matrix Unit Case

The special matrix structure which allows it to be transformed into a real matrix through an equivalent matrix transformation can be expressed as:

$$\begin{aligned} \mathbf{L} &= \begin{bmatrix} \boldsymbol{\alpha} & \boldsymbol{\beta} \\ \boldsymbol{\beta}^* & \boldsymbol{\alpha}^* \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times 2\bar{m}} \\ \mathbf{L} &= \begin{bmatrix} \boldsymbol{\alpha}_{re} + 1i \cdot \boldsymbol{\alpha}_{im} & \boldsymbol{\beta}_{re} + 1i \cdot \boldsymbol{\beta}_{im} \\ \boldsymbol{\beta}_{re} - 1i \cdot \boldsymbol{\beta}_{im} & \boldsymbol{\alpha}_{re} - 1i \cdot \boldsymbol{\alpha}_{im} \end{bmatrix} \end{aligned} \quad (\text{B.1})$$

$$\boldsymbol{\alpha}_{re} = \Re\{\boldsymbol{\alpha}\}, \boldsymbol{\alpha}_{im} = \Im\{\boldsymbol{\alpha}\}, \quad \boldsymbol{\beta}_{re} = \Re\{\boldsymbol{\beta}\}, \boldsymbol{\beta}_{im} = \Im\{\boldsymbol{\beta}\} \quad (\text{B.2})$$

$$\boldsymbol{\alpha} = \boldsymbol{\alpha}_{re} + 1i \cdot \boldsymbol{\alpha}_{im}, \quad \boldsymbol{\beta} = \boldsymbol{\beta}_{re} + 1i \cdot \boldsymbol{\beta}_{im}$$

$$\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathbb{C}^{\bar{p} \times \bar{m}}, \quad \boldsymbol{\alpha}_{re}, \boldsymbol{\beta}_{re}, \boldsymbol{\alpha}_{im}, \boldsymbol{\beta}_{im} \in \mathbb{R}^{\bar{p} \times \bar{m}}$$

where matrix \mathbf{L} in (B.1) is a generic matrix of no particular condition other than its given structure. Define the base unit cell of the unitary transformation matrix

which performs the real matrix transformation:

$$\mathcal{T}_{(x)} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{x \times x} & -1i \cdot \mathbf{I}_{x \times x} \\ \mathbf{I}_{x \times x} & 1i \cdot \mathbf{I}_{x \times x} \end{bmatrix} \in \mathbb{C}^{2x \times 2x} \quad (\text{B.3})$$

which has the easily verifiable inherent property:

$$\mathcal{T}_{(x)}^{-1} = \mathcal{T}_{(x)}^H \quad (\text{B.4})$$

The matrix \mathbf{L} can be turned real by the following matrix transformation:

$$\begin{aligned} & \mathcal{T}_{(\bar{p})}^H \cdot \mathbf{L} \cdot \mathcal{T}_{(\bar{m})} \\ &= \frac{1}{2} \begin{bmatrix} \mathbf{I}_{\bar{p} \times \bar{p}} & \mathbf{I}_{\bar{p} \times \bar{p}} \\ 1i \cdot \mathbf{I}_{\bar{p} \times \bar{p}} & -1i \cdot \mathbf{I}_{\bar{p} \times \bar{p}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_{re} + i\boldsymbol{\alpha}_{im} & \boldsymbol{\beta}_{re} + i\boldsymbol{\beta}_{im} \\ \boldsymbol{\beta}_{re} - i\boldsymbol{\beta}_{im} & \boldsymbol{\alpha}_{re} - i\boldsymbol{\alpha}_{im} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{\bar{m} \times \bar{m}} & -1i \cdot \mathbf{I}_{\bar{m} \times \bar{m}} \\ \mathbf{I}_{\bar{m} \times \bar{m}} & 1i \cdot \mathbf{I}_{\bar{m} \times \bar{m}} \end{bmatrix} = \\ & \frac{1}{2} \begin{bmatrix} \mathbf{I}_{\bar{p} \times \bar{p}} & \mathbf{I}_{\bar{p} \times \bar{p}} \\ 1i \cdot \mathbf{I}_{\bar{p} \times \bar{p}} & -1i \cdot \mathbf{I}_{\bar{p} \times \bar{p}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_{re} + i\boldsymbol{\alpha}_{im} + \boldsymbol{\beta}_{re} + i\boldsymbol{\beta}_{im} & -1i\boldsymbol{\alpha}_{re} + \boldsymbol{\alpha}_{im} + 1i\boldsymbol{\beta}_{re} - \boldsymbol{\beta}_{im} \\ \boldsymbol{\beta}_{re} - i\boldsymbol{\beta}_{im} + \boldsymbol{\alpha}_{re} - i\boldsymbol{\alpha}_{im} & -1i\boldsymbol{\beta}_{re} - \boldsymbol{\beta}_{im} + 1i\boldsymbol{\alpha}_{re} + \boldsymbol{\alpha}_{im} \end{bmatrix} \\ &= \frac{1}{2} \begin{bmatrix} 2\boldsymbol{\alpha}_{re} + 2\boldsymbol{\beta}_{re} & 2\boldsymbol{\alpha}_{im} - 2\boldsymbol{\beta}_{im} \\ -2\boldsymbol{\alpha}_{im} - 2\boldsymbol{\beta}_{im} & 2\boldsymbol{\alpha}_{re} - 2\boldsymbol{\beta}_{re} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\alpha}_{re} + \boldsymbol{\beta}_{re} & \boldsymbol{\alpha}_{im} - \boldsymbol{\beta}_{im} \\ -\boldsymbol{\alpha}_{im} - \boldsymbol{\beta}_{im} & \boldsymbol{\alpha}_{re} - \boldsymbol{\beta}_{re} \end{bmatrix} \\ & \mathcal{T}_{(\bar{p})}^H \cdot \mathbf{L} \cdot \mathcal{T}_{(\bar{m})} = \mathbf{L}_{re} = \begin{bmatrix} \boldsymbol{\alpha}_{re} + \boldsymbol{\beta}_{re} & \boldsymbol{\alpha}_{im} - \boldsymbol{\beta}_{im} \\ -\boldsymbol{\alpha}_{im} - \boldsymbol{\beta}_{im} & \boldsymbol{\alpha}_{re} - \boldsymbol{\beta}_{re} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times 2\bar{m}} \end{aligned} \quad (\text{B.5})$$

Incidentally, the matrices of the following forms also can be turned real using the same type of transformation matrices:

$$\mathbf{W} = \begin{bmatrix} \boldsymbol{\omega} & \boldsymbol{\omega}^* \end{bmatrix} \in \mathbb{C}^{\bar{p} \times 2\bar{m}}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times \bar{m}} \quad (\text{B.6})$$

The real transformation for the \mathbf{W} matrix is given as:

$$\begin{aligned}\mathbf{W} \cdot \mathcal{T}_{(\bar{m})} &= \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{\omega}_{re} + 1i \cdot \boldsymbol{\omega}_{im} & \boldsymbol{\omega}_{re} - 1i \cdot \boldsymbol{\omega}_{im} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{\bar{m} \times \bar{m}} & -1i \cdot \mathbf{I}_{\bar{m} \times \bar{m}} \\ \mathbf{I}_{\bar{m} \times \bar{m}} & 1i \cdot \mathbf{I}_{\bar{m} \times \bar{m}} \end{bmatrix} \\ \mathbf{W} \cdot \mathcal{T}_{(\bar{m})} &= \mathbf{W}_{re} = \begin{bmatrix} \sqrt{2} \cdot \boldsymbol{\omega}_{re} & \sqrt{2} \cdot \boldsymbol{\omega}_{im} \end{bmatrix} \in \mathbb{R}^{\bar{p} \times 2\bar{m}}, \quad \begin{aligned} \boldsymbol{\omega}_{re} &= \Re\{\boldsymbol{\omega}\} \\ \boldsymbol{\omega}_{im} &= \Im\{\boldsymbol{\omega}\} \end{aligned} \quad (\text{B.7})\end{aligned}$$

and the real transformation for the \mathbf{F} matrix is given as:

$$\begin{aligned}\mathcal{T}_{(\bar{p})}^H \cdot \mathbf{F} &= \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{\bar{p} \times \bar{p}} & \mathbf{I}_{\bar{p} \times \bar{p}} \\ 1i \cdot \mathbf{I}_{\bar{p} \times \bar{p}} & -1i \cdot \mathbf{I}_{\bar{p} \times \bar{p}} \end{bmatrix} \begin{bmatrix} \mathbf{f}_{re} + 1i \cdot \mathbf{f}_{im} \\ \mathbf{f}_{re} - 1i \cdot \mathbf{f}_{im} \end{bmatrix} \\ \mathcal{T}_{(\bar{p})}^H \cdot \mathbf{F} &= \mathbf{F}_{re} = \begin{bmatrix} \sqrt{2} \cdot \mathbf{f}_{re} \\ -\sqrt{2} \cdot \mathbf{f}_{im} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times \bar{m}}, \quad \begin{aligned} \mathbf{f}_{re} &= \Re\{\mathbf{f}\} \\ \mathbf{f}_{im} &= \Im\{\mathbf{f}\} \end{aligned}\end{aligned}$$

B.2 Full Matrix Case

We rewrite the matrix pattern (B.1) here with block matrix indexing:

$$\mathbf{L}_{\{i,j\}} = \begin{bmatrix} \boldsymbol{\alpha}_{\{i,j\}} & \boldsymbol{\beta}_{\{i,j\}} \\ \boldsymbol{\beta}_{\{i,j\}}^* & \boldsymbol{\alpha}_{\{i,j\}}^* \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times 2\bar{m}}$$

We do the same for the block row and block column matrices defined at (B.6):

$$\mathbf{W}_{\{j\}} = \begin{bmatrix} \boldsymbol{\omega}_{\{j\}} & \boldsymbol{\omega}_{\{j\}}^* \end{bmatrix} \in \mathbb{C}^{\bar{p} \times 2\bar{m}}, \quad \mathbf{F}_{\{i\}} = \begin{bmatrix} \mathbf{f}_{\{i\}} \\ \mathbf{f}_{\{i\}}^* \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times \bar{m}} \quad (\text{B.8})$$

In this thesis, the patterns $\mathbf{L}_{\{i,j\}}$, $\mathbf{W}_{\{j\}}$, and $\mathbf{F}_{\{i\}}$ are often repeated within larger matrices in the form of:

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_{\{1,1\}} & \mathbf{L}_{\{1,2\}} & \cdots & \mathbf{L}_{\{1,\bar{j}\}} \\ \mathbf{L}_{\{2,1\}} & \mathbf{L}_{\{2,2\}} & \cdots & \mathbf{L}_{\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{L}_{\{\bar{i},1\}} & \mathbf{L}_{\{\bar{i},2\}} & \cdots & \mathbf{L}_{\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{2\bar{i}\bar{p} \times 2\bar{j}\bar{m}}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{F}_{\{1\}} \\ \mathbf{F}_{\{2\}} \\ \cdots \\ \mathbf{F}_{\{\bar{i}\}} \end{bmatrix} \in \mathbb{C}^{2\bar{i}\bar{p} \times \bar{m}} \quad (\text{B.9})$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{\{1\}} & \mathbf{W}_{\{2\}} & \cdots & \mathbf{W}_{\{\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{\bar{p} \times 2\bar{j}\bar{m}}$$

In turn, these larger matrices can be turned real using the full transformation matrix defined as follow:

$$\mathbf{T}_{(x\langle y\rangle)} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathcal{T}_{(x)} & & & \\ & \mathcal{T}_{(x)} & & \\ & & \ddots & \\ & & & \mathcal{T}_{(x)} \end{bmatrix} \in \mathbb{C}^{2yx \times 2yx} \quad (\text{B.10})$$

which is a block diagonal matrix made by diagonally cascading the blocks $\mathcal{T}_{(x)} \in \mathbb{C}^{2x \times 2x}$ defined as (B.3) repeatedly y times. The real transform for \mathbf{L} can now be defined as:

$$\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \mathbf{L} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} =$$

$$\mathbf{L}_{re} = \begin{bmatrix} \mathbf{L}_{re\{1,1\}} & \mathbf{L}_{re\{1,2\}} & \cdots & \mathbf{L}_{re\{1,\bar{j}\}} \\ \mathbf{L}_{re\{2,1\}} & \mathbf{L}_{re\{2,2\}} & \cdots & \mathbf{L}_{re\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{L}_{re\{\bar{i},1\}} & \mathbf{L}_{re\{\bar{i},2\}} & \cdots & \mathbf{L}_{re\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{2\bar{i}\bar{p} \times 2\bar{j}\bar{m}} \quad (\text{B.11})$$

$$\mathbf{L}_{re\{i,j\}} = \begin{bmatrix} \Re\{\boldsymbol{\alpha}_{\{i,j\}}\} + \Re\{\boldsymbol{\beta}_{\{i,j\}}\} & \Im\{\boldsymbol{\alpha}_{\{i,j\}}\} - \Im\{\boldsymbol{\beta}_{\{i,j\}}\} \\ -\Im\{\boldsymbol{\alpha}_{\{i,j\}}\} - \Im\{\boldsymbol{\beta}_{\{i,j\}}\} & \Re\{\boldsymbol{\alpha}_{\{i,j\}}\} - \Re\{\boldsymbol{\beta}_{\{i,j\}}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times 2\bar{m}} \quad (\text{B.12})$$

The real transform for the \mathbf{W} and \mathbf{L} matrices are defined as:

$$\begin{aligned} \mathbf{W} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} &= \mathbf{W}_{re} = \begin{bmatrix} \mathbf{W}_{re\{1\}} & \mathbf{W}_{re\{2\}} & \cdots & \mathbf{W}_{re\{\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{\bar{p} \times 2\bar{j}\bar{m}} \\ \mathbf{W}_{re\{j\}} &= \begin{bmatrix} \sqrt{2} \cdot \Re\{\boldsymbol{\omega}_{\{j\}}\} & \sqrt{2} \cdot \Im\{\boldsymbol{\omega}_{\{j\}}\} \end{bmatrix} \in \mathbb{R}^{\bar{p} \times 2\bar{m}} \end{aligned} \quad (\text{B.13})$$

$$\mathcal{T}_{(\bar{p})}^H \cdot \mathbf{F} = \mathbf{F}_{re} = \begin{bmatrix} \mathbf{F}_{re\{1\}} \\ \mathbf{F}_{re\{2\}} \\ \vdots \\ \mathbf{F}_{re\{\bar{i}\}} \end{bmatrix} \in \mathbb{R}^{2\bar{i}\bar{p} \times \bar{m}}, \quad \mathbf{F}_{re\{j\}} = \begin{bmatrix} \sqrt{2} \cdot \Re\{\boldsymbol{f}_{\{i\}}\} \\ -\sqrt{2} \cdot \Im\{\boldsymbol{f}_{\{i\}}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times \bar{m}} \quad (\text{B.14})$$

B.3 Descriptor System Consideration

In this thesis, the SFLM (Section 2.2.3) and PSFLM (Chapter 5) methods use Loewner Matrices and offshoot matrices to directly replace the matrices in the descriptor system

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \quad (\text{B.15})$$

which are matrices having complex entries:

$$\mathbf{E}, \mathbf{A} \in \mathbb{C}^{2\bar{i}\bar{p} \times 2\bar{j}\bar{m}}, \quad \mathbf{C} \in \mathbb{C}^{\bar{p} \times 2\bar{j}\bar{m}}, \quad \mathbf{B} \in \mathbb{C}^{2\bar{i}\bar{p} \times \bar{m}}, \quad (\text{B.16})$$

Note \mathbf{D} is assumed real. Fortunately, the LM methods can manipulate the structure of the matrices such that \mathbf{E} and \mathbf{A} have similar structures as \mathbf{L} as defined in (B.9), and \mathbf{B} and \mathbf{C} have similar structures as \mathbf{F} and \mathbf{W} , respectively, as defined in (B.9). Further, the application of the real matrix transform (B.11), (B.13), and (B.14) does

not affect the outcome of the transfer function $\mathbf{G}(s)$ due to property (B.4) which states that the inverse of transformation matrices is their hermitian. To demonstrate this, assume $\bar{j} = \bar{i}$ and $\bar{p} = \bar{m}$ such that \mathbf{E} and \mathbf{A} are square and can have proper inverse. We perform the following derivation to show the introduction of $\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}$ and $\mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)}$ into $\mathbf{G}(s)$ does not perturb the transfer function:

$$\begin{aligned}
 \mathbf{G}(s) &= \mathbf{C} [s\mathbf{E} - \mathbf{A}]^{-1} \mathbf{B} + \mathbf{D} \\
 &= \mathbf{C} [\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)} (s\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \mathbf{E} \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} - s\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \mathbf{A} \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)}) \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)}^H]^{-1} \mathbf{B} + \mathbf{D} \\
 &= \mathbf{C} (\mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)}^H)^{-1} [(s\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \mathbf{E} \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} - s\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \mathbf{A} \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)})]^{-1} (\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)})^{-1} \mathbf{B} + \mathbf{D} \\
 &= \mathbf{C} \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} [(s\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \mathbf{E} \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} - s\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \mathbf{A} \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)})]^{-1} \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \mathbf{B} + \mathbf{D} \\
 &= \mathbf{C}_{re} [s\mathbf{E}_{re} - \mathbf{A}_{re}]^{-1} \mathbf{B}_{re} + \mathbf{D}
 \end{aligned} \tag{B.17}$$

where

$$\begin{aligned}
 \mathbf{C}_{re} &= \mathbf{C} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} \in \mathbb{R}^{\bar{p} \times 2\bar{j}\bar{m}}, \quad \mathbf{B}_{re} = \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \mathbf{B} \in \mathbb{R}^{2\bar{i}\bar{p} \times \bar{m}} \\
 \mathbf{E}_{re} &= \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \mathbf{E} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} \in \mathbb{R}^{2\bar{i}\bar{p} \times 2\bar{j}\bar{m}}, \quad \mathbf{A}_{re} = \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \mathbf{A} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} \in \mathbb{R}^{2\bar{i}\bar{p} \times 2\bar{j}\bar{m}}
 \end{aligned} \tag{B.18}$$

We note that in the square matrix case the transformation is a similarity transform.

B.4 Loewner Matrix Real Transform

Loewner Matrices (and their offshoot matrices) are generated as complex matrices because they are built using frequency data as building blocks. However, given the assumption that the frequency data conforms to a real physical system representable by a transfer function

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{E} - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D} \tag{B.19}$$

made of real matrices $\{\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$, frequency data have complex conjugacy ($\mathbf{G}(s^*) = \mathbf{G}(s)^*$). Data complex conjugacy allows the Loewner Matrix to be constructed with the complex conjugate padded initial data set [51]:

$$\begin{aligned} \Lambda &= \{\lambda_1, \lambda_1^*, \lambda_2, \lambda_2^*, \dots, \lambda_{\bar{j}}, \lambda_{\bar{j}}^*\} & \mathcal{H}_\Lambda &= \{\mathbf{H}(\lambda_j); \lambda_j \in \Lambda\} \\ \mathcal{M} &= \{\mu_1, \mu_1^*, \mu_2, \mu_2^*, \dots, \mu_{\bar{i}}, \mu_{\bar{i}}^*\} & \mathcal{H}_\mathcal{M} &= \{\mathbf{H}(\mu_i); \mu_i \in \mathcal{M}\} \\ \mathbf{H}(\lambda_j)^* &= \mathbf{H}(\lambda_j^*) \in \mathbb{C}^{\bar{p} \times \bar{m}} & \mathbf{H}(\mu_i)^* &= \mathbf{H}(\mu_i^*) \in \mathbb{C}^{\bar{p} \times \bar{m}} \end{aligned} \quad (\text{B.20})$$

Constructing the Loewner Matrix with the initial data set (B.20) using the Loewner Matrix unit block definition (2.18) that we rewrite here:

$$\mathbb{L}_{\{i,j\}} = \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{p} \times \bar{m}} \quad (\text{B.21})$$

would result in a Loewner Matrix \mathbb{L} of the exact structure as \mathbf{L} in (B.9):

$$\begin{aligned} \mathbb{L}_{cc} &= \begin{bmatrix} \mathbb{L}_{cc\{1,1\}} & \mathbb{L}_{cc\{1,2\}} & \cdots & \mathbb{L}_{cc\{1,\bar{j}\}} \\ \mathbb{L}_{cc\{2,1\}} & \mathbb{L}_{cc\{2,2\}} & \cdots & \mathbb{L}_{cc\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{cc\{\bar{i},1\}} & \mathbb{L}_{cc\{\bar{i},2\}} & \cdots & \mathbb{L}_{cc\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{2\bar{i}\bar{p} \times 2\bar{j}\bar{m}} \\ \mathbb{L}_{cc\{i,j\}} &= \begin{bmatrix} \mathbb{L}_{\alpha\{i,j\}} & \mathbb{L}_{\beta\{i,j\}} \\ \mathbb{L}_{\beta^*\{i,j\}} & \mathbb{L}_{\alpha^*\{i,j\}} \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times 2\bar{m}} \\ \mathbb{L}_{\alpha\{i,j\}} &= \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{p} \times \bar{m}}, \quad \mathbb{L}_{\beta\{i,j\}} = \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}^*}{\mu_i - \lambda_j^*} \in \mathbb{C}^{\bar{p} \times \bar{m}} \end{aligned} \quad (\text{B.22})$$

This structure of \mathbb{L}_{cc} allows it to be turned real by the same transformation as (B.11):

$$\begin{aligned} \mathbb{L}_{re} &= \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \mathbb{L}_{cc} \cdot \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} = \begin{bmatrix} \mathbb{L}_{re\{1,1\}} & \mathbb{L}_{re\{1,2\}} & \cdots & \mathbb{L}_{re\{1,\bar{j}\}} \\ \mathbb{L}_{re\{2,1\}} & \mathbb{L}_{re\{2,2\}} & \cdots & \mathbb{L}_{re\{2,\bar{j}\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{re\{\bar{i},1\}} & \mathbb{L}_{re\{\bar{i},2\}} & \cdots & \mathbb{L}_{re\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{2\bar{p}\times 2\bar{j}\bar{m}} \\ \mathbb{L}_{re\{i,j\}} &= \begin{bmatrix} \Re\{\mathbb{L}_{\alpha\{i,j\}}\} + \Re\{\mathbb{L}_{\beta\{i,j\}}\} & \Im\{\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\mathbb{L}_{\beta\{i,j\}}\} \\ -\Im\{\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\mathbb{L}_{\beta\{i,j\}}\} & \Re\{\mathbb{L}_{\alpha\{i,j\}}\} - \Re\{\mathbb{L}_{\beta\{i,j\}}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p}\times 2\bar{m}} \end{aligned} \quad (\text{B.23})$$

In the context of LM construction, there is the special direct current (DC, where $s = 0Hz$) case data. This s point is unique in that it is the only point where $s \in \mathbb{R}$ and the corresponding data $\mathbf{H}(s) = \mathbf{H}_0 \in \mathbb{R}^{\bar{p} \times \bar{m}}$ since at DC, there should be no signal delay of any kind. Inserting $s = 0$ in Λ or \mathcal{M} results in slightly different formulation of the real transform, but we perform the derivation for only the Λ case as demonstration as the \mathcal{M} derivation can be figured out following the Λ case process.

Update on the initial partitioning of data:

$$\begin{aligned} \Lambda &= \{0, \lambda_1, \lambda_1^*, \lambda_2, \lambda_2^*, \dots, \lambda_{\bar{j}}, \lambda_{\bar{j}}^*\} \quad \mathcal{H}_\Lambda = \{\mathbf{H}(\lambda_j); \lambda_j \in \Lambda\} \\ \mathcal{M} &= \{\mu_1, \mu_1^*, \mu_2, \mu_2^*, \dots, \mu_{\bar{i}}, \mu_{\bar{i}}^*\} \quad \mathcal{H}_\mathcal{M} = \{\mathbf{H}(\mu_i); \mu_i \in \mathcal{M}\} \\ \mathbf{H}(\lambda_j)^* &= \mathbf{H}(\lambda_j^*) \in \mathbb{C}^{\bar{p} \times \bar{m}} \quad \mathbf{H}(\mu_i)^* = \mathbf{H}(\mu_i^*) \in \mathbb{C}^{\bar{p} \times \bar{m}} \end{aligned} \quad (\text{B.24})$$

would result in the following complex LM following the LM formula (B.21):

$$\begin{aligned}
 \mathbb{L}_{cc} &= \begin{bmatrix} \mathbb{L}_{\Lambda 0cc\{1\}} & \mathbb{L}_{cc\{1,1\}} & \mathbb{L}_{cc\{1,2\}} & \cdots & \mathbb{L}_{cc\{1,\bar{j}\}} \\ \mathbb{L}_{\Lambda 0cc\{2\}} & \mathbb{L}_{cc\{2,1\}} & \mathbb{L}_{cc\{2,2\}} & \cdots & \mathbb{L}_{cc\{2,\bar{j}\}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{\Lambda 0cc\{\bar{i}\}} & \mathbb{L}_{cc\{\bar{i},1\}} & \mathbb{L}_{cc\{\bar{i},2\}} & \cdots & \mathbb{L}_{cc\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times (2\bar{j}+1)\bar{m}} \\
 \mathbb{L}_{cc\{i,j\}} &= \begin{bmatrix} \mathbb{L}_{\alpha\{i,j\}} & \mathbb{L}_{\beta\{i,j\}} \\ \mathbb{L}_{\beta^*\{i,j\}} & \mathbb{L}_{\alpha^*\{i,j\}} \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times 2\bar{m}}, \quad \mathbb{L}_{\Lambda 0cc\{i\}} = \begin{bmatrix} (\mathbf{H}_{\mu_i} - \mathbf{H}_0)/\mu_i \\ (\mathbf{H}_{\mu_i}^* - \mathbf{H}_0)/\mu_i^* \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times \bar{m}} \\
 \mathbb{L}_{\alpha\{i,j\}} &= \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{p} \times \bar{m}}, \quad \mathbb{L}_{\beta\{i,j\}} = \frac{\mathbf{H}_{\mu_i} - \mathbf{H}_{\lambda_j}^*}{\mu_i - \lambda_j^*} \in \mathbb{C}^{\bar{p} \times \bar{m}}
 \end{aligned} \tag{B.25}$$

The first \bar{m} sized block column of \mathbb{L}_{cc} can be denoted as:

$$\mathbb{L}_{\Lambda 0cc} = \begin{bmatrix} \mathbb{L}_{\Lambda 0cc\{1\}}^T & \mathbb{L}_{\Lambda 0cc\{2\}}^T & \cdots & \mathbb{L}_{\Lambda 0cc\{\bar{i}\}}^T \end{bmatrix}^T \in \mathbb{C}^{2\bar{p} \times \bar{m}} \tag{B.26}$$

and has the exact same structure as the \mathbf{F} block matrix vector defined in (B.9), which means that $\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H \cdot \mathbb{L}_{\Lambda 0cc} \in \mathbb{R}^{2\bar{p} \times \bar{m}}$. This means that the left-hand side transformation matrix $\mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)}^H$ alone turns the newly added block column real, so the only modification necessary for the real matrix transform is to add an identity matrix on the diagonal of the right hand side transformation matrix. The final

transformation formula goes as follow:

$$\mathbb{L}_{re} = \mathbf{T}_{\mathbb{L}L}^H \cdot \mathbb{L}_{cc} \cdot \mathbf{T}_{\mathbb{L}R} = \begin{bmatrix} \mathbb{L}_{\Lambda 0re\{1\}} & \mathbb{L}_{re\{1,1\}} & \mathbb{L}_{re\{1,2\}} & \cdots & \mathbb{L}_{re\{1,\bar{j}\}} \\ \mathbb{L}_{\Lambda 0re\{2\}} & \mathbb{L}_{re\{2,1\}} & \mathbb{L}_{re\{2,2\}} & \cdots & \mathbb{L}_{re\{2,\bar{j}\}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbb{L}_{\Lambda 0re\{\bar{i}\}} & \mathbb{L}_{re\{\bar{i},1\}} & \mathbb{L}_{re\{\bar{i},2\}} & \cdots & \mathbb{L}_{re\{\bar{i},\bar{j}\}} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times (2\bar{j}+1)\bar{m}} \quad (\text{B.27})$$

where

$$\mathbf{T}_{\mathbb{L}L} = \mathbf{T}_{(\bar{p}\langle\bar{i}\rangle)} \in \mathbb{C}^{2\bar{p} \times 2\bar{p}}, \quad \mathbf{T}_{\mathbb{L}R} = \begin{bmatrix} \mathbf{I}_{\bar{m} \times \bar{m}} \\ & \mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)} \end{bmatrix} \in \mathbb{C}^{2(\bar{j}+1)\bar{m} \times (2\bar{j}+1)\bar{m}} \quad (\text{B.28})$$

$$\mathbb{L}_{\Lambda 0re\{i\}} = \begin{bmatrix} \sqrt{2} \cdot \Re\{(\mathbf{H}_{\mu_i} - \mathbf{H}_0)/\mu_i\} \\ -\sqrt{2} \cdot \Im\{(\mathbf{H}_{\mu_i} - \mathbf{H}_0)/\mu_i\} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times \bar{m}} \quad (\text{B.29})$$

$$\mathbb{L}_{re\{i,j\}} = \begin{bmatrix} \Re\{\mathbb{L}_{\alpha\{i,j\}}\} + \Re\{\mathbb{L}_{\beta\{i,j\}}\} & \Im\{\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\mathbb{L}_{\beta\{i,j\}}\} \\ -\Im\{\mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\mathbb{L}_{\beta\{i,j\}}\} & \Re\{\mathbb{L}_{\alpha\{i,j\}}\} - \Re\{\mathbb{L}_{\beta\{i,j\}}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times 2\bar{m}} \quad (\text{B.30})$$

Given the process of real transform for the LM is shown in its primary form, we take the opportunity to directly define the real transform for the offshoot \mathbb{L} definitions.

B.4.1 Loewner Matrix Offshoot Cases

The first matrix we look at is the s -shifted Loewner Matrix $\sigma\mathbb{L}$ which was defined in (2.55) whose unit cell we rewrite here

$$\sigma\mathbb{L}_{\{i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i} - \lambda_j \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j} \in \mathbb{C}^{\bar{p} \times \bar{m}} \quad (\text{B.31})$$

Based on the initial data set (B.24), (B.31) leads to the full $\sigma\mathbb{L}$ matrix:

$$\sigma\mathbb{L}_{cc} = \begin{bmatrix} \sigma\mathbb{L}_{\Lambda 0cc\{1\}} & \sigma\mathbb{L}_{cc\{1,1\}} & \sigma\mathbb{L}_{cc\{1,2\}} & \cdots & \sigma\mathbb{L}_{cc\{1,\bar{j}\}} \\ \sigma\mathbb{L}_{\Lambda 0cc\{2\}} & \sigma\mathbb{L}_{cc\{2,1\}} & \sigma\mathbb{L}_{cc\{2,2\}} & \cdots & \sigma\mathbb{L}_{cc\{2,\bar{j}\}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sigma\mathbb{L}_{\Lambda 0cc\{\bar{i}\}} & \sigma\mathbb{L}_{cc\{\bar{i},1\}} & \sigma\mathbb{L}_{cc\{\bar{i},2\}} & \cdots & \sigma\mathbb{L}_{cc\{\bar{i},\bar{j}\}} \end{bmatrix} \quad (\text{B.32})$$

where

$$\sigma\mathbb{L}_{cc\{i,j\}} = \begin{bmatrix} \sigma\mathbb{L}_{\alpha\{i,j\}} & \sigma\mathbb{L}_{\beta\{i,j\}} \\ \sigma\mathbb{L}_{\beta}^*\{i,j\} & \sigma\mathbb{L}_{\alpha}^*\{i,j\} \end{bmatrix} \in \mathbb{C}^{2\bar{p} \times 2\bar{m}}, \quad \sigma\mathbb{L}_{\Lambda 0cc\{i\}} = \begin{bmatrix} \mathbf{H}_{\mu_i} \\ \mathbf{H}_{\mu_i}^* \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times \bar{m}} \quad (\text{B.33})$$

$$\sigma\mathbb{L}_{\alpha\{i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i} - \lambda_j \mathbf{H}_{\lambda_j}}{\mu_i - \lambda_j}, \quad \sigma\mathbb{L}_{\beta\{i,j\}} = \frac{\mu_i \mathbf{H}_{\mu_i} - \lambda_j^* \mathbf{H}_{\lambda_j}^*}{\mu_i - \lambda_j^*} \quad (\text{B.34})$$

$\sigma\mathbb{L}$ has the exact structure as the standard \mathbb{L} in the context of matrix real transform.

As such, $\sigma\mathbb{L}$'s real transform goes as follow:

$$\sigma\mathbb{L}_{re} = \mathbf{T}_{LR}^H \cdot \sigma\mathbb{L}_{cc} \cdot \mathbf{T}_{LL} = \begin{bmatrix} \sigma\mathbb{L}_{\Lambda 0re\{1\}} & \sigma\mathbb{L}_{re\{1,1\}} & \sigma\mathbb{L}_{re\{1,2\}} & \cdots & \sigma\mathbb{L}_{re\{1,\bar{j}\}} \\ \sigma\mathbb{L}_{\Lambda 0re\{2\}} & \sigma\mathbb{L}_{re\{2,1\}} & \sigma\mathbb{L}_{re\{2,2\}} & \cdots & \sigma\mathbb{L}_{re\{2,\bar{j}\}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sigma\mathbb{L}_{\Lambda 0re\{\bar{i}\}} & \sigma\mathbb{L}_{re\{\bar{i},1\}} & \sigma\mathbb{L}_{re\{\bar{i},2\}} & \cdots & \sigma\mathbb{L}_{re\{\bar{i},\bar{j}\}} \end{bmatrix} \quad (\text{B.35})$$

where

$$\sigma\mathbb{L}_{\Lambda 0re\{i\}} = \begin{bmatrix} \sqrt{2} \cdot \Re\{\mathbf{H}_{\mu_i}\} \\ -\sqrt{2} \cdot \Im\{\mathbf{H}_{\mu_i}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times \bar{m}} \quad (\text{B.36})$$

$$\sigma \mathbb{L}_{re\{i,j\}} = \begin{bmatrix} \Re\{\sigma \mathbb{L}_{\alpha\{i,j\}}\} + \Re\{\sigma \mathbb{L}_{\beta\{i,j\}}\} & \Im\{\sigma \mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\sigma \mathbb{L}_{\beta\{i,j\}}\} \\ -\Im\{\sigma \mathbb{L}_{\alpha\{i,j\}}\} - \Im\{\sigma \mathbb{L}_{\beta\{i,j\}}\} & \Re\{\sigma \mathbb{L}_{\alpha\{i,j\}}\} - \Re\{\sigma \mathbb{L}_{\beta\{i,j\}}\} \end{bmatrix} \in \mathbb{R}^{2\bar{p} \times 2\bar{m}} \quad (\text{B.37})$$

The second matrix we look at is the Λ data row vector matrix \mathbf{W} defined at (2.60). Based on the initial data set (B.24), \mathbf{W} takes the following form:

$$\mathbf{W}_{cc} = \begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_{\lambda_1} & \mathbf{H}_{\lambda_1}^* & \cdots & \mathbf{H}_{\lambda_{\bar{j}}} & \mathbf{H}_{\lambda_{\bar{j}}}^* \end{bmatrix} \in \mathbb{C}^{\bar{p} \times (2\bar{j}+1)\bar{m}} \quad (\text{B.38})$$

\mathbf{H}_0 is the only purely real matrix in \mathbf{W}_{cc} and removing it from \mathbf{W}_{cc} would turn \mathbf{W}_{cc} to a block row matrix of the same form as \mathbf{W} defined in (B.9), so a simple additional $\mathbf{I}_{\bar{m} \times \bar{m}}$ diagonally cascaded at the head of $\mathbf{T}_{(\bar{m}\langle\bar{j}\rangle)}$ would suffice to take into account the presence of \mathbf{H}_0 in \mathbf{W}_{cc} . This means that the real transformation of \mathbf{W}_{cc} can be performed as follow:

$$\begin{aligned} \mathbf{W}_{cc} \cdot \mathbf{T}_{\mathbb{L}R} &= \mathbf{W}_{re} \in \mathbb{R}^{\bar{p} \times (2\bar{j}+1)\bar{m}} \\ \mathbf{W}_{re} &= \begin{bmatrix} \mathbf{H}_0 & \sqrt{2} \cdot \Re\{\mathbf{H}_{\lambda_1}\} & \sqrt{2} \cdot \Im\{\mathbf{H}_{\lambda_1}\} & \cdots & \sqrt{2} \cdot \Re\{\mathbf{H}_{\lambda_{\bar{j}}}\} & \sqrt{2} \cdot \Im\{\mathbf{H}_{\lambda_{\bar{j}}}\} \end{bmatrix} \end{aligned} \quad (\text{B.39})$$

The third matrix we look at is the \mathcal{M} data column vector matrix \mathbf{F} defined at (2.61). Based on the initial data set (B.24), \mathbf{F} takes the following form:

$$\mathbf{F}_{cc} = \begin{bmatrix} \mathbf{H}_{\mu_1}^T & (\mathbf{H}_{\mu_1}^*)^T & \cdots & \mathbf{H}_{\mu_{\bar{i}}}^T & (\mathbf{H}_{\mu_{\bar{i}}}^*)^T \end{bmatrix}^T \in \mathbb{C}^{2\bar{i}\bar{p} \times \bar{m}} \quad (\text{B.40})$$

\mathbf{F}_{cc} has the exact structure as the \mathbf{F} matrix vector defined in (B.9), which means we can use the same real matrix transform as the one used in (B.14) on \mathbf{F}_{cc} :

$$\begin{aligned} \mathcal{T}_{(\bar{p})}^H \cdot \mathbf{F}_{cc} &= \mathbf{F}_{re} \in \mathbb{R}^{2i\bar{p} \times \bar{m}} \\ \mathbf{F}_{re} &= \left[\sqrt{2} \cdot \Re\{\mathbf{H}_{\mu_1}\}^T \quad \sqrt{2} \cdot \Im\{\mathbf{H}_{\mu_1}\}^T \quad \dots \quad \sqrt{2} \cdot \Re\{\mathbf{H}_{\mu_{\bar{j}}}\}^T \quad \sqrt{2} \cdot \Im\{\mathbf{H}_{\mu_{\bar{j}}}\}^T \right]^T \end{aligned} \quad (\text{B.41})$$

References

- [1] Allen Taflove and Susan C. Hagness. *Computational Electrodynamics: The Finite-Difference Time-Domain Method*. Artech House, 2005.
- [2] Jian-Ming Jin. *The Finite Element Method in Electromagnetics*. Wiley-IEEE Press, 2014.
- [3] Laurence W. Nagel. *SPICE2: A Computer Program to Simulate Semiconductor Circuits*. PhD thesis, EECS Department, University of California, Berkeley, May 1975.
- [4] T. Quarles. The spice3 implementation guide. Technical Report UCB/ERL M89/44, EECS Department, University of California, Berkeley, Apr 1989.
- [5] A.E. Ruehli and A.C. Cangellaris. Progress in the methodologies for the electrical modeling of interconnects and electronic packages. *Proceedings of the IEEE*, 89(5):740–771, 2001.
- [6] H.B. Bakoglu. *Circuits, Interconnections, and Packaging for VLSI*. Addison-Wesley VLSI systems series. Addison-Wesley Publishing Company, 1990.
- [7] A. Deutsch. Electrical characteristics of interconnections for high-performance systems. *Proceedings of the IEEE*, 86(2):315–357, 1998.
- [8] C.R. Paul. *Analysis of Multiconductor Transmission Lines*. Wiley Series in Microwave and Optical Engineering. Wiley, 1994.
- [9] R. Achar and M.S. Nakhla. Simulation of high-speed interconnects. *Proceedings of the IEEE*, 89(5):693–728, 2001.
- [10] R. Paul Clayton. *Analysis of Multiconductor Transmission Lines*. Wiley-IEEE Press, 2007.
- [11] A. Dounavis, R. Achar, and M.S. Nakhla. Efficient passive circuit models for distributed networks with frequency-dependent parameters. *IEEE Transactions on Advanced Packaging*, 23(3):382–392, 2000.

- [12] Chu-Sun Yen, Z. Fazarinc, and R.L. Wheeler. Time-domain skin-effect model for transient analysis of lossy transmission lines. *Proceedings of the IEEE*, 70(7):750–757, 1982.
- [13] Robert E. Collin. *Field Theory of Guided Waves*. Wiley-IEEE Press, 1991.
- [14] Simon Ramo, John R. Whinnery, and Theodore Van Duzer. *Fields and Waves in Communication Electronics*. Wiley, 1994.
- [15] A. Deutsch, H.H. Smith, C.W. Surovic, G.V. Kopcsay, D.A. Webber, P.W. Coteus, G.A. Katopis, W.D. Becker, A.H. Dansky, G.A. Sai-Halasz, and P.J. Restle. Frequency-dependent crosstalk simulation for on-chip interconnections. *IEEE Transactions on Advanced Packaging*, 22(3):292–308, 1999.
- [16] M. Gibbons, A.C. Cangellaris, and J.L. Prince. Finite-difference time-domain analysis of pulse propagation in multichip module interconnects. *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, 16(5):490–498, 1993.
- [17] A.C. Cangellaris, M. Gibbons, and J.L. Prince. Electrical characteristics of multichip module interconnects with perforated reference planes. *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, 16(1):113–118, 1993.
- [18] Allen Taflove. *Advances in Computational Electrodynamics: The Finite-Difference Time-Domain Method*. Artech House, 1998.
- [19] Chien-Nan Kuo, B. Houshmand, and T. Itoh. Full-wave analysis of packaged microwave circuits with active and nonlinear devices: an fdtd approach. *IEEE Transactions on Microwave Theory and Techniques*, 45(5):819–826, 1997.
- [20] Allen Taflove and Korada R. Umashankar. The finite-difference time-domain method for numerical modeling of electromagnetic wave interactions. *Electromagnetics*, 10(1-2):105–126, 1990.
- [21] Yui-Sheng Tsuei, A.C. Cangellaris, and J.L. Prince. Rigorous electromagnetic modeling of chip-to-package (first-level) interconnections. *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, 16(8):876–883, 1993.
- [22] A.E. Ruehli and H. Heeb. Circuit models for three-dimensional geometries including dielectrics. *IEEE Transactions on Microwave Theory and Techniques*, 40(7):1507–1516, 1992.

- [23] A. Bellen, N. Guglielmi, and A.E. Ruehli. Methods for linear systems of circuit delay differential equations of neutral type. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 46(1):212–215, 1999.
- [24] Pierce A. Brennan, Norman Raver, and Albert E. Ruehli. Three-dimensional inductance computations with partial element equivalent circuits. *IBM Journal of Research and Development*, 23(6):661–668, 1979.
- [25] L.M. Silveira, M. Kamon, and J. White. Direct computation of reduced-order models for circuit simulation of 3-d interconnect structures. In *Proceedings of 1994 IEEE Electrical Performance of Electronic Packaging*, pages 245–248, 1994.
- [26] A.C. Cangellaris, M. Celik, S. Pasha, and Li Zhao. Electromagnetic model order reduction for system-level modeling. *IEEE Transactions on Microwave Theory and Techniques*, 47(6):840–850, 1999.
- [27] A. C. Cangellaris and L. Zhao. Model order reduction techniques for electromagnetic macromodelling based on finite methods. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 13(2-3):181–197, 2000.
- [28] S.P. McCormick and J. Allen. Waveform moment methods for improved interconnection analysis. In *27th ACM/IEEE Design Automation Conference*, pages 406–412, 1990.
- [29] L.T. Pillage and R.A. Rohrer. Asymptotic waveform evaluation for timing analysis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 9(4):352–366, 1990.
- [30] E. Chiprout and M.S. Nakhla. *Asymptotic Waveform Evaluation and Moment Matching for Interconnect Analysis*. Springer New York, NY, 1993.
- [31] L. Miguel Silveira, M. Kamon, and J. White. Efficient reduced-order modeling of frequency-dependent coupling inductances associated with 3-d interconnect structures. *IEEE Transactions on Components, Packaging, and Manufacturing Technology: Part B*, 19(2):283–288, 1996.
- [32] R.W. Freund and P. Feldmann. Small-signal circuit analysis and sensitivity computations with the pvl algorithm. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 43(8):577–585, 1996.

- [33] K. Gallivan, G. Grimme, and P. Van Dooren. A rational lanczos algorithm for model reduction. *Numerical Algorithms*, 12:33–63, 1996.
- [34] A. Odabasioglu, M. Celik, and L.T. Pileggi. Prima: passive reduced-order interconnect macromodeling algorithm. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 17(8):645–654, 1998.
- [35] Athanasios C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. Society for Industrial and Applied Mathematics, 2005.
- [36] Peter Benner, Micheal Hinze, and Maten E. J. W. ter. *Model Reduction for Circuit Simulation*. Springer Dordrecht, 2011.
- [37] Daniel Ioan and Gabriela Ciuprina. *Reduced Order Models of On-Chip Passive Components and Interconnects, Workbench and Test Structures*, pages 447–467. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [38] Sheldon Tan and Lei He. *Advanced Model Order Reduction Techniques in VLSI Design*. Cambridge University Press, 2007.
- [39] 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*, 45(10):1941–1947, Oct 1997.
- [40] S.D. Corey and A.T. Yang. Interconnect characterization using time-domain reflectometry. *IEEE Transactions on Microwave Theory and Techniques*, 43(9):2151–2156, 1995.
- [41] L.P. Vakanas, A.C. Cangellaris, and O.A. Palusinski. Scattering parameter-based simulation of transients in lossy nonlinearly terminated packaging interconnections. *IEEE Transactions on Components, Packaging, and Manufacturing Technology: Part B*, 17(4):472–479, 1994.
- [42] B. Gustavsen and A. Semlyen. Rational approximation of frequency domain responses by vector fitting. *IEEE Transactions on Power Delivery*, 14(3):1052–1061, 1999.
- [43] M. Elzinga, K. L. Virga, and J. L. Prince. Improved global rational approximation macromodeling algorithm for networks characterized by frequency-sampled data. *IEEE Transactions on Microwave Theory and Techniques*, 48(9):1461–1468, Sep. 2000.

- [44] C. P. Coelho, J. R. Phillips, and L. M. Silveira. Passive constrained rational approximation algorithm using nevanlinna-pick interpolation. In *Proceedings 2002 Design, Automation and Test in Europe Conference and Exhibition*, pages 923–930, March 2002.
- [45] C.P. Coelho, J. Phillips, and L.M. Silveira. A convex programming approach for generating guaranteed passive approximations to tabulated frequency-data. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 23(2):293–301, 2004.
- [46] B. Gustavsen and A. Semlyen. A robust approach for system identification in the frequency domain. *IEEE Transactions on Power Delivery*, 19(3):1167–1173, 2004.
- [47] Giulio Antonini, Dirk Deschrijver, and Tom Dhaene. A comparative study of vector fitting and orthonormal vector fitting techniques for emc applications. *IEEE International Symposium on Electromagnetic Compatibility*, 1, 01 2006.
- [48] B. Gustavsen. Improving the pole relocating properties of vector fitting. *IEEE Transactions on Power Delivery*, 21(3):1587–1592, 2006.
- [49] A.J. Mayo and A.C. Antoulas. A framework for the solution of the generalized realization problem. *Linear Algebra and its Applications*, 425(2):634 – 662, 2007. Special Issue in honor of Paul Fuhrmann.
- [50] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter. Macromodeling of multiport systems using a fast implementation of the vector fitting method. *IEEE Microwave and Wireless Components Letters*, 18(6):383–385, 2008.
- [51] Sanda Lefteriu and Athanasios C. Antoulas. A new approach to modeling multiport systems from frequency-domain data. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 29(1):14–27, 2010.
- [52] S. Lefteriu and A. C. Antoulas. Modeling multi-port systems from frequency response data via tangential interpolation. In *2009 IEEE Workshop on Signal Propagation on Interconnects*, pages 1–4, May 2009.
- [53] Y. Wang, C. Lei, G. K. H. Pang, and N. Wong. Mfti: Matrix-format tangential interpolation for modeling multi-port systems. In *Design Automation Conference*, pages 683–686, June 2010.

- [54] M. Kabir and R. Khazaka. Macromodeling of distributed networks from frequency-domain data using the loewner matrix approach. *IEEE Transactions on Microwave Theory and Techniques*, 60(12):3927–3938, Dec 2012.
- [55] Muhammad Kabir, Roni Khazaka, Ramachandra Achar, and Michel Nakhla. Loewner-matrix based efficient algorithm for frequency sweep of high-speed modules. In *2012 IEEE 21st Conference on Electrical Performance of Electronic Packaging and Systems*, pages 185–188, 2012.
- [56] M. Kabir and R. Khazaka. Parametric macromodeling of high-speed modules from frequency-domain data using loewner matrix based method. In *2013 IEEE MTT-S International Microwave Symposium Digest (MTT)*, pages 1–4, June 2013.
- [57] Muhammad Kabir and Roni Khazaka. Order selection for loewner matrix based macromodels for accurate macromodeling of distributed high-speed modules from limited number of full-wave s-parameter data. In *2014 IEEE 18th Workshop on Signal and Power Integrity (SPI)*, pages 1–4, 2014.
- [58] M. Kabir and R. Khazaka. Loewner matrix macromodeling for y-parameter data with a priori \mathbf{D} matrix extraction. *IEEE Transactions on Microwave Theory and Techniques*, 64(12):4098–4107, Dec 2016.
- [59] M. Kabir, Y. Q. Xiao, and R. Khazaka. Loewner matrix interpolation for noisy s-parameter data. In *2016 IEEE 25th Conference on Electrical Performance Of Electronic Packaging And Systems (EPEPS)*, pages 95–98, Oct 2016.
- [60] M. T. Kassis, M. Kabir, Y. Q. Xiao, and R. Khazaka. Passive reduced order macromodeling based on loewner matrix interpolation. *IEEE Transactions on Microwave Theory and Techniques*, 64(8):2423–2432, Aug 2016.
- [61] M. Sahouli and A. Dounavis. Delay extraction-based modeling using loewner matrix framework. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 7(3):424–433, March 2017.
- [62] A. Hochman. Fastaaa: A fast rational-function fitter. In *2017 IEEE 26th Conference on Electrical Performance of Electronic Packaging and Systems (EPEPS)*, pages 1–3, Oct 2017.

- [63] Yuji Nakatsukasa, Olivier Sète, and Lloyd N. Trefethen. The aaa algorithm for rational approximation. *SIAM Journal on Scientific Computing*, 40(3):A1494–A1522, 2018.
- [64] M. Sahouli, S. Wahid, and A. Dounavis. Iterative loewner matrix macromodeling approach for noisy frequency responses. *IEEE Transactions on Microwave Theory and Techniques*, 67(2):634–641, Feb 2019.
- [65] Yi Qing Xiao, Stefano Grivet-Talocia, Paolo Manfredi, and Roni Khazaka. A novel framework for parametric loewner matrix interpolation. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 9(12):2404–2417, 2019.
- [66] W. Hendrickx and T. Dhaene. A discussion of "rational approximation of frequency domain responses by vector fitting". *IEEE Transactions on Power Systems*, 21(1):441–443, 2006.
- [67] T. Bradde, S. Grivet-Talocia, M. De Stefano, and A. Zanco. A scalable reduced-order modeling algorithm for the construction of parameterized interconnect macromodels from scattering responses. In *2018 IEEE Symposium on Electromagnetic Compatibility, Signal Integrity and Power Integrity (EMC, SI PI)*, pages 650–655, July 2018.
- [68] Alessandro Zanco, Stefano Grivet-Talocia, Tommaso Bradde, and Marco De Stefano. Uniformly stable parameterized macromodeling through positive definite basis functions. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 10(11):1782–1794, 2020.
- [69] Alessandro Zanco and Stefano Grivet-Talocia. Toward fully automated high-dimensional parameterized macromodeling. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 11(9):1402–1416, 2021.
- [70] Tommaso Bradde, Stefano Grivet-Talocia, Alessandro Zanco, and Giuseppe C. Calafiore. Data-driven extraction of uniformly stable and passive parameterized macromodels. *IEEE Access*, 10:15786–15804, 2022.
- [71] A. C. Antoulas and B. D. Q. Anderson. On the scalar rational interpolation problem. *IMA Journal of Mathematical Control and Information*, 3(2-3):61–88, 1986.

- [72] B. D. O. Anderson and A. C. Antoulas. Rational interpolation and state-variable realizations. *Linear Algebra and its Applications*, 137-138:479 – 509, 1990.
- [73] A. C. Antoulas and B. D. O. Anderson. *State-space and Polynomial approaches to Rational Interpolation*. Birkhäuser Boston, Boston, MA, 1990.
- [74] Ion Victor Gosea and Athanasios C. Antoulas. Stability preserving post-processing methods applied in the loewner framework. In *2016 IEEE 20th Workshop on Signal and Power Integrity (SPI)*, pages 1–4, 2016.
- [75] Peter Benner, Mario Ohlberger, Albert Cohen, and Karen Willcox. *Model Reduction and Approximation*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2017.
- [76] Steven Elsworth and Stefan Güttel. Conversions between barycentric, rkfun, and newton representations of rational interpolants, 2017.
- [77] Mark Embree and A. Cosmin Ionita. Pseudospectra of loewner matrix pencils, 2019.
- [78] Lucas Monzón, William Johns, Spatika Iyengar, Matthew Reynolds, Jonathan Maack, and Kumaraguru Prabakar. A multi-function aaa algorithm applied to frequency dependent line modeling. In *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pages 1–5, 2020.
- [79] A. C. Antoulas, C. A. Beattie, and S. Gügercin. *Interpolatory Methods for Model Reduction*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2020.
- [80] Joel D. Simard and Alessandro Astolfi. Nonlinear model reduction in the loewner framework. *IEEE Transactions on Automatic Control*, 66(12):5711–5726, 2021.
- [81] Luis Ernesto Carrera-Retana, Mario Marin-Sanchez, Christian Schuster, and Renato Rimolo-Donadio. Improving accuracy after stability enforcement in the loewner matrix framework. *IEEE Transactions on Microwave Theory and Techniques*, 70(2):1037–1047, 2022.
- [82] Ion Victor Gosea and Stefan Güttel. Algorithms for the rational approximation of matrix-valued functions, 2021.

- [83] Pieter Lietaert, Karl Meerbergen, Javier Pérez, and Bart Vandereycken. Automatic rational approximation and linearization of nonlinear eigenvalue problems. *IMA Journal of Numerical Analysis*, 42(2):1087–1115, 02 2021.
- [84] Andrea Carracedo Rodriguez, Linus Balicki, and Serkan Gugercin. The p-aaa algorithm for data driven modeling of parametric dynamical systems, 2022.
- [85] Tea Vojkovic, David Quero, Charles Poussot-Vassal, and Pierre Vuillemin. Low-order parametric state-space modeling of mimo systems in the loewner framework. *SIAM Journal on Applied Dynamical Systems*, 22(4):3130–3164, 2023.
- [86] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, June 1994.
- [87] D. Saraswat, R. Achar, and M. Nakhla. Enforcing passivity for rational function based macromodels of tabulated data. In *Electrical Performance of Electrical Packaging (IEEE Cat. No. 03TH8710)*, pages 295–298, 2003.
- [88] S. Grivet-Talocia. Passivity enforcement via perturbation of hamiltonian matrices. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 51(9):1755–1769, 2004.
- [89] D. Saraswat, R. Achar, and M.S. Nakhla. A fast algorithm and practical considerations for passive macromodeling of measured/simulated data. *IEEE Transactions on Advanced Packaging*, 27(1):57–70, 2004.
- [90] D. Saraswat, R. Achar, and M.S. Nakhla. Global passivity enforcement algorithm for macromodels of interconnect subnetworks characterized by tabulated data. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 13(7):819–832, 2005.
- [91] D.C. Sorensen. Passivity preserving model reduction via interpolation of spectral zeros. *Systems & Control Letters*, 54(4):347–360, 2005.
- [92] Bjorn Gustavsen. Fast passivity enforcement for pole-residue models by perturbation of residue matrix eigenvalues. *IEEE Transactions on Power Delivery*, 23(4):2278–2285, 2008.
- [93] Zuochang Ye, Luis Miguel Silveira, and Joel R. Phillips. Fast and reliable passivity assessment and enforcement with extended hamiltonian pencil. In *2009*

- IEEE/ACM International Conference on Computer-Aided Design - Digest of Technical Papers*, pages 774–778, 2009.
- [94] Bjorn Gustavsen. Fast passivity enforcement for s-parameter models by perturbation of residue matrix eigenvalues. *IEEE Transactions on Advanced Packaging*, 33(1):257–265, 2010.
 - [95] Yuanzhe Wang, Zheng Zhang, Cheng-Kok Koh, Guoyong Shi, Grantham K. H. Pang, and Ngai Wong. Passivity enforcement for descriptor systems via matrix pencil perturbation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 31(4):532–545, 2012.
 - [96] Salvatore Bernardo Olivadese and Stefano Grivet-Talocia. Compressed passive macromodeling. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2(8):1378–1388, 2012.
 - [97] Yi Qing Xiao, Muhammad Kabir, Marco T. Kassis, and Roni Khazaka. Passive reduced order macromodeling based on admittance parameter using hamiltonian-symplectic matrix pencil perturbation. In *2015 IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB)*, pages 1–4, 2015.
 - [98] Marco T. Kassis, Muhammad Kabir, and Roni Khazaka. Passive reduced order macromodeling using hamiltonian matrix pencil perturbation. In *2015 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)*, pages 1–3, 2015.
 - [99] Stefano Grivet-Talocia and Bjørn Gustavsen. *Passive Macromodeling: Theory and Applications*. Wiley, 01 2016.
 - [100] Alessandro Zanco, Stefano Grivet-Talocia, Tommaso Bradde, and Marco De Stefano. Enforcing passivity of parameterized lti macromodels via hamiltonian-driven multivariate adaptive sampling. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 39(1):225–238, 2020.
 - [101] D. Deschrijver and T. Dhaene. Stability and passivity enforcement of parametric macromodels in time and frequency domain. *IEEE Transactions on Microwave Theory and Techniques*, 56(11):2435–2441, 2008.

- [102] D. Deschrijver, T. Dhaene, and D. De Zutter. Robust parametric macromodeling using multivariate orthonormal vector fitting. *IEEE Transactions on Microwave Theory and Techniques*, 56(7):1661–1667, 2008.
- [103] T. Dhaene and D. Deschrijver. Stable parametric macromodeling using a recursive implementation of the vector fitting algorithm. *IEEE Microwave and Wireless Components Letters*, 19(2):59–61, 2009.
- [104] P. Triverio, S. Grivet-Talocia, and M. S. Nakhla. A parameterized macromodeling strategy with uniform stability test. *IEEE Transactions on Advanced Packaging*, 32(1):205–215, 2009.
- [105] F. Ferranti, L. Knockaert, T. Dhaene, and G. Antonini. Passivity-preserving parametric macromodeling for highly dynamic tabulated data based on lur'e equations. *IEEE Transactions on Microwave Theory and Techniques*, 58(12):3688–3696, 2010.
- [106] F. Ferranti, L. Knockaert, and T. Dhaene. Guaranteed passive parameterized admittance-based macromodeling. *IEEE Transactions on Advanced Packaging*, 33(3):623–629, 2010.
- [107] P. Triverio, M. S. Nakhla, and S. Grivet-Talocia. Passive parametric macromodeling from sampled frequency data. In *2010 IEEE 14th Workshop on Signal Propagation on Interconnects*, pages 117–120, 2010.
- [108] P. Triverio, M. Nakhla, and S. Grivet-Talocia. Passive parametric modeling of interconnects and packaging components from sampled impedance, admittance or scattering data. In *3rd Electronics System Integration Technology Conference ESTC*, pages 1–6, 2010.
- [109] F. Ferranti, L. Knockaert, and T. Dhaene. Passivity-preserving parametric macromodeling by means of scaled and shifted state-space systems. *IEEE Transactions on Microwave Theory and Techniques*, 59(10):2394–2403, 2011.
- [110] F. Ferranti, T. Dhaene, and L. Knockaert. Compact and passive parametric macromodeling using reference macromodels and positive interpolation operators. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2(12):2080–2088, 2012.
- [111] E. R. Samuel, L. Knockaert, F. Ferranti, and T. Dhaene. Guaranteed passive parameterized macromodeling by using sylvester state-space realizations. *IEEE*

Transactions on Microwave Theory and Techniques, 61(4):1444–1454, April 2013.

- [112] A.C. Antoulas, A.C. Ionita, and S. Lefteriu. On two-variable rational interpolation. *Linear Algebra and its Applications*, 436(8):2889 – 2915, 2012.
- [113] M. Kabir and R. Khazaka. Fixed-order parametric macromodeling of interconnects from s-parameter data using loewner matrix based method. In *2013 IEEE 22nd Conference on Electrical Performance of Electronic Packaging and Systems*, pages 141–144, Oct 2013.
- [114] A. Ionita and A. Antoulas. Data-driven parametrized model reduction in the loewner framework. *SIAM Journal on Scientific Computing*, 36(3):A984–A1007, 2014.
- [115] Tatjana Stykel. Analysis and numerical solution of generalized lyapunov equations. *Institut für Mathematik, Technische Universität, Berlin*, 2002.
- [116] David M. Pozar. *Microwave Engineering*. Wiley, USA, 2012.
- [117] K. Kurokawa. Power waves and the scattering matrix. *IEEE Transactions on Microwave Theory and Techniques*, 13(2):194–202, 1965.
- [118] Liyi Dai. *Singular Control Systems*. Springer Berlin Heidelberg, 1989.
- [119] Lotfi A. Zadeh and Charles A. Desoer. *Linear System Theory: The State Space Approach*. McGraw-Hill Book Company, Inc, 1963.
- [120] William L. Brogan. *Modern Control Theory*. Pearson, 1991.
- [121] Alan V. Oppenheim, Alan S. Willsky, and S. Hamid Nawab. *Signals and Systems (2nd Ed.)*. Prentice-Hall, Inc., USA, 1996.
- [122] M. R. Wohlers. *Lumped and Distributed Passive Networks*. Academic Press, New York and London, 1969.
- [123] D. C. Youla, L. J. Castriota, and H. J. Carlin. Bounded real scattering matrices and the foundations of linear passive network theory. *Circuit Theory, IRE Transactions on*, 6(1):102–124, 1959.
- [124] R.F. Curtain. Old and new perspectives on the positive-real lemma in systems and control theory. *ZAMM Zeitschrift fur Angewandte Mathematik und Mechanik*, 79(9):579 – 590, 1999. Cited by: 19.

- [125] J.C. Willems. Dissipative dynamical systems part i: General theory. *Arch. Rational Mech Anal.*, 45:321–351, 1972.
- [126] Jean-Paul Berrut and Lloyd N. Trefethen. Barycentric lagrange interpolation. *SIAM Review*, 46(3):501–517, 2004.
- [127] C. Sanathanan and J. Koerner. Transfer function synthesis as a ratio of two complex polynomials. *IEEE Transactions on Automatic Control*, 8(1):56–58, 1963.
- [128] P. Triverio, M. S. Nakhla, and S. Grivet-Talocia. Parametric macromodeling of multiport networks from tabulated data. In *2007 IEEE Electrical Performance of Electronic Packaging*, pages 51–54, 2007.
- [129] P. Triverio, S. Grivet-Talocia, and M. S. Nakhla. An improved fitting algorithm for parametric macromodeling from tabulated data. In *2008 12th IEEE Workshop on Signal Propagation on Interconnects*, pages 1–4, 2008.
- [130] S. Grivet-Talocia. A perturbation scheme for passivity verification and enforcement of parameterized macromodels. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 7(11):1869–1881, Nov 2017.
- [131] S. Grivet-Talocia and R. Trinchero. Behavioral, parameterized, and broadband modeling of wired interconnects with internal discontinuities. *IEEE Transactions on Electromagnetic Compatibility*, 60(1):77–85, Feb 2018.
- [132] Yao Yue, Lihong Feng, and Peter Benner. Reduced-order modelling of parametric systems via interpolation of heterogeneous surrogates. *Advanced Modeling and Simulation in Engineering Sciences*, 6, 2019.
- [133] S. Boyd, Balakrishnan V., and P. Kabamba. Bisection method for computing the H_∞ norm of a transfer matrix and related problems. *Math. Control Signals Systems*, 2:207–219, 1989.
- [134] Zheng Zhang, Chi-Un Lei, and Ngai Wong. Ghm: A generalized hamiltonian method for passivity test of impedance/admittance descriptor systems. In *2009 IEEE/ACM International Conference on Computer-Aided Design - Digest of Technical Papers*, pages 767–773, 2009.
- [135] Christian Schröder and Tatjana Stykel. Passivation of lti systems. Technical report, DFG Research Center Matheon, TU Berlin, Germany, 2007.

- [136] G.W. Stewart and J. Sun. *Matrix Perturbation Theory*. Computer Science and Scientific Computing. Elsevier Science, 1990.
- [137] T. Stykel. On some norms for descriptor systems. *IEEE Transactions on Automatic Control*, 51(5):842–847, 2006.
- [138] W. Koepf. *Hypergeometric Summation: An Algorithmic Approach to Summation and Special Function Identities*. Advanced Lectures in Mathematics. Vieweg+Teubner Verlag, 1998.
- [139] J. Preibisch, T. Reuschel, K. Scharff, J. Balachandran, B. Sen, and C. Schuster. Exploring efficient variability-aware analysis method for high-speed digital link design using pce. *DesignCon*, 2017.
- [140] W. Wolovich. On determining the zeros of state-space systems. *IEEE Transactions on Automatic Control*, 18(5):542–544, 1973.