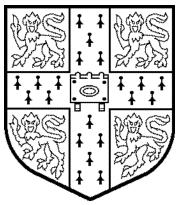
Weight Selection in Robust Control: An Optimisation Approach

Alexander Lanzon

Wolfson College



Control Group
Department of Engineering
University of Cambridge

A dissertation submitted for
the degree of Doctor of Philosophy

October 2000

To Laura and my parents
for their
love and understanding

Abstract

The primary purpose of this thesis is to present optimisation algorithms which facilitate the selection of weights in robust control techniques. The robust control paradigms investigated in this thesis are μ-synthesis and $\mathscr{H}_\infty$ loop-shaping, as these techniques offer a systematic framework for synthesising sensible controllers that meet performance objectives and guarantee robustness to model uncertainty and unmeasured disturbances. This dissertation is essentially composed of two parts.

In the first part of the thesis, a conceptually new approach to the μ-synthesis robust performance problem is presented, whereby an optimisation problem is proposed which maximises the performance weights in the frequency ranges of interest subject to the existence of an internally stabilising controller that guarantees robust performance with respect to these maximised weights. Thus, performance weights and a robustly stabilising controller are simultaneously synthesised by one algorithm in a systematic way. Two solution algorithms are given for the posed optimisation problem — one being pointwise in frequency and the other using state-space techniques. The latter solution eliminates all of the disadvantages of the pointwise approach and considerably enhances the benefits of this type of optimisation based weight selection. The resulting conceptually novel method for performing μ-synthesis robust performance based design is a valuable alternative to the standard D-K iterative procedure.

In the second part of the thesis, several steps of the standard $\mathscr{H}_\infty$ loop-shaping design procedure are combined into one optimisation problem that maximises the robust stability margin over the loop-shaping weights subject to constraints which ensure that the loop-shape and the singular values/condition numbers of the weights lie in pre-specified regions. In this framework, loop-shaping weights, which can be required to have either a diagonal or a non-diagonal structure, and a robustly stabilising controller are simultaneously synthesised by one algorithm systematically.

Correspondingly, the proposed algorithms greatly simplify the design of "good" performance weights and loop-shaping weights, and hence allow the designer to concentrate on more fundamental design issues. These algorithms also give an indication of what performance is achievable, although further research is required in that direction.

Keywords: weight selection, performance optimisation, μ-synthesis, $\mathscr{H}_\infty$ loop-shaping, robust performance, D-K iterations, $\mathscr{H}_\infty$-control, robust control.

Acknowledgements

This work is the conclusion of three years of research at the Department of Engineering of the University of Cambridge. Many people have helped me over the past three years and it is my great pleasure to take this opportunity to express my gratitude to them all.

Firstly, I would like to thank Dr Robert J Richards for allowing me the freedom to choose my research direction and for his encouragement. I would also like to thank Professor Keith Glover for his constant guidance and support throughout my studies here in Cambridge. I am especially grateful to both Dr Richards and Professor Glover who, finding me abandoned on their doorstep four years ago, took me in as one of their own. I thank also Dr Glenn Vinnicombe for several enlightening ideas and constructive comments which have helped to improve this research.

I am greatly indebted to Dr Michael Cantoni, who on very short notice would freely give time to discuss problems with me. Dr Cantoni has had a strong influence on my views of control theory and I shall certainly miss the numerous fruitful discussions we had in front of the white-board or a piece of paper. I would like to express my sincere gratitude to Dr Cantoni for his guidance and careful advice throughout my doctoral studies and for promptly proof-reading several drafts of this manuscript.

I am grateful to Dr George Papageorgiou and Mr Kelvin Halsey for proof-reading parts of this thesis. With their help, the number of typographical errors in this dissertation have been significantly reduced. Their much appreciated comments have especially improved Chapter 5.

I would also like to thank the members of the control group for providing an enjoyable working environment of mutual interest in each others' work in which many problems could be discussed and solved in a quick and informal manner. Thank you all.

This research has been supported financially by the Cambridge Commonwealth Trust, the Department of Engineering and the University of Cambridge. This help is also gratefully acknowledged.

On a more personal note, my family's love and support from afar has been consistent and hugely appreciated. Their phone calls brought rays of Maltese sunshine into the greyest of days.

Finally, but most importantly of all, my deepest and heartfelt gratitude, love, admiration and respect for Dr Laura Sbaffi, who has never lost faith in me and has given me endless patience, love and support. With her interest in my work, she has got me through the worst of times and with her beautiful smile has made everything interesting, successful and fun.

As required by University Statute, I hereby declare that this dissertation is not substantially the same as any that I have submitted for a degree or diploma or other qualification at any other university. This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration.

Alexander Lanzon
Cambridge
10 October 2000

Contents

Notation and Acronyms

Fields of Numbers

$\mathbb{R}$	real numbers
$\mathbb{R}_+$	strictly positive real numbers
$\overline{\mathbb{R}}_+$	non-negative real numbers
$\mathbb{R}^n$	real column vectors with n entries
$\mathbb{R}_+^n$	column vectors with each of the n entries belonging to $\mathbb{R}_+$
$\overline{\mathbb{R}}_+^n$	column vectors with each of the n entries belonging to $\overline{\mathbb{R}}_+$
$\mathbb{R}^{m\times n}$	real matrices of dimension $m \times n$
$\mathbb{Z}$	integer numbers
$\mathbb{Z}_+$	strictly positive integer numbers
j	the imaginary unit, i.e. $j = \sqrt{-1}$
$\mathbb{C}$	complex numbers
$\mathbb{C}_-$	open left-half plane
$\mathbb{C}_+$	open right-half plane
$\overline{\mathbb{C}}_+$	closed right-half plane
$\mathbb{C}^n$	complex column vectors with n entries
$\mathbb{C}^{m\times n}$	complex matrices of dimension $m \times n$

Relational Symbols

$\in$	belongs to
$\subset$	subset of
$\cup$	union
$:=$	defined by
$\approx$	approximately equal to
$<$	less than
$\leq$	less than or equal to

$\ll$	much less than
$>$	greater than
$\geq$	greater than or equal to
$\gg$	much greater than
$\mapsto$	maps to
$\rightarrow$	tends to
$\downarrow$	tends to from above
$\uparrow$	tends to from below
$\Rightarrow$	implies
$\Leftarrow$	is implied by
$\Leftrightarrow$	is equivalent to

Miscellaneous

$\exists$	there exists
$\forall$	for all
:	such that
$\square$	end of proof
$\Re(s)$	real part of $s \in \mathbb{C}$
$\Im(s)$	imaginary part of $s \in \mathbb{C}$
$\lfloor x \rfloor$	largest integer not exceeding $x \in \mathbb{R}$, known as the *floor* of x
(X, Y)	an ordered pair
$[P, C]$	standard feedback interconnection of plant P and controller C
$X^\star$	solution X to some optimisation problem
$x \in (a, b)$	$a < x < b$ where $a, x, b \in \mathbb{R}$
$x \in (a, b]$	$a < x \leq b$ where $a, x, b \in \mathbb{R}$
$x \in [a, b)$	$a \leq x < b$ where $a, x, b \in \mathbb{R}$
$x \in [a, b]$	$a \leq x \leq b$ where $a, x, b \in \mathbb{R}$
$\lim_{x \to a} f(x)$	$f(x)$ in the limit as x tends to a
$\lim_{x \downarrow a} f(x)$	$f(x)$ in the limit as x tends to a from above
$\min_{x \in \boldsymbol{\mathcal{X}}} f(x)$	minimum of the function $f(x)$ over $x \in \boldsymbol{\mathcal{X}}$
$\max_{x \in \boldsymbol{\mathcal{X}}} f(x)$	maximum of the function $f(x)$ over $x \in \boldsymbol{\mathcal{X}}$
$\inf_{x \in \boldsymbol{\mathcal{X}}} f(x)$	infimum of the function $f(x)$ over $x \in \boldsymbol{\mathcal{X}}$
$\sup_{x \in \boldsymbol{\mathcal{X}}} f(x)$	supremum of the function $f(x)$ over $x \in \boldsymbol{\mathcal{X}}$
ess $\sup_{x \in \boldsymbol{\mathcal{X}}} f(x)$	supremum of the function $f(x)$ over $x \in \boldsymbol{\mathcal{X}}$ omitting isolated points

Matrix Operations

0	zero matrix of compatible dimensions
I	identity matrix of compatible dimensions
I_n	identity matrix of dimension $n \times n$
A^T	transpose of matrix A
A^*	complex conjugate transpose of matrix A
$A_\perp$	orthogonal complement of matrix A
A^{-1}	inverse of matrix A
$A^\dagger$	pseudo-inverse (also known as Moore-Penrose inverse) of matrix A
A^{-T}	denotes $\left(A^{-1}\right)^T$ or equivalently $\left(A^T\right)^{-1}$
A^{-*}	denotes $\left(A^{-1}\right)^*$ or equivalently $(A^*)^{-1}$
$\det(A)$	determinant of matrix A
$\text{rank}(A)$	rank of matrix A
$\text{trace}(A)$	trace of matrix A
$\text{diag}(v)$	diagonal matrix with the elements of $v \in \mathbb{C}^n$ on the main diagonal
$\text{diag}(A_1, A_2, \ldots, A_n)$	block-diagonal matrix with matrices A_i on the main diagonal
$\text{diag}_{i=1}^n(A_i)$	block-diagonal matrix with matrices A_i on the main diagonal
$\text{vec}(A)$	a vector containing the columns of A stacked on top of each other
$A \otimes B$	Kronecker Product of matrices A and B
$A \oplus B$	Kronecker Sum of matrices A and B
$\mathcal{F}_l(P, Q)$	lower linear fractional transformation of matrices P and Q
$\mathcal{F}_u(P, Q)$	upper linear fractional transformation of matrices P and Q
$A > 0$	hermitian matrix $A = A^*$ with strictly positive eigenvalues
$A \geq 0$	hermitian matrix $A = A^*$ with non-negative eigenvalues
$A < 0$	hermitian matrix $A = A^*$ with strictly negative eigenvalues
$A \leq 0$	hermitian matrix $A = A^*$ with non-positive eigenvalues
$A < B$	denotes $(A - B) < 0$

Function Spaces

$\mathscr{L}_2$	space of square integrable functions on $j\mathbb{R}$ including ∞
$\mathscr{H}_2$	subspace of functions in $\mathscr{L}_2$ that are analytic in $\mathbb{C}_+$ and uniformly square integrable along $\Re(s) = \alpha$ for all $\alpha \in \mathbb{R}_+$
$\mathscr{L}_\infty$	space of functions that are bounded on $j\mathbb{R}$ including ∞

$\mathscr{H}_\infty$	subspace of functions in $\mathscr{L}_\infty$ that are analytic and bounded in $\mathbb{C}_+$
$\mathscr{R}$	space of real-rational functions
$\mathscr{R}^{m\times n}$	space of real-rational functions having n inputs and m outputs
prefix $\mathscr{R}$	subspace of real-rational functions, e.g. $\mathscr{RL}_2$, $\mathscr{RL}_\infty$

Operations on Systems

G^T	transpose (or dual) of real-rational system G, that is $G^T(s) = G(s)^T$
$G^\sim$	adjoint of real-rational system G, that is $G^\sim(s) = G^T(-s) = G(-s)^T$
$G(j\omega)^*$	complex congugate transpose of frequency-response function $G(j\omega)$ at each frequency ω, that is $G(j\omega)^* = G(-j\omega)^T$
G^{-1}	inverse of real-rational system G, that is $G^{-1}(s) = G(s)^{-1}$
G^{-T}	denotes $\left(G^{-1}\right)^T$ or equivalently $\left(G^T\right)^{-1}$
$G(j\omega)^{-*}$	denotes $\left(G(j\omega)^{-1}\right)^*$ or equivalently $\left(G(j\omega)^*\right)^{-1}$

Measures of Size

$\lambda_i(A)$	i-th eigenvalue of matrix A
$\overline{\lambda}(A)$	largest eigenvalue of matrix A (when all eigenvalues are real)
$\underline{\lambda}(A)$	smallest eigenvalue of matrix A (when all eigenvalues are real)
$\sigma_i(A)$	i-th singular value of matrix A
$\overline{\sigma}(A)$	largest singular value of matrix A
$\underline{\sigma}(A)$	smallest singular value of matrix A
$\kappa(A)$	condition number of matrix A
$\lvert x\rvert$	modulus (or magnitude) of $x \in \mathbb{C}$
$\lVert x\rVert$	Euclidean norm of $x \in \mathbb{C}^n$
$\lVert A\rVert_F$	Frobenius norm of matrix A
$\lVert G\rVert_2$	two-norm of $G \in \mathscr{RL}_2$
$\lVert G\rVert_\infty$	infinity-norm of $G \in \mathscr{RL}_\infty$
$\mu_{\mathbf{\Delta}}(M)$	structured singular value of matrix M with respect to some uncertainty structure $\mathbf{\Delta}$
$b(P,C)$	robust stability margin for plant P and controller C
$b_{opt}(P)$	largest robust stability margin for plant P

Shorthand Notation

$P + \{\cdot\}^T$	denotes the symmetric matrix $P + P^T$
$P + \{\cdot\}^*$	denotes the hermitian matrix $P + P^*$

$\begin{pmatrix} P & S \\ * & R \end{pmatrix}$	denotes the matrix $\begin{pmatrix} P & S \\ S^* & R \end{pmatrix}$
$\begin{pmatrix} P & * \\ S & R \end{pmatrix}$	denotes the matrix $\begin{pmatrix} P & S^* \\ S & R \end{pmatrix}$
$\left[\begin{array}{c\|c} A & B \\ \hline C & D \end{array}\right]$	shorthand for state-space realisation $C\,(sI - A)^{-1}\,B + D$

Acronyms

LHS	Left Hand Side
RHS	Right Hand Side
LHP	Left Half Plane
RHP	Right Half Plane
SISO	Single Input Single Output
MIMO	Multiple Input Multiple Output
LTI	Linear Time-Invariant
LPV	Linear Parameter-Varying
LFT	Linear Fractional Transformation
LMI	Linear Matrix Inequality

Chapter 1

Introduction

1.1 Background and Motivation

Central to the development of feedback control theory has been the notion of uncertainty. This arises in two forms: (a) discrepancy between the physical plant and the mathematical model used for controller design, and (b) unmeasured noises and disturbances that act on the physical plant. Feedback is used to desensitise the control system from the effect of both these types of uncertainty. Care must be exercised, however, as feedback in the presence of an uncertain plant can easily lead to instability if due consideration is not given to the way in which this uncertainty modifies the system behaviour.

Classical control techniques were developed with both these uncertainty types in mind. Graphical techniques capable of dealing with single-input single-output plants were the primary tool and quickly found wide use in practice. These allowed performance specifications to be met, and when used in conjunction with gain and phase margins, allowed the designer to account for uncertainties in an intuitive manner. Unfortunately, these techniques are somewhat ad-hoc and often require a lot of iteration and much intuition on the part of the designer. Also, they do not provide easy answers to fundamental questions such as "What is the achievable performance?". These techniques have also been found to be rather difficult to apply on complex plants, for example those having multiple unity-gain crossover frequencies, and they do not easily generalise to multivariable systems.

The classical control period gave way to the so-called Modern Control era which saw the development of optimisation techniques that were more able to deal with performance and existence issues. These design techniques are more systematic, with the designer only specifying the appropriate weights in the cost function to achieve the desired performance characteristics. Of these, the Linear Quadratic Gaussian technique (Anderson and Moore, 1989; Green and Limebeer, 1995; Zhou et al., 1996) proved to be very popular as it dealt with multivariable plants in an elegant way. This technique synthesises a controller which internally stabilises the feedback interconnection and minimises the variance of a chosen set of output signals, when noise of known statistical properties (usually assumed to be white noise) acts on the plant. These modern control techniques also found application in linear

time-varying systems and appeared to offer a systematic and single-step approach to the design of control systems. However, during this period, overemphasis on performance objectives at the expense of uncertainty considerations quickly led to a schism between practitioners and theoreticians. This was primarily because the controllers computed using these techniques would often perform inadequately in practise.

Not surprisingly, considerable research effort subsequently went into the development of design techniques which were based on optimisation principles but which allowed robustness properties to be built into the controller directly. The result of this effort is a comprehensive theory (Francis, 1987; Maciejowski, 1994; Green and Limebeer, 1995; Zhou et al., 1996) which has its origins in the seminal paper of Zames (1981) and which has come to be known as $\mathscr{H}_\infty$ control theory. The $\mathscr{H}_\infty$ control problem synthesises a controller which internally stabilises the feedback interconnection and minimises the $\mathscr{L}_2$ induced gain from exogenous inputs to regulated outputs. This induced gain has direct robustness interpretations in terms of the small gain theorem (Zames, 1966) and also satisfies the performance objective of minimising the energy in the output for the worst-case bounded-energy input. There now exist a number of elegant solutions to this problem using a wide variety of mathematical techniques. These range from the early operator-theoretic approaches (Francis, 1987) to the more recent state-space procedures (Doyle et al., 1989) and LMI techniques (Gahinet and Apkarian, 1994).

Modern robust control paradigms, such as μ-synthesis and $\mathscr{H}_\infty$ loop-shaping, have a lot to offer in that they provide systematic procedures for obtaining sensible controllers that meet performance objectives and guarantee robustness against model uncertainty and unmeasured disturbances. In μ-synthesis, for example, the designer specifies performance weights to reflect the desired closed-loop performance objectives in different frequency regions and a controller is synthesised to give robust performance guarantees. Similarly, in $\mathscr{H}_\infty$ loop-shaping, the designer specifies loop-shaping weights to reflect the desired performance and robustness objectives and a controller is then synthesised to give other types of robust performance guarantees. It is clear, however, that although these techniques are very systematic and give controllers which perform sensibly, fundamental questions such as "What is the achievable performance?" are not addressed since the success of these paradigms hinges strongly on the designer being able to specify performance weights or loop-shaping weights which meet the specifications. The design of these weights is non-trivial and may be time-consuming for complex plants.

Any progress with questions concerning achievable performance limits and the existence of satisfactory controllers is bound to involve some kind of optimisation theory. It is the opinion of the author that answering such existence questions is an important component of a good design methodology. One does not want to waste time trying to solve a problem that has no solution, nor does one want to accept specification compromises without knowing that these are necessary. The aim of this thesis is to develop a theoretical framework with which one may address these complex design questions in a systematic way. It is believed that in order to determine the achievable performance limits, simulta-

neous optimisation of (performance or loop-shaping) weights and controllers is necessary. With this view in mind, this thesis proposes sub-optimal algorithms which attempt to answer the above questions while ensuring that the robust performance properties of the standard approaches are still guaranteed. In doing so, powerful optimisation algorithms are developed which assist the designer in the selection of performance and loop-shaping weights.

1.2 Organisation of this Dissertation

This dissertation consists of six chapters and is organised as follows:

Chapter 2: Preliminaries
In this chapter, material is collected to facilitate understanding of the main components of this thesis. First, non-standard terms, quantities and operations are defined for the sake of unambiguity and relevant signal and system spaces are introduced. Then, μ-synthesis and $\mathscr{H}_\infty$ loop-shaping are motivated and quantities important in these frameworks are defined. Finally, some interesting mathematical results are collected which will be used throughout the dissertation. Some of the non-widely known results are proven for the sake of completeness.

Chapter 3: A Pointwise Algorithm for μ-Synthesis
A conceptually new approach to the μ-synthesis robust performance problem is proposed in this chapter. The optimisation problem posed maximises the performance weights with respect to a suitable cost function that captures the desired closed-loop performance. This maximisation of performance weights is limited by the fact that there must exist some internally stabilising controller that guarantees robust performance with respect to these maximised weights. Thus, performance weights and a robustly stabilising controller are simultaneously synthesised by one algorithm in a systematic way. The designer is only required to specify the plant set and an optimisation directionality. This directionality only appears in the cost function and reflects the desired closed-loop properties in particular frequency regions. It is pointed out that choosing this directionality is much easier than choosing the performance weights directly. Correspondingly, this approach greatly simplifies the often long and tedious process of designing "good" performance weights directly and gives an indication of what is the achievable performance. A pointwise in frequency solution to the posed optimisation problem is also developed in this chapter.

Chapter 4: A State-Space Algorithm for μ-Synthesis
Here, the optimisation problem posed in Chapter 3 is modified slightly to yield a new optimisation problem that admits a state-space solution without significantly compromising the properties of the original optimisation. The corresponding state-space solution eliminates the disadvantages of the

pointwise approach taken in Chapter 3 and considerably enhances the benefits of this type of optimisation based weight selection. The resulting conceptually novel method of performing μ-synthesis robust performance based design is a valuable alternative to the standard D-K iterative procedure.

Chapter 5: An Algorithm for $\mathscr{H}_\infty$ Loop-Shaping

In this chapter, several steps of the standard $\mathscr{H}_\infty$ loop-shaping design procedure are combined into one optimisation problem that maximises the robust stability margin over the loop-shaping weights subject to constraints which ensure that the loop-shape and the singular values/condition numbers of the weights lie in pre-specified regions. Thus, loop-shaping weights, which can be required to have either a diagonal or a non-diagonal structure, and a robustly stabilising controller are simultaneously synthesised by one algorithm in a systematic way. The designer is only required to specify the boundaries of these allowable regions and the optimisation problem automatically synthesises loop-shaping weights and a robustly stabilising controller which immediately satisfy the specifications and give an indication of what is really attainable. Correspondingly, this approach greatly simplifies the often long and tedious process of designing "good" loop-shaping weights directly and allows the designer to concentrate on more fundamental design issues.

Chapter 6: Conclusions

This chapter summarises the contributions of this thesis and outlines potential directions for future research.

Appendices A and B

In the appendices, proofs are given of results presented in the main chapters. Only original material is proved in these appendices, as well-known results are given in the chapters without proof (but with references). Most proofs have been moved to the appendices for the sake of continuity in the corresponding chapters.

Chapter 2

Preliminaries

In this chapter, basic mathematical tools and important background material are collected for use in subsequent chapters. This chapter is by no means self-contained, as it assumes that the reader is already familiar with various fundamental concepts. General references are cited which contain fuller expositions of these topics.

In Section 2.1, some matrix quantities and operations are defined. This is followed by formal definitions of function spaces in Section 2.2 and by state-space formulae in Section 2.3. Then, μ-synthesis and $\mathscr{H}_\infty$ loop-shaping are motivated in Sections 2.4 and 2.5. This chapter ends with a collection of mathematical results that will be extensively used in subsequent chapters. These results are on parahermitian functions in Section 2.6, from calculus in Section 2.7 and on LMIs in Section 2.8.

2.1 Linear Algebra

Some basic linear algebra (or more specifically, matrix) facts will reviewed in this section. A detailed treatment of this topic can be found in Horn and Johnson (1996) and Golub and Van Loan (1996).

A useful tool in matrix analysis is Singular Value Decomposition. This is because the singular values of a matrix are good measures of the "size" of the matrix and the corresponding singular vectors are good indicators of strong/weak input or output directions. Geometrically, the singular values of a matrix $A \in \mathbb{C}^{m\times n}$ are precisely the lengths of the semi-axes of the hyper-ellipsoid E defined by

$$E := \left\{ y \in \mathbb{C}^m \ : \ y = Ax, x \in \mathbb{C}^n, \|x\| = 1 \right\}.$$

The corresponding left (resp. right) singular vectors of A are the directions of y (resp. x) that achieve the same semi-axes of the hyper-ellipsoid. More formally, let σ_i be the i-th singular value of matrix A and let the vectors u_i and v_i be the corresponding i-th left and i-th right singular vectors respectively. Then, it is easy to verify that

$$\begin{aligned} Av_i &= \sigma_i u_i \\ A^* u_i &= \sigma_i v_i . \end{aligned}$$

These equations give

$$A^*Av_i = \sigma_i^2 v_i$$
$$AA^*u_i = \sigma_i^2 u_i.$$

Hence, σ_i^2 is an eigenvalue of A^*A or AA^*, v_i is the corresponding eigenvector of A^*A and u_i is the corresponding eigenvector of AA^*. Recall that the non-zero eigenvalues of A^*A are identical to the non-zero eigenvalues of AA^*. The first definition given here formally defines the singular values of a matrix A in terms of the eigenvalues of A^*A or AA^*.

Definition 2.1.1 *Given $A \in \mathbb{C}^{m\times n}$ with $m \geq n$,*

- *the i-th singular value of matrix A is given by*

$$\sigma_i(A) := \sqrt{\lambda_i(A^*A)}.$$

- *the largest singular value of matrix A is given by*

$$\overline{\sigma}(A) := \sqrt{\overline{\lambda}(A^*A)}.$$

- *the smallest singular value of matrix A is given by*

$$\underline{\sigma}(A) := \sqrt{\underline{\lambda}(A^*A)}.$$

If $A \in \mathbb{C}^{m\times n}$ is such that $m \leq n$, then A^*A is replaced by AA^* in the above definitions.

Next, the condition number of a matrix is defined. The condition number is a measure of the variation in gain of a matrix over different directions. It is always greater than or equal to unity and a large condition number means that the gain of the matrix is very different in different directions.

Definition 2.1.2 *The condition number of matrix A is defined by*

$$\kappa(A) := \frac{\overline{\sigma}(A)}{\underline{\sigma}(A)}.$$

The Moore-Penrose inverse (or pseudo-inverse) of a matrix A may be defined in terms of its singular value decomposition. This generalised inverse denoted by $A^\dagger$ always exists, is unique and satisfies the following conditions:

(i) $AA^\dagger A = A$,

(ii) $A^\dagger AA^\dagger = A^\dagger$,

(iii) $(AA^\dagger)^* = AA^\dagger$,

(iv) $(A^\dagger A)^* = A^\dagger A$.

Definition 2.1.3 *Given $A \in \mathbb{C}^{m\times n}$ with singular value decomposition $A = U \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} V^*$, where U and V are unitary matrices and Σ_r is a diagonal strictly positive real matrix, the pseudo-inverse of A is given by*

$$A^\dagger := V \begin{bmatrix} \Sigma_r^{-1} & 0 \\ 0 & 0 \end{bmatrix} U^*.$$

Now, the Frobenius norm of a matrix is defined. This norm is not an induced norm but has the very nice property that the square of the Frobenius norm of matrix A is equal to the sum of the squares of the singular values of A.

Definition 2.1.4 *The Frobenius norm of matrix A is defined by*

$$\|A\|_F := \sqrt{\text{trace}\,(A^*A)}.$$

The vector formed by stacking the columns of a matrix on top of each other into one long vector is defined next. It has important uses in conjunction with the Kronecker product and the Kronecker sum defined below.

Definition 2.1.5 *Let $A \in \mathbb{C}^{m\times n}$ be partitioned as $A = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix}$ with $a_i \in \mathbb{C}^m \ \ \forall i$. Then*

$$\text{vec}(A) := \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \in \mathbb{C}^{nm}.$$

The Kronecker product and Kronecker sum of two matrices are now defined. Interesting algebraic properties of the Kronecker product and sum, some of which are used in this dissertation, may be found in Brewer (1978).

Definition 2.1.6 *Let $A \in \mathbb{C}^{m\times n}$ and $B \in \mathbb{C}^{p\times q}$. Then the Kronecker product of A and B is defined by*

$$A \otimes B := \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix} \in \mathbb{C}^{mp\times nq}.$$

Definition 2.1.7 *Let $A \in \mathbb{C}^{n\times n}$ and $B \in \mathbb{C}^{m\times m}$. Then the Kronecker sum of A and B is defined by*

$$A \oplus B := (A \otimes I_m) + (I_n \otimes B) \in \mathbb{C}^{nm\times nm}.$$

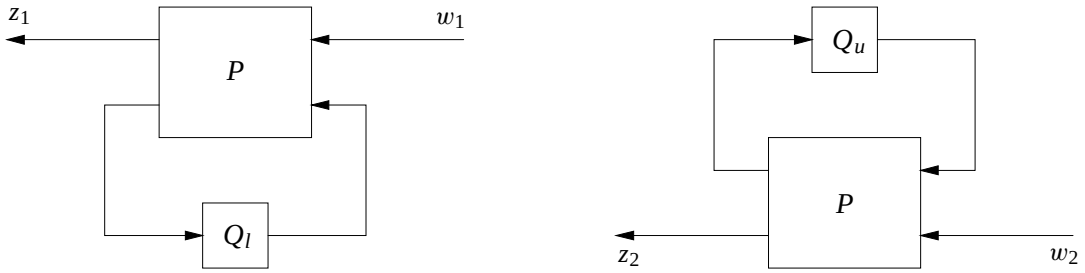


Figure 2.1: Diagrammatic representations of $\mathcal{F}_l(P, Q_l)$ and $\mathcal{F}_u(P, Q_u)$

A Linear Fractional Transformation is a matrix function which is usually used in control to formulate apparently different control problems into the same framework and hence to be able to treat all such problems in a similar way. Algebraic properties of LFTs are summarised in Doyle et al. (1991) and Zhou et al. (1996). The motivation for the terminologies "lower" and "upper" LFTs should be clear from the diagram representations of $\mathcal{F}_l(P, Q_l)$ and $\mathcal{F}_u(P, Q_u)$ given in Figure 2.1.

Definition 2.1.8 *Let* $P \in \mathbb{C}^{(p_1+p_2)\times(q_1+q_2)}$ *be partitioned as* $P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$ *with* $P_{11} \in \mathbb{C}^{p_1\times q_1}$ *and* $P_{22} \in \mathbb{C}^{p_2\times q_2}$. *Then, given* $Q_l \in \mathbb{C}^{q_2\times p_2}$ *and* $Q_u \in \mathbb{C}^{q_1\times p_1}$,

- *the lower LFT is defined by*

$$\mathcal{F}_l(P, Q_l) := P_{11} + P_{12}Q_l(I - P_{22}Q_l)^{-1}P_{21},$$

- *the upper LFT is defined by*

$$\mathcal{F}_u(P, Q_u) := P_{22} + P_{21}Q_u(I - P_{11}Q_u)^{-1}P_{12},$$

provided that the required inverses exist.

2.2 Function Spaces

This section defines the signal and system spaces used in this dissertation. Further details may be found in Rudin (1987) and Young (1988).

Definition 2.2.1 $\mathscr{R}$ *is a space of complex matrix valued functions defined on* $\mathbb{C}$ *and consists of all proper rational functions with real coefficients.*

Definition 2.2.2 $\mathscr{L}_2$ *is a Hilbert space of complex matrix valued functions defined on* $j\mathbb{R}$ *and consists of all functions* G *such that*

$$\int_{-\infty}^{\infty} \text{trace}\left[G(j\omega)^*G(j\omega)\right] d\omega \; < \infty.$$

The corresponding norm for this Hilbert space is defined by:

$$\|G\|_{\mathscr{L}_2} := \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} \text{trace}\left[G(j\omega)^* G(j\omega)\right] d\omega}.$$

Definition 2.2.3 *$\mathscr{H}_2$ is a Hardy space of complex matrix valued functions defined on $\mathbb{C}_+$ and consists of all functions G that are analytic in $\mathbb{C}_+$ and satisfy*

$$\sup_{\alpha>0} \int_{-\infty}^{\infty} \text{trace}\left[G(\alpha + j\omega)^* G(\alpha + j\omega)\right] d\omega \; < \infty.$$

The corresponding norm for this Hardy space is defined by:

$$\|G\|_{\mathscr{H}_2} := \sup_{\alpha>0} \left\{ \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} \text{trace}\left[G(\alpha + j\omega)^* G(\alpha + j\omega)\right] d\omega} \right\}.$$

The space $\mathscr{H}_2$ may also be regarded as a (closed) subspace of $\mathscr{L}_2$ since the boundary function defined by $G_b(j\omega) := \lim_{\alpha \downarrow 0} G(\alpha + j\omega)$ exists for almost all ω, $G_b \in \mathscr{L}_2$ and the mapping $G \mapsto G_b$ is linear, injective and satisfies (Francis, 1987)

$$\|G\|_{\mathscr{H}_2} = \|G_b\|_{\mathscr{L}_2}.$$

Hence, in the sequel, both the $\mathscr{L}_2$ and the $\mathscr{H}_2$ norms will be written simply as $\|G\|_2$.

Definition 2.2.4 *$\mathscr{L}_\infty$ is a Banach space of complex matrix valued functions defined on $j\mathbb{R}$ and consists of all functions G that are (essentially) bounded, with norm defined by:*

$$\|G\|_{\mathscr{L}_\infty} := \operatorname*{ess\,sup}_{\omega \in \mathbb{R}} \overline{\sigma}\left[G(j\omega)\right].$$

Definition 2.2.5 *$\mathscr{H}_\infty$ is a Hardy space of complex matrix valued functions defined on $\mathbb{C}_+$ and consists of all functions G that are analytic and bounded in $\mathbb{C}_+$. The corresponding norm for this Hardy space is defined by:*

$$\|G\|_{\mathscr{H}_\infty} := \sup_{s \in \mathbb{C}_+} \overline{\sigma}\left[G(s)\right].$$

The space $\mathscr{H}_\infty$ may also be regarded as a (closed) subspace of $\mathscr{L}_\infty$ since the boundary function defined by $G_b(j\omega) := \lim_{\alpha \downarrow 0} G(\alpha + j\omega)$ exists for almost all ω, $G_b \in \mathscr{L}_\infty$ and the mapping $G \mapsto G_b$ is linear, injective and satisfies (Boyd and Desoer, 1985)

$$\|G\|_{\mathscr{H}_\infty} = \|G_b\|_{\mathscr{L}_\infty}.$$

Hence, in the sequel, both the $\mathscr{L}_\infty$ and the $\mathscr{H}_\infty$ norms will be written simply as $\|G\|_\infty$.

2.3 State-Space Systems

Consider a real-rational system $G(s)$ with state-space realisation:

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right].$$

Definition 2.3.1 *The matrix A is said to be Hurwitz if all the eigenvalues of A are in $\mathbb{C}_-$.*

Definition 2.3.2 *(A, B) is said to be stabilisable if the matrix $\begin{bmatrix} A - \lambda I & B \end{bmatrix}$ has full row rank for all $\lambda \in \overline{\mathbb{C}}_+$, and (C, A) is said to be detectable if (A^T, C^T) is stabilisable.*

Definition 2.3.3 *The transpose (or dual) of G is given by*

$$G^T(s) = G(s)^T = \left[\begin{array}{c|c} A^T & C^T \\ \hline B^T & D^T \end{array}\right].$$

Definition 2.3.4 *The $\mathscr{L}_2$-adjoint of G is given by*

$$G^{\sim}(s) = G^T(-s) = \left[\begin{array}{c|c} -A^T & -C^T \\ \hline B^T & D^T \end{array}\right].$$

Definition 2.3.5 *The inverse of G, if it exists (i.e.if D is square and non-singular), is given by*

$$G^{-1}(s) = G(s)^{-1} = \left[\begin{array}{c|c} A - BD^{-1}C & -BD^{-1} \\ \hline D^{-1}C & D^{-1} \end{array}\right].$$

2.4 Motivation for μ-synthesis

As pointed out in Chapter 1, most control system design techniques rely on the use of a mathematical model and the quality of this model depends on how closely its responses match those of the physical plant. Since it is very difficult (or perhaps even impossible) to obtain a mathematical model that is identical to the physical plant, design techniques must be able to satisfy the stability and performance requirements in the face of uncertainty and unknown disturbances. The sources of this uncertainty in the mathematical model description may be unknown, in which case some general class of unstructured uncertainty representation is used. In other design situations, the sources of uncertainty in the model may be precisely known, in which case a structured uncertainty representation is used.

The basis of robust stability criteria for both unstructured and structured perturbations is the well-known "small gain theorem" introduced by Zames (1966). Before stating this theorem, the following two basic definitions are required.

Definition 2.4.1 *A feedback interconnection of real-rational proper transfer function matrices is said to be "well-posed" if all closed-loop transfer function matrices exist and are proper.*

Definition 2.4.2 *A feedback interconnection of real-rational proper transfer function matrices is said to be "internally stable" if the interconnection is well-posed and all closed-loop transfer function matrices belong to $\mathscr{RH}_\infty$.*

Internal stability is an important property of a feedback system, as it ensures that all internal signals are of bounded energy whenever the exogenous signals have bounded energy.

Several versions of the small gain theorem are available in the literature. The version presented here is sufficient to illustrate its importance and links well with the robust performance theorem for μ-analysis.

Theorem 2.4.1 (Small Gain Theorem) *Consider the feedback interconnection depicted in Figure 2.2. Suppose $M \in \mathscr{RH}_\infty$ and let $\gamma > 0$. Then this feedback interconnection is internally stable for all unstructured $\Delta \in \mathscr{RH}_\infty$ with $\|\Delta\|_\infty \leq 1/\gamma$ $(< 1/\gamma)$ if and only if $\|M\|_\infty < \gamma$ $(\leq \gamma)$.*

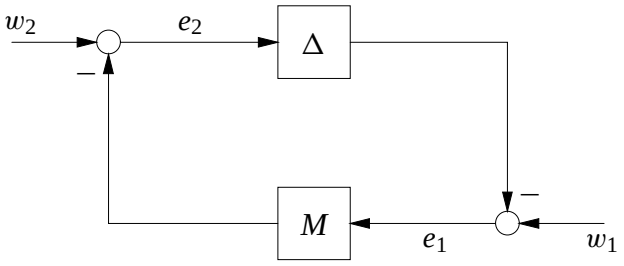


Figure 2.2: Standard feedback configuration

It can be shown that the above small gain condition is sufficient to guarantee internal stability even if Δ is a nonlinear time-varying "stable" operator, given an appropriately defined stability notion (Desoer and Vidyasagar, 1975).

Although the small gain theorem can be used directly to derive robust stability and performance results, it may be very conservative for systems with structured uncertainty. The exact stability and performance analysis for such systems requires the definition of another matrix function called the structured singular value, denoted by μ.

Definition 2.4.3 *Suppose $M \in \mathbb{C}^{n\times m}$ and let $\mathbf{\Delta} \subset \mathbb{C}^{m\times n}$ be some set which determines the uncertainty structure. Then the structured singular value of M with respect to uncertainty structure $\mathbf{\Delta}$ is defined by*

$$\mu_{\mathbf{\Delta}}(M) := \frac{1}{\min\left\{\overline{\sigma}(\Delta) \;:\; \Delta \in \mathbf{\Delta},\ \det(I - M\Delta) = 0\right\}}$$

unless no $\Delta \in \mathbf{\Delta}$ makes $I - M\Delta$ singular, in which case $\mu_{\mathbf{\Delta}}(M) := 0$.

Conceptually, the structured singular value is nothing but a straightforward generalisation of the singular values for constant matrices. To be more specific, consider again the robust stability problem depicted in Figure 2.2, where both $M(s)$ and $\Delta(s)$ are stable. An important question one might ask

is how large Δ can be (in the sense of $\|\Delta\|_\infty$) without destabilising the feedback system. Since the closed-loop poles are given by $\det(I - M(s)\Delta(s)) = 0$, the feedback system becomes unstable if $\det(I - M(s)\Delta(s)) = 0$ for some $s \in \overline{\mathbb{C}}_+$. Now, let $\alpha > 0$ be a sufficiently small number so that the closed-loop system is internally stable for all $\Delta \in \mathscr{RH}_\infty$ satisfying $\|\Delta\|_\infty < \alpha$. Then, start increasing the value α until the closed-loop system just becomes unstable. Denote the value of α which just makes the loop unstable by α_{max}. By the small gain theorem,

$$\|M\|_\infty := \sup_{\omega\in\mathbb{R}} \overline{\sigma}\left[M(j\omega)\right] = \frac{1}{\alpha_{max}}$$

if Δ is unstructured. Thus, at any fixed ω, $\overline{\sigma}\left[M(j\omega)\right]$ can be written as

$$\overline{\sigma}\left[M(j\omega)\right] = \frac{1}{\min\left\{\overline{\sigma}\left[\Delta(j\omega)\right] \;:\; \Delta \text{ is unstructured, } \det\left(I - M(j\omega)\Delta(j\omega)\right) = 0\right\}}$$

In other words, the reciprocal of the largest singular value of M is a measure of the smallest unstructured Δ that causes instability of the feedback system.

To quantify the smallest destabilising structured Δ, the concept of singular values has to be generalised. In view of the above characterisation, the structured singular value may be written as follows:

$$\mu_{\mathbf{\Delta}}\left[M(j\omega)\right] = \frac{1}{\min\left\{\overline{\sigma}\left[\Delta(j\omega)\right] \;:\; \Delta \text{ is structured, } \det\left(I - M(j\omega)\Delta(j\omega)\right) = 0\right\}}$$

where the set $\mathbf{\Delta}$ determines the structure of Δ. Then, the following theorem (Packard and Doyle, 1993) is a natural extension of the small gain theorem to the structured uncertainty case.

Theorem 2.4.2 *Consider the feedback interconnection depicted in Figure 2.2. Suppose $M \in \mathscr{RH}_\infty$ and let $\gamma > 0$. Then this feedback interconnection is internally stable for all structured $\Delta \in \mathscr{RH}_\infty$ with $\|\Delta\|_\infty \leq 1/\gamma$ $(< 1/\gamma)$ if and only if $\sup_{\omega\in\mathbb{R}} \mu_{\mathbf{\Delta}}\left[M(j\omega)\right] < \gamma$ $(\leq \gamma)$, where the set $\mathbf{\Delta}$ determines the structure of Δ.*

Hence, the peak value of the μ-plot of $M(j\omega)$ determines the size of the perturbations that the loop is robustly stable against.

Often, stability is not the only property of a closed-loop system that must be robust to perturbations. Typically, there are exogenous disturbances acting on the system which result in tracking and regulation errors. In most cases, long before the onset of instability, the closed-loop performance will degrade to the point of unacceptability, hence the need for robust performance analysis tests.

Although it is possible to analyse different problems individually, it is desirable to write all problems in a unified framework. The LFT definitions of Section 2.1 provide precisely this framework, as any linear interconnection of systems, inputs and outputs may be written into this form. Besides relieving the mathematical burden of dealing with specific problems individually, this unified approach also provides a framework in which robust stability and robust performance analysis and synthesis problems may be addressed. The following theorem (Packard and Doyle, 1993, Theorem 5.4) gives the required robust performance analysis test.

Theorem 2.4.3 (Robust Performance) *Consider the feedback interconnection depicted in Figure 2.3. Suppose $M \in \mathscr{RH}_\infty$ and let $\gamma > 0$. Then this feedback interconnection is internally stable and satisfies $\|\mathcal{F}_u(M, \Delta)\|_\infty < \gamma$ for all structured $\Delta \in \mathscr{RH}_\infty$ with $\|\Delta\|_\infty \leq 1/\gamma$ if and only if*

$$\sup_{\omega \in \mathbb{R}} \mu_{\mathbf{\Delta}_{TOT}}\left[M(j\omega)\right] < \gamma,$$

where $\mathbf{\Delta}_{TOT} := \left\{\text{diag}(\Delta, \Delta_P) : \Delta \in \mathbf{\Delta}, \Delta_P \in \mathbb{C}^{m \times n}\right\}$ and the set $\mathbf{\Delta}$ determines the structure of Δ.

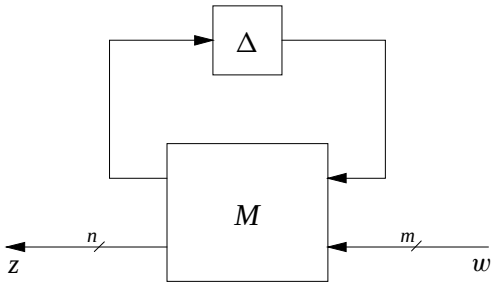


Figure 2.3: Performance analysis LFT framework

This is a remarkably important theorem which states that robust performance is equivalent to robust stability with an augmented uncertainty.

In order to apply the general structured singular value theory outlined above to control system design, the control problem has to first be recast into the appropriate LFT setting as shown in Figure 2.4. Here, G is the generalised plant which includes all system components that are given or specified, K

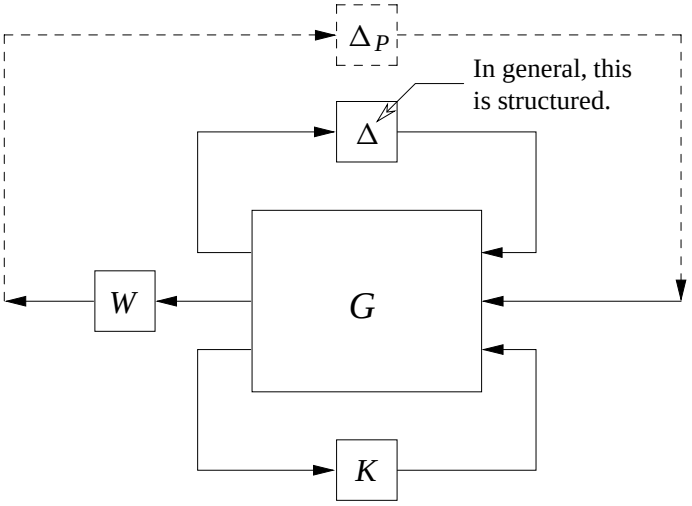


Figure 2.4: Performance synthesis LFT framework

is the controller which is to be synthesised, W represents the performance weights used to specify the relative importance of the magnitudes of the regulated variables, Δ is the structured uncertainty in the system and Δ_P is a fictitious uncertainty used only so as to view the robust performance problem as

an equivalent robust stability problem. It is clear that if G, W and K were all given, then they could be combined together to give the transfer function matrix M shown in Figure 2.3. Consequently, by Theorem 2.4.3, a necessary and sufficient condition for robust performance in the face of stable structured uncertainty that is norm bounded by unity is

$$\sup_{\omega\in\mathbb{R}} \mu_{\mathbf{\Delta}_{TOT}}\left[\begin{pmatrix} I & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(G(j\omega), K(j\omega)\right)\right] < 1.$$

In view of this, the general approach adopted in μ-synthesis is to seek to minimise the left-hand side of the above inequality over the set of internally stabilising controllers. Robust performance is then guaranteed if this minimisation achieves a value that is less than unity.

This synthesis problem is in general difficult as μ is not easily computable. Thus, a computable upper bound of μ has to be used instead. One such upper bound involves scalings that complicate slightly the synthesis procedure. However, this upper bound is desirable as it is equal to μ under some particular circumstances. When using this upper bound, alternate minimisation over these scalings and the set of internally stabilising controllers results in the so-called D-K iterative procedure which is probably the most common method used today in the design of robust controllers using μ-synthesis. The interested reader is referred to Zhou et al. (1996) for further details of the standard D-K iterative procedure.

2.5 Motivation for $\mathscr{H}_\infty$ Loop-Shaping

The $\mathscr{H}_\infty$ loop-shaping design procedure proposed by McFarlane and Glover (1992) is an effective method for designing robust controllers and has been successfully used in a variety of applications (see Papageorgiou and Glover (1999b) and references therein).

Desired closed-loop performance is specified by shaping the singular values of the scaled nominal plant P using pre- and post-compensators W_1 and W_2, as shown in Figure 2.5, to obtain the shaped plant $P_s = W_2 P W_1$. Since the notions of classical loop-shaping carry through, W_1 and W_2 are typically

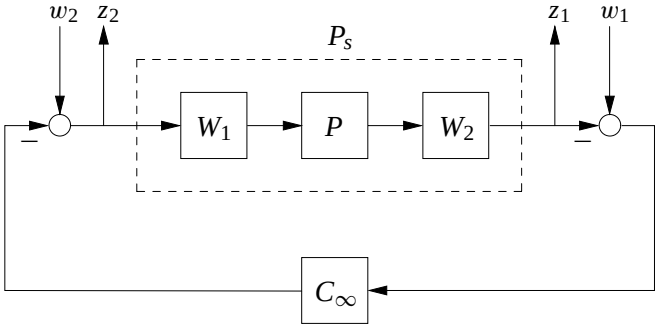


Figure 2.5: Typical $\mathscr{H}_\infty$ loop-shaping framework

chosen so that P_s has large gain at low frequency, small gain at high frequency and does not roll off

at a high rate near cross-over. However, in contrast with classical loop-shaping, the designer does not need to explicitly shape the phase of the nominal plant.

Loop-shaping weights W_1 and W_2 are usually designed in two stages. In the first stage, the desired loop-shape is determined. This usually involves translating time-response requirements and closed-loop performance specifications into the frequency domain. To do this, engineers largely rely on their intuition and their past experience with loop-shaping concepts. In the second stage, the designer selects loop-shaping weights W_1 and W_2 so that P_s has the desired loop-shape. Diagonal weights are often adequate to achieve the desired loop-shape (Hyde, 1995). However, some design examples have shown that diagonal weights do not work well for plants with strong cross-coupling between the channels. In such cases, non-diagonal weights are necessary which are of course more difficult to design.

Once a desired loop-shape is achieved, the optimal robust stability margin $b_{opt}(P_s)$ is computed. Glover and McFarlane (1989) showed that this optimal value can be explicitly calculated using a simple formula and gave a characterisation of the set of all internally stabilising controllers C_∞ that achieve some robust stability margin $b(P_s, C_\infty)$ less than this optimal value. Subsequently, McFarlane and Glover (1992) showed that the value $b_{opt}(P_s)$ is also a good indicator of the success of the loop-shaping stage. A large (resp. small) value of $b_{opt}(P_s)$ indicates compatibility (resp. incompatibility) between the specified loop-shape and closed-loop robust stability. A controller C for the scaled nominal plant P is finally obtained by pulling around the weights to obtain $C = W_1 C_\infty W_2$. A full tutorial on how to design robust controllers using the $\mathscr{H}_\infty$ loop-shaping design procedure can be found in Papageorgiou and Glover (1999b).

More formally, let the feedback interconnection of P_s and C_∞ shown in Figure 2.5 be denoted by $[P_s, C_\infty]$.

Definition 2.5.1 *The interconnection $[P_s, C_\infty]$ is said to be "well-posed" if each of the four transfer functions mapping $\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$ to $\begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$ exist.*

Note that $[P_s, C_\infty]$ is well-posed whenever $\det(I - C_\infty(\infty)P_s(\infty)) \neq 0$. In this case, these four transfer functions can be written as:

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} P_s \\ I \end{bmatrix} (I - C_\infty P_s)^{-1} \begin{bmatrix} -C_\infty & I \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}.$$

It is clear that the study of feedback interconnections that are not well-posed is meaningless. Henceforth, whenever a feedback interconnection is invoked, well-posedness will be implicitly assumed.

Definition 2.5.2 *The interconnection $[P_s, C_\infty]$ is said to be "internally stable" if it is well-posed and each of the four transfer functions mapping $\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$ to $\begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$ belongs to $\mathscr{RH}_\infty$.*

Next, the robust stability margin $b(P_s, C_\infty)$ and the optimal robust stability margin $b_{opt}(P_s)$ are defined.

Definition 2.5.3 *Given a plant P_s and a controller C_∞, the "robust stability margin" $b(P_s, C_\infty)$ is defined by:*

$$b(P_s, C_\infty) := \begin{cases} \left\| \begin{bmatrix} P_s \\ I \end{bmatrix} (I - C_\infty P_s)^{-1} \begin{bmatrix} -C_\infty & I \end{bmatrix} \right\|_\infty^{-1} & \text{if } [P_s, C_\infty] \text{ is internally stable,} \\ 0 & \text{otherwise.} \end{cases}$$

Definition 2.5.4 *Given a plant P_s, the largest value of the robust stability margin is defined by:*

$$b_{opt}(P_s) := \sup_{C_\infty} b(P_s, C_\infty).$$

It is shown in Glover and McFarlane (1989) that $b_{opt}(P_s) \leq 1$ for any P_s. Hence, the robust stability margin $b(P_s, C_\infty)$ is always some number between zero and one (both inclusive).

The following theorem (Zhou et al., 1996) relates the robust stability margin $b(P_s, C_\infty)$ to the size of the largest $\mathscr{H}_\infty$-norm bounded perturbation on the normalised coprime factors of the shaped plant P_s for which closed-loop stability is maintained.

Theorem 2.5.1 *Given a shaped plant P_s with a normalised right coprime factorisation over $\mathscr{RH}_\infty$ denoted by (N_s, M_s) and a controller C_∞, the following statements are equivalent for any $0 \leq \epsilon < 1$:*

(i) $b(P_s, C_\infty) > \epsilon$,

(ii) $[\tilde{P}_s, C_\infty]$ *is internally stable for all plants* $\tilde{P}_s \in \mathscr{P}$, *where*

$$\mathscr{P} := \left\{ \tilde{P}_s = (N_s + \Delta_{N_s})(M_s + \Delta_{M_s})^{-1} \text{ such that } \begin{bmatrix} \Delta_{N_s} \\ \Delta_{M_s} \end{bmatrix} \in \mathscr{RH}_\infty, \right.$$
$$\left. \left\| \begin{bmatrix} \Delta_{N_s} \\ \Delta_{M_s} \end{bmatrix} \right\|_\infty \leq \epsilon \text{ and } (M_s + \Delta_{M_s}) \text{ is invertible in } \mathscr{R} \right\}.$$

Recall that uncertainty in the coprime factors is the most general type of unstructured uncertainty as it allows for low/high frequency modelling errors and an unknown number of right-half plane poles and zeros (Zhou et al., 1996, Table 9.1). This theorem shows why the robust stability margin $b(P_s, C_\infty)$ is a good measure of the robustness of the closed-loop. The reader is also referred to Vinnicombe (2000) for further interesting properties of $b(P_s, C_\infty)$ and relations to the gap and ν-gap metrics.

It is also possible to motivate the use of $b(P_s, C_\infty)$ as a performance measure by noting that it bounds the gains of all closed-loop transfer functions between inputs and outputs at any point in the loop, as shown in the following theorem (Zhou et al., 1996). Tighter bounds are given in Vinnicombe (2000) which exploit the singular vector directions. However, the bounds presented in the following theorem are considerably simpler and hence easier to use in application.

Theorem 2.5.2 *Let P be the scaled nominal plant and let $C = W_1 C_\infty W_2$ be the associated controller obtained from the $\mathscr{H}_\infty$ loop-shaping design procedure (i.e. C_∞ is the robustly stabilising controller for the shaped plant $P_s = W_2 P W_1$). Then the size of each closed-loop transfer function matrix in the interconnection $[P, C]$ can be bounded frequency-by-frequency as follows:*

$$\begin{aligned}
\overline{\sigma}\Big[(I - CP)^{-1}\Big] &\leq \min\Big\{\gamma\overline{\sigma}(M_s)\,\kappa(W_1),\ 1 + \gamma\overline{\sigma}(\tilde{N}_s)\,\kappa(W_1)\Big\},\\
\overline{\sigma}\Big[(I - PC)^{-1}\Big] &\leq \min\Big\{\gamma\overline{\sigma}(\tilde{M}_s)\,\kappa(W_2),\ 1 + \gamma\overline{\sigma}(N_s)\,\kappa(W_2)\Big\},\\
\overline{\sigma}\Big[CP(I - CP)^{-1}\Big] &\leq \min\Big\{\gamma\overline{\sigma}(\tilde{N}_s)\,\kappa(W_1),\ 1 + \gamma\overline{\sigma}(M_s)\,\kappa(W_1)\Big\},\\
\overline{\sigma}\Big[PC(I - PC)^{-1}\Big] &\leq \min\Big\{\gamma\overline{\sigma}(N_s)\,\kappa(W_2),\ 1 + \gamma\overline{\sigma}(\tilde{M}_s)\,\kappa(W_2)\Big\},\\
\overline{\sigma}\Big[C(I - PC)^{-1}\Big] &\leq \gamma\overline{\sigma}(M_s)\,\overline{\sigma}(W_1)\,\overline{\sigma}(W_2),\\
\overline{\sigma}\Big[P(I - CP)^{-1}\Big] &\leq \frac{\gamma\overline{\sigma}(N_s)}{\underline{\sigma}(W_1)\,\underline{\sigma}(W_2)},
\end{aligned}$$

where

$$\begin{aligned}
\gamma &= \frac{1}{b(P_s, C_\infty)},\\
\overline{\sigma}(N_s) = \overline{\sigma}(\tilde{N}_s) &= \left[\frac{\overline{\sigma}(P_s)^2}{1 + \overline{\sigma}(P_s)^2}\right]^{1/2},\\
\overline{\sigma}(M_s) = \overline{\sigma}(\tilde{M}_s) &= \left[\frac{1}{1 + \underline{\sigma}(P_s)^2}\right]^{1/2},
\end{aligned}$$

and (N_s, M_s) (resp. $(\tilde{N}_s, \tilde{M}_s)$) is the normalised right (resp. left) coprime factorisation over $\mathscr{RH}_\infty$ of the shaped plant P_s.

It should be pointed out that all the above bounds are expressed in terms of $b(P_s, C_\infty)$, W_1, W_2 and P_s only. Thus, by specifying the sizes of these variables in different frequency regions, it is possible to indirectly constrain the size of each closed-loop transfer function matrix given in the above theorem.

Finally, the standard $\mathscr{H}_\infty$ loop-shaping design procedure of McFarlane and Glover (1992) is outlined and some justification is given as to why this design method is sensible.

The standard $\mathscr{H}_\infty$ loop-shaping design procedure

I. Scale the inputs and outputs of the nominal plant P_{nom} with pre- and post-diagonal scaling matrices S_1 and S_2 to give the scaled nominal plant $P = S_2 P_{nom} S_1$. This scaling is important so that differences between the units of each input or output channel are compensated for by the scaling matrices S_1 and S_2.

II. Shape the singular values of the scaled nominal plant P with frequency dependent pre- and post-compensators W_1 and W_2 such that the achieved loop-shape (i.e. the singular values of the shaped plant $P_s = W_2 P W_1$) satisfies the closed-loop performance requirements. It is important

to ensure that W_1 and W_2 are chosen such that the interconnection $W_2 P W_1$ contains no unstable hidden modes. Usually, W_1, W_1^{-1} and W_2, W_2^{-1} are restricted to belong to $\mathscr{RH}_\infty$.

III. Compute $b_{opt}(P_s)$. If $b_{opt}(P_s) \ll 0.3$, then it means that the designed loop-shape is incompatible with robust stability/performance and hence the designer should go back to Step II to design a "better" loop-shape[1].

IV. Synthesise a controller C_∞ that achieves a robust stability margin $b(P_s, C_\infty)$ that is slightly less than the computed $b_{opt}(P_s)$. The reason for this is that optimal controllers cannot be written in observer form, which is often desirable, and do not roll-off at high frequency. Design a command pre-filter (if required) and pull around the loop-shaping weights W_1, W_2 and the scaling matrices S_1, S_2.

V. Finally check time-simulations (for settling time, overshoot, actuator saturation, etc.) of the resulting closed-loop system to verify its robust performance and tune as necessary.

The theoretical basis for $\mathscr{H}_\infty$ loop-shaping is that C_∞ does not modify the loop-shape significantly at low and high frequencies if $b(P_s, C_\infty)$ is large, but it essentially shapes the phase of P_s around cross-over to improve robustness. Therefore, the designer can specify closed-loop performance objectives by shaping the singular values of the open-loop plant P with frequency dependent weights W_1 and W_2 and let the controller C_∞ take care of stability.

2.6 Results on Parahermitian Functions

The first lemma given in this section shows that parahermitian rational matrix functions can be rewritten with arbitrary (1,1)-block.

Lemma 2.6.1 *Let A, B, P, S, R be real matrices of compatible dimensions such that $P = P^T$, $R = R^T$ and $\lambda_i(A) \neq -\lambda_j(A)\ \forall i, j$. Define the parahermitian rational matrix function*

$$\Gamma(s) := \begin{bmatrix} B^T(-sI - A^T)^{-1} & I \end{bmatrix} \begin{bmatrix} P & S \\ S^T & R \end{bmatrix} \begin{bmatrix} (sI - A)^{-1}B \\ I \end{bmatrix}.$$

Then, given an arbitrary real matrix $\hat{P} = \hat{P}^T$ of the same dimensions as P, there exists a real matrix $\hat{S}$ of the same dimensions as S such that

$$\Gamma(s) = \begin{bmatrix} B^T(-sI - A^T)^{-1} & I \end{bmatrix} \begin{bmatrix} \hat{P} & \hat{S} \\ \hat{S}^T & R \end{bmatrix} \begin{bmatrix} (sI - A)^{-1}B \\ I \end{bmatrix}.$$

[1] Experience has shown that $b_{opt}(P_s) \geq 0.3$ is usually good enough, in the same way as a gain margin of 5.4dB and a phase margin of 35° are for a SISO design (Glover et al., 2000).

In fact, $\hat{S}$ is given by

$$\hat{S} = S + XB,$$

where the real matrix $X = X^T$ is the unique solution to the Lyapunov equation

$$XA + A^T X = (\hat{P} - P).$$

Proof Since $\lambda_i(A) + \lambda_j(A) \neq 0 \ \forall i, j$, the Lyapunov equation $XA + A^T X = (\hat{P} - P)$ has a unique solution (Zhou et al., 1996, Lemma 2.7). Furthermore, since A is real and $(\hat{P} - P)$ is real and symmetric, such a solution is real and symmetric. Now, note that $\Gamma(s)$ can be written in state-space form as:

$$\Gamma(s) = \left[\begin{array}{cc|c} A & 0 & B \\ -P & -A^T & -S \\ \hline S^T & B^T & R \end{array}\right].$$

Applying the similarity transformation $\begin{bmatrix} I & 0 \\ X & I \end{bmatrix}$ to this state-space realisation gives

$$\begin{aligned}
\Gamma(s) &= \left[\begin{array}{cc|c} A & 0 & B \\ -(XA + A^T X + P) & -A^T & -(S + XB) \\ \hline (S^T + B^T X) & B^T & R \end{array}\right] \\
&= \left[\begin{array}{cc|c} A & 0 & B \\ -\hat{P} & -A^T & -\hat{S} \\ \hline \hat{S}^T & B^T & R \end{array}\right] \\
&= \begin{bmatrix} B^T(-sI - A^T)^{-1} & I \end{bmatrix} \begin{bmatrix} \hat{P} & \hat{S} \\ \hat{S}^T & R \end{bmatrix} \begin{bmatrix} (sI - A)^{-1}B \\ I \end{bmatrix}.
\end{aligned}$$

□

As a consequence of Lemma 2.6.1, the following corollary states that a parahermitian rational matrix function can be decomposed into the sum of a stable transfer function matrix and its adjoint. Although the statement in this corollary assumes that A is Hurwitz, this is an over-restriction and a similar result can be proven (Francis, 1987) under a weaker assumption. However, this dissertation makes use of this corollary only in the situation when A is Hurwitz, and hence attention is limited to this case for ease of proof and notation.

Corollary 2.6.2 *Let A, B, P, S, R be real matrices of compatible dimensions such that $P = P^T$, $R = R^T$ and A is Hurwitz. Define the parahermitian rational matrix function*

$$\Gamma(s) := \begin{bmatrix} B^T(-sI - A^T)^{-1} & I \end{bmatrix} \begin{bmatrix} P & S \\ S^T & R \end{bmatrix} \begin{bmatrix} (sI - A)^{-1}B \\ I \end{bmatrix}.$$

Letting the real matrix $X = X^T$ be the unique solution to the Lyapunov equation $XA + A^T X + P = 0$, it follows that

$$\Gamma(s) = \left[\begin{array}{c|c} A & B \\ \hline S^T + B^T X & \frac{1}{2}R \end{array}\right] + \left[\begin{array}{c|c} A & B \\ \hline S^T + B^T X & \frac{1}{2}R \end{array}\right]^{\sim}.$$

Proof Selecting $\hat{P} = 0$ in Lemma 2.6.1 and letting X be the solution to $XA + A^T X + P = 0$ gives

$$\begin{aligned}\Gamma(s) &= \begin{bmatrix} B^T(-sI - A^T)^{-1} & I \end{bmatrix} \begin{bmatrix} 0 & S + XB \\ S^T + B^T X & R \end{bmatrix} \begin{bmatrix} (sI - A)^{-1}B \\ I \end{bmatrix} \\ &= \left[\begin{array}{c|c} A & B \\ \hline S^T + B^T X & \frac{1}{2}R \end{array}\right] + \left[\begin{array}{c|c} -A^T & -(S + XB) \\ \hline B^T & \frac{1}{2}R \end{array}\right] \\ &= \left[\begin{array}{c|c} A & B \\ \hline S^T + B^T X & \frac{1}{2}R \end{array}\right] + \left[\begin{array}{c|c} A & B \\ \hline S^T + B^T X & \frac{1}{2}R \end{array}\right]^{\sim}.\end{aligned}$$

□

The next lemma gives a complete parametrisation of frequency functions of the form $T(j\omega)^* T(j\omega)$ where T, $T^{-1} \in \mathscr{RH}_\infty$. Using 0 as the arbitrary (1,1)-block in this parametrisation reduces the number of potential decision variables in an eventual optimisation. This particular parametrisation will also turn out to be of crucial importance in the proof of Theorem 4.7.1.

Lemma 2.6.3 *Given $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ with A Hurwitz.*

(i) For every $C \in \mathbb{R}^{m\times n}$ and $D \in \mathbb{R}^{m\times m}$ such that $T(s) := \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \in \mathscr{RH}_\infty$ satisfies $T^{-1} \in \mathscr{RH}_\infty$, there exist $Q_{12} \in \mathbb{R}^{n\times m}$ and $Q_{22} = Q_{22}^T \in \mathbb{R}^{m\times m}$ such that

$$T(j\omega)^* T(j\omega) = \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix}^* \begin{bmatrix} 0 & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix} > 0 \quad \forall \omega \in \mathbb{R} \cup \{\infty\}.$$

(ii) For every $Q_{12} \in \mathbb{R}^{n\times m}$ and $Q_{22} = Q_{22}^T \in \mathbb{R}^{m\times m}$ such that

$$\begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix}^* \begin{bmatrix} 0 & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix} > 0 \quad \forall \omega \in \mathbb{R} \cup \{\infty\},$$

there exist $C \in \mathbb{R}^{m\times n}$ and $D \in \mathbb{R}^{m\times m}$ such that $T(s) := \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \in \mathscr{RH}_\infty$ satisfies $T^{-1} \in \mathscr{RH}_\infty$ and

$$\begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix}^* \begin{bmatrix} 0 & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix} = T(j\omega)^* T(j\omega) \quad \forall \omega \in \mathbb{R} \cup \{\infty\}.$$

Proof

(i) For any $C \in \mathbb{R}^{m\times n}$ and $D \in \mathbb{R}^{m\times m}$ such that $T(s) := \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \in \mathscr{RH}_\infty$ satisfies $T^{-1} \in \mathscr{RH}_\infty$, $T(j\omega)^*T(j\omega) > 0 \ \ \forall \omega \in \mathbb{R} \cup \{\infty\}$ and

$$T(j\omega)^*T(j\omega) = \begin{bmatrix} B^T(-j\omega I - A^T)^{-1} & I \end{bmatrix} \begin{bmatrix} C^TC & C^TD \\ D^TC & D^TD \end{bmatrix} \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix}$$

for all $\omega \in \mathbb{R} \cup \{\infty\}$. Now using Lemma 2.6.1 with $\hat{P} = 0$, there exist $Q_{12} \in \mathbb{R}^{n\times m}$ and $Q_{22} = Q_{22}^T \in \mathbb{R}^{m\times m}$ (in fact $Q_{22} = D^TD$) such that

$$T(j\omega)^*T(j\omega) = \begin{bmatrix} B^T(-j\omega I - A^T)^{-1} & I \end{bmatrix} \begin{bmatrix} 0 & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix} \ \ \forall \omega \in \mathbb{R} \cup \{\infty\}.$$

(ii) For any $Q_{12} \in \mathbb{R}^{n\times m}$ and $Q_{22} = Q_{22}^T \in \mathbb{R}^{m\times m}$ such that

$$\begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix}^* \begin{bmatrix} 0 & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix} > 0 \quad \forall \omega \in \mathbb{R} \cup \{\infty\},$$

it follows by (Zhou et al., 1996, Theorem 13.19 I(a)) and (Zhou et al., 1996, Corollary 13.20) that there exist a $C \in \mathbb{R}^{m\times n}$ and a $D \in \mathbb{R}^{m\times m}$ such that $T(s) := \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \in \mathscr{RH}_\infty$ satisfies $T^{-1} \in \mathscr{RH}_\infty$ and

$$\begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix}^* \begin{bmatrix} 0 & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix} = T(j\omega)^*T(j\omega) \quad \forall \omega \in \mathbb{R} \cup \{\infty\}.$$

□

2.7 A Result from Calculus

The lemma given in this section is a standard result from calculus. It will be used in the derivation of the cost function for one of the optimisation problems proposed in this thesis.

Lemma 2.7.1 *Suppose that A is Hurwitz. Then*

$$\int_{-\infty}^{\infty} (j\omega I - A)^{-1}\, d\omega \ = \ \pi I.$$

Proof Consider the contour $\mathscr{C}$ depicted in Figure 2.6 and note that since A is Hurwitz, $(sI - A)^{-1}$ is analytic for all $s \in \{s \in \mathbb{C} \ : \ \Re(s) > -\varepsilon\}$, where $\varepsilon \in \mathbb{R}_+$ is a sufficiently small number.

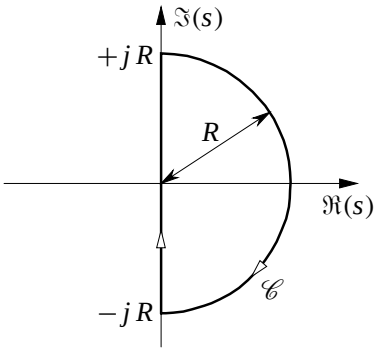


Figure 2.6: Contour $\mathscr{C}$ in the right half plane

Then, by Cauchy's Theorem (Rudin, 1987, Theorem 10.14)

$$\oint_{\mathscr{C}} (sI - A)^{-1}\, ds = 0 \quad \forall R.$$

Consequently,

$$\lim_{R\to\infty} \oint_{\mathscr{C}} (sI - A)^{-1}\, ds \;=\; 0$$

$$\Rightarrow \qquad \lim_{R\to\infty} \int_{-R}^{R} (j\omega I - A)^{-1} j\, d\omega \;+\; \lim_{R\to\infty} \int_{+\frac{\pi}{2}}^{-\frac{\pi}{2}} (Re^{j\theta} I - A)^{-1} j Re^{j\theta}\, d\theta \;=\; 0$$

$$\therefore \qquad \int_{-\infty}^{\infty} (j\omega I - A)^{-1}\, d\omega \;=\; \lim_{R\to\infty} \int_{-\frac{\pi}{2}}^{+\frac{\pi}{2}} (Re^{j\theta} I - A)^{-1} Re^{j\theta}\, d\theta$$

$$\text{so} \qquad \int_{-\infty}^{\infty} (j\omega I - A)^{-1}\, d\omega \;=\; \int_{-\frac{\pi}{2}}^{+\frac{\pi}{2}} \lim_{R\to\infty} \left(I - \frac{1}{R} e^{-j\theta} A \right)^{-1} d\theta$$

$$\text{hence} \qquad \int_{-\infty}^{\infty} (j\omega I - A)^{-1}\, d\omega \;=\; \int_{-\frac{\pi}{2}}^{+\frac{\pi}{2}} I\, d\theta \;=\; \pi I.$$

□

2.8 Linear Matrix Inequality Results

The computational methods proposed in this thesis heavily rely on convex optimisation problems and Linear Matrix Inequality (LMI) constraints. It is beyond the scope of this thesis to give a comprehensive treatment of this topic. However, the most fundamental results are collected here for completeness.

LMI constraints have the form

$$F(x) := F_0 + \sum_{i=1}^{m} x_i F_i > 0 \text{ (or } \geq 0),$$

where $x := [x_1 \;\; x_2 \;\; \ldots \;\; x_m]^T \in \mathbb{C}^m$ is the vector of decision variables and the hermitian matrices $F_i = F_i^* \in \mathbb{C}^{n\times n}$ are given. LMIs as the one above are convex constraints in x (i.e. the feasible set $\{x : F(x) > 0\}$ is a convex set). Multiple LMIs $F^1(x) > 0, \ldots, F^p(x) > 0$ may be combined into

a single LMI to give diag $\left[F^1(x), \ldots, F^p(x)\right] > 0$. As such, multiple coupled LMIs are treated as a single LMI with no additional difficulty.

A large variety of problems in control can be written as optimisation problems with LMI constraints. Hence, LMIs are now an area of considerable research interest. Analytic solutions to these linear matrix inequalities generally do not exist, but efficient numerical methods exist (which are coded in commercially available software packages) to find feasible solutions, if any. Boyd et al. (1994) stress that as long as analytic solutions are not required, reduction of an analysis or a synthesis problem to linear matrix inequalities is sufficient to have "solved" the problem. This is precisely the viewpoint taken in this thesis.

There are also quasi-convex optimisation problems (closely related to LMIs) which can be solved very efficiently. These problems usually take the form of generalised eigenvalue problems as follows:

$$\text{Minimise } \lambda \quad : \quad A(x) - \lambda B(x) < 0, \quad B(x) > 0, \quad C(x) > 0.$$

Here, $A(x)$, $B(x)$ and $C(x)$ are hermitian matrices that depend affinely on x.

In the remaining part of this section, some of the most important LMI results are collected for ease of reference in subsequent chapters of this dissertation.

Some nonlinear convex inequalities can be converted into LMI form using Schur Complements. The following lemma states such an equivalence (Horn and Johnson, 1996).

Lemma 2.8.1 (Schur Complement) *Suppose that* $P = P^* \in \mathbb{C}^{n\times n}$, $R = R^* \in \mathbb{C}^{m\times m}$ *and* $S \in \mathbb{C}^{n\times m}$. *Then*

$$R < 0 \;\; \textit{and} \;\; P - SR^{-1}S^* < 0 \quad \Leftrightarrow \quad \begin{bmatrix} P & S \\ S^* & R \end{bmatrix} < 0.$$

The next result by Kalman, Yakubovich and Popov states the equivalence between a frequency domain condition and an LMI. See (Willems, 1971, Theorems 3, 4) or (Rantzer, 1996, Theorem 1) for a proof.

Lemma 2.8.2 (Kalman-Yakubovich-Popov Lemma) *Given matrices* $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ *and* $Q = Q^T \in \mathbb{R}^{(n+m)\times(n+m)}$ *with* $\det(j\omega I - A) \neq 0 \; \forall \omega \in \mathbb{R}$ *and* (A, B) *controllable, the following two statements are equivalent:*

(i) For all $\omega \in \mathbb{R} \cup \{\infty\}$,

$$\begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix}^* Q \begin{bmatrix} (j\omega I - A)^{-1}B \\ I \end{bmatrix} \leq 0.$$

(ii) There exists a matrix $X = X^T \in \mathbb{R}^{n\times n}$ *such that*

$$Q + \begin{bmatrix} XA + A^TX & XB \\ B^TX & 0 \end{bmatrix} \leq 0.$$

The corresponding equivalence for strict inequalities holds even if (A, B) *is not controllable.*

The next lemma gives necessary and sufficient conditions for the existence of a matrix variable that enters an LMI in a particular form. See Gahinet and Apkarian (1994) or Boyd et al. (1994) for a proof.

Lemma 2.8.3 (Projection Lemma) *Suppose that* $Q = Q^T \in \mathbb{R}^{n\times n}$, $U \in \mathbb{R}^{p\times n}$ *and* $V \in \mathbb{R}^{q\times n}$. *Denote by* Ψ_U *and* Ψ_V *any matrices with columns that form bases for the null spaces of* U *and* V *respectively. Then there exists a* $\Phi \in \mathbb{R}^{p\times q}$ *such that*

$$Q + U^T \Phi V + V^T \Phi^T U < 0$$

if and only if

$$\Psi_U^T Q \Psi_U < 0 \;\; \textit{and} \;\; \Psi_V^T Q \Psi_V < 0.$$

The following lemma by Packard (1994) will be used to uncouple LMIs having matrix variables X and X^{-1}, which usually result from the application of the above Projection Lemma.

Lemma 2.8.4 *Suppose that* $P = P^T \in \mathbb{R}^{n\times n}$ *and* $Q = Q^T \in \mathbb{R}^{n\times n}$ *with* $P > 0$ *and* $Q > 0$. *Let* m *be a positive integer and let* ⧫ *denote "Don't Care" elements. Then there exist* $S \in \mathbb{R}^{n\times m}$ *and* $R = R^T \in \mathbb{R}^{m\times m}$ *such that*

$$X = \begin{bmatrix} P & S \\ S^T & R \end{bmatrix} > 0 \;\; \textit{satisfies} \;\; X^{-1} = \begin{bmatrix} Q & ⧫ \\ ⧫ & ⧫ \end{bmatrix}$$

if and only if

$$\begin{bmatrix} P & I_n \\ I_n & Q \end{bmatrix} \geq 0 \;\; \textit{and} \;\; \operatorname{rank}\left(P - Q^{-1}\right) \leq m.$$

Chapter 3

A Pointwise Algorithm for μ-Synthesis

3.1 Introduction

It is well known that the design of performance weights for $\mathscr{H}_\infty$-control and μ-synthesis problems is a non-trivial task. Usually, suitable performance weights are obtained via a long and tedious trial and error process based primarily on engineering judgement and intuition (Stoughton, 1990; Jovik and Lennartson, 1996). For simple single-input single-output control systems with only two or three performance outputs, trial and error can be manageable. However, this approach becomes increasingly complicated as the number of performance channels increases, since it may not be possible to choose the performance weights for each channel independently.

The focus in control systems design is often on tracking and disturbance rejection in the low-frequency region and robustness to unmodelled dynamics in the high-frequency region, although the problem specifications may in some cases determine otherwise. The former is usually captured by weighting the closed-loop sensitivity functions with weights that are large at low-frequencies, thereby ensuring that the sensitivity functions are small in this frequency region. The latter is usually captured by weighting the closed-loop complementary sensitivity functions with weights that are large at high-frequency, thereby ensuring that the complementary sensitivity functions are small in this frequency region. Note however that the crossover region is critical for both robustness and performance, and in this frequency region neither the sensitivity nor the complementary sensitivity functions are small (Kwakernaak, 1993). Thus, choosing the performance weights in this mid-frequency region to optimise the desired closed-loop performance while maintaining robust stability can be very difficult (Balas and Doyle, 1990).

The D-K iterative procedure of Doyle (1985) is probably the most popular method used in μ-synthesis to design robustly stabilising controllers. Other methods with different computational benefits have later been proposed, such as μ-K iterations in Lin et al. (1993), E-K iterations in Chang et al. (1994) and L-R iterations in Rotea and Iwasaki (1994). However, all these methods assume that the performance weights have already been chosen. Some authors have suggested "rules" for choos-

ing such performance weights for specific design problems (Stoughton, 1990; Jovik and Lennartson, 1996). However, all of this work heavily relies on the designer's experience and the final performance weights used are often the result of a long trial and error process.

In Fan and Tits (1992), a mathematical quantity (closely related to μ) was introduced to answer the question: "Determine the smallest α such that for any uncertainty bounded by unity, an $\mathscr{H}_\infty$ performance level of α is guaranteed". Although this may be considered as an initial step towards optimising robust performance (i.e. the determination of the smallest α) for a given uncertainty set, the value α is a constant bound over all frequencies and channel directions. In this chapter, the following more general problem is addressed: "Determine the largest performance weights (in some sense, at each frequency and channel direction) such that for any uncertainty bounded by unity, an $\mathscr{H}_\infty$ performance level of unity is guaranteed".

Consequently, a conceptually new approach to the robust performance μ-synthesis problem is proposed in this chapter. Performance weights, maximised with respect to a suitable cost function that captures the desired closed-loop performance, are synthesised simultaneously with an internally stabilising controller to immediately guarantee robust performance. This optimisation problem yields a closed-loop μ-curve that is as flat as possible across frequency and very close to unity, reflecting optimised robust performance. The designer is only required to specify the plant set and an optimisation directionality. This directionality only appears in the cost on the performance weights and reflects the desired closed-loop properties. Correspondingly, this approach greatly simplifies the often long and tedious process of designing "good" performance weights directly. Of course, a sensible plant set and directionality are still necessary for a sensible result.

3.2 Problem Formulation

Consider the LTI system depicted in Figure 3.1. This LFT framework is very general as any linear

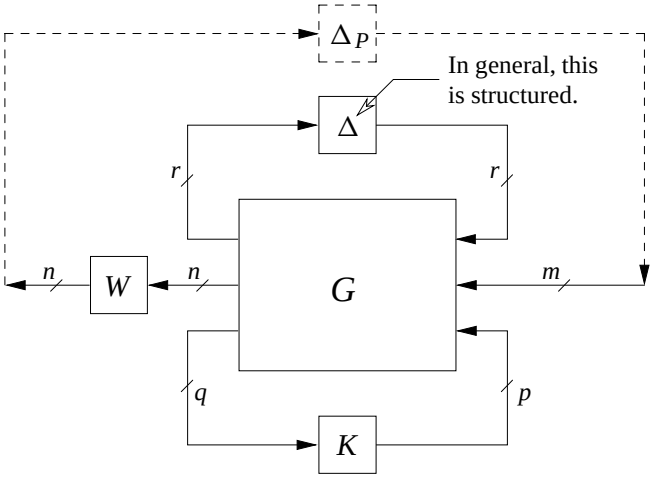


Figure 3.1: Typical μ-synthesis LFT framework

interconnection of systems, inputs, outputs and model uncertainties can be recast into this framework (Doyle et al., 1991). For notational convenience all uncertainty blocks, except those for performance, are assumed to be square. This can be done without loss of generality by adding dummy inputs and outputs (i.e. by padding the non-square uncertainty blocks and the corresponding rows/columns of the generalised plant G with zeros) (Helmersson, 1995, Section 5.2.1). Before formulating the problem of interest, some sets need to be defined.

Definition 3.2.1 *The sets of allowable perturbations are defined by:*

$$\begin{aligned}
\mathbf{\Delta} &:= \left\{ \operatorname*{diag}_{i=1}^{f} \left(I_{\alpha_i} \otimes \Delta_i \right) : \Delta_i \in \mathbb{C}^{\beta_i \times \beta_i}, \sum_{i=1}^{f} \alpha_i \beta_i = r \right\} \\
\mathbf{\Delta}_P &:= \left\{ \Delta_P \in \mathbb{C}^{m \times n} \right\} \\
\mathbf{\Delta}_{TOT} &:= \left\{ \operatorname{diag} \left(\Delta, \Delta_P \right) : \Delta \in \mathbf{\Delta}, \Delta_P \in \mathbf{\Delta}_P \right\} \\
\mathbf{B\Delta}^{TF} &:= \left\{ \Delta(s) \in \mathscr{RH}_\infty : \Delta(s_o) \in \mathbf{\Delta} \ \forall s_o \in \overline{\mathbb{C}}_+, \|\Delta\|_\infty \leq 1 \right\} \\
\mathbf{B\Delta}_P^{TF} &:= \left\{ \Delta_P(s) \in \mathscr{RH}_\infty : \Delta_P(s_o) \in \mathbf{\Delta}_P \ \forall s_o \in \overline{\mathbb{C}}_+, \|\Delta_P\|_\infty \leq 1 \right\}.
\end{aligned}$$

Here, α_i denotes the number of times that uncertainty block Δ_i is repeated, the prefix **B** *denotes a "ball" of uncertainty and the superscript TF denotes a set of transfer function matrices.*

It appears that the results presented in this thesis extend to the case where the set $\mathbf{\Delta}_P$ has a block-diagonal structure and/or to the case where the uncertainty sets defined above have mixed real and complex uncertainty. This, however, will not be done here in the interest of simpler notation. If the set $\mathbf{\Delta}_P$ is defined to have a block-diagonal structure, then the framework introduced in this chapter also allows for optimisation of specific input uncertainty weights associated with full uncertainty blocks and for specific channel-to-channel performance gain constraints.

Definition 3.2.2 *The set of diagonal complex matrices is defined by:*

$$\mathbf{\Lambda} := \left\{ \operatorname*{diag}_{i=1}^{n} \left(\ell_i \right) : \ell_i \in \mathbb{C} \right\}.$$

Definition 3.2.3 *The scaling sets $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{D}}^{TF}$ are defined by:*

$$\begin{aligned}
\boldsymbol{\mathcal{D}} &:= \left\{ D = \operatorname*{diag}_{i=1}^{f} \left(D_i \otimes I_{\beta_i} \right) : \det D \neq 0, D_i \in \mathbb{C}^{\alpha_i \times \alpha_i}, \sum_{i=1}^{f} \alpha_i \beta_i = r \right\} \\
\boldsymbol{\mathcal{D}}^{TF} &:= \left\{ D(s) \in \mathscr{RH}_\infty : D(s)^{-1} \in \mathscr{RH}_\infty, D(s_o) \in \boldsymbol{\mathcal{D}} \ \forall s_o \in \overline{\mathbb{C}}_+ \right\}.
\end{aligned}$$

Note that (both) these sets commute with $\mathbf{\Delta}$ and $\mathbf{B\Delta}^{TF}$.

Definition 3.2.4 *The sets of performance weights and directionality matrices are defined by:*

$$\begin{aligned}
\boldsymbol{\mathcal{W}}^{TF} &:= \left\{ W(s) \in \mathscr{RH}_\infty : W(s)^{-1} \in \mathscr{RH}_\infty, W(s_o) \in \mathbf{\Lambda} \ \forall s_o \in \overline{\mathbb{C}}_+ \right\} \\
\boldsymbol{\Upsilon}^{TF} &:= \left\{ \Upsilon(s) \in \mathscr{RH}_\infty : \Upsilon(s_o) \in \mathbf{\Lambda} \ \forall s_o \in \overline{\mathbb{C}}_+ \right\}.
\end{aligned}$$

Note that these two sets commute with $\mathbf{\Lambda}$.

Definition 3.2.5 *Given a generalised plant*

$$G(s) = \left[\begin{array}{c|cc:c} A & B_1 & B_2 & B_3 \\ \hline C_1 & D_{11} & D_{12} & D_{13} \\ C_2 & D_{21} & D_{22} & D_{23} \\ \hdashline C_3 & D_{31} & D_{32} & D_{33} \end{array}\right]$$

partitioned consistently with Figure 3.1, let the term "Standard Assumptions" refer to:

(A1) (A, B_3) *is stabilisable and* (C_3, A) *is detectable,*

(A2) $D_{33} = 0$.

Assumption (A1) is necessary and sufficient for the existence of an internally stabilising output-feedback controller (Green and Limebeer, 1995, Appendix A.4), whereas assumption (A2) incurs no loss of generality but considerably simplifies calculations. If $D_{33} \neq 0$, one may restore the $D_{33} = 0$ situation by a loop-shifting argument that absorbs D_{33} into $K(s)$ as described in Glover and Doyle (1988) or Safonov and Limebeer (1988).

Definition 3.2.6 *Given a generalised plant G, the set of internally stabilising output-feedback controllers $K \in \mathscr{R}^{p\times q}$ for the LFT interconnection $\mathcal{F}_l(G, K)$ is denoted by $\boldsymbol{\mathcal{K}}_G^{TF}$.*

Then, in Figure 3.1:

(i) G denotes the LTI generalised plant. Before constructing such a generalised plant, the designer would have already decided what "plant set" to design for. This "plant set" is determined from the nominal plant model, the structure, type and size of the uncertainty and any 'a priori' knowledge about the frequency content of the exogenous signals. Here it is assumed that all of this has already been done and $G(s)$ is given. See (Balas et al., 1996; Zhou et al., 1996; Skogestad and Postlethwaite, 1998) for a detailed explanation on how to construct such a generalised plant.

(ii) $K \in \boldsymbol{\mathcal{K}}_G^{TF}$ denotes an LTI controller which is to be designed. Besides being internally stabilising, $K(s)$ must also achieve the required closed-loop μ-value.

(iii) $W \in \boldsymbol{\mathcal{W}}^{TF}$ is a stable minimum-phase invertible (and hence bi-proper) diagonal transfer function matrix containing the performance weights. It is required to find the "biggest" $W(s)$, in some sense, such that some constraints are not violated.

(iv) $\Delta \in \mathbf{B}\boldsymbol{\Delta}^{TF}$ is the structured/unstructured uncertainty in the system. Thus, Δ is an unknown stable LTI system which has some block-diagonal structure and satisfies $\|\Delta\|_\infty \leq 1$.

(v) $\Delta_P \in \mathbf{B}\boldsymbol{\Delta}_P^{TF}$ is the "performance uncertainty". This uncertainty is fictitious and is only used to transform the robust performance problem into an equivalent robust stability problem. Hence, Δ_P is an unknown stable LTI system that satisfies $\|\Delta_P\|_\infty \leq 1$.

Given an internally stabilising controller $K \in \mathcal{K}_G^{TF}$ and a performance weight $W \in \mathcal{W}^{TF}$, it follows from Robust Performance Theorem 2.4.3 that robust performance is achieved for the setup of Figure 3.1 for all $\Delta \in \mathbf{B}\mathbf{\Delta}^{TF}$ if and only if

$$\sup_{\omega} \mu_{\mathbf{\Delta}_{TOT}} \left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(G(j\omega), K(j\omega)\right) \right] < 1.$$

Moreover, it is required to capture the size of the performance weights in some way.

Definition 3.2.7 *Given a system $P \in \mathcal{RH}_\infty$ and a frequency range $[\omega_L, \omega_H]$ with $\omega_L > 0$ and $\omega_H < \infty$, let $\|P\|_{[\omega_L,\omega_H]}$ be defined by:*

$$\|P\|_{[\omega_L,\omega_H]} := \sqrt{\int_{\log_{10}\omega_L}^{\log_{10}\omega_H} \|P(j\omega)\|_F^2 \; d(\log_{10}\omega)}.$$

This object is a semi-norm as it satisfies positivity, homogeneity and the triangular inequality but not positive-definiteness, and in some sense it is a measure of the size of $P(j\omega)$ in the frequency range $[\omega_L, \omega_H]$. Logarithmic frequency is chosen as the variable of integration so that $\|P\|_{[\omega_L,\omega_H]}$ has a direct interpretation when the singular values of P are plotted on a Bode diagram (recall that the square of the Frobenius norm of a matrix is equal to the sum of the squares of all singular values of that matrix).

Now consider the following optimisation problem for a given generalised plant $G(s)$ satisfying the standard assumptions stated in Definition 3.2.5, a given frequency range $[\omega_L, \omega_H]$ and an 'a priori' chosen directionality transfer function matrix $\Upsilon(s) \in \mathbf{\Upsilon}^{TF}$:

$$\max_{W \in \mathcal{W}^{TF}} \frac{1}{\left\|\Upsilon W^{-1}\right\|_{[\omega_L,\omega_H]}} \quad : \quad \min_{K \in \mathcal{K}_G^{TF}} \sup_{\omega} \mu_{\mathbf{\Delta}_{TOT}} \left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(G(j\omega), K(j\omega)\right) \right] < 1. \quad (3.1)$$

Here, $[\omega_L, \omega_H]$ is the frequency range where maximisation of the performance weight $W(s)$ is required. This frequency range should be chosen sensibly and a good rule-of-thumb is to consider two or three decades below and above the required closed-loop bandwidth. This ensures that all the closed-loop dynamics are captured by the above optimisation problem.

Some justification will now be given as to why this is a sensible optimisation problem to consider. First observe that the square of the cost function may be expressed as:

$$\frac{1}{\left\|\Upsilon W^{-1}\right\|_{[\omega_L,\omega_H]}^2} = \frac{1}{\displaystyle\int_{\log_{10}\omega_L}^{\log_{10}\omega_H} \left\|\Upsilon(j\omega)W(j\omega)^{-1}\right\|_F^2 \; d(\log_{10}\omega)} = \frac{1}{\displaystyle\int_{\log_{10}\omega_L}^{\log_{10}\omega_H} \sum_{i=1}^{n} \frac{1}{\left|\frac{w_i(j\omega)}{\upsilon_i(j\omega)}\right|^2} \; d(\log_{10}\omega)},$$

where $w_i(j\omega)$ (resp. $\upsilon_i(j\omega)$) is the i-th diagonal element of $W(j\omega)$ (resp. $\Upsilon(j\omega)$). From this decomposition, it is clear that the cost function $1 \Big/ \left\|\Upsilon W^{-1}\right\|_{[\omega_L,\omega_H]}$ is a cumulative measure of the frequency dependent size of the performance weights $w_i(j\omega)$ in the frequency range $[\omega_L, \omega_H]$. Each performance

weight $w_i(j\omega)$ is weighted differently across frequency in this cost function due to the directionality factors $\upsilon_i(j\omega)$. Thus, the directionality matrix $\Upsilon(s) \in \mathbf{\Upsilon}^{TF}$ may be chosen by the designer so as to direct the maximisation as desired. In fact, $\upsilon_i(j\omega)$ will be chosen large (resp. small) where the corresponding performance weight $w_i(j\omega)$ is required to be large (resp. small). Since maximisation will only take place in the frequency range $[\omega_L, \omega_H]$, $\Upsilon(j\omega)$ is only relevant in this frequency range.

This however does not make $\Upsilon(j\omega)$ a substitute for the performance weight $W(j\omega)$, as $\Upsilon(j\omega)$ only captures the desired directionality of the optimisation. The absolute size of each $\upsilon_i(j\omega)$ is completely irrelevant as this will only affect the value of the cost associated with the above optimisation problem. Only the shape across frequency and the relative sizes amongst the different diagonal entries of $\Upsilon(j\omega)$ are important. Furthermore, incompatible directionalities can never be specified, unlike directly specifying the performance weights. This is because the performance weights given by the above optimisation problem must always be feasible to the optimisation's constraint and hence will always satisfy $\mu < 1$. Sensible choice of $\Upsilon(j\omega)$ is of course still necessary (this is however much easier than choosing the actual performance weights) in order to obtain a controller which performs sensibly and satisfies reasonable stability/performance requirements (e.g. small sensitivity at low frequency and small complementary sensitivity at high frequency).

The constraint in optimisation problem (3.1) ensures that maximisation of the performance weight $W(s)$ is limited by the fact that there must exist some internally stabilising controller $K(s)$ which achieves robust performance for all uncertainty $\Delta \in \mathbf{B\Delta}^{TF}$.

3.3 Rewriting the Optimisation Problem

In this section, optimisation problem (3.1) is rewritten in a form which is more suitable for subsequent synthesis steps. This rewriting will make use of the following well-known theorem and corollary (Green and Limebeer, 1995; Zhou et al., 1996).

Theorem 3.3.1 (Youla Parametrisation) *Given a generalised plant G satisfying the standard assumptions stated in Definition 3.2.5, let F and L be such that $A+B_3F$ and $A+LC_3$ are Hurwitz and define*

$$J := \left[\begin{array}{c|cc} A+B_3F+LC_3 & -L & B_3 \\ \hline F & 0 & I \\ -C_3 & I & 0 \end{array}\right].$$

Then the set $\mathcal{K}_G^{TF}$ is parametrised by

$$\left\{K = \mathcal{F}_l(J, Q) \;\; : \;\; Q \in \mathcal{RH}_\infty\right\}.$$

Observe that the mapping from $Q \in \mathcal{RH}_\infty$ to $K \in \mathcal{K}_G^{TF}$ is bijective.

Corollary 3.3.2 *The set of all closed-loop transfer function matrices $\mathcal{F}_l(G,K)$ achievable by an internally stabilising controller $K \in \mathcal{K}_G^{TF}$ is parametrised by*

$$\left\{ \mathcal{F}_l(T,Q) = \begin{pmatrix} T_{11}+T_{13}QT_{31} & T_{12}+T_{13}QT_{32} \\ T_{21}+T_{23}QT_{31} & T_{22}+T_{23}QT_{32} \end{pmatrix} \;:\; Q \in \mathcal{RH}_\infty \right\},$$

where T is given by

$$T = \left[\begin{array}{cc:c} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ \hdashline T_{31} & T_{32} & 0 \end{array}\right] = \left[\begin{array}{cc:cc:c} A+B_3F & -B_3F & B_1 & B_2 & B_3 \\ 0 & A+LC_3 & B_1+LD_{31} & B_2+LD_{32} & 0 \\ \hline C_1+D_{13}F & -D_{13}F & D_{11} & D_{12} & D_{13} \\ C_2+D_{23}F & -D_{23}F & D_{21} & D_{22} & D_{23} \\ \hdashline 0 & C_3 & D_{31} & D_{32} & 0 \end{array}\right].$$

Note that all transfer function matrices T_{ij} are stable and that the parametrisation $\mathcal{F}_l(T,Q)$ is affine in the parameter Q. Moreover, since G is given at the beginning of the problem, T can be computed prior to evaluating the optimisation.

Consequently, optimisation problem (3.1) can equivalently be rewritten as:

$$\max_{W \in \mathcal{W}^{TF}} \frac{1}{\left\| \Upsilon W^{-1} \right\|_{[\omega_L,\omega_H]}} \quad : \quad \min_{Q \in \mathcal{RH}_\infty} \sup_{\omega} \mu_{\mathbf{\Delta}_{TOT}} \left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l(T(j\omega), Q(j\omega)) \right] < 1.$$

Furthermore, since only the arguments of the optimisation are of interest, this optimisation problem may be rewritten as:

$$\begin{gathered} \min_{W \in \mathcal{W}^{TF}} \left\| \Upsilon W^{-1} \right\|^2_{[\omega_L,\omega_H]} \\ \text{such that} \\ \exists\, Q \in \mathcal{RH}_\infty \text{ satisfying } \mu_{\mathbf{\Delta}_{TOT}} \left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l(T(j\omega), Q(j\omega)) \right] < 1 \;\; \forall \omega. \end{gathered} \tag{3.2}$$

This optimisation problem is however non-convex (due to the μ constraint) and hence its solution is not easily computable.

3.4 A Tractable Reformulation

In this section, each part of optimisation (3.2) will be investigated separately and a computationally tractable optimisation problem with tighter (i.e. more restrictive) constraints will be derived. The derivation is divided into several sub-sections for clarity.

3.4.1 Rewriting the Cost Function

In this sub-section, the cost function $\left\| \Upsilon W^{-1} \right\|^2_{[\omega_L,\omega_H]}$ is rewritten into a form more suitable for subsequent operations. To this end, first define the following sets:

Definition 3.4.1 *The sets of strictly-positive and non-negative vector valued functions are defined by*

$$\boldsymbol{\mathcal{V}} := \left\{ f : \mathbb{R} \mapsto \mathbb{R}_+^n \right\},$$
$$\overline{\boldsymbol{\mathcal{V}}} := \left\{ f : \mathbb{R} \mapsto \overline{\mathbb{R}}_+^n \right\}.$$

Then, for ease of notation, define the following vector functions:

$$v_{_W}(\omega) := \left[W(j\omega)^* W(j\omega)\right]^{-1} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} |w_1(j\omega)|^{-2} \\ |w_2(j\omega)|^{-2} \\ \vdots \\ |w_n(j\omega)|^{-2} \end{bmatrix} \in \boldsymbol{\mathcal{V}}, \tag{3.3}$$

$$\text{and} \quad v_{_\Upsilon}(\omega) := \quad \left[\Upsilon(j\omega)^* \Upsilon(j\omega)\right] \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} |\upsilon_1(j\omega)|^2 \\ |\upsilon_2(j\omega)|^2 \\ \vdots \\ |\upsilon_n(j\omega)|^2 \end{bmatrix} \in \overline{\boldsymbol{\mathcal{V}}}. \tag{3.4}$$

Using this notation, it is easy to see that

$$\begin{aligned} \left\| \Upsilon W^{-1} \right\|^2_{[\omega_{_L}, \omega_{_H}]} &= \int_{\log_{10}\omega_{_L}}^{\log_{10}\omega_{_H}} \text{trace}\left(W(j\omega)^{-*} \Upsilon(j\omega)^* \Upsilon(j\omega) W(j\omega)^{-1} \right) d(\log_{10}\omega) \\ &= \int_{\log_{10}\omega_{_L}}^{\log_{10}\omega_{_H}} \text{trace}\left(\Upsilon(j\omega)^* \Upsilon(j\omega) W(j\omega)^{-1} W(j\omega)^{-*} \right) d(\log_{10}\omega) \\ &= \int_{\log_{10}\omega_{_L}}^{\log_{10}\omega_{_H}} \text{trace}\left(\left[\Upsilon(j\omega)^* \Upsilon(j\omega)\right] \left[W(j\omega)^* W(j\omega)\right]^{-1} \right) d(\log_{10}\omega) \\ &= \int_{\log_{10}\omega_{_L}}^{\log_{10}\omega_{_H}} v_{_\Upsilon}(\omega)^T v_{_W}(\omega)\, d(\log_{10}\omega). \end{aligned} \tag{3.5}$$

The cost function in this last form is more suitable for subsequent operations as will be seen later. Note that $v_{_\Upsilon}(\omega) \in \overline{\boldsymbol{\mathcal{V}}}$ is determined by the designer on specified $\Upsilon(s) \in \boldsymbol{\Upsilon}^{TF}$. Furthermore, given a continuous $v_{_W}(\omega) \in \boldsymbol{\mathcal{V}}$, it is always possible to construct a $W(s) \in \boldsymbol{\mathcal{W}}^{TF}$ that satisfies equation (3.3) by fitting a stable minimum-phase transfer function to each magnitude function.

3.4.2 A Sufficient Condition so that $Q \in \mathscr{RH}_\infty$

A sufficient condition which ensures that $Q(s) \in \mathscr{RH}_\infty$ is obtained by parametrising a subspace of $\mathscr{RH}_\infty$ as follows:

$$Q(s) = \breve{Q} B(s) \quad \text{in which} \quad \begin{cases} \breve{Q} := \begin{bmatrix} \breve{Q}_0 & \breve{Q}_1 & \breve{Q}_2 & \dots & \breve{Q}_N \end{bmatrix} \in \mathbb{R}^{p \times (N+1)q} \\ B(s) := \begin{bmatrix} I_q & \left(\frac{\frac{2}{\tau}-s}{\frac{2}{\tau}+s}\right) I_q & \left(\frac{\frac{2}{\tau}-s}{\frac{2}{\tau}+s}\right)^2 I_q & \dots & \left(\frac{\frac{2}{\tau}-s}{\frac{2}{\tau}+s}\right)^N I_q \end{bmatrix}^T. \end{cases} \tag{3.6}$$

This parametrisation provides a uniform approximation of any $Q \in \mathscr{RH}_\infty$. In fact, τ will be chosen

sufficiently small to capture all fast dynamics and N will be chosen *sufficiently large* so that there are enough parameters $\breve{Q}_i$ to be able to closely model most transfer functions in $\mathscr{RH}_\infty$. This can be easily justified by considering a discrete-time finite impulse response with N samples, each spaced by τ seconds, as depicted in Figure 3.2.

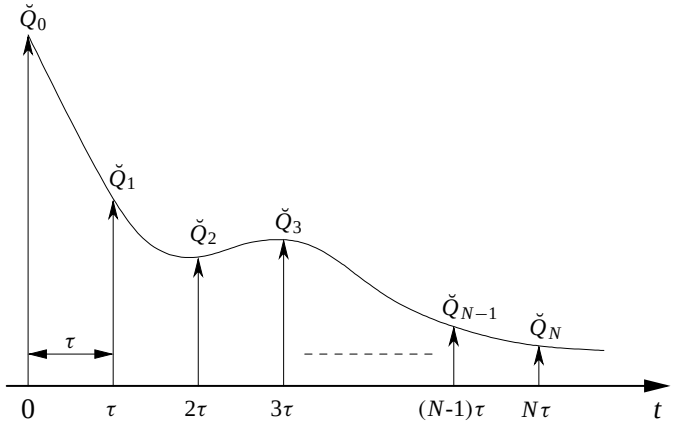


Figure 3.2: Impulse response in discrete and continuous time

The $\mathcal{Z}$-transfer function matrix for this discrete-time impulse response is given by

$$\breve{Q}_0 + \breve{Q}_1 z^{-1} + \breve{Q}_2 z^{-2} + \cdots + \breve{Q}_N z^{-N},$$

which when transformed into a continuous system by Tustin's Transformation $z^{-1} = \left(\frac{\frac{2}{\tau}-s}{\frac{2}{\tau}+s}\right)$ yields

$$Q(s) := \breve{Q}_0 + \breve{Q}_1 \left(\frac{\frac{2}{\tau}-s}{\frac{2}{\tau}+s}\right) + \breve{Q}_2 \left(\frac{\frac{2}{\tau}-s}{\frac{2}{\tau}+s}\right)^2 + \cdots + \breve{Q}_N \left(\frac{\frac{2}{\tau}-s}{\frac{2}{\tau}+s}\right)^N.$$

This is the Laguerre-like parametrisation stated in equation (3.6).

3.4.3 A Sufficient Condition so that $\mu(\cdot) < 1$

Since the μ constraint in optimisation problem (3.2) is not easily computable, it is necessary to replace $\mu_{\mathbf{\Delta}_{TOT}}[\,\cdot\,]$ with some computationally tractable upper bound. A sufficient condition[1] will thus be presented in this sub-section which ensures that

$$\mu_{\mathbf{\Delta}_{TOT}}\left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right)\right] < 1 \;\forall\omega.$$

Towards this end, note that at each fixed frequency ω,

$$\begin{aligned} &\mu_{\mathbf{\Delta}_{TOT}}\left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right)\right] \\ &\qquad \leq \inf_{D_\omega \in \mathcal{D}} \overline{\sigma}\left[\begin{pmatrix} D_\omega & 0 \\ 0 & I_n \end{pmatrix}\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right)\begin{pmatrix} D_\omega^{-1} & 0 \\ 0 & I_m \end{pmatrix}\right] \end{aligned}$$

[1] This sufficient condition is also necessary when the uncertainty set $\mathbf{\Delta}_{TOT}$ satisfies certain conditions.

with equality achieved when the conditions in (Packard and Doyle, 1993, Theorem 8.4) are satisfied. Then,

$$\mu_{\mathbf{\Delta}_{TOT}}\left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right)\right] < 1 \;\; \forall\omega$$

$$\Leftarrow \qquad \forall\omega \;\; \inf_{D_\omega \in \boldsymbol{\mathcal{D}}} \overline{\sigma}\left[\begin{pmatrix} D_\omega & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right) \begin{pmatrix} D_\omega^{-1} & 0 \\ 0 & I_m \end{pmatrix}\right] < 1$$

$$\Leftrightarrow \quad \forall\omega \;\; \exists D_\omega \in \boldsymbol{\mathcal{D}} \;\; : \;\; \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right)^* \begin{pmatrix} D_\omega^* D_\omega & 0 \\ 0 & W(j\omega)^* W(j\omega) \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right)$$
$$< \begin{pmatrix} D_\omega^* D_\omega & 0 \\ 0 & I_m \end{pmatrix}$$

$$\Leftrightarrow \quad \forall\omega \;\; \exists D_\omega \in \boldsymbol{\mathcal{D}} \;\; : \;\; \begin{bmatrix} \begin{pmatrix} \left(D_\omega^* D_\omega\right)^{-1} & 0 \\ 0 & \mathrm{diag}\left(v_W(\omega)\right) \end{pmatrix} & \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right) \\ * & \begin{pmatrix} \left(D_\omega^* D_\omega\right) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0. \qquad (3.7)$$

The last condition follows from a straightforward application of Schur Complement Lemma 2.8.1 and by replacing $\left[W(j\omega)^* W(j\omega)\right]^{-1}$ with $\mathrm{diag}\left(v_W(\omega)\right)$, where $v_W(\omega)$ is defined in equation (3.3). Now, since condition (3.7) is not simultaneously convex in $D_\omega \in \boldsymbol{\mathcal{D}}$ and $Q \in \mathcal{RH}_\infty$, $D_\omega \in \boldsymbol{\mathcal{D}}$ will be held fixed in an eventual optimisation. In order to reduce conservativeness[2] in such an optimisation, $D_\omega \in \boldsymbol{\mathcal{D}}$ will be chosen as the argument of the following frequency-by-frequency minimisation:

$$\min_{D_\omega \in \boldsymbol{\mathcal{D}}} \overline{\sigma}\left[\begin{pmatrix} D_\omega & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right) \begin{pmatrix} D_\omega^{-1} & 0 \\ 0 & I_m \end{pmatrix}\right].$$

This minimisation problem can be rewritten as:

$$\text{For each } \omega \in \mathbb{R},$$
$$\text{minimise } \gamma_\omega$$
$$\text{such that } \exists\, D_\omega \in \boldsymbol{\mathcal{D}} \text{ satisfying}$$
$$\overline{\sigma}\left[\begin{pmatrix} D_\omega & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right) \begin{pmatrix} D_\omega^{-1} & 0 \\ 0 & I_m \end{pmatrix}\right] < \gamma_\omega,$$

[2] That is, in order to ensure that condition (3.7) with a fixed $D_\omega \in \boldsymbol{\mathcal{D}}$ is not too restrictive when compared to the original condition $\mu_{\mathbf{\Delta}_{TOT}}[\,\cdot\,] < 1$.

which after some algebra yields:

$$\begin{gathered}\text{For each } \omega \in \mathbb{R},\\ \text{minimise } \gamma_\omega^2\\ \text{such that } \exists\,(D_\omega^* D_\omega) \in \boldsymbol{\mathcal{D}} \text{ with } (D_\omega^* D_\omega) > 0 \text{ satisfying}\\ \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right)^* \begin{pmatrix} (D_\omega^* D_\omega) & 0 \\ 0 & \operatorname{diag}\big(v_W(\omega)\big)^{-1} \end{pmatrix} \mathcal{F}_l\left(T(j\omega), Q(j\omega)\right) < \gamma_\omega^2 \begin{pmatrix} (D_\omega^* D_\omega) & 0 \\ 0 & I_m \end{pmatrix}.\end{gathered} \tag{3.8}$$

Minimisation problem (3.8) is easy to solve as it is a quasi-convex generalised eigenvalue problem in the variables $(D_\omega^* D_\omega)$ and γ_ω^2.

3.4.4 The Tractable Optimisation Problem

Consider the following optimisation problem, derived using the results of Sections 3.4.1 to 3.4.3:

$$\begin{gathered}\min_{v_W \in \mathcal{V}} \int_{\log_{10}\omega_L}^{\log_{10}\omega_H} v_\Upsilon(\omega)^T v_W(\omega)\; d(\log_{10}\omega)\\ \text{such that}\\ \exists \text{ a } \breve{Q} \in \mathbb{R}^{p\times(N+1)q} \text{ and } \forall\omega \text{ a } (D_\omega^* D_\omega) \in \boldsymbol{\mathcal{D}}\\ \text{satisfying}\\ \begin{bmatrix} \begin{pmatrix} \left(D_\omega^* D_\omega\right)^{-1} & 0 \\ 0 & \operatorname{diag}\big(v_W(\omega)\big) \end{pmatrix} & \mathcal{F}_l\left(T(j\omega), \breve{Q}B(j\omega)\right) \\ * & \begin{pmatrix} (D_\omega^* D_\omega) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0.\end{gathered} \tag{3.9}$$

It can be seen that the closed-loop properties guaranteed by optimisation problem (3.2) are also guaranteed by the above optimisation problem, as the latter has tighter (i.e. more restrictive) constraints. Recall also that $\mathcal{F}_l\left(T(j\omega), \breve{Q}B(j\omega)\right)$ appearing above is affine in $\breve{Q}$, as can be seen from the definition of $T(s)$ in Corollary 3.3.2.

Unfortunately, optimisation problem (3.9) cannot be easily solved as the constraint of this problem is not simultaneously convex in $(D_\omega^* D_\omega)$ and $\breve{Q}$. If however $(D_\omega^* D_\omega)$ is held fixed, then optimisation problem (3.9) reduces to a simple LMI optimisation problem. In order to reduce conservativeness, this fixed $(D_\omega^* D_\omega)$ should be chosen as the solution of optimisation problem (3.8), which itself requires $\breve{Q}$ and $v_W(\omega)$ to be fixed. This co-dependence between these optimisation problems indicates that some sort of iterative scheme must be used (see Section 3.6) to solve optimisation problem (3.9).

3.5 A Pointwise Approximation

In this section, optimisation problem (3.9) will be approximated by a pointwise optimisation problem, as (3.9) involves a search over a functional set with constraints holding for all $\omega \in \mathbb{R}$. To this end, let

$\tilde{\omega}$ denote logarithmic frequency (i.e. $\tilde{\omega} := \log_{10}\omega$) and consider Figure 3.3 in which

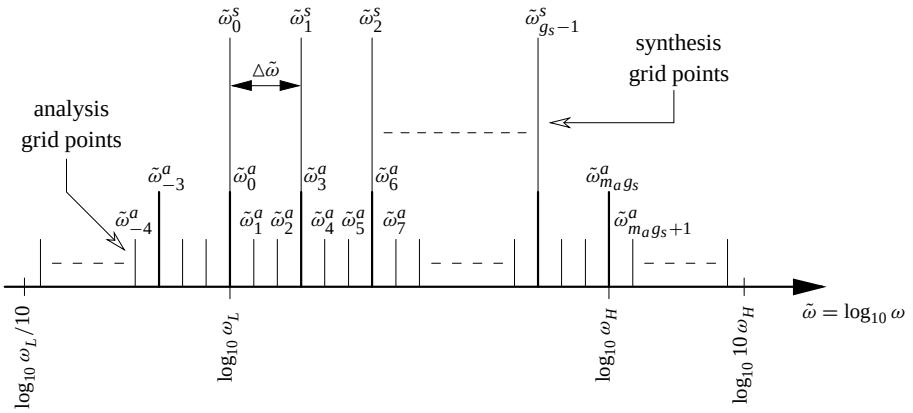


Figure 3.3: Synthesis and analysis grid points on a logarithmic scale

- $g_s \in \mathbb{Z}_+$ is the desired number of synthesis grid points between $\log_{10}\omega_L$ and $\log_{10}\omega_H$. It must be emphasised that the synthesis grid must be chosen dense enough to ensure that all changes in the transfer function matrices $T_{ij}(s)$, defined in Corollary 3.3.2, are captured;
- $m_a \in \mathbb{Z}_+$ denotes the multiplicity for the analysis grid points. That is, $m_a \in \mathbb{Z}_+$ represents how much denser the analysis gridding is required when compared to the synthesis gridding. Note that the analysis grid points are also required to extend a decade below and a decade above the synthesis grid points.

Now, define the (constant) spacing between the synthesis grid points to be:

$$\triangle\tilde{\omega} := \frac{\log_{10}\omega_H - \log_{10}\omega_L}{g_s} = \log_{10}\left[\left(\frac{\omega_H}{\omega_L}\right)^{1/g_s}\right]. \tag{3.10}$$

Then, the synthesis grid points are given by:

$$\tilde{\omega}^s_k := \log_{10}\omega_L + k\,\triangle\tilde{\omega} = \log_{10}\left[\omega_L\left(\frac{\omega_H}{\omega_L}\right)^{k/g_s}\right] \text{ for } k \in \mathbb{Z},\ 0 \le k \le (g_s-1)$$

$$\Rightarrow \quad \omega^s_k := 10^{\tilde{\omega}^s_k} = \omega_L\left(\frac{\omega_H}{\omega_L}\right)^{k/g_s} \text{ for } k \in \mathbb{Z},\ 0 \le k \le (g_s-1). \tag{3.11}$$

Note that there is no synthesis grid point at the frequency ω_H. The reason for this will become apparent later when the cost function of optimisation problem (3.9) is approximated by a pointwise in frequency cost function.

Before defining the analysis grid points in a similar way, the following set needs to be defined:

Definition 3.5.1 *Let the set of indices for the analysis grid points be defined by:*

$$\mathbf{\Omega} := \left\{k \in \mathbb{Z} \ : \ -\left\lfloor\frac{m_a}{\triangle\tilde{\omega}}\right\rfloor \le k \le \left\lfloor\frac{m_a}{\triangle\tilde{\omega}}\right\rfloor + m_a g_s\right\}$$

where $\lfloor x\rfloor$ is the "floor" of x.

Then, the analysis grid points are given by:

$$\tilde{\omega}_k^a := \log_{10}\omega_L + k\frac{\triangle\tilde{\omega}}{m_a} = \log_{10}\left[\omega_L\left(\frac{\omega_H}{\omega_L}\right)^{k/(m_a g_s)}\right] \text{ for } k \in \mathbf{\Omega}$$

$$\Rightarrow \quad \omega_k^a := 10^{\tilde{\omega}_k^a} = \omega_L\left(\frac{\omega_H}{\omega_L}\right)^{k/(m_a g_s)} \text{ for } k \in \mathbf{\Omega}. \tag{3.12}$$

Now, a first-order approximation of the cost function in optimisation problem (3.9) is given by:

$$\int_{\log_{10}\omega_L}^{\log_{10}\omega_H} v_\Upsilon(\omega)^T v_W(\omega)\, d(\log_{10}\omega) \;\approx\; (\triangle\tilde{\omega})\sum_{k=0}^{g_s-1} v_\Upsilon(\omega_k^s)^T v_W(\omega_k^s).$$

Then, for ease of notation, define the following vector (belonging to $\mathbb{R}^n$) for each $k \in \mathbf{\Omega}$:

$$v_{\Upsilon,k} := \begin{cases} v_\Upsilon(\omega^s_{k/m_a}) & \text{when } 0 \le \lfloor k/m_a \rfloor = k/m_a \le (g_s-1), \\ \begin{bmatrix}\epsilon & \epsilon & \cdots & \epsilon\end{bmatrix}^T & \text{otherwise,} \end{cases} \tag{3.13}$$

where $\epsilon \in \mathbb{R}_+$ is some very small number[3]. It is easy to see that $v_{\Upsilon,k}$ is simply the directionality vector $v_\Upsilon(\omega)$ whenever the synthesis grid points and the analysis grid points coincide (see Figure 3.3) and is a very small cost elsewhere.

Hence, a pointwise in frequency approximation of optimisation problem (3.9) is given by:

$$\min_{v_{W,k}\in\mathbb{R}^n\ \forall k\in\mathbf{\Omega}} \sum_{k\in\mathbf{\Omega}} \left(v_{\Upsilon,k}\right)^T \left(v_{W,k}\right)$$

such that

$$\exists \text{ a } \breve{Q} \in \mathbb{R}^{p\times(N+1)q} \text{ and } \forall k \in \mathbf{\Omega} \text{ a } \Theta_k \in \boldsymbol{\mathcal{D}} \tag{3.14}$$

satisfying

$$\begin{bmatrix} \begin{pmatrix}\Theta_k^{-1} & 0 \\ 0 & \mathrm{diag}\left(v_{W,k}\right)\end{pmatrix} & \mathcal{F}_l\left(T(j\omega_k^a), \breve{Q}B(j\omega_k^a)\right) \\ * & \begin{pmatrix}\Theta_k & 0 \\ 0 & I_m\end{pmatrix} \end{bmatrix} > 0.$$

On writing the above optimisation problem, $(\triangle\tilde{\omega})$ has been removed from the cost function as it is a constant and hence does not affect the arguments of the minimisation. Furthermore, observe that in the above optimisation problem, $v_{W,k}$ (resp. Θ_k) represents the pointwise values of $v_W(\omega)$ (resp. $D_\omega^* D_\omega$) at each analysis grid frequency $\omega = \omega_k^a$. Note also that there is no need to restrict the vectors $v_{W,k}$ to belong to $\mathbb{R}_+^n$ in the arguments of minimisation (3.14), as positivity of each element in $v_{W,k}$ is implicitly guaranteed by the constraint of this optimisation problem.

A small $\epsilon \in \mathbb{R}_+$ is used in the definition of $v_{\Upsilon,k}$ (see equation (3.13)) so as to ensure that the vectors $v_{W,k}$ resulting from optimisation problem (3.14) are "reasonably smooth" as k changes in $\mathbf{\Omega}$. This is

[3] By "very small" it is meant a factor of 100, say, less than the smallest element in $v_\Upsilon(\omega_k^s)\ \forall k \in \mathbb{Z},\ 0 \le k \le (g_s-1)$.

because the vectors $v_{w,k}$ at grid points corresponding to the analysis grid but not to the synthesis grid (see Figure 3.3) still appear in the cost function, although their contribution towards the reduction of the cost is small.

Furthermore, a pointwise in frequency approximation of optimisation problem (3.8) can be obtained in a similar way and is given by:

$$\begin{gathered}\text{For each } k \in \boldsymbol{\Omega},\\ \text{minimise } \gamma^2_{\omega^a_k}\\ \text{such that } \exists\, \Theta_k \in \boldsymbol{\mathcal{D}} \text{ with } \Theta_k > 0 \text{ satisfying}\\ \mathcal{F}_l\left(T(j\omega^a_k), \breve{Q}B(j\omega^a_k)\right)^* \begin{pmatrix}\Theta_k & 0\\ 0 & \operatorname{diag}\left(v_{w,k}\right)^{-1}\end{pmatrix} \mathcal{F}_l\left(T(j\omega^a_k), \breve{Q}B(j\omega^a_k)\right) < \gamma^2_{\omega^a_k} \begin{pmatrix}\Theta_k & 0\\ 0 & I_m\end{pmatrix}.\end{gathered} \tag{3.15}$$

Here, again, $v_{w,k}$ and Θ_k are the pointwise values of $v_w(\omega)$ and $(D^*_\omega D_\omega)$ at each analysis grid frequency $\omega = \omega^a_k$. This is a quasi-convex generalised eigenvalue problem which is easily solved using LMI routines.

3.6 Solution Algorithm

Optimisation problem (3.14) cannot be directly solved, as this problem is not simultaneously convex in Θ_k and $\breve{Q}$. If, however, the variables Θ_k are held fixed, then optimisation problem (3.14) reduces to a simple LMI optimisation problem. In order to reduce conservativeness, Θ_k should be chosen as the solution of optimisation problem (3.15) at each grid point $k \in \boldsymbol{\Omega}$. Now, since these two optimisation problems are interdependent, an iterative algorithm is proposed.

Inputs to the algorithm:

- Generalised plant $G(s)$ satisfying the standard assumptions of Definition 3.2.5,
- Directionality transfer function matrix $\Upsilon(s) \in \boldsymbol{\Upsilon}^{TF}$.

The solution algorithm:

1. Compute $T(s)$ using the definition in Corollary 3.3.2, $B(s)$ using the definition in equation (3.6) and $v_{\Upsilon,k}$ using both equations (3.4) and (3.13).

   Then, for every $k \in \boldsymbol{\Omega}$, select a $\Theta^\star_{k,0} \in \boldsymbol{\mathcal{D}}$ with $\Theta^\star_{k,0} > 0$ such that the constraint of the optimisation problem given in Step 3 below admits some feasible solution, provided that such a solution exists. These $\Theta^\star_{k,0} \in \boldsymbol{\mathcal{D}}$ constitute a feasible initial starting point for the algorithm. A systematic procedure for selecting such a feasible initial starting point is available and will be described in detail in Section 3.7. However, it should be pointed out that setting $\Theta^\star_{k,0} = I_r \ \forall k \in \boldsymbol{\Omega}$ is usually good enough.

   Set $i = 0$, where i denotes the iteration number, and $\eta^\star_0 = \infty$.

2. Increment i by 1.

3. Solve the following convex optimisation problem:

$$\min_{v_{w,k}\in\mathbb{R}^n\ \forall k\in\boldsymbol{\Omega}} \sum_{k\in\boldsymbol{\Omega}} \left(v_{\Upsilon,k}\right)^T \left(v_{W,k}\right)$$

such that $\exists\, \breve{Q} \in \mathbb{R}^{p\times(N+1)q}$ satisfying

$$\begin{bmatrix} \begin{pmatrix} (\Theta^\star_{k,i-1})^{-1} & 0 \\ 0 & \mathrm{diag}\left(v_{W,k}\right) \end{pmatrix} & \mathcal{F}_l\left(T(j\omega^a_k),\, \breve{Q}B(j\omega^a_k)\right) \\ * & \begin{pmatrix} (\Theta^\star_{k,i-1}) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0 \quad \forall k \in \boldsymbol{\Omega}.$$

Here $T(j\omega^a_k)$, $B(j\omega^a_k)$ and $v_{\Upsilon,k}$ are given and $\Theta^\star_{k,i-1}$ is the value of Θ_k obtained in the previous iteration. Recall that $\mathcal{F}_l\left(T(j\omega^a_k),\, \breve{Q}B(j\omega^a_k)\right)$ is affine in $\breve{Q}$ and hence the above optimisation problem is a simple LMI problem.

Let the value of this minimum cost be denoted by $\eta^\star_i$ and let the values of $\breve{Q}$ and $\mathrm{diag}\left(v_{W,k}\right)$ at each $k \in \boldsymbol{\Omega}$ that achieve this minimum be denoted by $\breve{Q}^\star_i$ and $\Pi^\star_{k,i}$ for each $k \in \boldsymbol{\Omega}$ respectively.

4. Solve the following convex optimisation problem *for each* $k \in \boldsymbol{\Omega}$:

$$\text{Minimise } \gamma^2_{\omega^a_k}$$

such that $\exists\, \Theta_k \in \boldsymbol{\mathcal{D}}$ with $\Theta_k > 0$ satisfying

$$\mathcal{F}_l\left(T(j\omega^a_k),\, \breve{Q}^\star_i B(j\omega^a_k)\right)^* \begin{pmatrix} \Theta_k & 0 \\ 0 & \left(\Pi^\star_{k,i}\right)^{-1} \end{pmatrix} \mathcal{F}_l\left(T(j\omega^a_k),\, \breve{Q}^\star_i B(j\omega^a_k)\right) < \gamma^2_{\omega^a_k} \begin{pmatrix} \Theta_k & 0 \\ 0 & I_m \end{pmatrix}.$$

Here $T(j\omega^a_k)$ and $B(j\omega^a_k)$ are given, and $\breve{Q}^\star_i$ and $\Pi^\star_{k,i}$ are the values of the $\breve{Q}$ and $\mathrm{diag}\left(v_{W,k}\right)$ obtained in Step 3. Note that the above optimisation problem can be easily solved using LMI routines.

For each $k \in \boldsymbol{\Omega}$, let the square root of the above minimum cost be denoted by $\gamma^\star_{\omega^a_k,\,i}$ and let the value of Θ_k that achieves this minimum be denoted by $\Theta^\star_{k,i}$.

5. Evaluate $(\eta^\star_{i-1} - \eta^\star_i)$. If this difference (which is always positive) is very small and has remained very small for the last few iterations, then go to Step 6. Otherwise return to Step 2.

6. Construct the controller $K^\star_i(s)$ corresponding to the above $\breve{Q}^\star_i$ using Theorem 3.3.1 with $Q(s) = \breve{Q}^\star_i B(s)$ and then model reduce $K^\star_i(s)$ if necessary.

Outputs from the algorithm: (after i iterations)

- The element-by-element magnitude of the largest performance weights obtained by the algorithm in $\left(\Pi^\star_{k,i}\right)^{-\frac{1}{2}}\ \forall k \in \boldsymbol{\Omega}$,
- The final D-scales used by the algorithm in $\left(\Theta^\star_{k,i}\right)^{\frac{1}{2}}\ \forall k \in \boldsymbol{\Omega}$,

- The controller $K_i^\star(s) \in \boldsymbol{\mathcal{K}}_G^{TF}$ that achieves robust performance with respect to these weights,
- The value of the minimum cost $\eta_i^\star$ and the final upper bound of $\mu_{\boldsymbol{\Delta}_{TOT}}\big[\,\cdot\,\big]$ in $\gamma^\star_{\omega_k^a,\,i}$ $\forall k \in \boldsymbol{\Omega}$.

Note that at each iteration i, Step 3 ensures that $\max_{k\in\boldsymbol{\Omega}} \gamma^\star_{\omega_k^a,\,i} \leq 1$ and Step 4 minimises $\gamma^\star_{\omega_k^a,\,i}$ at each fixed $k \in \boldsymbol{\Omega}$. This immediately guarantees robust performance for all uncertainty $\Delta \in \mathbf{B}\boldsymbol{\Delta}^{TF}$.

Moreover, as the iterations proceed (i.e. as i increases), the minimum cost $\eta_i^\star$ is monotonically non-increasing. This is because the solutions $\breve{Q}_i^\star$ and $\Pi^\star_{k,i}$ $\forall k \in \boldsymbol{\Omega}$ obtained in Step 3 at the i-th iteration always satisfy the constraint of the same optimisation problem in Step 3 at the $(i+1)$-th iteration. This can be seen by observing that at Step 4, $\breve{Q}_i^\star$ and $\Pi^\star_{k,i}$ $\forall k \in \boldsymbol{\Omega}$ are held fixed while $\gamma^2_{\omega_k^a}$ (which is always ≤ 1) is minimised at each fixed $k \in \boldsymbol{\Omega}$ over Θ_k to give the new $\Theta^\star_{k,i}$ for each $k \in \boldsymbol{\Omega}$. This new $\Theta^\star_{k,i}$ will then be used in Step 3 at the $(i+1)$-th iteration. The fact that $\eta_i^\star$ is monotonically non-increasing and is bounded below by 0 means that it will converge to a limit point as the number of iterations tend to infinity, by the 'Principle of Monotone Sequences' (Haggarty, 1994). However, iterative algorithms as the one presented above cannot be guaranteed to converge to the *global minimum*. Only monotonic properties can be proved.

3.7 Finding a Feasible Initial Starting Point

The algorithm proposed in Section 3.6 requires in its first step a feasible initial starting point that initialises the algorithm. That is, it requires for all $k \in \boldsymbol{\Omega}$ some $\Theta^\star_{k,0} \in \boldsymbol{\mathcal{D}}$ with $\Theta^\star_{k,0} > 0$ such that the constraint of the optimisation problem given in Step 3 admits some feasible solution. The aim of this section is to find such a $\Theta^\star_{k,0}$. However, before doing this, the following lemma will be stated which considerably simplifies the problem.

Lemma 3.7.1 *Let $T(s)$ be defined as in Corollary 3.3.2 and $B(s)$ be defined as in equation* (3.6)*. Then the following two statements are equivalent:*

(i) There exists a $\breve{Q} \in \mathbb{R}^{p\times(N+1)q}$ and $\forall k \in \boldsymbol{\Omega}$ a $\Theta^\star_{k,0} \in \boldsymbol{\mathcal{D}}$ and a $v_{w,k} \in \mathbb{R}^n$ satisfying

$$\begin{bmatrix} \begin{pmatrix} (\Theta^\star_{k,0})^{-1} & 0 \\ 0 & \mathrm{diag}\big(v_{w,k}\big) \end{pmatrix} & \mathcal{F}_l\left(T(j\omega_k^a),\, \breve{Q}B(j\omega_k^a)\right) \\ * & \begin{pmatrix} (\Theta^\star_{k,0}) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0.$$

(ii) There exists a $\breve{Q} \in \mathbb{R}^{p\times(N+1)q}$ and $\forall k \in \boldsymbol{\Omega}$ a $\Theta^\star_{k,0} \in \boldsymbol{\mathcal{D}}$ satisfying

$$\begin{bmatrix} (\Theta^\star_{k,0})^{-1} & T_{11}(j\omega_k^a) + T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a) \\ * & (\Theta^\star_{k,0}) \end{bmatrix} > 0.$$

Proof See Appendix A.1 for proof. □

This lemma gives a simpler equivalent condition (which can be given the interpretation of a robust stability condition) that will be used to construct a feasible initial starting point for the algorithm of Section 3.6. Towards this end, note that the inequality in condition (ii) of Lemma 3.7.1 is equivalent to:

$$\overline{\sigma}\left[(\Theta_{k,0}^{\star})^{\frac{1}{2}}\left[T_{11}(j\omega_k^a)+T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)\right](\Theta_{k,0}^{\star})^{-\frac{1}{2}}\right]<1 \text{ with } \Theta_{k,0}^{\star}>0 \quad \forall k\in\mathbf{\Omega}.$$

Consequently, one way of finding a feasible initial starting point for the algorithm of Section 3.6 is to minimise

$$\overline{\sigma}\left[(\Theta_{k,0}^{\star})^{\frac{1}{2}}\left[T_{11}(j\omega_k^a)+T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)\right](\Theta_{k,0}^{\star})^{-\frac{1}{2}}\right] \tag{3.16}$$

over both $\Theta_{k,0}^{\star}\in\boldsymbol{\mathcal{D}}$ with $\Theta_{k,0}^{\star}>0$ for all $k\in\mathbf{\Omega}$ and $\breve{Q}\in\mathbb{R}^{p\times(N+1)q}$, and stop this minimisation when expression (3.16) is less than unity. This minimisation problem is however not simultaneously convex in both variables and hence expression (3.16) has to be alternately minimised over each variable in succession. The two steps in the resulting iterative minimisation are obtained by noting that *at each* $k\in\mathbf{\Omega}$:

$$\overline{\sigma}\left[(\Theta_{k,0}^{\star})^{\frac{1}{2}}\left[T_{11}(j\omega_k^a)+T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)\right](\Theta_{k,0}^{\star})^{-\frac{1}{2}}\right]<\varsigma_{\omega_k^a}$$

$$\Updownarrow$$

$$\left[T_{11}(j\omega_k^a)+T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)\right]^{*}(\Theta_{k,0}^{\star})\left[T_{11}(j\omega_k^a)+T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)\right]<\varsigma_{\omega_k^a}^2(\Theta_{k,0}^{\star}),$$

and that:

$$\overline{\sigma}\left[(\Theta_{k,0}^{\star})^{\frac{1}{2}}\left[T_{11}(j\omega_k^a)+T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)\right](\Theta_{k,0}^{\star})^{-\frac{1}{2}}\right]<\varsigma \qquad \forall k\in\mathbf{\Omega}$$

$$\Updownarrow$$

$$\begin{bmatrix}\varsigma.(\Theta_{k,0}^{\star})^{-1} & T_{11}(j\omega_k^a)+T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)\\ * & \varsigma.(\Theta_{k,0}^{\star})\end{bmatrix}>0 \qquad \forall k\in\mathbf{\Omega}$$

$$\Updownarrow$$

$$\begin{bmatrix}0 & T_{11}(j\omega_k^a)+T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)\\ * & 0\end{bmatrix}<\varsigma\begin{bmatrix}(\Theta_{k,0}^{\star})^{-1} & 0\\ 0 & (\Theta_{k,0}^{\star})\end{bmatrix} \qquad \forall k\in\mathbf{\Omega}.$$

The former inequality can be used for minimisation of $\varsigma_{\omega_k^a}^2$ over $(\Theta_{k,0}^{\star})$, whereas the latter inequality can be used for minimisation of ς over $\breve{Q}$. This iterative minimisation may be stopped when either $\varsigma\leq 1$ or $\varsigma_{\omega_k^a}^2\leq 1\ \forall k\in\mathbf{\Omega}$.

3.8 Numerical Example

The algorithm proposed in Section 3.6 will now be illustrated by a numerical example. Consider the block diagram shown in Figure 3.4, which is a typical Sensitivity/Complementary Sensitivity reduction problem.

The actual plant is uncertain but is known to belong to the set $\left\{P_o(1+\Delta W_u)\ :\ \Delta\in\mathscr{RH}_\infty,\ \|\Delta\|_\infty\leq 1\right\}$ parametrised by Δ (see dashed box). Here, the nominal plant P_o was chosen as $\frac{10}{s(s+10)}$ and

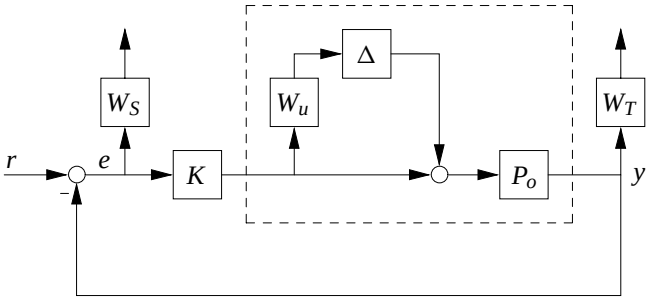


Figure 3.4: Block diagram for a typical S/T problem

the uncertainty weight W_u as $\frac{10(s+5)}{(s+100)}$. The uncertainty weight W_u represents any 'a priori' knowledge about the frequency dependent size of the uncertainty. In this example, the chosen W_u allows the magnitude of the actual plant to differ from that of the nominal plant by as much as 50% in the low-frequency region (say, below 5 `rad/s`) and by as much as 1000% in the high-frequency region.

It is required to maximise the performance weights W_S and W_T according to some pre-specified directionality while ensuring that there exists an internally stabilising controller $K(s) \in \mathcal{K}_G^{TF}$ that achieves robust performance with respect to these maximised weights. The directionality used in this example is shown in Figure 3.5. This directionality basically states that the optimisation problem

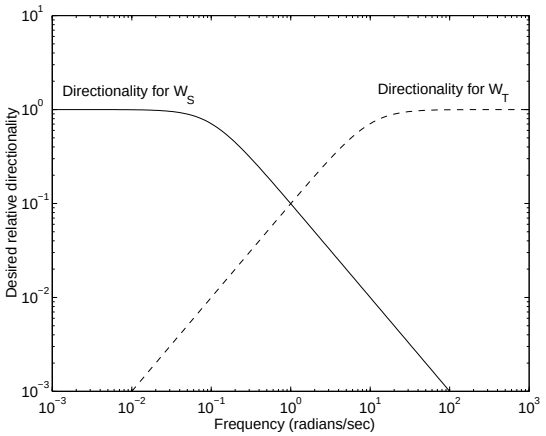


Figure 3.5: Desired directionality for the optimisation

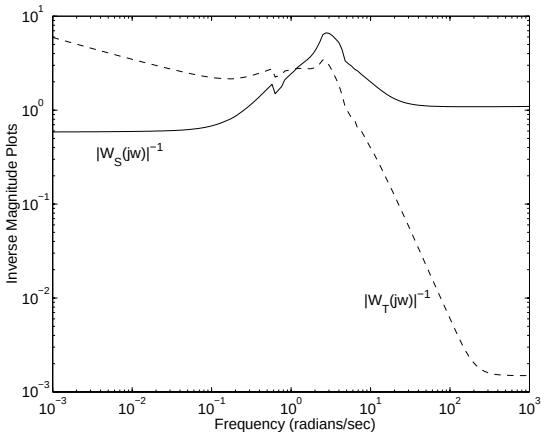
should maximise W_S (resp. W_T) in the low-frequency (resp. high-frequency) region and that it should not bother too much about maximising W_S (resp. W_T) in the remaining high-frequency (resp. low-frequency) region. The scale on the y-axis of this figure is unimportant as it only affects the cost associated with the optimisation. Only the relative sizes between the different curves and the shape of each curve across frequency is important. Note that the low-frequency value of the solid curve is equal to the high-frequency value of the dashed curve. This means that the optimisation problem should value the maximisation of W_S at low-frequency equally as the maximisation of W_T at high-frequency.

At around 1 `rad/s`, the solid curve and the dashed curve are also equal. This again means that around this frequency, the maximisation of W_S is equally important as the maximisation of W_T. However, since $S + T = 1$, it is not possible to make both W_S and W_T large at this same frequency. Thus, in this case, the optimisation problem needs to sort out by itself how much maximisation is possible in each of these weights. Furthermore, since the magnitude of the directionalities at 1 `rad/s` is about a decade less than the magnitude of the directionalities at low and high-frequency, maximisation of W_S and W_T at this mid-frequency region should be considered less important than maximisation of W_S and W_T at low and high-frequency.

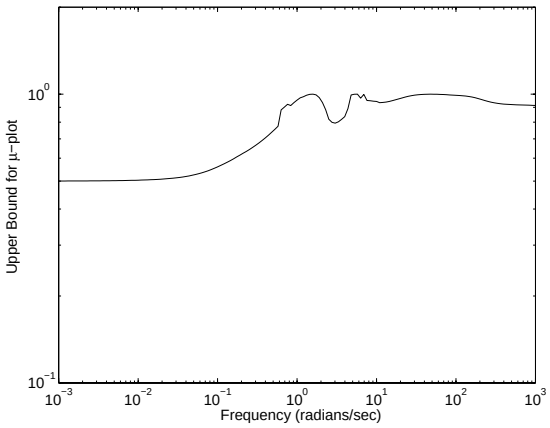
The frequency range $[\omega_L, \omega_H]$ selected for maximising the performance weights was $\left[10^{-2}, 10^{2}\right]$, whereas that selected for gridding the constraint was $\left[10^{-3}, 10^{3}\right]$. Furthermore, $\tau = 0.2$ and $N = 1$ were found to be sufficiently good, as smaller values of τ and larger values of N did not give any improvement[4]. In order to illustrate the behaviour of the algorithm as iterations proceed, consider Figure 3.6 which gives plots of intermediate results at the i-th and $(i+1)$-th iterations. Figure 3.6(a) shows typical inverse magnitude pointwise plots of the performance weights obtained by the algorithm at some i-th iteration. These weights together with some internally stabilising controller give the computed upper bound of μ depicted in Figure 3.6(b). This upper bound can be seen to be less than unity at all frequencies, as it must satisfy the optimisation's constraint. The "space" between this computed upper bound of μ and unity will then be exploited by the algorithm at the $(i+1)$-th iteration to synthesise better pointwise performance weights. In other words, at the $(i+1)$-th iteration, the algorithm will push $|W_S(j\omega)|^{-1}$ and $|W_T(j\omega)|^{-1}$ down at appropriate frequencies (according to the directionalities of Figure 3.5) while ensuring that the upper bound of μ never exceeds unity. The new pointwise performance weights resulting from this minimisation are shown in Figure 3.6(c). It is clear that these new performance weights guarantee a higher level of robust performance. The upper bound of μ resulting from this change in performance weights is the solid line in Figure 3.6(d). It is flat across frequency and is slightly less than unity, thereby implying that all freedom has been exploited. Then, this new upper bound of μ is minimised at each frequency over the D-scales, keeping the new performance weights and the new internally stabilising controller fixed. The result of this D-scale minimisation is seen as the dashed-dotted curve in Figure 3.6(d). Consequently, there is now some further "space" between the computed upper bound of μ and unity which can be exploited by the algorithm in the next iteration.

Six iterations were found to be sufficient for convergence of the algorithm and the total iteration time taken was less than 7 minutes (on a 400 MHz Pentium II PC). Figure 3.7 shows plots of the final pointwise performance weights $|W_S(j\omega)|^{-1}$ and $|W_T(j\omega)|^{-1}$ obtained by the algorithm. The magnitude plot of the sensitivity (resp. complementary sensitivity) function of any plant in the set $\left\{P_o(1 + \Delta W_u) : \Delta \in \mathscr{RH}_\infty, \|\Delta\|_\infty \leq 1\right\}$ will be below the $|W_S(j\omega)|^{-1}$ (resp. $|W_T(j\omega)|^{-1}$) plot

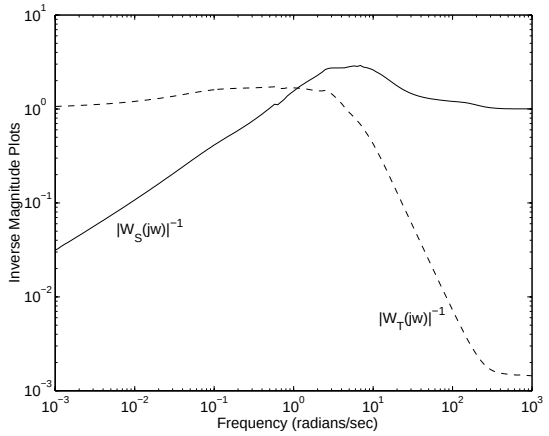
[4]Such a low-order basis is due to the simplicity of the problem considered. For more complex problems, a higher-order basis is obviously required.



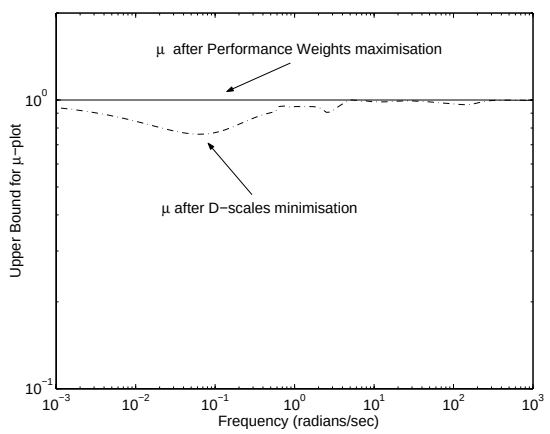
(a) Performance weights at i-th iteration



(b) μ upper bound at i-th iteration



(c) Performance weights at $(i+1)$-th iteration



(d) μ upper bounds at $(i+1)$-th iteration

Figure 3.6: Intermediate results at the i-th and $(i+1)$-th iterations

since robust performance is guaranteed. A controller that achieves robust performance with respect to these maximised performance weights was computed at the end of the iterations and is given by the following state-space representation:

$$K(s) = \left[\begin{array}{rrrr|r} -5.80 & 2.09 & -57.60 & 13.42 & -7.62 \\ -3.06 & -10.07 & 2.40 & -0.44 & 0.30 \\ 0 & 1.00 & -1.74 & 0 & -0.17 \\ 0 & 0 & 44.72 & -10.00 & 4.47 \\ \hline -3.06 & -0.07 & 1.87 & -0.44 & 0.25 \end{array}\right].$$

Its Bode plot is shown in Figure 3.8. This controller together with the performance weights of Figure 3.7 gave a flat curve across frequency for the computed upper bound of μ that was slightly less than unity.

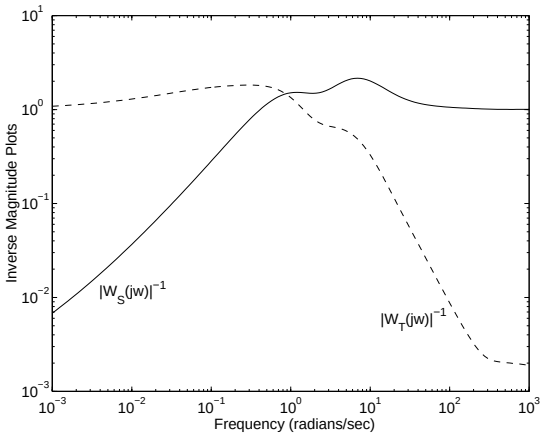


Figure 3.7: Final pointwise inverse magnitude performance weights

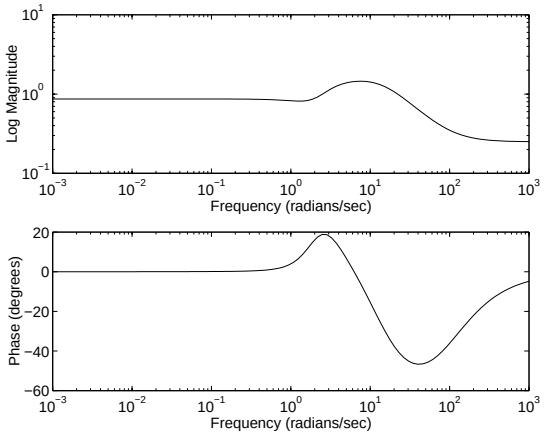


Figure 3.8: Bode plot for controller $K(s) = \mathcal{F}_l\big(J(s), \breve{Q}B(s)\big)$

The algorithm is said to have converged as the computed upper bound of μ could not be minimised any further over the D-scales and hence no further improvement of the performance weights could be achieved.

3.9 Summary and Comments

The problem of maximising performance weights in the frequency range of interest, subject to the existence of an internally stabilising controller that guarantees robust performance with respect to these maximised weights, was posed as an optimisation problem in Section 3.2. However, this optimisation problem was difficult to compute. Thus, a computationally tractable optimisation problem was formulated in Section 3.4 which had tighter and hence more restrictive constraints than the original problem. A pointwise approximation of this optimisation problem was then given in Section 3.5. Reduction of conservativeness in this problem required D-scales to be computed from yet another optimisation,

that was unfortunately interlinked with the chosen optimisation problem. Consequently, an iterative procedure was proposed in Section 3.6 to take care of this. Both these optimisation problems were written as LMI problems.

Some important advantages of this algorithm over existing methods for robust performance μ-synthesis (e.g. D-K iterations) are given here:

I. Performance weights, which maximise some cost function that captures the desired closed-loop performance, are synthesised simultaneously with an internally stabilising controller by one systematic algorithm.

II. The controller synthesised by this algorithm immediately guarantees robust performance with respect to these maximised weights. Also, incompatible performance weights can never be obtained by this algorithm, as the performance weights obtained must be feasible to the optimisation's constraint.

III. This optimisation usually gives a final closed-loop μ-curve that is as flat as possible across frequency and very close to unity, reflecting the fact that robust performance has been maximised.

IV. Performance weights and D-scales are found and used pointwise in frequency and hence need not be fitted with stable minimum-phase transfer function matrices.

Consequently, this approach greatly simplifies the often long and tedious trial and error process of designing "good" performance weights directly.

The proposed algorithm does, however, suffer from some important disadvantages:

I. The Laguerre-like parametrisation $Q(s) = \breve{Q}B(s)$ usually causes high-order controllers to be synthesised.

II. Frequency gridding causes loss of information between the grid-points and hence a dense grid can only give confidence that $\mu < 1$ rather than absolute certainty.

III. Maximisation of performance weights occurs only in the frequency range $[\omega_L, \omega_H]$.

All these problems will be directly addressed in Chapter 4, where a very similar optimisation problem is computed using state-space methods that completely eliminate the above mentioned difficulties.

In summary, this chapter addresses the μ-synthesis robust performance problem from a new and conceptually different point of view. The results presented in this chapter were also published in Lanzon and Richards (1999).

Chapter 4

A State-Space Algorithm for μ-Synthesis

The optimisation problem posed in Chapter 3 is modified slightly in this chapter to yield a new optimisation problem that admits a state-space solution without significantly compromising the properties of the original optimisation. This state-space solution eliminates the disadvantages of the pointwise approach taken in Chapter 3 and considerably enhances the benefits of this type of optimisation based weight selection. For instance, the controller is no longer parametrised by a basis function and its order is guaranteed to be less than or equal to that of the scaled generalised plant. Consequently, this chapter presents a significant generalisation of the work and results given in the previous chapter and develops an algorithm that is a valuable alternative to the standard D-K iterative procedure.

4.1 Problem Modification

Consider the optimisation framework introduced in Section 3.2. This same framework will also be used in this chapter with only one technical difference — the set $\boldsymbol{\Upsilon}^{TF}$ will be restricted to contain only strictly proper transfer function matrices.

Definition 4.1.1 *Re-define the set of directionality transfer function matrices by:*

$$\boldsymbol{\Upsilon}^{TF} := \left\{ \Upsilon(s) \in \mathscr{RH}_\infty : \Upsilon(\infty) = 0,\, \Upsilon(s_o) \in \boldsymbol{\Lambda}\ \forall s_o \in \overline{\mathbb{C}}_+ \right\}.$$

The reason for this technical restriction is that the cost function of the new optimisation problem involves $\left\| \Upsilon W^{-1} \right\|_2$. Now, since $W(s) \in \boldsymbol{\mathcal{W}}^{TF}$ is invertible, a strictly proper $\Upsilon(s) \in \boldsymbol{\Upsilon}^{TF}$ is necessary for a finite two-norm. The fact that $\Upsilon(s)$ is forced to be strictly proper does not limit its choice, as any directionality function $\Upsilon(s)$ can be rolled-off at sufficiently high frequencies (beyond the system dynamics) without affecting its functionality.

Now consider the following optimisation problem for a given generalised plant $G(s)$ satisfying the standard assumptions stated in Definition 3.2.5 and an 'a priori' chosen directionality transfer

function matrix $\Upsilon(s) \in \boldsymbol{\Upsilon}^{TF}$:

$$\max_{W \in \boldsymbol{W}^{TF}} \frac{1}{\left\| \Upsilon W^{-1} \right\|_2} \quad : \quad \min_{K \in \boldsymbol{\mathcal{K}}_G^{TF}} \sup_{\omega} \mu_{\boldsymbol{\Delta}_{TOT}} \left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(G(j\omega), K(j\omega)\right) \right] < 1. \qquad (4.1)$$

It can be easily seen that this optimisation problem is very similar to optimisation problem (3.1) proposed in Section 3.2 — the only difference being in the cost function. Thus, the motivation given in Section 3.2 as to why optimisation problem (3.1) is an interesting and sensible problem to consider also justifies the interest in the above optimisation problem. The interpretation of the above constraint and the meaning of the directionality transfer function matrix $\Upsilon(s)$ is identical. However, unlike the cost function used in optimisation problem (3.1), $1 \Big/ \left\| \Upsilon W^{-1} \right\|_2$ covers the entire frequency range from $-\infty$ to ∞. Furthermore, a simple $\|\cdot\|_2$ is used here instead of the semi-norm defined in Definition 3.2.7, which involved logarithmic frequency as the variable of integration, since optimisation problem (4.1) will be solved using state-space techniques, and gridding on a logarithmic scale is no longer required.

4.2 Replacing μ with an Upper Bound

Since the μ constraint in optimisation problem (4.1) is not easily computable, it is necessary (as in Chapter 3) to replace $\mu_{\boldsymbol{\Delta}_{TOT}}\left[\,\cdot\,\right]$ with a computationally tractable upper bound. To this end, recall that (Packard and Doyle, 1993)

$$\mu_{\boldsymbol{\Delta}}\left[P(j\omega)\right] \;\leq\; \inf_{D_\omega \in \boldsymbol{\mathcal{D}}} \overline{\sigma}\left[D_\omega P(j\omega) D_\omega^{-1}\right],$$

with equality achieved when certain conditions are satisfied (Packard and Doyle, 1993, Theorem 8.4). Then,

$$\begin{aligned} &\sup_{\omega} \mu_{\boldsymbol{\Delta}_{TOT}} \left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(G(j\omega), K(j\omega)\right) \right] \\ &\qquad \leq \sup_{\omega} \inf_{D_\omega \in \boldsymbol{\mathcal{D}}} \overline{\sigma} \left[\begin{pmatrix} D_\omega & 0 \\ 0 & I_n \end{pmatrix} \begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathcal{F}_l\left(G(j\omega), K(j\omega)\right) \begin{pmatrix} D_\omega^{-1} & 0 \\ 0 & I_m \end{pmatrix} \right] \\ &\qquad \leq \inf_{D \in \boldsymbol{\mathcal{D}}^{TF}} \left\| \begin{pmatrix} D & 0 \\ 0 & W \end{pmatrix} \mathcal{F}_l\left(G, K\right) \begin{pmatrix} D^{-1} & 0 \\ 0 & I_m \end{pmatrix} \right\|_\infty . \end{aligned}$$

The last inequality follows from (Zhou et al., 1996, Section 11.4). It is not an equality as $D \in \boldsymbol{\mathcal{D}}^{TF}$ is restricted to be *real-rational* and hence cannot uniformly approximate discontinuous $D_\omega \in \boldsymbol{\mathcal{D}}$.

Recall also that $\sup_\omega \mu_{\boldsymbol{\Delta}_{TOT}}\left[\,\cdot\,\right] < 1$ is necessary and sufficient for robust stability/performance against linear time-invariant structured perturbations of norm no greater than unity (see Theorem 2.4.3). So, by the above upper bound, $\inf_{D \in \boldsymbol{\mathcal{D}}^{TF}} \left\| \cdot \right\|_\infty < 1$ is a sufficient condition (in some cases conservative and in others tight) which guarantees that $\sup_\omega \mu_{\boldsymbol{\Delta}_{TOT}}\left[\,\cdot\,\right] < 1$. However, $\inf_{D \in \boldsymbol{\mathcal{D}}^{TF}} \left\| \cdot \right\|_\infty < 1$ is important in its own right since it is necessary and sufficient for robust stability/performance against

arbitrarily slow time-varying linear structured perturbations of norm no greater than unity (Poolla and Tikku, 1995).

In view of these points, the following optimisation problem

$$\max_{W\in\boldsymbol{\mathcal{W}}^{TF}} \frac{1}{\left\|\Upsilon W^{-1}\right\|_2} \quad : \quad \min_{K\in\boldsymbol{\mathcal{K}}_G^{TF}} \inf_{D\in\boldsymbol{\mathcal{D}}^{TF}} \left\| \begin{pmatrix} D & 0 \\ 0 & W \end{pmatrix} \mathcal{F}_l\left(G,K\right) \begin{pmatrix} D^{-1} & 0 \\ 0 & I_m \end{pmatrix} \right\|_\infty < 1 \tag{4.2}$$

will be considered henceforth for a given generalised plant $G(s)$ satisfying the standard assumptions stated in Definition 3.2.5 and an 'a priori' chosen directionality transfer function matrix $\Upsilon(s) \in \boldsymbol{\Upsilon}^{TF}$.

Optimisation problem (4.2) will now be rewritten into a form more suitable for subsequent synthesis steps. First note that since only the arguments of the optimisation are of interest, an equivalent problem to optimisation (4.2) is

$$\min_{W\in\boldsymbol{\mathcal{W}}^{TF}} \left\|\Upsilon W^{-1}\right\|_2^2 \quad : \quad \min_{K\in\boldsymbol{\mathcal{K}}_G^{TF}} \inf_{D\in\boldsymbol{\mathcal{D}}^{TF}} \left\| \begin{pmatrix} D & 0 \\ 0 & W \end{pmatrix} \mathcal{F}_l\left(G,K\right) \begin{pmatrix} D^{-1} & 0 \\ 0 & I_m \end{pmatrix} \right\|_\infty < 1.$$

Furthermore, since $\left\|P\right\|_{2,(\infty)} = \left\|P^T\right\|_{2,(\infty)}$ and using the definitions

$$\bar{D} := D^{-T} \in \boldsymbol{\mathcal{D}}^{TF}$$
$$\text{and} \quad \bar{W} := W^{-T} \in \boldsymbol{\mathcal{W}}^{TF},$$

this optimisation problem may be rewritten as

$$\min_{\bar{W}\in\boldsymbol{\mathcal{W}}^{TF}} \left\|\bar{W}\Upsilon\right\|_2^2 \quad : \quad \min_{K\in\boldsymbol{\mathcal{K}}_G^{TF}} \inf_{\bar{D}\in\boldsymbol{\mathcal{D}}^{TF}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l\left(G,K\right)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1. \tag{4.3}$$

The reasons for considering the dual systems inside the norms rather than the original systems will become apparent in the proofs of the main theorems given in this chapter. As an indication, note that this transpose operation moves the performance weights from the left of $\mathcal{F}_l\left(G,K\right)$ to the right of $\mathcal{F}_l\left(G,K\right)^T$, which is desirable, without altering the stability properties of the systems inside the norms.

This optimisation problem is however not simultaneously convex in all of the variables. Hence, a sub-optimal iterative algorithm will be derived in subsequent sections to solve optimisation problem (4.3). This algorithm basically alternates between:

- holding $K \in \boldsymbol{\mathcal{K}}_G^{TF}$ fixed and solving optimisation problem (4.3) over $\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}$ and $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}$ and
- holding $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}$ fixed and solving optimisation problem (4.3) over $\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}$ and $K \in \boldsymbol{\mathcal{K}}_G^{TF}$.

Sections 4.5, 4.6 and 4.7 explore this in more detail. However, the commuting properties satisfied by the state-space realisations of $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}$, $\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}$ and $\Upsilon \in \boldsymbol{\Upsilon}^{TF}$ need to be explicitly defined first.

4.3 Commuting Properties

This section derives the commuting properties which need to be satisfied by the state-space realisations of $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}$, $\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}$ and $\Upsilon \in \boldsymbol{\Upsilon}^{TF}$. To this end, select *arbitrary* realisations

$$\bar{D} := \left[\begin{array}{c|c} A_{\bar{D}} & B_{\bar{D}} \\ \hline C_{\bar{D}} & D_{\bar{D}} \end{array}\right] \in \boldsymbol{\mathcal{D}}^{TF},$$

$$\bar{W} := \left[\begin{array}{c|c} A_{\bar{W}} & B_{\bar{W}} \\ \hline C_{\bar{W}} & D_{\bar{W}} \end{array}\right] \in \boldsymbol{\mathcal{W}}^{TF},$$

$$\text{and} \quad \Upsilon := \left[\begin{array}{c|c} A_{\Upsilon} & B_{\Upsilon} \\ \hline C_{\Upsilon} & 0 \end{array}\right] \in \boldsymbol{\Upsilon}^{TF},$$

with $A_{\bar{D}}$, $A_{\bar{W}}$ and A_{Υ} Hurwitz. Furthermore, define

$$T^o_{\bar{D}}(s) := \begin{bmatrix} (sI_{s_D} - A_{\bar{D}})^{-1}B_{\bar{D}} \\ I_r \end{bmatrix} \quad \text{and} \quad T^o_{\bar{W}}(s) := \begin{bmatrix} (sI_{s_w} - A_{\bar{W}})^{-1}B_{\bar{W}} \\ I_n \end{bmatrix}. \tag{4.4}$$

Then, by Lemma 2.6.3, the frequency function:

I. $\bar{D}(j\omega)^*\bar{D}(j\omega)$ can be written as $T^o_{\bar{D}}(j\omega)^*\breve{D}T^o_{\bar{D}}(j\omega)$ for some $\breve{D} := \begin{bmatrix} 0 & \breve{D}_{12} \\ \breve{D}_{12}^T & \breve{D}_{22} \end{bmatrix}$ with $\breve{D}_{12} \in \mathbb{R}^{s_D \times r}$ and $\breve{D}_{22} = \breve{D}_{22}^T \in \mathbb{R}^{r \times r}$,

II. $\bar{W}(j\omega)^*\bar{W}(j\omega)$ can be written as $T^o_{\bar{W}}(j\omega)^*\breve{W}T^o_{\bar{W}}(j\omega)$ for some $\breve{W} := \begin{bmatrix} 0 & \breve{W}_{12} \\ \breve{W}_{12}^T & \breve{W}_{22} \end{bmatrix}$ with $\breve{W}_{12} \in \mathbb{R}^{s_w \times n}$ and $\breve{W}_{22} = \breve{W}_{22}^T \in \mathbb{R}^{n \times n}$.

Different parametrisations are of course also possible. However, some parametrisations result in non-convex conditions and others are unnecessarily complicated or have considerable redundancy. The above parametrisations are chosen with this in mind and according to what is needed in subsequent sections.

Since it is required that $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}$, $\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}$ and $\Upsilon \in \boldsymbol{\Upsilon}^{TF}$, it is clear that $\bar{D}(j\omega)^*\bar{D}(j\omega)$ should commute with $\boldsymbol{\Delta}$, $\bar{W}(j\omega)^*\bar{W}(j\omega)$ should commute with $\boldsymbol{\Lambda}$ and $\Upsilon(j\omega)$ should commute with $\boldsymbol{\Lambda}$. As such, $T^o_{\bar{D}}(j\omega)^*\breve{D}T^o_{\bar{D}}(j\omega)$ needs to commute with $\boldsymbol{\Delta}$ and $T^o_{\bar{W}}(j\omega)^*\breve{W}T^o_{\bar{W}}(j\omega)$ needs to commute with $\boldsymbol{\Lambda}$. These commuting conditions ensure that states associated with one D-scale block, performance weight or directionality factor do not get mixed up with states associated with other D-scale blocks, performance weights or directionality factors. An approach similar to that in (Helmersson, 1995, Section 8.4.3) is taken here to ensure all of this.

Definition 4.3.1 *Define the following sets:*

$$\hat{\mathbf{\Delta}} := \left\{ \overset{f}{\underset{i=1}{\operatorname{diag}}} \left(I_{\zeta_i} \otimes \Delta_i \right) \ : \ \Delta_i \in \mathbb{C}^{\beta_i \times \beta_i}, \ \sum_{i=1}^{f} \zeta_i \beta_i = s_{_D} \right\}$$

$$\hat{\mathbf{\Lambda}}_W := \left\{ \overset{n}{\underset{i=1}{\operatorname{diag}}} \left(I_{\zeta_i^W} \otimes \ell_i \right) \ : \ \ell_i \in \mathbb{C}, \ \sum_{i=1}^{n} \zeta_i^W = s_{_W} \right\}$$

$$\hat{\mathbf{\Lambda}}_\Upsilon := \left\{ \overset{n}{\underset{i=1}{\operatorname{diag}}} \left(I_{\zeta_i^\Upsilon} \otimes \ell_i \right) \ : \ \ell_i \in \mathbb{C}, \ \zeta_i^\Upsilon \neq 0, \ \sum_{i=1}^{n} \zeta_i^\Upsilon = s_{_\Upsilon} \right\},$$

in which $s_{_D}$, $s_{_W}$ and $s_{_\Upsilon}$ are the total number of states allocated to $\bar{D}(s)$, $\bar{W}(s)$ and $\Upsilon(s)$ respectively.

Here, $\zeta_i \beta_i$ is the number of states allocated to the i-th D-scale block, ζ_i^W is the number of states allocated to the i-th performance weight and ζ_i^Υ is the number of states allocated to the i-th diagonal element in $\Upsilon(s)$. If $\zeta_i = 0$ or $\zeta_i^W = 0$, then there are no states associated with the i-th block of $\bar{D}(s)$ or $\bar{W}(s)$ respectively. Consequently, these scalings are constant across frequency, hence allowing for nonlinear time-varying uncertainty or constant performance across frequency (Helmersson, 1995, Section 8.4.3). Note however that ζ_i^Υ is not allowed to be 0 in the definition of set $\hat{\mathbf{\Lambda}}_\Upsilon$. This is because $\zeta_i^\Upsilon = 0$ would imply that the i-th diagonal element of $\Upsilon(j\omega)$ is identically zero across frequency (as each diagonal element of $\Upsilon(j\omega)$ is strictly proper), and in turn this would mean that the i-th performance weight, which was selected for maximisation, would not be maximised at all.

Definition 4.3.2 *Define the "copy" operators $\mathcal{C}$, $\mathcal{C}_W$ and $\mathcal{C}_\Upsilon$ by the following mappings:*

$$\begin{aligned}
\mathcal{C} \ : \quad & \overset{f}{\underset{i=1}{\operatorname{diag}}} \left(I_{\alpha_i} \otimes \Delta_i \right) \in \mathbf{\Delta} \quad \mapsto \quad \overset{f}{\underset{i=1}{\operatorname{diag}}} \left(I_{\zeta_i} \otimes \Delta_i \right) \in \hat{\mathbf{\Delta}} \\
\mathcal{C}_W \ : \quad & \overset{n}{\underset{i=1}{\operatorname{diag}}} \left(I_1 \otimes \ell_i \right) \in \mathbf{\Lambda} \quad \mapsto \quad \overset{n}{\underset{i=1}{\operatorname{diag}}} \left(I_{\zeta_i^W} \otimes \ell_i \right) \in \hat{\mathbf{\Lambda}}_W \\
\mathcal{C}_\Upsilon \ : \quad & \overset{n}{\underset{i=1}{\operatorname{diag}}} \left(I_1 \otimes \ell_i \right) \in \mathbf{\Lambda} \quad \mapsto \quad \overset{n}{\underset{i=1}{\operatorname{diag}}} \left(I_{\zeta_i^\Upsilon} \otimes \ell_i \right) \in \hat{\mathbf{\Lambda}}_\Upsilon .
\end{aligned}$$

Then, it follows that:

I. $\bar{D}(j\omega)^* \bar{D}(j\omega) = T_{\bar{D}}^o(j\omega)^* \breve{D} T_{\bar{D}}^o(j\omega)$ commutes with $\mathbf{\Delta}$ if

$$\begin{bmatrix} \hat{\Delta} & 0 \\ 0 & \Delta \end{bmatrix} \breve{D} = \breve{D} \begin{bmatrix} \hat{\Delta} & 0 \\ 0 & \Delta \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \hat{\Delta} & 0 \\ 0 & \Delta \end{bmatrix} T_{\bar{D}}^o(j\omega) = T_{\bar{D}}^o(j\omega) \Delta$$

for all $\hat{\Delta} = \mathcal{C}(\Delta)$, $\Delta \in \mathbf{\Delta}$;

II. $\bar{W}(j\omega)^* \bar{W}(j\omega) = T_{\bar{W}}^o(j\omega)^* \breve{W} T_{\bar{W}}^o(j\omega)$ commutes with $\mathbf{\Lambda}$ if

$$\begin{bmatrix} \hat{\Lambda}_W & 0 \\ 0 & \Lambda_W \end{bmatrix} \breve{W} = \breve{W} \begin{bmatrix} \hat{\Lambda}_W & 0 \\ 0 & \Lambda_W \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \hat{\Lambda}_W & 0 \\ 0 & \Lambda_W \end{bmatrix} T_{\bar{W}}^o(j\omega) = T_{\bar{W}}^o(j\omega) \Lambda_W$$

for all $\hat{\Lambda}_W = \mathcal{C}_W(\Lambda_W)$, $\Lambda_W \in \mathbf{\Lambda}$.

This leads to the following characterisations of the required structure of $\breve{D}$, $\breve{W}$, $A_{\bar{D}}$, $B_{\bar{D}}$, $A_{\bar{W}}$ and $B_{\bar{W}}$.

Definition 4.3.3 *The following sets define the structure of the parameters $\breve{D}$ and $\breve{W}$.*

$$\mathbf{\Xi}_{\breve{D}} := \left\{ \breve{D} = \begin{bmatrix} 0 & \breve{D}_{12} \\ \breve{D}_{12}^T & \breve{D}_{22} \end{bmatrix} \;:\; \breve{D}_{12} \in \mathbb{R}^{s_D \times r},\ \breve{D}_{22} = \breve{D}_{22}^T \in \mathbb{R}^{r \times r}, \right.$$
$$\left. \begin{bmatrix} \hat{\Delta} & 0 \\ 0 & \Delta \end{bmatrix} \breve{D} = \breve{D} \begin{bmatrix} \hat{\Delta} & 0 \\ 0 & \Delta \end{bmatrix} \ \forall \hat{\Delta} = \mathcal{C}(\Delta),\ \Delta \in \mathbf{\Delta} \right\},$$

$$\mathbf{\Xi}_{\breve{W}} := \left\{ \breve{W} = \begin{bmatrix} 0 & \breve{W}_{12} \\ \breve{W}_{12}^T & \breve{W}_{22} \end{bmatrix} \;:\; \breve{W}_{12} \in \mathbb{R}^{s_W \times n},\ \breve{W}_{22} = \breve{W}_{22}^T \in \mathbb{R}^{n \times n}, \right.$$
$$\left. \begin{bmatrix} \hat{\Lambda}_W & 0 \\ 0 & \Lambda_W \end{bmatrix} \breve{W} = \breve{W} \begin{bmatrix} \hat{\Lambda}_W & 0 \\ 0 & \Lambda_W \end{bmatrix} \ \forall \hat{\Lambda}_W = \mathcal{C}_W(\Lambda_W),\ \Lambda_W \in \mathbf{\Lambda} \right\}.$$

The commuting conditions on $T_{\breve{D}}^o(j\omega)$ and $T_{\breve{W}}^o(j\omega)$ may be rewritten as commuting conditions on $(A_{\bar{D}}, B_{\bar{D}})$ and $(A_{\bar{W}}, B_{\bar{W}})$ respectively as follows:

(a) $$\begin{bmatrix} \hat{\Delta} & 0 \\ 0 & \Delta \end{bmatrix} T_{\breve{D}}^o(j\omega) = T_{\breve{D}}^o(j\omega)\Delta \quad \forall \hat{\Delta} = \mathcal{C}(\Delta),\ \Delta \in \mathbf{\Delta}. \quad \Leftrightarrow \quad \hat{\Delta}A_{\bar{D}} = A_{\bar{D}}\hat{\Delta} \text{ and } \hat{\Delta}B_{\bar{D}} = B_{\bar{D}}\Delta \quad \forall \hat{\Delta} = \mathcal{C}(\Delta),\ \Delta \in \mathbf{\Delta}.$$

(b) $$\begin{bmatrix} \hat{\Lambda}_W & 0 \\ 0 & \Lambda_W \end{bmatrix} T_{\breve{W}}^o(j\omega) = T_{\breve{W}}^o(j\omega)\Lambda_W \quad \forall \hat{\Lambda}_W = \mathcal{C}_W(\Lambda_W),\ \Lambda_W \in \mathbf{\Lambda}. \quad \Leftrightarrow \quad \hat{\Lambda}_W A_{\bar{W}} = A_{\bar{W}}\hat{\Lambda}_W \text{ and } \hat{\Lambda}_W B_{\bar{W}} = B_{\bar{W}}\Lambda_W \quad \forall \hat{\Lambda}_W = \mathcal{C}_W(\Lambda_W),\ \Lambda_W \in \mathbf{\Lambda}.$$

Definition 4.3.4 *The following sets define the structure of $(A_{\bar{D}}, B_{\bar{D}})$ and $(A_{\bar{W}}, B_{\bar{W}})$.*

$$\mathbf{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})} := \left\{ (A_{\bar{D}}, B_{\bar{D}}) \;:\; A_{\bar{D}} \in \mathbb{R}^{s_D \times s_D},\ B_{\bar{D}} \in \mathbb{R}^{s_D \times r},\ A_{\bar{D}} \textit{ is Hurwitz}, \right.$$
$$\left. \hat{\Delta}A_{\bar{D}} = A_{\bar{D}}\hat{\Delta} \textit{ and } \hat{\Delta}B_{\bar{D}} = B_{\bar{D}}\Delta \ \forall \hat{\Delta} = \mathcal{C}(\Delta),\ \Delta \in \mathbf{\Delta} \right\},$$

$$\mathbf{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})} := \left\{ (A_{\bar{W}}, B_{\bar{W}}) \;:\; A_{\bar{W}} \in \mathbb{R}^{s_W \times s_W},\ B_{\bar{W}} \in \mathbb{R}^{s_W \times n},\ A_{\bar{W}} \textit{ is Hurwitz}, \right.$$
$$\left. \hat{\Lambda}_W A_{\bar{W}} = A_{\bar{W}}\hat{\Lambda}_W \textit{ and } \hat{\Lambda}_W B_{\bar{W}} = B_{\bar{W}}\Lambda_W \ \forall \hat{\Lambda}_W = \mathcal{C}_W(\Lambda_W),\ \Lambda_W \in \mathbf{\Lambda} \right\}.$$

Finally, note that $\Upsilon(s)$ commutes with $\mathbf{\Lambda}$ if

$$\begin{bmatrix} \hat{\Lambda}_\Upsilon & 0 \\ 0 & \Lambda_\Upsilon \end{bmatrix} \begin{bmatrix} A_\Upsilon & B_\Upsilon \\ C_\Upsilon & 0 \end{bmatrix} = \begin{bmatrix} A_\Upsilon & B_\Upsilon \\ C_\Upsilon & 0 \end{bmatrix} \begin{bmatrix} \hat{\Lambda}_\Upsilon & 0 \\ 0 & \Lambda_\Upsilon \end{bmatrix} \quad \forall \hat{\Lambda}_\Upsilon = \mathcal{C}_\Upsilon(\Lambda_\Upsilon),\ \Lambda_\Upsilon \in \mathbf{\Lambda}. \tag{4.5}$$

4.4 Restrictions of the Optimisation Sets

As is usual with state-space methods used to address optimisation problems such as the one posed here, attention has to be limited to optimisation over a subclass of performance weights and D-scales

— that is, those spanned by some fixed basis functions. This is necessary to obtain convex state-space conditions. Thus, the optimisation sets $\boldsymbol{\mathcal{W}}^{TF}$ and $\boldsymbol{\mathcal{D}}^{TF}$ in optimisation problem (4.3) need to be restricted in some way.

Since the frequency functions $\bar{D}(j\omega)^*\bar{D}(j\omega)$ and $\bar{W}(j\omega)^*\bar{W}(j\omega)$ are completely parametrised by $T^o_{\bar{D}}(j\omega)^*\breve{D}T^o_{\bar{D}}(j\omega)$ and $T^o_{\bar{W}}(j\omega)^*\breve{W}T^o_{\bar{W}}(j\omega)$ respectively (as explained in Section 4.3), it is natural to restrict these parametrisations by holding the basis functions $T^o_{\bar{D}}(j\omega)$ and $T^o_{\bar{W}}(j\omega)$ fixed. This effectively amounts to keeping $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}},B_{\bar{D}})}$ and $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}},B_{\bar{W}})}$ fixed. Thus, $\bar{D}(j\omega)^*\bar{D}(j\omega)$ and $\bar{W}(j\omega)^*\bar{W}(j\omega)$ will be solely parametrised in terms of $\breve{D} \in \boldsymbol{\Xi}_{\breve{D}}$ and $\breve{W} \in \boldsymbol{\Xi}_{\breve{W}}$. This of course restricts the parametrisation of $\bar{D}(j\omega)^*\bar{D}(j\omega)$ and $\bar{W}(j\omega)^*\bar{W}(j\omega)$ to the span of $T^o_{\bar{D}}(j\omega)$ and $T^o_{\bar{W}}(j\omega)$ respectively. However, with a sufficiently large number of states $s_{_D}$ and $s_{_W}$, a large enough class can usually be parametrised.

It is desirable, however, to choose fixed values of $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}},B_{\bar{D}})}$ and $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}},B_{\bar{W}})}$ that are sufficiently close to the optimal values which would have been obtained if these quantities were free variables. This is desirable so that the above parametrisations introduce little restriction. Towards constructing such "close to optimal" values, observe that for a given G satisfying the standard assumptions stated in Definition 3.2.5, a fixed $K \in \boldsymbol{\mathcal{K}}^{TF}_G$ and an 'a priori' chosen $\Upsilon \in \boldsymbol{\Upsilon}^{TF}$, the following optimisation problem

$$\min_{\bar{W}\in\boldsymbol{\mathcal{W}}^{TF}} \left\|\bar{W}\Upsilon\right\|_2^2 \quad : \quad \inf_{\bar{D}\in\boldsymbol{\mathcal{D}}^{TF}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l\left(G, K\right)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1 \tag{4.6}$$

is convex if solved pointwise in frequency. To see this, define the following sets and quantities in a similar way to Section 3.4.1.

Definition 4.4.1 *The sets of strictly-positive and non-negative vector valued functions are defined by:*

$$\boldsymbol{\mathcal{V}} := \left\{ f : \mathbb{R} \mapsto \mathbb{R}^n_+ \right\},$$
$$\overline{\boldsymbol{\mathcal{V}}} := \left\{ f : \mathbb{R} \mapsto \overline{\mathbb{R}}^n_+ \right\}.$$

Furthermore, for ease of notation, define the following vector functions:

$$v_{_W}(\omega) := \left[\bar{W}(j\omega)^*\bar{W}(j\omega)\right] \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} |w_1(j\omega)|^{-2} \\ |w_2(j\omega)|^{-2} \\ \vdots \\ |w_n(j\omega)|^{-2} \end{bmatrix} \in \boldsymbol{\mathcal{V}}, \tag{4.7}$$

$$\text{and} \quad v_{_\Upsilon}(\omega) := \left[\Upsilon(j\omega)\Upsilon(j\omega)^*\right] \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} |\upsilon_1(j\omega)|^2 \\ |\upsilon_2(j\omega)|^2 \\ \vdots \\ |\upsilon_n(j\omega)|^2 \end{bmatrix} \in \overline{\boldsymbol{\mathcal{V}}}. \tag{4.8}$$

Then, using this notation, optimisation problem (4.6) can be rewritten as:

$$\min_{v_W \in \mathcal{V}} \int_{-\infty}^{\infty} v_\Upsilon(\omega)^T v_W(\omega)\, d\omega$$

such that

$$\forall \omega \in \mathbb{R} \cup \{\infty\} \;\; \exists \text{ a continuous } \Theta_\omega \in \boldsymbol{\mathcal{D}} \text{ with } \Theta_\omega > 0 \tag{4.9}$$

satisfying

$$\left[\mathcal{F}_l\left(G(j\omega), K(j\omega)\right)^T\right]^* \begin{pmatrix} \Theta_\omega & 0 \\ 0 & I_m \end{pmatrix} \left[\mathcal{F}_l\left(G(j\omega), K(j\omega)\right)^T\right] < \begin{pmatrix} \Theta_\omega & 0 \\ 0 & \operatorname{diag}\big(v_W(\omega)\big) \end{pmatrix}.$$

Here, Θ_ω denotes the pointwise values of $\bar{D}(j\omega)^* \bar{D}(j\omega)$ at every ω. It is now easy to see that this optimisation is convex in the variables and can be solved pointwise in frequency over a finite grid using LMI routines. Moreover, given a continuous Θ_ω $\forall \omega$ satisfying the above conditions, it is always possible to fit (to any required accuracy) a self-adjoint real-rational unit in $\mathscr{RL}_\infty$, denoted by $\Theta(s)$, which is also positive at infinity. A spectral factor $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}$ which satisfies the constraint of optimisation problem (4.6) can then be computed for this $\Theta(s)$. Similarly, given a continuous $v_W(\omega) \in \mathcal{V}$, one can always construct a $\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}$ that satisfies equation (4.7) by fitting a stable minimum-phase transfer function to each magnitude function. Finally, $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$ and $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$ are obtained from the appropriate state-space realisations of $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}$ and $\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}$ respectively. The first few iterations in the solution algorithm proposed in Section 4.8 construct such "close to optimal" values, besides minimising the appropriate cost function.

Once "close to optimal" values for $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$ and $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$ are found, optimisation problem (4.3) may be restricted by restricting the sets $\boldsymbol{\mathcal{W}}^{TF}$ and $\boldsymbol{\mathcal{D}}^{TF}$.

Definition 4.4.2 *Given* $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$ *and* $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$, *define*

$$\boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}}, B_{\bar{D}})} := \left\{ \bar{D}(s) = \left[\begin{array}{c|c} A_{\bar{D}} & B_{\bar{D}} \\ \hline C_{\bar{D}} & D_{\bar{D}} \end{array}\right] \;:\; \bar{D}(s) \in \boldsymbol{\mathcal{D}}^{TF} \right\} \subset \boldsymbol{\mathcal{D}}^{TF}$$

$$\boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})} := \left\{ \bar{W}(s) = \left[\begin{array}{c|c} A_{\bar{W}} & B_{\bar{W}} \\ \hline C_{\bar{W}} & D_{\bar{W}} \end{array}\right] \;:\; \bar{W}(s) \in \boldsymbol{\mathcal{W}}^{TF} \right\} \subset \boldsymbol{\mathcal{W}}^{TF}.$$

Using these definitions, optimisation problem (4.3) can be restricted to:

$$\min_{\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})}} \left\| \bar{W} \Upsilon \right\|_2^2 \quad : \quad \min_{K \in \boldsymbol{\mathcal{K}}^{TF}_G} \; \inf_{\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}}, B_{\bar{D}})}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l\left(G, K\right)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1, \tag{4.10}$$

where $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$, $\Upsilon \in \boldsymbol{\Upsilon}^{TF}$, $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$ and G are known prior to solving the optimisation. This optimisation problem is of course sub-optimal to optimisation problem (4.3), but as justified above, conservatism is small if s_D and s_W are chosen to be sufficiently large numbers and $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$ and $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$ are chosen close to their optimal values. Each part of optimisation problem (4.10) will be thoroughly investigated in the following sections.

4.5 The Cost Function

In this section, the cost function $\left\|\bar{W}\Upsilon\right\|_2^2$ appearing in optimisation problem (4.10) is rewritten in a more suitable form.

Theorem 4.5.1 *Given* $\Upsilon(s) = \left[\begin{array}{c|c} A_\Upsilon & B_\Upsilon \\ \hline C_\Upsilon & 0 \end{array}\right] \in \boldsymbol{\Upsilon}^{TF}$ *with* A_Υ *Hurwitz,* $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$ *and any* $\bar{W}(s) \in \boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})}$*, define* $T^o_{\bar{W}}(s)$ *as in equation* (4.4) *and let* $\breve{W} \in \boldsymbol{\Xi}_{\breve{W}}$ *be such that* $T^o_{\bar{W}}(j\omega)^*\breve{W}T^o_{\bar{W}}(j\omega) = \bar{W}(j\omega)^*\bar{W}(j\omega)\ \forall\omega \in \mathbb{R}\cup\{\infty\}$*. Then*

$$\left\|\bar{W}\Upsilon\right\|_2^2 = c^T \mathrm{vec}\left(\breve{W}\right)$$

where

$$c := -\left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & C_\Upsilon \end{bmatrix} \otimes \begin{bmatrix} I_{s_w} & 0 \\ 0 & C_\Upsilon \end{bmatrix}\right)\left(\begin{bmatrix} A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon \end{bmatrix} \oplus \begin{bmatrix} A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon \end{bmatrix}\right)^{-1}\left(\begin{bmatrix} 0 \\ B_\Upsilon \end{bmatrix} \otimes \begin{bmatrix} 0 \\ B_\Upsilon \end{bmatrix}\right)\mathrm{vec}(I_n).$$

Proof See Appendix B.1 for proof. □

This theorem implicitly states that "Minimising $\left\|\bar{W}\Upsilon\right\|_2^2$ over $\bar{W}(s) \in \boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})}$ subject to some constraint" is equivalent to "Minimising $c^T\mathrm{vec}\left(\breve{W}\right)$ over $\breve{W} \in \boldsymbol{\Xi}_{\breve{W}}$ subject to the same constraint", provided that $T^o_{\bar{W}}(j\omega)^*\breve{W}T^o_{\bar{W}}(j\omega) > 0\ \ \forall\omega \in \mathbb{R}\cup\{\infty\}$ is guaranteed by the constraint.

4.6 Holding K Fixed in the Constraint

This section shows that for a fixed $K \in \boldsymbol{\mathcal{K}}^{TF}_G$, the constraint

$$\inf_{\bar{D}\in\boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}}, B_{\bar{D}})}} \left\|\begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix}\mathcal{F}_l(G,K)^T\begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix}\right\|_\infty < 1$$

appearing in optimisation problem (4.10) can be rewritten as a set of LMIs in the free variables.

Theorem 4.6.1 *Given* $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$ *and any* $\bar{W}(s) \in \boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})}$*, define* $T^o_{\bar{W}}(s)$ *as in equation* (4.4) *and let* $\breve{W} \in \boldsymbol{\Xi}_{\breve{W}}$ *be such that* $T^o_{\bar{W}}(j\omega)^*\breve{W}T^o_{\bar{W}}(j\omega) = \bar{W}(j\omega)^*\bar{W}(j\omega)\ \forall\omega \in \mathbb{R}\cup\{\infty\}$*. Then given also*

$$\mathcal{F}_l(G,K) = \left[\begin{array}{c|cc} A_{cl} & B_{1cl} & B_{2\,cl} \\ \hline C_{1cl} & D_{11cl} & D_{12\,cl} \\ C_{2\,cl} & D_{21cl} & D_{22\,cl} \end{array}\right],$$

where $A_{cl} \in \mathbb{R}^{s_{cl}\times s_{cl}}$ *is Hurwitz and the partitioning is consistent with Figure 3.1, the following two statements are equivalent for any* $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$*:*

(i) $\inf_{\bar{D}\in\boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}}, B_{\bar{D}})}} \left\|\begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix}\mathcal{F}_l(G,K)^T\begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix}\right\|_\infty < 1.$

(ii) $\exists\, \breve{D} \in \mathbf{\Xi}_{\breve{D}}$, $X = X^T \in \mathbb{R}^{s_D \times s_D}$ *and* $Y = Y^T \in \mathbb{R}^{(s_{cl}+2s_D+s_W)\times(s_{cl}+2s_D+s_W)}$ *such that*

$$\begin{bmatrix} XA_{\bar{D}} + A_{\bar{D}}^T X & XB_{\bar{D}} \\ B_{\bar{D}}^T X & 0 \end{bmatrix} + \breve{D} > 0,$$

$$\begin{bmatrix} Y\grave{A} + \grave{A}^T Y & Y\grave{B} \\ \grave{B}^T Y & 0 \end{bmatrix} + \begin{bmatrix} \grave{C}^T \\ \grave{D}^T \end{bmatrix} \grave{Q} \begin{bmatrix} \grave{C} & \grave{D} \end{bmatrix} < 0;$$

where $\grave{A}$, $\grave{B}$, $\grave{C}$, $\grave{D}$ *and* $\grave{Q}$ *are defined by*

$$\begin{bmatrix} \grave{A} & \grave{B} \\ \grave{C} & \grave{D} \end{bmatrix} := \left[\begin{array}{cccc:cc} A_{\bar{D}} & 0 & 0 & 0 & B_{\bar{D}} & 0 \\ 0 & A_{\bar{W}} & 0 & 0 & 0 & B_{\bar{W}} \\ 0 & 0 & A_{\bar{D}} & B_{\bar{D}}B_{1cl}^T & B_{\bar{D}}D_{11cl}^T & B_{\bar{D}}D_{21cl}^T \\ 0 & 0 & 0 & A_{cl}^T & C_{1cl}^T & C_{2cl}^T \\ \hdashline 0 & 0 & I_{s_D} & 0 & 0 & 0 \\ 0 & 0 & 0 & B_{1cl}^T & D_{11cl}^T & D_{21cl}^T \\ 0 & 0 & 0 & B_{2cl}^T & D_{12cl}^T & D_{22cl}^T \\ I_{s_D} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I_r & 0 \\ 0 & I_{s_W} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_n \end{array}\right],$$

$$\begin{aligned} \grave{Q} &:= \operatorname{diag}\left(\breve{D},\ I_m,\ -\breve{D},\ -\breve{W}\right) \\ &= \operatorname{diag}\left(\begin{bmatrix} 0 & \breve{D}_{12} \\ \breve{D}_{12}^T & \breve{D}_{22} \end{bmatrix},\ I_m,\ -\begin{bmatrix} 0 & \breve{D}_{12} \\ \breve{D}_{12}^T & \breve{D}_{22} \end{bmatrix},\ -\begin{bmatrix} 0 & \breve{W}_{12} \\ \breve{W}_{12}^T & \breve{W}_{22} \end{bmatrix}\right). \end{aligned}$$

Proof See Appendix B.2 for proof. □

The conditions given in Part (ii) of the Theorem are a set of LMIs which are also simultaneously affine in $\breve{W}$. Thus one could optimise some convex cost function over $\breve{W} \in \mathbf{\Xi}_{\breve{W}}$ subject to the existence of variables $\breve{D}$, X and Y satisfying the LMI constraints given in Part (ii) of the Theorem.

4.7 Holding $\bar{D}$ Fixed in the Constraint

This section shows that for a fixed $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}}, B_{\bar{D}})}$, the constraint

$$\min_{K \in \boldsymbol{\mathcal{K}}_G^{TF}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l(G, K)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1$$

appearing in optimisation problem (4.10) can be rewritten as a set of LMIs in the free variables. Controllers $K \in \boldsymbol{\mathcal{K}}_G^{TF}$ that satisfy this constraint will also be constructed at the end of this section.

Theorem 4.7.1 *Given a generalised plant $G(s)$ satisfying the standard assumptions stated in Definition 3.2.5 and scalings $\bar{D}(s) \in \boldsymbol{\mathcal{D}}^{TF}$, define the scaled generalised plant $\tilde{G}(s)$ by*

$$\tilde{G}(s) := \begin{bmatrix} \bar{D}(s)^{-T} & 0 & 0 \\ 0 & I_n & 0 \\ 0 & 0 & I_q \end{bmatrix} G(s) \begin{bmatrix} \bar{D}(s)^{T} & 0 & 0 \\ 0 & I_m & 0 \\ 0 & 0 & I_p \end{bmatrix}$$

and let

$$\left[\begin{array}{c|cc:c} \tilde{A} & \tilde{B}_1 & \tilde{B}_2 & \tilde{B}_3 \\ \hline \tilde{C}_1 & \tilde{D}_{11} & \tilde{D}_{12} & \tilde{D}_{13} \\ \tilde{C}_2 & \tilde{D}_{21} & \tilde{D}_{22} & \tilde{D}_{23} \\ \hdashline \tilde{C}_3 & \tilde{D}_{31} & \tilde{D}_{32} & \tilde{D}_{33} \end{array}\right]$$

be a stabilisable and detectable realisation for $\tilde{G}(s)$ with $\tilde{A} \in \mathbb{R}^{s_{\tilde{G}} \times s_{\tilde{G}}}$, $\tilde{D}_{11} \in \mathbb{R}^{r \times r}$, $\tilde{D}_{22} \in \mathbb{R}^{n \times m}$ and $\tilde{D}_{33} = 0 \in \mathbb{R}^{q \times p}$. Furthermore, given $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$ and any $\bar{W}(s) \in \boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})}$, define $T^o_{\bar{W}}(s)$ as in equation (4.4) and let $\breve{W} \in \boldsymbol{\Xi}_{\breve{W}}$ be such that $T^o_{\bar{W}}(j\omega)^* \breve{W} T^o_{\bar{W}}(j\omega) = \bar{W}(j\omega)^* \bar{W}(j\omega)$ $\forall \omega \in \mathbb{R} \cup \{\infty\}$. Then the following two statements are equivalent:

(i) $$\min_{K \in \boldsymbol{\mathcal{K}}_G^{TF}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l(G, K)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1.$$

(ii) $\exists\, P = P^T \in \mathbb{R}^{(s_w + s_{\tilde{G}}) \times (s_w + s_{\tilde{G}})}$, $R = R^T \in \mathbb{R}^{s_w \times s_w}$, $S \in \mathbb{R}^{s_w \times s_{\tilde{G}}}$ *and* $T = T^T \in \mathbb{R}^{s_{\tilde{G}} \times s_{\tilde{G}}}$ *such that*

$$P > 0, \quad R > 0, \quad T > 0,$$

$$\begin{pmatrix} P & \begin{bmatrix} R & -S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} \\ \begin{bmatrix} R & 0 \\ -S^T & I_{s_{\tilde{G}}} \end{bmatrix} & \begin{bmatrix} R & 0 \\ 0 & T \end{bmatrix} \end{pmatrix} \geq 0,$$

$$\Psi_P^T \cdot \begin{pmatrix} P \begin{bmatrix} A_{\bar{W}} & 0 \\ 0 & \tilde{A}^T \end{bmatrix} + \{\cdot\}^T & P \begin{bmatrix} 0 & B_{\bar{W}} \\ \tilde{C}_1^T & \tilde{C}_2^T \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ \tilde{B}_1 & \tilde{B}_2 \end{bmatrix} \\ * & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} \\ * & * & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} \end{pmatrix} \cdot \Psi_P$$

$$< \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_3^T \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_3 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right),$$

and

$$\Psi_Q^T \cdot \begin{pmatrix} \begin{bmatrix} R & S \\ -S^T & T \end{bmatrix} \begin{bmatrix} A_{\bar{W}} & 0 \\ 0 & \tilde{A} \end{bmatrix} + \{\cdot\}^T & * & * \\ \begin{bmatrix} 0 & \tilde{B}_1^T \\ 0 & \tilde{B}_2^T \end{bmatrix} \begin{bmatrix} I_{s_w} & 0 \\ S^T & T \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} & * \\ \begin{bmatrix} 0 & \tilde{C}_1 \\ B_{\bar{W}}^T & \tilde{C}_2 \end{bmatrix} \begin{bmatrix} R & -S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \cdot \Psi_Q$$

$$< \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \right) \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \right),$$

where

$$\Psi_P := \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_1 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & \psi_2 \\ 0 & \psi_3 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} I_r & 0 \\ 0 & I_m \end{bmatrix} \end{pmatrix}, \qquad \Psi_Q := \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_4 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & \psi_5 \\ 0 & \psi_6 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} I_r & 0 \\ 0 & I_n \end{bmatrix} \end{pmatrix}$$

and the columns of $\begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \end{bmatrix}$ *(respectively,* $\begin{bmatrix} \psi_4 \\ \psi_5 \\ \psi_6 \end{bmatrix}$*) form bases for the null space of* $\begin{bmatrix} \tilde{B}_3^T & \tilde{D}_{13}^T & \tilde{D}_{23}^T \end{bmatrix}$ *(respectively,* $\begin{bmatrix} \tilde{C}_3 & \tilde{D}_{31} & \tilde{D}_{32} \end{bmatrix}$*).*

Proof See Appendix B.3 for proof. □

The conditions given in Part (ii) of the Theorem are a set of LMIs which are also simultaneously affine in $\breve{W}$. Thus one could optimise some convex cost function over $\breve{W} \in \mathbf{\Xi}_{\breve{W}}$ subject to the existence of variables P, R, S and T satisfying the LMI constraints given in Part (ii) of the Theorem.

Corollary 4.7.2 *Let the suppositions of Theorem 4.7.1 hold. Then there exist controllers* $K \in \boldsymbol{\mathcal{K}}_G^{TF}$ *of order* s_K *satisfying*

$$\left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l(G, K)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1$$

if and only if the LMI constraints given in Part (ii) *of Theorem 4.7.1 hold for some* $P = P^T \in \mathbb{R}^{(s_w+s_{\tilde{G}})\times(s_w+s_{\tilde{G}})}$, $R = R^T \in \mathbb{R}^{s_w \times s_w}$, $S \in \mathbb{R}^{s_w \times s_{\tilde{G}}}$ *and* $T = T^T \in \mathbb{R}^{s_{\tilde{G}} \times s_{\tilde{G}}}$ *that further satisfy*

$$\operatorname{rank}\left(\begin{bmatrix} I_{s_w} & S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} P \begin{bmatrix} I_{s_w} & 0 \\ S^T & I_{s_{\tilde{G}}} \end{bmatrix} - \begin{bmatrix} R & 0 \\ 0 & T^{-1} \end{bmatrix} \right) \leq s_K.$$

Such controllers $K(s) = \left[\begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array}\right]$ *can then be constructed by solving*

$$F + U^T \Phi_K V + V^T \Phi_K^T U < 0$$

for the controller parameters $\Phi_K := \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix}$, *where* F, U *and* V *are defined by:*

$$F := \begin{pmatrix} X \begin{bmatrix} A_{\bar{W}} & 0 & 0 \\ 0 & \tilde{A}^T & 0 \\ 0 & 0 & 0 \end{bmatrix} + \{\cdot\}^T & X \begin{bmatrix} 0 & B_{\bar{W}} \\ \tilde{C}_1^T & \tilde{C}_2^T \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ \tilde{B}_1 & \tilde{B}_2 \\ 0 & 0 \end{bmatrix} \\ * & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} \\ * & * & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} \end{pmatrix}$$

$$- \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right),$$

$$U := \left(\begin{bmatrix} 0 & 0 & I_{s_\kappa} \\ 0 & \tilde{B}_3^T & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ \tilde{D}_{13}^T & \tilde{D}_{23}^T \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right),$$

$$V := \left(\begin{bmatrix} 0 & 0 & I_{s_\kappa} \\ 0 & \tilde{C}_3 & 0 \end{bmatrix} X \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ \tilde{D}_{31} & \tilde{D}_{32} \end{bmatrix} \right),$$

and X *is constructed as follows:*

- *Define* $Q := \begin{bmatrix} I_{s_w} & 0 \\ S^T & I_{s_{\bar{G}}} \end{bmatrix} \begin{bmatrix} R^{-1} & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} I_{s_w} & S \\ 0 & I_{s_{\bar{G}}} \end{bmatrix}$.
- *Factorise* $P - Q^{-1} = HH^T$ *with* $H \in \mathbb{R}^{(s_w + s_{\bar{G}}) \times s_K}$.
- *Define* $X := \begin{bmatrix} P & H \\ H^T & I_{s_K} \end{bmatrix}$.

Proof See Appendix B.4 for proof. □

4.8 Solution Algorithm

This section summarises a sub-optimal iterative algorithm for solving optimisation problem

$$\min_{\bar{W}\in\boldsymbol{\mathcal{W}}^{TF}} \left\|\bar{W}\Upsilon\right\|_2^2 \quad : \quad \min_{K\in\boldsymbol{\mathcal{K}}_G^{TF}} \inf_{\bar{D}\in\boldsymbol{\mathcal{D}}^{TF}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l\left(G, K\right)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1$$

given in Section 4.2. Analogous to D-K iterations and other solution methods for μ-type optimisations, an iterative algorithm must be used as the above constraint is not simultaneously convex in $K \in \boldsymbol{\mathcal{K}}_G^{TF}$ and $\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}$.

Inputs to the algorithm:

- Generalised plant $G(s)$ satisfying the standard assumptions stated in Definition 3.2.5,
- Directionality transfer function matrix $\Upsilon(s) \in \boldsymbol{\Upsilon}^{TF}$.

The solution algorithm:

1. First find a controller $K_0^\star$ which robustly stabilises the interconnection $\mathcal{F}_u\Big(\mathcal{F}_l\big(\hat{G}, K_0^\star\big), \Delta\Big)$ for all $\Delta \in \mathbf{B}\boldsymbol{\Delta}^{TF}$, where

$$\hat{G} := \left[\begin{array}{c|cc} A & B_1 & B_3 \\ \hline C_1 & D_{11} & D_{13} \\ C_3 & D_{31} & 0 \end{array}\right].$$

   Such a controller may be found by a variety of design methods, the most popular being the standard D-K iterative procedure. Note that one iteration is usually good enough as initially interest is only in a controller which achieves robust stability regardless of performance requirements (compare $\hat{G}$ with the original generalised plant G). It is not required to find a controller that minimises the value of μ, but only some controller that achieves $\mu < 1$. Such a controller $K_0^\star$ provides an initial feasible starting point for the algorithm proposed here.

   Set $i = 0$, where i denotes the iteration number, and $\eta_0^\star = \infty$.

2. Increment i by 1.

3. • *During the first few iterations:*

   (a) Solve the following convex optimisation problem

$$\min_{\bar{W}\in\boldsymbol{\mathcal{W}}^{TF}} \left\|\bar{W}\Upsilon\right\|_2^2 \quad : \quad \inf_{\bar{D}\in\boldsymbol{\mathcal{D}}^{TF}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l\left(G, K_{i-1}^\star\right)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1$$

   pointwise in frequency on a sufficiently dense but finite grid using the reformulation given in optimisation problem (4.9). Here, Υ and G are given and $K_{i-1}^\star$ is the controller obtained in the previous iteration.

Let the respective values of $v_{_W}(\omega)$ and Θ_ω at each grid frequency $\omega = \omega_k$ (these are vector/matrix decision variables in optimisation problem (4.9)) that achieve the above minimum be denoted by $v^\star_{W,\omega_k}$ and $\Theta^\star_{\omega_k}$.

(b) Construct a $\bar{W}^\star \in \boldsymbol{\mathcal{W}}^{TF}$ by fitting a stable minimum-phase transfer function to each magnitude function in $v^\star_{W,\omega_k}$. This is always possible because $v^\star_{W,\omega_k} \in \mathbb{R}^n_+ \ \forall \omega_k$, as guaranteed by optimisation problem (4.9).

(c) Construct a self-adjoint real-rational unit in $\mathscr{RL}_\infty$ which is positive at infinity by fitting real-rational functions to each element in $\Theta^\star_{\omega_k}$. Denote this unit by $\Theta^\star(s)$. Such a construction is always possible because $\Theta^\star_{\omega_k} > 0 \ \forall \omega_k$, as guaranteed by optimisation problem (4.9). Then compute, for this $\Theta^\star(s)$, a spectral factor[1] $\bar{D}_i^\star \in \boldsymbol{\mathcal{D}}^{TF}$ as outlined in (Francis, 1987, Section 7.3) and model reduce this spectral factor if necessary.

(d) Let $(A_{\bar{W}}, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$ and $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$ be obtained from the appropriate state-space realisations of $\bar{W}^\star(s)$ and $\bar{D}_i^\star(s)$ respectively.

- *During the last few iterations:*

(a) Solve the following convex optimisation problem

$$\min_{\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})}} \left\| \bar{W} \Upsilon \right\|_2^2 \quad : \quad \inf_{\bar{D} \in \boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}}, B_{\bar{D}})}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l \left(G, K^\star_{i-1} \right)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1$$

by making use of Theorems 4.5.1 and 4.6.1. Note that $(A_{\bar{W}}, B_{\bar{W}})$ and $(A_{\bar{D}}, B_{\bar{D}})$ have been previously fixed, Υ and G are given and $K^\star_{i-1}$ is the controller obtained in the previous iteration.

Theorem 4.5.1 states that $\min_{\bar{W} \in \boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})}} \left\| \bar{W} \Upsilon \right\|_2^2$ can be replaced by $\min_{\breve{W} \in \boldsymbol{\Xi}_{\breve{W}}} c^T \mathrm{vec}\big(\breve{W}\big)$, and Theorem 4.6.1 states that the constraint of the above optimisation problem can be replaced by a set of LMI constraints (see Part (ii) of Theorem 4.6.1). Hence, the above optimisation problem reduces to a simple LMI problem.

Let the value of $\breve{D}$ (a matrix decision variable in the LMI constraints) that achieves this minimum be denoted by $\breve{D}_i^\star$.

(b) Using the values of $(A_{\bar{D}}, B_{\bar{D}}) \in \boldsymbol{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$ and the value of $\breve{D}_i^\star \in \boldsymbol{\Xi}_{\breve{D}}$ just obtained, define the parahermitian rational matrix function

$$\Theta_i^\star(s) \ := \ \left[B_{\bar{D}}^T (-s I_{s_{_D}} - A_{\bar{D}}^T)^{-1} \quad I_r \right] \breve{D}_i^\star \begin{bmatrix} (s I_{s_{_D}} - A_{\bar{D}})^{-1} B_{\bar{D}} \\ I_r \end{bmatrix}.$$

Then compute, for this $\Theta_i^\star(s)$, a spectral factor $\bar{D}_i^\star \in \boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}}, B_{\bar{D}})}$ (see Footnote 1 on Page 61). This is always possible because $\Theta_i^\star(j\omega) > 0 \ \forall \omega \in \mathbb{R} \cup \{\infty\}$, as guaranteed by the LMI constraints in Part (ii) of Theorem 4.6.1.

[1] Since spectral factors are not unique and the required spectral factor has to have a block-diagonal structure, each individual diagonal block in $\Theta^\star(s)$ is spectrally factored separately.

4. *(a)* Solve the following convex optimisation problem

$$\min_{\bar{W}\in\boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}},B_{\bar{W}})}} \left\|\bar{W}\Upsilon\right\|_2^2 \quad : \quad \min_{K\in\boldsymbol{\mathcal{K}}^{TF}_{G}} \left\| \begin{pmatrix} \bar{D}_i^{\star} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l\left(G,K\right)^T \begin{pmatrix} \bar{D}_i^{\star\,-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_{\infty} < 1$$

by making use of Theorems 4.5.1 and 4.7.1. Note that $(A_{\bar{W}},\, B_{\bar{W}})$ has been previously fixed, Υ and G are given and $\bar{D}_i^{\star}$ is the scaling just obtained in Step 3.

Theorem 4.5.1 states that $\min_{\bar{W}\in\boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}},B_{\bar{W}})}} \left\|\bar{W}\Upsilon\right\|_2^2$ can be replaced by $\min_{\breve{W}\in\boldsymbol{\Xi}_{\breve{W}}} c^T\mathrm{vec}\big(\breve{W}\big)$, and Theorem 4.7.1 states that the constraint of the above optimisation problem can be replaced by a set of LMI constraints (see Part (ii) of Theorem 4.7.1). Hence, the above optimisation problem reduces to a simple LMI problem.

Let the value of this minimum cost be denoted by $\eta_i^{\star}$ and let the value of $\breve{W}$ (a matrix decision variable in the cost function/LMI constraints) that achieves this minimum be denoted by $\breve{W}_i^{\star}$.

(b) Using the values of $(A_{\bar{W}},\, B_{\bar{W}}) \in \boldsymbol{\Xi}_{(A_{\bar{W}},B_{\bar{W}})}$ and the value of $\breve{W}_i^{\star} \in \boldsymbol{\Xi}_{\breve{W}}$ just obtained, define the parahermitian rational matrix function

$$\Pi_i^{\star}(s) \;:=\; \left[B_{\bar{W}}^T(-sI_{s_w}-A_{\bar{W}}^T)^{-1} \quad I_n\right] \breve{W}_i^{\star} \begin{bmatrix} (sI_{s_w}-A_{\bar{W}})^{-1}B_{\bar{W}} \\ I_n \end{bmatrix}.$$

Then compute, for this $\Pi_i^{\star}(s)$, a spectral factor $\bar{W}_i^{\star} \in \boldsymbol{\mathcal{W}}^{TF}_{(A_{\bar{W}},B_{\bar{W}})}$ (see Footnote 1 on Page 61). This is always possible because $\Pi_i^{\star}(j\omega) > 0 \;\; \forall\omega \in \mathbb{R}\cup\{\infty\}$, as guaranteed by the LMI constraints in Part (ii) of Theorem 4.7.1.

(c) Using Corollary 4.7.2, find a controller that satisfies the constraint of the above optimisation problem and denote this controller by $K_i^{\star}$.

The additional freedom in selecting this controller may be exploited to simultaneously guarantee other properties such as $\mathscr{H}_2$-norm minimisation, regional pole-placement, etc..

5. Evaluate $(\eta_{i-1}^{\star} - \eta_i^{\star})$. If this difference (which is always positive) is very small and has remained very small for the last few iterations, then EXIT. Otherwise return to Step 2.

Outputs from the algorithm: (after i iterations)

- The inverse of the largest performance weights obtained by the algorithm in $\bar{W}_i^{\star} \in \boldsymbol{\mathcal{W}}^{TF}$,
- A controller $K_i^{\star} \in \boldsymbol{\mathcal{K}}_G^{TF}$ that achieves robust performance with respect to these weights,
- The final scalings $\bar{D}_i^{\star} \in \boldsymbol{\mathcal{D}}^{TF}$ used by the algorithm,
- The value of the minimum cost $\eta_i^{\star}$ obtained.

Note that $\eta_i^\star$ is monotonically non-increasing as i increases. This can be seen by observing that the solutions of the optimisation problem at any one step of the algorithm are always feasible to the constraints of the optimisation problem at the next step of the algorithm. The fact that $\eta_i^\star$ is monotonically non-increasing and is bounded below by 0 means that it will converge to a limit point as the number of iterations tend to infinity, by the 'Principle of Monotone Sequences' (Haggarty, 1994). However, iterative algorithms as the one presented above cannot be guaranteed to converge to the *global minimum*. Only monotonic properties can be proven.

Observe also that in the first few iterations, the algorithm constructs $(A_{\bar{W}}, B_{\bar{W}}) \in \mathbf{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$ and $(A_{\bar{D}}, B_{\bar{D}}) \in \mathbf{\Xi}_{(A_{\bar{D}}, B_{\bar{D}})}$ that are reasonably close to their optimal values. This is because no parametrisation of performance weights and D-scales is required for a pointwise in frequency solution of the optimisation problem in Step 3. Once the performance weights and D-scales are found pointwise in frequency, rational approximations need to be computed which must satisfy the optimisation's constraint. If this can be done, then the values of $(A_{\bar{W}}, B_{\bar{W}})$ and $(A_{\bar{D}}, B_{\bar{D}})$ are updated to the new computed values, otherwise no update will take place. In the last few iterations, the algorithm makes use of these "close to optimal" values and the state-space formulation of both optimisation problems to ensure that the solutions obtained by the algorithm guarantee the attained robust performance level.

4.9 Numerical Example

The algorithm proposed in the previous section will now be illustrated by a numerical example. The same example used in Balas et al. (1996) to illustrate the standard D-K iterative procedure will be used here for ease of comparison. This example considers the design of a pitch axis controller for an experimental highly maneuverable aeroplane, the HIMAT. The HIMAT vehicle was a scaled-down remotely piloted vehicle version of an advanced fighter which was flight tested in the late 1970s. The actual HIMAT vehicle is currently on display in the Smithsonian National Aerospace Museum in Washington D.C.

This design example only considers the longitudinal dynamics of the aeroplane. These dynamics are assumed to be decoupled from the lateral dynamics. The state vector consists of the vehicle's basic rigid body variables — i.e. forward velocity, angle-of-attack, pitch rate and pitch angle respectively. The problem is posed as a robust performance problem with multiplicative plant uncertainty at the plant input, and plant output weighted sensitivity/complementary sensitivity as the performance criterion. A block diagram for the closed-loop system, which includes the feedback structure of the plant and controller and elements associated with the uncertainty model and performance objectives, is shown in Figure 4.1. The dashed box represents the "true" aeroplane with associated transfer function matrix P. Inside this box is the nominal model of the aeroplane dynamics, P_o, and the two elements, W_u and Δ, which parametrise the uncertainty in the model. The transfer function matrix W_u reflects any 'a priori' knowledge about the frequency dependent size of the uncertainty in the model. Thus, the

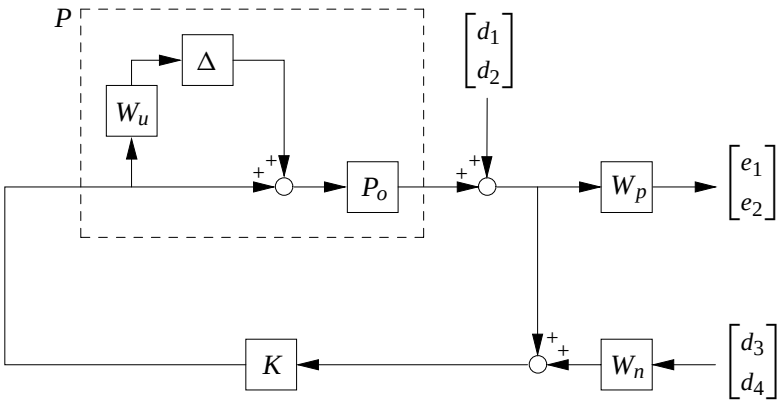
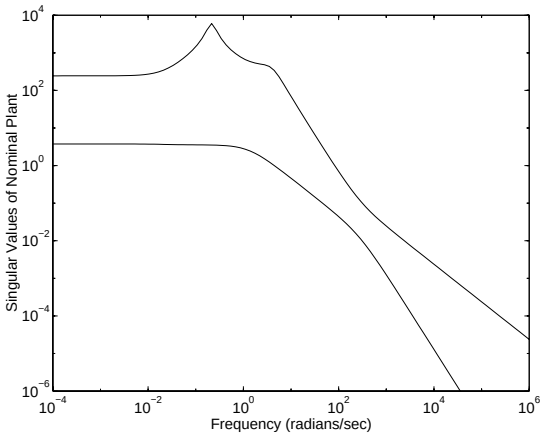


Figure 4.1: Block diagram of HIMAT and required feedback structure

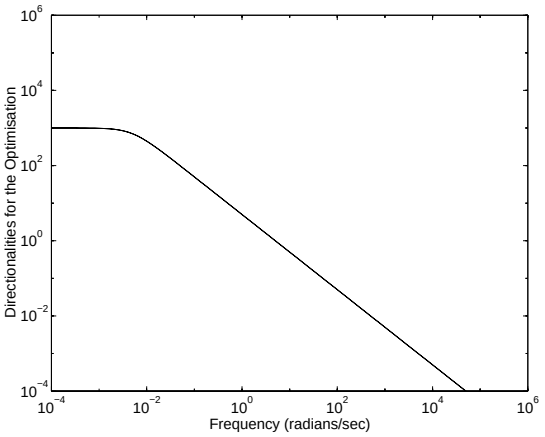
true plant is uncertain but is known to belong to the set $\{P_o(I + \Delta W_u) : \Delta \in \mathscr{RH}_\infty, \|\Delta\|_\infty \leq 1\}$. Similar to W_u, the transfer function matrix W_n reflects any 'a priori' knowledge about the frequency dependent size of the sensor noise. The state-space realisations of P_o, W_u and W_n are given in Balas et al. (1996). Here, only their singular value plots are given for the sake of the discussion.

It is required to maximise the performance weight W_p (in the sense of a suitable cost function that captures the desired closed-loop performance) subject to the existence of an internally stabilising controller K that guarantees robust performance with respect to this maximised weight. For a sensible control problem, W_p should be maximised in the low-frequency region, thereby achieving low-frequency disturbance rejection at the plant output. Figure 4.2 gives singular value plots of the nominal plant P_o, the directionality matrix Υ, the uncertainty weight W_u and the noise weight W_n used in this example. Note that the directionality functions are such that the performance weights are maximised in the low frequency region, and the weights W_u and W_n introduce high frequency model uncertainty and sensor noise respectively.

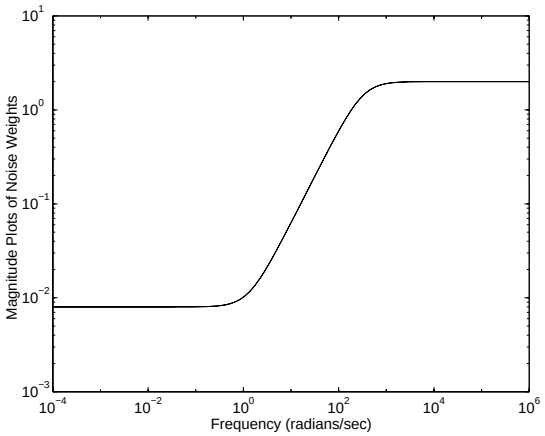
The algorithm proposed in Section 4.8 was programmed using MATLAB 5.3. The results obtained by using this algorithm are depicted in Figure 4.3 together with the results obtained from the standard D-K iterative procedure, so that comparison can be made. First of all, observe that unlike the μ-curve resulting from the standard D-K iterative procedure, the final μ-curve obtained by the proposed algorithm is flat across frequency and very close to unity, as depicted in Figure 4.3(a). This reflects that robust performance has been optimised. In fact, it can be seen from Figure 4.3(b) that the inverse performance weights synthesised by the proposed algorithm are everywhere less than those used by Balas et al. (1996) to explain the standard D-K iterative procedure. That is, a higher level of robust performance is attained by the proposed algorithm. Of course, different performance weights then lead to different D-scales and a different internally stabilising controller, as illustrated in Figures 4.3(c) and 4.3(d). The final controller synthesised by the proposed algorithm had 26 states and hence was of the same order as the scaled generalised plant, as expected.



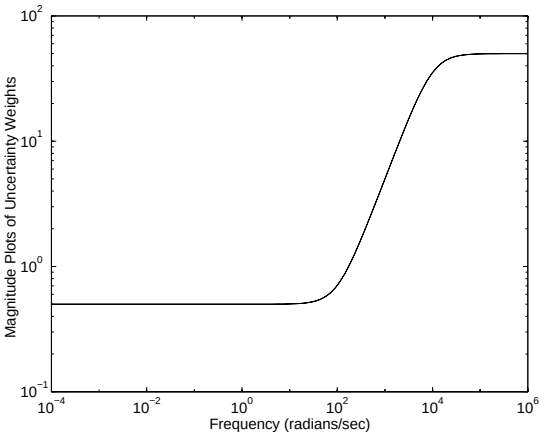
(a) Singular values of P_o



(b) Directionalities $|\upsilon_{11}(j\omega)|$ and $|\upsilon_{22}(j\omega)|$

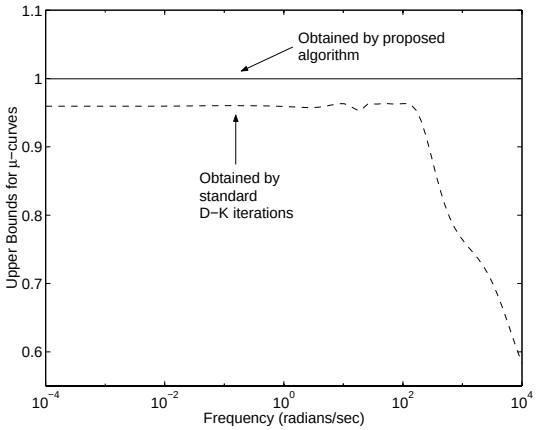


(c) Magnitude plots of $|w_{n_{11}}(j\omega)|$ and $|w_{n_{22}}(j\omega)|$

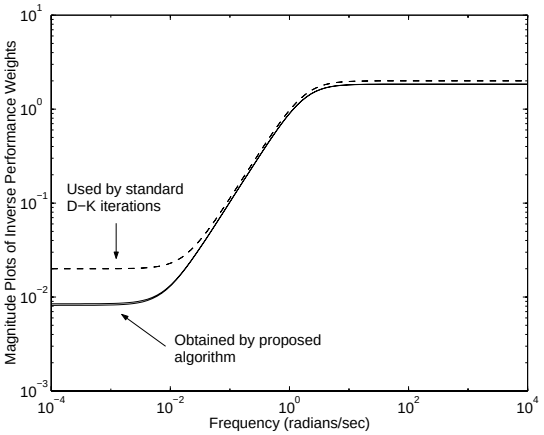


(d) Magnitude plots of $|w_{u_{11}}(j\omega)|$ and $|w_{u_{22}}(j\omega)|$

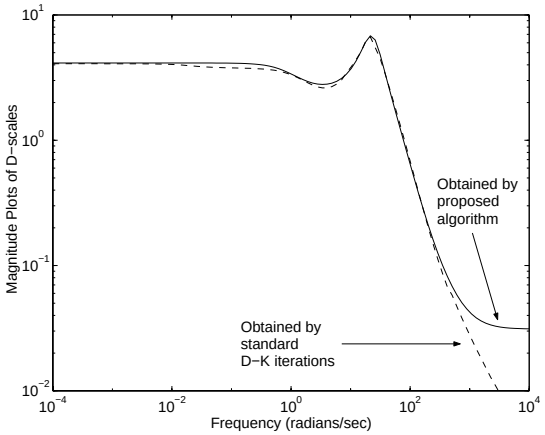
Figure 4.2: Singular value plots of input data



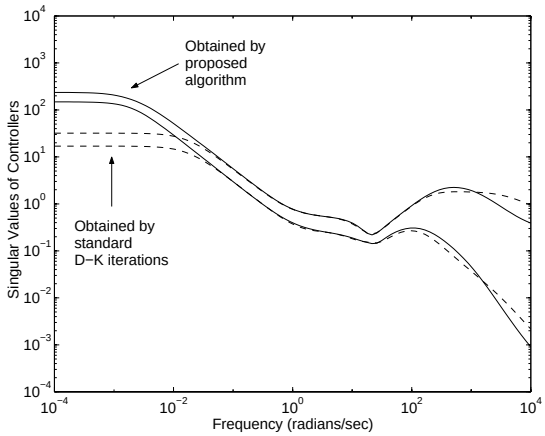
(a) Upper bounds for μ-curves



(b) Magnitude plots of $|w_{p_{11}}(j\omega)|^{-1}$, $|w_{p_{22}}(j\omega)|^{-1}$



(c) Magnitude plots of D-scales



(d) Singular values of controllers K

Figure 4.3: Singular value plots of results obtained by proposed algorithm and D-K iterations

4.10 Summary and Comments

The problem of maximising performance weights, subject to the existence of an internally stabilising controller that guarantees robust performance with respect to these maximised weights, was again posed as an optimisation problem in Section 4.1. This optimisation problem is very similar to the one posed in Chapter 3, the important difference being that the optimisation problem proposed here admits a solution algorithm based on state-space techniques.

Unfortunately, the μ constraint renders the posed optimisation problem difficult to solve. Hence, this constraint was replaced by a computationally tractable upper bound in Section 4.2. The chosen upper bound was justified by noting that it is necessary and sufficient for robust stability/performance against arbitrarily slow time-varying linear structured perturbations of norm no greater than unity. Each part of the resulting optimisation problem was studied in detail in the subsequent sections and equivalent LMI conditions were given. A solution algorithm for this optimisation problem was then proposed in Section 4.8. This solution algorithm first constructs good approximations of the optimal basis functions by solving one of the iterations pointwise in frequency and then uses these good approximations together with the state-space formulation of both optimisation problems to find exact solutions that guarantee the designed robust performance level.

As stated in the introductory paragraph, this chapter is an important generalisation of the pointwise work presented in Chapter 3. The algorithm presented here eliminates all of the disadvantages of the pointwise approach and considerably enhances the benefits of using such a method. More specifically,

I. the controller is no longer parametrised by a basis function and its order is guaranteed to be less than or equal to that of the scaled generalised plant,

II. the state-space conditions given here guarantee that $\mu < 1$ rather than simply give confidence,

III. maximisation of performance weights occurs over the entire frequency range from $-\infty$ to ∞,

IV. the solution algorithm given here is numerically more robust than the pointwise algorithm of Chapter 3 (this is because the Laguerre-like parametrisation used in Chapter 3 introduces all sorts of numerical difficulties when the designed controller is of high order),

V. the additional freedom in selecting a controller that achieves the designed robust performance level may be exploited by adding additional LMI constraints to simultaneously achieve other closed-loop objectives such as pole placement, $\mathscr{H}_2$-norm minimisation, etc.

In summary, this chapter presents a new and conceptually different method for performing μ-synthesis robust performance based designs and is hence a valuable alternative to the standard D-K iterative procedure. The approach presented here greatly simplifies the often long and tedious trial and error process of designing "good" performance weights directly. The results presented in this chapter were also published in Lanzon and Cantoni (2000).

Chapter 5

An Algorithm for $\mathscr{H}_\infty$ Loop-Shaping

5.1 Introduction

The $\mathscr{H}_\infty$ loop-shaping design procedure is an effective method for designing robust controllers and has been successfully used in a wide variety of applications. Motivation and justification for this was given in Section 2.5. The typical design framework is re-depicted in Figure 5.1 for ease of reference.

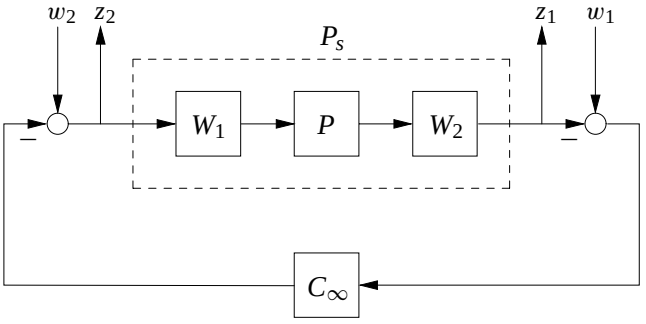


Figure 5.1: Typical $\mathscr{H}_\infty$ loop-shaping framework

Despite its success, the selection of loop-shaping weights W_1, W_2 to achieve a desired loop-shape is not always straightforward, especially for plants with strong cross-coupling. This is because it is not always clear how each element in the weights affects the singular values of the scaled nominal plant and the complexity of this relationship considerably increases when non-diagonal weights are used.

In addition, lack of design experience with loop-shaping concepts may lead to a designed loop-shape that does not achieve a sufficiently large robust stability margin $b_{opt}(P_s)$. In this case, the designer has to first determine the factors in the designed loop-shape that are giving rise to a small $b_{opt}(P_s)$ and then understand how to modify these factors (without compromising the specifications) in order to increase the robust stability margin. This may not be obvious and the designer may have to iterate between the selection of loop-shaping weights and the evaluation of $b_{opt}(P_s)$ several times before a sufficiently large value of $b_{opt}(P_s)$ is achieved. For instance, the designer must ensure that

the loop-gain is large around the frequencies and in the directions of open-loop unstable poles, small around the frequencies and in the directions of open-loop unstable zeros and that the loop-gain does not roll-off at a high rate around cross-over. If any one of these is violated, a small $b_{opt}(P_s)$ is obtained.

Besides achieving the desired loop-shape, the designer might also want to ensure that some of the closed-loop transfer function matrices appearing in Theorem 2.5.2 are small in appropriate frequency ranges. This can be achieved by ensuring that their corresponding upper bounds, given in the same theorem, are small in the required frequency ranges. However, these upper bounds depend on the condition number and the maximum/minimum singular values of the loop-shaping weights, and the standard $\mathscr{H}_\infty$ loop-shaping design procedure gives no direct handle on these variables. Thus, the designer can only check the size of these closed-loop transfer function matrices at the end of the design cycle and if they are not sufficiently small, the whole design must be repeated from the beginning.

All of this can be fairly time-consuming if done in an ad-hoc manner and although designers usually arrive at very good loop-shaping weights and controllers using their engineering insight and intuition, trial and error can never be guaranteed to yield the best possible results. The length of this iterative process strongly depends on the experience of the designer with loop-shaping concepts and on the cross-coupling present in the plant.

Consequently, it is believed that by combining several steps of the standard $\mathscr{H}_\infty$ loop-shaping design procedure into one optimisation problem, the design procedure can be made even more systematic and hence even easier to use in application. The proposed optimisation problem maximises the robust stability margin over the loop-shaping weights subject to constraints which ensure that the loop-shape and the singular values/condition numbers of the weights lie in pre-specified regions. Thus, loop-shaping weights, which can be required to have a diagonal or a non-diagonal structure, and a robustly stabilising controller are simultaneously synthesised by one algorithm in a systematic way. This algorithm enables the designer to quickly get a feel of what performance is achievable, determine whether non-diagonal weights would be beneficial and easily understand the tradeoffs involved in the particular problem at hand.

5.2 A New Optimisation Problem

In this section, a new optimisation problem is proposed which directly addresses the aforementioned difficulties. Before posing this optimisation problem, the following assumption is made.

Assumption: *Let the scaled nominal plant $P \in \mathscr{R}^{m\times n}$ be such that $m \geq n$.*

This assumption incurs no loss of generality but considerably simplifies notation. If the plant has strictly fewer outputs than inputs (i.e. $m < n$), then the dual problem to that shown in Figure 5.1 would be considered. That is, one would use the same optimisation framework proposed in this chapter with P^T replacing P. Then, the resulting pre-compensator for the original plant is given by W_2^T, the

resulting post-compensator for the original plant is given by W_1^T and the resulting robust stabilising controller for the shaped plant is given by C_∞^T.

Now, consider the following optimisation problem:

$$\max_{\substack{W_1, W_1^{-1} \in \mathscr{RH}_\infty \\ W_2, W_2^{-1} \in \mathscr{RH}_\infty}} b_{opt}(P_s)$$

subject to

(a) $|\underline{s}(j\omega)| < \sigma_i\big(P_s(j\omega)\big) < |\overline{s}(j\omega)| \quad \forall i, \omega,$

(b) $|\underline{w}_1(j\omega)| < \sigma_i\big(W_1(j\omega)\big) < |\overline{w}_1(j\omega)| \quad \text{and} \quad |\underline{w}_2(j\omega)| < \sigma_i\big(W_2(j\omega)\big) < |\overline{w}_2(j\omega)| \quad \forall i, \omega,$

(c) $\kappa\big(W_1(j\omega)\big) < |k_1(j\omega)| \quad \text{and} \quad \kappa\big(W_2(j\omega)\big) < |k_2(j\omega)| \quad \forall \omega,$

where the scaled nominal plant P is given and satisfies the above assumption, and $\underline{s}$, $\overline{s}$, $\underline{w}_i$, $\overline{w}_i$ and k_i ($i = 1, 2$) are SISO transfer functions specified by the designer such that:

(i) the frequency functions $|\underline{s}(j\omega)|$ and $|\overline{s}(j\omega)|$ are boundaries for an allowable loop-shape (see Figure 5.2),

(ii) the frequency functions $|\underline{w}_i(j\omega)|$ and $|\overline{w}_i(j\omega)|$ delimit the allowable region for the singular values of loop-shaping weight $W_i(j\omega)$ ($i = 1, 2$),

(iii) the frequency function $|k_i(j\omega)|$ bounds the condition number of loop-shaping weight $W_i(j\omega)$ ($i = 1, 2$).

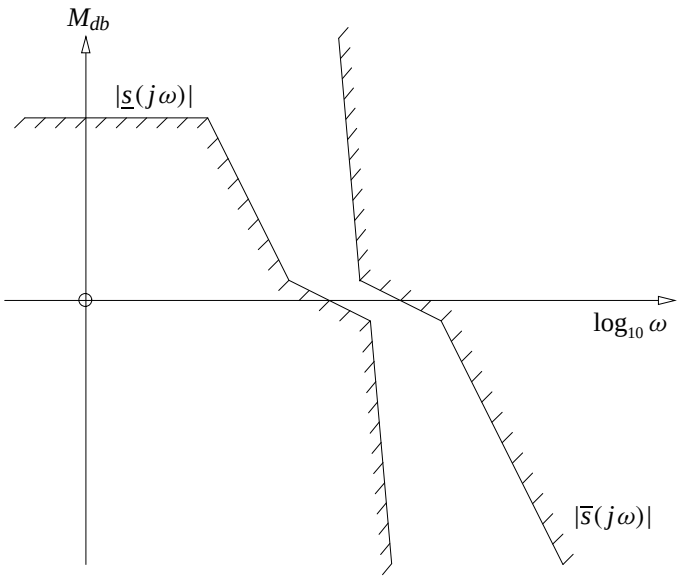


Figure 5.2: Typical loop-shape boundary

5.3 Rewriting the Optimisation Problem

The optimisation problem proposed in the previous section will now be rewritten into a form which is more suitable for a subsequent algorithm. First, however, the following set will be defined.

Definition 5.3.1 *Let the set of real diagonal matrices of dimension $n \times n$ be defined by:*

$$\mathbf{\Lambda}_n := \left\{ \operatorname*{diag}_{i=1}^{n} (x_i) \quad : \quad x_i \in \mathbb{R} \ \ \forall i \right\}.$$

This definition will be used at the end of this section. For the time being, note that the optimisation problem posed in Section 5.2 can be rewritten as:

$$\begin{gathered} \text{Minimise } \gamma \\ \text{such that } \exists\, W_1,\, W_2,\, C_\infty \text{ satisfying} \end{gathered}$$

(a) $W_1,\, W_1^{-1} \in \mathscr{RH}_\infty, \quad W_2,\, W_2^{-1} \in \mathscr{RH}_\infty$ and $[P_s,\, C_\infty]$ is internally stable,

(b) $\left\| \begin{bmatrix} P_s \\ I \end{bmatrix} (I - C_\infty P_s)^{-1} \begin{bmatrix} -C_\infty & I \end{bmatrix} \right\|_\infty < \gamma,$

(c) $|\underline{s}(j\omega)| < \sigma_i\big(P_s(j\omega)\big) < |\overline{s}(j\omega)| \quad \forall i, \omega,$

(d) $|\underline{w}_1(j\omega)| < \sigma_i\big(W_1(j\omega)\big) < |\overline{w}_1(j\omega)|$ and $|\underline{w}_2(j\omega)| < \sigma_i\big(W_2(j\omega)\big) < |\overline{w}_2(j\omega)| \quad \forall i, \omega,$

(e) $\kappa\big(W_1(j\omega)\big) < |k_1(j\omega)|$ and $\kappa\big(W_2(j\omega)\big) < |k_2(j\omega)| \quad \forall \omega.$

Then, using $P_s = W_2 P W_1$ and $C = W_1 C_\infty W_2$, it follows (after some algebra) that the above optimisation problem can be rewritten as:

$$\begin{gathered} \text{Minimise } \gamma_\omega \text{ at each } \omega \\ \text{such that } \exists\, W_1,\, W_2,\, C \text{ satisfying} \end{gathered}$$

(a) $W_1,\, W_1^{-1} \in \mathscr{RH}_\infty, \quad W_2,\, W_2^{-1} \in \mathscr{RH}_\infty$ and $[P,\, C]$ is internally stable,

(b) $\overline{\sigma}\left(\begin{bmatrix} W_2 & 0 \\ 0 & W_1^{-1} \end{bmatrix} \begin{bmatrix} 0 & P \\ 0 & I \end{bmatrix} \begin{bmatrix} I & P \\ C & I \end{bmatrix}^{-1} \begin{bmatrix} W_2^{-1} & 0 \\ 0 & W_1 \end{bmatrix} \right) (j\omega) < \gamma_\omega \quad \forall \omega,$

(c) $|\underline{s}(j\omega)| < \sigma_i\big(W_2(j\omega) P(j\omega) W_1(j\omega)\big) < |\overline{s}(j\omega)| \quad \forall i, \omega,$

(d) $\dfrac{1}{|\overline{w}_1(j\omega)|} < \sigma_i\big(W_1(j\omega)^{-1}\big) < \dfrac{1}{|\underline{w}_1(j\omega)|}$ and $|\underline{w}_2(j\omega)| < \sigma_i\big(W_2(j\omega)\big) < |\overline{w}_2(j\omega)| \quad \forall i, \omega,$

(e) $\kappa\big(W_1(j\omega)^{-1}\big) < |k_1(j\omega)|$ and $\kappa\big(W_2(j\omega)\big) < |k_2(j\omega)| \quad \forall \omega.$

Dropping the dependence on $(j\omega)$ for P, C, W_1 and W_2 in the interest of clarity, the above optimisation problem can be restated as:

$$\begin{gathered}\text{Minimise } \gamma_\omega^2 \text{ at each } \omega \\ \text{such that } \exists\, W_1,\, W_2,\, C \text{ satisfying}\end{gathered}$$

(a) $W_1,\, W_1^{-1} \in \mathscr{RH}_\infty, \quad W_2,\, W_2^{-1} \in \mathscr{RH}_\infty$ and $[P, C]$ is internally stable,

(b) $$\begin{bmatrix} 0 & P \\ 0 & I \end{bmatrix}^* \begin{bmatrix} W_2^* W_2 & 0 \\ 0 & W_1^{-*} W_1^{-1} \end{bmatrix} \begin{bmatrix} 0 & P \\ 0 & I \end{bmatrix} < \gamma_\omega^2 \begin{bmatrix} I & P \\ C & I \end{bmatrix}^* \begin{bmatrix} W_2^* W_2 & 0 \\ 0 & W_1^{-*} W_1^{-1} \end{bmatrix} \begin{bmatrix} I & P \\ C & I \end{bmatrix} \quad \forall \omega,$$

(c) $$|\underline{s}(j\omega)|^2 \left(W_1^{-*} W_1^{-1}\right) < P^* \left(W_2^* W_2\right) P < |\overline{s}(j\omega)|^2 \left(W_1^{-*} W_1^{-1}\right) \quad \forall \omega,$$

(d) $\forall \omega,\ \exists\, \underline{\xi}_{2\omega}, \overline{\xi}_{2\omega}$ satisfying
$$\underline{\xi}_{2\omega} I < W_2^* W_2 < \overline{\xi}_{2\omega} I, \quad \overline{\xi}_{2\omega} < |k_2(j\omega)|^2\, \underline{\xi}_{2\omega},$$
$$|\underline{w}_2(j\omega)|^2 < \underline{\xi}_{2\omega}, \quad \overline{\xi}_{2\omega} < |\overline{w}_2(j\omega)|^2,$$

(e) $\forall \omega,\ \exists\, \underline{\xi}_{1\omega}, \overline{\xi}_{1\omega}$ satisfying
$$\underline{\xi}_{1\omega} I < W_1^{-*} W_1^{-1} < \overline{\xi}_{1\omega} I, \quad \overline{\xi}_{1\omega} < |k_1(j\omega)|^2\, \underline{\xi}_{1\omega},$$
$$\frac{1}{|\overline{w}_1(j\omega)|^2} < \underline{\xi}_{1\omega}, \quad \overline{\xi}_{1\omega} < \frac{1}{|\underline{w}_1(j\omega)|^2}.$$

If the loop-shaping weights are required to have a diagonal structure, then the frequency functions $W_1^{-*} W_1^{-1}$ and $W_2^* W_2$ reduce to simple strictly positive diagonal real matrices at each frequency ω. Let these strictly positive frequency dependent diagonal matrices be denoted by $\Lambda_{1\omega}$ and $\Lambda_{2\omega}$ respectively (i.e. at each fixed ω, $0 < \Lambda_{1\omega} := W_1(j\omega)^{-*} W_1(j\omega)^{-1} \in \mathbf{\Lambda}_n$ and $0 < \Lambda_{2\omega} := W_2(j\omega)^* W_2(j\omega) \in \mathbf{\Lambda}_m$). Note that given any $\Lambda_{1\omega} \in \mathbf{\Lambda}_n$ with $\Lambda_{1\omega} > 0\ \forall \omega$ (resp. $\Lambda_{2\omega} \in \mathbf{\Lambda}_m$ with $\Lambda_{2\omega} > 0\ \forall \omega$), it is always possible to construct a diagonal weight W_1 (resp. W_2) that is a unit in $\mathscr{RH}_\infty$ and satisfies $W_1(j\omega)^{-*} W_1(j\omega)^{-1} = \Lambda_{1\omega}\ \forall \omega$ (resp. $W_2(j\omega)^* W_2(j\omega) = \Lambda_{2\omega}\ \forall \omega$) by fitting stable minimum phase transfer functions to each magnitude function on the main diagonal of $\Lambda_{1\omega}$ (resp. $\Lambda_{2\omega}$).

If, on the other hand, the loop-shaping weights are required to have a non-diagonal structure, then the frequency functions $W_1^{-*} W_1^{-1}$ and $W_2^* W_2$ are strictly positive non-diagonal complex hermitian matrices at each frequency ω. The problem, in this case, is significantly more difficult if approached directly. However, the technique developed by Papageorgiou and Glover (1997) may be used to simplify the problem. Building on that work, let $\hat{U}(s)$ and $\hat{V}(s)$ be units in $\mathscr{RL}_\infty$ that approximately interpolate the frequency-by-frequency unitary matrices containing the left and right singular vectors of the scaled nominal plant P. Then it is possible to parametrise:

- $W_1(j\omega)^{-*} W_1(j\omega)^{-1}$ by $\hat{V}(j\omega)^{-*} \Lambda_{1\omega} \hat{V}(j\omega)^{-1}$ for some $\Lambda_{1\omega} \in \mathbf{\Lambda}_n$ with $\Lambda_{1\omega} > 0\ \forall \omega$,
- $W_2(j\omega)^* W_2(j\omega)$ by $\hat{U}(j\omega) \Lambda_{2\omega} \hat{U}(j\omega)^*$ for some $\Lambda_{2\omega} \in \mathbf{\Lambda}_m$ with $\Lambda_{2\omega} > 0\ \forall \omega$,

with very little restriction. This is because the parameters in $\Lambda_{1\omega}$ and $\Lambda_{2\omega}$ are able to directly influence the singular values of $P(j\omega)$. The construction of $\hat{U}(s)$ and $\hat{V}(s)$ will be explored in more detail in the

next section. The interested reader is referred to Papageorgiou and Glover (1997) for a full exposition of the original idea. As before, note that given any $\Lambda_{1\omega} \in \mathbf{\Lambda}_n$ with $\Lambda_{1\omega} > 0\ \forall\omega$ (resp. $\Lambda_{2\omega} \in \mathbf{\Lambda}_m$ with $\Lambda_{2\omega} > 0\ \forall\omega$), it is always possible to construct a diagonal weight D_1 (resp. D_2) that is a unit in $\mathscr{RH}_\infty$ and satisfies $D_1(j\omega)^{-*}D_1(j\omega)^{-1} = \Lambda_{1\omega}\ \forall\omega$ (resp. $D_2(j\omega)^*D_2(j\omega) = \Lambda_{2\omega}\ \forall\omega$). Then, corresponding non-diagonal weights W_1 and W_2 that are units in $\mathscr{RH}_\infty$ are obtained by solving the following co-spectral and spectral factorisations:

$$\begin{aligned} W_1 W_1^{\sim} &= \hat{V} D_1 D_1^{\sim} \hat{V}^{\sim}, \\ W_2^{\sim} W_2 &= \hat{U} D_2^{\sim} D_2 \hat{U}^{\sim}. \end{aligned}$$

With the above argument in mind, it follows that the previous optimisation problem can be rewritten as:

$$\begin{gathered} \text{Minimise } \gamma_\omega^2 \text{ at each } \omega \\ \text{such that} \\ \exists \text{ a } C \text{ and } \forall\omega \text{ a } \Lambda_{1\omega} \in \mathbf{\Lambda}_n \text{ and a } \Lambda_{2\omega} \in \mathbf{\Lambda}_m \\ \text{satisfying} \end{gathered}$$

(a) $[P, C]$ is internally stable,

(b) $$\begin{bmatrix} 0 & P \\ 0 & I \end{bmatrix}^* \begin{bmatrix} \hat{U}\Lambda_{2\omega}\hat{U}^* & 0 \\ 0 & \hat{V}^{-*}\Lambda_{1\omega}\hat{V}^{-1} \end{bmatrix} \begin{bmatrix} 0 & P \\ 0 & I \end{bmatrix} < \gamma_\omega^2 \begin{bmatrix} I & P \\ C & I \end{bmatrix}^* \begin{bmatrix} \hat{U}\Lambda_{2\omega}\hat{U}^* & 0 \\ 0 & \hat{V}^{-*}\Lambda_{1\omega}\hat{V}^{-1} \end{bmatrix} \begin{bmatrix} I & P \\ C & I \end{bmatrix},$$

(c) $$|\underline{s}(j\omega)|^2 \left(\hat{V}^{-*}\Lambda_{1\omega}\hat{V}^{-1}\right) < P^*\left(\hat{U}\Lambda_{2\omega}\hat{U}^*\right)P < |\overline{s}(j\omega)|^2 \left(\hat{V}^{-*}\Lambda_{1\omega}\hat{V}^{-1}\right),$$

(d) $\exists\, \underline{\xi}_{2\omega}, \overline{\xi}_{2\omega}$ satisfying $$\begin{aligned} &\underline{\xi}_{2\omega} I < \left(\hat{U}\Lambda_{2\omega}\hat{U}^*\right) < \overline{\xi}_{2\omega} I, \quad \overline{\xi}_{2\omega} < |k_2(j\omega)|^2\, \underline{\xi}_{2\omega}, \\ &|\underline{w}_2(j\omega)|^2 < \underline{\xi}_{2\omega}, \quad \overline{\xi}_{2\omega} < |\overline{w}_2(j\omega)|^2, \end{aligned}$$

(e) $\exists\, \underline{\xi}_{1\omega}, \overline{\xi}_{1\omega}$ satisfying $$\begin{aligned} &\underline{\xi}_{1\omega} I < \left(\hat{V}^{-*}\Lambda_{1\omega}\hat{V}^{-1}\right) < \overline{\xi}_{1\omega} I, \quad \overline{\xi}_{1\omega} < |k_1(j\omega)|^2\, \underline{\xi}_{1\omega}, \\ &\frac{1}{|\overline{w}_1(j\omega)|^2} < \underline{\xi}_{1\omega}, \quad \overline{\xi}_{1\omega} < \frac{1}{|\underline{w}_1(j\omega)|^2}. \end{aligned}$$

Note that $\Lambda_{1\omega}$ and $\Lambda_{2\omega}$ are implicitly restricted to be strictly positive matrices at each frequency ω by constraints (d) and (e) above. Also, $\hat{U}(s) = I_m$ and $\hat{V}(s) = I_n$ when diagonal weights are required, whereas $\hat{U}(s)$ and $\hat{V}(s)$ are units in $\mathscr{RL}_\infty$ that approximately interpolate the frequency-by-frequency unitary matrices containing the left and right singular vectors respectively of the scaled nominal plant P when non-diagonal weights are required. Furthermore, observe that the above problem is a quasi-convex optimisation problem if the controller C is held fixed at an internally stabilising controller for the scaled nominal plant P.

5.4 Construction of $\hat{U}(s)$ and $\hat{V}(s)$

There are a variety of reasons for requiring both diagonal and non-diagonal loop-shaping weights. Diagonal weights are simple to construct, easy to tune by hand and very clear in the way they affect the nominal plant. Furthermore, diagonal weights are very easy to interpolate in gain-scheduling. However, diagonal weights suffer from the problem that they do not directly affect the singular values of the nominal plant. In fact, examples have shown that for plants with strong cross-coupling between the channels, diagonal loop-shaping weights are not very effective and non-diagonal weights have to be used instead. The task of choosing such loop-shaping weights can be very difficult and time consuming if done in an ad-hoc manner. In view of this, Papageorgiou and Glover (1997) proposed a systematic procedure to assist such designs. The reader is referred there for a detailed exposition of the original idea. This section is based on that work, only outlines the construction of $\hat{U}(s)$ and $\hat{V}(s)$ and is given here for completeness.

First recall that the scaled nominal plant $P \in \mathscr{R}^{m\times n}$ is assumed to have more outputs than inputs (i.e. $m \geq n$). Then, the following frequency-by-frequency decomposition[1] of the scaled nominal plant $P(j\omega)$ is computed on a sufficiently dense grid:

$$P(j\omega_k) = U_{\omega_k}\Sigma_{\omega_k}V^*_{\omega_k},$$

where $U_{\omega_k} \in \mathbb{C}^{m\times n}\ \forall\omega_k$ and $V_{\omega_k} \in \mathbb{C}^{n\times n}\ \forall\omega_k$ are matrices with orthonormal columns containing the left and right singular vectors of $P(j\omega_k)$, and $\Sigma_{\omega_k} \in \mathbb{R}^{n\times n}\ \forall\omega_k$ is a diagonal matrix containing the singular values of $P(j\omega_k)$. As explained in Papageorgiou and Glover (1997), it is important to ensure that the ordering of the singular values is uniform across frequency. By uniform it is meant that the ordering of the singular values at each frequency is the same as the ordering of the singular values at zero frequency. This will ensure correct ordering of the singular vectors in U_{ω_k} and V_{ω_k} as frequency varies.

However, observe that the matrices of singular vectors U_{ω_k} and V_{ω_k} are not unique in the above decomposition and hence can vary discontinuously with frequency. Any unitary matrix (possibly a different one at each frequency ω_k) that commutes with Σ_{ω_k} can post-multiply U_{ω_k} and V_{ω_k} to give the same type of decomposition. Thus, one could try to select these unitary matrices (up to which U_{ω_k} and V_{ω_k} are unique) in such a way so that U_{ω_k} and V_{ω_k} vary continuously with frequency. This is possible and indeed easy when the singular values are distinct, as these unitary matrices reduce to simple *diagonal* all-pass factors (Horn and Johnson, 1996, Lemma 7.3.1). However, at frequencies where repeated singular values occur, this is simply not possible as there are multiple vectors (at this single frequency point) which can be used as perfectly valid singular vectors. This is not of major difficulty in the method of Papageorgiou and Glover (1997), as continuity of singular vectors can be ensured at every frequency except the finite number of frequencies, if any, where the plant has repeated singular values.

[1] This is very similar to Singular Value Decomposition and can be easily obtained from it.

After determining the diagonal all-pass factors that make the singular vectors in U_{ω_k} and V_{ω_k} piecewise continuous, real-rational (possibly high order) transfer function matrices $\hat{U}(s)$ and $\hat{V}(s)$ are found which approximately interpolate U_{ω_k} and V_{ω_k} in the sense that:

$$\begin{aligned} \overline{\sigma}\left(\hat{U}(j\omega_k) - U_{\omega_k}\right) &\leq \delta^u_{\omega_k} \\ \text{and} \quad \overline{\sigma}\left(\hat{V}(j\omega_k) - V_{\omega_k}\right) &\leq \delta^v_{\omega_k}, \end{aligned}$$

for each frequency ω_k. The above approximations can be done element-by-element, as suggested in Papageorgiou and Glover (1997), or using more elaborate techniques which immediately find real-rational transfer function matrix approximations. If $\delta^u_{\omega_k}$ and $\delta^v_{\omega_k}$ are small, then it is easy to show that all the singular values of $\hat{U}(j\omega_k)$ and $\hat{V}(j\omega_k)$ are close to unity. This follows from:

$$\begin{aligned} 1 - \delta^u_{\omega_k} \;\leq\; \underline{\sigma}\left(\hat{U}(j\omega_k)\right) \quad &\leq \quad \overline{\sigma}\left(\hat{U}(j\omega_k)\right) \;\leq\; 1 + \delta^u_{\omega_k}, \\ 1 - \delta^v_{\omega_k} \;\leq\; \underline{\sigma}\left(\hat{V}(j\omega_k)\right) \quad &\leq \quad \overline{\sigma}\left(\hat{V}(j\omega_k)\right) \;\leq\; 1 + \delta^v_{\omega_k}. \end{aligned}$$

Now, if the scaled nominal plant $P \in \mathscr{R}^{m\times n}$ is tall (i.e. $m > n$), then $\hat{U}(s)$ is a tall transfer function matrix. The following lemma (Zhou et al., 1996, Theorem 13.32) can be used to construct a complementary factor[2] for this $\hat{U}(s)$. The suppositions in this lemma are easily satisfied in practise as they simply state that $\hat{U}(s)$ should have no zeros on the imaginary axis.

Lemma 5.4.1 *Suppose* $\hat{U}(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \in \mathscr{R}^{m\times n}$ *is such that* $m > n$ *and* $\hat{U}(j\omega)^*\hat{U}(j\omega) > 0$ *for all* $\omega \in \mathbb{R} \cup \{\infty\}$. *Furthermore, assume that* (A, B) *is stabilisable and that* $\begin{bmatrix} A - j\omega I & B \\ C & D \end{bmatrix}$ *has full column rank for all* $\omega \in \mathbb{R}$. *Then a complementary factor* $\hat{U}_\perp(s)$ *is given by:*

$$\hat{U}_\perp(s) = \left[\begin{array}{c|c} A + BF & -X^\dagger C^T D_\perp \\ \hline C + DF & D_\perp \end{array}\right],$$

where

(i) $D_\perp$ *is the orthogonal complement of* D,

(ii) $F = -R^{-1}(B^T X + D^T C)$ *with* $R = D^T D > 0$,

(iii) $X = X^T \geq 0$ *is the unique real stabilising solution to*

$$\left(A - BR^{-1}D^T C\right)^T X + X\left(A - BR^{-1}D^T C\right) - X\left(BR^{-1}B^T\right)X + C^T\left(I - DR^{-1}D^T\right)C = 0.$$

Consequently, both $\hat{U}(s)$ and $\hat{V}(s)$ are now completely constructed and are units in $\mathscr{RL}_\infty$.

[2] A transfer function matrix $\hat{U}_\perp(s)$ is called a *complementary factor* of a tall transfer function matrix $\hat{U}(s)$ satisfying $\hat{U}(j\omega)^*\hat{U}(j\omega) > 0\ \forall\omega \in \mathbb{R} \cup \{\infty\}$ if $\hat{U}_\perp^\sim(s)\left[\hat{U}(s) \quad \hat{U}_\perp(s)\right] = \left[0 \quad I\right]$.

5.5 Solution Algorithm

This section presents a sub-optimal iterative algorithm for solving the optimisation problem proposed in Section 5.2. This optimisation problem has been reformulated in several different ways in Section 5.3 and the following sub-optimal iterative algorithm is based on the last reformulation given in that section. Analogous to D-K iterations and other solution methods for these types of optimisations, an iterative algorithm must be used since the posed problem is not simultaneously convex in all variables.

Inputs to the algorithm:

- Scaled nominal plant P satisfying the assumption stated in Section 5.2,
- Frequency functions $|\underline{s}(j\omega)|$ and $|\overline{s}(j\omega)|$ that are boundaries for an allowable loop-shape,
- Frequency functions $|\underline{w}_i(j\omega)|$ and $|\overline{w}_i(j\omega)|$ that delimit the allowable region for the singular values of loop-shaping weight $W_i(j\omega)$ $(i = 1, 2)$,
- Frequency function $|k_i(j\omega)|$ that bounds the condition number of loop-shaping weight $W_i(j\omega)$ $(i = 1, 2)$.

The solution algorithm:

1. Find a controller $C_0^\star$ such that the interconnection $[P, C_0^\star]$ is internally stable. This controller can be found by a variety of standard techniques and provides a feasible initial starting point for the algorithm.

   Furthermore, let $\hat{U}(s) = I_m$ (resp. $\hat{V}(s) = I_n$) if a diagonal loop-shaping weight W_2 (resp. W_1) is required, or let $\hat{U}(s)$ (resp. $\hat{V}(s)$) be constructed as described in Section 5.4 if a non-diagonal weight W_2 (resp. W_1) is required.

   Set $i = 0$, where i denotes the iteration number, and let $\epsilon^\star_{\max,0} = -1$.

2. Increment i by 1.

3. Solve the following quasi-convex optimisation problem *at each frequency* ω:

$$\begin{aligned}
&\text{Minimise } \gamma_\omega^2 \\
&\text{such that } \exists\, \Lambda_{1\omega} \in \mathbf{\Lambda}_n,\ \Lambda_{2\omega} \in \mathbf{\Lambda}_m \text{ satisfying} \\
\text{(a)}\ &\begin{bmatrix} 0 & \hat{U}^*P\hat{V} \\ 0 & I \end{bmatrix}^* \begin{bmatrix} \Lambda_{2\omega} & 0 \\ 0 & \Lambda_{1\omega} \end{bmatrix} \begin{bmatrix} 0 & \hat{U}^*P\hat{V} \\ 0 & I \end{bmatrix} \\
&\qquad < \gamma_\omega^2 \begin{bmatrix} I & \hat{U}^*P\hat{V} \\ \hat{V}^{-1}C_{i-1}^\star\hat{U}^{-*} & I \end{bmatrix}^* \begin{bmatrix} \Lambda_{2\omega} & 0 \\ 0 & \Lambda_{1\omega} \end{bmatrix} \begin{bmatrix} I & \hat{U}^*P\hat{V} \\ \hat{V}^{-1}C_{i-1}^\star\hat{U}^{-*} & I \end{bmatrix},
\end{aligned}$$

(b) $|\underline{s}(j\omega)|^2\,\Lambda_{1\omega} \;<\; (\hat{U}^*P\hat{V})^*\Lambda_{2\omega}\,(\hat{U}^*P\hat{V}) \;<\; |\overline{s}(j\omega)|^2\,\Lambda_{1\omega}$,

(c) $\exists\,\underline{\xi}_{2\omega},\overline{\xi}_{2\omega}\in\mathbb{R} \quad : \quad \underline{\xi}_{2\omega}\big(\hat{U}^*\hat{U}\big)^{-1} < \Lambda_{2\omega} < \overline{\xi}_{2\omega}\big(\hat{U}^*\hat{U}\big)^{-1}$,

$$\overline{\xi}_{2\omega} < |k_2(j\omega)|^2\,\underline{\xi}_{2\omega}, \quad |\underline{w}_2(j\omega)|^2 < \underline{\xi}_{2\omega}, \quad \overline{\xi}_{2\omega} < |\overline{w}_2(j\omega)|^2,$$

(d) $\exists\,\underline{\xi}_{1\omega},\overline{\xi}_{1\omega}\in\mathbb{R} \quad : \quad \underline{\xi}_{1\omega}\big(\hat{V}^*\hat{V}\big) < \Lambda_{1\omega} < \overline{\xi}_{1\omega}\big(\hat{V}^*\hat{V}\big)$,

$$\overline{\xi}_{1\omega} < |k_1(j\omega)|^2\,\underline{\xi}_{1\omega}, \quad \frac{1}{|\overline{w}_1(j\omega)|^2} < \underline{\xi}_{1\omega}, \quad \overline{\xi}_{1\omega} < \frac{1}{|\underline{w}_1(j\omega)|^2}.$$

Note that $C^\star_{i-1}$ is the controller synthesised in the previous iteration. The above minimisation problem can thus be easily solved using LMI routines.

Denote by $\Lambda^\star_{1\omega}$ and $\Lambda^\star_{2\omega}$ the values of $\Lambda_{1\omega}$ and $\Lambda_{2\omega}$ that achieve the minimum of the above optimisation problem at each frequency ω.

4. Construct *diagonal* transfer function matrices $D^\star_1(s)$ and $D^\star_2(s)$ that are units in $\mathscr{RH}_\infty$ by fitting stable minimum phase transfer functions to each magnitude function on the main diagonal of $\big(\Lambda^\star_{1\omega}\big)^{-\frac{1}{2}}$ and $\big(\Lambda^\star_{2\omega}\big)^{\frac{1}{2}}$ respectively.

5. Solve the following co-spectral and spectral factorisations (Francis, 1987, Section 7.3)

$$\begin{aligned}\big(W^\star_{1,i}\big)\big(W^\star_{1,i}\big)^\sim &= \big(\hat{V}\big)\,\big(D^\star_1\big)\big(D^\star_1\big)^\sim\,\big(\hat{V}\big)^\sim,\\ \big(W^\star_{2,i}\big)^\sim\big(W^\star_{2,i}\big) &= \big(\hat{U}\big)\,\big(D^\star_2\big)^\sim\big(D^\star_2\big)\,\big(\hat{U}\big)^\sim,\end{aligned}$$

to obtain loop-shaping weights $W^\star_{1,i}(s)$ and $W^\star_{2,i}(s)$ that are units in $\mathscr{RH}_\infty$ and have the required structure. Note that the above factorisations reduce to simply setting $W^\star_{1,i}(s) = D^\star_1(s)$ (resp. $W^\star_{2,i}(s) = D^\star_2(s)$) when $\hat{V}(s) = I_n$ (resp. $\hat{U}(s) = I_m$).

6. Compute $b_{opt}\big(W^\star_{2,i}PW^\star_{1,i}\big)$ as detailed in Glover and McFarlane (1989) and let this value be denoted by $\epsilon^\star_{\max,i}$. Furthermore, synthesise a controller $C^\star_{\infty,i}$ that achieves $b\big(W^\star_{2,i}PW^\star_{1,i}, C^\star_{\infty,i}\big) = \epsilon^\star_{\max,i}$ using the state-space formula given in (Glover, 1984, Theorem 6.3). Finally, let $C^\star_i = W^\star_{1,i}C^\star_{\infty,i}W^\star_{2,i}$.

   Note that all these calculations are easily computed using well-known formulae which are coded in commercially available software.

7. Evaluate $(\epsilon^\star_{\max,i} - \epsilon^\star_{\max,i-1})$. If this difference (which is always positive) is very small and has remained very small for the last few iterations, then EXIT. Otherwise return to Step 2.

Outputs from the algorithm: (after i iterations)

- The largest value of $b_{opt}(P_s)$ obtained by the algorithm in the variable $\epsilon^\star_{\max,i}$,
- Loop-shaping weights $W^\star_{1,i}(s)$ and $W^\star_{2,i}(s)$ that achieve this maximised robust stability margin,
- A controller $C^\star_{\infty,i}(s)$ that achieves $b\big(W^\star_{2,i}PW^\star_{1,i}, C^\star_{\infty,i}\big) = \epsilon^\star_{\max,i}$.

Note that the algorithm presented above is an ascent algorithm. By this it is meant that the value $\epsilon^{\star}_{\max,i}$ is monotonically non-decreasing as i increases and that at each iteration i, the reciprocal of the square-root of the minimum cost γ^2_ω obtained in Step 3 of the algorithm is greater than or equal to $\epsilon^{\star}_{\max,i-1}$ for all frequency ω. Note however that iterative algorithms as the one presented above cannot be guaranteed to converge to the *global maximum*. Only monotonicity properties can be guaranteed.

5.6 Numerical Example

The algorithm proposed in Section 5.5 will now be illustrated by a numerical example. Two different designs will be considered — one using a diagonal pre-compensator and the other using a non-diagonal pre-compensator. The post-compensator will be held fixed in both design cases for simplicity of illustration.

The plant used to demonstrate the applicability of the proposed algorithm is a scaled-down version of the High Incidence Research Model (HIRM) developed by the Defence Evaluation and Research Agency in Bedford, UK. A physical model of this was constructed at the Department of Engineering of the University of Cambridge in order to investigate problems associated with the control of air-vehicles at high angles of attack. This model was flown in a wind-tunnel at a speed of 30 `m/s` and at a fixed pitch angle of 30°. The following model was identified from measurements taken off the experimental rig:

$$\hat{P}_{nom}(s) = \left[\begin{array}{cccc|cc} 0 & 1 & 0 & 0 & 0 & 0 \\ -23.8 & -3.36 & 4.60 & -0.239 & 1.67 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -16.8 & -0.0248 & 22.8 & -0.916 & 0 & 1.39 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{array}\right].$$

This linear time-invariant model of the plant has two inputs, roll and yaw thrusters, and four states, roll and yaw angles and their corresponding rates. The measured outputs that are used for feedback are roll and yaw angles. The torque produced by the thrusters is measured in `Nm` and the angles in `rad`. Each of the actuators which drives the thrusters may be modelled by a 50 `rad/s` low-pass filter in series with a first-order Padé approximation of a 0.05 `s` time delay. This actuator model is *not* included in $\hat{P}_{nom}(s)$ above. Thus, the nominal open-loop plant P_{nom} with the actuator model included has 8 states. Further details about the HIRM can be found in Papageorgiou and Glover (1999a) and Halsey (2000).

The nominal plant P_{nom} is unstable and non-minimum phase. The RHP pole, due to an unstable yaw mode, has a natural frequency of about 4 `rad/s` and the RHP zero, coming from the Padé approximation, has a natural frequency of about 40 `rad/s`. These restrict the closed-loop bandwidth of each channel to lie between 4 and 40 `rad/s`. The nominal plant P_{nom} requires no scaling as the

units of the input and output channels are already consistent (hence $P = P_{nom}$). The singular values of the scaled nominal plant P are plotted in Figure 5.3.

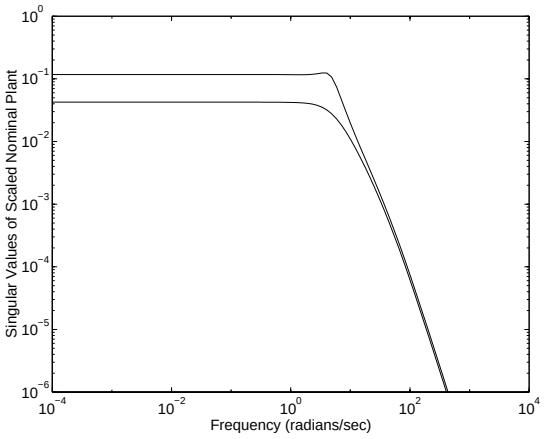


Figure 5.3: Singular values of scaled nominal plant P

Besides the scaled nominal plant P, the algorithm presented in Section 5.5 also requires loop-shape boundaries $|\underline{s}(j\omega)|$ and $|\overline{s}(j\omega)|$. The boundaries used in this example are shown in Figure 5.4. These

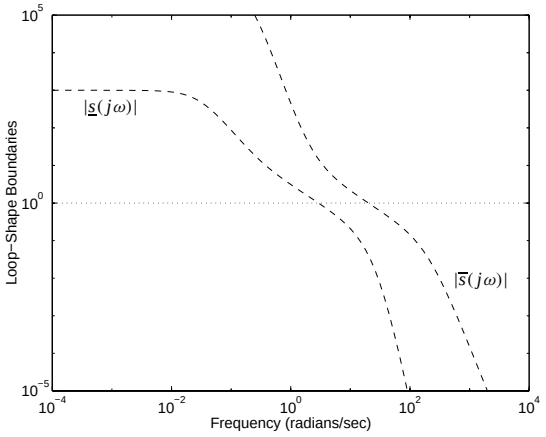


Figure 5.4: Frequency plots of loop-shape boundaries

loop-shape boundaries are usually determined from the closed-loop performance specifications. For example, the bandwidth determines the rise time and the low-frequency gain determines the sensitivity reduction and hence the reference tracking capabilities.

The post-compensator W_2 was fixed in both design cases (i.e. no optimisation was required on W_2, as will usually be the case) and was chosen, for sensor noise rejection, to be a first-order low-pass filter with a corner frequency of 300 `rad/s` on each output channel. The frequency functions $|\underline{w}_1(j\omega)|$ and $|\overline{w}_1(j\omega)|$ that delimit the allowable region for the singular values of the pre-compensator W_1 were chosen to be 10^{-10} and 10^{10} respectively, whereas the frequency function $|k_1(j\omega)|$ that bounds the condition number of the pre-compensator W_1 was chosen to be 20. These bounds were chosen as stated

above for ease of specification and never became active in this particular design example. In other design problems (say, for plants with very large condition number and/or stringent requirements on gains from plant output disturbances to control signal or from plant input disturbances to output signal), it may be necessary to specify more complicated, perhaps frequency dependent, bounds $|\underline{w}_1(j\omega)|$, $|\overline{w}_1(j\omega)|$ and $|k_1(j\omega)|$ to satisfy the problem specifications.

The algorithm presented in Section 5.5 was coded up in MATLAB 5.3 and this code was used to design two loop-shaping controllers for the same plant. In first design case, a diagonal pre-compensator W_1 was synthesised, whereas in the second design case, a non-diagonal pre-compensator W_1 was synthesised. Table 5.1 summarises the results obtained for both design cases. A detailed description is given below.

	Diagonal W_1	Non-diagonal W_1
No. of iterations for convergence	4 iterations	4 iterations
Time taken for convergence	$\approx$ 5 minutes	$\approx$ 6 minutes
Order of weight W_1	4 states + 4 states	38 states (model reduced to 12 states)
Condition number of weight W_1	$< 3\ \forall\omega$	$< 4\ \forall\omega$
Order of controller C_∞	17 states	46 states (model reduced to 11 states)
Order of C after model reduction	17 states	17 states
Robust stability margin	0.368	0.382

Table 5.1: Comparison of results for diagonal and non-diagonal pre-compensator designs

For the diagonal pre-compensator design, four iterations were found to be sufficient for convergence of the algorithm and this took around 5 minutes on a 400 MHz Pentium II PC. The pre-compensator W_1 synthesised by the algorithm had 8 states (i.e. a 4th-order weight in each channel) and hence the synthesised controller C_∞ had 17 states (i.e. 8 states due to W_1, 2 states due to W_2 and another 8 states due to P less 1 state due to optimal controller synthesis). However, it was observed that the controller $C = W_1 C_\infty W_2$ could easily be model reduced to 17 states with very minor changes in the loop-gain. The singular values of this pre-compensator W_1 are plotted in Figure 5.5(a), the corresponding loop-shape depicted in Figure 5.5(c) and the singular values of the simultaneously synthesised robustly stabilising controller C_∞ illustrated in Figure 5.5(e). It can be easily seen that the loop-shape lies in the pre-specified region and that it rolls-off at a very small rate around cross-over. Furthermore, both the loop-shaping weight W_1, which is clearly bi-proper, and the controller C_∞ introduce some phase lead around cross-over to improve the robust stability margin. The final robust stability margin $b_{opt}(P_s)$ obtained by the algorithm for this loop-shaping design is 0.368, which is immediately very good. It is probably also worth mentioning that the condition number of W_1 was everywhere less than 3. All these important design properties were quickly obtained by one systematic algorithm.

For the non-diagonal pre-compensator design, four iterations were again found to be sufficient for convergence of the algorithm and this took just under 6 minutes on a 400 MHz Pentium II PC.

The pre-compensator W_1 synthesised by the algorithm had 38 states, due to the high order fitting required to accurately reproduce the matrix of singular vectors, but this could easily be model reduced to 12 states without any significant deterioration in the loop-shape and the correspondingly achieved robust stability margin. Furthermore, the synthesised robustly stabilising controller C_∞ had 46 states, which again could easily be model reduced to 11 states without any significant deterioration in the robust stability margin. The singular values of this pre-compensator W_1 are plotted in Figure 5.5(b), the corresponding loop-shape depicted in Figure 5.5(d) and the singular values of the simultaneously synthesised robustly stabilising controller C_∞ illustrated in Figure 5.5(f). Again, it can be easily seen that the loop-shape lies in the pre-specified region, rolls-off at a very small rate around cross-over and has both singular values overlapping each other. This is because the non-diagonal pre-compensator W_1 can completely influence the singular values of the shaped plant P_s and maximising the robust stability margin leads to this overlapping of the singular values. The final robust stability margin $b_{opt}(P_s)$ obtained by the algorithm for this loop-shaping design is 0.382, which is again very good and is slightly better than that obtained in the diagonal pre-compensator design case. This is of course due to the fact that a non-diagonal weight W_1 has more "authority" on the singular values of the shaped plant P_s than a diagonal weight. Again, the condition number of W_1 is small, in fact it is less than 4 at all frequencies, as expected.

Although the synthesised weight and controller were of much higher order in the non-diagonal pre-compensator design case, the achieved robust stability margin was slightly higher too. It was also noted that the controller $C = W_1 C_\infty W_2$ could be model reduced to 17 states in *both* design cases with negligible deterioration of the designed properties. Thus, one should not be scared 'a priori' of a non-diagonal design simply on the grounds that it may give higher order weights, as these can often be model reduced afterwards. However, if diagonal weights achieve approximately the same robust stability margin as non-diagonal weights, then it is futile to use more complicated weights to achieve very minor improvements.

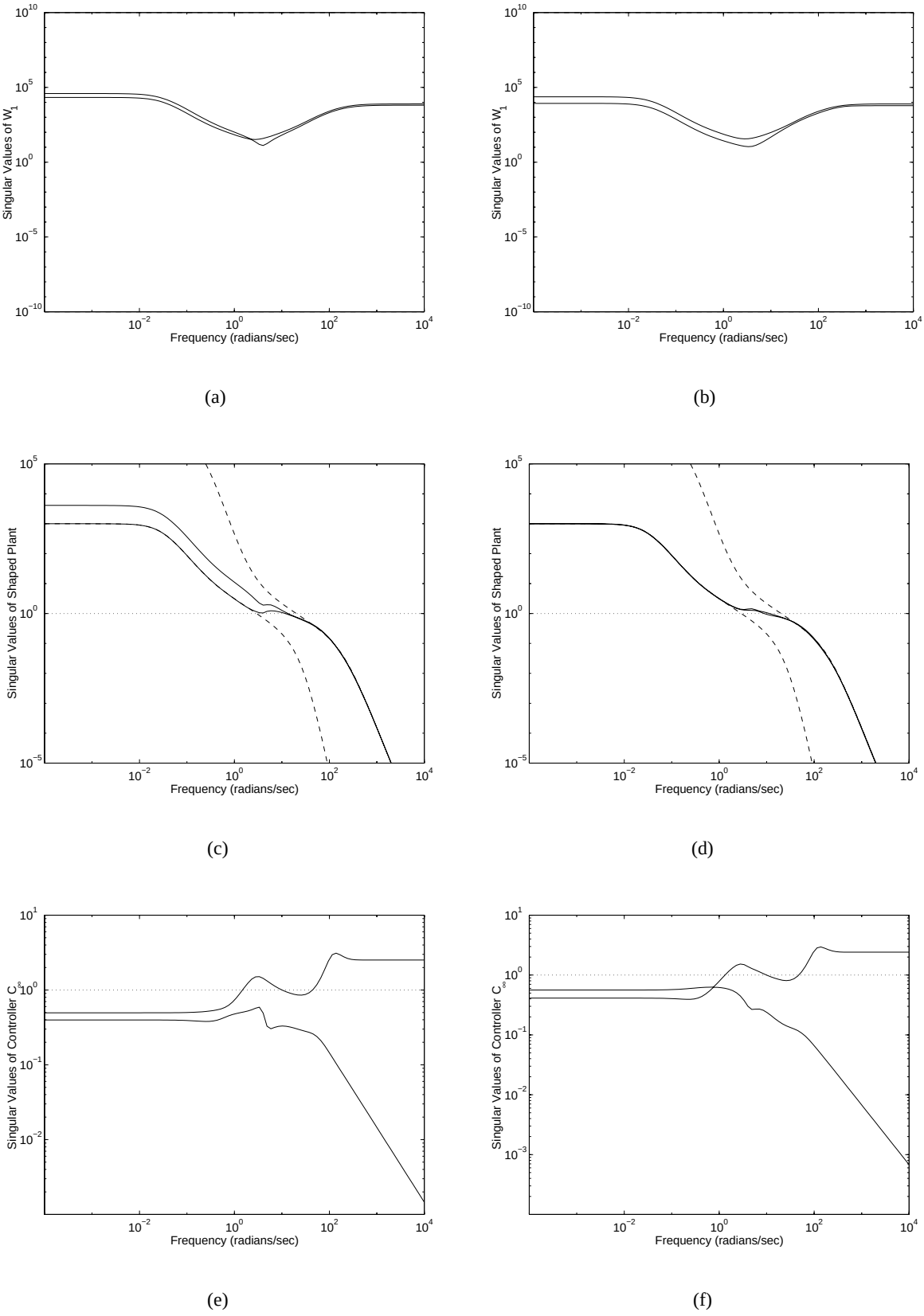


Figure 5.5: Plots on the left are for diagonal W_1 and on the right for non-diagonal W_1

5.7 Summary and Comments

An algorithm for the simultaneous synthesis of weights and controllers in $\mathscr{H}_\infty$ loop-shaping has been presented. This algorithm requires pre-specified allowable regions for the loop-shape and the singular values/condition numbers of the weights. These regions are usually determined from the closed-loop performance specifications. This step is often not easy as it typically involves translating time domain specifications into the frequency domain. Although the proposed algorithm does *not* improve the standard $\mathscr{H}_\infty$ loop-shaping design procedure in this respect, it does however introduce several other benefits:

I. Loop-shaping weights are synthesised by a systematic algorithm to immediately have the required structure (i.e. diagonal/non-diagonal), satisfy the designer-specified constraints on their singular values and condition numbers and achieve a loop-shape which falls in a pre-specified region and maximises the robust stability margin $b_{opt}(P_s)$. A robustly stabilising controller C_∞ is also synthesised by the algorithm to achieve this maximised robust stability margin.

II. Specifying acceptable regions rather than exact weights and hence an exact loop-shape makes it more difficult for inexperienced designers to obtain very bad loop-shapes. This is because the algorithm always tries to select a loop-shape, within the acceptable regions, that maximises $b_{opt}(P_s)$.

III. This algorithm gives direct handles that enable the designer to constrain the singular values and condition numbers of the loop-shaping weights in appropriate frequency ranges.

IV. Since the algorithm is not time-consuming, the designer can quickly determine whether a diagonal weight design is sufficiently good or if non-diagonal weights are necessary. Furthermore, it allows the designer to concentrate on more fundamental design issues than simply finding weights that achieve the desired loop-shape.

Chapter 6

Concluding Remarks

By way of conclusion, the main contributions of this thesis are summarised and suggestions for possible future research directions are outlined. The central idea behind this thesis was to provide optimisation algorithms which assist the designer in selecting weights for robust control techniques. The proposed algorithms also partly address the question of determining the achievable performance for a particular control system design.

6.1 Contributions

The main contributions of this thesis are summarised below.

In the area of μ-synthesis

- A new and conceptually different method for performing μ-synthesis robust performance based designs is proposed in this thesis. It is shown that this approach greatly simplifies the often long and tedious trial and error process of designing "good" performance weights directly and is hence a valuable alternative to the standard D-K iterative procedure. The proposed algorithm maximises performance weights in the frequency ranges of interest subject to the existence of an internally stabilising controller that guarantees robust performance with respect to these maximised weights.

- The cost function chosen in Chapters 3 and 4 to ensure the desired maximisation of performance weights is not obvious. It has the form of the reciprocal of the cumulative weighted sum of reciprocals of each frequency dependent performance weight. This form ensures that the performance weights resulting from the proposed optimisation problem have the desired characteristics.

In the area of $\mathscr{H}_\infty$ loop-shaping

- The optimisation problem proposed in this thesis for $\mathscr{H}_\infty$ loop-shaping robust performance based designs combines various steps of the standard $\mathscr{H}_\infty$ loop-shaping design procedure. It is quali-

tatively shown that this optimisation problem facilitates the design of loop-shaping weights and controllers by making the design procedure more systematic and hence even easier to use in application. This optimisation problem maximises the robust stability margin subject to constraints on the loop-shape and the singular values/condition numbers of the weights.

- The proposed solution algorithm for this optimisation problem provides a unified framework for the synthesis of both diagonal and non-diagonal loop-shaping weights. This algorithm also gives handles that enable the designer to directly influence the size of the singular values/condition numbers of the loop-shaping weights at each frequency. Furthermore, provided that the loop-shape boundaries are correctly specified, the proposed algorithm always gives a loop-shape that is feasible to the specifications.

6.2 Directions for Future Research

Some possible directions for future research are hereby outlined.

In the area of μ-synthesis

- The sets of allowable perturbations were restricted to complex blocks for notational convenience. It was claimed that most proofs presented in the thesis generalise to the case where the allowable perturbations have mixed real and complex blocks. This generalisation needs to be clearly written down and problems particular to this framework need to be individually investigated.

- The "fictitious" performance uncertainty set $\mathbf{\Delta}_P$ was defined to contain unstructured full complex blocks, as is standard in the literature. More complicated structures could be defined which would allow the proposed algorithm to have other meaningful uses (for example, the imposition of specific channel-to-channel performance gain constraints and the optimisation of specific input uncertainty weights).

- A natural extension of this μ-synthesis work is the generalisation of the proposed optimisation problem to LPV systems. An algorithm similar to the one proposed in this thesis would be particularly beneficial in that scenario because maximisation of performance weights would occur at each frequency and at each point in the parameter space. This would help the designer to identify problematic regions in the parameter space and hence compromise the specifications (if necessary) only in those regions.

In the area of $\mathscr{H}_\infty$ loop-shaping

- Tight lower and upper bounds of $b_{opt}(P_s)$ (in terms of the loop-shaping weights and the plant only) could perhaps be used to obtain a non-iterative solution to the optimisation problem posed in Chapter 5.

- It is desirable to incorporate complexity constraints (i.e. smoothness constraints) on the loop-shaping weights and the controller. This would indirectly lead to the synthesis of lower order weights and controllers which is greatly desirable in applications involving highly resonant plants (e.g. flexible structures). Smoothness constraints on the loop-shaping weights would also allow the direct construction of non-diagonal weights from the frequency-by-frequency optimisation data. This is important as the interpolation of singular vectors used in this thesis suffers from the possible lack of continuity at points of repeated singular values.

- An conceptually different approach might also be taken. Instead of maximising the robust stability margin $b_{opt}(P_s)$, one might optimise the loop-shape and constrain $b_{opt}(P_s)$ to be greater than 0.3, say.

- Again, an important extension of this $\mathscr{H}_\infty$ loop-shaping work is the generalisation of the proposed optimisation problem to LPV systems. An algorithm similar to the one proposed in this thesis would be particularly beneficial in that scenario as loop-shaping weights would be synthesised pointwise in both frequency and parameter space. This would help the designer to identify problematic regions in the parameter space and hence compromise the specifications (if necessary) only in those regions.

In other similar areas

- It is believed that the benefits introduced by this thesis in the area of μ-synthesis robust performance based designs can be transported to the newly formulated Robust $\mathscr{H}_2$ control framework without much difficulty.

Appendix A

Appendix to Chapter 3

A.1 Proof of Lemma 3.7.1

Statements (i) and (ii) in the Lemma will be connected by a sequence of equivalent reformulations.

(a) There exists a $\breve{Q} \in \mathbb{R}^{p\times(N+1)q}$ and $\forall k \in \boldsymbol{\Omega}$ a $\Theta^{\star}_{k,0} \in \boldsymbol{\mathcal{D}}$ and a $v_{w,k} \in \mathbb{R}^{n}$ satisfying

$$\begin{bmatrix} \begin{pmatrix} (\Theta^{\star}_{k,0})^{-1} & 0 \\ 0 & \mathrm{diag}\big(v_{w,k}\big) \end{pmatrix} & \mathcal{F}_l\left(T(j\omega^a_k), \breve{Q}B(j\omega^a_k)\right) \\ * & \begin{pmatrix} (\Theta^{\star}_{k,0}) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0.$$

(b) There exists a $\breve{Q} \in \mathbb{R}^{p\times(N+1)q}$ and $\forall k \in \boldsymbol{\Omega}$ a $\Theta^{\star}_{k,0} \in \boldsymbol{\mathcal{D}}$ and a $v_{w,k} \in \mathbb{R}^{n}_{+}$ satisfying

$$\begin{bmatrix} (\Theta^{\star}_{k,0})^{-1} & H_{11}(j\omega^a_k) & H_{12}(j\omega^a_k) \\ * & (\Theta^{\star}_{k,0}) & 0 \\ * & * & I_m \end{bmatrix} > \begin{bmatrix} 0 \\ H_{21}(j\omega^a_k)^* \\ H_{22}(j\omega^a_k)^* \end{bmatrix} \mathrm{diag}\big(v_{w,k}\big)^{-1} \begin{bmatrix} 0 & H_{21}(j\omega^a_k) & H_{22}(j\omega^a_k) \end{bmatrix}, \tag{A.1}$$

where

$$\begin{aligned} \begin{bmatrix} H_{11}(j\omega^a_k) & H_{12}(j\omega^a_k) \\ H_{21}(j\omega^a_k) & H_{22}(j\omega^a_k) \end{bmatrix} &:= \begin{bmatrix} T_{11}(j\omega^a_k) & T_{12}(j\omega^a_k) \\ T_{21}(j\omega^a_k) & T_{22}(j\omega^a_k) \end{bmatrix} + \begin{bmatrix} T_{13}(j\omega^a_k) \\ T_{23}(j\omega^a_k) \end{bmatrix} \breve{Q}B(j\omega^a_k) \begin{bmatrix} T_{31}(j\omega^a_k) & T_{32}(j\omega^a_k) \end{bmatrix} \\ &= \mathcal{F}_l\left(T(j\omega^a_k), \breve{Q}B(j\omega^a_k)\right). \end{aligned}$$

The equivalence (a) $\Leftrightarrow$ (b) follows from Schur Complement Lemma 2.8.1 applied around the term "$\mathrm{diag}\big(v_{w,k}\big)$" appearing in (a).

(c) There exists a $\breve{Q} \in \mathbb{R}^{p\times(N+1)q}$ and $\forall k \in \boldsymbol{\Omega}$ a $\Theta^{\star}_{k,0} \in \boldsymbol{\mathcal{D}}$ satisfying

$$\begin{bmatrix} (\Theta^{\star}_{k,0})^{-1} & H_{11}(j\omega^a_k) & H_{12}(j\omega^a_k) \\ * & (\Theta^{\star}_{k,0}) & 0 \\ * & * & I_m \end{bmatrix} > 0.$$

Note that (b) $\Rightarrow$ (c) trivially follows from inequality (A.1), whereas (b) $\Leftarrow$ (c) is because given that (c) is true there always exists a sufficiently large $v_{w,k} \in \mathbb{R}_+^n$ at every $k \in \mathbf{\Omega}$ that makes the RHS of inequality (A.1) sufficiently small to also imply that (b) is true.

(d) There exists a $\breve{Q} \in \mathbb{R}^{p\times(N+1)q}$, a $\rho \in \mathbb{R}_+$ and $\forall k \in \mathbf{\Omega}$ a $\Theta_{k,0}^\star \in \boldsymbol{\mathcal{D}}$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^\star)^{-1} & H_{11}(j\omega_k^a) & H_{12}(j\omega_k^a) \\ * & (\Theta_{k,0}^\star) & 0 \\ * & * & \rho I_m \end{bmatrix} > 0. \tag{A.2}$$

The equivalence (c) $\Leftrightarrow$ (d) can be seen through the application of the congruence transformation $\text{diag}\big(\sqrt{\rho}I_r, \frac{1}{\sqrt{\rho}}I_r, \frac{1}{\sqrt{\rho}}I_m\big)$ on inequality (A.2).

(e) There exists a $\breve{Q} \in \mathbb{R}^{p\times(N+1)q}$, a $\rho \in \mathbb{R}_+$ and $\forall k \in \mathbf{\Omega}$ a $\Theta_{k,0}^\star \in \boldsymbol{\mathcal{D}}$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^\star)^{-1} & H_{11}(j\omega_k^a) \\ * & (\Theta_{k,0}^\star) \end{bmatrix} > \frac{1}{\rho}\begin{bmatrix} H_{12}(j\omega_k^a) \\ 0 \end{bmatrix}\begin{bmatrix} H_{12}(j\omega_k^a)^* & 0 \end{bmatrix}. \tag{A.3}$$

The equivalence (d) $\Leftrightarrow$ (e) follows from Schur Complement Lemma 2.8.1 applied around the (3,3)-element of inequality (A.2).

(f) There exists a $\breve{Q} \in \mathbb{R}^{p\times(N+1)q}$ and $\forall k \in \mathbf{\Omega}$ a $\Theta_{k,0}^\star \in \boldsymbol{\mathcal{D}}$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^\star)^{-1} & H_{11}(j\omega_k^a) \\ * & (\Theta_{k,0}^\star) \end{bmatrix} > 0.$$

Note that (e) $\Rightarrow$ (f) trivially follows from inequality (A.3), whereas (e) $\Leftarrow$ (f) is because given that (f) is true there always exists a sufficiently large $\rho \in \mathbb{R}_+$ that makes the RHS of inequality (A.3) sufficiently small to also imply that (e) is true. This last condition is in fact Statement (ii) in the Lemma (since $H_{11}(j\omega_k^a) = T_{11}(j\omega_k^a) + T_{13}(j\omega_k^a)\breve{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)$) and hence the proof is complete.

Appendix B

Appendix to Chapter 4

B.1 Proof of Theorem 4.5.1

First note that Lemma 2.6.3 together with Section 4.3 ensure that given any $\bar{W}(s) \in \mathcal{W}^{TF}_{(A_{\bar{W}}, B_{\bar{W}})}$, there exists a $\breve{W} \in \mathbf{\Xi}_{\breve{W}}$ such that $T^o_{\bar{W}}(j\omega)^* \breve{W} T^o_{\bar{W}}(j\omega) = \bar{W}(j\omega)^* \bar{W}(j\omega)\ \forall \omega \in \mathbb{R} \cup \{\infty\}$. Then

$$\begin{aligned}\Upsilon(j\omega)^* \bar{W}(j\omega)^* \bar{W}(j\omega)\Upsilon(j\omega) &= \Upsilon(j\omega)^* T^o_{\bar{W}}(j\omega)^* \breve{W} T^o_{\bar{W}}(j\omega)\Upsilon(j\omega) \\ &= \varphi(j\omega)^* \left(\begin{array}{ccc} \begin{bmatrix} I_{s_w} & 0 \\ 0 & C_\Upsilon^T \end{bmatrix} \breve{W} \begin{bmatrix} I_{s_w} & 0 \\ 0 & C_\Upsilon \end{bmatrix} & & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \end{bmatrix} & & 0 \end{array} \right) \varphi(j\omega),\end{aligned}$$

where

$$\varphi(s) := \left[\begin{array}{cc} \left(sI - \begin{bmatrix} A_{\bar{W}} & B_{\bar{W}} C_\Upsilon \\ 0 & A_\Upsilon \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ B_\Upsilon \end{bmatrix} \\ I_n \end{array} \right].$$

Using Corollary 2.6.2 on the $j\omega$-axis (noting that $\begin{bmatrix} A_{\bar{W}} & B_{\bar{W}} C_\Upsilon \\ 0 & A_\Upsilon \end{bmatrix}$ is Hurwitz), it follows that

$$\Upsilon(j\omega)^* \bar{W}(j\omega)^* \bar{W}(j\omega)\Upsilon(j\omega) = E(j\omega) + E(j\omega)^*$$

where

$$E(s) := \left[\begin{array}{cc|c} A_{\bar{W}} & B_{\bar{W}} C_\Upsilon & 0 \\ 0 & A_\Upsilon & B_\Upsilon \\ \hline \begin{pmatrix} 0 & B_\Upsilon^T \end{pmatrix} X & & 0 \end{array} \right]$$

and the real matrix $X = X^T$ is the unique solution to Lyapunov equation

$$X \begin{bmatrix} A_{\bar{W}} & B_{\bar{W}} C_\Upsilon \\ 0 & A_\Upsilon \end{bmatrix} + \begin{bmatrix} A_{\bar{W}} & B_{\bar{W}} C_\Upsilon \\ 0 & A_\Upsilon \end{bmatrix}^T X = - \begin{bmatrix} I_{s_w} & 0 \\ 0 & C_\Upsilon^T \end{bmatrix} \breve{W} \begin{bmatrix} I_{s_w} & 0 \\ 0 & C_\Upsilon \end{bmatrix}. \tag{B.1}$$

Consequently,

$$\begin{aligned}
\left\|\bar{W}\Upsilon\right\|_2^2 &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \operatorname{trace}\left\{\Upsilon(j\omega)^*\bar{W}(j\omega)^*\bar{W}(j\omega)\Upsilon(j\omega)\right\} d\omega \\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty} \operatorname{trace}\left\{E(j\omega)+E(j\omega)^*\right\} d\omega \\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty} 2\,\Re\left[\operatorname{trace}\left\{E(j\omega)\right\}\right] d\omega \\
&= \frac{1}{\pi}\int_{-\infty}^{\infty} \Re\left[\operatorname{trace}\left\{\begin{bmatrix}0 & B_\Upsilon^T\end{bmatrix} X \left(j\omega I - \begin{bmatrix}A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon\end{bmatrix}\right)^{-1}\begin{bmatrix}0 \\ B_\Upsilon\end{bmatrix}\right\}\right] d\omega \\
&= \Re\left[\operatorname{trace}\left\{\begin{bmatrix}0 & B_\Upsilon^T\end{bmatrix} X \,.\, \frac{1}{\pi}\int_{-\infty}^{\infty}\left(j\omega I - \begin{bmatrix}A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon\end{bmatrix}\right)^{-1} d\omega \,.\, \begin{bmatrix}0 \\ B_\Upsilon\end{bmatrix}\right\}\right] \\
&= \operatorname{trace}\left\{\begin{bmatrix}0 & B_\Upsilon^T\end{bmatrix} X \begin{bmatrix}0 \\ B_\Upsilon\end{bmatrix}\right\} \quad \text{by Lemma 2.7.1} \\
&= \operatorname{vec}(I_n)^T \operatorname{vec}\left(\begin{bmatrix}0 & B_\Upsilon^T\end{bmatrix} X \begin{bmatrix}0 \\ B_\Upsilon\end{bmatrix}\right) \\
&= \operatorname{vec}(I_n)^T \left(\begin{bmatrix}0 \\ B_\Upsilon\end{bmatrix} \otimes \begin{bmatrix}0 \\ B_\Upsilon\end{bmatrix}\right)^T \operatorname{vec}(X). \qquad \text{(B.2)}
\end{aligned}$$

Then note that Lyapunov equation (B.1) can be rewritten as

$$\left(\begin{bmatrix}A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon\end{bmatrix}^T \oplus \begin{bmatrix}A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon\end{bmatrix}^T\right) \operatorname{vec}(X) = -\left(\begin{bmatrix}I_{s_w} & 0 \\ 0 & C_\Upsilon\end{bmatrix}^T \otimes \begin{bmatrix}I_{s_w} & 0 \\ 0 & C_\Upsilon\end{bmatrix}^T\right) \operatorname{vec}\left(\breve{W}\right).$$

Since the matrix $\begin{bmatrix}A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon\end{bmatrix}$ is Hurwitz, the required inverse exists and

$$\operatorname{vec}(X) = -\left(\begin{bmatrix}A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon\end{bmatrix} \oplus \begin{bmatrix}A_{\bar{W}} & B_{\bar{W}}C_\Upsilon \\ 0 & A_\Upsilon\end{bmatrix}\right)^{-T} \left(\begin{bmatrix}I_{s_w} & 0 \\ 0 & C_\Upsilon\end{bmatrix} \otimes \begin{bmatrix}I_{s_w} & 0 \\ 0 & C_\Upsilon\end{bmatrix}\right)^T \operatorname{vec}\left(\breve{W}\right). \quad \text{(B.3)}$$

The required result then follows directly from equations (B.2) and (B.3).

B.2 Proof of Theorem 4.6.1

Statements (i) and (ii) in the Theorem will be connected by a sequence of equivalent reformulations.

(a) $$\inf_{\bar{D}\in\boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}},B_{\bar{D}})}} \left\|\begin{pmatrix}\bar{D} & 0 \\ 0 & I_m\end{pmatrix} \mathcal{F}_l(G,K)^T \begin{pmatrix}\bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1}\end{pmatrix}\right\|_\infty < 1.$$

(b) $\exists\, \bar{D} \in \boldsymbol{\mathcal{D}}^{TF}_{(A_{\bar{D}}, B_{\bar{D}})}$: $\left[\mathcal{F}_l\left(G(j\omega), K(j\omega)\right)^T\right]^* \begin{pmatrix} \bar{D}(j\omega)^*\bar{D}(j\omega) & 0 \\ 0 & I_m \end{pmatrix} \left[\mathcal{F}_l\left(G(j\omega), K(j\omega)\right)^T\right]$

$$< \begin{pmatrix} \bar{D}(j\omega)^*\bar{D}(j\omega) & 0 \\ 0 & \bar{W}(j\omega)^*\bar{W}(j\omega) \end{pmatrix} \quad \forall \omega \in \mathbb{R} \cup \{\infty\}.$$

(c) $\exists\, \breve{D} \in \boldsymbol{\Xi}_{\breve{D}}$ such that

$$T^o_{\bar{D}}(j\omega)^*\breve{D}T^o_{\bar{D}}(j\omega) > 0 \qquad \forall \omega \in \mathbb{R} \cup \{\infty\},$$

$$\left[\mathcal{F}_l\left(G(j\omega), K(j\omega)\right)^T\right]^* \begin{pmatrix} T^o_{\bar{D}}(j\omega)^*\breve{D}T^o_{\bar{D}}(j\omega) & 0 \\ 0 & I_m \end{pmatrix} \left[\mathcal{F}_l\left(G(j\omega), K(j\omega)\right)^T\right]$$

$$< \begin{pmatrix} T^o_{\bar{D}}(j\omega)^*\breve{D}T^o_{\bar{D}}(j\omega) & 0 \\ 0 & T^o_{\bar{W}}(j\omega)^*\breve{W}T^o_{\bar{W}}(j\omega) \end{pmatrix} \quad \forall \omega \in \mathbb{R} \cup \{\infty\},$$

where $T^o_{\bar{D}}(s)$ is defined as in equation (4.4). Note that (b) $\Rightarrow$ (c) follows from Lemma 2.6.3 (i) and Section 4.3, whereas (b) $\Leftarrow$ (c) follows from Lemma 2.6.3 (ii) and Section 4.3 on noting that the above two inequalities together implicitly guarantee that $T^o_{\bar{W}}(j\omega)^*\breve{W}T^o_{\bar{W}}(j\omega) > 0$ for all $\omega \in \mathbb{R} \cup \{\infty\}$.

(d) $\exists\, \breve{D} \in \boldsymbol{\Xi}_{\breve{D}}$ such that

$$T^o_{\bar{D}}(j\omega)^*\breve{D}T^o_{\bar{D}}(j\omega) > 0 \qquad \forall \omega \in \mathbb{R} \cup \{\infty\},$$
$$\varphi(j\omega)^*\grave{Q}\,\varphi(j\omega) < 0 \qquad \forall \omega \in \mathbb{R} \cup \{\infty\},$$

where

$$\varphi(s) := \begin{pmatrix} \begin{bmatrix} T^o_{\bar{D}}(s) & 0 \\ 0 & I_m \end{bmatrix} \mathcal{F}_l\left(G(s), K(s)\right)^T \\ \\ \begin{bmatrix} T^o_{\bar{D}}(s) & 0 \\ 0 & T^o_{\bar{W}}(s) \end{bmatrix} \end{pmatrix},$$

$$\grave{Q} := \operatorname{diag}\left(\breve{D},\ I_m,\ -\breve{D},\ -\breve{W}\right).$$

Observe that (c) $\Leftrightarrow$ (d) follows from simple algebraic manipulations.

(e) $\exists\, \breve{D} \in \boldsymbol{\Xi}_{\breve{D}}$ such that

$$\begin{bmatrix} \left(j\omega I - A_{\bar{D}}\right)^{-1}B_{\bar{D}} \\ I_r \end{bmatrix}^* \breve{D} \begin{bmatrix} \left(j\omega I - A_{\bar{D}}\right)^{-1}B_{\bar{D}} \\ I_r \end{bmatrix} > 0 \qquad \forall \omega \in \mathbb{R} \cup \{\infty\},$$

$$\begin{bmatrix} \left(j\omega I - \grave{A}\right)^{-1}\grave{B} \\ I \end{bmatrix}^* \begin{bmatrix} \grave{C}^T \\ \grave{D}^T \end{bmatrix} \grave{Q} \begin{bmatrix} \grave{C} & \grave{D} \end{bmatrix} \begin{bmatrix} \left(j\omega I - \grave{A}\right)^{-1}\grave{B} \\ I \end{bmatrix} < 0 \qquad \forall \omega \in \mathbb{R} \cup \{\infty\},$$

where $\grave{A}$, $\grave{B}$, $\grave{C}$, $\grave{D}$ and $\grave{Q}$ are defined as in Part (ii) of the Theorem. Then (d) $\Leftrightarrow$ (e) easily follows by noting that $\varphi(s) = \left[\begin{array}{c|c} \grave{A} & \grave{B} \\ \hline \grave{C} & \grave{D} \end{array}\right]$.

(f) $\exists\, \breve{D} \in \mathbf{\Xi}_{\breve{D}}$, $X = X^T \in \mathbb{R}^{s_D \times s_D}$ and $Y = Y^T \in \mathbb{R}^{(s_{cl}+2s_D+s_w)\times(s_{cl}+2s_D+s_w)}$ such that

$$\begin{bmatrix} XA_{\bar{D}} + A_{\bar{D}}^T X & XB_{\bar{D}} \\ B_{\bar{D}}^T X & 0 \end{bmatrix} + \breve{D} > 0,$$

$$\begin{bmatrix} Y\grave{A} + \grave{A}^T Y & Y\grave{B} \\ \grave{B}^T Y & 0 \end{bmatrix} + \begin{bmatrix} \grave{C}^T \\ \grave{D}^T \end{bmatrix} \grave{Q} \begin{bmatrix} \grave{C} & \grave{D} \end{bmatrix} < 0.$$

The equivalence (e) $\Leftrightarrow$ (f) follows from KYP Lemma 2.8.2.

B.3 Proof of Theorem 4.7.1

Before proving the equivalence between the Theorem's two statements, some notation needs to be defined. To this end, let the controller $K(s) \in \boldsymbol{\mathcal{K}}_G^{TF}$ have a state-space realisation $\left[\begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array}\right]$, where $A_K \in \mathbb{R}^{s_K \times s_K}$ and $D_K \in \mathbb{R}^{p\times q}$. The order s_K of this controller is not yet known (i.e. s_K is a variable), as the set $\boldsymbol{\mathcal{K}}_G^{TF}$ contains all internally stabilising controllers for G of *any* order. Define

$$\begin{bmatrix} \hat{A} & \hat{B}_1 & \hat{B}_2 & \hat{B}_3 \\ \hat{C}_1 & \hat{D}_{11} & \hat{D}_{12} & \hat{D}_{13} \\ \hat{C}_2 & \hat{D}_{21} & \hat{D}_{22} & \hat{D}_{23} \\ \hat{C}_3 & \hat{D}_{31} & \hat{D}_{32} & \Phi_K^T \end{bmatrix} := \left[\begin{array}{cc:c:c:cc} \tilde{A} & 0 & \tilde{B}_1 & \tilde{B}_2 & 0 & \tilde{B}_3 \\ 0 & 0 & 0 & 0 & I_{s_K} & 0 \\ \hdashline \tilde{C}_1 & 0 & \tilde{D}_{11} & \tilde{D}_{12} & 0 & \tilde{D}_{13} \\ \hdashline \tilde{C}_2 & 0 & \tilde{D}_{21} & \tilde{D}_{22} & 0 & \tilde{D}_{23} \\ \hdashline 0 & I_{s_K} & 0 & 0 & A_K^T & C_K^T \\ \tilde{C}_3 & 0 & \tilde{D}_{31} & \tilde{D}_{32} & B_K^T & D_K^T \end{array}\right],$$

$$\text{and} \quad \begin{bmatrix} \tilde{A}_{cl} & \tilde{B}_{1cl} & \tilde{B}_{2\,cl} \\ \tilde{C}_{1cl} & \tilde{D}_{11cl} & \tilde{D}_{12\,cl} \\ \tilde{C}_{2\,cl} & \tilde{D}_{21cl} & \tilde{D}_{22\,cl} \end{bmatrix} := \begin{bmatrix} \hat{A} & \hat{B}_1 & \hat{B}_2 \\ \hat{C}_1 & \hat{D}_{11} & \hat{D}_{12} \\ \hat{C}_2 & \hat{D}_{21} & \hat{D}_{22} \end{bmatrix} + \begin{bmatrix} \hat{B}_3 \\ \hat{D}_{13} \\ \hat{D}_{23} \end{bmatrix} \Phi_K \begin{bmatrix} \hat{C}_3 & \hat{D}_{31} & \hat{D}_{32} \end{bmatrix}.$$

Then $\mathcal{F}_l\big(\tilde{G}, K\big) = \left[\begin{array}{c|cc} \tilde{A}_{cl} & \tilde{B}_{1cl} & \tilde{B}_{2\,cl} \\ \hline \tilde{C}_{1cl} & \tilde{D}_{11cl} & \tilde{D}_{12\,cl} \\ \tilde{C}_{2\,cl} & \tilde{D}_{21cl} & \tilde{D}_{22\,cl} \end{array}\right]$ and $K(s) \in \boldsymbol{\mathcal{K}}_G^{TF}$ if and only if $K(s) \in \boldsymbol{\mathcal{K}}_{\tilde{G}}^{TF}$.

Statements (i) and (ii) in the Theorem will now be connected by a sequence of equivalent reformulations.

(a) $$\min_{K\in\boldsymbol{\mathcal{K}}_G^{TF}} \left\| \begin{pmatrix} \bar{D} & 0 \\ 0 & I_m \end{pmatrix} \mathcal{F}_l\,(G,K)^T \begin{pmatrix} \bar{D}^{-1} & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1.$$

(b) $$\min_{K\in\boldsymbol{\mathcal{K}}_{\tilde{G}}^{TF}} \left\| \mathcal{F}_l\big(\tilde{G},K\big)^T \begin{pmatrix} I_r & 0 \\ 0 & \bar{W}^{-1} \end{pmatrix} \right\|_\infty < 1.$$

(c) $\exists\, K \in \mathcal{K}_{\tilde{G}}^{TF}$ such that

$$\left[\mathcal{F}_l\big(\tilde{G}(j\omega), K(j\omega)\big)^T\right]^* \left[\mathcal{F}_l\big(\tilde{G}(j\omega), K(j\omega)\big)^T\right] < \begin{pmatrix} I_r & 0 \\ 0 & \bar{W}(j\omega)^*\bar{W}(j\omega) \end{pmatrix}$$

for all $\omega \in \mathbb{R} \cup \{\infty\}$.

(d) $\exists\, K \in \mathcal{K}_{\tilde{G}}^{TF}$ such that

$$\left[\mathcal{F}_l\big(\tilde{G}(j\omega), K(j\omega)\big)^T\right]^* \left[\mathcal{F}_l\big(\tilde{G}(j\omega), K(j\omega)\big)^T\right] < \begin{pmatrix} I_r & 0 \\ 0 & T_{\bar{W}}^o(j\omega)^*\breve{W}T_{\bar{W}}^o(j\omega) \end{pmatrix}$$

for all $\omega \in \mathbb{R} \cup \{\infty\}$. Note that (c) $\Rightarrow$ (d) follows from Lemma 2.6.3 (i) and Section 4.3, whereas (c) $\Leftarrow$ (d) follows from Lemma 2.6.3 (ii) and Section 4.3 on noting that the above frequency domain inequality implicitly guarantees that $T_{\bar{W}}^o(j\omega)^*\breve{W}T_{\bar{W}}^o(j\omega) > 0$ for all $\omega \in \mathbb{R} \cup \{\infty\}$.

(e) $\exists\, K \in \mathcal{K}_{\tilde{G}}^{TF}$ such that

$$\begin{pmatrix} \mathcal{F}_l\big(\tilde{G}(j\omega), K(j\omega)\big)^T \\ \begin{bmatrix} I_r & 0 \\ 0 & T_{\bar{W}}^o(j\omega) \end{bmatrix} \end{pmatrix}^* \begin{pmatrix} I_{r+m} & 0 \\ 0 & \begin{bmatrix} -I_r & 0 \\ 0 & -\breve{W} \end{bmatrix} \end{pmatrix} \begin{pmatrix} \mathcal{F}_l\big(\tilde{G}(j\omega), K(j\omega)\big)^T \\ \begin{bmatrix} I_r & 0 \\ 0 & T_{\bar{W}}^o(j\omega) \end{bmatrix} \end{pmatrix} < 0$$

for all $\omega \in \mathbb{R} \cup \{\infty\}$. Observe that (d) $\Leftrightarrow$ (e) follows from simple algebraic manipulations.

(f) $\exists\, s_K \in \mathbb{Z}_+$ and $\Phi_K \in \mathbb{R}^{(s_K+p)\times(s_K+q)}$ such that

$$\tilde{A}_{cl} \text{ is Hurwitz,}$$

$$\begin{bmatrix} (j\omega I - \acute{A})^{-1}\acute{B} \\ I \end{bmatrix}^* \begin{bmatrix} \acute{C}^T \\ \acute{D}^T \end{bmatrix} \acute{Q} \begin{bmatrix} \acute{C} & \acute{D} \end{bmatrix} \begin{bmatrix} (j\omega I - \acute{A})^{-1}\acute{B} \\ I \end{bmatrix} < 0 \qquad \forall \omega \in \mathbb{R} \cup \{\infty\},$$

where $\acute{A}$, $\acute{B}$, $\acute{C}$, $\acute{D}$ and $\acute{Q}$ are defined by

$$\begin{bmatrix} \acute{A} & \acute{B} \\ \acute{C} & \acute{D} \end{bmatrix} := \left[\begin{array}{cc:cc} A_{\bar{W}} & 0 & 0 & B_{\bar{W}} \\ 0 & \tilde{A}_{cl}^T & \tilde{C}_{1cl}^T & \tilde{C}_{2cl}^T \\ \hdashline 0 & \tilde{B}_{1cl}^T & \tilde{D}_{11cl}^T & \tilde{D}_{21cl}^T \\ 0 & \tilde{B}_{2cl}^T & \tilde{D}_{12cl}^T & \tilde{D}_{22cl}^T \\ 0 & 0 & I_r & 0 \\ I_{s_w} & 0 & 0 & 0 \\ 0 & 0 & 0 & I_n \end{array}\right],$$

$$\acute{Q} := \operatorname{diag}\left(I_r,\ I_m,\ -I_r,\ -\breve{W}\right) = \operatorname{diag}\left(I_r,\ I_m,\ -I_r,\ -\begin{bmatrix} 0 & \breve{W}_{12} \\ \breve{W}_{12}^T & \breve{W}_{22} \end{bmatrix}\right).$$

The equivalence (e) $\Leftrightarrow$ (f) follows by rewriting the frequency domain inequality in (e) using state-space realisations and further noting that $K(s)$ internally stabilises $\tilde{G}(s)$ if $\tilde{A}_{cl}$ is Hurwitz, and $\tilde{A}_{cl}$ is Hurwitz if there is a stabilisable and detectable state-space realisation for $K(s)$ which internally stabilises $\tilde{G}(s)$ (Green and Limebeer, 1995, Lemma A.4.1).

(g) $\exists\, s_K \in \mathbb{Z}_+$, $\Phi_K \in \mathbb{R}^{(s_K+p)\times(s_K+q)}$ and $X = X^T \in \mathbb{R}^{(s_W+s_{\tilde{G}}+s_K)\times(s_W+s_{\tilde{G}}+s_K)}$ such that

$$X > 0,$$
$$\begin{bmatrix} X\acute{A} + \acute{A}^T X & X\acute{B} \\ \acute{B}^T X & 0 \end{bmatrix} + \begin{bmatrix} \acute{C}^T \\ \acute{D}^T \end{bmatrix} \acute{Q} \begin{bmatrix} \acute{C} & \acute{D} \end{bmatrix} < 0, \tag{B.4}$$

where $\acute{A}$, $\acute{B}$, $\acute{C}$, $\acute{D}$ and $\acute{Q}$ are defined as in (f) above. Since inequality (B.4) implicitly guarantees that $X\acute{A} + \acute{A}^T X < -\acute{C}^T\acute{Q}\,\acute{C} \leq 0$, a standard Lyapunov type argument gives $X > 0$ if and only if $\acute{A}$ is Hurwitz. Now $\acute{A}$ is Hurwitz if and only if $\tilde{A}_{cl}$ is Hurwitz, as $A_{\bar{W}}$ is already restricted to be Hurwitz in the set $\mathbf{\Xi}_{(A_{\bar{W}}, B_{\bar{W}})}$. Then (f) $\Leftrightarrow$ (g) follows from a straightforward application of KYP Lemma 2.8.2.

(h) $\exists\, s_K \in \mathbb{Z}_+$, $\Phi_K \in \mathbb{R}^{(s_K+p)\times(s_K+q)}$ and $X = X^T \in \mathbb{R}^{(s_W+s_{\tilde{G}}+s_K)\times(s_W+s_{\tilde{G}}+s_K)}$ such that

$$X > 0,$$
$$\begin{pmatrix} X\begin{bmatrix} A_{\bar{W}} & 0 \\ 0 & \tilde{A}_{cl}^T \end{bmatrix} + \begin{bmatrix} A_{\bar{W}}^T & 0 \\ 0 & \tilde{A}_{cl} \end{bmatrix} X & X\begin{bmatrix} 0 & B_{\bar{W}} \\ \tilde{C}_{1cl}^T & \tilde{C}_{2\,cl}^T \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ \tilde{B}_{1cl} & \tilde{B}_{2\,cl} \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{C}_{1cl} \\ B_{\bar{W}}^T & \tilde{C}_{2\,cl} \end{bmatrix} X & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11cl} & \tilde{D}_{12\,cl} \\ \tilde{D}_{21cl} & \tilde{D}_{22\,cl} \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{B}_{1cl}^T \\ 0 & \tilde{B}_{2\,cl}^T \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11cl}^T & \tilde{D}_{21cl}^T \\ \tilde{D}_{12\,cl}^T & \tilde{D}_{22\,cl}^T \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} \end{pmatrix}$$
$$< \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right).$$

Note that (g) $\Leftrightarrow$ (h) follows by applying Schur Complement Lemma 2.8.1 around the (3,3)-block of the above inequality and then re-arranging to give inequality (B.4).

(i) $\exists\, s_K \in \mathbb{Z}_+$, $\Phi_K \in \mathbb{R}^{(s_K+p)\times(s_K+q)}$ and $X = X^T \in \mathbb{R}^{(s_W+s_{\tilde{G}}+s_K)\times(s_W+s_{\tilde{G}}+s_K)}$ such that

$$X > 0,$$
$$F + U^T \Phi_K V + V^T \Phi_K^T U < 0,$$

where the real matrices F, U and V are defined by

$$F := \begin{pmatrix} X\begin{bmatrix} A_{\bar{W}} & 0 \\ 0 & \hat{A}^T \end{bmatrix} + \begin{bmatrix} A_{\bar{W}}^T & 0 \\ 0 & \hat{A} \end{bmatrix} X & X\begin{bmatrix} 0 & B_{\bar{W}} \\ \hat{C}_1^T & \hat{C}_2^T \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ \hat{B}_1 & \hat{B}_2 \end{bmatrix} \\ \begin{bmatrix} 0 & \hat{C}_1 \\ B_{\bar{W}}^T & \hat{C}_2 \end{bmatrix} X & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} \hat{D}_{11} & \hat{D}_{12} \\ \hat{D}_{21} & \hat{D}_{22} \end{bmatrix} \\ \begin{bmatrix} 0 & \hat{B}_1^T \\ 0 & \hat{B}_2^T \end{bmatrix} & \begin{bmatrix} \hat{D}_{11}^T & \hat{D}_{21}^T \\ \hat{D}_{12}^T & \hat{D}_{22}^T \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} \end{pmatrix}$$

$$- \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right),$$

$$U := \left(\begin{bmatrix} 0 & \hat{B}_3^T \end{bmatrix} \begin{bmatrix} \hat{D}_{13}^T & \hat{D}_{23}^T \end{bmatrix} \begin{bmatrix} 0 & 0 \end{bmatrix} \right),$$

$$V := \left(\begin{bmatrix} 0 & \hat{C}_3 \end{bmatrix} X \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{D}_{31} & \hat{D}_{32} \end{bmatrix} \right).$$

This equivalence follows by simply extracting Φ_K from the closed-loop matrices. This is done in preparation for the Projection Lemma which will be used next.

(j) $\exists\, s_K \in \mathbb{Z}_+$ and $X = X^T \in \mathbb{R}^{(s_W+s_{\bar{G}}+s_K)\times(s_W+s_{\bar{G}}+s_K)}$ such that

$$X > 0,$$

$$\Psi_U^T F \Psi_U < 0 \quad \text{and} \quad \Psi_V^T F \Psi_V < 0,$$

where F is defined as in (i) and Ψ_U and Ψ_V are matrices with columns that form bases for the null spaces of U and V respectively. Since

$$U = \left(\begin{bmatrix} 0 & 0 & I_{s_K} \\ 0 & \tilde{B}_3^T & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ \tilde{D}_{13}^T & \tilde{D}_{23}^T \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right)$$

$$\text{and } V = \left(\begin{bmatrix} 0 & 0 & I_{s_K} \\ 0 & \tilde{C}_3 & 0 \end{bmatrix} X \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ \tilde{D}_{31} & \tilde{D}_{32} \end{bmatrix} \right),$$

one possible choice of Ψ_U and Ψ_V is

$$\Psi_U := \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_1 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & \psi_2 \\ 0 & \psi_3 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} I_r & 0 \\ 0 & I_m \end{bmatrix} \end{pmatrix}, \qquad \hat{\Psi}_V := \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_4 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} I_r & 0 \\ 0 & I_n \end{bmatrix} \\ \begin{bmatrix} 0 & \psi_5 \\ 0 & \psi_6 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix},$$

$$\text{and } \Psi_V := \begin{pmatrix} X^{-1} & 0 & 0 \\ 0 & I_{r+n} & 0 \\ 0 & 0 & I_{r+m} \end{pmatrix} \hat{\Psi}_V.$$

Above, the columns of $\begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \end{bmatrix}$ (resp. $\begin{bmatrix} \psi_4 \\ \psi_5 \\ \psi_6 \end{bmatrix}$) form bases for the null space of $\begin{bmatrix} \tilde{B}_3^T & \tilde{D}_{13}^T & \tilde{D}_{23}^T \end{bmatrix}$ (resp. $\begin{bmatrix} \tilde{C}_3 & \tilde{D}_{31} & \tilde{D}_{32} \end{bmatrix}$). The equivalence (i) $\Leftrightarrow$ (j) follows by Projection Lemma 2.8.3.

(k) $\exists\, s_K \in \mathbb{Z}_+$ and $X = X^T \in \mathbb{R}^{(s_W+s_{\tilde{G}}+s_K)\times(s_W+s_{\tilde{G}}+s_K)}$ such that

$$X > 0,$$

$$\Psi_U^T \cdot \begin{pmatrix} X\begin{bmatrix} A_{\bar{W}} & 0 & 0 \\ 0 & \tilde{A}^T & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} A_{\bar{W}}^T & 0 & 0 \\ 0 & \tilde{A} & 0 \\ 0 & 0 & 0 \end{bmatrix} X & X\begin{bmatrix} 0 & B_{\bar{W}} \\ \tilde{C}_1^T & \tilde{C}_2^T \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ \tilde{B}_1 & \tilde{B}_2 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{C}_1 & 0 \\ B_{\bar{W}}^T & \tilde{C}_2 & 0 \end{bmatrix} X & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{B}_1^T & 0 \\ 0 & \tilde{B}_2^T & 0 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11}^T & \tilde{D}_{21}^T \\ \tilde{D}_{12}^T & \tilde{D}_{22}^T \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} \end{pmatrix} \cdot \Psi_U$$

$$< \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_3^T \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_3 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right),$$

$$\hat{\Psi}_V^T \cdot \begin{pmatrix} \begin{bmatrix} A_{\tilde{W}} & 0 & 0 \\ 0 & \tilde{A}^T & 0 \\ 0 & 0 & 0 \end{bmatrix} X^{-1} + X^{-1} \begin{bmatrix} A_{\tilde{W}}^T & 0 & 0 \\ 0 & \tilde{A} & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & B_{\tilde{W}} \\ \tilde{C}_1^T & \tilde{C}_2^T \\ 0 & 0 \end{bmatrix} & X^{-1} \begin{bmatrix} 0 & 0 \\ \tilde{B}_1 & \tilde{B}_2 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{C}_1 & 0 \\ B_{\tilde{W}}^T & \tilde{C}_2 & 0 \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{B}_1^T & 0 \\ 0 & \tilde{B}_2^T & 0 \end{bmatrix} X^{-1} & \begin{bmatrix} \tilde{D}_{11}^T & \tilde{D}_{21}^T \\ \tilde{D}_{12}^T & \tilde{D}_{22}^T \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} \end{pmatrix} \cdot \hat{\Psi}_V$$

$$< \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 & 0 \\ 0 & \psi_4^T & 0 \end{bmatrix} X^{-1} \begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} X^{-1} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_4 \\ 0 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \right),$$

where Ψ_U and $\hat{\Psi}_V$ are defined as in (j). Observe that (k) is simply a restatement of (j) with each matrix written out in full so that its structure can be seen. This structure (i.e. the all-zero rows and columns) will be exploited next.

(l) $\exists\, s_K \in \mathbb{Z}_+$, $P = P^T \in \mathbb{R}^{(s_w+s_{\tilde{G}})\times(s_w+s_{\tilde{G}})}$ and $Q = Q^T \in \mathbb{R}^{(s_w+s_{\tilde{G}})\times(s_w+s_{\tilde{G}})}$ such that

$$P > 0, \quad Q > 0,$$

$$\begin{bmatrix} P & I \\ I & Q \end{bmatrix} \geq 0,$$

$$\operatorname{rank}\left(P - Q^{-1}\right) \leq s_K,$$

$$\Psi_P^T \cdot \begin{pmatrix} P \begin{bmatrix} A_{\tilde{W}} & 0 \\ 0 & \tilde{A}^T \end{bmatrix} + \begin{bmatrix} A_{\tilde{W}}^T & 0 \\ 0 & \tilde{A} \end{bmatrix} P & P \begin{bmatrix} 0 & B_{\tilde{W}} \\ \tilde{C}_1^T & \tilde{C}_2^T \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ \tilde{B}_1 & \tilde{B}_2 \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{C}_1 \\ B_{\tilde{W}}^T & \tilde{C}_2 \end{bmatrix} P & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{B}_1^T \\ 0 & \tilde{B}_2^T \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11}^T & \tilde{D}_{21}^T \\ \tilde{D}_{12}^T & \tilde{D}_{22}^T \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} \end{pmatrix} \cdot \Psi_P$$

$$< \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_3^T \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_3 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right),$$

$$\Psi_Q^T \cdot \begin{pmatrix} \begin{bmatrix} A_{\bar{W}} & 0 \\ 0 & \tilde{A}^T \end{bmatrix} Q + Q \begin{bmatrix} A_{\bar{W}}^T & 0 \\ 0 & \tilde{A} \end{bmatrix} & Q \begin{bmatrix} 0 & 0 \\ \tilde{B}_1 & \tilde{B}_2 \end{bmatrix} & \begin{bmatrix} 0 & B_{\bar{W}} \\ \tilde{C}_1^T & \tilde{C}_2^T \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{B}_1^T \\ 0 & \tilde{B}_2^T \end{bmatrix} Q & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11}^T & \tilde{D}_{21}^T \\ \tilde{D}_{12}^T & \tilde{D}_{22}^T \end{bmatrix} \\ \begin{bmatrix} 0 & \tilde{C}_1 \\ B_{\bar{W}}^T & \tilde{C}_2 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \cdot \Psi_Q$$

$$< \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_4^T \end{bmatrix} Q \begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} Q \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_4 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \right), \tag{B.5}$$

where Ψ_P and Ψ_Q are defined as in Part (ii) of the Theorem. Note that (k) $\Leftrightarrow$ (l) follows after some algebra by exploiting the all-zero rows in Ψ_U and $\hat{\Psi}_V$ and by applying Lemma 2.8.4.

(m) $\exists s_K \in \mathbb{Z}_+$, $P = P^T \in \mathbb{R}^{(s_w+s_{\tilde{G}})\times(s_w+s_{\tilde{G}})}$, $R = R^T \in \mathbb{R}^{s_w \times s_w}$, $S \in \mathbb{R}^{s_w \times s_{\tilde{G}}}$ and $T = T^T \in \mathbb{R}^{s_{\tilde{G}} \times s_{\tilde{G}}}$ such that

$$P > 0, \quad R > 0, \quad T > 0,$$

$$\begin{pmatrix} P & \begin{bmatrix} R & -S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} \\ \begin{bmatrix} R & 0 \\ -S^T & I_{s_{\tilde{G}}} \end{bmatrix} & \begin{bmatrix} R & 0 \\ 0 & T \end{bmatrix} \end{pmatrix} \geq 0,$$

$$\operatorname{rank} \left(\begin{bmatrix} I_{s_w} & S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} P \begin{bmatrix} I_{s_w} & 0 \\ S^T & I_{s_{\tilde{G}}} \end{bmatrix} - \begin{bmatrix} R & 0 \\ 0 & T^{-1} \end{bmatrix} \right) \leq s_K,$$

$$\Psi_P^T \cdot \begin{pmatrix} P \begin{bmatrix} A_{\bar{W}} & 0 \\ 0 & \tilde{A}^T \end{bmatrix} + \{\cdot\}^T & P \begin{bmatrix} 0 & B_{\bar{W}} \\ \tilde{C}_1^T & \tilde{C}_2^T \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ \tilde{B}_1 & \tilde{B}_2 \end{bmatrix} \\ * & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} \\ * & * & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} \end{pmatrix} \cdot \Psi_P$$

$$< \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_3^T \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & \psi_3 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right),$$

$$\Psi_Q^T \cdot \begin{pmatrix} \begin{bmatrix} R & S \\ -S^T & T \end{bmatrix} \begin{bmatrix} A_{\tilde{W}} & 0 \\ 0 & \tilde{A} \end{bmatrix} + \{\cdot\}^T & * & * \\ \begin{bmatrix} 0 & \tilde{B}_1^T \\ 0 & \tilde{B}_2^T \end{bmatrix} \begin{bmatrix} I_{s_w} & 0 \\ S^T & T \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & -I_m \end{bmatrix} & * \\ \begin{bmatrix} 0 & \tilde{C}_1 \\ B_{\tilde{W}}^T & \tilde{C}_2 \end{bmatrix} \begin{bmatrix} R & -S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} & \begin{bmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{21} & \tilde{D}_{22} \end{bmatrix} & \begin{bmatrix} -I_r & 0 \\ 0 & 0 \end{bmatrix} \end{pmatrix} \cdot \Psi_Q$$

$$< \begin{pmatrix} \begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \end{pmatrix} \breve{W} \left(\begin{bmatrix} I_{s_w} & 0 \\ 0 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 \\ 0 & I_n \end{bmatrix} \right), \quad \text{(B.6)}$$

where Ψ_P and Ψ_Q are defined as in Part (ii) of the Theorem. The equivalence (l) $\Leftrightarrow$ (m) follows by observing the following four facts:

- Any $Q = Q^T \in \mathbb{R}^{(s_w+s_{\tilde{G}})\times(s_w+s_{\tilde{G}})}$ satisfying $Q > 0$ can be decomposed as follows:

$$Q = \begin{bmatrix} I_{s_w} & 0 \\ S^T & I_{s_{\tilde{G}}} \end{bmatrix} \begin{bmatrix} R^{-1} & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} I_{s_w} & S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix},$$

  where $R = R^T \in \mathbb{R}^{s_w \times s_w}$, $S \in \mathbb{R}^{s_w \times s_{\tilde{G}}}$ and $T = T^T \in \mathbb{R}^{s_{\tilde{G}} \times s_{\tilde{G}}}$ are *completely independent* variables which fully parametrise Q. Furthermore, $Q > 0$ if and only if $R > 0$ and $T > 0$.

- Using such a decomposition for Q,

$$\begin{bmatrix} P & I \\ I & Q \end{bmatrix} \geq 0 \quad \Leftrightarrow \quad \begin{pmatrix} P & \begin{bmatrix} I_{s_w} & -S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} \\ \begin{bmatrix} I_{s_w} & 0 \\ -S^T & I_{s_{\tilde{G}}} \end{bmatrix} & \begin{bmatrix} R^{-1} & 0 \\ 0 & T \end{bmatrix} \end{pmatrix} \geq 0$$

$$\Leftrightarrow \quad \begin{pmatrix} P & \begin{bmatrix} R & -S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} \\ \begin{bmatrix} R & 0 \\ -S^T & I_{s_{\tilde{G}}} \end{bmatrix} & \begin{bmatrix} R & 0 \\ 0 & T \end{bmatrix} \end{pmatrix} \geq 0.$$

- Again, using the above decomposition for Q,

$$\begin{aligned} \operatorname{rank}\left(P - Q^{-1}\right) &= \operatorname{rank}\left(P - \begin{bmatrix} I_{s_w} & -S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} \begin{bmatrix} R & 0 \\ 0 & T^{-1} \end{bmatrix} \begin{bmatrix} I_{s_w} & 0 \\ -S^T & I_{s_{\tilde{G}}} \end{bmatrix}\right) \\ &= \operatorname{rank}\left(\begin{bmatrix} I_{s_w} & S \\ 0 & I_{s_{\tilde{G}}} \end{bmatrix} P \begin{bmatrix} I_{s_w} & 0 \\ S^T & I_{s_{\tilde{G}}} \end{bmatrix} - \begin{bmatrix} R & 0 \\ 0 & T^{-1} \end{bmatrix}\right). \end{aligned}$$

- Finally, the equivalence between inequality (B.5) and inequality (B.6) is obtained through the application of the congruence transformation

$$\begin{pmatrix} \begin{bmatrix} R & -S\psi_4 \\ 0 & I \end{bmatrix} & 0 \\ 0 & \begin{bmatrix} I_r & 0 \\ 0 & I_n \end{bmatrix} \end{pmatrix}$$

  on inequality (B.5).

(n) The proof is completed by noting that for any $s_K \geq (s_W + s_{\tilde{G}})$, the 'rank' constraint in (m) is redundant and the remaining conditions are exactly those stated in Part (ii) of the Theorem. The controller order s_K can be chosen as desired since the set $\mathcal{K}_G^{TF}$ contains all internally stabilising controllers for G of *any* order (as stated at the beginning of the proof).

B.4 Proof of Corollary 4.7.2

The additional 'rank' condition for the existence of internally stabilising controllers of order s_K directly follows from Step (m) in the proof of Theorem 4.7.1 given in Appendix B.3, whereas the LMI for the construction of such controllers follows from Step (i) of the same proof on noting that it is always possible to construct an $X = X^T \in \mathbb{R}^{(s_W+s_{\tilde{G}}+s_K)\times(s_W+s_{\tilde{G}}+s_K)}$ satisfying

$$X = \begin{bmatrix} P & \blacklozenge \\ \blacklozenge & \blacklozenge \end{bmatrix} > 0 \qquad \text{and} \qquad X^{-1} = \begin{bmatrix} Q & \blacklozenge \\ \blacklozenge & \blacklozenge \end{bmatrix},$$

where $\blacklozenge$ denotes "Don't Care" elements. This is because given

$$X = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^T & X_{22} \end{bmatrix} > 0,$$

it is easy to verify that

$$X^{-1} = \begin{bmatrix} (X_{11} - X_{12}X_{22}^{-1}X_{12}^T)^{-1} & \blacklozenge \\ \blacklozenge & \blacklozenge \end{bmatrix}.$$

Consequently, if $X_{11} = P$ and $(X_{11} - X_{12}X_{22}^{-1}X_{12}^T)^{-1} = Q$, then one possible way of constructing the matrix X using the given P and Q is by letting $X_{22} = I$ and computing X_{12} from the following factorisation $P - Q^{-1} = X_{12}X_{12}^T$.

Bibliography

Anderson, B. D. O. and Moore, J. B. (1989). *Optimal Control — Linear Quadratic Methods*. Prentice-Hall, Inc.

Balas, G. J. and Doyle, J. C. (1990). Robustness and performance tradeoffs in control design for flexible structures. In *Proceedings of the 29th IEEE Conference on Decision and Control*, volume 6, pages 2999–3010, Honolulu, Hawaii.

Balas, G. J., Doyle, J. C., Glover, K., Packard, A., and Smith, R. (1996). *μ-Analysis and Synthesis Toolbox: User's Guide*. The MathWorks, Inc. For use with MATLAB.

Bollobás, B. (1992). *Linear Analysis: An Introductory Course*. Cambridge Mathematical Textbooks. Cambridge University Press.

Boyd, S. and Desoer, C. A. (1985). Subharmonic functions and performance bounds in linear time-invariant feedback systems. *IMA Journal of Mathematics, Control and Information*, 2:153–170.

Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear Matrix Inequalities in System and Control Theory*. SIAM Studies in Applied Mathematics. Society for Industrial and Applied Mathematics.

Brewer, J. W. (1978). Kronecker products and matrix calculus in system theory. *IEEE Transactions on Circuits and Systems*, 25(9):772–781.

Chang, C.-Y., Yang, C.-D., and Lin, C. E. (1994). A new iteration scheme for robust performance problem: E-K iterations. In *Proceedings of the American Control Conference*, pages 2809–2813, Baltimore, MD.

Desoer, C. A. and Vidyasagar, M. (1975). *Feedback Systems: Input-Output Properties*. Academic Press.

Doyle, J. C. (1985). Structured uncertainty in control systems design. In *Proceedings of the 24th IEEE Conference on Decision and Control*, pages 260–265, Ft. Lauderdale, FL.

Doyle, J. C., Francis, B. A., and Tannenbaum, A. R. (1992). *Feedback Control Theory*. Macmillan Publishing Company.

Doyle, J. C., Glover, K., Khargonekar, P. P., and Francis, B. A. (1989). State-space solutions to the standard $\mathscr{H}_2$ and $\mathscr{H}_\infty$ control problems. *IEEE Transactions on Automatic Control*, 34(8):831–847.

Doyle, J. C., Packard, A., and Zhou, K. (1991). Review of LFTs, LMIs and μ. In *Proceedings of the 30th IEEE Conference on Decision and Control*, pages 1227–1232, England.

Fan, M. K. H. and Tits, A. L. (1992). A measure of worst-case $\mathscr{H}_\infty$ performance and of largest acceptable uncertainty. *Systems and Control Letters*, 18(6):409–421.

Francis, B. A. (1987). *A course in $\mathscr{H}_\infty$ control theory*, volume 88 of *Lecture notes in Control and Information Sciences*. Springer-Verlag, first edition.

Gahinet, P. and Apkarian, P. (1994). A Linear Matrix Inequality approach to $\mathscr{H}_\infty$ control. *International Journal of Robust and Nonlinear Control*, 4:421–448.

Glover, K. (1984). All optimal Hankel-norm approximations of linear multivariable systems and their $\mathscr{L}_\infty$-error bounds. *International Journal of Control*, 39(6):1115–1193.

Glover, K. and Doyle, J. C. (1988). State-space formulae for all stabilizing controllers that satisfy an $\mathscr{H}_\infty$ norm bound and relations to risk sensitivity. *Systems and Control Letters*, 11:167–172.

Glover, K. and McFarlane, D. (1989). Robust stabilization of normalized coprime factor plant descriptions with $\mathscr{H}_\infty$-bounded uncertainty. *IEEE Transactions on Automatic Control*, 34(8):821–830.

Glover, K., Vinnicombe, G., and Papageorgiou, G. (2000). Guaranteed multi-loop stability margins and the gap metric. In *Proceeding of the 39th IEEE Conference on Decision and Control*, Sydney, Australia.

Golub, G. H. and Van Loan, C. F. (1996). *Matrix Computations*. The Johns Hopkins University Press, third edition.

Green, M. and Limebeer, D. J. N. (1995). *Linear Robust Control*. Prentice-Hall, Inc.

Haggarty, R. (1994). *Fundamentals of Mathematical Analysis*. Addison-Wesley, second edition.

Halsey, K. M. (2000). Robust control of highly manoeuvrable air-vehicles. First year report (for Ph.D. degree), University of Cambridge, United Kingdom.

Helmersson, A. (1995). *Methods for Robust Gain Scheduling*. Number 406 in Linköping Studies in Science and Technology, Dissertations. Linköping University, Sweden.

Horn, R. A. and Johnson, C. R. (1996). *Matrix Analysis*. Cambridge University Press.

Hyde, R. A. (1995). $\mathscr{H}_\infty$ aerospace control design — a VSTOL flight application. In *Advances in Industrial Control Series*. Springer-Verlag.

Jovik, I. and Lennartson, B. (1996). On the choice of criteria and weighting functions of an $\mathscr{H}_2/\mathscr{H}_\infty$-design problem. In *IFAC Symposia Series — Proceedings of the 13th Triennial World Congress*, pages 373–378, San Francisco, USA.

Kreyszig, E. (1989). *Introductory Functional Analysis with Applications*. John Wiley and Sons.

Kwakernaak, H. (1993). Robust control and $\mathscr{H}_\infty$-optimisation — tutorial paper. *Automatica*, 29(2):255–273.

Lanzon, A. and Cantoni, M. (2000). A state-space algorithm for the simultaneous optimisation of performance weights and controllers in μ-synthesis. In *Proceeding of the 39th IEEE Conference on Decision and Control*, Sydney, Australia.

Lanzon, A. and Richards, R. J. (1999). A frequency domain optimisation algorithm for simultaneous design of performance weights and controllers in μ-synthesis. In *Proceedings of the 38th IEEE Conference on Decision and Control*, Phoenix, Arizona.

Lin, J.-L., Postlethwaite, I., and Gu, D.-W. (1993). μ-K iteration: A new algorithm for μ synthesis. *Automatica*, 29(1):219–224.

Maciejowski, J. M. (1994). *Multivariable Feedback Design*. Addison-Wesley.

McFarlane, D. and Glover, K. (1992). A loop shaping design procedure using $\mathscr{H}_\infty$ synthesis. *IEEE Transactions on Automatic Control*, 37(6):759–769.

Packard, A. (1994). Gain scheduling via linear fractional transformations. *Systems and Control Letters*, 22(2):79–92.

Packard, A. and Doyle, J. (1993). The complex structured singular value. *Automatica*, 29(1):71–109.

Papageorgiou, G. and Glover, K. (1997). A systematic procedure for designing non-diagonal weights to facilitate $\mathscr{H}_\infty$ loop-shaping. In *Proceedings of the 36th IEEE Conference on Decision and Control*, pages 2127–2132, San Diego, CA.

Papageorgiou, G. and Glover, K. (1999a). Design, development and control of the HIRM wind tunnel model. In *Proceedings of the 38th IEEE Conference on Decision and Control*, Phoenix, Arizona.

Papageorgiou, G. and Glover, K. (1999b). $\mathscr{H}_\infty$ loop-shaping: Why is it a sensible procedure for designing robust flight controllers? In *Proceedings of the AIAA Conference on Guidance, Navigation and Control*.

Poolla, K. and Tikku, A. (1995). Robust performance against time-varying structured perturbations. *IEEE Transactions on Automatic Control*, 40(9):1589–1602.

Rantzer, A. (1996). On the Kalman-Yakubovich-Popov lemma. *Systems and Control Letters*, 28(1):7–10.

Rotea, M. A. and Iwasaki, T. (1994). An alternative to the D-K iteration? In *Proceedings of the American Control Conference*, pages 53–57, Baltimore, MD.

Rudin, W. (1976). *Principles of Mathematical Analysis*. Mathematics Series. McGraw-Hill International Editions, third edition.

Rudin, W. (1987). *Real and Complex Analysis*. Mathematics Series. McGraw-Hill International Editions, third edition.

Safonov, M. G. and Limebeer, D. J. N. (1988). Simplifying the $\mathscr{H}_\infty$ theory via loop shifting. In *Proceedings of the 27th IEEE Conference on Decision and Control*, pages 1399–1404, Austin, Texas.

Skogestad, S. and Postlethwaite, I. (1998). *Multivariable Feedback Control: Analysis and Design*. John Wiley and Sons.

Stoughton, R. M. (1990). Formulation of an improved set of weighting functions for $\mathscr{H}_\infty$ control of flexible beam-like systems. In *Proceedings of the American Control Conference*, pages 1745–1751, San Diego, CA, USA.

Vinnicombe, G. (2000). *Uncertainty and Feedback: $\mathscr{H}_\infty$ loop-shaping and the ν-gap metric*. Imperial College Press. In press.

Willems, J. C. (1971). Least squares stationary optimal control and the algebraic Riccati equation. *IEEE Transactions on Automatic Control*, 16(6):621–634.

Young, N. (1988). *An Introduction to Hilbert Space*. Cambridge University Press.

Zames, G. (1966). On the input-output stability for time-varying nonlinear feedback systems – Part I: Conditions using concepts of loop gain, conicity and positivity. *IEEE Transactions on Automatic Control*, 11(2):228–238.

Zames, G. (1981). Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms and approximate inverses. *IEEE Transactions on Automatic Control*, 26:301–320.

Zhou, K. and Doyle, J. C. (1998). *Essentials of Robust Control*. Prentice-Hall, Inc.

Zhou, K., Doyle, J. C., and Glover, K. (1996). *Robust and Optimal Control*. Prentice-Hall, Inc.