Algorithms for identifying guaranteed stable and passive models from noisy data

Thesis submitted in fulfillment of the requirements for the degree of Doctor in de ingenieurswetenschappen (Doctor in Engineering) by

ir. Tom D’haene

Adviser: Prof. Dr. ir Rik Pintelon

January 2008
Algorithms for identifying guaranteed stable and passive models from noisy data

Thesis submitted in fulfillment of the requirements for the degree of Doctor in de ingenieurswetenschappen (Doctor in Engineering) by

ir. Tom D’haene

Chairman:
  Prof. Dr. ir. Gert Desmet
Vice chairman:
  Prof. Dr. ir. Jean Vereecken
Secretary:
  Prof. Dr. ir. Patrick Guillaume
Adviser:
  Prof. Dr. ir. Rik Pintelon
Jury:
  Prof. Dr. ir. Paul Van Dooren
  Prof. Dr. ir. Tom Dhaene
  Dr. Martine Olivi
  Prof. Dr. ir. Gerd Vandersteen
ABSTRACT

The purpose of this thesis is to show an approach to estimate constrained models. The constraints where we look at are the stability and passivity constraints. Here we propose a two step approach. In the first step a (high) degree model without any constraints will be estimated that passes the validation tests (analysis cost function, whiteness weighted residues,...). This step suppresses in an optimal way the noise without introducing systematic errors. In the second step the constraint will be added: The (high) order model will be approximated by a constrained one. For this the weighted difference between the unconstrained model and the constrained model is minimized. The big advantage of the two step procedure is that it provides models with uncertainty and bias error bounds, which is not the case when the constraint is imposed during the noise removal step.

Imposing the stability or passivity constraint in the second step with application of a user defined weighting is the crucial step and is more difficult than it appears. So starting from a model that passed step one, the following steps have to be carried out: The transfer function $G(\Omega)$ is already estimated with an optimal noise removal criterion, however the model is unstable or non passive. To constrain the transfer function, we propose an iterative algorithm that minimizes the following cost function $V$ in a user specified frequency band:

$$V = \frac{1}{F} \sum_{l=1}^{F} w_l^2 |G(\Omega_l) - \hat{G}(\Omega_l, \theta)|^2$$

with $\hat{G}(\Omega, \theta)$ the constrained version of $G(\Omega)$, $\theta$ the transfer function parameters, $w_l$ a user defined weighting function, and $F$ the number of frequencies. The iterative algorithm needs stable or passive initial values $\hat{G}(\Omega_l, \theta_0)$. This can be created by some simple or advanced techniques. The iterative algorithm will create from these initial values a stable or passive model that performs at least as good as the initial values. The basic idea is to decrease the cost function $V$ by leaving some freedom to the gain and the positions of the zeros and the poles of the transfer function.
This method is applicable for single input single output continuous-time and discrete-time systems and multiple input multiple output continuous-time and discrete-time systems.

This technique is illustrated on several measurement examples and compared with other stabilizing and passivity enforcement techniques.
ACKNOWLEDGEMENTS

For being able to accomplish this thesis I am indebted to many people. First of all, I would like to thank my adviser Rik Pintelon, especially for his unbiased view on the research topic and for careful reading of this manuscript and for his helpful comments. I also want to thank Johan Schoukens, Yves Rolain and Gerd Vandersteen, who make this team an inexhaustible source of ideas. Special thanks to my room-mates Johan Paduart and Ludwig De Locht for the pleasant years, the fruitful discussions, and the help they were always willing to provide. To everyone in the ELEC group; thanks for your help and the ever present friendly, enthusiastic atmosphere and the wonderful cake and ice-cream moments; I very much enjoyed my time as a Ph.D. student in Brussels.

I would also like to thank Laurent Baratchart, Martine Olivi and Fabien Seyfert for the interesting discussions and support during my stay at the APICS team.

Finally, I would like to thank Carolien and my family for their support, which they gave abundantly again and again over the years.

Tom D’haene
Aalst, January 2008
Operators and notational conventions

Outline uppercase font denotes a set, for example, \(\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}\) and \(\mathbb{C}\) are, respectively, the natural, the integer, the rational, the real and the complex numbers.

* convolution operator

\(\text{Re}(\ )\) real part of

\(\text{Im}(\ )\) imaginary part of

\(\text{det}(\ )\) determinant of

\(\text{adj}(\ )\) adjunct of

\(O(x)\) an arbitrary function with the property \(\lim_{x \to 0} |O(x)/x| < \infty\)

\(\hat{\theta}\) estimated value of \(\theta\)

\(\bar{x}\) complex conjugate of \(x\)

subscript 0 true value

subscript \(s\) stable part

subscript \(u\) unstable part

subscript \(\text{Re}\) \(A_{\text{Re}} = \begin{bmatrix} \text{Re}(A) & -\text{Im}(A) \\ \text{Im}(A) & \text{Re}(A) \end{bmatrix}\)

subscript \(\text{re}\) \(A_{\text{re}} = \begin{bmatrix} \text{Re}(A) \\ \text{Im}(A) \end{bmatrix}\)

superscript \(T\) matrix transpose

superscript \(-T\) transpose of the inverse matrix

superscript \(H\) Hermitian transpose: complex conjugate transpose of a matrix

superscript \(-H\) Hermitian transpose of the inverse matrix

superscript \(F\) 'fixed' part

superscript \(V\) 'variable' part

\(\angle x\) phase (argument) of the complex number \(x\)

\(X_{[i]}(s)\) \(i\)th entry of the vector function \(X(s)\)

\(A_{[i,j]}(s)\) \(i, j\)th entry of the matrix function \(A(s)\)

\(A_{[:, j]}\) \(j\)th column of \(A\)

\(A_{[i, :]}\) \(i\)th row of \(A\)

\(X[k](s)\) \(k\)th realization of a random process \(X(s)\)

\(\lambda(A)\) eigenvalue of a square matrix \(A\)

\(\sigma(A)\) singular value of an \(n \times m\) matrix \(A\)
$\kappa(A) = (\max_i \sigma_i(A)) / (\min_i \sigma_i(A))$  condition number of an $n \times m$ matrix $A$

$|x| = \sqrt{(\text{Re}(x))^2 + (\text{Im}(x))^2}$  magnitude of a complex number $x$

$\|A\|_1 = \max_{1 \leq j \leq m} \sum_{i=1}^{n} |A_{i,j}|$ 1-norm of an $n \times m$ matrix $A$

$\|A\|_2 = \max_{1 \leq i \leq n} \sigma_i(A)$ 2-norm of an $n \times m$ $(n \geq m)$ matrix $A$

$\|X\|_2 = \sqrt{X^HX}$ 2-norm of the column vector $X$

$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^{m} |A_{i,j}|$ $\infty$-norm of an $n \times m$ matrix $A$

$\|A\|_F = \sqrt{\text{Tr}(A^HA)}$ Frobenius norm of an $n \times m$ matrix $A$

$\text{diag}(A_1, A_2, \ldots, A_K)$ block diagonal matrix with blocks $A_k, k = 1, 2, \ldots, K$

$\text{herm}(A) = (A + A^H)/2$  Hermitian symmetric part of an $n \times m$ matrix $A$

$\text{null}(A)$ null space of the $n \times m$ matrix $A$, linear subspace of $C^m$ defined by $Ax = 0$

$\text{rank}(A)$ rank of the $n \times m$ matrix $A$, maximum number of linear independent rows (columns) of $A$

$\text{span}\{a_1, a_2, \ldots, a_m\}$ the span of the vectors $a_1, a_2, \ldots, a_m$ is the linear subspace obtained by making all possible linear combinations of $a_1, a_2, \ldots, a_m$.

$\text{tr}(A) = \sum_{i=1}^{n} A_{i,i}$ trace of an $n \times n$ matrix $A$

$\text{vec}(A)$ a column vector formed by stacking the columns of the matrix $A$ on top of each other

$\text{Lim}$ limit in distribution

$E\{\}$ mathematical expectation

$\text{Prob}(\cdot)$ probability

$b_X = X - E\{X\}$ bias of the estimate $X$

$\text{Cov}(X, Y) = E\{(X - E\{X\})(Y - E\{Y\})^H\}$ cross-covariance matrix of $X$ and $Y$

$\text{covar}(x, y) = E\{(x - E\{x\})(y - E\{y\})\}$ covariance of $x$ and $y$

$\text{cum}(\cdot)$ cumulant

$\text{var}(x) = E\{|x - E\{x\}|^2\}$ variance of $x$

$C_X = \text{Cov}(X) = \text{Cov}(X, X)$ covariance matrix of $X$

$\hat{C}_X = \frac{1}{M-1} \sum_{m=1}^{M} (X^{[m]} - \hat{X})(X^{[m]} - \hat{X})^H$ sample covariance matrix of $M$ realizations of $X$

$C_{XY} = \text{Cov}(X, Y)$ cross-covariance matrix of $X$ and $Y$
\[ \hat{C}_{XY} = \frac{1}{M-1} \sum_{m=1}^{M} (X[m] - \hat{X})(Y[m] - \hat{Y})^H \] sample cross-covariance matrix of \( M \) realizations of \( X \) and \( Y \)

DFT(\( x(t) \)) discrete Fourier transform of the samples \( x(t), t = 0, 1, \ldots, N-1 \)

\( I_m \) \( m \times m \) identity matrix

\[ \text{MSE}(X) = E \{ (X - X_0)(X - X_0)^H \} \] mean square error of the estimate \( X \)

\[ R_{xx}(\tau) = E \{ x(t)x^H(t-\tau) \} \] auto-correlation of \( x(t) \)

\[ R_{xy}(\tau) = E \{ x(t)y^H(t-\tau) \} \] cross-correlation of \( x(t) \) and \( y(t) \)

\( S_{XX}(j\omega) \) Fourier transform of \( R_{xx}(\tau) \) (autopower spectrum of \( x(t) \))

\( S_{XY}(j\omega) \) Fourier transform of \( R_{xy}(\tau) \) (crosspower spectrum of \( x(t) \) and \( y(t) \))

\[ \hat{X} = \frac{1}{M} \sum_{m=1}^{M} X[m] \] sample mean of \( M \) realizations (experiments) of \( X \)

\[ \mu_x = E \{ x \} \] mean value of \( x \)

\[ \sigma_x^2 = \text{var}(x) \] variance of the \( x \)

\[ \hat{\sigma}_x^2 = \frac{1}{M-1} \sum_{m=1}^{M} |X[m] - \hat{X}|^2 \] sample variance of \( M \) realizations of \( x \)

\[ \sigma_{xy}^2 = \text{covar}(x, y) \] covariance of \( x \) and \( y \)

\[ \hat{\sigma}_{xy}^2 = \frac{1}{M-1} \sum_{m=1}^{M} (X[m] - \hat{X})(Y[m] - \hat{Y}) \] sample covariance of \( M \) realizations of \( x \) and \( y \)
Symbols

\[ A(\Omega, \theta) = \sum_{r=0}^{n_p} a_r p_r(\Omega) \text{ denominator polynomial plant model expanded in the polynomial basis } p_r(\Omega) \]

\[ A(\Omega, \theta) = \sum_{r=0}^{n_a} a_r \Omega^r \text{ denominator polynomial plant model} \]

\[ B(\Omega, \theta) = \sum_{r=0}^{n_q} b_r q_r(\Omega) \text{ numerator plant model expanded in the polynomial basis } q_r(\Omega) \]

\[ B(\Omega, \theta) = \sum_{r=0}^{n_b} b_r \Omega^r \text{ numerator polynomial plant model} \]

\[ e(t) \text{ white noise at time } t \]

\[ E(k) \text{ discrete Fourier transform of the samples } e(tT_s), t = 0, 1, \ldots, N - 1 \]

\[ f \text{ frequency} \]

\[ \Delta f \text{ frequency spacing} \]

\[ F \text{ number of frequency domain data samples} \]

\[ f_s \text{ sampling frequency} \]

\[ G(j\omega) \text{ frequency response function} \]

\[ G_0(j\omega) \text{ true underlying linear system of a non-linear plant} \]

\[ G_B(j\omega) \text{ bias or deterministic non-linear contribution of a non-linear plant} \]

\[ G_R(j\omega) \text{ best linear approximation of a non-linear plant} \]

\[ G_S(j\omega) \text{ zero mean stochastic non-linear contribution of a non-linear plant} \]

\[ G(\Omega, \theta) = B(\Omega, \theta)/A(\Omega, \theta) \text{ parametric plant model} \]

\[ j \]

\[ j^2 = -1 \]

\[ M \text{ number of (repeated) experiments} \]

\[ N \text{ number of time domain data samples} \]

\[ N_G \text{ measurement noise} \]

\[ N_i \text{ non-synchronized periodic disturbance} \]

\[ n_a, n_b, \text{ and } n_c \text{ order of the polynomials } A(\Omega, \theta), B(\Omega, \theta) \text{ and } C(\Omega, \theta) \]

\[ n_\theta \text{ dimension of the parameter vector } \theta \]

\[ n_u(t), n_y(t) \text{ disturbing time domain noise on the input } u(t) \text{ and output } y(t) \text{ signals, respectively} \]

\[ N_u(k), N_y(k) \text{ discrete Fourier transform of the samples } n_u(tT_s) \text{ and } n_y(tT_s), \]

\[ k = 0, 1, \ldots, N - 1, \text{ respectively} \]

\[ s \text{ Laplace transform variable} \]
$s_k$  Laplace transform variable evaluated along the imaginary axis at DFT frequency $k$: $s_k = j\omega_k$

t  continuous or discrete time variable

$T_s$  sampling period

$T_w$  waiting time

$U(e^{j\omega T_s}), Y(e^{j\omega T_s})$  Fourier transform of $u(t T_s)$ and $y(t T_s)$

$U(k), Y(k)$  discrete Fourier transform of the samples $u(t T_s)$ and $y(t T_s)$, $t = 0, 1, \ldots, N-1$

$U_k, Y_k$  Fourier coefficients of the periodic signals $u(t), y(t)$

$U(j\omega), Y(j\omega)$  Fourier transform of $u(t)$ and $y(t)$

$U(s), Y(s)$  one sided Laplace transform of $u(t)$ and $y(t)$

$u(t), y(t)$  input and output time signals

$U(z), Y(z)$  one sided $Z$-transform of $u(t T_s), y(t T_s)$

$V_F(\theta, z)$  cost function based on $F$ measurements

$Z(k) = [Y(k)U(k)]^T$  data vector containing the measured input and output (DFT) spectra at (DFT) frequency $k$

$Z = [Z^T(1)Z^T(2)\ldots Z^T(F)]^T$  data vector containing the measured input and output DFT spectra (dimension $2F$)

$z$  $z$-transform variable

$z_k$  $z$-transform variable evaluated along the unit circle at DFT frequency $k$: $z_k = e^{j\omega T_s} = e^{j2\pi k/N}$

$\delta(\theta, Z)$  column vector of the (weighted) model residuals (dimension $F$)

$\theta$  column vector of the model parameters

$\hat{\theta}(Z)$  estimated model parameters, minimizing argument of the cost function $V_F(\theta, Z)$

$\sigma^2_U(k) = \var(U(k))$  variance of the measured input DFT spectrum

$\sigma^2_Y(k) = \var(Y(k))$  variance of the measured output DFT spectrum

$\sigma^2_{YU}(k) = \covar(Y(k), U(k))$  covariance of the measured output and input DFT spectra

$J(\theta, Z) = \partial \delta(\theta, Z)/\partial \theta$  gradient of residuals $\delta(\theta, Z)$ w.r.t. the parameters $\theta$ (dimension $F \times n_\theta$)

$\omega = 2\pi f$  angular frequency
$\Omega$  \hspace{1cm} generalized transform variable: Laplace domain $\Omega = s$, $z$-domain $\Omega = z$.

$\Omega_k$  \hspace{1cm} generalized transform variable evaluated at DFT frequency $k$: Laplace domain $\Omega_k = j\omega_k$, $z$-domain $\Omega_k = e^{j\omega_k T_s}$, with $\omega_k = 2\pi k / N$
### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>BR</td>
<td>Bounded Real</td>
</tr>
<tr>
<td>BRF</td>
<td>Bounded Real Function</td>
</tr>
<tr>
<td>CT</td>
<td>Continuous-time</td>
</tr>
<tr>
<td>DT</td>
<td>Discrete-time</td>
</tr>
<tr>
<td>EV</td>
<td>Errors-in-Variables</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>FOH</td>
<td>First-Order-Hold</td>
</tr>
<tr>
<td>FRF</td>
<td>Frequency Response Function</td>
</tr>
<tr>
<td>iid</td>
<td>independent identically distributed</td>
</tr>
<tr>
<td>IV</td>
<td>Instrumental Variables</td>
</tr>
<tr>
<td>IWLS</td>
<td>Iterative Weighted linear Least Squares</td>
</tr>
<tr>
<td>KYP</td>
<td>Kalman-Yacubovich-Popov</td>
</tr>
<tr>
<td>LHP</td>
<td>Left Half Plane</td>
</tr>
<tr>
<td>LMI</td>
<td>Linear Matrix Inequality</td>
</tr>
<tr>
<td>LS</td>
<td>Least Squares</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple Input Multiple Output</td>
</tr>
<tr>
<td>ML</td>
<td>Maximum Likelihood</td>
</tr>
<tr>
<td>NLS</td>
<td>Non-linear Least Squares</td>
</tr>
<tr>
<td>NLS-FRF</td>
<td>Non-linear Least Squares based on FRF measurements</td>
</tr>
<tr>
<td>PR</td>
<td>Positive Real</td>
</tr>
<tr>
<td>PRF</td>
<td>Positive Real Function</td>
</tr>
<tr>
<td>RHP</td>
<td>Right Half Plane</td>
</tr>
<tr>
<td>rms</td>
<td>root mean square value</td>
</tr>
<tr>
<td>SBR</td>
<td>Strictly Bounded Real</td>
</tr>
<tr>
<td>SISO</td>
<td>Single Input Single Output</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>SPFV</td>
<td>Separating the Poles in a ‘Fixed’ and ‘Variable’ group</td>
</tr>
<tr>
<td>SPR</td>
<td>Strictly Positive Real</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular Value Decomposition</td>
</tr>
<tr>
<td>VF</td>
<td>Vector Fitting</td>
</tr>
<tr>
<td>WLS</td>
<td>Weighted Least Squares</td>
</tr>
<tr>
<td>ZOH</td>
<td>Zero-Order-Hold</td>
</tr>
</tbody>
</table>
PUBLICATIONS

Articles in scientific journals with an international referee system

Variance analysis of frequency response function measurements using periodic excitations

Authors: D'HAENE Tom, PINTELON RIK, SCHOUKENS JOANNES, VAN GHEEM ELS

An Iterative Method to Stabilize a Transfer Function in the s- and z-Domains

Authors: D'HAENE Tom, PINTELON RIK, VANDERSTEEN Gerd

Passivity Enforcement of a Transfer Function

Authors: D'HAENE Tom, PINTELON RIK
Reference: IEEE Transactions on Instrumentation and Measurement, to be published

Communications at international congresses / symposia integrally published in proceedings

Variance Analysis of Frequency Response Function Measurements Using Periodic Excitations

Authors: D'HAENE Tom, PINTELON RIK, SCHOUKENS JOANNES, VAN GHEEM ELS

An Iterative Method to Stabilise a Transfer function in the s- and z-Domains

Authors: D'HAENE Tom, PINTELON RIK, VANDERSTEEN Gerd

Passivity Enforcement of a Transfer Function
Authors: D'HAENE Tom, PINTELON RIK


Stable Approximations of Unstable Models
Authors: D'HAENE Tom, PINTELON RIK, GUILLAUME PATRICK


Communications at international congresses / symposia not published or only available as an abstract

Variance analysis of frequency response function measurements using periodic excitations
Authors: D'HAENE Tom, PINTELON RIK, SCHOUKENS JOANNES, VAN GHEEM ELS
Reference: 23rd Benelux meeting on Systems and Control, Helvoirt, The Netherlands, March 17-19, 2004

Optimized Based Stable Constrained Fitting
Authors: D'HAENE Tom, PINTELON RIK
Reference: 24th Benelux Meeting on Systems and Control, Houffalize, Belgium, March 22-24, 2005, pp. 68

Passitivity Enforcement of a Transfer Function
Authors: D'HAENE Tom, PINTELON RIK

Stable Approximations of Unstable Models

Authors: D'HAENE Tom, PINTEلون RIK

# TABLE OF CONTENT

Abstract  i  
Acknowledgements  iii  
Operators and notational conventions  v  
Symbols  viii  
Abbreviations  xi  
Publications  xiii  
Table of Content  xvii  

## Chapter 1

**Introduction**  
1.1 System identification  1  
1.2 One step versus two steps  3  
1.3 Assumptions  4  
1.4 Overview  5  

## Chapter 2

**Unconstrained modeling step**  
2.1 Introduction  8  
2.2 Variance analysis of frequency response function measurements using periodic excitations  9  
2.2.1 Introduction  9  
2.2.2 Non-stationary signal disturbances  14  
2.2.2.1 Non-linear distortion  16  
2.2.2.2 Disturbing noise  17  
2.2.2.3 The non-stationary distortions  17  
2.2.2.4 Simulation-based example of application  17  
2.2.2.5 Measurement-based example of application  18  
2.3 Parametric modeling  19  
2.4 Model quality  22  
2.5 FDIDENT  23  
2.6 Multivariable systems  23  
2.7 Appendix  24  

## Chapter 3

**Stable Models**  
3.1 Introduction  27  
3.2 Stability  27
3.3 Stability enforcement
3.3.1 Outline 31
3.3.2 Link with Chapter 1 32
3.3.3 Stable starting values 33
3.3.4 Refinement algorithm 37
3.3.5 Constraint border 40
3.3.6 Line search 41
3.4 Separation of the poles in a ‘fixed’ and ‘variable’ group 42
3.5 Adding the minimax constraint 45
3.6 Increasing the order 47
3.7 Extension to multivariable systems 49
3.7.1 Minimax for MIMO 50
3.7.2 Increasing the order for MIMO 51
3.8 Stability enforcement software 52
3.8.1 RARL2 52
3.8.2 Vector fitting 53
3.9 Disadvantages 54
3.10 Appendix 55
3.10.1 Lemma of Lyapounov 55
3.10.2 Conservation of the phase with initial 4 55

Chapter 4
Passive models 59
4.1 Introduction 59
4.2 Passive system 59
4.2.1 Definitions 59
4.2.2 Continuous-time systems 60
4.2.2.1 Checking the passivity 62
4.2.3 Discrete-time systems 65
4.2.3.1 Checking the passivity 68
4.2.4 Discrete-time passivity constraints 69
4.2.5 Continuous-time to discrete-time transformations 70
4.2.6 Reciprocity 71
4.2.7 Properties of a BRF and PRF 71
4.3 Passivity enforcement 72
4.3.1 Starting values for SISO systems 72
4.3.2 Starting values for MIMO systems 74
4.3.3 Refinement algorithm 75
4.4 Generating passive models with convex solvers 76
4.5 Passivity refinement by inversion of the transfer function 78
4.6 Reparameterizing 79
4.7 Minimax models 85
4.8 Increasing the order 85
4.9 Passivity enforcement software 86
  4.9.1 Vector fitting with passivity enforcement 86
  4.9.2 IdEM toolbox 87
4.10 Appendix 88
  4.10.1 Constraints on the spectral zeros 88
  4.10.2 Eigenvalues of Hamiltonian 88
  4.10.3 Relative degree of one 89
  4.10.4 Illustrations of discrete-time passivity constraints 89
  4.10.5 The order of a passive discrete-time model 92
  4.10.6 Residue of a pool on the unit circle 92
    4.10.6.1 Discretization through bilinear transformation 94
    4.10.6.2 Discretization through impulse invariant transformation 95
    4.10.6.3 Discretization through First-Order-Hold 95
  4.10.7 ZOH destroys the passivity property 96
  4.10.8 Calculation of the orthogonal base 97

Chapter 5
Applications 99

5.1 Introduction 99
5.2 Crankcase 100
  5.2.1 Advantage of separating the parameters 104
  5.2.2 Minimax constraint 104
  5.2.3 Increasing the order 104
  5.2.4 Comparison with freely available toolboxes 108
  5.2.5 Outband energy 109
  5.2.6 Summary of the results 110
5.3 Transformer 111
  5.3.1 Passive model 113
  5.3.2 Minimax constraint 115
  5.3.3 Reparameterization of the numerator 115
  5.3.4 Comparison with freely available toolboxes 117
  5.3.5 Summary of the results 120
5.4 Copper bar 121
  5.4.1 Minimax constraint 123
  5.4.2 Passive model 124
  5.4.3 Reparameterization of the numerator 126
  5.4.4 Comparison with freely available toolboxes 126
  5.4.5 Summary of the results 129
5.5 Two-port Inductor 130
  5.5.1 Increasing the order 131
  5.5.2 Comparison with freely available toolboxes 133
  5.5.3 Passive models 134
  5.5.4 Summary of the results 135
5.6 SMD resistor 135
  5.6.1 Stable models 136
5.6.2 Passive models 138
5.6.3 Comparison with freely available toolboxes 140
5.6.4 Reparameterization of the numerator 142
5.6.5 Summary of the results 142

5.7 High frequency planar inductor 144
5.7.1 Increasing the order 146
5.7.2 Comparison with freely available toolboxes 147
5.7.3 Passive models 148
5.7.4 Summary of the results 149

Chapter 6
Conclusions 151

Chapter 7
Future Work 157

References 1
1.1 System identification

Identification means determining models of physical systems from noisy measured data. Since the modeling of nature is the basis of our understanding of the world, identification methods have applications in virtually every field of sciences, especially technical ones. On the one hand these models give us a better physical understanding of the system behavior, and on the other hand they predict how the system will react on any possible excitation. As an example we can think of the analysis, verification and optimization of prototypes, products and systems and also the usage of the system (product) in other applications as originally designed for.

One of the most often used models for dynamic physical systems is an ordinary linear differential equation with constant coefficients. This model appears in the same mathematical form in very different fields. Thus, the common properties of these equations, and the measurement and estimation procedures of their coefficients can be treated independently of the dimensions of the physical quantities. This is
done in the framework of linear system theory. Linear dynamical systems have two equivalent
descriptions: in the time domain (differential equations), and in the frequency domain (transfer
functions in the $s$-domain). Also the discrete-time descriptions exhibit this duality: difference
equations are equivalent to the $z$-domain transfer functions. This work will only cover the
frequency domain approach. System identification in the frequency domain for linear time-
invariant systems is extensively explained in [49]. In this work we are searching for models
with certain properties. This means that we have to constrain our identification procedure to
those models that have the right properties. Since it is not possible to discuss all possible
constraints, we restricted ourselves to the stability and passivity constraint.

Instability of the fitted models is often a problem. The physical systems studied are usually
stable (otherwise measurements cannot be done, except if the unstable system is stabilized by a
feedback loop) thus, for our purpose, unstable models are usually not acceptable. For a finite
number of data points, the identified model of a stable system can be unstable due to several
causes, like

- Over modeling (too large model orders)
- Improperly chosen value of the delay
- Outliers
- Too small signal-to-noise ratio
- Local minimum of the cost function
- Non-linear behavior

There exist several techniques to obtain a stable approximation. One obvious solution is to
search the appropriate model in a restricted subset of the whole parameter space, but this leads
to a suboptimal but stable approximation. These techniques can be categorized in two main
groups. A group where the stability constraints are enforced during the estimation of the model
from the noisy data, and a group where the stability constraints are added as a posterior step
once a model is obtained. The second constraint we are looking at is the passivity constraint or
the inability to create energy. The passivity property is a less common property than the
stability property, although it is very important for simulators whenever models are
interconnected with each other. Passive models are a subgroup of stable models. This means
that the constraints on these models are stricter than on stable models. So non-passivity of the identified models is due to the same causes as for instability.

![Classification of the models](image)

**FIGURE 1-1** Classification of the models

### 1.2 One step versus two steps

A lot of methods [11, 12, 24, 28, 37, 46, 62, 70] impose the constraint during the estimation of the model parameters from noisy data. The disadvantage of this approach is that contradictory demands are imposed that lead to non-controllable systematic errors. Indeed, to suppress the noise on the measurements, the cost function should be weighted with the noise variance. Imposing a constraint introduces model errors, which should be distributed over the frequency band of interest according to some user defined criteria. Since the corresponding frequency weighting will be different from the noise variance, optimal noise removal and applying a constraint are conflicting demands.

A second approach is the two-step method, which will be used in this work. In the first step a (high) degree model without any constraints will be estimated that passes the validation tests (analysis cost function, whiteness weighted residues, ...). This step suppresses in an optimal way the noise without introducing systematic errors. In the second step the constraint will be added: The (high) order model will be approximated by a constrained one. For this the weighted difference between the unconstrained model and the constrained model is minimized. The big advantage of the two step procedure is that it provides models with uncertainty and
bias error bounds, which is not the case when the constraint is imposed during the noise removal step.

This post processing approach is very popular to create passive approximants from stable models [13, 15, 18, 26, 29, 45]. It is clear that this stable model should be as close as possible to the measurement data, and to verify this one needs uncertainty bounds. This is why the first step of this two step approach is very important.

1.3 Assumptions

The poles violating the stability or passivity conditions may lie within the useful frequency band or outside this band. When the poles lie within the desired frequency band, no tractable solutions can be found without changing the phase characteristics. The only way to solve this problem for a Single Input Single Output (SISO) system with a small approximation error is to add a well chosen delay in the phase [4, 72]. This is allowed for open-loop applications, but it cannot be used in closed-loop. Whenever the poles that are violating the constraints lie outside the useful frequency band, then a good approximation can be found without introducing a delay.

Since the purpose of this work is to create nice constrained models in a user desired frequency band, we assume that the constraints violations appear outside this user defined frequency band. This can easily be understood with a simple example. Let’s consider the unstable second order transfer function $G$ with its stable approximant $\hat{G}$, as shown in Figure 1-2. If one looks at the complex error in Figure 1-2, it is easy to see that $\hat{G}$ is a better approximant of $G$ in the frequency band 1 mHz till 100 mHz, compared to 100 mHz till 10 Hz. This is of course related to the position of the unstable poles which lie around 1 Hz. The further these unstable poles lie outside the user-defined frequency band, the less there influence on the cost function, and the easier it will be, to create a stable approximant.

The techniques of this thesis are still applicable if the violated constraints lie within the useful frequency band. However it is shown in [5] that a model with arbitrary small approximation
error in the useful frequency band can be found at the cost of an exploding amplitude outside this frequency band, which is of course unwanted.

1.4 Overview

This work is divided in four major chapters:

The unconstrained modeling step: This chapter briefly repeats the main steps of identifying a linear time invariant system from noisy measurements. A new extension of the nonparametric modeling step, to detect the presence of non synchronous components, is described in detail.

The stability constraint: This chapter describes an algorithm for generating stable approximations of unstable models with a small approximation error in a user-defined frequency band. This consists of creating stable starting values followed by a refinement step to fine tune the model parameters. This is described for SISO as well as for Multiple Input
Multiple Output (MIMO) systems. Extensions are made to create stable minimax models as well as a technique to improve the obtained stable model by increasing the model order.

The passivity constraint: This chapter describes an algorithm for generating passive approximations of unconstrained models with a small approximation error in a user-defined frequency band of a non-passive model. First it shows a technique how to check if a model is passive or not for continuous-time. Analogy to discrete-time models is posted and preservation of the passivity property under discretization is studied. Some simple techniques are shown to creating passive starting values. Some techniques are shown to improve passive models whenever the refinement algorithm gets stuck in local minima.

Application: In this chapter the techniques described in this thesis are illustrated on several measurement examples. Four SISO examples and two MIMO examples are estimated and approximated by a stable and a passive model. The extensions like increasing the order of the constrained model and creating a minimax approximation are also illustrated. Comparison with other software packages (Vector fitting, IdEM and RARL2) are shown when possible.
The techniques needed to perform the first step of the two-step approach are described in this chapter. From noisy measurements a parametric model will be estimated with its uncertainty bounds to illustrate the quality of the estimation. This estimation step consists of a non-parametric and parametric estimation step. The contribution of this thesis in this chapter consists of an extension to the variance analysis of the non-parametric estimation step in the presence of non-stationary components.
2.1 Introduction

As described in the introduction chapter, the first step of the two step method consists of an unconstrained estimation step from the noisy data. This step consists of finding the “best” model that modelizes all the dynamics of a system that can be extracted from the data. “Best” in a context of noisy input and output measurements, this will be extensively explained in Section 2.3. So the remaining residuals are only due to noise and/or non-linear contributions. However, due to the noisy character of the data, this “best” model does not always have the right properties, one would like or even would expect.

Since stability and passivity of the model are studied in this thesis, a restriction is made to only use linear time-invariant models. Most of the stability and passivity checking algorithms are based on the parameters of the model. Since there is no general parametric modeling framework available to check the stability or passivity of a non-linear measured system, these models will be excluded from this work. However linear models of “slightly” non-linear systems can be handled with the described techniques of this work. Here we suppose that the non-linearities are small with respect to the range of input signals the user wants to use to measure his/her system.

Parametric and nonparametric modeling is already well studied in the literature [40, 49, 63]. The main results will be repeated here briefly. The use of frequency response function (FRF) measurements gives the user a great deal of information about the device under test. Qualitative measurements can be found by the choice of a good excitation and the right processing of the raw data. This step will simplify the task of building parametric models significantly.

Though the focus of this work lies in the second step of the two step method a contribution is also made to this first step. The variance analysis of the non-parametric modeling step is extended to detect the presence of non-stationary components. In this chapter the techniques are explained using single input single output systems. At the end of the chapter a generalization will be made to multivariable systems.
2.2 Variance analysis of frequency response function measurements using periodic excitations

2.2.1 Introduction

FRF’s are an interesting intermediate step in the identification process. The complexity of the modeling problem is visualized before starting the parametric modeling. Although FRF’s are strictly speaking only defined for linear time invariant systems, they are very useful to get insight in the behavior of a real system that is only approximately linear and time-invariant. A large number of methods are developed to detect the presence of non-linear distortions. An extensive overview is given in the literature; see for example [71]. Many of these are very time consuming and require dedicated experiments. Only a few give detailed information about the distortion levels.

The influence of non-linear distortions on FRF measurements using random phase multisines has already been studied in detail in [49, 60, 50]. Random phase multisines are periodic signals consisting of the sum of $N$ harmonically related sine waves with user defined amplitudes $R_k$, and random phases $\phi_k$.

$$s(t) = N^{-1/2} \sum_{k=-N, (k \neq 0)}^{N} R_k e^{j(2\pi t f_{\text{max}}/N + \phi_k)}$$  \hspace{1cm} (2-1)

Where the normalization $N^{-1/2}$ is introduced to keep the power of $s(t)$ constant as $N$ increases. In [49, 60, 50] it has been shown that the FRF of a wide class of non-linear systems, obtained using a random phase multisine, can be written as:

$$G(f_k) = G_0(f_k) + G_B(f_k) + G_S(f_k) + N_G(f_k)$$  \hspace{1cm} (2-2)

with $G_0$ the FRF of the true underlying linear system, $N_G$ the
measurement noise, \( f_k = (k \cdot f_{max})/N \), and \( G_B \) the bias or deterministic non-linear contribution which depends on the odd degree non-linear distortions and the power spectrum of the input only. \( G_S \) is the zero mean stochastic non-linear contribution:

\[
E\{G_S(f_k)\} = 0, \quad E\left\{|G_S(f_k)|^2\right\} = \sigma_S^2(f_k) \tag{2-3}
\]

\[
E\{G_S(f_k)\bar{G}_S(f_l)\} = O(N^{-1}) \quad k \neq l
\]

where the overline denotes the complex conjugate, and where the expected values are taken with respect to the different random phase realizations of the excitation. Due to property (2-3), the stochastic non-linear contributions \( G_S \) act as circular complex noise for \( N \) sufficiently large. Hence over different realizations of the random multisine, \( G_S \) cannot be distinguished from the measurement noise \( N_G \). The sum

\[
G_R(f_k) = G_0(f_k) + G_B(f_k) \tag{2-4}
\]

is the best linear approximation of the non-linear system for the class of Gaussian excitation signals (normally distributed noise, periodic Gaussian noise, and random phase multisines) with a given power spectrum [49, 50].

In [51, 53] a variance analysis procedure has been proposed that allows to detect and quantify the stochastic non-linear distortions \( G_S \) and the disturbing noise \( N_G \). The strategy for this analysis is the following. First for a given random choice of the phases \( \varphi_k \), \( s(t) \) (2-1) is calculated. This signal is applied to the device under test, and \( P \geq 2 \) consecutive periods of the steady state response are measured. In practice the steady state response is reached after a waiting time \( T_w \) such that the transients are well below the disturbing noise level. Next using a new random choice of the \( \varphi_k \)'s, a second experiment with the new \( s(t) \) is performed. This procedure is repeated \( M \geq 6 \) times giving a set of \([M \times P]\) FRF measurements \( G^{[m,p]}(f_k) \) for each frequency \( k = 1, 2, \ldots, F \), \( m = 1, 2, \ldots, M \); \( p = 1, 2, \ldots, P \) (see Figure 2-1). For ease
Variance analysis of frequency response function measurements using periodic excitations

of the notation the frequency argument is dropped in the sequel of this chapter. The following sample means and sample variances are calculated:

\[
G^{[m,p]} = G_R + G_S^{[m]} + N_G^{[m,p]} \tag{2-5}
\]

\[
\hat{G} = \frac{1}{M} \sum_{m=1}^{M} \hat{G}^{[m]} \quad \sigma^2_G = \sum_{m=1}^{M} \frac{\left[\hat{G}^{[m]} - \hat{G}\right]^2}{M(M-1)} \tag{2-6}
\]

\[
\sigma^2_{G,1} = \frac{\sigma^2_G}{M} \quad \sigma^2_{G,2} = \sum_{m=1}^{M} \frac{\sigma^2_G^{[m]}}{M^2}
\]

**FIGURE 2-1** Frequency response function \(G(f)\) measurement: applying \(M\) different random phase realizations of the excitation, and measuring each time \(P\) periods after a waiting time \(T_w\).
Assuming that measurement model (2-5) is valid, the sample mean $\hat{G}$ and the sample variances $\hat{\sigma}_{G,1}^2$ and $\hat{\sigma}_{G,2}^2$ converge for $M \to \infty$ to:

$$\hat{G} \to G_R$$

$$\hat{\sigma}_{G,1}^2 \to \frac{\hat{\sigma}_S^2}{M} + \frac{\hat{\sigma}_n^2}{MP}$$

$$\hat{\sigma}_{G,2}^2 \to \frac{\hat{\sigma}_n^2}{MP}$$

(2-7)

where $\sigma_n^2 = \text{var}(N_G^{[m,p]})$ and $\sigma_S^2 = \text{var}(G_S^{[m]})$ (proof see [49]).

These results can intuitively be understood as follows. If we consider systems for which a periodic input results in a periodic output with the same period, it is clear that $\hat{\sigma}_{G,2}$ only depends on the variations from one period to the other, which are due to the disturbing noise. $G_S$ is periodic over a realization, so it doesn’t influence $\hat{\sigma}_{G,2}$, while it does change over the different realization, so it has a contribution in $\hat{\sigma}_{G,1}$. Using (2-7), the variance of $G_S$ and $N_G$ can easily be estimated as:

$$\hat{\sigma}_n^2 = MP \hat{\sigma}_{G,2}^2$$

$$\hat{\sigma}_S^2 = M(\hat{\sigma}_{G,1}^2 - \hat{\sigma}_{G,2}^2)$$

(2-8)

Now let’s illustrate this variance analysis procedure on a simple simulation of system with a third order non-linearity in the presence of disturbing noise:

$$y(t) = u(t) + 0.1 u(t)^3 + 0.1 n(t)$$

(2-9)

with $P = 50$, $M = 5$, $n(t)$ normally distributed white noise and $u(t)$ an odd random phase multisine with a logarithmic random harmonic grid. On Figure 2-2 one can see the sample mean $\hat{G}$ and the sample variances $\hat{\sigma}_{G,1}^2$ and $\hat{\sigma}_{G,2}^2$. It is clear from Figure 2-2 that a good separation between the noise and non-linearity can be made. The expected value for $\hat{\sigma}_{G,1}^2$ can be calculated from (2-9) and is equal to -64.5 dB.
Variance analysis of frequency response function measurements using periodic excitations

The same technique was applied in [52] for the measurement of the negative power supply gain of an operational amplifier. However, it has been observed in [52] that $\hat{\sigma}_{G,1}$ can be smaller than $\hat{\sigma}_{G,2}$, although the system was measured with the same measurement grid for 25 realizations of 5 periods. The phenomenon is illustrated on Figure 2-3 where $\hat{\sigma}_{G,1}$ becomes smaller than $\hat{\sigma}_{G,2}$ for the lower frequencies. This implies that $\hat{\sigma}_S^2 < 0$ (see (2-8)), which is an undesired property.

The contribution to the FRF framework of this thesis is to explain the contradiction of $\hat{\sigma}_S^2$ (2-8) being negative and to generalize the variance analysis for detecting the influence of non-stationary disturbances such as non-synchronized periodic signals (e.g. 50Hz mains and its harmonics).

**FIGURE 2-2** The top green line: the best linear approximation of the system, the *: $\hat{\sigma}_{G,1}$, the dotted line: $\sigma_{G,2}$.

![Graph showing variance analysis of frequency response function measurements using periodic excitations](image-url)
2.2.2 Non-stationary signal disturbances

Because non-stationary disturbances cause leakage, it is only possible to detect their presence if periodic excitation signals are used [49].

In the presence of non-synchronized periodic disturbances at the output, measurement model (2-5) becomes then:

\[
G_{mp}^{[m,p]} = G_R + G_S^{[m]} + N_G^{[m,p]} + N_I^{[m,p]} \tag{2-10}
\]

where \(N_I\) stands for the contribution of the non-synchronized periodic disturbance. Starting from the variance analysis procedure explained in Section 2.2.1 a third sample variance can be calculated:

**FIGURE 2-3** The top green line: the spectrum of the negative power supply gain, the *: \(\sigma_{G_{1}}\), the dotted line: \(\sigma_{G_{2}}\).
Variance analysis of frequency response function measurements using periodic excitations

\[
\hat{\sigma}_{G,3}^2 = \frac{1}{M} \sum_{m=1}^{M} \sum_{p=1}^{P} \frac{|G[m,p] - \hat{G}|^2}{MP(MP-1)} \quad (2-11)
\]

Assuming that (2-10) is valid, the mean \( \hat{G} \) and the sample variances \( \hat{\sigma}_{G,i}^2 \), \( i = 1, 2, 3 \) converge for \( M \to \infty \) to:

\[
\hat{\sigma}_{G,1}^2 \to \frac{\sigma_n^2}{MP} + \frac{\sigma_I^2}{M^2} + \frac{\sigma_S^2}{M} \\
\hat{\sigma}_{G,2}^2 \to \frac{\sigma_n^2}{MP} + \frac{(P-1) \sigma_I^2}{M^2} \quad (2-12) \\
\hat{\sigma}_{G,3}^2 \to \frac{\sigma_n^2}{MP} + \frac{(MP^2 - 2P + 1) \sigma_I^2}{M^2 (MP-1)} + \frac{(M-1) \sigma_S^2}{M(MP-1)}
\]

where \( \sigma_S^2 \) and \( \sigma_n^2 \) are defined in (2-7) and with

\[
\sigma_I^2 = \text{var}(N_t^{[m,p]}) \quad (2-13)
\]

(proof: see Appendix 2.7)

Note that the sample variance that should be used for parametric estimation of the plant transfer function is \( \hat{\sigma}_{G,1}^2 \), because it is an unbiased and consisted estimate of the variance of the sample mean \( \hat{G} \).

For \( M, P \) sufficiently large the right-hand side of (2-12) can be simplified as:

\[
\hat{\sigma}_{G,1}^2 \to \frac{\sigma_n^2}{MP} + \frac{\sigma_I^2}{M^2} + \frac{\sigma_S^2}{M} \\
\hat{\sigma}_{G,2}^2 \to \frac{\sigma_n^2}{MP} + \frac{\sigma_I^2}{MP} \\
\hat{\sigma}_{G,3}^2 \to \frac{\sigma_n^2}{MP} + \frac{\sigma_I^2}{MP} + \frac{\sigma_S^2}{MP} \quad (2-14)
\]
Using (2-12) the contradiction of $\hat{\sigma}_s^2$ being negative in the presence of a significant level of non-linearities can be explained. $\hat{\sigma}_{G,2}^2$ can become greater than $\hat{\sigma}_{G,1}^2$ only if:

$$\sigma_i^2 > \frac{p^2}{(P-2)} \sigma_s^2$$

(2-15)

By comparing $\hat{\sigma}_{G,2}^2$ with $\hat{\sigma}_{G,3}^2$ in (2-14) one can see that for $\hat{\sigma}_i^2$ dominant, $\hat{\sigma}_{G,2}^2$ will have the same value as $\hat{\sigma}_{G,3}^2$, so the subtraction of $\hat{\sigma}_{G,2}^2$ from $\hat{\sigma}_{G,3}^2$ can’t be used to get quantitative information about $\hat{\sigma}_s^2$. Although using (2-14) this subtraction seems to result in a correct estimate:

$$\hat{\sigma}_s^2 = MP(\hat{\sigma}_{G,3}^2 - \hat{\sigma}_{G,2}^2)$$

(2-16)

Hence the comparison of $\hat{\sigma}_{G,2}^2$ with $\hat{\sigma}_{G,3}^2$ gives the user more qualitative information why $\hat{\sigma}_s^2$ is negative, because when $\hat{\sigma}_{G,1}^2 \approx \hat{\sigma}_{G,2}^2$ then $\hat{\sigma}_{G,3}^2$ will be equal to $\hat{\sigma}_{G,2}^2$, which indicates the presence of non-stationary distortions. So in the regions where $\hat{\sigma}_{G,3}^2 \neq \hat{\sigma}_{G,2}^2$ one can make a correct estimate for $\hat{\sigma}_s^2$, while in the regions were $\hat{\sigma}_{G,3}^2 \approx \hat{\sigma}_{G,2}^2$ no quantitative information about $\hat{\sigma}_s^2$ can be found.

One could also think to use (2-12) for extracting $\hat{\sigma}_n^2$ and $\hat{\sigma}_i^2$. However, since (2-12) gives a qualitative rather than a quantitative description of the influence of the non-stationary distortions, this should be done with great care. (2-12) describes correctly the influence of $\hat{\sigma}_i^2$ to $\hat{\sigma}_{G,1}^2$, $\hat{\sigma}_{G,2}^2$ and $\hat{\sigma}_{G,3}^2$ but does not quantify it exactly.

### 2.2.2.1 Non-linear distortion $\sigma_S^2$

Since the even degree non-linear distortions do not affect the bias term $G_B$, while they increase the variance of $G_S$, the variability of the FRF measurement (2-2) can be reduced by using random phase multisines which excite the odd harmonics only: (2-1) with $R_{2k} = 0$. These so called odd random phase multisines allow detecting the presence and the level of even degree non-linear distortions by looking at the even harmonics in the output spectrum. The disadvantage of this approach is that the frequency resolution of the measurement is halved for
Variance analysis of frequency response function measurements using periodic excitations

a given period length, since only the odd frequencies are present. Hence the period length should be doubled in order to maintain the required frequency resolution. To detect the presence and the level of odd degree non-linear distortions one should leave out some of the odd harmonics. Therefore the odd harmonics are grouped in equal number of consecutive lines, and one line out of each group is eliminated randomly. This can be done for a linear as well as a logarithmic frequency distribution. The resulting excitation is an odd random phase multisine with a random harmonic grid. The random grid is the same for all $M$ the experiments of the FRF measurements.

2.2.2.2 Disturbing noise $\sigma_n^2$

The only possibility to reduce the disturbing noise levels is to carefully design the experimental setup. The noise levels are not influenced by the choice of the excitation signal. Making $PM$ as large as possible minimizes the impact on the final result.

2.2.2.3 The non-stationary distortions $\sigma_i^2$

A careful design of the experimental setup may reduce the influence of non-synchronous periodic components such as the mains frequency and its harmonics. Another way to reduce these distortions is by augmenting the number of periods, because the leakage error decreases as $1/P$ [49].

2.2.2.4 Simulation-based example of application

In this section we illustrate the variance analysis procedure on a system with a third order non-linearity disturbed by noise and two different non-synchronous periodic components:

$$y(t) = u(t) + 0.1u(t)^3 + \sin(2\pi f_1 t) + \sin(2\pi f_2 t) + 0.1n(t)$$

(2-17)

with $P = 50$, $M = 5$, $f_1 = 500\text{Hz}$, $f_2 = 2300\text{Hz}$, $n(t)$ normally distributed white noise and $u(t)$ an odd random phase multisine with a logarithmic random harmonic grid.
This simulation shows that even for large values of the number of periods $P$, $\hat{\sigma}_{G,2}^2$ becomes larger than $\hat{\sigma}_{G,1}^2$ in the neighborhood of the frequencies $f_1$ and $f_2$.

It is clear by observing Figure 2-4 that the presence of the non-stationary distortion is minimal for $\hat{\sigma}_{G,1}^2$. The mathematical explanation for this phenomenon can be found in (2-12). If we compare the two different non-synchronous periodic components in Figure 2-4, it is clear that $f_2$ is responsible for a higher level of distortion than $f_1$. The reason for this is the distance between the non-synchronous periodic component and the excited frequency line.

2.2.2.5 Measurement-based example of application

A measurement of the negative power supply gain of an operational amplifier [52] illustrates the effect of non-stationary distortions. In Figure 2-5 $\hat{\sigma}_{G,1}^2$ is smaller than $\hat{\sigma}_{G,2}^2$ for the low frequency components, while $\hat{\sigma}_{G,3}^2$ has almost the same value as $\hat{\sigma}_{G,2}^2$ due to the presence of

![Figure 2-4](image-url)

**FIGURE 2-4** The top green line: the best linear approximation of the system, the *: $\hat{\sigma}_{G,1}$, the dotted line: $\sigma_{G,2}$, the +: $\sigma_{G,3}$. 

18
the mains and its harmonics. The level of non-linear distortions can only be found for frequencies larger the 1kHz.

![Graph](image.png)

**FIGURE 2-5** The top green line: the spectrum of the negative power supply gain, the *: $\sigma_{G,1}$, the dotted line: $\sigma_{G,2}$, the +: $\sigma_{G,3}$.

### 2.3 Parametric modeling

As explained in the introduction one needs parametric models to check the stability or passivity property of the system under test. Since every measurement experiment only contains a finite number of measurements points in a finite frequency band, there is no information between the measurement points and outside this frequency band. A parametric model describes a system from DC till infinity. However it only approximates the real system well in the frequency band of the experiment. The parameters of the model allow checking easily the stability or passivity of a model with some simple calculations (see Chapter 3 and Chapter 4).

Parametric modeling is a well know technique and is studied tediously in the literature [40, 49]. It consists of finding the model that is as close as possible to the data according to some
user defined criteria. The user has to answer some questions before he/she can begin with the
parametric modeling step: Which stochastic framework do we work in? Which estimator to
use? Which cost function must he/she minimize? We suppose to measure and model the
transfer function $G_0(\Omega)$ of a plant in an errors-in-variable framework, starting from noisy
input and output measurements (see Figure 2-6). An intuitive approach is to extract, first, a

\begin{figure}[h]
\centering
\begin{tikzpicture}
\node (in) at (0,0) {$U(k)$};
\node (out) at (2,0) {$Y(k)$};
\node (sys) at (1,0) {$G_0(\Omega)$};
\node (noise1) at (0,-1) {$N_g(k)$};
\node (noise2) at (2,-1) {$N_p(k)$};
\node (measure) at (0,-2) {$U_0(k)$};
\node (measure1) at (2,-2) {$U_0(k)$};
\node (measure2) at (0,-3) {$M_{U}(k)$};
\node (measure3) at (2,-3) {$M_{Y}(k)$};
\node (measure4) at (0,-4) {$Y(k)$};
\node (measure5) at (2,-4) {$Y(k)$};
\draw [->] (in) -- (sys);
\draw [->] (sys) -- (out);
\draw [->] (measure) -- (noise1);
\draw [->] (measure1) -- (noise2);
\draw [->] (measure2) -- (sys);
\draw [->] (measure3) -- (sys);
\draw [->] (measure4) -- (measure2);
\draw [->] (measure5) -- (measure3);
\end{tikzpicture}
\caption{Frequency domain representation of the measurement process. Note that the
system can be captured in a feedback loop.}
\end{figure}

measured FRF $G(\Omega_k), k = 1, \ldots, F$ of the systems’ transfer function at a set of well chosen
frequencies. Next, these measurements are approximated by a parametric model $G(\Omega_k, \theta)$ that
explains the measurements as good as possible. The quality of the match between
measurements and model is measured by the cost function. The parameters are then tuned to
minimize the cost function so that a best match is obtained. There is no unique choice for the
cost function. If we suppose that we have normally distributed noise, then the least squares
estimation is a natural choice. Starting from the FRF data, (2-18) is an obvious choice for the
cost function.

$$V_F(\theta, Z) = \frac{1}{F} \sum_{k=1}^{F} \frac{|G(\Omega_k) - G(\Omega_k, \theta)|^2}{\sigma^2_G(k)}.$$  \hspace{1cm} (2-18)

The weighted least squares distance between the measurement and the model is minimized.
Measurements with a small uncertainty ($\sigma^2_G(k)$ is small) are more important than those with a
large uncertainty ($\sigma^2_G(k)$ is large). Although this method works amazingly well in many cases,
it suffers from a major drawback. It is not always that easy to get a good measurement of $G_0$
due to the presence of the noise $M_U(k)$ on the input. If the classical correlation methods (H1-method) are used, a bias appears [49]. The measured transfer function converges for an increasing number of averages to:

$$\lim_{M \to \infty} G(\Omega_k) = \frac{S_{YU}(\Omega_k)}{S_{UU}(\Omega_k)} = G_0(\Omega_k) \frac{1}{1 + S_{M_U M_U}(\Omega_k)/S_{UU}(\Omega_k)}.$$  (2-19)

An instrumental variables approach could be used to get rid of this bias, however there exists better estimators who perform better in this errors-in-variable framework.[49, 68]

The problems with minimizing (2-18) lies in the presence of a division $Y(k)/U(k)$ which is a highly non-linear operation. The denominator can become almost zero (the noise cancels the input) at some frequencies and this creates outliers. The errors-in-variables approach avoids a direct division of both measured spectra. Instead the input and output spectra are considered as unknown parameters, connected by the parametric transfer function model:

$$\begin{align*}
Y(k) &= Y_0(k) + N_Y(k) \\
U(k) &= U_0(k) + N_U(k)
\end{align*}$$  (2-20)

with $Y_0(k) = G(\Omega_k, \theta)U_0(k)$ and where $N_Y(k)$ and $N_U(k)$ include the generator noise, the process noise, and the measurement noise. Since the exact Fourier coefficients $Y_0(k)$ and $U_0(k)$ are unknown, they are replaced by the parameters $Y_p(k)$ and $U_p(k)$ that are estimated by minimizing the distance between the measurements and the parameters ($|U(k) - U_p(k)|$, $|Y(k) - Y_p(k)|$), leading to a new constrained optimization problem. It is possible to eliminate $U_p(k)$, $Y_p(k)$ explicitly from the problem resulting in the following cost function:

$$V_F(\theta, Z) = \frac{1}{\bar{F}} \sum_{k=1}^{\bar{F}} \frac{|Y(k) - G(\Omega_k, \theta)U(k)|^2}{\sigma_Y^2(k) + \sigma_U^2(k)} G(\Omega_k, \theta)^2 - 2\text{Re}(\sigma_Y^2(k)G(\Omega_k, \theta))$$  (2-21)

In the previous case of repeated periodic measurements are $Y(k)$ and $U(k)$ replaced by their sample means. Also the different (co-)variances can be replaced by there sample (co-)variances. By replacing in the cost function $G(\Omega_k, \theta) = B(\Omega_k, \theta)/A(\Omega_k, \theta)$ and multiplying the numerator and denominator with $|A(\Omega_k, \theta)|^2$ a complete symmetric formulation is found. The input and output have exactly the same role in the problem:
Unconstrained modeling step

\[ V_F(\theta, Z) = \frac{1}{F} \sum_{k=1}^{F} \frac{|A(\Omega_k, \theta)Y(k) - B(\Omega_k, \theta)U(k)|^2}{\sigma^2(k)|A(\Omega_k, \theta)|^2 + \sigma^2(k)|B(\Omega_k, \theta)|^2 - 2 \text{Re}(\sigma^2(k)A(\Omega_k, \theta)B(\Omega_k, \theta))} \tag{2-22} \]

The minimum of (2-22) results in our “best” unconstrained parametric model that explains the measurement data. Every choice for the order of \( B(\Omega_k, \theta) \) and \( A(\Omega_k, \theta) \) results in a different parametric model. To choose the order of the model, one should know how good the model is. To answer this question the user has to have an idea about the systematic error (or bias) and uncertainty level. The difference between the model and the FRF should lie within the uncertainty bounds to be sure that there are no model errors or bias. If this is not the case, then this indicates that the order is too low or point at the presence of non-linear distortions. To make a distinction between these last two, a correlation test on the residuals should be executed. When the proper order for \( B(\Omega_k, \theta) \) and \( A(\Omega_k, \theta) \) are selected this model will have no unmodeled dynamics. There can still be some bias available due to non-linear behavior of the system. One can use an information criterion like AIC or MDL [49] to select the proper order of the model, these methods punish models that are too complex and don’t further decrease the systematic error significantly. In [58] an automatic order estimation procedure is explained. This should start with an overestimate of the order of the model, followed by eliminating poles and zeros that have almost no influence on the values of the cost function.

The above formulas are parameterized in the coefficients of the numerator and denominator. Other parameterization could also be used like zero-pool-gain or pool-residue-gain or state space representations. However most of these other parameterization have major drawbacks and will not be used here [49].

2.4 Model quality

These information criteria detect the presence of over-modeling. Under-modeling can also appear, so that the remaining unmodeled dynamics, due to too low order or to non-linear distortions. This gives us systematic errors. These errors can be detected based on the residuals.
that are given as the difference between the measured output and the predicted output. These residuals should be white if there are no systematic errors present.

The uncertainty bounds on the transfer function are calculated from the covariance matrix of the parameters (generated by the estimation algorithm). The bounds can be used together with the level of the residuals as a quality indicator of the constraint models. As explained in the introduction and following chapters the second step minimize the difference between an unconstrained model and its constrained approximant. If this difference is below one or all these bounds, then this means that no systematic error is introduced by forcing the wanted properties on the unconstrained model.

As a guideline one may say that the constrained model in the approximation step is acceptable if the introduced model errors are of the same order of magnitude as the residuals between the unconstrained model and the data/measurements. In the case one starts from simulation data, the residuals of the unconstrained model with the data can be of the order of magnitude of the calculation uncertainty. Here it will be almost impossible to achieve this level of model errors in the approximation step, so the maximal approximation error should be set by the user.

2.5 FDIDENT

All these parametric modeling techniques are available in the frequency domain identification toolbox [36]. In further chapters this toolbox will be used in a couple of SISO examples to estimate the unconstrained model. For the MIMO examples proper ML estimators will be used to estimate the unconstrained model.

2.6 Multivariable systems

The above described techniques are also applicable on multivariable systems. The techniques for the FRF measurements remain the same. Some adaptations and some subtleties have to take into account to perform the parametric modeling step [49]. $G(\Omega_k, \theta)$ will now be a transfer function matrix $G_{i,j}(\Omega_k, \theta)$ with $i$ and $j$ associated index of the different inputs and outputs.
The relationship between these different transfer functions are created by a common denominator. \( B(\Omega_k, \theta) \) becomes a polynomial matrix \( B_{[i,j]}(\Omega_k, \theta) \) and \( A(\Omega_k, \theta) \) is the common denominator polynomial [49]. Another representation that can be used is the state space models. They are very popular to represent a multivariable model, since no adoptions have to be made to represent a SISO system. [64] shows that both methods are very similar and perform comparable. Since we want to be free to adapt the order of our constrained model (see further) and allow improper models, the common denominator representation is an obvious choice. In further sections some techniques will also be explained for the state space representation.

One could say that a drawback of the common-denominator representation lies in the fact that the complexity of the model increases with the square of the number of ports. This is due to the fact that the rank of the residue matrices is not constrained to one. However, since most real-life systems behave to some extent nonlinearly, the identified best linear approximation will not have rank one residue matrices. This has experimentally been verified on the operational modal analysis of an arc bridge [54]. Hence, constraining the rank of the residue matrices to one in the noise removal step can introduce (significant) model errors [10]. Therefore, using a common-denominator representation is rather an advantage than a drawback. Of course if one likes to restrict the rank of the residue matrices to one, a state-space representation can be used. Indeed, the techniques described in this thesis are also applicable to state-space models.

<table>
<thead>
<tr>
<th>State-Space</th>
<th>Common-denominator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same representation SISO as MIMO</td>
<td>Full control of the order of the numerator and denominator</td>
</tr>
<tr>
<td>Less parameters</td>
<td>Allows rank of the residue matrices larger than one</td>
</tr>
</tbody>
</table>

**TABLE 2-7.** Advantages of a state-space representation and a common-denominator representation

### 2.7 Appendix

From (2-10) and (2-6) it follows that:
This leads to the following formula in the general case:

\[
\hat{G}^{[m]} - \hat{G} = G_R + G_S^{[m]} + \frac{1}{P} \sum_{k=1}^{P} (N_G^{[m,k]} + N_t^{[m,k]})
\]

\[-G_R - \frac{1}{M} \sum_{l=1}^{M} \left( G_s^{[l]} + \frac{1}{P} \sum_{o=1}^{P} (N_G^{[l,o]} + N_t^{[l,o]}) \right) \tag{2-23} \]

All the different terms of (2-24) are uncorrelated, hence the expected value of the cross products are zero. The expected value of the square of the absolute value of (2-24) leads to:

\[
\delta_e = e^{[m]} - \frac{1}{M} \sum_{l=1}^{M} e^{[l]} = \frac{(M-1)}{M} e^{[m]} - \frac{1}{M} \sum_{l=1, l \neq m}^{M} e^{[l]} \tag{2-24} \]

\[
E\left[ |\delta_e|^2 \right] = \frac{(M-1)^2}{M^2} \sigma_e^2 + \frac{(M-1)}{M^2} \sigma_e^2 = \frac{M-1}{M} \sigma_e^2 \tag{2-25} \]

\[
N_t^{[m]} = \frac{1}{P} \sum_{n=1}^{P} N_t^{[n,p]} \tag{2-26} \]

\([N_t^{[m]}] \) contains leakage over \( P \) consecutive periods. This results in:

\[
var(N_t^{[m]}) = \frac{var(N_t^{[m,p]})}{P^2} = \frac{\sigma_t^2}{P^2} \tag{2-27} \]

since the leakage influence decreases in \( 1/P \) and stochastic mean also decreases in \( 1/P \) [59]. The formulas from above can now be used to calculate the different contributions of \( \sigma_t^2 \) to the different expressions of (2-12):
Unconstrained modeling step

\[ \delta = N_t^{[m, p]} - \frac{1}{P} \sum_{n=1}^{P} N_t^{[n, p]} \]  
(2-28)

\[ E\{ |\delta|^2 \} = \sigma_t^2 + \frac{\sigma_t^2}{P^2} - \frac{2}{P} \sigma_t^2 = \frac{(P-1)^2}{P^2} \sigma_t^2 \]  
(2-29)

\[ \delta = N_t^{[m]} - \frac{1}{M} \sum_{n=1}^{M} N_t^{[n]} \]  
(2-30)

\[ E\{ |\delta|^2 \} = \sigma_t^2 + \frac{\sigma_t^2}{M} \frac{\sigma_t^2}{P^2} - \frac{2}{MP} \sigma_t^2 = \frac{M-1}{MP^2} \sigma_t^2 \]  
(2-31)

\[ \delta = N_t^{[m, p]} - \frac{1}{M} \sum_{n=1}^{M} N_t^{[n]} \]  
(2-32)

\[ E\{ |\delta|^2 \} = \sigma_t^2 + \frac{1}{M^2} \frac{\sigma_t^2}{P^2} - \frac{2}{MP} \sigma_t^2 = \frac{MP^2 - 2P + 1}{MP^2} \sigma_t^2 \]  
(2-33)

After the appropriate scaling of (2-29), (2-31) and (2-33) the final contributions to (2-12) is found:

\[ \sigma_{G, 1}^2 : \frac{M-1}{MP^2} \frac{\sigma_t^2}{M(M-1)} = \frac{\sigma_t^2}{MP^2} \]  
(2-34)

\[ \sigma_{G, 2}^2 : \frac{(P-1)^2}{P^2} \frac{\sigma_t^2}{MP^2} = \frac{(P-1)^2}{MP^2} \sigma_t^2 \]  
(2-35)

\[ \sigma_{G, 3}^2 : \frac{MP^2 - 2P + 1}{MP^2} \frac{\sigma_t^2}{MP(MP-1)} = \frac{MP^2 - 2P + 1}{MP^2(MP-1)} \sigma_t^2 \]
3.1 Introduction

This chapter addresses the following questions: ‘What is a stable model/system?’ ‘How can I detect if a model is stable or not?’ and ‘How can I stabilize a model?’. In this chapter the second step of the two-steps approach is explained in the case of the stability constraint. This step approximates an unstable model by a stable one, and consists itself of two steps. First creating stable initial values, and secondly refining the model within an iterative algorithm where the gain, zeros and poles are allowed to vary. Extensions to this technique are proposed to create minimax stable models and higher orders stable models.

3.2 Stability

A physical system is called stable if its free solutions cannot increase indefinitely with time [6]. In the first step of the two step problem, transfer function matrices are estimated. This model can be transformed to other representations like state-space equations, pole-
Stable Models

zero-gain or pole-residue form. The stability properties of a linear (time invariant) system described by a transfer function matrix \( G(\Omega) \) (\( \Omega = z \) or \( s \)) may usually be inferred from the pole positions of the entries of the transfer function matrix. In [3] two definitions of stable models are given:

**Definition 3.1**: If each entry of a rational transfer-function matrix \( G(\Omega) \) with \( G(\infty) < \infty \) has all its poles in \( \text{Re}[s] < 0 \) for continuous-time systems and \( |z| < 1 \) for discrete-time systems, the system with transfer-function matrix \( G(\Omega) \) is such that bounded inputs will produce bounded outputs, and outputs associated with nonzero initial conditions will decay to zero.

**Definition 3.2**: If each entry of a rational transfer-function matrix \( G(\Omega) \) with \( G(\infty) < \infty \) has all its poles in \( \text{Re}[s] \leq 0 \) for continuous-time systems, and \( |z| \leq 1 \) for discrete-time systems, with any pole on \( \text{Re}[s] = 0 \) or \( |z| = 1 \) being simple, the system with transfer-function matrix \( G(\Omega) \) is such that bounded inputs will produce bounded outputs, outputs associated with nonzero initial conditions will not be unstable, but may not decay to zero.

In the further text we will call systems that satisfy Definition 3.1 strictly stable and systems that fulfill Definition 3.2 stable systems. An example of a stable system that is not strictly stable is an ideal oscillator. For example, an oscillator \( G(s) = \frac{s}{s^2 + 1} \) with an angular frequency of 1 rad/s.

A transfer function for generic finite dimensional continuous-time and discrete-time systems can be converted in respectively the following state-space models,

\[
\frac{dx(t)}{dt} = Ax(t) + Bu(t) \quad (3-1)
\]

\[
y(t) = Cx(t) + Du(t)
\]

and
Stability

\begin{align*}
    x(t+1) &= Ax(t) + Bu(t) \\
    y(t) &= Cx(t) + Du(t)
\end{align*}

(3-2)

with \( G(\Omega) = D + C(\Omega I_n - A)^{-1}B \) as relation between the transfer function matrix and the state-space matrices. Here, for a system of order \( n \) with \( n_u \) input and \( n_y \) output, \( A \in \mathbb{R}^{n \times n} \) is the state matrix, \( B \in \mathbb{R}^{n \times n_u} \) is the input matrix, \( C \in \mathbb{R}^{n_y \times n} \) is the output matrix, and \( D \in \mathbb{R}^{n_y \times n_u} \) is the direct term, giving the instantaneous input-output interaction. Starting with the transfer function matrix of the general state-space model, we first observe that the poles of \( G(\Omega) \) are either the same as (under minimality assumptions) or some subset of the poles of \( G_p(\Omega) = (\Omega I - A)^{-1} \). Applying Cramer's rule for the matrix inverse, the denominator polynomial for \( (\Omega I - A)^{-1} \) is given by the determinant \( L(\Omega) = \det(\Omega I - A) \), where \( \det(Q) \) denotes the determinant of the square matrix \( Q \). In linear algebra, the polynomial \( L(\Omega) \) is called the characteristic polynomial of the matrix \( A \), and the roots of the characteristic polynomial are called the eigenvalues of \( A \). So if \( \lambda \) is a pole of \( G(s) \), then it is also an eigenvalue of \( A \). Thus we can always infer the pole position properties of \( G(s) \), traditionally associated with stability, from the known stability properties of the state-space equations. Under the assumptions of minimality of the state space representations, these two representations are equivalent.

There are several methods to test whether a system is stable are not. One can test the stability of the (common) denominator of a transfer function matrix using a Routh test or Hurwitz test [3, 27]. These tests check the stability of a prescribed polynomial, without the necessity of factoring the polynomial. A finite number of arithmetic operations and checking of sign of certain quantities are the operations required in these stability tests. If the system is described by its state-space equations, then the stability will be checked via the eigenvalues of \( A \). This corresponds to applying the Routh or other test to \( \det(\Omega I - A) \). These Routh-Hurwitz tests are however very sensitive, so they are not good candidates. An alternative test is the Lemma of Lyapunov (shown here for continuous-time systems; similar expressions exist for discrete-time systems).
**Lemma of Lyapunov:** Let \( A \) be a \( n \times n \) matrix. Then \( \text{Re}(\lambda_i[A]) \leq 0 \) if and only if for any \( C \) with \([A, C]\) completely observable ((3-3) of full rank),

\[
\begin{bmatrix}
C \\
CA \\
CA^2 \\
\vdots \\
CA^{n-1}
\end{bmatrix}
\]

(3-3)

there exists a unique positive definite symmetric \( P \) such that [3]:

\[
PA + A^TP = -C^TC
\]

(3-4)

Let \( A \) be a \( n \times n \) matrix, if there exists a positive definite matrix \( P \) such that \( PA + A^TP = -C^TC \) with \( C = 0 \) permitted, then \( \text{Re}(\lambda_i[A]) \leq 0 \) and the Jordan form of \( A \) has no blocks of size greater than \( 1 \times 1 \) with purely imaginary diagonals elements (only purely imaginary eigenvalues).

Observe that in applying the test, there is a great freedom in the possible choice of \( C \). A simple choice that can always be made is \( C^TC = I_n \), with \( I_n \) the \( n \times n \) identity matrix (see Appendix 3.10.1).

In the algorithms described in the sequel of this chapter, we are not only interested in the stability of the model, but also in the positions of the poles. For many applications poles close to or on the stability border are not allowed, so this will lead to stable models with user defined constraints on the poles (see further). However, since it is not possible to check whether these constraints are violated with a simple Routh-Hurwitz test or the Lyapunov-criterion, the constraint test can only be performed via the values of the poles. The only methods that remain are calculating the eigenvalues of \( A \), or the roots of the (common) denominator of the transfer function (matrix) and will be used in the further sections. Note that calculating the eigenvalues of \( A \) is still possible in case of an improper system. Indeed, it is possible to create a numerator
order that is larger than the denominator order by adding one or more matrices to the output equation. For example, if the order of the numerator is one higher than the denominator, (3-2) becomes:

\[
\frac{dx(t)}{dt} = Ax(t) + Bu(t) \\
y(t) = Cx(t) + Du(t) + E\frac{du(t)}{dt}
\] (3-5)

with \(G(s) = D + C(sI_n - A)^{-1}B + sE\) as relation between the transfer function matrix and the state-space matrices.

### 3.3 Stability enforcement

#### 3.3.1 Outline

In a first step an unconstrained (validated) parametric model is derived from the noisy measurements. In a following step one has to check whether the model is stable or not with one of the previously described methods. If the model is stable, then we are lucky and the modeling step stops here. If not, then one has to check the position of the unstable poles. As explained in Chapter 1, Section 1.3, one can only create stable models with small approximation error if the unstable poles lie outside the measured frequency band. The reason for this is that it is not possible to compensate simultaneously the amplitude and phase of an unstable pole, without changing the order of the model. As a consequence, there will always remain an error in the phase and/or amplitude spectrum. If the order may be increased, then it is possible at the cost of an exploding amplitude outside the measured frequency band [5].

In this chapter the focus lies on the second step of the two-step procedure. To learn more about the first step, we refer the reader to the previous chapter. Imposing the stability constraint in the second step with application of a user defined weighting is the crucial step and is more difficult than it appears. So starting from a model that passed step one, the following steps have to be carried out: The transfer function \(G(\Omega)\) is already estimated with an optimal noise removal
criterion, however the model is unstable. To stabilize the transfer function, we propose in Section 3.3.4 an iterative algorithm that minimizes the following cost function \( V \) in a user specified frequency band:

\[
V = \frac{1}{F} \sum_{l=1}^{F} w_l^2 \left| G(\Omega_l) - \hat{G}(\Omega_l, \theta) \right|^2
\]  

(3-6)

subject to \( \theta \in \Theta \), with \( \hat{G}(\Omega, \theta) \) the stabilized version of \( G(\Omega) \), \( \theta \) the transfer function parameters, \( w_l \) a user defined weighting function, \( F \) the number of frequencies and \( \Theta \) a user defined region that constrains the poles. This iterative algorithm needs stable initial values \( \hat{G}(\Omega, \theta_0) \). How to get these initial estimates is explained in Section 3.3.3. Without loss of generality the method is first explained for single input single output (SISO) continuous-time and discrete-time systems, and next generalized to multiple input multiple output (MIMO), continuous-time and discrete-time systems.

The obvious choices for \( w_l \) will be:

- Such that the error in some frequency bands is allowed to be larger.
- Such that the stable solution becomes a stable minimax solution (see section 3.5)

### 3.3.2 Link with Chapter 1

The error-in-variables framework was introduced in the previous chapter to find a model that matches as closely as possible the measurement data. For this the ML estimator was used to estimate the unconstrained model. However, (3-6) is slightly different, because it consists of the minimization of the squared difference between two models. This means that there is no measurement noise present and, therefore, the ML estimator can be reduced to the frequency response function non-linear least squares (FRF-NLS) estimator.
3.3.3 Stable starting values

In a first step stable starting values of the unstable model should be derived. There are a lot of different stabilization techniques described in the literature [12, 37, 41, 42, 48, 70]. Either they create stable models starting from an unstable model or either from the raw data. Here the focus will lie on some very simple time efficient stabilization techniques that were proposed in [42]. Relaxation methods [70], replacing the denominator by its closest stable polynomial [48], augmenting the data by manipulating the observability matrix [12, 41], or convex optimization techniques [37] can also be used to create stable initial values, but they are more complex and are not considered here. The stabilization techniques proposed in [42] are described for discrete-time models. These results will be repeated here and the analogue representation will be presented here for the continuous-time models.

A transfer matrix for generic continuous-time or discrete-time systems (order numerator ≤ order denominator) can be converted in a state-space model using (3-1) and (3-2). Now, consider the characteristic polynomial of the matrix $A$

$$\chi(\lambda) = \det(\lambda I - A)$$  \hspace{1cm} (3-7)

and suppose that it has some roots $\lambda_1, \ldots, \lambda_u$ with real part larger than zero, and the rest $\lambda_{u+1}, \ldots, \lambda_n$ with real part smaller than zero (in the case of continuous-time systems) or $\lambda_1, \ldots, \lambda_u$ with absolute value larger than one, and the rest $\lambda_{u+1}, \ldots, \lambda_n$ with absolute value smaller than one (in the case of discrete-time systems); none of these groups being empty. It can be shown [35] that there exists a non-singular matrix $T$, with first $u$ columns $T_u$ and last $s$ ($s = n - u$) columns $T_s$, which separates $A$ in $\hat{A}_u$ and $\hat{A}_s$, respectively the unstable and stable part of $A$.

$$\hat{A} = \begin{bmatrix} \hat{A}_u & 0 \\ 0 & \hat{A}_s \end{bmatrix} = T^{-1}AT, \quad \hat{B} = \begin{bmatrix} \hat{B}_u \\ \hat{B}_s \end{bmatrix} \quad \text{and} \quad \hat{C} = \begin{bmatrix} \hat{C}_u \\ \hat{C}_s \end{bmatrix}^T$$  \hspace{1cm} (3-8)
Consider now \((A, B, C, D)\) from (3-1) or (3-2) and its similarity transformed set \((\hat{A}, \hat{B}, \hat{C}, D)\), where \(\hat{B} = T^{-1}B\) and \(\hat{C} = CT\). Partition \(\hat{B}\) and \(\hat{C}\) according to \(\hat{A}\) in (3-8). It can easily be shown that \(G(\Omega)\) can be decomposed as (see also Figure 3-1):

\[
G(\Omega) = G_s(\Omega) + G_u(\Omega) + D
\]

with:

\[
G_s(\Omega) = \hat{C}_s(\Omega I_s - \hat{A}_s)^{-1}\hat{B}_s
\]

\[
G_u(\Omega) = \hat{C}_u(\Omega I_u - \hat{A}_u)^{-1}\hat{B}_u
\]

Since all the poles of \(G_u(\Omega)\) are in the right half plane (RHP) for the \(s\)-domain or outside the unit circle for the \(z\)-domain, we call \(G_u(\Omega)\) the unstable part of \(G(\Omega)\) and similarly \(G_s(\Omega)\) the stable part of \(G(\Omega)\). Of course the zeros of respectively \(G_u(\Omega)\) and \(G_s(\Omega)\) are totally different from the zeros of \(G(\Omega)\). From here on we shall drop the \(\text{“hats”}\), and simply write \(A_u\), instead of \(\hat{A}_u\), and so on.

The additive decomposition of (3-9) immediately suggests several ways to “correct” the instability of \(G(\Omega)\). The most obvious solutions stabilize just \(G_u(\Omega)\) [42] (see also Figure 3-2).

Initial 1: The most simple stable approximation of \(G(\Omega)\) is leaving out the unstable part \(G_u(\Omega)\) in (3-9). This results in a lower order model that “over” simplifies the problem. However, this is the optimal \(L2\)-norm approximant for the whole imaginary axis or the unit circle. The order is now of the dimension of \(A_s\).

Initial 2: A second possibility consists of inverting the norm of the unstable poles with respect to the unit circle for the \(z\)-domain. The equivalent in the \(s\)-domain is flipping all the unstable poles around the imaginary axis. The positions of the zeros remain the same. Similar one could
replace the matrix $A_u$ by $A_u^{-1}$ for the $z$-domain or by $-A_u$ for the $s$-domain. However, this changes the zeros of $G_u(\Omega)$.

Initial 3: A third possibility consists of replacing $G_u(s)$ by $G_u(-s)$ for the $s$-domain. This changes the positions of the poles and zeros of $G_u(s)$ instead of only the poles of $G_u(s)$ as in Initial 2. Hence (3-9) results in $\hat{G} = G_u(-s) + G_s(s) + D$. In the $z$-domain we replace $G_u(z)$ by $G_u(z^{-1})$, giving $\hat{G} = G_u(z^{-1}) + G_s(z) + D$.

Initial 4: A last stable approximation exists of multiplying $G(\Omega)$ with $\det((\Omega I_u - A_u)(\Omega^* I_u - A_u))$. 

**FIGURE 3-1** Separation of the unstable model $G(\Omega)$ in its stable part $G_s(\Omega)$ and unstable part $G_u(\Omega)$. 

FIGURE 3-2  Simple stabilization methods for discrete-time and continuous-time models.
Where this operation preserves the phase spectrum, while changing the amplitude spectrum (see Appendix 3.10.2). The order of the numerator will be increased with the number of unstable poles of \( G(\Omega) \). In the \( s\)-domain the order of the denominator will drop at the same time with the same number, while the order of the denominator remains the same in the \( z\)-domain. The poles of the new model are defined as \( \det(\Omega I_s - A_s) \) (see Appendix 3.10.2). For the \( z\)-domain all the new extra poles lie in the origin. However, since simulation software does not like improper models, one should be cautious using these initial values. These initial values can be implemented in a good numerical way by multiplying \( G(\Omega) \) with an appropriate all-pass filter that cancels the unstable poles [69].

**3.3.4 Refinement algorithm**

In this section we propose an iterative method to find a stable model or to improve a stable model. For this stable initial values are needed. These can be found using stabilization techniques described in Section 3.3.3 and in the literature [12, 37, 41, 42, 48, 70]. Every different stabilization method usually results in different initial values. This will mostly lead to (different) optimized stable models in a non-convex set (local minima!). The basic idea is to decrease the cost function (3-6) by leaving some freedom to the gain and the positions of the zeros and the stable poles. The reason why this algorithm mostly creates better models than the methods described in Section 3.3.3, lies in the fact that the created models are optimized in a user-defined frequency band. Indeed, this is not the case for the techniques described in section 3.3.3 for example initial 1 gives us the optimal \( L2\)-norm stable approximant over the whole frequency band (unit circle or imaginary axis). Since there is no single stable starting value method that outperforms for any unstable system, as much methods as possible should be tried as initial values for the iterative method.
The iterative method uses the method of Levenberg-Marquardt [22]. Starting from the initial values \( \theta_k \) \((k = 0)\), one can calculate the Jacobian \( J \) (dimensions: measurements times parameters), to calculate the step \( \Delta \theta \):

\[
(J^T re J re + \lambda I) \Delta \theta = -J^T re e re
\]

with the \( e \) weighted difference between the unstable model and the stabilized model:

\[
e(l) = w_l(G(\Omega_l) - \hat{G}(\Omega_l, \theta)), e = \begin{bmatrix} e(1) \\ e(2) \\ \vdots \\ e(N) \end{bmatrix}
\]

\( J = \frac{\partial e}{\partial \theta} \) the Jacobian matrix, where \( re \) indicates that the real and imaginary parts are put on top of each other, e.g.

\[
e_{re} = \begin{bmatrix} \text{Re}(e) \\ \text{Im}(e) \end{bmatrix}
\]

\( J^T \) is the transpose of \( J \) and \( \lambda \) the Marquardt parameter (the value of \( \lambda \) is increased with 10 and decreased with 1.2 (see figure 3-6). These numbers were derived from a high number of experiments [22]. As initial \( \lambda \) one can choose the maximum singular value of \( J_{re} \). This results in new values \( \theta_{k+1} \) that lower the cost function (3-6).

\[
\theta_{k+1} = \theta_k + \Delta \theta
\]

These new values \( \theta_{k+1} \) should correspond to a stable model \( \hat{G}(\Omega, \theta_{k+1}) \). Knowing that \( \theta_k \) corresponds to a stable model \( \hat{G}(\Omega, \theta_k) \), there are two possible results. Either \( \hat{G}(\Omega, \theta_{k+1}) \) is stable, and one can start with the next iteration, or \( \hat{G}(\Omega, \theta_{k+1}) \) is unstable, and \( \Delta \theta \) is reduced with, for example, a factor 2. If \( \hat{G}(\Omega, \theta_{k+1}) \) is still unstable after reducing \( \Delta \theta \), then \( \Delta \theta \) is further decreased till \( \hat{G}(\Omega, \theta_{k+1}) \) becomes stable. Proceeding in this way the obtained refined stable model performs at least as good as the initial stable values. Note that in...
Stability enforcement

FIGURE 3-3  Flow-chart of the Levenberg-Marquardt algorithm with stability constraints, where $V$ is the cost function (3-6)
practice (3-13) is solved using the single value decomposition or a QR factorization of $J_{re}$ [49]. Figure 3-3 illustrates this algorithm by a flow-chart.

3.3.5 Constraint border

The proposed method has the disadvantage that the stabilized poles want to return to their unstable position. Hence this algorithm will stop them at the stability border. For many applications poles on the stability border are not allowed. To avoid this one can add some extra constraints like a maximum value for the real part of every pole or a maximum angle for every pole or zero (see Figure 3-4). This results in a new border which the poles are not allowed to cross. Since the used parametrization are the coefficients of the numerator and denominator, it is hard to embed these extra constraints in the cost function.

**FIGURE 3-4** The dashed lines show 2 examples of constraint borders on the poles of the model, A: maximum real part, B: maximum angle.
3.3.6 Line search

In the iterative algorithm the step size of the parameters is reduced if the new parameter values do not correspond to a stable model. One could think to reduce only the step size of the denominator coefficients and keep the full step for the numerator, since there are no constraints on the numerator. However, this is not a good idea, since this step configuration does not correspond to the original Jacobian, so it can increase the cost function instead of lowering it. A better idea is to perform a line search between the points of no step ($\alpha = 0$) and the full restricted step $\Delta \theta_s$ ($\alpha = 1$) for all the parameters. The full restricted step corresponds to the largest step that still corresponds to a stable model $\hat{G}(\Omega, \theta)$, possibly after some decrease of $n$ steps $\Delta \theta_s = \frac{\Delta \theta}{2^n}$ with $n \geq 0$.

$$e(\alpha) = e(\theta_k + \alpha \Delta \theta_s)$$

(3-17)

Here a quadratic polynomial in $\alpha$ is fitted on three stable steps and the $\hat{\alpha}$ that minimizes the polynomial is chosen [22]. One has to keep in mind that $\alpha$ can not be greater than 1, otherwise the $\hat{G}(\Omega, \theta)$ becomes unstable. This algorithm was tried out on several examples, but there was no significant (in calculation time and obtained local minimum) gain with respect to the full restricted step.
3.4 Separation of the poles in a ‘fixed’ and ‘variable’ group

The major drawback of this algorithm is that the poles that reach the constraint borders will prevent the model from further improvement. The reason for this is that the step will be reduced till zero for every new iteration step. This is illustrated in Figure 3-5 were the constraint of a maximum real part is used for continuous-time systems. In this figure we do not show the zeros of the stable model to keep the illustration clear. The arrows next to the poles give the direction to the next positions of the poles. The techniques of this section will be explained for continuous-time systems, but is completely equivalent for discrete-time systems.

Till now we used $\theta$ to address our parameters. We define 2 sets of parameters $\theta^V$ (variable) and $\theta^F$ (fixed) containing respectively all the poles and zeros that fulfill the constraints and the ones that lie on the constraint borders. Initially $\theta^V$ contains all the parameters and $\theta^F$ is empty. Although we speak about poles and zeros, the numerator and denominator coefficients of the transfer function are used as parameters. However, it are the poles that can violate the constraints. One could think to use directly the poles and zeros as the parameters for the iterative algorithm but this leads to a cost function with a large number of local minima. The modifications to the original algorithm are the following. Once the algorithm has found new

![Figure 3-5](image-url)  
**FIGURE 3-5** Drawback of the refinement algorithm; The arrows next to the poles give the direction to the next positions of the poles.
Separation of the poles in a ‘fixed’ and ‘variable’ group

parameters $\theta_{k+1}$ which correspond to a stable model, one has to check whether one or more poles lie on the constraint border. If so, then they are moved from $\theta^V$ to $\theta^F$. Next we continue the iteration with less free variables $\theta^V$. The stable model equals then:

$$\hat{G}(s_f, \theta) = \hat{G}^V(s_f, \theta^V)\hat{G}^F(s_f, \theta^F)$$  (3-18)

This extra feature allows the user to improve his/her stable models even when one or more poles reach the constraint borders, and will result in a local stable minimum, or in the worst case a model with all the poles on the constraint border.

An extension of this technique can be added to further improve the stable model. Whenever a pole becomes a member of $\theta^F$, then we still have the possibility to move the pole along the constraint border. The parameter group $\theta^F$ should be decomposed in a new group of free parameters along the constraint border $\theta^{FV}$ and a fixed group $\theta^{FF}$ perpendicular to the constraint border. One could argue to allow the $\theta^{FF}$ parameters to step away from the constraint border. However, since these parameters already found their way to the border it is very unlikely that they can escape from the border.

If the constraint border is of type A (see Figure 3-4), then the $\theta^F$ will be decomposed as follows in the case of a complex pole pair:

$$(s - p)(s - \bar{p}) = s^2 - 2\text{Re}[p]s + |p|^2$$  (3-19)

with $\bar{p}$ the complex conjugate of $p$. The coefficient of $s$ lies on the constraint border so it cannot be changed and will become a member of $\theta^{FV}$, while $|p|^2$ becomes a member of $\theta^{FF}$. This allows movements along the constraint border which are parallel to the imaginary axis. The reader should remark that although $|p|^2$ becomes a member of $\theta^{FV}$, it still must remain larger or equal than $(\text{Re}[p])^2$. If the poles are real, then they become a member of $\theta^{FF}$, whenever they touch the constraint border. The number of $\theta^{FV}$ does not change for this type of poles.
In the case of a constraint border of type B (see Figure 3-4), the $\theta^F$ have to be decomposed as follows in the case of a complex pole pair:

$$p = -re^{-j\phi}$$  \hspace{1cm} (3-20)

$$(s-p)(s-\bar{p}) = s^2 + 2r\cos(\phi)s + r^2$$  \hspace{1cm} (3-21)

For this type of constraint, one may only change the value of $r$ in the coefficients of $s$. So $r$ is put in $\theta^{FV}$ and $\phi$ in $\theta^{FF}$. From Figure 3-4 it is clear that the real poles have to be smaller then (or equal to) 0.

In the general case of an arbitrary user-defined constraint area (see Figure 3-6, gray area), with border described by the function

$$g(\text{Re}(s_1), \text{Im}(s_1)) = 0$$  \hspace{1cm} (3-22)
and where $\tilde{s}_1$ is a point on the border, the constraint

$$\text{Re}(s_1) - \text{Re}(\tilde{s}_1) \leq 0$$

(3-23)

is verified by solving

$$g(\text{Re}(\tilde{s}_1), \text{Im}(s_1)) = 0$$

(3-24)

for $\text{Re}(\tilde{s}_1)$. For example, for the type A and B constraint of Figure 3-4, (3-23) become respectively $\text{Re}(s_1) + a \leq 0$ with $a > 0$ and $\text{Re}(s_1) - |\text{Im}(s_1)|\tan \phi \leq 0$.

We like to make a small remark concerning the technique of decomposing $\theta^F$ in $\theta^{FV}$ and $\theta^{FF}$. It may happen that this technique performs less than calculating a refinement step without decomposing $\theta^F$, which is of course due to the presence of local minima.

Of course the same idea can also be applied to the zeros, if one wants to create stable minimum phase models.

A disadvantage of this decomposition is that the steps for the parameters of $\theta^F$ are restricted to be perpendicular to the constraint border. This means that steps that push one or more poles of the constraint border are not possible. To resolve this problem, one can take a full Jacobian and check if one or more poles violate the constraints. If so, then the restricted Jacobian has to be used. This increases the calculation time since in every iteration step at least one Jacobian and one step has to be calculated. This corresponds to the following Jacobians respectively:

$$J = \begin{bmatrix} J^V & J^F \end{bmatrix}$$ (full Jacobian) and $J = \begin{bmatrix} J^V & J^{FV} \end{bmatrix}$ (restricted Jacobian).

### 3.5 Adding the minimax constraint

The minimax error of the model can be defined as follows:

$$\min_{\theta} \max_{\Omega \in \Omega} (|e(\Omega, 0)|)$$

(3-25)
with $\Omega$ the user-defined frequency band, where the user wants a good model through his/her data. To find a minimax solution for a stable model based on the existing iterative algorithm, an iterative reweighted least squares method (Lawson’s algorithm) is a justified choice [20, 21, 39, 72], since no change should be made to the constrained iterative algorithm. Of course if other techniques [17, 57] can also be used if one adapt the iterative algorithm.

The Lawson’s weight update procedure consists of weighting the error in the $k + 1$ th iteration step as

$$V_{k+1} = \frac{1}{F} \sum_{l=1}^{F} W_{k+1}(\Omega_l) |e_{k+1}(\Omega_l, \theta_{k+1})|^2$$

(3-26)

where $W_{k+1}$ is a function of the previous error $e_k(\Omega_l, \theta_k)$.

$$\tilde{W}_{k+1}(\Omega_l) = W_k^2(\Omega_l) |e_k(\Omega_l, \theta_k)|^\alpha$$

$$W_{k+1}^2(\Omega_l) = \frac{\tilde{W}_{k+1}^2(\Omega_l)}{\sum_{\Omega_i \in \Omega} \tilde{W}_{k+1}^2(\Omega_i)}$$

(3-27)

and where $\alpha = 1$ for $k$ odd and $\alpha = 2$ for $k$ even (results in faster convergence then selecting $\alpha = 1$ for all $k$ [72]). Initially $W_0^2(\Omega_l) = 1$.

If $e(\Omega_l, \theta_k)$ is linear in $\theta_k$ then it has been shown in [20, 21, 39] that (3-26) converges for $k \to \infty$ to the minimax solution (3-25). For rational transfer functions (3-26) converges to an equiripple solution that is not necessarily the minimizer of (3-25). In [72] this algorithm was used to design digital filters. The described minimax method often does not work without activating the previously described option that blocks the position of the poles violating the border constraints. Without activating this function the algorithm will automatically stop whenever the constraint border is reached. Hence by moving the poles along the constraint border and leaving the other poles and zeros free, the algorithm is allowed to iterate further.
This minimax technique is illustrated on the following IIR-filter problem (see Figure 3-7). This filter has an amplitude equal to one from DC till 1/6 Hz and attenuates 100 dB above 1/3 Hz. This gabarit is approximated by an 8th order model. In Figure 3-7 shows the minimax version of this 8th order model and the unconstrained approximation. Both models are unstable, since no stability constraint was added, this is however a common problem in filter design [72].

3.6 Increasing the order

The model can be improved further by increasing the number of zeros and poles, however, the initial positions of these extra poles and zeros is hard to choose. Here a method is proposed for increasing the order of the stable model.
There are two easy ways to increase the order of the model without violating the stability constraints. By adding or multiplying the stable transfer function $\hat{G}(\Omega, \theta)$ with another stable transfer function $\hat{G}_{\text{extra}}(\Omega, \theta_{\text{extra}})$ that lowers the global error $e_{\text{tot}}(\Omega)$. 

\[
\hat{G}_{\text{tot}}(\Omega, \theta_{\text{tot}}) = \hat{G}(\Omega, \theta) + \hat{G}_{\text{extra}}(\Omega, \theta_{\text{extra}})
\]

\[
\hat{G}_{\text{tot}}(\Omega, \theta_{\text{tot}}) = \hat{G}(\Omega, \theta) \cdot \hat{G}_{\text{extra}}(\Omega, \theta_{\text{extra}})
\]

\[
e_{\text{tot}}(\Omega) = w(\Omega)(G(\Omega) - \hat{G}_{\text{tot}}(\Omega, \theta_{\text{tot}}))
\]

Depending on the example (local minima of the cost function) the sum or multiplication form (3-28) will perform better. Of course it is possible to increase the order of the transfer function in several steps instead of one big step. The order increasing procedure consists of 2 steps. In a first step $\hat{G}_{\text{tot}}(\Omega, \theta_{\text{tot}})$ is put in the iterative algorithm described in Section 3.3.4 but only $\theta_{\text{extra}}$ is allowed to adapt. This step creates optimal starting values for $\hat{G}_{\text{extra}}(\Omega, \theta_{\text{extra}})$. The second step is a final optimization of $\hat{G}_{\text{tot}}(\Omega, \theta_{\text{tot}})$ where all parameters $\theta_{\text{tot}}$ are modified. This will further lower $\|e_{\text{tot}}(\Omega)\|_2$.

The parameters $\theta_{\text{extra}}$ extracted from this first step can correspond to an unstable model $\hat{G}_{\text{extra}}(\Omega, \theta_{\text{extra}})$. The simple stabilization techniques from Section 3.3.3 combined with the refinement step of Section 3.3.4 should be used to stabilize $\hat{G}_{\text{extra}}(\Omega, \theta_{\text{extra}})$. However in this case it is possible that $\|e_{\text{tot}}(\Omega)\|_2$ after this first step is larger than the norm of the original error (3-14). The only case that guarantees $\|e_{\text{tot}}(\Omega)\|_2$ to be equal to or smaller than the norm of (3-14) after the first step, is the case were $\hat{G}_{\text{extra}}(\Omega, \theta_{\text{extra}})$ is identified as a stable model.

One has to keep the following in mind when increasing the model order by letting the number of zeros and poles free. Depending on the numerator/denominator orders of the model the amplitude of the transfer function can go to infinity outside the desired frequency band [5].

When implementing the increasing of the order of $\hat{G}(\Omega, \theta)$ by multiplication, one has to be cautious not to use $(G(\Omega))/(\hat{G}(\Omega, \theta))$ as data for $\hat{G}_{\text{extra}}(\Omega, \theta_{\text{extra}})$. This would be an
obvious choice if one would like to reuse the non-linear least squares algorithm. However in those regions where $\hat{G}(\Omega_p, \theta)$ is small, this results in a bad numerical conditioning.

Guidelines for the user: How should one choose the amount of extra parameters $\theta_{extra}$? For this an inspection of (3-14) is needed. As rule of thumb a model of degree two is needed to catch the dynamics of a peak. Hence one has to count the number of peaks in $|e(\Omega_p)|$ multiply this number by 2, and use this as degree for $\hat{G}_{extra}(\Omega_p \theta_{extra})$. Of course the user has to keep in mind which order for $\hat{G}_{tot}(\Omega_p \theta_{tot})$ is acceptable for the intended application.

### 3.7 Extension to multivariable systems

To extend the proposed techniques to multivariable systems, one has to choose which representation he/she wants to use. Two representations regularly used are obvious choices for this, namely the state space or common-denominator representations. The advantages and disadvantages of both representations have been studied in detail in the literature [35, 64].

Since state space models of SISO and MIMO systems are conceptually the same, the generalization to MIMO systems for this representation is straightforward for this representation. However, the user does not have full control over the numerator order of the corresponding transfer function. This is not a major drawback for the algorithm presented in Section 3.3.4, but causes problems for the pole blocking method of Section 3.4. To implement this method one has to transform the state space matrix $A$ to its modal canonical form [35]. The reason for this is that in such a form, the real part of the poles can be found on the main diagonal of the matrix. So the Jacobian of these elements can be included or excluded depending on their values. Another obvious choice for $A$ would be the diagonal matrix containing the eigenvalues of $A$. However this would result in complex parameters. A last choice would be to decompose the matrices $A, B, C, D$ in a fixed $A^F, B^F, C^F$ and variable $A^V, B^V, C^V, D$ part.
So (3-18) has to replaced by:

$$\hat{G}(s_p, \theta) = \hat{G}^V(s_p, \theta^V) + \hat{G}^F(s_p, \theta^F)$$  (3-31)

In the further iteration steps we only allow $A^V, B^V, C^V, B^F, C^F, D$ to vary and keep $A^F$ fixed. This can be done in the same way as was shown for the starting values step (3-8).

When the common-denominator transfer function representation is used to represent a multivariable system, only small adaptations have to be carried out. The numerator is now a matrix instead of a scalar, so the Jacobian matrix of the corresponding equation error has to be extended to all inputs and outputs. Since there are only constraints on the denominator, nothing has to be changed with respect to the SISO algorithm. The same is true for the algorithm of separating the poles in a ‘fixed’ and ‘variable’ group.

### 3.7.1 Minimax for MIMO

To introduce minimax approximations for MIMO systems, several choices are possible. All the errors can be put in one big vector that is minimized in minimax sense as for one single transfer function. It is also possible to introduce a separate weighting function for every entry of the MIMO system. The advantage of the first approach is that all the errors will be of the same order of magnitude. This is at the same time a disadvantage if there is a large difference in magnitude between the different entries of the transfer matrix. However, this can be solved by adding an appropriate frequency weighting for every entry of the transfer matrix. The advantage of the second approach is the disadvantage of the first approach and visa versa.
### 3.7.2 Increasing the order for MIMO

Suppose that the stable approximant \( \hat{G}(\Omega) = \hat{D} + \hat{C}(\Omega I - \hat{A})^{-1}\hat{B} \) is of order \( n \). To increase the order of the approximant, a transfer matrix \( \hat{G}_{\text{extra}} \) of the same dimensions with a user defined order \( m \) must be added to \( \hat{G} \).

\[
\hat{G}_{\text{extra}}(\Omega) = \hat{D}_{\text{extra}} + \hat{C}_{\text{extra}}(\Omega I - \hat{A}_{\text{extra}})^{-1}\hat{B}_{\text{extra}}
\]  
(3-32)

Defining the new approximation as \( \hat{G}_{\text{tot}} = \hat{G}_{\text{extra}}(\Omega) + \hat{G}(\Omega) \) gives the following state-space matrices for \( \hat{G}_{\text{tot}}(\Omega) = \hat{D}_{\text{tot}} + \hat{C}_{\text{tot}}(\Omega I - \hat{A}_{\text{tot}})^{-1}\hat{B}_{\text{tot}} \):

\[
\hat{A}_{\text{tot}} = \begin{bmatrix}
\hat{A} & 0_{nm} \\
0_{nm} & \hat{A}_{\text{extra}}
\end{bmatrix}
\]  
(3-33)

\[
\hat{B}_{\text{tot}} = \begin{bmatrix}
\hat{B} \\
\hat{B}_{\text{extra}}
\end{bmatrix}^T
\]  
(3-34)

\[
\hat{C}_{\text{tot}} = \begin{bmatrix}
\hat{C} \\
\hat{C}_{\text{extra}}
\end{bmatrix}
\]  
(3-35)

\[
\hat{D}_{\text{tot}} = \hat{D} + \hat{D}_{\text{extra}}
\]  
(3-36)

with \( 0_{nm} \) a zero matrix of dimensions \( n \times m \). A similar notation can be found for the common-denominator representation, but they are less elegant than the state-space representation.

Common-denominator models are increased by multiplying or adding the corresponding entries of the stable model \( \hat{G}(\Omega_p, \theta) \) with another stable model \( \hat{G}_{\text{extra}}(\Omega_p, \theta_{\text{extra}}) \). This can easily be done by vectorizing the two models and multiply or add them element-wise. For example, for a 2-port this gives:

\[
\hat{G}(\Omega_p, \theta) = \begin{bmatrix}
N_{11}(\Omega_p, \theta) & N_{12}(\Omega_p, \theta) \\
D(\Omega_p, \theta) & D(\Omega_p, \theta)
\end{bmatrix}
\]  
(3-37)
The multiplying case of (3-28) is easier for the common-denominator representation than for the state-space representation.

### 3.8 Stability enforcement software

#### 3.8.1 RARL2

In a first step a bounded-input/bounded-output stable transfer function is recovered from a family of experimental point-wise values on the imaginary axis. In [5] it has been shown that the identification procedure consists of a map from finite sets of data to stable transfer functions that converges uniformly in the bandwidth while meeting some norm constraints at remaining frequencies. The technique is also applicable on the unit disc.

This step was implemented in the software package Hyperion [32]. For each $G_{ij}$, Hyperion searches a function $\hat{G}_{ij}$, which is analytic in the disc (i.e. stable), that approximates $G_{ij}$ at best in the measurement interval, and is not too big outside. In fact, the L2 norm outside the interval is equal to $M$, a number given by the user. This first step can fail, in case where the system to identify is unstable, too noisy, or has lags, etc. The bigger $M$ the smaller the approximation error in the measured frequency band.

In a second step this stable transfer function is approximated by a rational form of a user defined order [43]. This step was implemented in the software package RARL2. Here the stability constraint is taken into account by the parameterization.

This two-step approach looks similar to the approach presented in this chapter. The main difference lies in the first step. The approximation error in of the approach presented in this thesis is bounded to the order that is selected using a model selection criterion (whiteness test...
of the residuals, value of the cost function, ...). Hyperion can make this error arbitrary small in
the user-defined frequency band, at the cost of destabilizing at the remaining frequencies.

3.8.2 Vector fitting

Vector Fitting (VF) is a robust numerical method for rational approximation in the frequency
domain. It permits to identify state-space models directly from measured or computed
frequency responses, both for SISO or MIMO systems.

Basically, VF is a pole relocation technique where the poles are improved on in an iterative
manner. This is achieved by repeatedly solving a linear problem until convergence is achieved.
Convergence is usually very fast (2-3 iterations). Unstable poles are flipped into the left half
plane to enforce stable poles. This makes VF applicable to high order systems \((n_a > 100\) ) and
wide frequency bands.

VF identifies the poles \(a_m\), residues \(r_m\), and terms \(d\) and \(h\) (optional) of the model

\[
G(s) = \sum_{m=1}^{n_a} \frac{r_m}{s - a_m} + d + sh \quad \text{(3-39)}
\]

The poles are identified by solving the following problem that is linear in \(r_m, r_m, d\) and \(h\)

\[
\sigma(s)G(s) = \sum_{m=1}^{n_a} \frac{r_m}{s - a_m} + d + sh \quad \text{(3-40)}
\]

with

\[
\sigma(s) = \sum_{m=1}^{n_a} \frac{r_m}{s - a_m} + 1 \quad \text{(3-41)}
\]

Where the poles \(a_m\) in (3-40) and (3-41) are a set of initial (chosen) poles. Next (3-40) and (3-
41) are rewritten in their pool-zero-gain representation. This results in:
Stable Models

\[
\sigma(s)G(s) = h \prod_{m=1}^{n_a} \left( s - z_m \right) \quad \sigma(s) = \prod_{m=1}^{n_a} \left( s - \tilde{z}_m \right)
\]

\[
G(s) = \prod_{m=1}^{n_a} \left( s - z_m \right)
\]

From (3-43) it is clear that the poles \( a_m \) of \( G(s) \) must be equal to the zeros of \( \sigma(s) \). The new poles are substituted in (3-40) and finally, the residues \( r_m \) and terms \( d \) and \( h \) in (3-39) are calculated, which is again a linear problem. This process is repeated until convergence (normally 2-3 iterations).

VF is applicable also to vectors of frequency responses (hence its name), which results in an approximation with a common pole set.

### 3.9 Disadvantages

The advantage of the stability enforcement approach presented in this thesis is, that the uncertainty bounds extracted from the first step, allows the user to get an idea about the bias, if any, that is introduced by approximating the unconstrained model by a stable one. Of course since we need two steps, one estimation step and one approximation step, this approach will probably have a longer calculation time than the one-step approaches. Another disadvantage lies in the fact that the global minimum of (3-6) lies in the RHP. This means that the stable approximant can get stuck on the stability border during the refinement step. The obtained
local minimum is of course useful when the introduced bias is lower than the uncertainty bounds of the unconstrained model.

3.10 Appendix

3.10.1 Lemma of Lyapounov

\[ P_i A + A^T P_i = -C_i^T C_i \]  

for \( i = 1, \ldots, n \) and \( C_i \) linearly independent. Summation over the \( i \) results in

\[
\left( \sum_{i=1}^{n} P_i \right) A + A^T \left( \sum_{i=1}^{n} P_i \right) = - \sum_{i=1}^{n} C_i^T C_i
\]  

(3-45)

Now if one chooses \( C_i = \begin{bmatrix} 0 & \ldots & 1 & \ldots & 0 \end{bmatrix} \) with only a 1 on the \( i \)th element one finds

\[
\sum_{i=1}^{n} C_i^T C_i = I_n
\]  

(3-46)

3.10.2 Conservation of the phase with initial 4

Initial 4 of the unconstrained model \( G(\Omega) \) is created by multiplying \( G(\Omega) \) with (3-47) and (3-48).

\[
\det((\Omega I_u - A_u)(\Omega^* I_u - A_u)) = \det(\Omega I_u - A_u)\det(\Omega^* I_u - A_u) = P(\Omega, A_u)
\]  

(3-47)

\[
P(\Omega, A_u) = \begin{cases} 
-\det(s^2 I_u - A_u^2) & \text{for } \Omega = s \\
\det(I_u - A_u(z + z^{-1}) + A_u^2) & \text{for } \Omega = z
\end{cases}
\]  

(3-48)

From (3-9), (3-10), (3-11) and (3-47) it is clear that multiplying \( G(\Omega) \) by \( \det((\Omega I_u - A_u)(\Omega^* I_u - A_u)) \) will annihilate the unstable poles of \( G(\Omega) \), while leaving the
initial stable poles of $G(\Omega)$ untouched. The initial unstable poles of $G(\Omega)$ are changed to new zeros $\det(\Omega I_u + A_u)$.

\[
G(\Omega)P(\Omega, A_u) = (D + C(\Omega I - A)^{-1}B)\det((\Omega I_u - A_u)(\Omega^* I_u - A_u)) \tag{3-49}
\]

\[
(\Omega I - A)^{-1} = \frac{\text{adj}(\Omega I - A)}{\det(\Omega I_u - A_u)\det(\Omega I_s - A_s)} \tag{3-50}
\]

Combining (3-49) and (3-50) gives:

\[
G(\Omega)P(\Omega, A_u) = \frac{(D\det(\Omega I - A) + C\text{adj}(\Omega I - A)B)\det(\Omega^* I_u - A_u)}{\det(\Omega I_s - A_s)} \tag{3-51}
\]

where \text{adj} correspond to the adjunct of a matrix. This operation preserves the original phase spectrum, but changes the amplitude spectrum. Indeed, for respectively the $s$-domain and $z$-domain, the phase of $P(\Omega, A_u)$ is zero:

\[
P(j\omega, A_u) = \det(j\omega I_u - A_u)\overline{\det(j\omega I_u - A_u)} = |\det(j\omega I_u - A_u)|^2 \tag{3-52}
\]

\[
P(e^{j\omega}, A_u) = \det(e^{j\omega} I_u - A_u)\overline{\det(e^{j\omega} I_u - A_u)} = |\det(e^{j\omega} I_u - A_u)|^2 \tag{3-53}
\]

The above stabilization technique is illustrated on the following first order example in the $s$-domain ($a > 0$).

\[
G(s) = \frac{1}{(s - a)} \tag{3-54}
\]

$G(s)$ is stabilized as follows:

\[
G_{\text{stable}}(s) = -G(s)\det(a^2 - s^2) = s + a \tag{3-55}
\]

\[
\angle G(j\omega) = \tan\left(\frac{\omega}{a}\right) \tag{3-56}
\]

\[
\angle G_{\text{stable}}(j\omega) = \tan\left(\frac{\omega}{a}\right) \tag{3-57}
\]
From (3-56) and (3-57) it is clear that the phase spectrum is preserved. The equivalence in the $z$-domain ($a > 1$)

$$G(z) = \frac{1}{(z - a)}$$  \hspace{1cm} (3-58)

$G(z)$ is stabilized as follows:

$$G_{stable}(z) = G(z)\text{det}((z - a)(z^{-1} - a)) = \frac{1 - az}{z}$$  \hspace{1cm} (3-59)

$$G_{stable}(e^{j\theta}) = \cos \theta - j\sin \theta - a$$  \hspace{1cm} (3-60)

$$\angle G(e^{j\theta}) = \tan\left(-\frac{-\sin \theta}{\cos \theta - a}\right)$$  \hspace{1cm} (3-61)

$$\angle G_{stable}(e^{j\theta}) = \tan\left(-\frac{-\sin \theta}{\cos \theta - a}\right)$$  \hspace{1cm} (3-62)

From (3-61) and (3-62) it is clear that the phase spectrum is preserved.
4.1 Introduction

In this chapter the second (approximation) step with passivity constraints is explained. A set of passive starting values are proposed and the same refinement algorithm as in Chapter 3 is used. However, this approximation step is much harder for the passivity constraints than for the stability constraints. All the described techniques are explained for continuous-time models; and the analogy to discrete-time models is made. The conservation of the passivity property of discretized continuous-time models is also studied. Some post-processing techniques are proposed for creating minimax and higher order passive models.

4.2 Passive system

4.2.1 Definitions

A system is called passive if it cannot generate energy. For a \( n \)-port system, assumed to be storing no energy at \( t_0 \), one can say that the
energy received by the \( n \)-port is at each time instance always bigger than or equal to the energy generated by the \( n \)-port. Since the purpose of system identification is to describe a system by a mathematical model, one needs a clear definition of a passive model. A model is called passive if the corresponding transfer matrix is positive real (PR) (in case of a hybrid representation [3], for example admittance or impedance functions) or bounded real (BR) (in case of a scattering representation). An important sub group of the passive systems are the so-called dissipative systems. Like a passive network is made by passive elements, a dissipative network is made up by lossy elements [44]. Now a model is called dissipative if its corresponding transfer matrix is strictly positive real (SPR) or strictly bounded real (SBR).

The definitions will first be stated for continuous-time (CT) multi input multi output (MIMO) systems, and next for discrete-time (DT) systems. Further, it will be verified whether some well-known continuous-time to discrete-time transformations (impulse invariant, bilinear, zero-order-hold and first-order-hold) preserve the passivity property.

### 4.2.2 Continuous-time systems

Consider a continuous-time linear time-invariant system described by its state-space equations

\[
\begin{align*}
\frac{dx_c(t)}{dt} &= A_c x_c(t) + B_c u_c(t) \\
y_c(t) &= C_c x_c(t) + D_c u_c(t)
\end{align*}
\]  

(4-1)

where \( x_c(t) \) is an \( n \)-dimensional real vector, \( u_c(t) \) and \( y_c(t) \) are \( m \)-dimensional vectors and \( A_c, B_c, C_c \) and \( D_c \) are constant real matrices of appropriate dimensions. The transfer matrix of (4-1) from \( u_c(t) \) to \( y_c(t) \) is given by

\[
G_c(s) = D_c + C_c(sI_n - A_c)^{-1}B_c
\]  

(4-2)

**Definition 4.1** A real rational transfer matrix \( G_c(s) \) is passive, if and only if it is positive real (PR) [3, 6]. \( G_c(s) \) is PR if and only if it satisfies

\[
G_c(s) \text{ is analytic, for } \text{Re}(s) > 0
\]  

(4-3)
In practice condition (4-4) is difficult to verify and it can be replaced by conditions 2 and 3 of the following definition which is equivalent to the previous one [3].

**Definition 4.2**

1. \( G_c(s) \) is analytic, for \( \text{Re}(s) > 0 \) \hspace{1cm} (4-5)
2. The poles of \( G_c(s) \) on the \( j\omega \)-axis are simple and the associated residue matrices of \( G_c(s) \) at these poles are positive semi-definite. \hspace{1cm} (4-6)
3. \( G_c(j\omega) + G_c^T(-j\omega) \geq 0 \quad \forall \omega \in \mathbb{R} \) which are not poles of \( G_c(j\omega) \) \hspace{1cm} (4-7)

**Definition 4.3** A real rational transfer matrix \( G_c(s) \) is a strictly positive real matrix if \( G_c(s - \varepsilon) \) is a positive real matrix for some \( \varepsilon > 0 \).

In [66] it has been shown that Definition 4.3 can be rewritten as

**Definition 4.4** An \( m \times m \) rational matrix \( G_c(s) \) is a strictly positive real matrix if and only if

\[
G_c(s) \text{ is analytic, for } \text{Re}(s) \geq 0
\]

\[
G_c(j\omega) + G_c^T(-j\omega) > 0 \quad \forall \omega \in \mathbb{R}
\]

\[
\lim_{\omega^2 \to \infty} \omega^2[G_c(j\omega) + G_c^T(-j\omega)] > 0, \text{ if } \det[D_c + D_c^T] = 0
\]

\[
\lim_{\omega \to \infty} [G_c(j\omega) + G_c^T(-j\omega)] > 0, \text{ if } \det[D_c + D_c^T] \neq 0
\]
Using the bilinear relation between (S)PR en (S)BR, one can deduce the following definition

\[ Z = (I + S)(I - S)^{-1} \]  

(4-12)

with \( Z \) the impedance matrix and \( S \) the normalized scattering matrix.

**Definition 4.5** A real rational scattering function matrix \( S_c(s) \) is passive, if and only if it is bounded real [3, 6]. \( S_c(s) \) is bounded real if and only if it satisfies

\[ S_c(s) \text{ is analytic, for } \text{Re}(s) \geq 0 \]  

(4-13)

\[ I - S_c(-j\omega)^T S_c(j\omega) \geq 0 \quad \forall \omega \in R \]  

(4-14)

**Definition 4.6** A real rational transfer matrix \( S_c(s) \) is a strictly bounded real matrix if \( S_c(s - \varepsilon) \) is a bounded real matrix for some \( \varepsilon > 0 \). So (4-14) becomes

\[ I - S_c(-j\omega)^T S_c(j\omega) > 0 \quad \forall \omega \in R \]  

(4-15)

**4.2.2.1 Checking the passivity**

Direct application of the definitions (4-7) and (4-9) for testing passivity, requires a frequency sweep from DC to infinity since these conditions need to be checked at any frequency on the imaginary axis. Therefore, the results of such tests depend on a sufficiently dense sampling of the frequency axis, which is not a trivial task. However, fitting data at a set of discrete points in general does not represent a strong enough constraint to ensure positive- or bounded-realness over the entire imaginary axis. The reason for this is that the behavior of the model outside the frequency range or in between the frequency points cannot be controlled. Erroneous results may occur. Therefore, purely algebraic passivity tests are highly desirable. Fortunately, the Positive Real Lemma and the Bounded Real Lemma provide an answer to this problem [3].
These results provide a connection between the passivity definitions and various equivalent algebraic conditions. These conditions can be expressed via the feasibility of linear matrix inequalities, or via the existence of solutions to equivalent algebraic Riccati equations, or via the spectral properties. These spectral properties can be verified via the associated Hamiltonian matrices, or the poles and spectral zeros of the transfer matrices. These results are summarized in the following theorems.

**Theorem 4.7** $G_c(s)$ (4-2) (state space formulation) is PR if and only if $G_c(s)$ is stable and the eigenvalues $\lambda$ of the following Hamiltonian (4-16) for respectively the hybrid representation are quadrant symmetric if they are complex, have even multiplicity if they are purely imaginary, and are symmetric if they are purely real.

\[
\begin{bmatrix}
A_c - B_c Q^{-1} C_c & -B_c Q^{-1} B_c^T \\
C_c^T Q^{-1} C_c & -A_c^T + C_c^T Q^{-1} B_c^T
\end{bmatrix}
\]  

(4-16)

Equation (4-16) is derived from the Riccati equations [3, 26] with $Q = D_c + D_c^T > 0$.

Proof: see reference [3].

**Theorem 4.8** $G_c(s)$ (4-2) (transfer matrix formulation) is PR if and only if the conditions of Table 4-1 are fulfilled.

Proof: The first passivity constraint (4-3) and (4-8) imposes constraints on the position of the poles of $G_c(s)$ (see Table 4-1). The equivalent of the second passivity condition (4-7) and (4-9) can be found by calculating the zeros of the rational transfer function matrix:

\[
\det(G_c(s) + G_c^T(-s)) = 0
\]

(4-17)
Passive models

which is the analytic continuation of condition (4-7) and (4-9). The zeros of (4-17) are the same as the eigenvalues of (4-16) (see Appendices 4.10.2 and 4.10.3). The constraints imposed on the zeros of (4-17) or on the eigenvalues of (4-16) are shown in Table 4-1 and the proof is given in Appendix 4.10.1. However, this test can not make the difference between negative real functions and positive real functions. To make a distinction between these 2 cases the value of (4-7) at one frequency point (for example: infinity or DC) must be checked. This should of course be bigger than zero for strictly passive models and bigger than or equal to zero for passive models.

In [2, 3] it is shown that the case when is singular can be solved for synthesis problems. For example, for a positive real for which is singular, there exists a sequence of elementary manipulations on , such as the shunt extraction of capacitors, which reduces the problem of synthesizing to one of synthesizing a second positive real matrix, , say, for which is non-singular. To handle improper models, one need to check if the high frequency behavior is positive (see Appendix 4.10.3).

Equivalent constraints can be derived for BR systems, and (4-17) and (4-16) become then respectively

<table>
<thead>
<tr>
<th>constraints on:</th>
<th>$s$-domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>zeros of $G_c(s)$</td>
<td>left half plane</td>
</tr>
<tr>
<td>poles of $G_c(s)$</td>
<td>left half plane</td>
</tr>
<tr>
<td>complex zeros of $\det(G_c(s) + G^T_c(-s))$</td>
<td>$s_0$, $\overline{s_0}$, $-s_0$, $-\overline{s_0}$</td>
</tr>
<tr>
<td>imaginary zeros of $\det(G_c(s) + G^T_c(-s))$</td>
<td>even multiplicity</td>
</tr>
<tr>
<td>real zeros of $\det(G_c(s) + G^T_c(-s))$</td>
<td>$s_0$, $-s_0$</td>
</tr>
<tr>
<td>$\exists \omega$ such that $\det(G_c(s) + G^T_c(-s)) \geq 0$</td>
<td>$s = j\omega$</td>
</tr>
</tbody>
</table>

TABLE 4-1. Necessary and sufficient conditions on the hybrid matrices $G_c(s)$ to be passive
Passive system

<table>
<thead>
<tr>
<th>constraints on:</th>
<th>$s$-domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>poles of $S_c(s)$</td>
<td>left half plane</td>
</tr>
<tr>
<td>complex zeros of $\det(I - S_c(-s)^T S_c(s))$</td>
<td>$s_0, \overline{s}_0, -s_0, -\overline{s}_0$</td>
</tr>
<tr>
<td>imaginary zeros of $\det(I - S_c(-s)^T S_c(s))$</td>
<td>even multiplicity</td>
</tr>
<tr>
<td>real zeros of $\det(I - S_c(-s)^T S_c(s))$</td>
<td>$s_0, -s_0$</td>
</tr>
<tr>
<td>$\exists \omega$ such that $\det(I - S_c(-s)^T S_c(s)) \geq 0$</td>
<td>$s = j\omega$</td>
</tr>
</tbody>
</table>

**TABLE 4-2.** Necessary and sufficient conditions on the scattering matrices $S_c(s)$ to be passive

\[
\det(I - S_c(-s)^T S_c(s)) = 0
\]

\[
\begin{bmatrix}
A_c - B_c R^{-1} D_c^T C_c & -B_c R^{-1} B_c^T \\
C_c V^{-1} C_c & -A_c^T + C_c^T D_c R^{-1} B_c^T
\end{bmatrix}
\]

(4-18) \hspace{3cm} (4-19)

with $R = D_c^T D_c$ and $V = D_c^T D_c$ (see Table 4-2).

One last remark should be made when using Table 4-1 and Table 4-2. For SPR and SBR models no poles on the imaginary axis are allowed neither purely imaginary zeros of (4-17) and (4-18).

**4.2.3 Discrete-time systems**

The passivity checking procedures for CT systems follow respectively from the positive real and bounded real lemma under state-space representation (minimal representation):

\[
PA_c + A_c^T P = -LL^T
\]

\[
PB_c = C_c^T - LW_0
\]

(4-20)

\[
D_c + D_c^T = W_0^T W_0
\]
Passive models

\[ P A_c + A_c^T P = -C_c C_c^T - L L^T \]
\[ -P B_c = C_c D_c + L W_0 \]
\[ I - D_c^T D_c = W_0^T W_0 \]  

(4-21)

with \( P, L \) and \( W_0 \) real matrices and \( P \) positive definite and symmetric. However (4-20) till (4-21) are only applicable to continuous-time systems. Equivalent formulations of (4-20) and (4-21) for discrete systems can be found in [30, 73], this results respectively in (4-22) and (4-23).

\[ P - A_d^T P A_d = L L^T \]
\[ C_d^T - A_d^T P B_d = L^T W_0 \]  
\[ D_d + D_d^T - B_d^T P B_d = W_0^T W_0 \]  

(4-22)

\[ A_d^T P A_d + C_d^T C_d + L^T L = P \]
\[ B_d^T P B_d + D_d^T D_d + W_0^T W_0 = I \]
\[ A_d^T P B_d + C_d^T D_d + L^T W_0 = 0 \]  

(4-23)

Consider the discrete-time linear time-invariant system described by

\[ x_d(t + 1) = A_d x_d(t) + B_d u_d(t) \]
\[ y_d(t) = C_d x_d(t) + D_d u_d(t) \]  

(4-24)

where \( x_d(t) \) is an \( n \)-dimensional real vector, \( u_d(t) \) and \( y_d(t) \) are \( m \)-dimensional vectors and \( A_d, B_d, C_d \) and \( D_d \) are constant real matrices of appropriate dimensions. The transfer matrix of (4-1) from \( u_d(t) \) to \( y_d(t) \) is given by

\[ G_d(z) = D_d + C_d (z I_n - A_d)^{-1} B_d \]  

(4-25)

**Definition 4.9** A real rational transfer matrix \( G_d(z) \) is discrete-time positive real (DTPR) if and only if it satisfies

\[ G_d(z) \text{ is analytic, for } |z| > 1 \]  

(4-26)
\( G_d(z) + G_d^T(z^{-1}) \geq 0, \) for \(|z| > 1\) \hspace{1cm} (4-27)

Similar to the continuous-time case, condition (4-27) is difficult to be verified and can be replaced by the following conditions 2 and 3 in the following definition

**Definition 4.10**

1. \( G_d(z) \) is analytic, for \(|z| > 1\) \hspace{1cm} (4-28)

2. If \( z_0 = e^{j\theta_0} \), with \( \theta_0 \) real, is a pole of \( G_d(z) \), and if \( R_0 \) is the residue matrix of \( G_d(z) \) at \( z = z_0 \), then \( e^{-j\theta_0}R_0 \geq 0 \). \hspace{1cm} (4-29)

3. \( G_d(e^{j\theta}) + G_d^T(e^{-j\theta}) \geq 0 \) \( \forall \theta \) at which \( G_d(e^{j\theta}) \) exists \hspace{1cm} (4-30)

Proof: see Appendix 4.10.6

In [73] it has been shown that the analogy between the continuous-time definitions and discrete-time definitions is harder than it seems. This is in particular true for (4-29), which is often wrongly replaced by \( R_0 \) is nonnegative definite Hermitian: see [9, 16, 34 and references in 73].

**Definition 4.11** A real rational transfer matrix \( G_d(z) \) is a discrete-time strictly positive real (DTSPR) matrix if \( G_d(\rho z) \) is a DTSPR matrix for some \( 0 < \rho < 1 \).

In [66] it has been shown that Definition 4.11 can be rewritten in
**Definition 4.12** An \( m \times m \) rational matrix \( G_d(z) \) is a DTSPR matrix if and only if

\[
G_d(z) \text{ is analytic, for } |z| \geq 1
\]

\[
G_d(e^{j\theta}) + G_d^T(e^{-j\theta}) > 0 \quad \forall \theta \in [0, 2\pi]
\]

From the bilinear relation (4-12) between DT(S)PR en DT(S)BR, one can make the following definitions

**Definition 4.13** A real rational scattering function matrix \( S_d(z) \) is passive, if and only if it is bounded real [1]. \( S_d(z) \) is bounded real if and only if it satisfies

\[
S_d(z) \text{ is analytic, for } |z| \geq 1
\]

\[
I - S_d(e^{-j\theta})S_d^T(e^{j\theta}) \geq 0 \quad \forall \theta \in [0, 2\pi]
\]

**Definition 4.14** A real rational transfer matrix \( S_e(z) \) is a strictly discrete-time bounded real matrix [1] if \( S_e(\rho z) \) is a DTBR matrix for some \( 0 < \rho < 1 \). So (4-34) becomes

\[
I - S_d(e^{-j\theta})S_d^T(e^{j\theta}) > 0 \quad \forall \theta \in [0, 2\pi]
\]

4.2.3.1 Checking the passivity

Like for CT systems, the direct application of the definitions (4-30) and (4-32) for testing passivity, however, requires these conditions to be checked at any frequency on the unit circle. However, since in the literature no Hamiltonian matrices were derived from the Discrete-Time Positive Real Lemma and the Discrete-Time Bounded Real Lemma, no algebraic passivity tests are available. Algebraic passivity checking conditions are posed without proof in the Section 4.2.4.
4.2.4 Discrete-time passivity constraints

We are convinced that similar passivity conditions exist for discrete-time models like the ones shown in Table 4-1. However, we do not have a proof of this, since we couldn’t find Hamiltonian expressions of (4-22) and (4-23) in the literature, if they exist. So, we will postulate them as a conjecture and illustrate them with a couple simulations (see Appendix 4.10.4). In Table 4-3 and Table 4-4 one can find the constraints on the discrete-time models (compare to Table 4-1 and Table 4-2).

Note that the order of the numerator and denominator of the discrete model has to be the same, otherwise the spectral zeros or poles can never be quadrant symmetric (see Appendix 4.10.5).

<table>
<thead>
<tr>
<th>constraints on:</th>
<th>$z$ -domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>zeros of $G(z)$</td>
<td>inside the unit circle</td>
</tr>
<tr>
<td>poles of $G(z)$</td>
<td>inside the unit circle</td>
</tr>
<tr>
<td>complex zeros of $\det(G(z) + G^T(z^{-1}))$</td>
<td>$z_0, \bar{z}_0, z_0^{-1}, \bar{z}_0^{-1}$</td>
</tr>
<tr>
<td>zeros $</td>
<td>z</td>
</tr>
<tr>
<td>real zeros of $\det(G(z) + G^T(z^{-1}))$</td>
<td>$z_0, z_0^{-1}$</td>
</tr>
<tr>
<td>$\exists \theta$ such that $\det(G(e^{j\theta}) + G^T(e^{-j\theta})) \geq 0$</td>
<td>$z = e^{j\theta}$</td>
</tr>
</tbody>
</table>

If $R_0$ is the residue matrix of a simple pole $z_0 = e^{j\theta_0}$, then $e^{-j\theta_0}R_0$ nonnegative definite

<table>
<thead>
<tr>
<th>constraints on:</th>
<th>$z$ -domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>poles of $S(z)$</td>
<td>inside the unit circle</td>
</tr>
<tr>
<td>complex zeros of $\det(1 - S(z^{-1})S(z))$</td>
<td>$z_0, \bar{z}_0, z_0^{-1}, \bar{z}_0^{-1}$</td>
</tr>
<tr>
<td>zeros $</td>
<td>z</td>
</tr>
<tr>
<td>real zeros of $\det(1 - S(z^{-1})S(z))$</td>
<td>$z_0, z_0^{-1}$</td>
</tr>
<tr>
<td>$\exists \theta$ such that $\det(1 - S(e^{-j\theta})S(e^{j\theta})) \geq 0$</td>
<td>$z = e^{j\theta}$</td>
</tr>
</tbody>
</table>

TABLE 4-3. Necessary and sufficient conditions on the admittance or the impedance function $G(\Omega)$ to be passive

TABLE 4-4. Necessary and sufficient conditions on the scattering functions $S(\Omega)$ to be passive
4.2.5 Continuous-time to discrete-time transformations

In many cases a discrete-time model is created by discretizing a continuous-time model. However depending on which discretization technique (Zero-Order-Hold (ZOH), First-Order-Hold (FOH), bilinear or impulse invariant) one chooses, the passivity properties are preserved or not. The bilinear (4-36) or impulse invariant (4-37) transformation with $T_s$ the sampling period, preserve the passivity.

\[ s = \frac{2}{T_s} \left( \frac{z - 1}{z + 1} \right) \quad (4-36) \]

\[ z = e^{jsT_s} \quad (4-37) \]

The proof of this is given in [34]. However since [34], did not use the right definition for the residues of the poles on the unit circle, this proof is completed in appendix 4.10.6.1 and 4.10.6.2.

The ZOH transformation (4-38) does not preserve the passivity properties, a simple counter-example is given in the appendix 4.10.7.

\[ G_{ZOH}(z) = \left( \frac{z - 1}{z} \right) \mathcal{Z} \left[ \frac{G_c(s)}{s} \right] \quad (4-38) \]

For the FOH transformation we must distinguish two situations, the causal one (4-40) and non-causal one (4-39).

\[ G_{FOH}(z) = \left( \frac{z - 1}{z} \right)^2 \mathcal{Z} \left[ \frac{1}{s^2} G_c(s) \right] \quad (4-39) \]

\[ G_{FOH}(z) = \left( \frac{z - 1}{z} \right)^2 \mathcal{Z} \left[ \frac{1}{s^2} G_c(s) \right] \quad (4-40) \]

The non-causal FOH transformation was treated in [31], were the proof of the residue is completed in appendix 4.10.6.3. The same proof in the causal case is also given in appendix
4.10.6.3. From appendix 4.10.6.3 and [31] it follows that the non-causal FOH transformation does preserve the passivity properties, while the causal FOH transformation does not preserve it.

4.2.6 Reciprocity

A distinction has to be made between the property of a passive model and a realizable \( n \)-port of passive construction blocks (resistor, inductance, capacitor and transformer). \( n \)-ports constructed from these building blocks will be represented by a reciprocal hybrid matrix. From the energy point of view a gyrator is also a passive element [6]. However, this passive element will destroy the reciprocity of the \( n \)-port. So reciprocity is not a necessary condition of a model to be passive or not. This is important if one likes to realize the passive model by an equivalent circuit in SPICE like demonstrated in [47], which does not use gyrators.

4.2.7 Properties of a BRF and PRF

The properties of PRF are described extensively in the literature [3, 6, 8, 27, 67]. The most important ones will be repeated here and will be used in the following sections.

Property 4.15 If \( G_1(\omega) \) and \( G_2(\omega) \) are a PRF, then is \( G_1(\omega) + G_2(\omega) \) a PRF (\( \omega = s \) or \( z \)).

Property 4.16 In [3] it is shown that if \( S(\omega) \) is BRF and if \( 1/(1 - S(\omega)) \) exists for all \( \omega \) in \( \text{Re}(s) > 0 \) or \( |z| > 1 \), then \( G(\omega) = (1 + S(\omega))/(1 - S(\omega)) \) is positive real. The inverse relation between PRF and BRF is trivial.

Property 4.17 If \( G(\omega) \) is a PRF, then is \( 1/G(\omega) \) also a PRF. From this property one recognizes that the zeros of \( G(\omega) \) must also be within the unit circle or the LHP. Hence, the numerator and the denominator must be Hurwitz polynomials.

Property 4.18 (continuous-time only) The lowest as well as the highest powers of numerator and denominator polynomials can differ at most by unity. This follows from the fact that the poles
and zeros at infinity and DC of a PRF $G(s)$ should be simple. This is a special case of the property that the zeros and poles on the stability border must be simple.

The techniques of the following sections are only studied for continuous-time systems. So we will use $G(s)$ instead of $G_c(s)$.

### 4.3 Passivity enforcement

The passivity enforcement algorithm is analogue to the stability enforcement algorithm of the previous chapter. So starting from an unconstrained non-passive model $G(s)$ that is already estimated with an optimal noise removal criterion, the following cost function $V$ in a user specified frequency band has to be minimized subject to the constraints of Table 4-1:

$$V = \frac{1}{F} \sum_{l=1}^{F} w_l^2 \left| G(s_l) - \hat{G}(s_l, \theta) \right|^2$$

(4-41)

with $\hat{G}(s, \theta)$ the passive version of $G(s)$, $\theta$ the transfer function parameters, $w_l$ a user defined weighting function, and $F$ the number of frequencies.

#### 4.3.1 Starting values for SISO systems

Now starting from the unconstrained model $G(s)$, we want to find some passive starting values for the iterative algorithm that minimizes $V$. In this work, it is not the purpose to compare all the different passivity enforcement methods, so we will focus on simple time efficient algorithms, for generating passive initial values. Most passivation methods requires a stable model [13, 18, 26, 29, 42, 44, 47], otherwise the passivity constraints are non-linear in the parameters and cannot be solved easily. Of course all these techniques can be used as starting values for the passive iterative algorithm of Section 4.3.3.

The first and second passivity constraint (Table 4-1) can be easily imposed. This can be done by flipping the poles and zeros, who are violating this condition, around the imaginary axis or
around the unit circle. This preserves the amplitude spectrum but changes the phase spectrum. The constraints on $\hat{G}(s, \theta) + \hat{G}(-s, \theta)$ are harder to correct without destroying the good approximation of the system. The most simple ways exist in deplacing the zeros. This can be done for example by adding a constant term to the model $\hat{G}(s, \theta)$. However this mostly results in a poor passive approximation of the system.

A better method consists in decomposing the rational transfer function in a sum of low-order rational functions (partial fraction expansion). Now a sufficient but not necessary condition is to replace each of these low-order functions by their passive approximations.

$$G(s) = \sum_{k=1}^{k_{max}} \frac{b_{1k}s + b_{0k}}{s^2 + a_{1k}s + a_{0k}} + \sum_{l=1}^{l_{max}} \frac{b_{0l}}{s + a_{0l}} + (c + ds)$$  \hspace{1cm} (4-42)

From Property 4.15 it is clear that (4-42) is a PRF, if all the contributions are PRF’s. The constant and linear term are PRF’s if and only if $c, d > 0$. The first order contributions of (4-42) are passive if $a_{0l}, b_{0l} > 0$ for $l = 1 \ldots l_{max}$.

$$G_{1l}(s) + G_{1l}(-s) = \frac{b_{0l}}{s + a_{0l}} + \frac{b_{0l}}{-s + a_{0l}} = \frac{2b_{0l}a_{0l}}{(s + a_{0l})(-s + a_{0l})}$$  \hspace{1cm} (4-43)

Indeed, applying the constraints of Table 4-1 to the first order contributions, it is clear that the poles of $G_{1l}(s) + G_{1l}(-s)$ are quadrant symmetric and that $2G_{1l}(0) > 0$. The second order contributions of (4-42) are passive if $a_{ik}, b_{ik} > 0$ and $a_{ik}b_{1k} - b_{0k} > 0$ for $i = 0, 1$ and $k = 1 \ldots k_{max}$.

$$G_{2k}(s) + G_{2k}(-s) = \frac{2(b_{0k} - b_{1k}a_{1k})s^2 + 2a_{0k}b_{0k}}{(s^2 + a_{1k}s + a_{0k})(s^2 - a_{1k}s + a_{0k})}$$  \hspace{1cm} (4-44)

Indeed, applying the constraints of Table 4-1 on the second order contributions, it is clear that the poles and zeros of $G_{2k}(s) + G_{2k}(-s)$ are quadrant symmetric and that $2G_{2k}(0) > 0$. 

---

73
Now from the constraints on the coefficients of the first and second order contributions of $G(s)$ we can propose an approach to create passive starting values. All the $a_i$ should be set positive by flipping the unstable poles with respect to the stability border. This results in stabilizing the model. A stable model can also be created using the techniques described in the previous chapter. The $b_i$ can however be negative, so in a first step one should take the absolute value of $b_i$. This step is followed by adapting $b_{ik}$ and/or $b_{0k}$ so that $a_{ik}b_{1k} - b_{0k} > 0$ is fulfilled. One could think to adapt $a_{ik}$, however this corresponds to the (flipped) original poles, which are probably the best starting values for the $a_{ik}$. The two easiest methods consist in adapting only $b_{1k}$ or $b_{0k}$. A more sophisticated approach consists in changing $b_{1k}$ and $b_{0k}$ at the same time by quadratic constrained programming. This can be done with the routine quadprog in MATLAB.

To get passive starting values for scattering representations one can use the bilinear transformation to transform the scattering model to an impedance or admittance model.

### 4.3.2 Starting values for MIMO systems

If the same approach is used to create passive starting values for multivariable models, then some problems can occur. Consider the following $2 \times 2$ common denominator model $G(s)$.

$$
G(s) = \begin{bmatrix}
- \sum_{j=0}^{n_b} b_j s^j & \sum_{l=0}^{n_d} d_l s^l \\
\sum_{k=0}^{n_c} c_k s^k & \sum_{m=0}^{n_e} e_m s^m \\
\sum_{i=0}^{n_a} a_i s^i
\end{bmatrix}
$$

(4-45)

The same constraints as for the SISO case can be used to make all the diagonal elements of $G(s)$ PRF. This will put restrictions on all the $a_i$, $b_j$, $e_m$ in the same way as for (4-42). Since a necessary condition for $G(s)$ to be passive, consists of the positive definite nature of
Passivity enforcement

\[ G(j\omega) + G^T(-j\omega), \] a difficult problem remains in the choice of the remaining \( c_k, d_l \). This is not a trivial task and becomes even more complex if the multiport has even more than 4 ports. If one stabilizes the denominator by, for example, flipping the unstable poles, one can create passive MIMO starting values by applying for example [29, 47] to the residues of \( G(s) \).

4.3.3 Refinement algorithm

The refinement step is similar to the one in the Chapter 3 Section 3.3.4. The main difference is that the passivity constraint is implemented instead of the stability constraint. To check the constraints shown in Table 4-1 and Table 4-2 for multivariable system, a symbolic calculation toolbox [55] is used to keep the calculations simple. Similar to the stability case, the cost function (4-41) will be lowered by leaving some freedom to the gain and the positions of the zeros and the poles of the transfer matrix. So the refinement step will create passive models that are as least as good as their passive initial values.

In most examples the first constraint that is violated, is the quadrant symmetry of the zeros of \( \det(G(s) + G^T(-s)) \). This is not surprising, since this constraint is violated as soon as one eigenvalue of \( G(j\omega) + G^T(-j\omega) \) from DC till infinity is no longer positive. If this value lies outside the frequency band of interest, there is reasonable hope that the algorithm will not get stuck in a local minimum, but if it is inside the frequency band the algorithm will usually stop here.

Although the passive starting values are created by rewriting the model in its pole-residue-gain representation, the parameters used for the refinement algorithm are the coefficients of the numerator and denominator. The astute reader may wonder why the refinement algorithm is not simply based on the decomposition of the transfer function into sums of positive-real functions, as suggested in [47]. The reason is that not all positive-real rational functions can be represented as the sum of low-order rational functions, each positive real. It is in fact a sufficient but not necessary condition for a rational function to be positive-real. A trivial counter-example can be constructed by taking two scalar positive real functions, one whose real part is larger than the other uniformly in the RHP, and subtracting them. For this reason,
algorithms based on summing low-order positive real functions are always less accurate than algorithms that utilize representations with one-to-one correspondence with the full set of high-order positive-real functions, and may not be able to fit complicated data at all.

### 4.4 Generating passive models with convex solvers

The challenge, and art, in using convex optimization lies in recognizing and formulating the problem. Once this formulation is done, solving the problem is, like least-squares or linear programming, (almost) technology [7].

In the literature [13-15, 18, 23, 29, 44, 45] the passivity problem (for SISO as well as for MIMO) is often solved in a convex way by second-order-cone programming or using semi-definite programming [7]. Since the passivity constraints are non-linear in the parameters, for example (4-20) and (4-21), some assumptions have to be made. The basic assumption is that one has a good estimation of the stable poles [13, 18, 23, 29]. Under this assumption, the stable denominator is kept fixed and the Kalman-Yacubovich-Popov (KYP) lemma becomes a Linear Matrix Inequality (LMI). Under the same conditions (4-41), which is a NLS-FRF with passivity constraints, becomes a weighted LS problem with passivity constraints. It is worthwhile to try the overall approach, since, for an accurate rational approximation to be generated, it first must have been possible to get good estimates for the poles that are relevant to the frequency band of interest. Within the convex solutions, one can find grid based passivity testing [14, 29], or parametric passivity testing [13, 18, 23].

The idea of finding the global minimum is very attractive, so these methods would be the ideal candidates for excellent starting values for the refinement algorithm presented in Section 4.3.3. To keep the starting value step simple, a technique based on [18, 44, 45] was proposed. Starting from a stable transfer function $G(s, \theta)$, the numerator will be modified with a convex solver to fulfil the positive real constraints, while the denominator is kept fixed. In [44, 45] it has been shown that there is a linear relationship between the numerator coefficients of
Generating passive models with convex solvers

\( G(s, \theta) + G^T(-s, \theta) \) and the numerator coefficients of \( G(s, \theta) \). Equations (4-48) and (4-49) show the relation between these coefficients.

\[
G(s, \theta) = \frac{B(s, b_i)}{A(s, a_j)} = \frac{b_m s^m + \ldots + b_1 s + b_0}{a_n s^n + \ldots + a_1 s + a_0} \tag{4-46}
\]

\[
G(s, \theta) + G^T(-s, \theta) = \frac{c_p s^{2p} + \ldots + c_1 s^2 + c_0}{(a_n s^n + \ldots + a_1 s + a_0)(a_n (-s)^n + \ldots + a_1 (-s) + a_0)} \tag{4-47}
\]

\[
\begin{bmatrix}
  c_0 & c_1 & c_2 & \ldots & c_{p-1} & c_p
\end{bmatrix}^T = 2X(a) \cdot \begin{bmatrix}
  b_0 & b_1 & b_2 & \ldots & b_i & b_{m-1} & b_m
\end{bmatrix}^T \tag{4-48}
\]

\[
X(a) = \begin{bmatrix}
  a_0 & 0 & 0 & \ldots & 0 & \ldots & 0 & 0 \\
  a_2 & -a_1 & a_0 & \ldots & 0 & \ldots & 0 & 0 \\
  a_4 & -a_3 & a_2 & \ldots & (-1)^i \cdot a_{4-i} & \ldots & 0 & 0 \\
  \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\
  a_n & -a_{n-1} & a_{n-2} & \ldots & (-1)^i \cdot a_{n-i} & \ldots & \ldots & \ldots \\
  \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\
  0 & 0 & 0 & \ldots & 0 & \ldots & (-1)^{m-1} \cdot a_{n-1} & (-1)^m \cdot a_{n-2} \\
  0 & 0 & 0 & \ldots & 0 & \ldots & 0 & (-1)^m \cdot a_n
\end{bmatrix} \tag{4-49}
\]

With \( X(a) \in \mathbb{R}^{p \times m} \) is a function of the denominator coefficients \( a \) and \( p = \min(n, m) \). In order to be sure that the \( c \)’s correspond to a PRF, one can use the following sufficient, but not necessary condition. The \( c_k \geq 0 \) if \( k \) even and \( c_k \leq 0 \) if \( k \) odd, for all \( k = 0, 1, \ldots, p - 1, p \).

Here one has to distinguish three cases:

1.  \( m = n - 1 \)

   In this case \( X(a) \in \mathbb{R}^{m \times m} \), so there are as many equations as variables and there is a bijective relation between the \( c \)’s and the \( b \)’s.

2.  \( m = n \)

   Like in the previous case \( X(a) \) is a square matrix for strictly proper models \( (m = n) \), so there is a bijective relation between the \( c \)’s and the \( b \)’s.

77
3. \( m = n + 1 \)

In this case there are less equations than variables. In [44] the improper case is solved
were \( b_m \) is chosen as a positive real value, which is responsible for the high frequency
behavior. The remaining problem is of the form \( m = n \). Of course, also for this case one
could use the approach of evaluating (4-64) on a dense frequency grid.

To solve this problem we used the optimization toolbox YALMIP [74] written in MATLAB.
This tool itself uses SeDuMi [65] to solve the constraint optimization problem.

\[
\text{minimize}_{b_i} \left( \left| \frac{G(s) - B(s, b_i)}{A(s, a_j)} \right|^2 \right) \tag{4-50}
\]

with the constraints \( c_k \geq 0 \) if \( k \) even and \( c_k \leq 0 \) if \( k \) odd, for all \( k = 0, 1, \ldots, p - 1, p \).

In general the starting values look better and have a lower complex error than the residue
method. However, once \( n \) or \( m \) becomes larger than 20, the solver ran in numerical problems,
even with proper frequency weighting. Nevertheless, the passive starting value with the
smallest value for (4-41), does not guarantee to be the best solution after the refinement step. It
often happens that the refinement step can not improve any further the starting values from this
convex solver, since the \( G(j\omega, \theta) + G^T(-j\omega, \theta) \) is already very close to zero at some
frequencies.

As further research, other convex solving methods [13-18, 23, 29, 45] which keep the poles
fixed should be tried out as starting values for the refinement step. However, if these models are
very close to the constraint border, it is unlikely to improve these models can be improved
further by the refinement step.

4.5 Passivity refinement by inversion of the transfer function

In most of the cases the first constraint that is violated by the refinement algorithm is
\( \text{Re}(G(j\omega, \theta)) \geq 0 \). This corresponds with one or more zeros of \( \text{det}(G(s, \theta) + G^T(-s, \theta)) \) that
lie on the imaginary axis or are very close to the imaginary axis. Whenever this constraint is
reached, the algorithm will stop improving. The following can be used to escape from these
Reparameterizing

local minima. From the properties of a PRF it follows that the inverse of a PRF is also a PRF. Hence, imposing the PRF conditions on $1/G(s, \theta)$ is a necessary and sufficient condition for $G(s, \theta)$ to be PRF. The real part of a SISO transfer function and its inverse are given by:

$$\text{Re}(G(j\omega, \theta)) = \frac{B(j\omega, b_i)}{A(j\omega, a_j)} + \frac{B(-j\omega, b_i)}{A(-j\omega, a_j)} = \frac{A(j\omega, a_j)B(-j\omega, b_i) + A(-j\omega, a_j)B(j\omega, b_i)}{A(j\omega, a_j)A(-j\omega, a_j)}$$

(4-51)

$$\text{Re}\left(\frac{1}{G(j\omega, \theta)}\right) = \frac{A(j\omega, a_j)}{B(j\omega, b_i)} + \frac{A(-j\omega, a_j)}{B(-j\omega, b_i)} = \frac{A(j\omega, a_j)B(-j\omega, b_i) + A(-j\omega, a_j)B(j\omega, b_i)}{B(j\omega, b_i)B(-j\omega, b_i)}$$

(4-52)

From (4-51) and (4-52) one can see that the numerator is the same, while the denominator is different for both formulas. Now, whenever the algorithm gets stuck in local minima for (4-51), we continue iterating on the inverse function. Thanks to the different denominator of (4-51) and (4-52) it is possible to escape from these minima.

This method is also available for MIMO systems. However, there is no simple formula to calculate the parameters of the inverse model starting from the model parameters, independently of their parameterization (common denominator or state-space). So, whenever a passive $A, B, C$ and $D$ (state-space representation) is found, $G(j\omega, \theta)$ should be evaluated over a dense frequency grid. This data should be inverted, and a new set of state-space variables should be estimated from it. These parameters correspond by definition to a passive model. Finally, we can continue the refinement step on this inverse model. The drop in cost function was in general not more than 10 percent for the measurement examples (see Chapter 5). No drop in cost function was obtained for examples with highly resonant poles.

**4.6 Reparameterizing**

Similar to Section 4.5, this section deals with the problem of the local minima due to the violation of the passivity constraint $\text{Re}(G(j\omega, \theta)) \geq 0$. The described technique is at the moment only applicable to SISO models. Here the aim is to use another parametrization for $G(s, \theta)$ in order to be able to fix the zeros of $G(s, \theta) + G^T(-s, \theta)$, whenever they reach the constraint border (see Table 4-1 and Table 4-2), and let the other ones free. $G(s, \theta)$ is
parameterized in \( b_0, \ldots, b_m, a_0, \ldots, a_n \) (see (4-46)), while \( G(s, \theta) + G^T(-s, \theta) \) in \( c_0, \ldots, c_p, a_0, \ldots, a_n \) (see (4-47)). The relation between both is given by (4-48). Although the cost function (4-41) is parameterized in \( b_i \), the refinement step will be calculated in \( c_i \). The following adaptation has to be carried out to calculate the Jacobian of \( c_k \) for \( k = 0, \ldots, p \):

\[
J_{a_j} = \frac{\partial e}{\partial a_j} = \left( -\frac{B(s, b_i)}{A(s, a_j)^2} \right) (-s^j)
\]

\[
J_{b_i} = \frac{\partial e}{\partial b_i} = \left( -\frac{1}{A(s, a_j)} \right) (s^i)
\]

\[
J_{c_k} = \frac{\partial e}{\partial c_k} = \sum_{i=0}^{m} \frac{\partial e}{\partial b_i} \frac{\partial b_i}{\partial c_k} + \sum_{j=0}^{n} \frac{\partial e}{\partial a_j} \frac{\partial a_j}{\partial c_k} = \sum_{i=0}^{m} J_{b_i} Y_{b}(i,k) + \sum_{j=0}^{n} J_{a_j} Y_{a}(j,k)
\]

with \( Y_{b} = 0.5X(a)^{-1} \) and \( Y_{a} = 0.5X(b)^{-1} \), where

\[
X(b) = \begin{bmatrix}
    b_0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 \\
    b_2 & -b_1 & b_0 & \cdots & 0 & \cdots & 0 & 0 \\
    b_4 & -b_3 & b_2 & \cdots & (-1)^j \cdot b_{4-j} & \cdots & 0 & 0 \\
    \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\
    b_m & -b_{m-1} & b_{m-2} & \cdots & (-1)^j \cdot b_{m-j} & \cdots & \cdots & \cdots \\
    \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\
    0 & 0 & 0 & \cdots & 0 & \cdots & (-1)^{n-1} \cdot b_{m-1} & (-1)^n \cdot b_{m-2} \\
    0 & 0 & 0 & \cdots & 0 & \cdots & 0 & (-1)^n \cdot b_m
\end{bmatrix}
\]

\( X(b) \in \mathbb{R}^{p \times n} \). So starting from the original algorithm parameterized in \( b_0, \ldots, b_m, a_0, \ldots, a_n \), only the calculation of matrix \( X \) has to be added to change the parameterization. Whenever a zero of \( c_p s^2 + \cdots + c_2 s^2 + c_0 = 0 \) lies on the imaginary axis, the \( C \) can be separated in a group of fixed \( C^F \) and variable \( C^V \) parameters. This method is similar as described in the section of fixing the poles to improve stability (Chapter 3 Section...
Reparameterizing

3.4). This separation can be written as the following matrix product (example for 2 fixed zeros $c^F_2 s^2 + c^F_1 s + c^F_0$):

$$
\begin{bmatrix}
  c_0 \\
  c_2 \\
  c_4 \\
  c_6 \\
  \vdots \\
  c_{p-2} \\
  c_p
\end{bmatrix}
= 
\begin{bmatrix}
  c^F_0 & 0 & 0 & \ldots & 0 & 0 \\
  0 & c^F_1 & 0 & \ldots & 0 & 0 \\
  0 & c^F_2 & c^F_1 & 0 & \ldots & 0 \\
  \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
  0 & 0 & 0 & \ldots & c^F_{p-1} & c^F_p \\
  0 & 0 & 0 & \ldots & 0 & c^F_p
\end{bmatrix}
\begin{bmatrix}
  c^V \\
  c^V \\
  c^V \\
  c^V \\
  c^V \\
  c^V \\
  c^V
\end{bmatrix}
$$

(4-57)

Whenever a real zero gets fixed, $Y_a$ and $Y_b$ in (4-55) should be replaced by respectively $Y_a C^F$ and $Y_b C^F$. The problem with this technique is that in every iteration step the inverse of the new $X$ has to be calculated. It has been observed that $X$ can become badly conditioned for a relative low order of the model (<10). For example, this occurs when a purely real zero of $G(s, \theta) + G^T(-s, \theta)$ tends to zero.

To overcome this problem one can use orthogonal polynomials like Forsythe polynomials. However, some adaptations have to be carried out to create a similar framework. Using these polynomials it is no longer possible to have an analytic relation between the coefficients of $G(s, \theta) + G^T(-s, \theta)$ and the coefficients of the numerator and denominator of $G(s, \theta)$ like (4-48). From the frequency vector and the weighting vector, the orthogonal vectors for the numerator and denominator coefficients are extracted during the starting values step with Iterative Weighted linear Least Squares (IWLS) [49 p239]. A three terms recursion formula can be used to perform the orthogonalization step, since we work with real polynomials that are evaluated on the imaginary axis [49 p426]. In the rest of the algorithm, namely the refinement step with the NLS-FRF, the same orthogonal basis is used. This choice leads to well, but not best, conditioned normal equations. Once one has the coefficients $b_{f0}, \ldots, b_{fm}, a_{f0}, \ldots, a_{fn}$ of the orthogonal polynomials of respectively the numerator and denominator, the coefficients of the real part:
Passive models

are estimated by evaluating the real part of the transfer function on a dense grid. This is done with a simple linear least squares step. The corresponding Jacobian (4-53), (4-54) and (4-55) become respectively:

\[
G(q(s), \theta_f) + G(q(s), \theta_f) = \sum_{k=0}^{p} c_{fk} q_{ck}(s) \\
A(q_a(s), a_f) A(q_a(s), a_f)
\]  

(4-58)

Next, the partial differentials are calculated from the following matrix relation, since there is no longer a simple analytic relation between the different coefficients:

\[
J_{a_{fi}} = \frac{\partial e}{\partial a_{fi}} = \left( \frac{B(q_b(s), b_{fi})}{A(q_{aj}(s), a_{fi})^2} \right) (-q_{aj}(s))
\]  

(4-59)

\[
J_{b_{fi}} = \frac{\partial e}{\partial b_{fi}} = \left( \frac{1}{A(q_{aj}(s), a_{fi})} \right) (q_{bi}(s))
\]  

(4-60)

\[
J_{c_{fk}} = \frac{\partial e}{\partial c_{fk}} = \sum_{i=0}^{m} \frac{\partial e}{\partial b_{fi}} \frac{\partial b_{fi}}{\partial c_{fk}} + \sum_{j=0}^{n} \frac{\partial e}{\partial a_{fj}} \frac{\partial a_{fj}}{\partial c_{fk}}
\]  

(4-61)

Next, the partial differentials \( \frac{\partial b_{fi}}{\partial c_{fk}}, \frac{\partial a_{fi}}{\partial c_{fk}} \) are calculated from the following matrix relation \((k = 0, ..., p)\), since there is no longer a simple analytic relation between the different coefficients:

\[
[B(q_b(s), b_f)(A(q_a(s), a_f)) + B(q_b(s), b_f)A(q_a(s), a_f)] = \sum_{k=0}^{p} c_{fk} q_{ck}(s)
\]  

(4-62)

This results in:

\[
\begin{bmatrix}
B(q_b(s), b_f)q_{aj}(s) + B(q_b(s), b_f)q_{aj}(s) \\
A(q_a(s), a_f)b_{fi}(s) + A(q_a(s), a_f)b_{fi}(s)
\end{bmatrix}
\begin{bmatrix}
\frac{\partial a_{fi}}{\partial c_{fk}} \\
\frac{\partial b_{fi}}{\partial c_{fk}} \\
\frac{\partial c_{fk}}{\partial c_{fk}}
\end{bmatrix}
= q_{ck}(s)
\]  

(4-63)
For every iteration step the new $b_f$ can be found from the new $c_{f0}, \ldots, c_{fp}, a_{f0}, \ldots, a_{fn}$ with the following relation, where the chosen grid points for $s$ are sufficiently dense to cover all the dynamics of the model:

$$
\sum_{i=0}^{m} b_{fi}(q_{bi}(s)A(q_{a}(s), a_f) + q_{bi}(s)A(q_{a}(s), a_f)) = \sum_{k=0}^{p} c_{fk}q_{ck}(s) \quad (4-64)
$$

Hence, the price to be paid to improve the numerical stability is a longer calculation time.

To separate of the parameters $c_f$ in a free and variable group, one needs to know the values of the spectral zeros. This can be calculated using the following relation, where the chosen grid points for $s$ are sufficiently dense to cover all the dynamics of the model:

$$
\sum_{k=0}^{p} c_{fk}q_{ck}(s) = K \prod_{k=1}^{p} (s-s_k) \quad (4-65)
$$

(4-64) is not as simple as it seems, since no longer a simple three terms recursion can be used but a full recursion should be used. To calculate (4-63) in a numerically stable way, the bases for $q_a(s)$ and $q_b(s)$ should be changed to be also orthogonal with $B(q_b(s), b_f)q_a(s) + B(q_a(s), b_f)q_a(s)$ and respectively with $q_b(s)A(q_a(s), a_f) + q_a(s)A(q_a(s), a_f)$. However, it is not possible to find a base that is orthogonal for the 2 conditions, neither for the $a_f$ coefficients as for the $b_f$ coefficients. To overcome this problem, we have chosen to adapt only the $c_f$ coefficients with the $b_f$ coefficients and keep the $a_f$ coefficients constant in this calculation. So $\frac{\partial d_{fi}}{\partial c_{fk}}$ becomes zero in (4-61) and (4-63). This allows to change the base of $b_f$ for every iteration step of the refinement algorithm. The calculation of the base of $b_f$ is explained in the Appendix 4.10.8.

What about improper $(m > n)$ models? In the case where the orthogonal polynomials are used, nothing changes, and all the formulas remain the same. However, in the simple case the $X(a)$ is no longer a square matrix, we have less equations than unknowns. So to solve this, a similar equation as (4-64) can be used. In [44] another technique is described to derive the $b_{0}, \ldots, b_m$
from the \( c_0, \ldots, c_p, a_0, \ldots, a_n \); \( b_m \) has to be chosen as an arbitrary positive real number, and the other values \( b_0, \ldots, b_{m-1} \) are found via the following relation:

\[
\begin{bmatrix} b_0 & \ldots & b_{m-1} \end{bmatrix}^T = Y(a)(c - w b_m) \tag{4-66}
\]

where \( w \in \mathbb{R}^m \) is a column matrix of the form:

\[
w = \begin{bmatrix} 0 & \ldots & a_0 & \ldots & a_m \end{bmatrix}^T \quad \text{if } m \text{ even} \tag{4-67}
\]

\[
w = \begin{bmatrix} 0 & \ldots & -a_1 & \ldots & -a_m \end{bmatrix}^T \quad \text{if } m \text{ odd} \tag{4-68}
\]

One could think to parameterize \( G \) in the parameters of the real part and imaginary part of the numerator of \( G(j\omega, 0) \) separately. However, there are some problems that can occur that make this parameterization useless. The SISO transfer function can be rewritten in:

\[
G(j\omega, 0) = \frac{B(j\omega, b_i)}{A(j\omega, a_j)} = \frac{C(\omega^2, c_k) + j\omega D(\omega^2, d_i)}{A(j\omega, a_j)(A(-j\omega, a_j))} \tag{4-69}
\]

with respectively as real and imaginary part:

\[
\text{Re}(G(j\omega, 0)) = \frac{C(\omega^2, c_k)}{A(j\omega, a_j)(A(-j\omega, a_j))} \tag{4-70}
\]

\[
\text{Im}(G(j\omega, 0)) = \frac{\omega D(\omega^2, d_i)}{A(j\omega, a_j)(A(-j\omega, a_j))} \tag{4-71}
\]

Now if the same algorithm is used on the parameters \( c_k, d_i, a_j \), then one can find improved values for these parameters. However, from these new values one cannot extract the corresponding new \( b_i \) from the relation (4-69). This is because there is no constraint that restricts that the sum of \( C(\omega^2, c_k) \) and \( j\omega D(\omega^2, d_i) \) must have the polynomial \( A(-j\omega, a_j) \) in common with the denominator. So this will not allow one to extract the new \( b_i \) from relation (4-69).
4.7 Minimax models

The minimax constraint can also be added to the passivity constraint in the same way as was done for the stability constraint. In contrast to the stability constraint this adapting weighting procedure has less chance to change the fit to a minimax passive model, since the passivity constraint is much stricter than the stability constraint. By changing the weight, some areas become more important than other areas, this means that the positions of poles and zeros will be change a bit by the refinement algorithm to improve the fit, but the eigenvalues of $G(j\omega, \theta) + G^T(-j\omega, \theta)$ must remain bigger than zero at the same time. It is observed that this constraint makes it mostly impossible to make a passive minimax model.

4.8 Increasing the order

The same idea as for stable models can also be used to increase the order of a passive model to find a better passive approximation. The only difference with the stability case is that the order can only be increased by adding two passive models. This a well known property of PRF’s. The multiplication of two PRF’s does not guarantee that the product remains a PRF. For example, the product of 2 improper passive models leads to a model with a relative order difference of two, which is non passive.

One has also to be careful when adding two PRF’s. Indeed, if one of the models is very close to a user-defined constraint border, then the resulting sum could violate this user-defined constraint border. For example, lets consider the following 2 passive models $G_1(s)$ and $G_2(s)$, and the resulting sum $G_{tot}(s)$. Now lets add a user-defined stability border, where the absolute value of the real part of the zeros of $G(s) + G(-s)$ is at least 0.1 (This restriction is of interest whenever we want to separate the spectral zeros in a fixed and ‘variable’ group as explained in Section 4.6).

\[
G_1(s) = \frac{s^2 + 1.232s + 0.416}{s^2 + 0.369s + 0.526}, \quad G_2(s) = \frac{s^2 + 1.763s + 0.968}{s^2 + 0.024s + 0.87}
\]  

(4-72)
The zeros of $G(s)$ are given in Table 4-5. One sees from Table 4-5 that the summation violates the user-defined constraint border. The astute reader remarks that the constraint on the zeros of the real part of the transfer function is violated by the poles of $G_2(s)$ (see Table 4-6).

The relation between the constraints on the poles of $G(s)$ and the zeros of $G(s) + G(-s)$ should be investigated, to give the user some more guidelines.

### 4.9 Passivity enforcement software

#### 4.9.1 Vector fitting with passivity enforcement

This software package is implemented in Matlab and requires the Optimization toolbox of Matlab. The passivation algorithm is described in [29]. This technique needs a stable model which is created by the Vector Fitting (VF) toolbox. The passivity is enforced by adapting only the residue matrix and keeping the stable poles fixed. This makes the algorithm simple, but this is also a disadvantage since no better positions for the poles can be found. The major disadvantage of this method is that the passivity check is at a discrete frequency grid. This
means that the user must check the passivity property with another analytic method to be sure that the model is truly passive. It gives the user false hope by adding a tolerance parameter and extra frequency bands where the passivity should also be checked. Playing with these 2 extra aids does not guarantee that a passive model will be found. During the preprocessing step the linear term \( h \) (in (3-39)) is set to zero, if this one is negative [29].

The passive estimate is illustrated by comparing the eigenvalues before and after the enforcement. The clipping of the linear term has no effect on the real part of the fit, so the eigenvalues can still be very close to the original model. However, the imaginary part can be quite useless, so the user should be careful and not only look at the real part of the model. Also scattering data cannot be handled by this method, only the positive real constraint is handled.

### 4.9.2 IdEM toolbox

This toolbox contains the passivity enforcement algorithm of [26]. The IdEM toolbox allows the user to create passive models from time or frequency domain data. It consists of a stable identification step followed by a passive compensation step. Here the user must give the software a frequency range where the possible poles should be located, and the order of the model. The first step is a vector fitting technique. Here an extra constraint can be added to enforce asymptotic passivity (positive real part at infinity). The toolbox has been used on several examples and gives in general nice results. The best results were found if the model was automatically passive after the first step. Often it was better to change the model order to find a nice passive model in the first step, then fix the order and do the compensation step. The compensation step is applied to the state-space representation of the model. The disadvantage of the compensation step is that it only changes the zeros of the model by adapting the \( C \) matrix, while all the other parameters \( A \), \( B \) and \( D \) remain the same. This is not the case for the method presented in this thesis where all the parameters are allowed to vary. Another disadvantage is that no improper models can be modeled. The advantage is that it uses a state space representation, which can handle SISO and MIMO systems without modifications of the algorithm. Another advantage is that this method does not need passive initial values, but only a stable model with a positive real \( D \).
4.10 Appendix

4.10.1 Constraints on the spectral zeros

Applying (4-7) on \( G(\Omega) = \frac{B(\Omega)}{A(\Omega)} \) results in:

\[
\frac{C(\Omega)}{A(\Omega)A(\Omega^*)} = \frac{A(\Omega)B(\Omega^*) + B(\Omega)A(\Omega^*)}{A(\Omega)A(\Omega^*)}
\]  \hspace{1cm} (4-73)

Now lets consider that we have a CT system and that \( C(\omega) \) (\( \Omega = j\omega \)) is of order \( n \) in the variable \( \omega^2 \), then one can write:

\[
C(\Omega) = K(\omega^2 + z_1^2)(\omega^2 + z_2^2)\ldots(\omega^2 + z_n^2)
\]  \hspace{1cm} (4-74)

with \( K > 0 \). The \( \omega^2 \)-roots, which are denoted by \( z_1^2, z_2^2, \ldots, z_n^2 \), may be complex as well as real, but since (4-74) has real coefficients, any complex roots, if present, must occur in conjugate pairs. It is clear that (4-74) becomes surely negative over some part of the range \( 0 \leq \omega^2 < \infty \), if there are \( \omega^2 \)-roots with negative real part with odd multiplicity.

4.10.2 Eigenvalues of Hamiltonian

The zeros of a transfer function matrix are defined [35] as the zeros of the determinant of the numerator matrix. Here the analogy between the eigenvalues of the Hamiltonian of the transfer function matrix \( G(s) \) (4-20) and the zeros of \( \det(G(s) + G^T(-s)) \) will be shown. If \( G(s) \) is written in it state-space representation \( G(s, \theta) = D + C(sI - A)^{-1}B \), then the zeros of \( G(s) \) can be computed as the eigenvalues of the following matrix [35]:

\[
A - BD^{-1}C
\]  \hspace{1cm} (4-75)

for \( D \) regular. The state space representation of \( G(s) + G^T(-s) \) is \( Q + C_m(sI - A_m)^{-1}B_m \) with:

\[
A_m = \begin{bmatrix} A & 0 \\ 0 & -A^T \end{bmatrix}, \quad B_m = \begin{bmatrix} B \\ -C^T \end{bmatrix}, \quad C_m = \begin{bmatrix} C^T \\ B \end{bmatrix}^T \quad \text{and} \quad Q = D + D^T
\]  \hspace{1cm} (4-76)
Applying (4-76) to (4-75), gives (4-16).

\[
\begin{bmatrix}
A - BQ^{-1}C & -BQ^{-1}B^T \\
C^TQ^{-1}C & -A^T + C^TQ^{-1}B^T
\end{bmatrix}
\] (4-77)

4.10.3 Relative degree of one

In [2] it is shown that the spectral decomposition of \( \det(G_c(s, \theta) + G_c^T(-s, \theta)) = 0 \) is still valid when \( \det(G_c(\infty, \theta)) = 0 \). The case where \( G_c(\infty, \theta) \) is non-finite can be rewritten according to [2] as

\[
G_c(s, \theta) = Ls + G_{c1}(s, \theta_1)
\] (4-78)

with \( G_{c1}(\infty, \theta_1) \) finite and \( L \) non-negative definite. It is easy to see that the linear term does not have an influence on \( \det(G_c(s, \theta) + G_c^T(-s, \theta)) = 0 \), since \( Ls \) only influences the imaginary part of \( G_c(j\omega, \theta) \).

4.10.4 Illustrations of discrete-time passivity constraints

To illustrate the passivity constraints on zeros of \( \det(G(z) + G^T(z^{-1})) \) for discrete-time systems, four simulation examples are given, two SISO and two MIMO examples. The SISO examples consist of a second order and sixth order SISO transfer function \( G(z) \) for three different positions of the zeros. As shown in Figure 4-1 and Figure 4-2 it is easy to see that the zeros of \( \det(G(z) + G^T(z^{-1})) \) become a member of the unit circle as soon as there is a point of \( \text{Re}(G(e^{j\theta})) \) that becomes equal to zero. This is still allowed for passive systems. If \( \text{Re}(G(e^{j\theta})) \) becomes smaller than zero, then the zeros of \( G(z) + G(z^{-1}) \) on the unit circle with even multiplicity diverges from each other.

The MIMO examples consist of a symmetric \( 2 \times 2 \) and \( 3 \times 3 \) second order transfer matrix \( H(z) \) for respectively three and two different values of the gain of \( H_{13}(z) \) and \( H_{23}(z) \). Similarly to the SISO examples one can see in Figure 4-3 and Figure 4-4 the eigenvalues \( E \) of \( \text{Re}(H(e^{j\theta})) \) and the corresponding zeros of \( \det(H(z) + H^T(z^{-1})) \). It is easy to see that the
zeros corresponding to the largest eigenvalue do not change a lot by changing the values of the gain of $H_{13}(z)$ and $H_{23}(z)$, while the zeros corresponding to the smallest eigenvalue move towards the unit circle and diverge once they are on the unit circle.

**FIGURE 4-1** The real part of a transfer function and the corresponding zero pole plot of a second order system. Three transfer functions with $\min(\text{Re}(G(e^{j\theta}))) > 0$ (green), $\min(\text{Re}(G(e^{j\theta}))) = 0$ (blue) and $\min(\text{Re}(G(e^{j\theta}))) < 0$ (red).

**FIGURE 4-2** The real part of a transfer function and the corresponding zero pole plot of a sixth order system. Three transfer functions with $\min(\text{Re}(G(e^{j\theta}))) > 0$ (green), $\min(\text{Re}(G(e^{j\theta}))) = 0$ (blue) and $\min(\text{Re}(G(e^{j\theta}))) < 0$ (red).
FIGURE 4-3 The eigenvalues of the real part of a transfer matrix and the corresponding zero pole plot of a $2 \times 2$ second order system. The real part of the eigenvalues with \( \min(\text{Re}(E(e^{j\theta}))) > 0 \) (green), \( \min(\text{Re}(E(e^{j\theta}))) = 0 \) (blue) and \( \min(\text{Re}(E(e^{j\theta}))) < 0 \) (red).

FIGURE 4-4 The eigenvalues of the real part of a transfer matrix and the corresponding zero pole plot of a $3 \times 3$ second order system. Spectral zeros on the unit circle: with and without even multiplicity (green) and only without even multiplicity (blue).
4.10.5 The order of a passive discrete-time model

\[ G(z) = \frac{b_m z^m + \ldots + b_1 z + b_0}{a_n z^n + \ldots + a_1 z + a_0} \]  

(4-79)

if \( m = n + 1 \)

\[ G(z^{-1}) = \frac{b_m z^{-m} + \ldots + b_1 z^{-m+1} + b_0 z^m}{a_n z^{-n} + \ldots + a_1 z^{-n} + a_0 z^{n+1}} = \frac{b_m z^{-m} + b_1 z^{-m+1} + b_0 z^m}{z(a_n z^{-n} + a_1 z^{-n} + a_0 z^{n})} \]  

(4-80)

It is clear that the poles of \( G(z) + G(z^{-1}) \) can never be quadrant symmetric for \( m = n + 1 \), since there is a simple pole at \( z = 0 \).

For \( z = re^{j\theta} \), with \( r \) very large, (4-79) can be simplified to

\[ \frac{b_m z^{-m-n}}{a_0} = \frac{b_m e^{j\theta(m-n)}}{a_0} \]  

(4-81)

\[ G(z) + G(z^{-1}) \] becomes

\[ \frac{b_m e^{j\theta(m-n)}}{r a_0} + \frac{b_m e^{-j\theta(m-n)}}{r a_0} = \frac{2b_m}{a_0} \cos \theta(m-n) \]  

(4-82)

which becomes negative if \( m \neq n \) for a certain value of \( \theta \). So the order of the numerator and denominator of a discrete-time model needs to be the same, if the model has to be passive.

4.10.6 Residue of a pool on the unit circle

See reference [30]; in [30] condition (4-29) is given without proof. In this appendix the proof of (4-29) is given. Consider \( H(z) = G(z) + G^T(z^{-1}) \geq 0 \) in the neighborhood of \( z_0 = e^{j\theta_0} \) on a circle with centre \( z_0 \) and radius \( r \) (see Figure 4-5), then the contribution of \( z_0 \) can be written as:

\[ H(z) = \frac{A}{z-e^{j\theta_0}} + \frac{A^H}{z^{-1}-e^{-j\theta_0}} \]  

(4-83)
Along the small circle \( \psi \) varies as with \( \psi = \theta + \theta_0 \), \( \theta = -\frac{\pi}{2} \to 0 \to \frac{\pi}{2} \). Defining \( A_1 = Ae^{-j\theta_0} \) equation (4-85) can be written as

\[
H(z) = \frac{1}{r}(Ae^{-j\theta} + A^H e^{j\theta})
\]

(4-86)

\[
H(z) = \frac{1}{r}((A_1 + A_1^H)\cos \theta + j(A_1^H - A_1)\sin \theta)
\]

(4-87)

Since according to the passivity definition \( H(z) \geq 0 \) for \( \theta = -\frac{\pi}{2} \to 0 \to \frac{\pi}{2} \), the following conditions must be satisfied:

\[
\begin{align*}
A_1^H - A_1 &= 0 \\
A_1 + A_1^H &\geq 0
\end{align*}
\Rightarrow \begin{cases} 
A_1 = A_1^H \\
A e^{-j\theta_0} \geq 0 
\end{cases}
\]

(4-88)

**FIGURE 4-5** Calculation of the residue of the pole \( z_0 \).
which proves the residue condition.

4.10.6.1 Discretization through bilinear transformation

The residue of the simple pole \( s_0 = j\omega_0 \) of the continuous-time model \( G_c(s) \) is calculated as

\[
R_c = \lim_{s \to s_0} (s - s_0)G_c(s).
\]

After discretization through the bilinear transformation (4-36) one finds

\[
R_c = \lim_{z \to z_0} \frac{\left( \frac{2z - 1}{Tsz + 1} - \frac{2z_0 - 1}{Tsz_0 + 1} \right)}{z - z_0}R_d
\]  
(4-89)

with \( R_d = \lim_{z \to z_0} G_d(z)(z - z_0) \), \( T_s \) the sampling period and \( z_0 = \frac{1 + \frac{T_s j\omega_0}{2}}{1 - \frac{T_s j\omega_0}{2}} = e^{j\theta_0} \). Applying l’Hôpital’s rule, one finds

\[
R_c = R_d \frac{2}{T_s(z_0 + 1)^2}
\]  
(4-90)

\[
R_c = R_d \frac{2}{T_s} e^{j\theta_0} \left( e^{\frac{j\theta_0}{2}} + e^{-\frac{j\theta_0}{2}} \right)
\]  
(4-91)

\[
R_c = R_d \frac{4}{T_s} \frac{1}{4e^{j\theta_0}\cos^2\left(\frac{\theta_0}{2}\right)}
\]  
(4-92)

\[
e^{-j\theta_0}R_d = T_s\cos\left(\frac{\theta_0}{2}\right)R_c
\]  
(4-93)

(4-93) shows that \( e^{-j\theta_0}R_d \) is non-negative definite for every value of \( \theta_0 \), which completes the proof of [34].
4.10.6.2 Discretization through impulse invariant transformation

The residue of the simple pole \( s_0 = j\omega_0 \) of the continuous-time model \( G_c(s) \) is calculated as

\[
R_c = \lim_{{s \to s_0}} (s - s_0)G_c(s) \]

After discretization through the impulse invariant transformation (4-37) one finds

\[
R_c = \lim_{{z \to z_0}} \frac{1}{T_s} \frac{\log z - \frac{1}{T_s} \log z_0}{z - z_0} R_d
\]

(4-94)

with \( R_d = \lim_{{z \to z_0}} G_d(z)(z - z_0) \), \( T_s \) the sampling period and \( z_0 = e^{j\omega_0 T_s} = e^{j\theta_0} \). Applying l'Hôpital's rule, one finds

\[
R_c = \frac{1}{T_s} \frac{1}{z_0 - z} \Leftrightarrow e^{-j\theta_0} R_d = T_s R_c
\]

(4-95)

(4-95) shows that \( e^{-j\theta_0} R_d \) is non-negative definite, which completes the proof of [34].

4.10.6.3 Discretization through First-Order-Hold

When using the FOH as discretization technique, one must make the difference between the causal

\[
G_{FOH}(z) = \frac{(z - 1)^2}{T_s z^2} \mathcal{Z} \left\{ \frac{1}{s^2} G_c(s) \right\}
\]

(4-96)

and the non causal case

\[
G_{FOH}(z) = \frac{(z - 1)^2}{T_s z^2} \mathcal{Z} \left\{ \frac{1}{s^2} G_c(s) \right\}
\]

(4-97)

where \( \mathcal{Z}\{G_c(s)/s^2\} = \sum_{i=1}^{n} \text{Res} \left\{ \frac{G_c(s)}{s^2} \frac{z}{z - z_0} \right\} \) with \( \mathcal{Z} \) the z-transform, \( T_s \) the sampling period and \( n \) the number of poles of \( G_c(s)/s^2 \).
The residue of the simple pole \( s_0 = j\omega_0 \) of the continuous-time model \( G_c(s) \) is calculated as 
\[
R_c = \lim_{s \to s_0} (s - s_0)G_c(s).
\]
After discretization through the causal FOH transformation one finds
\[
R_d = \lim_{z \to z_0} \frac{(z - 1)^2R_c}{T_s z^2} \frac{z}{s_0^2 z - z_0}(z - z_0) \tag{4-98}
\]
with \( R_d = \lim_{z \to z_0} G_d(z)(z - z_0) \), \( T_s \) the sampling period and \( z_0 = e^{j\omega_0 T_s} = e^{j\theta_0} \).
Simplifying (4-98) gives
\[
R_d = \frac{(z_0 - 1)^2}{T_s z_0^2} R_c \tag{4-99}
\]
\[
R_d = \frac{4\sin^2 \left(\frac{\theta_0}{2}\right)}{T_s z_0^2} R_c \tag{4-100}
\]
(4-100) shows that \( e^{-j\theta_0}R_d \) is not non-negative definite for \( \frac{\pi}{2} < \theta_0 < \frac{3\pi}{2} \). Similarly one can find the following condition for the non-causal FOH discretization.
\[
e^{-j\theta_0}R_d = \frac{4\sin^2 \left(\frac{\theta_0}{2}\right)}{T_s z_0^2} R_c \tag{4-101}
\]
where \( e^{-j\theta_0}R_d \) is non-negative definite for all \( \theta_0 \), which completes the proof of [31].

4.10.7 ZOH destroys the passivity property

As shown in Section 4.3.1 \( G_{1f}(s) \) is PR if \( a_{0h} b_{0l} > 0 \). Now if one calculates the ZOH approximation \( G_{ZOH}(z) \) of \( G_{1f}(s) \), one finds the following results.
\[
G_{1f}(s) = \frac{b_{0l}}{s + a_{0l}} \tag{4-102}
\]
\[ G_{ZOH}(z) = \left( \frac{z-1}{z} \right) Z\left\{ \frac{G_{1f}(s)}{s} \right\} \]  

(4-103)

with \( Z \) the z-transform and \( T_s \) the sampling period.

\[ G_{ZOH}(z) = \left( \frac{z-1}{z} \right) \left[ \frac{b_{0l}}{a_{0l}z - e^{-a_{0l}T_s}} - \frac{b_{0l}}{a_{0l}z - e^{-a_{0l}T_s}} \right] \]  

(4-104)

\[ G_{ZOH}(z) = \frac{b_{0l}}{a_{0l}} \left[ 1 - \frac{z-1}{z - e^{-a_{0l}T_s}} \right] \]  

(4-105)

\[ G_{ZOH}(z) = \frac{b_{0l}}{a_{0l}} \left[ 1 - e^{-a_{0l}T_s} \right] \]  

(4-106)

When evaluated on the unit circle, the real part of \( G_{ZOH}(z) \) can be represented as

\[ \text{Re}(G_{ZOH}(z)) = \frac{b_{0l}(1 - e^{-a_{0l}T_s})(\cos \theta - e^{-a_{0l}T_s})}{a_{0l} \left[ 1 - 2 \cos \theta e^{-a_{0l}T_s} + e^{-2a_{0l}T_s} \right]} \]  

(4-107)

A simple manipulation shows that for \( \text{Re}(G_{ZOH}(z)) \) to be non-negative for all choices of \( \theta \), the following condition must be met:

\[ e^{-a_{0l}T_s} \geq 1 \]  

(4-108)

which is only possible for \( T_s = 0 \) or \( a_{0l} = 0 \). It can be concluded that (4-106) is not passive for non-zero \( T_s \) and non-zero \( a_{0l} \).

4.10.8 Calculation of the orthogonal base

The Gram-Schmidt orthogonalization calculates an orthonormal set \( \{q_1, q_2, \ldots, q_p\} \) from a given linear independent set \( \{x_1, x_2, \ldots, x_p\} \) with the property

\[ \text{span}\{q_1, q_2, \ldots, q_k\} = \text{span}\{x_1, x_2, \ldots, x_k\} \text{ for } k = 1, 2, \ldots, p \]  

(4-109)
The denominator polynomial $A$ is fixed in all the calculations. It works as follows. In the first step we assign $z_1 = x_1$ and calculate $q_1 = z_1/\|z_1\|$ with. In the second step we choose an element $z_2 \in \text{span}\{x_1, x_2\}$ that is orthogonal to $q_1: z_2 = x_2 + \alpha_{21} q_1$ and $\langle z_2, q_1 \rangle = 0$.

We find $\alpha_{21} = -\langle x_2, q_1 \rangle$ and calculate $q_2 = z_2/\|z_2\|$. In the $s$th step we take an element $z_s \in \text{span}\{x_1, x_2, \ldots, x_s\}$ that is orthogonal to $q_1, q_2, \ldots, q_{s-1}: z_s = x_s + \sum_{r=1}^{s-1} \alpha_{sr} q_r$ and $\langle q_s, q_r \rangle = 0, \ r = 1, 2, \ldots, k-1$. We find $\alpha_{sr} = -\langle x_s, q_r \rangle$ with $r = 1, 2, \ldots, k-1$, and calculate $q_s = z_s/\|z_s\|$. 

\[
\langle z_1(x), z_2(x) \rangle = \text{Re} \left( \sum_{k=1}^{p} (z_1(x_k)\bar{A} + \bar{z}_1(x_k)A)(z_2(x_k)\bar{A} + \bar{z}_2(x_k)A) \right) 
\]

(4-110)

\[
\sqrt{\langle z, z \rangle} = \|z\| 
\]

(4-111)
5.1 Introduction

In this chapter the system identification of guaranteed stable and passive models from noisy data is illustrated on several examples. Although the full two step approach is carried out, the focus will lie on the second step of the two step approach and the different techniques of Chapter 3 and Chapter 4 will be demonstrated with four single input single output and two multiple input multiple output examples. If possible, comparisons with other toolboxes will be made. The available toolboxes are Vectfit2 and the passivity enforcement extension, containing the VF method presented in [28, 29]; IdEM presented in [26]; and RARL2 presented in [43]. We would like to make a small remark concerning the use of RARL2 in this chapter. RARL2 was only implemented to handle scattering data and since the Mobius transformation [3] \( Z = (I + S)(I - S)^{-1} \) with \( Z \) the impedance matrix and \( S \) the normalized scattering matrix) does not preserve the stability property, it will not be possible to use it on impedance or admittance examples.
Normally RARL2 should be used in combination with Hyperion [32], this was not possible due to technical problems (the software was not available), so it uses the unconstrained model to create a rational transfer matrix. Since the minimum of the solution lies outside the left half plane (LHP), we had to constrain some tolerances to create a stable approximation. This is of course not the case if Hyperion would have been used for the first step.

5.2 Crankcase

This measurement example will be used to show the different advantages and disadvantages of the techniques presented in the previous chapters.

We wish to thank Prof. Johan Schoukens and Prof. Rik Pintelon in the Department of ELEC, Free University Brussels, Belgium for providing the measurement data. The experiment consists of a modal analysis test on a crankcase [61]. A crankcase is excited with a multisine composed of 308 sinusoids in the frequency band 99.609 Hz-399.414 Hz for thirteen independent experiments; the frequency spacing $\Delta f$ between consecutive sinusoids in the frequency spectrum is 0.977Hz. Before the estimation can begin, the data of the different

![Amplitude vs Frequency Graph](image)

**FIGURE 5-1** Unconstrained approximation of a measured crankcase: FRF measurements (black dots), unconstrained estimated transfer function (bold green line), residuals between the unconstrained model and the measurements (green dashed-dotted line), standard deviation on the measurements (red pluses), standard deviation on the unconstrained model (blue pluses).
experiments have to be synchronized and averaged. Then the transfer function is estimated with the frequency domain system identification toolbox \[36\]. A 6th order model with 4 stable poles and 2 unstable poles (see TABLE 5-1) explains the measurements very well. Indeed, from Figure 5-1 it can be seen that the difference between the unconstrained model and the measurements is of the same order of magnitude as the standard deviation of the measurements.

Starting from this unconstrained model a stable approximant will be created. To do so a set of stable starting values (initial 1-4) are generated from the unconstrained model, as explained in Chapter 3 Section 3.3.3. On Figure 5-2 one can see the four different initial stable approximants and their difference with the unconstrained model. It is clear that (initial 2-4) gives acceptable results, while initial 1 does not succeed in capturing all the dynamics of the model, because it has a lower model order. The smallest least squares error corresponds to

<table>
<thead>
<tr>
<th>[order]</th>
<th>Pole 1/2</th>
<th>Pole 3/4</th>
<th>Pole 5/6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original [6/6]</td>
<td>-4.342 ± 1807i</td>
<td>-3.884 ± 2022i</td>
<td>73.81 ± 5269i</td>
</tr>
<tr>
<td>Initial 1 [4/4]</td>
<td>-4.342 ± 1807i</td>
<td>-3.884 ± 2022i</td>
<td></td>
</tr>
<tr>
<td>Initial 2 [6/6]</td>
<td>-4.342 ± 1807i</td>
<td>-3.884 ± 2022i</td>
<td>-73.81 ± 5269i</td>
</tr>
<tr>
<td>Initial 3 [6/6]</td>
<td>-4.342 ± 1807i</td>
<td>-3.884 ± 2022i</td>
<td>-73.81 ± 5269i</td>
</tr>
<tr>
<td>Initial 4 [8/4]</td>
<td>-4.342 ± 1807i</td>
<td>-3.884 ± 2022i</td>
<td></td>
</tr>
<tr>
<td>Iter initial 1</td>
<td>-4.287 ± 1806i</td>
<td>-3.730 ± 2022i</td>
<td></td>
</tr>
<tr>
<td>Iter initial 2</td>
<td>-4.342 ± 1807i</td>
<td>-3.884 ± 2022i</td>
<td>-1.942 ± 5299i</td>
</tr>
<tr>
<td>Iter initial 3</td>
<td>-4.342 ± 1807i</td>
<td>-3.884 ± 2022i</td>
<td>-1.942 ± 5299i</td>
</tr>
<tr>
<td>Iter initial 4</td>
<td>-4.342 ± 1807i</td>
<td>-3.884 ± 2022i</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE 5-1** Position of the poles for the four starting values methods and the iterative version of the four starting values methods.

Further, all these stable initial values are given to the iterative algorithm. To illustrate the possibility of adding a user defined constraint border, instead of the stability border (imaginary axis), we took half of the largest real part of the poles of the stable initial value as constraint border. This corresponds to a maximum allowed real part of -1.942. The poles of initial 1-4 with model order \[$\text{numerator/denominator}$\] are listed in Table 5-1.
The iterative algorithm lowers the cost function for the four different initial values, from a factor 50 till 6700. Especially the phase spectrum inside the desired frequency band is

**FIGURE 5-2** Comparison between the unstable model (bold green line) and the stable approximations. The stable approximant [order] (solid line) and its complex error with the unstable model (dashed line): Initial 1 (red) [4/4], Initial 2 (blue) [6/6], Initial 3 (green) [6/6] and Initial 4 (black) [8/4]

**FIGURE 5-3** Comparison between the unstable model (bold green line) and the iterated stable approximations. The stable approximant [order] after the refinement step (solid line) and its complex error with the unstable model (dashed line): Initial 1 (red) [4/4], Initial 2 (blue) coincides with Initial 3 [6/6], Initial 3 (green) [6/6] and Initial 4 (black) [8/4]

The iterative algorithm lowers the cost function for the four different initial values, from a factor 50 till 6700. Especially the phase spectrum inside the desired frequency band is
improved significantly. One can see on Figure 5-3 that the iterated version of initial 2 and 3 almost coincide, so these two different initial values result in the same local minimum. From Table 5-1 one can see that the stabilized poles converge to a place on the constrained border. Relaxes the constrained border will move these poles even closer to the imaginary axis, but no large decrease in complex error was observed by taking the imaginary axis as constraint border. This corresponds with a drop of 5 percent in cost function for initial 2 and 3. Initial 4 gives the best performance and its complex difference with the unstable model is also lower than uncertainty bound of the unstable model. For this example no large advantages were achieved by separating the poles in a ‘fixed’ and ‘variable’ group. In the figure caption the abbreviation of SPFV will be used to denote the separation of the poles in a ‘fixed’ and ‘variable’ group. The gain in cost function was less than one percent. However the refinement step finds a local minimum for initial 2 and 3 before it reaches its maximum allowed iterations (in the simple case), which translates in a gain of calculation time.

![Graph of Amplitude vs Frequency](image)

**FIGURE 5-4** Comparison between the unstable model (bold green line) and the iterated stable approximations. The stable approximant after the refinement step (solid line) and its complex error with the unstable model (dashed line): Initial 3 without SPFV (red) and Initial 3 with SPFV (black).
5.2.1 Advantage of separating the parameters.

To illustrate the power of separating the poles in a ‘fixed’ and ‘variable’ group as explained in Chapter 3 Section 3.4, the extra constraint border is put on the value -3.923. This is one percent smaller than the real part of the largest pole of the starting values (see Table 5-1) and will clip the real part of this pole for all the stable starting values. To make the figure not to crowded we only show the results for initial 3, which corresponds to the lowest initial error. If one carefully compares Figure 5-2 with Figure 5-4, it is clear that the standard algorithm can not improve its stable starting values. This is due to the fact that the constraint is automatically violated. However, separating the poles in a ‘fixed’ and ‘variable’ group allows the algorithm to improve the fit significantly. Only in a small area close to the fixed complex pole pair, the algorithm can only slightly improve the stable fit, which is of course completely natural.

5.2.2 Minimax constraint

To create minimax versions of the stable models, we started from the stable models illustrated in Figure 5-3 (see Chapter 3 Section 3.5). Since the changes of the amplitude spectrum are not that big, only the complex error with the unstable model is shown here. Initial 2 and 3 are put together since they correspond to the same local minimum. From Figure 5-5 it can easily be seen that the areas were the complex error was large becomes smaller in the minimax version and the places were the complex error was small it becomes larger. The maximum error between the unstable model and the stable minimax approximant drops respectively 2 dB (Initial 1), 4.5 dB (Initial 2 and 3) and 6 dB (Initial 4) with respect to the stable approximants of Figure 5-3.

5.2.3 Increasing the order

Figure 5-6 shows four examples of increasing the order. Since there is no significant difference between the method of multiplication and summation Chapter 3 Section 3.6, we only show here the results for the summation. We start from the iterated stable approximant of initial 3. We did not select initial 4 to illustrate this, which clearly performed best (see Figure 5-3), since
this model is highly improper (order 8/4) it contains a lot of energy outside the useful frequency band which is often not wanted. For the stable model represented by the blue curve we added a $\hat{G}_{\text{extra}}(s, \theta_{\text{extra}})$ of order 1/1. This results in a decrease of the complex error of 0.1 dB. Using an order 2/2 for $\hat{G}_{\text{extra}}(s, \theta_{\text{extra}})$ results in a drop of 3.75 dB, an order 3/3 gives a drop of 18.8 dB, and an order 4/4 corresponds with a drop of 19.4 dB. If we compare these complex errors with the uncertainty bound of Figure 5-1, then one can conclude that all these stable models are good candidates to represent the data. Now if the same order increase of the stable model is performed, but now allowing the poles to be separated in a ‘fixed’ and ‘variable’ group, huge improvements can be found (see Figure 5-7). The same color convention of Figure 5-6 is used for Figure 5-7. The smallest applied order increase of 1/1 for $\hat{G}_{\text{extra}}(s, \theta_{\text{extra}})$ with the constrained poles fixed, performs already better than the 4/4 example without the separation of the parameters in a ‘fixed’ and ‘variable’ group. The improvement of the complex error for the orders 1/1 till 4/4 ranges from 24 dB till 64 dB.
From now on the option of separation of the poles in a ‘fixed’ and ‘variable’ group will be standard activated.

FIGURE 5-6 Comparison between the unstable model (bold green line) and the iterated stable approximations for different order increases. The iterated stable approximant (solid line) from initial 3 and its complex error with the unstable model (dashed line), with an order increase of: 1/1 (blue), 2/2 (green), 3/3 (red) and 4/4 (black)

FIGURE 5-7 Comparison between the unstable model (bold green line) and the iterated stable approximations for different order increases with SPFV. The iterated stable approximant (solid line) from initial 3 with SPFV and its complex error with the unstable model (dashed line), with an order increase of: 1/1 (blue), 2/2 (green), 3/3 (red) and 4/4 (black)

compared to the results of Figure 5-6.
Note that there are two ways to obtain a larger order stable approximant: one can choose to increase the order in one big step, or incrementally by for example small steps of 1. In Figure 5-8 two different choices of order increase are shown. The dashed lines in Figure 5-8 correspond to increments of 1, while the solid lines correspond with increments of 2. One can see from Figure 5-8 that for this measurement example we almost gain nothing by increasing the order by steps of one for the numerator and denominator polynomials, for more than one time. It remains in the same local minimum that has been obtained after one order increase 1/1 (see Figure 5-7). If we compare Figure 5-7 with Figure 5-8, we see that the solution of the first order increase of 2 results in the same minimum as Figure 5-7, while the second increase is much worse than the 4/4 order increase of Figure 5-7. If the same experiment is performed on initial 4, then we see that the incremental steps of 2/2 or 1 big step of the same total order results in the same minima [19]. In general large order increases perform better than small order increases, although the inverse is possible due to local minima. As rule of thumb we suggest an order increase of at least 2 to catch at least one peak of the error vector.
Comparison with freely available toolboxes

Here a comparison is made with some freely available toolboxes. The RARL2 cannot be used here since this is not scattering data. The vector fitting (VF) toolbox can be used. The unconstrained model is given to the VF toolbox and the same number of poles is selected. The resulting model was a $7/6$ stable model with corresponds to the red solid line in Figure 5-9. To have a fair comparison we increased the order of the iterated initial 3 with 1/0 to create also a $7/6$ stable model. This model corresponds to the blue solid line in Figure 5-9. From Figure 5-9 it can be seen that the complex error of our model is 6 dB lower than the VF model. For larger orders this difference between the errors becomes even larger. This improvement follows from the fact that in the VF method only the zeros and the gain are allowed to move freely while the poles are flipped versions of the original poles. If the stable model obtained with VF is used as starting values for the iterative algorithm presented in Chapter 3 Section 3.3.4, then the blue curve of Figure 5-9 is retrieved again.

**FIGURE 5-9** Comparison of different stabilization techniques: The transfer function (solid line) of the stable models and its complex error with the unstable model (dashed line) for: vector fitting (red) and the method presented in this thesis (blue)
For this measurement example no passive model can be generated since the relation between the input and output quantities (force versus acceleration) is not passive. To create a passive model force versus speed should be modeled. This can be created by integrating the input data (dividing the input by \( j\omega \)). However after performing the unconstrained modeling step this automatically corresponded to a passive model, so no second (approximation) step was needed.

### 5.2.5 Outband energy

In [5] an arbitrary good stable model can be estimated from noisy measurements if the constraint on energy outside the user-defined frequency band is unbounded. To demonstrate the amount of energy outband, three models are evaluated over the frequency band from 0 Hz to 2000 Hz. In Figure 5-10 one can see the unconstraint model, the iterated stable approximation from initial 3 and the increased version of this stable approximation with an order increase of 4/4. From Figure 5-10 it can be easily seen that the stable approximations are improved by the refinement step, especially by creating a worse approximation of the complex pole pair around 850 Hz. The increase in error around this pole pair is respectively 20 dB and 40 dB for the stable approximation and the increased stable approximation. It is clear that for simulation

![Comparison between the unstable model (bold green line) and the (increased) stable approximations over a larger frequency band: The transfer function (solid line) of the stable models and its complex error with the unstable model (dashed line) for: iterated version of initial 3 (green) and iterated version of initial 3 with an order increase of 4/4 (black) (Figure 5-10)](image-url)
purposes the higher order approximation is more sensible to noise at frequencies around 850 Hz, than the unconstrained model.

5.2.6 Summary of the results

<table>
<thead>
<tr>
<th>Complex error between the unconstrained model and ...</th>
<th>Cost function [dB]</th>
<th>Maximum error [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>the measurements</td>
<td>-23.17</td>
<td>-4.36</td>
</tr>
<tr>
<td>standard deviation on the measurements</td>
<td>-8.53</td>
<td>12.77</td>
</tr>
<tr>
<td>standard deviation on the unconstrained model</td>
<td>-29.00</td>
<td>-7.85</td>
</tr>
<tr>
<td>Initial 1</td>
<td>8.81</td>
<td>10.07</td>
</tr>
<tr>
<td>Initial 2</td>
<td>-24.98</td>
<td>-5.17</td>
</tr>
<tr>
<td>Initial 3</td>
<td>-37.20</td>
<td>-35.62</td>
</tr>
<tr>
<td>Initial 4</td>
<td>-24.09</td>
<td>-7.18</td>
</tr>
<tr>
<td>Iterated Initial 1</td>
<td>-14.27</td>
<td>-7.25</td>
</tr>
<tr>
<td>Iterated Initial 2</td>
<td>-54.27</td>
<td>-45.95</td>
</tr>
<tr>
<td>Iterated Initial 3</td>
<td>-54.27</td>
<td>-45.95</td>
</tr>
<tr>
<td>Iterated Initial 4</td>
<td>-62.35</td>
<td>-51.72</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/1</td>
<td>-54.32</td>
<td>-45.75</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 2/2</td>
<td>-58.04</td>
<td>-49.10</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 3/3</td>
<td>-73.05</td>
<td>-56.50</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 4/4</td>
<td>-73.66</td>
<td>-65.08</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/1 (SPFV)</td>
<td>-78.36</td>
<td>-68.32</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 2/2 (SPFV)</td>
<td>-121.86</td>
<td>-110.94</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 3/3 (SPFV)</td>
<td>-128.66</td>
<td>-117.42</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 4/4 (SPFV)</td>
<td>-137.78</td>
<td>-125.64</td>
</tr>
<tr>
<td>Iterated Initial 1 with Lawson weighting</td>
<td>-13.4</td>
<td>-9.37</td>
</tr>
<tr>
<td>Iterated Initial 2 with Lawson weighting</td>
<td>-52.83</td>
<td>-50.44</td>
</tr>
<tr>
<td>Iterated Initial 3 with Lawson weighting</td>
<td>-52.83</td>
<td>-50.44</td>
</tr>
<tr>
<td>Iterated Initial 4 with Lawson weighting</td>
<td>-61.08</td>
<td>-57.86</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/0 (SPFV)</td>
<td>-78.32</td>
<td>-68.29</td>
</tr>
<tr>
<td>VF model</td>
<td>-72.39</td>
<td>-62.38</td>
</tr>
</tbody>
</table>

**TABLE 5-2** Summary of the results of the crankcase example
5.3 Transformer

We wish to thank Prof. Patrick Guillaume in the Department of WERK and Prof. Rik Pintelon in the Department of ELEC, Free University Brussels, Belgium for providing the measurement data. The second example consists of the measurement of an admittance parameter of a three-phase transformer in the frequency band 975 Hz till 51 kHz. The admittance $Y_{11}$ was estimated with the Frequency Domain System Identification-toolbox [36] giving a 17th order model with 15 stable poles an 2 unstable poles (see Figure 5-11). These 2 unstable poles are closer the imaginary axis and closer to the border of frequency band of interest compared with the crankcase example. Figure 5-11 shows the unconstrained model with its uncertainty bounds. The residuals between the unconstrained model en the measurements are larger than the standard deviation on the measurements. This can be due to the presence of non-linear distortions. Like for the crankcase example, first a set of stable starting values is generated from this unconstrained model. The four different starting values and their complex error with the unstable model are shown in Figure 5-12. From Figure 5-12 it is easy to see that these

![Figure 5-11](image-url)

**FIGURE 5-11** Unconstrained approximation of a measured $Y_{11}$ parameter: FRF measurements (black dots), unconstrained estimated transfer function (bold green line), residuals between the unconstrained model and the measurements (green dashed-dotted line), standard deviation on the measurements (red pluses), standard deviation on the unconstrained model (blue pluses).
stable initial 2 is most similar to the unstable model, while initial 4 is a terrible stable approximation. Initial 4 performs bad, because the two unstable poles that are transformed in zeros lie at the low frequencies. After the refinement step, one finds the results plotted in Figure 5-13. Like in the crankcase example we find that initial 2 and 3 get stuck in the same local minimum. The iterated version of initial 4 is improved a lot, but is not acceptable as a stable approximant, since its complex error is still bigger than the residuals between the unconstrained model and the measurements (see Figure 5-11). No significant changes with or without separating the poles in a ‘fixed’ and ‘variable’ group were observed for initial 2 and 3, only the iterated version of initial 1 and 4 perform much worse without the separation of the parameters. The obtained stable models from initial 2 and 3 are also passive. This property was not enforced, but happened by fortune. If one compare Figure 5-11 with Figure 5-13 it is easy to see that the bias introduced by stabilizing the unconstrained model is small. Only at the low frequencies we have a complex error that is larger than the residuals and the standard deviations. This is because of the influence of the unstable poles at the low frequencies. This is also illustrated by the stable starting values where the error is the largest at the low frequencies.

FIGURE 5-12 Comparison between the unstable model (bold green line) and the stable approximations. The stable approximant [order] (solid line) and its complex error with the unstable model (dashed line): Initial 1 (red) [15/15], Initial 2 (blue) [17/17], Initial 3 (green) [17/17] and Initial 4 (black) [19/15]
5.3.1 Passive model

Although we already have created passive models by stabilizing the unconstrained model, we still want to illustrate the algorithm with the passivity constraints activated. For this three passive starting values are generated from the unstable model as explained in Chapter 4 Section 4.3.1. In Figure 5-14 shows the three different passive starting values and their complex error with the unstable model. If these passive starting values are fed to the iterative algorithm, we find the following results (see Figure 5-15). All the different passive initial value result in the same local minimum, which is the same minimum as the one found during the stabilization of the unstable model (see Figure 5-13). Increasing the order of the model does not improve the approximation much. The largest peak in the complex error (see Figure 5-13 and Figure 5-15) is at the low frequencies and as explained before, it is very difficult to improve the fit in that area due to the strong influence of the unstable poles. Increasing the order with more than one for the numerator and denominator polynomials, improves the approximation less than one percent. Again this is due to the low frequency
behavior, because there is a significant improvement for the higher frequencies (see Figure 5-16). In Figure 5-16 one can see the passive approximation and a version with increased order 1/1. If we compare this blue curve with the uncertainty bounds in Figure 5-11 then we see that
the complex error with the unstable model lies between the standard deviation of the measurements and the standard deviation of the unconstrained model, for the largest part of the frequency band.

5.3.2 Minimax constraint

Making a minimax constrained approximation of the unstable model is only possible for the stability constraint. For the passivity case no improvement step was found, that did not automatically violate the constraints. In Figure 5-17 one can see that the maximum error is lowered a bit, while the error becomes larger for the rest of the frequency band. The minimax stable model no longer coincides with the unconstrained model. The drop in maximum error is 4 dB.

5.3.3 Reparameterization of the numerator

To illustrate the reparameterization described in Chapter 4 Section 4.6, we start from the passive model obtained in Figure 5-15 to improve the approximation it in a post processing
step. In general creating passive models in this way from the passive starting values perform less well. Since the order is greater than 10 we used orthogonal polynomials in the refinement algorithm. The results shown in Figure 5-18 are not outstanding, the reparameterization succeeded in lowering the cost function with 0.75 dB.

**FIGURE 5-17**  Iterated stable approximations of the unstable model and minimax stable model: The iterated (minimax) stable approximant (solid line) and its complex error with the unstable model (dashed line): without minimax (blue) and with minimax (red).

**FIGURE 5-18**  Passive approximations of the non-passive model: The passive approximant (solid line) and its complex error with the non-passive model (dashed line): normal parametrization (black) and parametrization on the real part of the numerator (red).
5.3.4 Comparison with freely available toolboxes

RARL2 was not usable since the data is not scattering data. The VF method is used to create a stable approximant of order 18/17 and the results are shown in Figure 5-19. The passivation method presented in [29] cannot be used, because this method starts from a stable model created with VF and this model is already passive. As one can see in Figure 5-19 the model has a complex error that is about 5 dB lower compared with the VF model. If the VF model is used as starting values for the method presented in this thesis then the blue curve of Figure 5-19 is retrieved. Again for this comparison the numerator of the iterated version of Initial 3 is increased with one to have the same order for both models.

Also a 17th order passive model is created with the IdEM toolbox. The model was automatically passive after the stable identification step, so no compensation of the Hamiltonian matrix was needed. On Figure 5-20 one can see the passive model of the same order with the refinement algorithm and the IdEM toolbox. On can see on Figure 5-20 that the method presented in this thesis has a lower complex error 4 dB. If one uses the IdEM passive

![Comparison of different stabilization techniques](image)

**FIGURE 5-19** Comparison of different stabilization techniques: The transfer function (solid line) of the stable models and its complex error with the unstable model (dashed line) for: vector fitting (red) and our method (blue).
model as starting values, one sees that one retrieves more or less the same model than starting from one of the passive starting values.

In a last comparison the difference is shown between creating a stable model in a two-step approach, and directly selecting a stable model in the first step. In the unconstrained estimation step, different model orders were tried out. According to the MDL information criteria, the 18/17 model was the first stable model with the lowest MDL value. If we select this model and compare it with the refined stable model, where the unstable poles were flipped around the stability border, we get the results shown in Figure 5-21. From Figure 5-21 one can see that the stable model selected in the first step, is a quite good approximant of the data. However its overall complex error is about 5 dB larger compared with the stable model created in the two step approach. Even the complex error with the measurement data is 5 dB larger for the 1 step approach than for the 2 step approach.

FIGURE 5-20  Comparison of different passivation techniques: The transfer function (solid line) of the passive models and its complex error with the unconstrained model (dashed/dotted line) for: IdEM (bold blue), our method (back) and our method on Idem as starting values (gray).
FIGURE 5-21  Comparison of 1 step vs. 2 step approach: The transfer function (solid line) of the stable models and its complex error with the unconstrained model (dashed line) for: Refined initial 2 (blue), selected stable model in 1st step (black).
5.3.5 *Summary of the results*

<table>
<thead>
<tr>
<th>Complex error between the unconstrained model and...</th>
<th>Cost function [dB]</th>
<th>Maximum error [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>the measurements</td>
<td>-99.83</td>
<td>-90.49</td>
</tr>
<tr>
<td>standard deviation on the measurements</td>
<td>-107.54</td>
<td>-95.27</td>
</tr>
<tr>
<td>standard deviation on the unconstrained model</td>
<td>-116.00</td>
<td>-98.79</td>
</tr>
<tr>
<td>Initial 1</td>
<td>-72.54</td>
<td>-55.89</td>
</tr>
<tr>
<td>Initial 2</td>
<td>-72.58</td>
<td>-53.85</td>
</tr>
<tr>
<td>Initial 3</td>
<td>-68.95</td>
<td>-53.27</td>
</tr>
<tr>
<td>Initial 4</td>
<td>-71.02</td>
<td>-55.99</td>
</tr>
<tr>
<td>Iterated Initial 1</td>
<td>-95.77</td>
<td>-83.96</td>
</tr>
<tr>
<td>Iterated Initial 2</td>
<td>-103.30</td>
<td>-88.77</td>
</tr>
<tr>
<td>Iterated Initial 3</td>
<td>-103.30</td>
<td>-88.77</td>
</tr>
<tr>
<td>Iterated Initial 4</td>
<td>-90.01</td>
<td>-78.26</td>
</tr>
<tr>
<td>Passive 1 (change $b_{0i}$)</td>
<td>-67.41</td>
<td>-56.52</td>
</tr>
<tr>
<td>Passive 2 (change $b_{1i}$)</td>
<td>-67.41</td>
<td>-56.52</td>
</tr>
<tr>
<td>Passive 3 (change $b_{0i}$ and $b_{1i}$)</td>
<td>-67.89</td>
<td>-55.96</td>
</tr>
<tr>
<td>Iterated Passive 1 (change $b_{0i}$)</td>
<td>-103.30</td>
<td>-88.77</td>
</tr>
<tr>
<td>Iterated Passive 2 (change $b_{1i}$)</td>
<td>-103.30</td>
<td>-88.77</td>
</tr>
<tr>
<td>Iterated Passive 3 (change $b_{0i}$ and $b_{1i}$)</td>
<td>-103.30</td>
<td>-88.77</td>
</tr>
<tr>
<td>Increase of iterated Passive 2 with 1/1 (SPFV)</td>
<td>-108.09</td>
<td>-91.18</td>
</tr>
<tr>
<td>Iterated Initial 2 with Lawson weighting</td>
<td>-94.72</td>
<td>-92.80</td>
</tr>
<tr>
<td>Increase of iterated Initial 2 with 1/0 (SPFV)</td>
<td>-104.34</td>
<td>-88.85</td>
</tr>
<tr>
<td>IdEM Model</td>
<td>-99.26</td>
<td>-85.15</td>
</tr>
<tr>
<td>Iterated IdEM Model</td>
<td>-104.01</td>
<td>-89.71</td>
</tr>
<tr>
<td>VF model</td>
<td>-99.57</td>
<td>-85.55</td>
</tr>
</tbody>
</table>

**TABLE 5-3**  Summary of the results of the transformer example
5.4 Copper bar

We wish to thank Prof. Rik Pintelon and Dr. Kathleen De Belder in the Department of ELEC, Free University Brussels, Belgium for providing the measurement data. The experiment that was performed, was a longitudinal vibration analysis of a yellow copper bar (length: 2405 mm, radius: 5.9 mm) using an odd multisine of 151 sinusoids. The resulting validated model had 12 zeros and 13 poles, from which 5 were unstable. The frequency band of interest is between 810 Hz and 1170 Hz. In Figure 5-22 one sees the unconstrained model through the measurement data with its corresponding uncertainty bounds. First a stable approximation will be created starting from the four initial starting values. If one compares the different stable starting values in Figure 5-23, one sees that only initial 2 looks similar to the unconstrained model. Initial 1 is obviously a bad choice, since the order is reduced till 8, so it won’t be possible to capture all the dynamics. After the refinement step one finds the following results (see Figure 5-24). All the models improve with more than 20 dB compared with the starting values, but only the iterated version of initial 2 captures all the dynamics of the unconstrained model. The complex

![Figure 5-22](image-url)  
**FIGURE 5-22** Unconstrained approximation of a measured copper bar: FRF measurements (black dots), unconstrained estimated transfer function (bold green line), residuals between the unconstrained model and the measurements (green dashed-dotted line), standard deviation on the measurements (red pluses), standard deviation on the unconstrained model (blue pluses).
error between this model and the unconstrained model is smaller or of the same order of magnitude as the residuals. However it is still larger than the standard deviation on the

FIGURE 5-23 Comparison between the unstable model (bold green line) and the stable approximations. The stable approximant [order] (solid line) and its complex error with the unstable model (dashed line): Initial 1 (red) [7/8], Initial 2 (blue) [12/13], Initial 3 (green) [12/13] and Initial 4 (black) [17/8]

FIGURE 5-24 Comparison between the unstable model (bold green line) and the iterated stable approximations. The stable approximant [order] after the refinement step (solid line) and its complex error with the unstable model (dashed line): Initial 1 (red) [7/8], Initial 2 (blue) [12/13], Initial 3 (green) [12/13] and Initial 4 (black) [17/8]
unconstrained model. To make a stable model with a smaller complex error, we will increase the order of our best stable model. The stable model is increased in the same way like the crankcase example. From 1/1 to 4/4 in one big step. This results in a drop in error of respectively about 1, 15, 20 and 38 dB (see Figure 5-25). So by increasing the order, it is possible to create a stable model with a smaller complex error than the standard deviation of the unconstrained model.

5.4.1 Minimax constraint

To create a minimax stable model, the iterated version of initial 2 is again selected as starting value. This result in the following model (see Figure 5-26). The maximum error drops with about 6 dB by making a minimax approximation. If we compare Figure 5-26 with Figure 5-22 we see that the error is of the same order of magnitude as the residuals.
5.4.2 Passive model

The iterated version of initial 2 is only stable and not passive, so the three passive initial values are generated to initiate the algorithm with the passivity constraints. One can see in Figure 5-27 that only the passive initial value, where the passivity constraint is imposed by adapting $a_{0i}$, follows the dynamics of the unconstrained model. The other 2 passive initial values have a much larger complex error with the unconstrained model (more than 20 dB). However after the refinement step, we get a completely other picture. The worst passive starting values give now the best results, while the best starting values results in the worst passive model (see Figure 5-28). Although the best (lowest cost function) passive model coincides with the unconstrained model, there is still a bias introduced by passivating the unstable model. The residuals of the unstable model with the measurements are still 10 to 20 dB smaller than the complex error between the unstable model and the passive best model. Increasing the model order could be a solution to improve the passive fit, but this was no big success; only when just the order of the numerator was increased, a much better passive model. These models are illustrated in Figure 5-30 for an increase of 2/0 and Figure 5-31 for an increase of 1/0.

![Iterated stable approximations of the unstable model and minimax stable model](image)
FIGURE 5-27 Comparison between the unstable model (bold green line) and the passive approximations. The passive approximant (solid line) and its complex error with the unstable model (dashed line): change $b_{1i}$ (red), change $b_{0i}$ (blue), change $b_{1i}$ and $b_{0i}$ (green).

FIGURE 5-28 Comparison between the unstable model (bold green line) and the passive approximations. The passive approximant after iteration (solid line) and its complex error with the unstable model (dashed line): change $b_{1i}$ (red), change $b_{0i}$ (blue), change $b_{1i}$ and $b_{0i}$ (green).
5.4.3 Reparameterization of the numerator

To illustrate the reparameterization described in Chapter 4 Section 4.6, we start from the passive model obtained in Figure 5-28 to improve the approximation it in a post processing step. In general creating passive models in this way from the passive starting values perform less well. Since the order is greater than 10 we used orthogonal polynomials in the refinement algorithm. The results shown in Figure 5-29 are not outstanding, however, the reparameterization succeeded in lowering the cost function with 0.25 dB.

5.4.4 Comparison with freely available toolboxes

The VF toolbox was used to create a stable model with 13 poles en 14 zeros. The best stable model (the blue curve in Figure 5-24) is increased with 2/0 to create a stable model of the same order. From Figure 5-30, it can easily be seen that the stable model created with the presented method of this thesis performs better that the VF model. The complex error with the unconstrained model is 9 dB lower for the presented method.
Comparison with the passivating technique presented in [29] was done. In a first instance the VF method was used to provide the stable model for the passivating method. However, the linear term \( h \) (in (3-39)) was negative so the passivating algorithm puts this term equal to zero [29]. Since this term has no influence on the real part of the transfer function, we had a good approximation of the real part but the imaginary part was terrible. This resulted in a complex error that was even worse than the worst passive starting value (the green curve in Figure 5-27). Even starting from the best stable model after refinement was not possible to obtain a passive model, even for a high tolerance number on the real part of the function (tol =100).

The IdEM toolbox resulted in a much better solution. Figure 5-31 shows the 13th order passive model created by the IdEM toolbox. The created model was automatically passive after the stable identification step. It has a complex error with the unconstrained model that is 8 dB larger than the passive model created by the presented method of the same order (13/13). As explained before, we increased the model (the green curve in Figure 5-28) to a proper model by adding a constant. Now, if we use the IdEM model as passive starting values, then we still

![Comparison of different stabilization techniques](image-url)
can lower the error by 7 dB. So this passive model is of the same order of magnitude as the residuals of the measurements and the unstable model, so the introduced bias is minimal. We also used the IdEM toolbox to create passive models of order 12 and 14. These models needed a compensation step after the stable identification step and it created passive models of the same order of magnitude as the green line in Figure 5-28, so the corresponding cost functions is about 15 dB larger than for the 13th order model.

FIGURE 5-31  Comparison of different passivation techniques: The transfer function (solid line) of the stable models and its complex error with the unconstrained model (dashed/dotted line) for: IdEM (bold blue), the refinement method (black) and the refinement method on Idem as starting values (gray).
### 5.4.5 Summary of the results

<table>
<thead>
<tr>
<th>Complex error between the unconstrained model and ...</th>
<th>Cost function [dB]</th>
<th>Maximum error [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>the measurements</td>
<td>9.30</td>
<td>26.53</td>
</tr>
<tr>
<td>standard deviation on the measurements</td>
<td>-6.68</td>
<td>9.33</td>
</tr>
<tr>
<td>standard deviation on the unconstrained model</td>
<td>-13.08</td>
<td>6.10</td>
</tr>
<tr>
<td>Initial 1</td>
<td>48.83</td>
<td>56.52</td>
</tr>
<tr>
<td>Initial 2</td>
<td>35.48</td>
<td>45.34</td>
</tr>
<tr>
<td>Initial 3</td>
<td>53.79</td>
<td>61.51</td>
</tr>
<tr>
<td>Initial 4</td>
<td>53.12</td>
<td>62.83</td>
</tr>
<tr>
<td>Iterated Initial 1</td>
<td>35.78</td>
<td>55.82</td>
</tr>
<tr>
<td>Iterated Initial 2</td>
<td>-4.60</td>
<td>6.15</td>
</tr>
<tr>
<td>Iterated Initial 3</td>
<td>36.94</td>
<td>55.73</td>
</tr>
<tr>
<td>Iterated Initial 4</td>
<td>-36.62</td>
<td>55.84</td>
</tr>
<tr>
<td>Increase of iterated Initial 2 with 1/1 (SPFV)</td>
<td>-5.60</td>
<td>4.56</td>
</tr>
<tr>
<td>Increase of iterated Initial 2 with 2/2 (SPFV)</td>
<td>-20.08</td>
<td>-4.66</td>
</tr>
<tr>
<td>Increase of iterated Initial 2 with 3/3 (SPFV)</td>
<td>-27.81</td>
<td>-13.77</td>
</tr>
<tr>
<td>Increase of iterated Initial 2 with 4/4 (SPFV)</td>
<td>-46.17</td>
<td>-35.1</td>
</tr>
<tr>
<td>Iterated Initial 2 with Lawson weighting</td>
<td>-1.73</td>
<td>0.30</td>
</tr>
<tr>
<td>Passive 1 (change $b_{0i}$)</td>
<td>27.76</td>
<td>33.59</td>
</tr>
<tr>
<td>Passive 2 (change $b_{1i}$)</td>
<td>47.39</td>
<td>53.84</td>
</tr>
<tr>
<td>Passive 3 (change $b_{0i}$ and $b_{1i}$)</td>
<td>48.75</td>
<td>56.48</td>
</tr>
<tr>
<td>Iterated Passive 1 (change $b_{0i}$)</td>
<td>25.22</td>
<td>29.49</td>
</tr>
<tr>
<td>Iterated Passive 2 (change $b_{1i}$)</td>
<td>13.11</td>
<td>20.00</td>
</tr>
<tr>
<td>Iterated Passive 3 (change $b_{0i}$ and $b_{1i}$)</td>
<td>6.50</td>
<td>17.31</td>
</tr>
<tr>
<td>Increase of iterated Initial 2 with 2/0 (SPFV)</td>
<td>-8.26</td>
<td>1.89</td>
</tr>
<tr>
<td>VF model</td>
<td>0.84</td>
<td>10.75</td>
</tr>
<tr>
<td>Increase of iterated Passive 3 with 1/0 (SPFV)</td>
<td>-7.568</td>
<td>4.28</td>
</tr>
<tr>
<td>IdEM Model</td>
<td>1.06</td>
<td>12.19</td>
</tr>
<tr>
<td>Iterated IdEM Model</td>
<td>-6.78</td>
<td>5.19</td>
</tr>
</tbody>
</table>

**TABLE 5-4** Summary of the results of the copper bar example
5.5 Two-port Inductor

We wish to thank Prof. Gerd Vandersteen in the Department of ELEC, Free University Brussels, Belgium for providing the measurement data. In the following experiment we want to extract a scalable inductor model from microwave measurements. This is required when introducing state-of-the-art inductor technology into a Spice-like simulator. The off-chip inductors made in MCM (Multi Chip Module) technology have a planar structure. Due to this structure and the high frequency band of interest (45 MHz-50 GHz), 2-port measurements are required. The 1.5-turn inductors had identical diameters (100 μm) and conductor spacing (20 μm) and were excited with a stepped sine. This resulted in a spectrum of 402 sinusoids in the frequency band; the frequency spacing Δf between consecutive sinusoids is 249.775 MHz. The 11th order model of the 1.62 nH inductor with a conductor width of 50 μm, is unstable. In Figure 5-32 one can see the transfer function of the unconstrained model and its complex error with the original measurement data.

Two of the eleven poles are unstable and lie outside the frequency band of interest. The imaginary part of the 2 unstable poles lie twice as far as the highest angular frequency of interest, and have also the largest real part of all the poles. We took as stability border the

![Figure 5-32](image)

**FIGURE 5-32** The 4 figures show the transfer function for the 4 S-parameters: FRF measurements (black dots), unconstrained estimated transfer function (bold green line), residuals between the unconstrained model and the measurements (green dashed-dotted line).
largest real part of the stable poles. For these measurements also we applied the three initial stabilization methods (initial 1-3) to the unstable model (see Figure 5-33). Initial 4 was not selected, since it was the purpose to create a passive approximant. Applying initial 4 on the unstable model would result in a 13/9 stable model, however, to create a passive model, one can only have a maximum absolute relative difference between the numerator and denominator of one. The lowest complex error corresponds to initial 1 and the highest to initial 3. After the refinement step we get a total different picture. The worst starting value corresponds now to the best model, that coincides with the iterated version of initial 2; while initial 1 did not succeed in improving a lot the approximation error. Comparing Figure 5-34 with Figure 5-32 it can be calculated that we have created good stable models without introducing a significant bias. Deactivating the fixing the poles constraint did not give worse results since the stabilized unstable poles have little influence in the user defined frequency band.

5.5.1 Increasing the order

Also for MIMO systems we can increase the order. The order of numerator and denominator of iterated version of initial 2 is increased by 1 to 3. This increase in order results in a drop of the
FIGURE 5-34 The 4 figures show the transfer function for the 4 S-parameters: unconstrained estimated transfer function (bold green line), The stable approximant after the refinement step (solid line) and its complex error with the unconstrained model (dashed line): Initial 1 (red), Initial 2 coincides with Initial 3 (green)

FIGURE 5-35 Iterated stable approximations of the unstable model for different order increases: The iterated stable approximant (solid line) from initial 3 and its complex error with the unstable model (dashed line), with an order increase of: 1/1 (blue), 2/2 (red) and 3/3 (green)

complex error with the unstable model of respectively 3, 21 and 17 dB (see Figure 5-35). It may seem strange that an increase of the order by 3/3 performs less than an increase by 2/2.
This is perfectly normal, since this is a non-convex optimization, so the local minima of the highest increase can lie higher than the other one.

5.5.2 Comparison with freely available toolboxes

This time we have scattering data so the RARL2 toolbox can be used. RARL2 will search the closest stable model to the unconstrained model. However, since the reference is an unstable model, this iterative process can fail if the constraint border is reached. Also the VF toolbox is used to create a stable model. In Figure 5-36 one can see the stable approximations of the 3 different methods with there corresponding complex error with the unstable model. From Figure 5-36 it is easy to see that VF does a better job than RARL2, which is not surprising because RARL2 works only optimal in combination with Hyperion [32]. If we compare in Figure 5-36, the transfer function of the VF method with the best stable model of Figure 5-34 with an order increase of 1/0 to make it also improper, we see that their complex errors are of the same order of magnitude for S12 and S21, while we have a better approximation for S11 and S22. The total complex error V is 11 dB smaller compared to the VF model;
with the number of output entries and the number of input entries.

5.5.3 Passive models

Since we don’t have a standard method to create passive initial values for MIMO models, we used the passive model \( \hat{G}(\theta, s) \) created by the IdEM toolbox as starting values for the refinement algorithm. To check the passivity constraints, a poly-toolbox for Matlab was used [55] to calculate the roots of \( \det(\hat{G}(\theta, s) + \hat{G}^T(\theta, -s)) \). This results in the passive model shown in Figure 5-37. The passive model created by the IdEM model needed a compensation step after the stable identification step. The refinement algorithm succeeds in lowering the total complex error by 4 dB, especially the high frequency behavior of the model is improved for S12, S21 and S22. If one compare Figure 5-32 carefully to Figure 5-37 it is easy to see that no

\[
V = \frac{1}{F} \sum_{j=1}^{p} \sum_{i=1}^{m} \sum_{l=1}^{F} w^2_{ij} |G_{ij}(\Omega_l) - \hat{G}_{ij}(\Omega_p, \theta)|^2
\]

(5-1)

with \( m \) the number of output entries and \( p \) the number of input entries.
significant bias is introduced for 3 of the 4 S-parameters. Only at the low frequencies of S21 we have a complex error that is larger than the residuals.

5.5.4 Summary of the results

<table>
<thead>
<tr>
<th>Complex error between the unconstrained model and ...</th>
<th>Cost function [dB]</th>
<th>Maximum error [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>the measurements</td>
<td>-31.75</td>
<td>-22.16</td>
</tr>
<tr>
<td>Initial 1</td>
<td>-14.97</td>
<td>-13.63</td>
</tr>
<tr>
<td>Initial 2</td>
<td>-14.81</td>
<td>-15.62</td>
</tr>
<tr>
<td>Initial 3</td>
<td>-10.25</td>
<td>-8.35</td>
</tr>
<tr>
<td>Iterated Initial 1</td>
<td>-36.85</td>
<td>-26.21</td>
</tr>
<tr>
<td>Iterated Initial 2</td>
<td>-65.11</td>
<td>-56.04</td>
</tr>
<tr>
<td>Iterated Initial 3</td>
<td>-65.11</td>
<td>-56.04</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/1 (SPFV)</td>
<td>-67.75</td>
<td>-57.99</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 2/2 (SPFV)</td>
<td>-86.07</td>
<td>-75.11</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 3/3 (SPFV)</td>
<td>-81.61</td>
<td>-70.75</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/0 (SPFV)</td>
<td>-67.75</td>
<td>-57.99</td>
</tr>
<tr>
<td>RARL2 model</td>
<td>-37.13</td>
<td>-29.18</td>
</tr>
<tr>
<td>VF model</td>
<td>-56.88</td>
<td>-48.51</td>
</tr>
<tr>
<td>IdEM Model</td>
<td>-39.51</td>
<td>-32.05</td>
</tr>
<tr>
<td>Iterated IdEM Model</td>
<td>-43.06</td>
<td>-41.37</td>
</tr>
</tbody>
</table>

TABLE 5-5 Summary of the results of the two-port inductor example

5.6 SMD resistor

We wish to thank ir. Stephane Bronckers of IMEC, Leuven, Belgium for providing the measurement data. In this experiment a 50 Ohm surface-mount-device (SMD) (package 0603) resistor was measured with an impedance analyzer to analyze its high frequency behavior. Ten independent measurements have been performed in the frequency band 1MHz to 1GHz with 500 logarithmically distributed frequency points. An ideal resistor should have a constant value over the whole frequency band. The unconstrained model was estimated with the Frequency
Domain System Identification-toolbox [36] giving a 3rd order model with 2 stable poles and 1 unstable pole. The transfer function and the measurements with their standard deviations are shown in Figure 5-38. From Figure 5-38 one can see that there is a very good agreement between the measurements and the unconstrained model.

5.6.1 Stable models

The initial stable models are generated from this unstable model. Initial 4 does not physically have the right properties since its magnitude grows with the frequencies (order of 4/2). However if one looks at Figure 5-39 it is clear that this stable initial model corresponds to the lowest complex error. Leaving out the unstable part results in a poor stable approximation. After the refinement step we see in Figure 5-40 that all the iterated stable models perform pretty good. It is clear that the iterated initial 4 outer performs. If we compare Figure 5-40 with Figure 5-38, it can be seen that only the iterated version of initial 4 is of the same order of magnitude as the residuals between the measurements and the unconstrained model and that all the others have introduced a bias by stabilizing the unconstrained model. The stable models
cannot model accurately the high frequency behavior. This is not surprising because the flipped original pole lies at these frequencies. To decrease the approximation below the standard
deviation of the unconstrained model, one can try to increase the order of the model. The model order was increased by respectively 1/1 to 4/4 in one big step, started from the proper stable model created from initial 2. The order increase of the stable model by 1/1 to 3/3 does not change a lot the quality of the stable model: the cost function improved at most 1.5 dB. Only when the model order is increased by 4/4 we have a serious drop in the approximation error. If we compare now Figure 5-41 with Figure 5-38 than we see that for the low frequencies we have a lower error than the standard deviation on the unconstrained model, but for the higher frequencies it becomes somewhat larger.

5.6.2 Passive models

The stable models obtained in the previous subsection are not automatically passive. Hence, the three passive initial models are generated from the unconstrained model. This results in the models shown in Figure 5-42. All the three different initial passive models are very close to each other, so they coincide on Figure 5-42. If these models are iterated then we find the models shown in Figure 5-43. This results in a drop of 23 dB compared with the starting
values. However it is clear that by making a passive approximation of the unstable model we have introduced a significant bias if we compare Figure 5-43 with Figure 5-38. Increasing the
order has no great success, because we only gain at most 3.5 dB. So with the presented method it is not possible to push the bias error below the uncertainty of the unstable model.

5.6.3 Comparison with freely available toolboxes

Also for this example no comparison is possible for the RARL2 software. The VF toolbox was used to create a stable model of order 4/3. The passive model from Figure 5-43 is increased with 1/0 to create a stable model of the same order. From Figure 5-44 one can see that the stable model obtained with the techniques described in this thesis performs better than the VF model. The gain in cost function is 2 dB. Now, if we use this stable model (VF) as starting values for the refinement step, then we see on Figure 5-44 that we can still lower the approximation error with more than 14 dB.

The stable VF model was also used to create a passive approximation [29]. This was not successful because, there was always a place out-band where the real part became smaller than zero. Even by adding extra frequency points outside this band, we did not succeed in creating a passive approximation. The only way that a passive approximation could be made from this stable model was to add a very dense grid of out band frequency points and a high tolerance,
but the passive model obtained was useless. Only when we start from a lower order VF approximation 3/2, we succeeded to create a passive model that looks similar to the unconstrained model. But again the complex error with the unstable model is worse than the passive initial values in Figure 5-42.

The toolbox IdEM was used to create a passive model from the unconstrained model. The obtained 3rd order model was passive after the stable identification step, so no compensation of the Hamiltonian matrix was needed. Figure 5-45 shows the passive model created with IdEM, and the iterated passive model from Figure 5-43. It is clear that the model created with the passivation technique of this thesis has a poorer approximation at the lower frequencies, while it performs better above 100 MHz. However the cost function of the passive approximation with the unconstrained model lies 3.5 dB lower than the one created with the IdEM toolbox. The passive model created by IdEM is not a minimum for the presented refinement algorithm, we still succeeded to lower the cost function by 2 dB. It is clear that both methods introduced a significant bias by making the approximations passive.
5.6.4 Reparameterization of the numerator

To illustrate the reparameterization described in Chapter 4 Section 4.6, we start from the passive model obtained in Figure 5-43 to improve the approximation it in a post processing step. In Figure 5-46 one can see that the created approximation with this post processing step has a smaller approximation error at the lower frequencies, although we only have a drop of the cost function of 2.5 dB. If we compare it with the passive model created with the IdEM toolbox (see Figure 5-45) then we see that this new passive model has a better approximation over the whole frequency band.

5.6.5 Summary of the results
<table>
<thead>
<tr>
<th>Complex error between the unconstrained model and ...</th>
<th>Cost function [dB]</th>
<th>Maximum error [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>the measurements</td>
<td>-38.31</td>
<td>-26.50</td>
</tr>
<tr>
<td>standard deviation on the measurements</td>
<td>-41.75</td>
<td>-36.91</td>
</tr>
<tr>
<td>standard deviation on the unconstrained model</td>
<td>-65.42</td>
<td>-52.80</td>
</tr>
<tr>
<td>Initial 1</td>
<td>32.46</td>
<td>32.61</td>
</tr>
<tr>
<td>Initial 2</td>
<td>17.33</td>
<td>27.59</td>
</tr>
<tr>
<td>Initial 3</td>
<td>19.01</td>
<td>29.89</td>
</tr>
<tr>
<td>Initial 4</td>
<td>-2.63</td>
<td>11.12</td>
</tr>
<tr>
<td>Iterated Initial 1</td>
<td>-25.78</td>
<td>-9.92</td>
</tr>
<tr>
<td>Iterated Initial 2</td>
<td>-26.35</td>
<td>-10.28</td>
</tr>
<tr>
<td>Iterated Initial 3</td>
<td>-26.35</td>
<td>-10.28</td>
</tr>
<tr>
<td>Iterated Initial 4</td>
<td>-39.19</td>
<td>-28.10</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/1 (SPFV)</td>
<td>-26.46</td>
<td>-10.97</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 2/2 (SPFV)</td>
<td>-27.98</td>
<td>-16.35</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 3/3 (SPFV)</td>
<td>-27.77</td>
<td>-15.84</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 4/4 (SPFV)</td>
<td>-49.28</td>
<td>-33.55</td>
</tr>
<tr>
<td>Passive 1 (change $b_{0i}$)</td>
<td>12.07</td>
<td>23.19</td>
</tr>
<tr>
<td>Passive 2 (change $b_{1i}$)</td>
<td>12.07</td>
<td>23.19</td>
</tr>
<tr>
<td>Passive 3 (change $b_{0i}$ and $b_{1i}$)</td>
<td>12.07</td>
<td>23.19</td>
</tr>
<tr>
<td>Iterated Passive 1 (change $b_{0i}$)</td>
<td>-11.28</td>
<td>2.64</td>
</tr>
<tr>
<td>Iterated Passive 2 (change $b_{1i}$)</td>
<td>-11.28</td>
<td>2.64</td>
</tr>
<tr>
<td>Iterated Passive 3 (change $b_{0i}$ and $b_{1i}$)</td>
<td>-11.28</td>
<td>2.64</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/0 (SPFV)</td>
<td>-26.46</td>
<td>-10.97</td>
</tr>
<tr>
<td>VF model</td>
<td>-24.02</td>
<td>-7.25</td>
</tr>
<tr>
<td>IdEM Model</td>
<td>-7.61</td>
<td>8.33</td>
</tr>
<tr>
<td>Iterated IdEM Model</td>
<td>-9.75</td>
<td>6.16</td>
</tr>
</tbody>
</table>

**TABLE 5-6**  Summary of the results of the SMD resistor example
5.7 High frequency planar inductor

We wish to thank ir. Karen Scheir of IMEC, Leuven, Belgium for providing the measurement data. This measurement example is similar to the one of Section 5.5. It consist of measuring the S-parameters of a planar inductor in the frequency band of 20 GHz to 40 GHz. The unconstrained model is of order 6 and has 4 unstable poles. Two of these unstable poles lie in the frequency band of interest, so these violate a prerequisite of the proposed method. Figure 5-47 shows the transfer function of the unconstrained model with its complex error with the original measurement data.

For these measurements we also applied the three initial stabilization methods (initial 1-3) on the unstable model (Figure 5-48). Initial 4 was not selected, since it was the purpose to create a passive approximation, namely the corresponding model order is 8/4, which can never be passive. We took as extra constraint border that the real part of the poles cannot be larger than half of the largest real part of the stable poles of the starting values. On Figure 5-48 one can see a peak in the error for all the four S-parameters. This corresponds to the positions of the poles.

FIGURE 5-47 The 4 figures show the TF for the 4 S-parameters: FRF measurements (black dots), unconstrained estimated transfer function (bold green line), residuals between the unconstrained model and the measurements (green dashed-dotted line).
unstable poles in this frequency band. The smallest complex error corresponds to initial 1 and the largest to initial 3. After the refinement step we get a total different picture. The worst
starting value corresponds now to the best model, that coincides with the iterated version of initial 2; while initial 1 did not succeed in improving a lot. Still we see that the refinement algorithm has the most trouble in lowering the error around the original unstable poles in the frequency band of interest. If one compare Figure 5-49 with Figure 5-47 it can easily be seen that we have created good stable models without introducing a significant bias. Deactivating the separation of the parameters in a ‘fixed’ and ‘variable’ group corresponds to a cost function that is as twice as large. If looser user defined constraint border is used as stability border, the cost function drops by a factor 2.

5.7.1 Increasing the order

Also for MIMO systems we can increase the order. The order of the numerator and denominator of the iterated version of initial 2 is increased by 1 to 3. This increase in order results in a drop of the complex error with respect to the unstable model of respectively 0.4, 7.3 and 9 dB (see Figure 5-50). This measurement example is a good illustration why the order should be increased with at least 2. From Figure 5-50 it can be seen that only the order increase
2 or 3 succeed in capturing the original peak in the complex error between the iterated version of initial 2 and the unconstrained model.

### 5.7.2 Comparison with freely available toolboxes

This time we have scattering data, so the RARL2 toolbox can be used. RARL2 will search the closest model to the unconstrained model, since the completion step could not be performed as explained in section 5.1. Also the VF toolbox is used to create a stable model. In Figure 5-51 one can see the stable approximations of the 3 different methods with their corresponding complex error with the unstable model. From Figure 5-51 it can easily be seen that RARL2 does a better job than VF: the corresponding cost function is 5 dB lower for the stable model created by RARL2. This is due to the bad VF fit of the S21 and S12 parameters. Still if we compare these stable models with the iterated version of initial 2, then one can see in Figure 5-51 that the presented method of this thesis still outer performs the other two.
5.7.3 Passive models

It was possible to create a ‘handmade’ passive model by flipping the unstable poles around the imaginary axis and shift them away from the imaginary axis. After the refinement step one gets the passive model illustrated by the black line in Figure 5-52. Also the IdEM toolbox was used to created a passive model of the same order. The toolbox needed two compensation steps to create a passive model. Also this model was used as starting values for the refinement algorithm. Figure 5-52 shows the three different passive models with their complex error with the unconstrained model. The cost function $V$ (see (5-1)) of the ad hoc passive model, the IdEM model and the iterated version of the Idem model are respectively -41.1 dB, -43.3 dB and -49.3 dB. If we compare the passive IdEM model with its iterated version, we see that especially the high frequency behavior of the model is improved.
### 5.7.4 Summary of the results

<table>
<thead>
<tr>
<th>Complex error between the unconstrained model and...</th>
<th>Cost function [dB]</th>
<th>Maximum error [dB]</th>
</tr>
</thead>
<tbody>
<tr>
<td>the measurements</td>
<td>-51.20</td>
<td>-46.14</td>
</tr>
<tr>
<td>Initial 1</td>
<td>-41.09</td>
<td>-39.61</td>
</tr>
<tr>
<td>Initial 2</td>
<td>-39.92</td>
<td>-35.69</td>
</tr>
<tr>
<td>Initial 3</td>
<td>-36.79</td>
<td>-34.55</td>
</tr>
<tr>
<td>Iterated Initial 1</td>
<td>-51.62</td>
<td>-41.53</td>
</tr>
<tr>
<td>Iterated Initial 2</td>
<td>-66.98</td>
<td>-57.82</td>
</tr>
<tr>
<td>Iterated Initial 3</td>
<td>-66.98</td>
<td>-57.82</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/1 (SPFV)</td>
<td>-67.34</td>
<td>-57.94</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 2/2 (SPFV)</td>
<td>-74.23</td>
<td>-68.11</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 3/3 (SPFV)</td>
<td>-75.92</td>
<td>-68.90</td>
</tr>
<tr>
<td>Increase of iterated Initial 3 with 1/0 (SPFV)</td>
<td>-67.34</td>
<td>-57.94</td>
</tr>
<tr>
<td>RARL2 model</td>
<td>-54.78</td>
<td>-41.75</td>
</tr>
<tr>
<td>VF model</td>
<td>-50.32</td>
<td>-43.62</td>
</tr>
<tr>
<td>IdEM Model</td>
<td>-43.37</td>
<td>-37.74</td>
</tr>
<tr>
<td>Iterated IdEM Model</td>
<td>-49.33</td>
<td>-45.76</td>
</tr>
</tbody>
</table>

**TABLE 5-7** Summary of the results of the high frequency planar inductor example
The purpose of this work was to create stable and passive models from noisy measurement data. To do so a two-step approach was proposed consisting of an unconstrained estimation step from the noisy data followed by a constrained approximation step. In this first step a model without any constraints is estimated that passes the validation tests (analysis cost function, whiteness weighted residues, ...). This step suppresses in an optimal way the noise without introducing systematic errors. In the second step the constraint is added: The (high) order model is approximated by a constrained one. For this the weighted difference between the unconstrained model and the constrained model is minimized. The big advantage of the two step procedure is that it provides models with uncertainty and bias error bounds.

The contribution to the first step was an extension of the nonparametric identification step in the presence of non-stationary components. This consists of calculating three different sample variances of repeated measurements. Comparison of these sample variances allows the user to detect the presence of non-stationary components.
Conclusions

In the literature the most stability enforcement techniques are one step approaches where the user has no idea of the amount of bias that is introduced by creating a stable model. Now, the presented method of this thesis gives the user an idea about the amount of introduced bias (if any) by approximating the unconstrained model by a stable model. This information is of course due to the first step of the two-step approach. It is even possible to lower this bias by increasing the order of the stable approximation. This stable approximation step consists itself of two steps. First creating stable starting values, by means of some simple manipulations of the unconstrained model or by other more advanced techniques shown in the literature. Secondly putting these starting values in a refinement algorithm where the gain and all the poles and zeros and gain are allowed to vary. So the stable model obtained with the refinement step is always at least as good as its starting value. Every different starting value results in a different minimum of the cost function. In most of the examples starting from initial 2 or initial 3 resulted in the stable model with the smallest model errors. The only method that is very similar to the presented two-step approach is the one that is available as a software package RARL2. Here the user must have an idea about the amount of energy that should be allowed outside the user-defined frequency band.

In the literature, the passivity enforcement techniques are in general 2 step approaches. However, most of these techniques need a stable model as initial values for the passive approximation step. This is because the approximation step becomes much easier if only the numerator is allowed to change to create a passive approximant. Of course these techniques only works fine if on has a good approximation of the poles. Nevertheless, as shown in the application chapter, allowing also the denominator to change can still improve the passive model significantly. The refinement algorithm is the same as for the stability approximation with as difference that for this case the passivity constraints need to checked. Since these passivity constraints are much stricter than the stability constraints, the refinement algorithm will more often get stuck in to a local minimum. Some extra techniques were proposed to deal with the problem of local minima in the case of the passivity constraints, like reparameterizing the model in the parameters of the denominator and the parameters of the real part of the numerator.
It is also shown that it is possible to create minimax constrained models, via an adaptive weighting scheme in combination with the same refinement algorithm. However the results are much better for the stability constraint as for the passivity constraint.

An overview of the identification procedure presented in this thesis is summarized in Figure 6-1.

A disadvantage is that the cost function can only become zero in a point that is not a member of the set of possible solutions (for example models with unstable poles, in the stability case). That's why the algorithm can get stuck in a point where some poles become a member of the stability border. Since poles on the stability border are often unwanted, one can add some user-defined constraint borders.

Nevertheless, this technique is very easy to implement and gives quite good results as shown in the application chapter. Mostly the stable approximations introduce a systematic error that is smaller than the residuals of the unconstrained model with the stable model. When the systematic error introduced is larger, then it can be still possible to improve the constrained model by increasing the order of the constrained model. To create a passive approximation of the unconstrained model, it is observed that more often a bias is introduced in the approximation step. Although the presented approximation techniques does in general a better job than the freely available toolboxes IdEM and Vect.fit2.

To finish the conclusions, the main contributions of this thesis are summarized:

- Non-parametric identification in the presence of non-stationary components.
- Two-step approach: an unconstrained estimation step followed by a constrained approximation step; this to have knowledge about the bias possibly introduced by the approximation step.
Conclusions

Data

Unconstrained modeling step

Stable starting values

Refinement Algorithm with user-defined constraints

Passive starting values

Increase of the order of the stable model

Increase of the order of the passive model

Reparameterization

Constrained minimax approximation

User-defined stable model

User-defined passive model

FIGURE 6-1 Flow-chart of the identification of guaranteed stable and passive models from noisy data.
• Stabilization and passivation technique based on a constrained non-linear least squares algorithm, where all the parameters are allowed to vary.

• MIMO passivity check algorithm based on spectral zeros.

• Improvement of the constrained model by increasing the order.

• Some techniques to deal with the local minima caused by the passive constraints.

• Improvement of the refinement algorithm by separating the poles in a ‘fixed’ and ‘variable’ group.

• User-defined constrained models by adding arbitrary constraint borders.

• An adaptive weighting procedure for creating constrained minimax models.
A new stable optimization scheme would consist of using the method of Sanathanan [49] with stability constraints. This constraint could be added as a linear matrix inequality (LMI) (Lyapunov equation) or as Routh-Hurwitz constraints on the parameters. This would be quite similar to the vector fitting method, with as differences that the gain, the poles and the zeros are fitted at the same time, while the poles are restricted by the constraints, and not by flipping them around the stability border. However adding additional constraint borders will no longer be easy.

Adding the minimum phase constraint to the stability constraint. This extension is very straight forward and similar to the stability constraint.

Extensions should be made to allow to free the fixed parameters. This can be done by calculating in each iteration step the full Jacobian and the restricted Jacobian. When the step of the full Jacobian pushes the poles over the constraint border, the restricted Jacobian should be used. This will of course slow down the refinement step significantly, since two possible steps should be checked.
As explained in [5] one can create an excellent stable model through the data in a user-defined frequency band at the cost of nearly destabilizing it at the remaining frequencies. The reason for this is that one needs constraints on the amount of energy that should be allowed outside the frequency band of interest. This could be a problem if the order is increased to improve the stable fit. To solve this problem extra constraints should be added to control the amount of energy outband. This is however not a simple task.

Similar to the stability constraint, a new passivity optimization scheme would consist of using the method of Sanathanan [49] with passivity constraints. These constraints could be added as a Hamiltonian or as constraints on the coefficients of the numerator and denominator. This method would consist of a NLLS step on the model, were only the denominator is allowed to vary, followed by a LS step on the model were only the numerator coefficients are allowed to change. Of course in both steps there are respectively passivity constraints on the poles and the zeros of the model. This approach would be quite similar to the vector fitting method, with the main difference that the poles and zeros are allowed to be adapted instead of only the zeros.

The refinement algorithm could be further improved by de-emphasize the areas where \( G(s, \theta) + G^T(-s, \theta) \) is very close to zero. Due to these areas, the refinement algorithm often ends up in a local minimum. These areas can be detected by the zeros of \( G(s, \theta) + G^T(-s, \theta) \) with a very small real part (see Table 4-1 and Table 4-2). Of course this is only applicable for the violation of \( G(j\omega, \theta) + G^T(-j\omega, \theta) \geq 0 \) in the user-defined frequency band.

The use of passivity conserving order-reduction techniques in combination with increasing the order can be a possible path to escape from the local minima caused by the passivity constraints.
REFERENCES


