INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL
Int. J. Robust Nonlinear Control 2005; **15**:171–199
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/rnc.984

Pointwise in frequency performance weight optimization in μ-synthesis

Alexander Lanzon[1,2,*,†]

[1] *Department of Information Engineering, Research School of Information Sciences and Engineering, The Australian National University, Canberra ACT 0200, Australia*
[2] *National ICT Australia Ltd., Locked Bag 8001, Canberra ACT 2601, Australia*

SUMMARY

A conceptually different approach to the μ-synthesis robust performance problem is proposed in this article. The optimization problem posed maximizes the performance weights with respect to a suitable cost function that captures the desired closed-loop performance. This maximization of performance weights is limited by the fact that there must exist some internally stabilizing controller that guarantees robust performance with respect to these maximized weights. Thus, performance weights and a controller that achieves an optimized level of robust performance are synthesized together by one algorithm in a systematic way. The designer is only required to specify the plant set and an optimization 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 optimization problem is also developed in this article. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: optimizing performance; performance weight synthesis; robust performance; μ-synthesis; D–K iterations; skewed-μ; $\mathscr{H}_\infty$-control

1. INTRODUCTION

Due to recent developments in the area of numerical optimization that have led to a widening of the class of addressable problems, many control systems design problems are nowadays cast in optimization-based frameworks [1–4]. In these frameworks, desired performance and robustness requirements are typically specified via weights, which are regarded by control systems engineers as the 'tuning knobs' of the design. Once such weights are specified, a controller is synthesized to

*Correspondence to: Alexander Lanzon, Department of Information Engineering, Research School of Information Sciences and Engineering, The Australian National University, Canberra ACT 0200, Australia.
†E-mail: Alexander.Lanzon@anu.edu.au

Contract/grant sponsor: ARC Discovery-Projects Grant; contract/grant number: DP0342683

Received 13 November 2003
Accepted 3 November 2004

achieve the performance and robustness level specified by the weights by solving a correspondingly weighted optimization problem.

However, it is widely known in the practising engineering community that the design of such weights is often a challenging task requiring a long and tedious trial and error process based in large part on engineering judgement and intuition [5–7]. This problem is particularly acute when the performance specifications, the robustness requirements and/or fundamental plant limitations (e.g. due to right-half plane poles and zeros) are incompatible. This incompatibility frequently occurs when designers seek to achieve the best possible performance/robustness for a particular design. In this paper, a new optimization problem is proposed that facilitates the selection of such weights (or more accurately, that synthesizes optimal weights) for a class of robust performance problems.

Over the last decade or so, $\mathscr{H}_\infty$ control techniques have become popular in control systems design since these systematic techniques handle robustness issues explicitly [8–10]. An internally stabilizing feedback controller is said to achieve robust performance if a certain level of closed-loop performance is achieved for all plants in a specified set. Again, the level of performance is specified through the so-called performance weights and the set of plants for which we seek robust performance is typically characterized by a nominal plant surrounded by a ball of norm bounded uncertainty. When the uncertainty characterizing the set of plants to be controlled is structured, use of the structured singular value (i.e. μ) results into the μ-synthesis method [11, 12] (whose essence is rooted in $\mathscr{H}_\infty$ control theory) for designing controllers that achieve robust performance with respect to the pre-specified weights. A clever trick also allows us to use the structured singular value (i.e. μ) to convert the problem of synthesizing a controller that achieves robust performance into one of synthesizing a controller that achieves robust stability [10, 11].

The D–K iterative procedure of References [11, 12] is probably the most popular method used today in μ-synthesis to design robustly stabilizing controllers. Other methods with different computational benefits have later been proposed, such as μ–K iterations in Reference [13], E–K iterations in Reference [14] and L–R iterations in Reference [15]. However, all these methods assume that the performance weights have already been chosen. Some authors have suggested 'rules' for choosing such performance weights for specific design problems [5, 7]. 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 Reference [16], a mathematical quantity (closely related to μ and commonly referred to as the skewed-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 optimizing 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 article, the following more general problem is studied: 'Determine the largest performance weights (in some sense, at each frequency and in all channel directions) such that for any uncertainty bounded by unity, the $\mathscr{H}_\infty$ performance level characterized by these performance weights is guaranteed'.

Consequently, in the proposed optimization-based procedure, we wish to synthesize the largest performance weights, in some sense, subject to ensuring that there exists[‡] an internally stabilizing controller that guarantees robust performance with respect to the maximized weights.

[‡] We also wish to be able to explicitly construct this internally stabilizing controller.

In the proposed optimization procedure, the designer is required to specify the uncertain plant set by prescribing a generalized plant, an uncertainty block structure and an associated norm bound (which is typically normalized to unity). In addition to this, the designer must specify an optimization directionality, which appears in the cost associated with the optimization problem by which the performance weights and the internally stabilizing controller are synthesized. The optimization directionality qualitatively reflects desired performance objectives over all frequency, and should be specified as small (resp. large) at frequencies and in channel directions where the performance weight would be required to be small (resp. large). It will be seen in this paper that it is much easier to specify the optimization directionality than the performance weight directly itself, since incompatible specifications will be resolved through the constrained optimization. In this sense, the proposed procedure provides an indication of achievable performance, in addition to a controller achieving robust performance with respect to the suitably optimized performance weights and the uncertain plant set.

The interested reader is also referred to Reference [17], where a similar concept was explored for the $\mathscr{H}_\infty$ loop-shaping design paradigm (as opposed to μ-synthesis type problems investigated here), and to Reference [18] where the effects of small weight changes on a synthesized $\mathscr{H}_\infty$ controller and the resulting closed-loop transfer function matrices are studied in some detail.

This paper is organized as follows: in Section 2, we formulate the problem of interest. We start by introducing the framework in which μ-synthesis problems are frequently cast, and then we formally pose the optimization problem of interest. In Section 3, we rewrite the posed optimization problem using the well known Youla parametrization of all stabilizing controllers and the resulting stable closed-loops so that the resulting optimization problem is easier to handle. In Section 4, we formulate a tractable optimization problem that has tighter constraints than the posed optimization problem and hence this new problem guarantees the same desirable closed-loop properties of the posed optimization problem. In Section 5, we discuss a pointwise in frequency approximation of the tractable optimization problem so that an iterative solution algorithm can be given in Section 6 to this pointwise in frequency approximation. A numerical example that illustrates the use of this conceptually different method is given in Section 8 and finally some concluding remarks are given in Section 9.

2. PROBLEM FORMULATION

Most linear interconnection of systems, inputs, outputs and model uncertainties can be redrawn into the linear fractional transformation (LFT) framework [8, 12] depicted in Figure 1, where G is the generalized plant, Δ is the uncertainty in the system and K is an internally stabilizing controller. For notational convenience, all uncertainty blocks in Δ are assumed to be square. This can be done without loss of generality by adding dummy inputs or outputs and padding the associated systems by all zero rows or columns [19]. Since the uncertainty Δ can in general be structured, let the set $\mathbf{\Delta} := \{\mathrm{diag}(\delta_1 I_{\alpha_1}, \ldots, \delta_s I_{\alpha_s}, \Delta_1, \ldots, \Delta_f) : \delta_i \in \mathbb{C}, \Delta_i \in \mathbb{C}^{\beta_i \times \beta_i}, \sum_{i=1}^{s} \alpha_i + \sum_{i=1}^{f} \beta_i = r\}$ denote its structure and the set $\mathbf{B\Delta}^{\mathrm{TF}} := \{\Delta \in \mathscr{RH}_\infty : \Delta(s_o) \in \mathbf{\Delta}\ \forall s_o \in \overline{\mathbb{C}}_+, \|\Delta\|_\infty \leqslant 1\}$ denote a normalized ball of stable uncertainty with this structure. Thus, in the sequel, we will always assume that $\Delta \in \mathbf{B\Delta}^{\mathrm{TF}}$ and hence has been appropriately normalized.

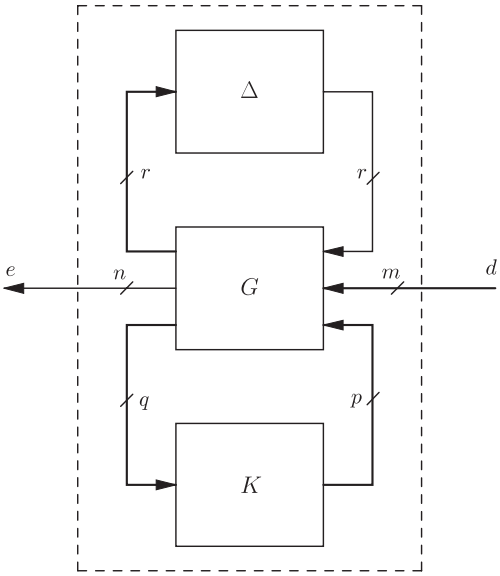

Figure 1. Linear fractional transformation framework.

Furthermore, in what follows, we always require $K \in \mathscr{K}_G^{\mathrm{TF}}$, where $\mathscr{K}_G^{\mathrm{TF}}$ denotes the set of all internally stabilizing controllers $K \in \mathscr{R}^{p\times q}$ for the feedback interconnection $\mathscr{F}_l(G,K)$.

In Figure 1, d denotes the exogenous signals entering the feedback interconnection and e denotes the error signals that need to be attenuated. Since each element in the vector e needs to be attenuated by different amounts and in different frequency regions, e is typically multiplied by a diagonal frequency dependent weight W (known as a 'performance weight' since it is used to capture the performance specifications) to give a normalized error $e' = We$. Then one typically wishes to achieve

$$\|T_{e'd}\|_\infty = \|WT_{ed}\|_\infty = \|W\mathscr{F}_u(\mathscr{F}_l(G,K),\Delta)\|_\infty < 1 \quad \forall \Delta \in \mathbf{B}\boldsymbol{\Delta}^{\mathrm{TF}} \tag{1}$$

as this will guarantee that the robust performance level specified by the performance weight $W \in \mathscr{W}^{\mathrm{TF}} := \{\mathrm{diag}_{i=1}^n(w_i) : w_i, w_i^{-1} \in \mathscr{RH}_\infty\}$ is achieved for all perturbed plants in the specified plant set.

In this paper, we will take a different approach to what is standard practice in robust control literature. As stated in the introduction, here *we wish to maximize the size* (*in some sense*) *of the performance weight W limited by the requirement that there must exist an internally stabilizing controller K that achieves robust performance with respect to this maximized weight*. Thus the performance weight $W \in \mathscr{W}^{\mathrm{TF}}$ will be synthesized in this article, rather than specified. The statement in italics can be formulated mathematically as an optimization problem of the form

$$\max_{W\in\mathscr{W}^{\mathrm{TF}}} J(W) \quad \text{subject to} \quad \min_{K\in\mathscr{K}_G^{\mathrm{TF}}} \max_{\Delta\in\mathbf{B}\boldsymbol{\Delta}^{\mathrm{TF}}} \|W\mathscr{F}_u(\mathscr{F}_l(G,K),\Delta)\|_\infty < 1 \tag{2}$$

where $J(W)$ is some cost function that captures the size of the performance weight W in some sense. It is well known [20, Theorem 5.4] that condition (1) is equivalent to

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

Here, $\mathbf{\Delta}_{\mathrm{TOT}} := \{\mathrm{diag}(\Delta, \Delta_P) : \Delta \in \mathbf{\Delta}, \Delta_P \in \mathbb{C}^{m\times n}\}$ denotes the total uncertainty structure with respect to which the structured singular value $\mu_{\mathbf{\Delta}_{\mathrm{TOT}}}[\cdot]$ is computed.

Now let us describe in what sense we wish to maximize the performance weight W. First note that the direction of steepest ascent in maximizing a cost function of the form

$$\frac{1}{(1/R_1) + \cdots + (1/R_N)}$$

over any one $R_i > 0$ is always the direction corresponding to the smallest R_i. Analogously, the direction of steepest ascent in maximizing a cost function of the form

$$\frac{1}{\int_{\log_{10}\omega_{\mathrm{L}}}^{\log_{10}\omega_{\mathrm{H}}} \sum_{i=1}^{n} \left(1 \Big/ \left|\frac{w_i(j\omega)}{\upsilon_i(j\omega)}\right|^2\right)\, d(\log_{10}\omega)} \tag{3}$$

over any one weight $|w_i(\cdot)|$ at any one frequency ω (for a fixed $|\upsilon_i(j\omega)|$) is always the direction corresponding to the smallest $|w_i(j\omega)/\upsilon_i(j\omega)|$, since the integral simply represents incremental summation over frequency. Logarithmic frequency is used as the variable of integration instead of linear frequency so that equal importance is given to low frequencies ($\omega \ll 1$) as is given to high frequencies ($\omega \gg 1$) and furthermore, so that $\int_{\log_{10}\omega_{\mathrm{L}}}^{\log_{10}\omega_{\mathrm{H}}} |\upsilon_i(j\omega)/w_i(j\omega)|^2 d(\log_{10}\omega)$ has direct interpretation when $|\upsilon_i(j\omega)/w_i(j\omega)|^2$ is plotted on a Bode diagram. This integral in fact represents the area below the Bode plot of $|\upsilon_i(j\omega)/w_i(j\omega)|^2$ between frequencies ω_{L} and ω_{H}. The following definition allows us to write cost function (3) in a more compact form.

Definition 1

Given a system $P \in \mathscr{RH}_\infty$ and a frequency range $[\omega_{\mathrm{L}}, \omega_{\mathrm{H}}]$ with $0 < \omega_{\mathrm{L}} \leqslant \omega_{\mathrm{H}} < \infty$, let $\|P\|_{[\omega_{\mathrm{L}},\omega_{\mathrm{H}}]}$ be defined by

$$\|P\|_{[\omega_{\mathrm{L}},\omega_{\mathrm{H}}]} := \sqrt{\int_{\log_{10}\omega_{\mathrm{L}}}^{\log_{10}\omega_{\mathrm{H}}} \|P(j\omega)\|_F^2\, d(\log_{10}\omega)}$$

Since $w_i(j\omega)$ in cost function (3) is the ith diagonal element of $W(j\omega)$, one can pack in a similar way $\upsilon_i(j\omega)$ into $\Upsilon(j\omega) := \mathrm{diag}(\upsilon_1(j\omega), \ldots, \upsilon_n(j\omega))$ to give

$$\frac{1}{\|\Upsilon W^{-1}\|^2_{[\omega_{\mathrm{L}},\omega_{\mathrm{H}}]}} = \frac{1}{\int_{\log_{10}\omega_{\mathrm{L}}}^{\log_{10}\omega_{\mathrm{H}}} \sum_{i=1}^{n} \dfrac{1}{|w_i(j\omega)/\upsilon_i(j\omega)|^2}\, d(\log_{10}\omega)}$$

This cost function is a cumulative measure of the frequency dependent size of the performance weight $W(j\omega)$ in the frequency range $[\omega_{\mathrm{L}}, \omega_{\mathrm{H}}]$. Each performance weight $w_i(j\omega)$ is scaled differently across frequency in this cost function due to the fixed quantities $\upsilon_i(j\omega)$. In the sequel, Υ will be called the 'directionality transfer function matrix' as it is made up from little υ_i which can be specified by the designer to direct the maximization 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 maximization will only take place in the frequency range $[\omega_L, \omega_H]$, $\Upsilon(j\omega)$ is only relevant in this frequency range.

Consequently, given a generalized plant G, a frequency range $[\omega_L, \omega_H]$ and a directionality transfer function matrix $\Upsilon \in \mathbf{\Upsilon}^{TF} := \{\text{diag}_{i=1}^{n}(\upsilon_i) : \upsilon_i \in \mathscr{RH}_\infty\}$, optimization problem (2) can be written as

$$\max_{W \in \mathscr{W}^{TF}} \frac{1}{\|\Upsilon W^{-1}\|_{[\omega_L, \omega_H]}}$$

$$\text{subject to} \tag{4}$$

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

Here, $[\omega_L, \omega_H]$ is the frequency range where maximization 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 optimization problem.

It is important to realize that the directionality transfer function matrix Υ is specified by the designer to capture qualitatively the desired specifications. It is hence *not* a substitute for the performance weight $W(j\omega)$. For instance, 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 optimization problem. Only the shape across frequency and the relative sizes amongst the different diagonal entries of $\Upsilon(j\omega)$ are of relevance. Furthermore, incompatible directionalities can never be specified, unlike directly specifying the performance weights. This is because the performance weights given by the above optimization problem must always be feasible to the optimization'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 useful controller that satisfies reasonable stability/performance requirements (e.g. small sensitivity at low frequency and small complementary sensitivity at high frequency).

It is worth noting that it is not a requirement of the new method that specification of the performance weight W be left entirely upto the optimization procedure and judicious selection of the optimization directionality (even though this is certainly possible). If the designer has an initial weight, then this weight can be absorbed into the generalized plant G and the optimization-based synthesis of W can then be thought of as a systematic mechanism for 'tuning' the design.

One should point out at this stage that optimization problem (4) has some similarities with the corresponding optimization problem proposed in Reference [21], since both problems seek to find the largest (in some sense) robust performance level achievable for the specified plant set. However, this similarity is only at the conceptual level (i.e. at the level of posing the problem of interest) and when both manuscripts are analysed closely, one recognizes a number of significant differences which cannot in any way be overlooked. For example, in this paper the posed problem is solved entirely in the frequency domain, unlike in Reference [21] where the solution is based on state-space data and techniques, and hence here we do not need to construct rational D-scales and performance weights from the pointwise data as these are constructed and used

pointwise in frequency. Also, unlike Reference [21], here we do not need to specify a basis for the poles of the performance weights and the D-scales as the problem is convex in the frequency data of these variables. Furthermore, as stated above, logarithmic frequency is used in the cost function of this article so as to give low frequencies ($\omega \ll 1$) equal importance as high frequencies ($\omega \gg 1$) and also for easy interpretation of the cost function as the area below the singular values on a Bode-diagram. Other technical differences are due to the fact that here weights are maximized at one step of the solution algorithm and extra freedom generated at the other step for subsequent weight maximization, unlike in Reference [21] where weights are maximized at both steps in the iterative solution. Also, here we consider a finite frequency range around crossover for optimization of the cost function (thus not requiring the directionality matrices to be strictly proper), and here parametrization of the Youla parameter is done via a Laguerre basis.

3. REWRITING THE OPTIMIZATION PROBLEM

In this section, optimization problem (4) is rewritten in a form which is more suitable for subsequent synthesis steps. This rewriting will make use of the following well-known Youla parametrization theorem [8, 10]

Theorem 1 (*Youla Parametrization*)
Given a generalized 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 1 with (A, B_3) stabilizable and (A, C_3) detectable, let F and L be such that $A + B_3F$ and $A + LC_3$ are Hurwitz and define

$$J := \left[\begin{array}{c|cc} A + B_3F + LC_3 + LD_{33}F & -L & B_3 + LD_{33} \\ \hline F & 0 & I \\ -C_3 - D_{33}F & I & -D_{33} \end{array}\right]$$

Then the set $\mathscr{K}_G^{\mathrm{TF}}$ is parametrized by

$$\{K = \mathscr{F}_l(J, Q) : Q \in \mathscr{RH}_\infty\}$$

and the set of all closed-loop transfer function matrices $\mathscr{F}_l(G, K)$ achievable by an internally stabilizing controller $K \in \mathscr{K}_G^{\mathrm{TF}}$ is parametrized by

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

where T is given by

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

Observe that the mapping from $Q \in \mathscr{RH}_\infty$ to $K \in \mathscr{K}_G^{\mathrm{TF}}$ is bijective. Furthermore, all transfer function matrices T_{ij} are stable and the parametrization $\mathscr{F}_l(T, Q)$ is affine in the parameter Q. Since G is given at the beginning of the problem, T can be computed prior to evaluating the optimization.

Consequently, optimization problem (4) can equivalently be rewritten as

$$\max_{W \in \mathscr{W}^{\mathrm{TF}}} \frac{1}{\|\Upsilon W^{-1}\|_{[\omega_{\mathrm{L}},\omega_{\mathrm{H}}]}}$$

subject to

$$\min_{Q \in \mathscr{RH}_\infty} \sup_{\omega} \mu_{\mathbf{\Delta}_{\mathrm{TOT}}}\left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathscr{F}_l(T(j\omega), Q(j\omega))\right] < 1$$

and since only the arguments of the optimization problem are of interest, this optimization problem may be rewritten as

$$\begin{gathered} \min_{W \in \mathscr{W}^{\mathrm{TF}}} \|\Upsilon W^{-1}\|^2_{[\omega_{\mathrm{L}},\omega_{\mathrm{H}}]} \\ \text{subject to} \\ \exists\, Q \in \mathscr{RH}_\infty \text{ satisfying } \mu_{\mathbf{\Delta}_{\mathrm{TOT}}}\left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathscr{F}_l(T(j\omega), Q(j\omega))\right] < 1 \quad \forall \omega \end{gathered} \tag{5}$$

Note that, unfortunately, this optimization problem is non-convex (due to the μ constraint) and hence its solution is not easily computable.

4. A TRACTABLE REFORMULATION

In this section, each part of optimization (5) will be investigated separately and a computationally tractable optimization problem with tighter (i.e. more restrictive) constraints will be derived. The derivation is divided into several sub-sections for clarity.

4.1. Rewriting the cost function

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

Definition 2
The sets of strictly-positive and non-negative vector valued functions are defined by

$$\mathscr{V} \coloneqq \{f : \mathbb{R} \mapsto \mathbb{R}_+^n\}$$
$$\overline{\mathscr{V}} \coloneqq \{f : \mathbb{R} \mapsto \overline{\mathbb{R}}_+^n\}$$

where $\mathbb{R}_+$ denotes the strictly positive real numbers and $\overline{\mathbb{R}}_+$ denotes the non-negative real numbers.

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

$$v_W(\omega) \coloneqq [W(j\omega)^* W(j\omega)]^{-1} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} |w_1(j\omega)|^{-2} \\ |w_2(j\omega)|^{-2} \\ \vdots \\ |w_n(j\omega)|^{-2} \end{bmatrix} \in \mathscr{V} \tag{6}$$

and

$$v_\Upsilon(\omega) \coloneqq [\Upsilon(j\omega)^* \Upsilon(j\omega)] \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} |\upsilon_1(j\omega)|^2 \\ |\upsilon_2(j\omega)|^2 \\ \vdots \\ |\upsilon_n(j\omega)|^2 \end{bmatrix} \in \overline{\mathscr{V}} \tag{7}$$

Using this notation, it is easy to see that

$$\begin{aligned} \|\Upsilon W^{-1}\|^2_{[\omega_L,\omega_H]} &= \int_{\log_{10}\omega_L}^{\log_{10}\omega_H} \operatorname{trace}(W(j\omega)^{-*}\Upsilon(j\omega)^*\Upsilon(j\omega)W(j\omega)^{-1})\, d(\log_{10}\omega) \\ &= \int_{\log_{10}\omega_L}^{\log_{10}\omega_H} \operatorname{trace}\left(\Upsilon(j\omega)^*\Upsilon(j\omega)W(j\omega)^{-1}W(j\omega)^{-*}\right) d(\log_{10}\omega) \\ &= \int_{\log_{10}\omega_L}^{\log_{10}\omega_H} \operatorname{trace}([\Upsilon(j\omega)^*\Upsilon(j\omega)][W(j\omega)^*W(j\omega)]^{-1})\, d(\log_{10}\omega) \\ &= \int_{\log_{10}\omega_L}^{\log_{10}\omega_H} v_\Upsilon(\omega)^{\mathrm{T}} v_W(\omega)\ d(\log_{10}\omega) \end{aligned} \tag{8}$$

The cost function in this last form is more suitable for subsequent operations as will be seen later. Note that $v_\Upsilon(\omega) \in \overline{\mathscr{V}}$ is determined by the designer on specification of $\Upsilon(s) \in \mathbf{\Upsilon}^{\mathrm{TF}}$. Furthermore, given a continuous $v_W(\omega) \in \mathscr{V}$, it is always possible to construct a $W(s) \in \mathscr{W}^{\mathrm{TF}}$ that satisfies Equation (6) by fitting a stable minimum-phase transfer function to each magnitude function.

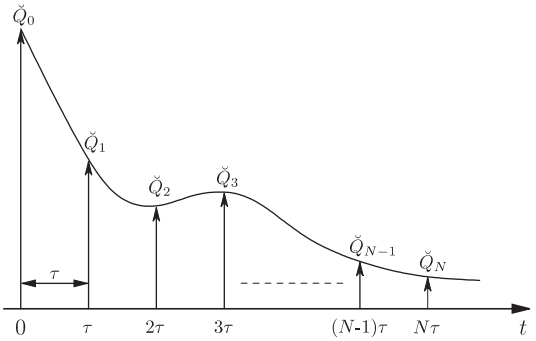

Figure 2. Impulse response in discrete and continuous time.

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 parametrizing a subspace of $\mathscr{RH}_\infty$ as follows:

$$Q(s) = \breve{Q}B(s) \quad \text{where} \quad \begin{cases} \breve{Q} := [\breve{Q}_0 \;\; \breve{Q}_1 \;\; \breve{Q}_2 \;\; \dots \;\; \breve{Q}_N] \in \mathbb{R}^{p\times(N+1)q} \\ B(s) := \left[I_q \;\; \left(\frac{(2/\tau)-s}{(2/\tau)+s}\right) I_q \;\; \left(\frac{(2/\tau)-s}{(2/\tau)+s}\right)^2 I_q \;\; \dots \;\; \left(\frac{(2/\tau)-s}{(2/\tau)+s}\right)^N I_q \right]^{\mathrm{T}} \end{cases} \tag{9}$$

This parametrization 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 2.

The $\mathscr{Z}$-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 $z^{-1} = \left(\frac{(2/\tau)-s}{(2/\tau)+s}\right)$ yields

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

This is the Laguerre-like parametrization stated in Equation (9).

4.3. *A sufficient condition so that* $\mu(\cdot) < 1$

Since the μ constraint in optimization problem (5) is not easily computable, it is necessary to replace $\mu_{\mathbf{\Delta}_{\mathrm{TOT}}}[\cdot]$ with some computationally tractable upper bound. A sufficient condition[§] will

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

thus be presented in this sub-section which ensures that

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

Before giving this computationally tractable upper bound, let $\mathscr{D}$ be the set of scalings (called D-scales) that commute with $\Delta \in \mathbf{\Delta}$. That is, $\mathscr{D} := \{D = \mathrm{diag}(D_1, \ldots, D_s, d_1 I_{\beta_1}, \ldots, d_f I_{\beta_f}) : D = D^* > 0, D_i \in \mathbb{C}^{\alpha_i \times \alpha_i}, \sum_{i=1}^{s} \alpha_i + \sum_{i=1}^{f} \beta_i = r\}$. Then, note that at each fixed frequency ω,

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

with equality achieved when the conditions in Reference [20, Theorem 8.4] are satisfied. Then,

$$\begin{aligned} &\mu_{\mathbf{\Delta}_{\mathrm{TOT}}}\left[\begin{pmatrix} I_r & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathscr{F}_l(T(j\omega), Q(j\omega))\right] < 1 \quad \forall \omega \\ &\Leftarrow \quad \forall \omega \ \inf_{D_\omega \in \mathscr{D}} \bar{\sigma}\left[\begin{pmatrix} D_\omega & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathscr{F}_l(T(j\omega), Q(j\omega))\begin{pmatrix} D_\omega^{-1} & 0 \\ 0 & I_m \end{pmatrix}\right] < 1 \\ &\Leftrightarrow \quad \forall \omega \ \exists D_\omega \in \mathscr{D} : \mathscr{F}_l(T(j\omega), Q(j\omega))^* \begin{pmatrix} D_\omega^* D_\omega & 0 \\ 0 & W(j\omega)^* W(j\omega) \end{pmatrix} \mathscr{F}_l(T(j\omega), Q(j\omega)) \\ &\qquad\qquad < \begin{pmatrix} D_\omega^* D_\omega & 0 \\ 0 & I_m \end{pmatrix} \\ &\Leftrightarrow \quad \forall \omega \ \exists D_\omega \in \mathscr{D} : \begin{bmatrix} \begin{pmatrix} (D_\omega^* D_\omega)^{-1} & 0 \\ 0 & \mathrm{diag}(v_W(\omega)) \end{pmatrix} & \mathscr{F}_l(T(j\omega), Q(j\omega)) \\ * & \begin{pmatrix} (D_\omega^* D_\omega) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0 \end{aligned} \tag{10}$$

The last condition follows from a straightforward application of a standard Schur Complement Lemma [1] and by replacing $[W(j\omega)^* W(j\omega)]^{-1}$ with $\mathrm{diag}(v_W(\omega))$, where $v_W(\omega)$ is defined in Equation (6).

Now, since condition (10) is not simultaneously convex in $D_\omega \in \mathscr{D}$, $Q \in \mathscr{RH}_\infty$ and $v_W(\omega) \in \mathscr{V}$, we will hold $D_\omega \in \mathscr{D}$ fixed in an eventual optimization. In order to reduce conservativeness[‖] in such an optimization, $D_\omega \in \mathscr{D}$ will be chosen as the argument of the

[‖] That is, in order to ensure that condition (10) with a fixed $D_\omega \in \mathscr{D}$ is not too restrictive when compared to the original condition $\mu_{\mathbf{\Delta}_{\mathrm{TOT}}}[\,\cdot\,] < 1$.

following frequency-by-frequency minimization:

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

This minimization problem can be rewritten as

$$\begin{gathered} \text{For each } \omega \in \mathbb{R}, \\ \text{minimize } \gamma_\omega \\ \text{subject to } \exists\, D_\omega \in \mathscr{D} \text{ satisfying} \\ \bar{\sigma}\left[\begin{pmatrix} D_\omega & 0 \\ 0 & W(j\omega) \end{pmatrix} \mathscr{F}_l(T(j\omega), Q(j\omega)) \begin{pmatrix} D_\omega^{-1} & 0 \\ 0 & I_m \end{pmatrix}\right] < \gamma_\omega \end{gathered}$$

which after some algebra yields:

$$\begin{gathered} \text{For each } \omega \in \mathbb{R} \\ \text{minimize } \gamma_\omega^2 \\ \text{subject to } \exists\, (D_\omega^* D_\omega) \in \mathscr{D} \text{ with } (D_\omega^* D_\omega) > 0 \text{ satisfying} \\ \mathscr{F}_l(T(j\omega), Q(j\omega))^* \begin{pmatrix} (D_\omega^* D_\omega) & 0 \\ 0 & \operatorname{diag}(v_W(\omega))^{-1} \end{pmatrix} \mathscr{F}_l(T(j\omega), Q(j\omega)) < \gamma_\omega^2 \begin{pmatrix} (D_\omega^* D_\omega) & 0 \\ 0 & I_m \end{pmatrix} \end{gathered} \tag{11}$$

Minimization problem (11) is easy to solve as it is a quasi-convex generalized eigenvalue problem in the variables $(D_\omega^* D_\omega)$ and γ_ω^2.

4.4. The tractable optimization problem

Consider the following optimization problem, derived using the results of Sections 4.1–4.3:

$$\begin{gathered} \min_{v_W \in \mathscr{V}} \int_{\log_{10}\omega_L}^{\log_{10}\omega_H} v_\Upsilon(\omega)^T v_W(\omega)\, d(\log_{10}\omega) \\ \text{subject to} \\ \exists \text{ a } \check{Q} \in \mathbb{R}^{p\times(N+1)q} \quad \text{and } \forall\omega \text{ a } (D_\omega^* D_\omega) \in \mathscr{D} \\ \text{satisfying} \\ \begin{bmatrix} \begin{pmatrix} (D_\omega^* D_\omega)^{-1} & 0 \\ 0 & \operatorname{diag}(v_W(\omega)) \end{pmatrix} & \mathscr{F}_l(T(j\omega), \check{Q}B(j\omega)) \\ * & \begin{pmatrix} (D_\omega^* D_\omega) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0 \end{gathered} \tag{12}$$

It can be seen that the closed-loop properties guaranteed by optimization problem (5) are also guaranteed by the above optimization problem, as the latter has tighter (i.e. more restrictive) constraints. Recall also that $\mathscr{F}_l(T(j\omega), \check{Q}B(j\omega))$ appearing above is affine in $\check{Q}$, as can be seen from the definition of $T(s)$ in Theorem 1.

Unfortunately, optimization problem (12) still cannot be easily solved as the constraint of this problem is not simultaneously convex in $(D_\omega^* D_\omega)$, $\tilde{Q}$ and $v_W(\omega)$. If however $(D_\omega^* D_\omega)$ is held fixed, then optimization problem (12) reduces to a simple LMI optimization problem. In order to reduce conservativeness, this fixed $(D_\omega^* D_\omega)$ should be chosen as the solution of optimization problem (11), which itself requires $\tilde{Q}$ and $v_W(\omega)$ to be fixed. This co-dependence between these optimization problems indicates that some sort of iterative scheme must be used (see Section 6) to solve optimization problem (12).

5. A POINTWISE APPROXIMATION

In this section, optimization problem (12) will be approximated by a pointwise optimization problem, as (12) involves a search over a functional set with constraints holding for all $\omega \in \mathbb{R}$.

Whenever one constructs a pointwise in frequency approximation of an optimization problem through gridding, there is always the risk that the constraint of the optimization problem is violated between grid points or outside the frequency range where optimization takes place. Note that the constraint of optimization problem (12) guarantees $\mu_{\mathbf{\Delta}_{\mathrm{TOT}}}[\cdot] < 1$, and hence it is important for robust stability and performance that this constraint is not violated at any frequency. In order to increase our confidence that the constraint in (12) is not violated between/outside the grid points where optimization occurs, we shall use two grid frameworks, one for synthesis (i.e. used in optimizing the cost function) and one for analysis (i.e. used for checking the constraint), with the analysis grid having more points and spanning a larger frequency range than the synthesis grid.

To this end, let $\tilde{\omega}$ denote logarithmic frequency (i.e. $\tilde{\omega} := \log_{10}\omega$) and consider Figure 3 in which

- $g_s \in \mathbb{Z}_+$ (where $\mathbb{Z}_+$ denotes the strictly positive integers) is the desired number of synthesis grid points between $\log_{10}\omega_{\mathrm{L}}$ and $\log_{10}\omega_{\mathrm{H}}$. It must be emphasized that the synthesis grid must be chosen dense enough to ensure that all changes in the transfer function matrices $T_{ij}(s)$, defined in Theorem 1, are captured;

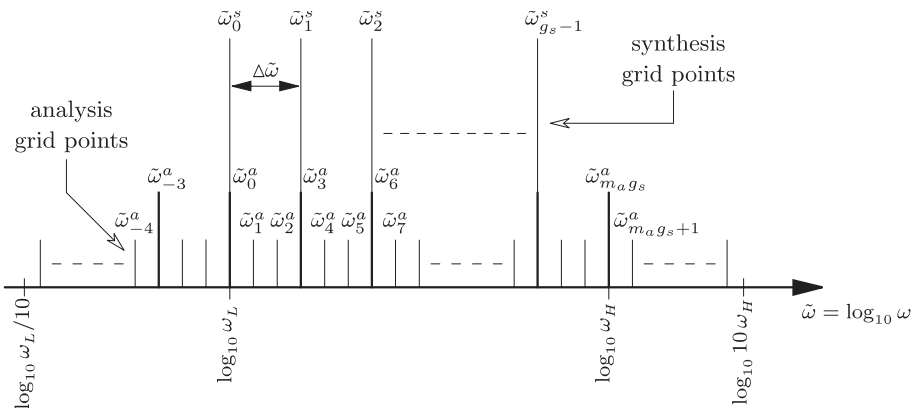

Figure 3. Synthesis and analysis grid points on a logarithmic scale.

- $m_a \in \mathbb{Z}_+$ 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

$$\Delta\tilde{\omega} := \frac{\log_{10}\omega_H - \log_{10}\omega_L}{g_s} = \log_{10}\left[\left(\frac{\omega_H}{\omega_L}\right)^{1/g_s}\right] \tag{13}$$

Then, the synthesis grid points are given by

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

$$\Rightarrow \quad \omega_k^s := 10^{\tilde{\omega}_k^s} = \omega_L\left(\frac{\omega_H}{\omega_L}\right)^{k/g_s} \quad \text{for } k \in \mathbb{Z},\ 0 \leqslant k \leqslant (g_s - 1) \tag{14}$$

Note that there is no synthesis grid point at the frequency ω_H. The reason for this will become apparent when the cost function of optimization problem (12) is approximated by a pointwise in frequency cost function. As an indication, we do not need this final grid point because we are going to perform a first-order approximation of a definite integral.

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

Definition 3
Let the set of indices for the analysis grid points be defined by

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

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

Then, the analysis grid points are given by

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

$$\Rightarrow \omega_k^a := 10^{\tilde{\omega}_k^a} = \omega_L\left(\frac{\omega_H}{\omega_L}\right)^{k/(m_a g_s)} \quad \text{for } k \in \mathbf{\Omega} \tag{15}$$

Now, a first-order approximation of the cost function in optimization problem (12) is given by

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

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

$$v_{\Upsilon,k} := \begin{cases} v_\Upsilon(\omega^s_{k/m_a}) & \text{when } 0 \leqslant \lfloor k/m_a \rfloor = k/m_a \leqslant (g_s - 1) \\ [\varepsilon\ \ \varepsilon\ \ \cdots\ \ \varepsilon]^T & \text{otherwise} \end{cases} \tag{16}$$

where $\varepsilon \in \mathbb{R}_+$ is some very small number.[¶] 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) and is a very small cost elsewhere.

Hence, a pointwise in frequency approximation of optimization problem (12) is given by

$$\begin{gathered}
\min_{v_{W,k}\in\mathbb{R}^n \ \forall k\in\mathbf{\Omega}} \ \sum_{k\in\mathbf{\Omega}} (v_{\Upsilon,k})^{\mathrm{T}}(v_{W,k}) \\
\text{subject to} \\
\exists \text{ a } \check{\boldsymbol{Q}} \in \mathbb{R}^{p\times(N+1)q} \quad \text{and} \quad \forall k \in \mathbf{\Omega} \text{ a } \Theta_k \in \mathscr{D} \\
\text{satisfying} \\
\begin{bmatrix} \begin{pmatrix} \Theta_k^{-1} & 0 \\ 0 & \operatorname{diag}(v_{W,k}) \end{pmatrix} & \mathscr{F}_l(T(j\omega_k^{\mathrm{a}}), \check{\boldsymbol{Q}}B(j\omega_k^{\mathrm{a}})) \\ * & \begin{pmatrix} \Theta_k & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0
\end{gathered} \tag{17}$$

On writing the above optimization problem, $(\Delta\tilde{\omega})$ has been removed from the cost function as it is a constant and hence does not affect the arguments of the minimization. Furthermore, observe that in the above optimization 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 $\omega = \omega_k^{\mathrm{a}}$. Note also that there is no need to restrict the vectors $v_{W,k}$ to belong to $\mathbb{R}_+^n$ in the arguments of minimization (17), as positivity of each element in $v_{W,k}$ is implicitly guaranteed by the constraint of this optimization problem.

A small $\varepsilon \in \mathbb{R}_+$ is used in the definition of $v_{\Upsilon,k}$ (see Equation (16)) so as to ensure that the vectors $v_{W,k}$ resulting from optimization problem (17) are 'reasonably smooth' as k changes in $\mathbf{\Omega}$. This is because the vectors $v_{W,k}$ at grid points corresponding to the analysis grid but not to the synthesis grid (see Figure 3) still appear in the cost function, although their contribution towards the reduction of the cost is small.

Furthermore, as pointed out at the end of Section 4, $\Theta_k \in \mathscr{D}$ must be held fixed in optimization problem (17) for convexity reasons and these fixed values are obtained via optimization problem (11). Consequently, a pointwise in frequency approximation of optimization problem (11) can be obtained in a similar way and is given by

$$\begin{gathered}
\text{For each } k \in \mathbf{\Omega} \\
\text{minimize } \gamma^2_{\omega_k^{\mathrm{a}}} \\
\text{subject to } \exists\, \Theta_k \in \mathscr{D} \text{ with } \Theta_k > 0 \text{ satisfying} \\
\mathscr{F}_l(T(j\omega_k^{\mathrm{a}}), \check{\boldsymbol{Q}}B(j\omega_k^{\mathrm{a}}))^* \begin{pmatrix} \Theta_k & 0 \\ 0 & \operatorname{diag}(v_{W,k})^{-1} \end{pmatrix} \mathscr{F}_l(T(j\omega_k^{\mathrm{a}}), \check{\boldsymbol{Q}}B(j\omega_k^{\mathrm{a}})) < \gamma^2_{\omega_k^{\mathrm{a}}} \begin{pmatrix} \Theta_k & 0 \\ 0 & I_m \end{pmatrix}
\end{gathered} \tag{18}$$

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 $\omega = \omega_k^{\mathrm{a}}$. This is a quasi-convex generalized eigenvalue problem which is easily solved using LMI routines.

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

6. SOLUTION ALGORITHM

Optimization problem (17) cannot be directly solved, as this problem is not simultaneously convex in Θ_k, $\check{Q}$ and $v_{W,k}$. If, however, the variables Θ_k are held fixed, then optimization problem (17) reduces to a simple LMI optimization problem. In order to reduce conservativeness, Θ_k should be chosen as the solution of optimization problem (18) at each grid point $k \in \mathbf{\Omega}$. Now, since these two optimization problems are interdependent, an iterative solution algorithm is proposed.

Inputs to the algorithm:

- Generalized plant $G(s)$ such that the associated uncertainty in Figure 1 is normalized;
- Directionality transfer function matrix $\Upsilon(s) \in \mathbf{\Upsilon}^{\mathrm{TF}}$.

The solution algorithm:

1. Compute $T(s)$ using the definition in Theorem 1, $B(s)$ using the definition in Equation (9) and $v_{\Upsilon,k}$ using both Equations (7) and (16).

   Then, for every $k \in \mathbf{\Omega}$, select a $\Theta^{\bigstar}_{k,0} \in \mathscr{D}$ with $\Theta^{\bigstar}_{k,0} > 0$ such that the constraint of the optimization problem given in Step 3 below admits some feasible solution, provided that such a solution exists. These $\Theta^{\bigstar}_{k,0} \in \mathscr{D}$ 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 7. However, it should be pointed out that setting $\Theta^{\bigstar}_{k,0} = \varphi I_r \ \forall k \in \mathbf{\Omega}$ (for a sufficiently small $\varphi > 0$) is usually good enough.**

   Set $i = 0$, where i denotes the iteration number, and $\eta^{\bigstar}_0 = \infty$.
2. Increment i by 1.
3. Solve the following convex optimization problem:

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

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

$$\begin{bmatrix} \begin{pmatrix} (\Theta^{\bigstar}_{k,i-1})^{-1} & 0 \\ 0 & \mathrm{diag}(v_{W,k}) \end{pmatrix} & \mathscr{F}_l(T(j\omega^{\mathrm{a}}_k), \check{Q} B(j\omega^{\mathrm{a}}_k)) \\ * & \begin{pmatrix} (\Theta^{\bigstar}_{k,i-1}) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0 \quad \forall k \in \mathbf{\Omega}$$

   Here $T(j\omega^{\mathrm{a}}_k)$, $B(j\omega^{\mathrm{a}}_k)$ and $v_{\Upsilon,k}$ are given and $\Theta^{\bigstar}_{k,i-1}$ is the value of Θ_k obtained in the previous iteration. Recall that $\mathscr{F}_l(T(j\omega^{\mathrm{a}}_k), \check{Q} B(j\omega^{\mathrm{a}}_k))$ is affine in $\check{Q}$ and hence the above optimization problem is a simple LMI problem.

   Let the value of this minimum cost be denoted by $\eta^{\bigstar}_i$ and let the values of $\check{Q}$ and $\mathrm{diag}(v_{W,k})$ at each $k \in \mathbf{\Omega}$ that achieve this minimum be denoted by $\check{Q}^{\bigstar}_i$ and $\Pi^{\bigstar}_{k,i}$ for each $k \in \mathbf{\Omega}$, respectively.

**This is evident from the proof of Lemma 2.

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

$$\text{Minimize } \gamma^2_{\omega^{\mathrm{a}}_k}$$

$$\text{such that } \exists\, \Theta_k \in \mathscr{D} \text{ with } \Theta_k > 0 \text{ satisfying}$$

$$\mathscr{F}_l(T(j\omega^{\mathrm{a}}_k), \check{Q}^{\bigstar}_i B(j\omega^{\mathrm{a}}_k))^* \begin{pmatrix} \Theta_k & 0 \\ 0 & (\Pi^{\bigstar}_{k,i})^{-1} \end{pmatrix} \mathscr{F}_l(T(j\omega^{\mathrm{a}}_k), \check{Q}^{\bigstar}_i B(j\omega^{\mathrm{a}}_k)) < \gamma^2_{\omega^{\mathrm{a}}_k} \begin{pmatrix} \Theta_k & 0 \\ 0 & I_m \end{pmatrix}$$

Here $T(j\omega^{\mathrm{a}}_k)$ and $B(j\omega^{\mathrm{a}}_k)$ are given, and $\check{Q}^{\bigstar}_i$ and $\Pi^{\bigstar}_{k,i}$ are the values of the $\check{Q}$ and $\mathrm{diag}(v_{W,k})$ obtained in Step 3. Note that the above optimization problem can be easily solved using LMI routines.

For each $k \in \mathbf{\Omega}$, let the square root of the above minimum cost be denoted by $\gamma^{\bigstar}_{\omega^{\mathrm{a}}_k,i}$ and let the value of Θ_k that achieves this minimum be denoted by $\Theta^{\bigstar}_{k,i}$.

5. Evaluate $(\eta^{\bigstar}_{i-1} - \eta^{\bigstar}_i)$. 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 $K^{\bigstar}_i(s)$ corresponding to the above $\check{Q}^{\bigstar}_i$ using Theorem 1 with $Q(s) = \check{Q}^{\bigstar}_i B(s)$ and then model reduce $K^{\bigstar}_i(s)$ if necessary.

Outputs from the algorithm (after i iterations):

- The element-by-element magnitude of the largest performance weights obtained by the algorithm in $(\Pi^{\bigstar}_{k,i})^{-1/2}\ \forall k \in \mathbf{\Omega}$,
- The final D-scales used by the algorithm in $(\Theta^{\bigstar}_{k,i})^{1/2}\ \forall k \in \mathbf{\Omega}$,
- The controller $K^{\bigstar}_i(s) \in \mathscr{K}^{\mathrm{TF}}_G$ that achieves robust performance with respect to these weights,
- The value of the minimum cost $\eta^{\bigstar}_i$ and the final upper bound of $\mu_{\mathbf{\Delta}_{\mathrm{TOT}}}[\cdot]$ in $\gamma^{\bigstar}_{\omega^{\mathrm{a}}_k,i}\ \forall k \in \mathbf{\Omega}$.

Note that at each iteration i, Step 3 ensures that $\max_{k\in\mathbf{\Omega}} \gamma^{\bigstar}_{\omega^{\mathrm{a}}_k,\,i} \leqslant 1$ and Step 4 minimizes $\gamma^{\bigstar}_{\omega^{\mathrm{a}}_k,\,i}$ at each fixed $k \in \mathbf{\Omega}$. This immediately guarantees robust performance for all uncertainty $\Delta \in \mathbf{B\Delta}^{\mathrm{TF}}$.

Moreover, as the iterations proceed (i.e. as i increases), the minimum cost $\eta^{\bigstar}_i$ is monotonically non-increasing. This is because the solutions $\check{Q}^{\bigstar}_i$ and $\Pi^{\bigstar}_{k,i}\ \forall k \in \mathbf{\Omega}$ obtained in Step 3 at the ith iteration always satisfy the constraint of the same optimization problem in Step 3 at the $(i+1)$th iteration. This can be seen by observing that at Step 4, $\check{Q}^{\bigstar}_i$ and $\Pi^{\bigstar}_{k,i}\ \forall k \in \mathbf{\Omega}$ are held fixed while $\gamma^2_{\omega^{\mathrm{a}}_k}$ (which is always $\leqslant 1$) is minimized at each fixed $k \in \mathbf{\Omega}$ over Θ_k to give the new $\Theta^{\bigstar}_{k,i}$ for each $k \in \mathbf{\Omega}$. This new $\Theta^{\bigstar}_{k,i}$ will then be used in Step 3 at the $(i+1)$th iteration. The fact that $\eta^{\bigstar}_i$ 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' [22]. However, iterative algorithms as the one presented above cannot be guaranteed to converge to the *global minimum*. Only monotonic properties can be proved.

7. FINDING A FEASIBLE INITIAL STARTING POINT

The algorithm proposed in Section 6 requires in its first step a feasible initial starting point that initializes the algorithm. That is, it requires for all $k \in \mathbf{\Omega}$ some $\Theta^{\bigstar}_{k,0} \in \mathscr{D}$ with $\Theta^{\bigstar}_{k,0} > 0$ such that the constraint of the optimization problem given in Step 3 admits some feasible solution. The aim of this section is to find such a $\Theta^{\bigstar}_{k,0}$. However, before doing this, the following lemma will be stated which considerably simplifies the problem.

Lemma 2
Let $T(s)$ be defined as in Theorem 1 and $B(s)$ be defined as in Equation (9). Then, given a $\check{Q} \in \mathbb{R}^{p\times(N+1)q}$, the following two statements are equivalent for all $k \in \mathbf{\Omega}$:

(i) there exists a $\Theta_{k,0}^{\bigstar} \in \mathscr{D}$ and a $v_{W,k} \in \mathbb{R}^n$ satisfying

$$\begin{bmatrix} \begin{pmatrix} (\Theta_{k,0}^{\bigstar})^{-1} & 0 \\ 0 & \mathrm{diag}(v_{W,k}) \end{pmatrix} & \mathscr{F}_l(T(j\omega_k^{\mathrm{a}}), \check{Q}B(j\omega_k^{\mathrm{a}})) \\ * & \begin{pmatrix} (\Theta_{k,0}^{\bigstar}) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0$$

(ii) there exists a $\Theta_{k,0}^{\bigstar} \in \mathscr{D}$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^{\bigstar})^{-1} & T_{11}(j\omega_k^{\mathrm{a}}) + T_{13}(j\omega_k^{\mathrm{a}})\check{Q}B(j\omega_k^{\mathrm{a}})T_{31}(j\omega_k^{\mathrm{a}}) \\ * & (\Theta_{k,0}^{\bigstar}) \end{bmatrix} > 0$$

Proof
See Appendix A 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 6. Towards this end, note that the inequality in condition (ii) of Lemma 2 is equivalent to

$$\bar{\sigma}[(\Theta_{k,0}^{\bigstar})^{1/2}[T_{11}(j\omega_k^{\mathrm{a}}) + T_{13}(j\omega_k^{\mathrm{a}})\check{Q}B(j\omega_k^{\mathrm{a}})T_{31}(j\omega_k^{\mathrm{a}})](\Theta_{k,0}^{\bigstar})^{-1/2}] < 1 \quad \text{with } \Theta_{k,0}^{\bigstar} > 0 \quad \forall k \in \mathbf{\Omega}$$

Consequently, one way of finding a feasible initial starting point for the algorithm of Section 6 is to minimize

$$\bar{\sigma}\left[(\Theta_{k,0}^{\bigstar})^{1/2}[T_{11}(j\omega_k^{\mathrm{a}}) + T_{13}(j\omega_k^{\mathrm{a}})\check{Q}B(j\omega_k^{\mathrm{a}})T_{31}(j\omega_k^{\mathrm{a}})](\Theta_{k,0}^{\bigstar})^{-1/2}\right] \tag{19}$$

over both $\Theta_{k,0}^{\bigstar} \in \mathscr{D}$ with $\Theta_{k,0}^{\bigstar} > 0$ for all $k \in \mathbf{\Omega}$ and $\check{Q} \in \mathbb{R}^{p\times(N+1)q}$, and stop this minimization when expression (19) is less than unity. This minimization problem is however not simultaneously convex in both variables and hence expression (19) has to be alternately minimized over each variable in succession. The two steps in the resulting iterative minimization are obtained by noting that *at each* $k \in \mathbf{\Omega}$

$$\bar{\sigma}[(\Theta_{k,0}^{\bigstar})^{1/2}[T_{11}(j\omega_k^{\mathrm{a}}) + T_{13}(j\omega_k^{\mathrm{a}})\check{Q}B(j\omega_k^{\mathrm{a}})T_{31}(j\omega_k^{\mathrm{a}})](\Theta_{k,0}^{\bigstar})^{-1/2}] < \varsigma_{\omega_k^{\mathrm{a}}}$$

$$\Updownarrow$$

$$[T_{11}(j\omega_k^{\mathrm{a}}) + T_{13}(j\omega_k^{\mathrm{a}})\check{Q}B(j\omega_k^{\mathrm{a}})T_{31}(j\omega_k^{\mathrm{a}})]^{*}(\Theta_{k,0}^{\bigstar})[T_{11}(j\omega_k^{\mathrm{a}}) + T_{13}(j\omega_k^{\mathrm{a}})\check{Q}B(j\omega_k^{\mathrm{a}})T_{31}(j\omega_k^{\mathrm{a}})] < \varsigma_{\omega_k^{\mathrm{a}}}^2(\Theta_{k,0}^{\bigstar})$$

and that

$$\bar{\sigma}\left[(\Theta_{k,0}^{\bigstar})^{1/2}[T_{11}(j\omega_k^{\mathrm{a}})+T_{13}(j\omega_k^{\mathrm{a}})\check{\mathbf{Q}}B(j\omega_k^{\mathrm{a}})T_{31}(j\omega_k^{\mathrm{a}})](\Theta_{k,0}^{\bigstar})^{-1/2}\right]<\varsigma \quad \forall k\in\mathbf{\Omega}$$

$$\Updownarrow$$

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

$$\Updownarrow$$

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

The former inequality can be used for minimization of $\varsigma^2_{\omega_k^{\mathrm{a}}}$ over $(\Theta_{k,0}^{\bigstar})$, whereas the latter inequality can be used for minimization of ς over $\check{\mathbf{Q}}$. This iterative minimization may be stopped when either $\varsigma\leqslant 1$ or $\varsigma^2_{\omega_k^{\mathrm{a}}}\leqslant 1 \ \forall k\in\mathbf{\Omega}$.

8. NUMERICAL EXAMPLE

The algorithm proposed in Section 6 will now be illustrated by a numerical example. The example considered in this section is taken from the LMI Control Toolbox MATLAB manual [23]. There is also a demonstration in MATLAB for this example and this is called by typing 'radardem' at the MATLAB prompt.

Let us now define the problem setting. Figure 4 shows a simplified mechanical model of a radar antenna. The stiffness k accounts for flexibilities in the coupling between the motor (with inertia J_{m}) and the antenna (with inertia J_{a}). The corresponding nominal transfer function model from the motor torque τ to the angular velocity $\dot{\theta}_{\mathrm{a}}$ of the antenna is given by

$$P_{\mathrm{o}}=\frac{30\,000}{(s+0.02)(s^2+0.99s+30\,030)}$$

It is evident that this plant possesses a highly resonant mode at around 173 rad/s with damping ratio of approximately 0.003. Our goal is to control $\dot{\theta}_{\mathrm{a}}$ through the torque τ.

The feedback interconnection given in Figure 5 involves a tracking loop from r to y, a 'two degrees of freedom' controller $K=[K_1 \ \ K_2]$, plant output disturbances d and sensor noise n that must be attenuated, and a ball of multiplicative dynamical stable uncertainty $\Delta(s)$ that surrounds the nominal plant model P_{o} to account for neglected high frequency dynamics and flexibilities. Since disturbances are mainly present at low frequencies in this example, the disturbance input weight W_{d} is specified as

$$W_{\mathrm{d}}=\frac{10}{(s+100)}$$

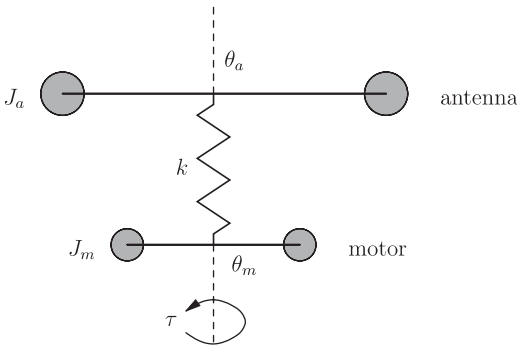

Figure 4. Second-order model of a radar antenna.

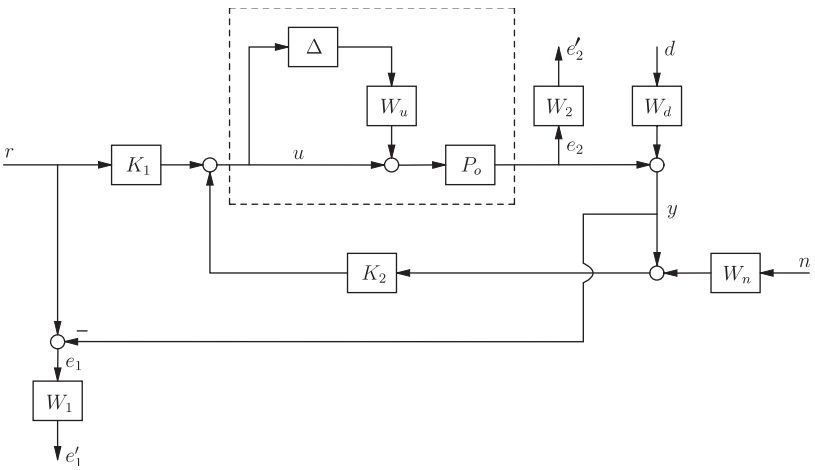

Figure 5. Control structure for the radar antenna.

to allow disturbances d upto 100 rad/s to enter the feedback interconnection with normalized size 0.1. Similarly, since sensor noise is mainly present at high frequencies in this example, the sensor noise input weight W_{n} is specified as

$$W_{\mathrm{n}} = \frac{0.1s}{(s+100)}$$

to allow sensor noise n beyond 100 rad/s to enter the feedback interconnection with normalized size 0.1. As stated above, the true physical plant is uncertain but is known to belong to the set $\{P_{\mathrm{o}}(1 + W_{\mathrm{u}}\Delta) : \Delta \in \mathscr{RH}_\infty,\ \|\Delta\|_\infty \leqslant 1\}$ parametrized by Δ (see dashed box). The uncertainty weight W_{u} represents any '*a priori*' knowledge about the frequency dependent size of the uncertainty. In this example, W_{u} is specified as

$$W_{\mathrm{u}} = \frac{(s+1)}{(s+1000)}$$

to allow the magnitude of the actual plant to differ from that of the nominal plant by as much as 0.1% in the low-frequency region (say, below 1 rad/s) and by as much as 100% in the high-frequency region (say, beyond 1000 rad/s).

It is required that signal y tracks input signal r as well as it possibly can upto 10 rad/s and that the feedback interconnection attenuates as much as possible all exogenous excitations beyond 10 rad/s as seen at the plant output signal e_2. This statement can be reformulated more precisely as follows: we wish to maximize performance weight W_1 as much as possible in the frequency region upto 10 rad/s and we wish to maximize performance weight W_2 as much as possible in the frequency region beyond 10 rad/s, limited of course by the requirement that there must exist controllers K_1 and K_2 that achieve the robust performance level demanded by these maximized performance weights.

As argued in the preceeding sections of this paper (see especially Section 2), these maximization requirements are captured by the directionality frequency response functions $\upsilon_1(j\omega)$ (associated to W_1) and $\upsilon_2(j\omega)$ (associated to W_2) given in Figure 6. These directionalities state that the optimization problem should maximize W_1 (resp. W_2) in the low-frequency (resp. high-frequency) region and that it should not bother too much about maximizing W_1 (resp. W_2) 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 optimization. Only the relative sizes between the different curves and the shape of each curve across frequency are relevant. 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 optimization problem should value the maximization of W_1 at low-frequency equally as the maximization of W_2 at high-frequency. At around 10 rad/s, the solid curve and the dashed curve are also equal. This again means that around this frequency, the maximization of W_1 is equally important as the maximization of W_2. However, due to the algebraic constraints resulting from the feedback interconnection, it is not possible to make both W_1 and W_2 large at this same frequency. Thus, in this case, the optimization problem needs to sort out by itself how much maximization is possible in each of

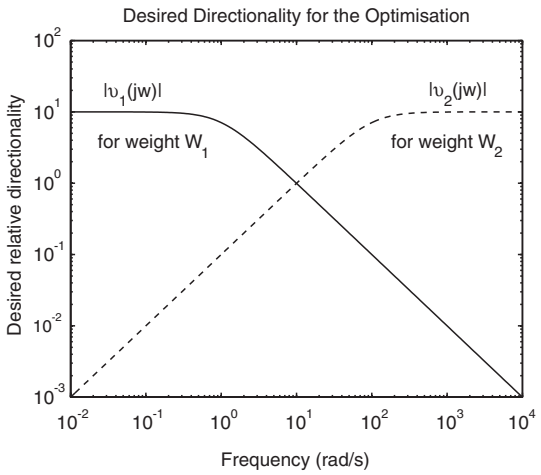

Figure 6. Desired directionality for the optimization.

these weights. Furthermore, since the magnitude of the directionalities at 10 rad/s is about a decade less than the magnitude of the directionalities at low and high-frequency, maximization of W_1 and W_2 in this mid-frequency region should be considered less important than maximization of W_1 and W_2 at low and high-frequency.

The frequency range $[\omega_L, \omega_H]$ selected for maximizing the performance weights was $[10^{-1}, 10^3]$ as this corresponds to two decades of frequency below and above the desired closed-loop bandwidth of 10 rad/s (see Figure 6), whereas that used for gridding the constraint was a decade larger on each side (i.e. $[10^{-2}, 10^4]$). We choose 100 grid points for synthesis (i.e. $g_s = 100$) and a multiplier $m_a = 2$ for the denser analysis gridding. Furthermore, the parameters T and N of the Laguerre basis for the Youla parameter were chosen as $T = 0.1$ (corresponding to a sampling frequency of 10 Hz which is 6 times more than the desired closed-loop bandwidth of 10 rad/s) and $N = 6$. Larger values of N and/or smaller values of T did not yield any improvement and hence were considered adequate.

In order to illustrate the behaviour of the algorithm as iterations proceed, consider Figure 7 which gives plots of intermediate results after the 1st and 2nd iterations. Figure 7(a) shows inverse magnitude pointwise plots of the performance weights obtained by the algorithm after the 1st iteration. These extremely liberal weights together with some internally stabilizing controller give the computed upper bound of μ depicted in Figure 7(b). This upper bound can be seen to be less than unity at all frequencies, as it must satisfy the optimization's constraint. The 'space' between this computed upper bound of μ and unity will then be exploited by the algorithm at the 2nd iteration to synthesize better pointwise performance weights. In other words, at the 2nd iteration, the algorithm will push $|W_1(j\omega)|^{-1}$ and $|W_2(j\omega)|^{-1}$ down at appropriate frequencies (according to the directionalities of Figure 6) while ensuring that the upper bound of μ never exceeds unity. The new pointwise performance weights resulting from this minimization are shown in Figure 7(c). It is clear that these new performance weights guarantee a higher level of robust performance (they are in fact much tighter than the crude inverse performance weights of Figure 7(a)). The upper bound of μ resulting from this change in performance weights is the solid line in Figure 7(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 minimized at each frequency over the D-scale, keeping the new performance weights and the new internally stabilizing controller fixed. The result of this D-scale minimization is seen as the dashed curve in Figure 7(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.

Five iterations were found to be sufficient for convergence of the algorithm[††] and the total iteration time taken was approximately 6 min (on a standard desktop PC). Figure 8 shows plots of the final pointwise performance weights $|W_1(j\omega)|^{-1}$ and $|W_2(j\omega)|^{-1}$ obtained by the algorithm. Controllers K_1 and K_2 that achieve robust performance with respect to these maximized performance weights were computed at the end of the iterations (as detailed in Step 6 of the algorithm) and their bode plots are given in Figure 9. Note that $[K_1 \; K_2]$ has McMillan degree of 11 and hence no model reduction was considered necessary for this example. These controllers (which are essentially 'lead' compensators) together with the inverse performance weights of Figure 8 gave a flat curve across frequency for the computed upper bound of μ that

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

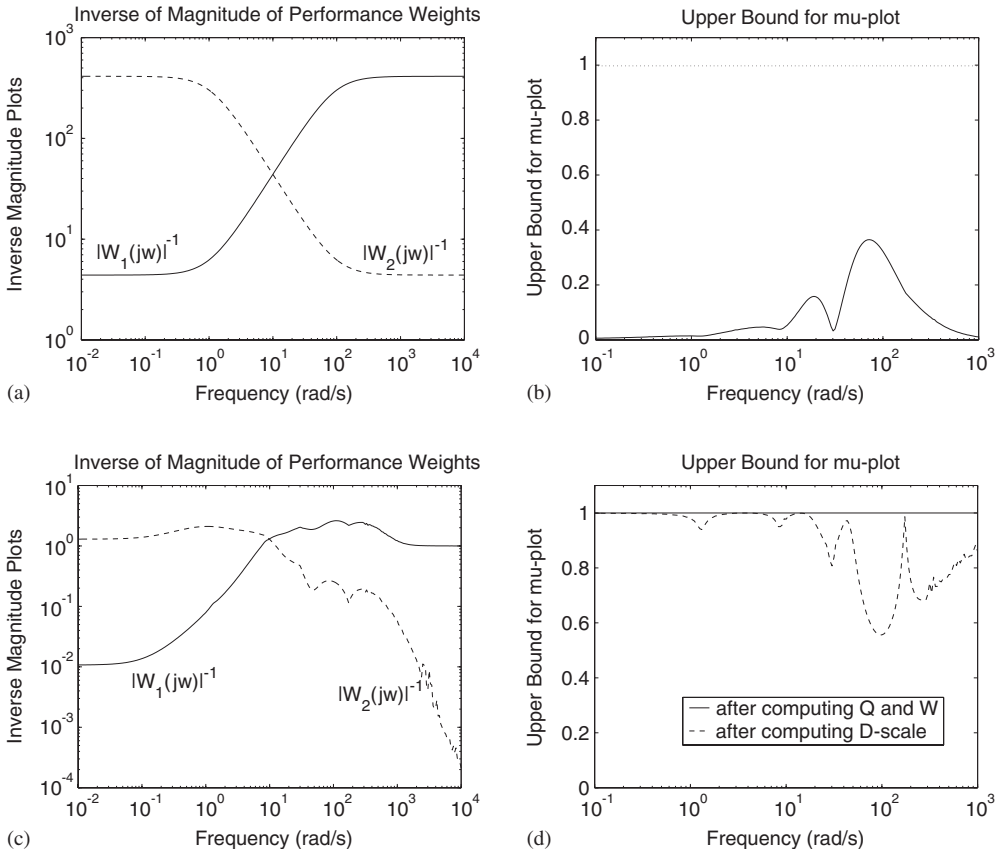

Figure 7. Intermediate results after the 1st and 2nd iterations: (a) performance weights after 1st iteration; (b) μ upper bound after 1st iteration; (c) performance weights after 2nd iteration; and (d) μ upper bounds after 2nd iteration.

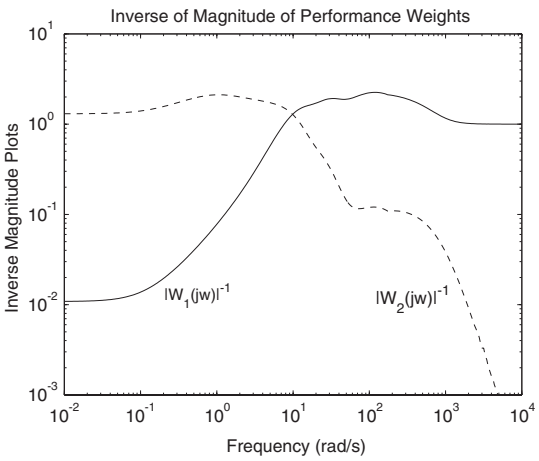

Figure 8. Final pointwise inverse magnitude performance weights.

was slightly less than unity. The associated final pointwise D-scale used by the algorithm is given in Figure 10.

For completeness sake, we also give in Figure 11 magnitude plots of the nominal transfer functions from r, d and n to e_1 and e_2 (which are the unweighted errors that we wanted attenuated—see Figure 5). Figure 11(a) shows that y tracks r well (to 99% accuracy) in the low frequency region and that the feedback interconnection attenuates the high frequencies of r at the plant output e_2. Also, from Figure 11(b), disturbances d are effectively rejected from the signal $e_1 = r - y$ and also high frequency components of d, if any, are not propagated through the loop as seen from the plant output signal e_2. Similarly, from Figure 11(c), sensor noise n is

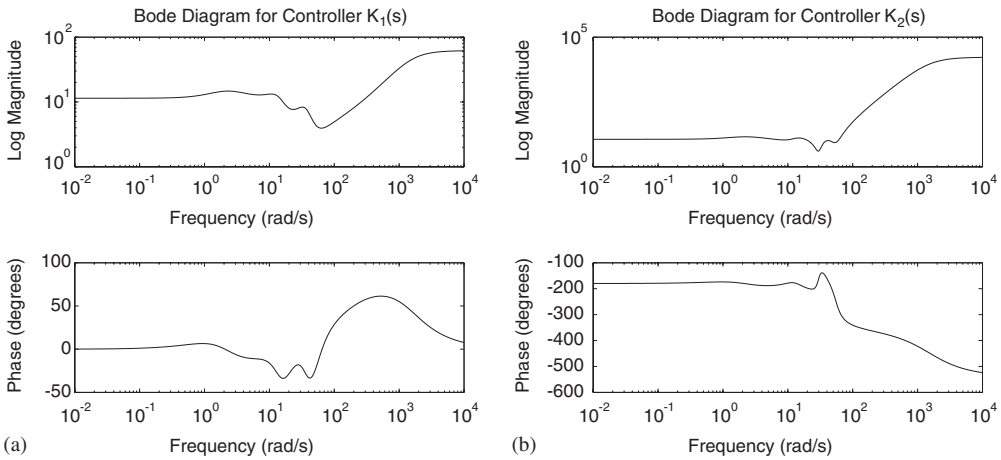

Figure 9. Bode plots for controllers $[K_1 \;\; K_2] = K(s) = \mathscr{F}_l(J(s), \check{Q}B(s))$: (a) controller $K_1(s)$; and (b) controller $K_2(s)$.

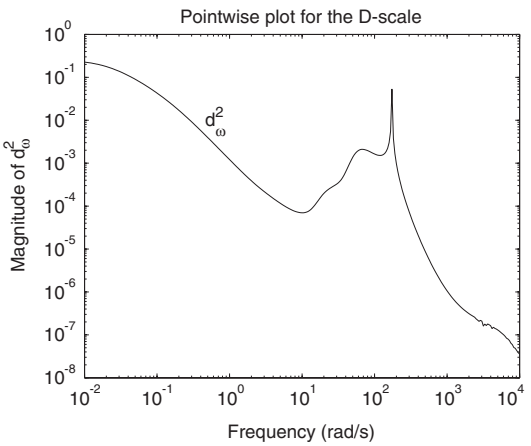

Figure 10. Final pointwise D-scale.

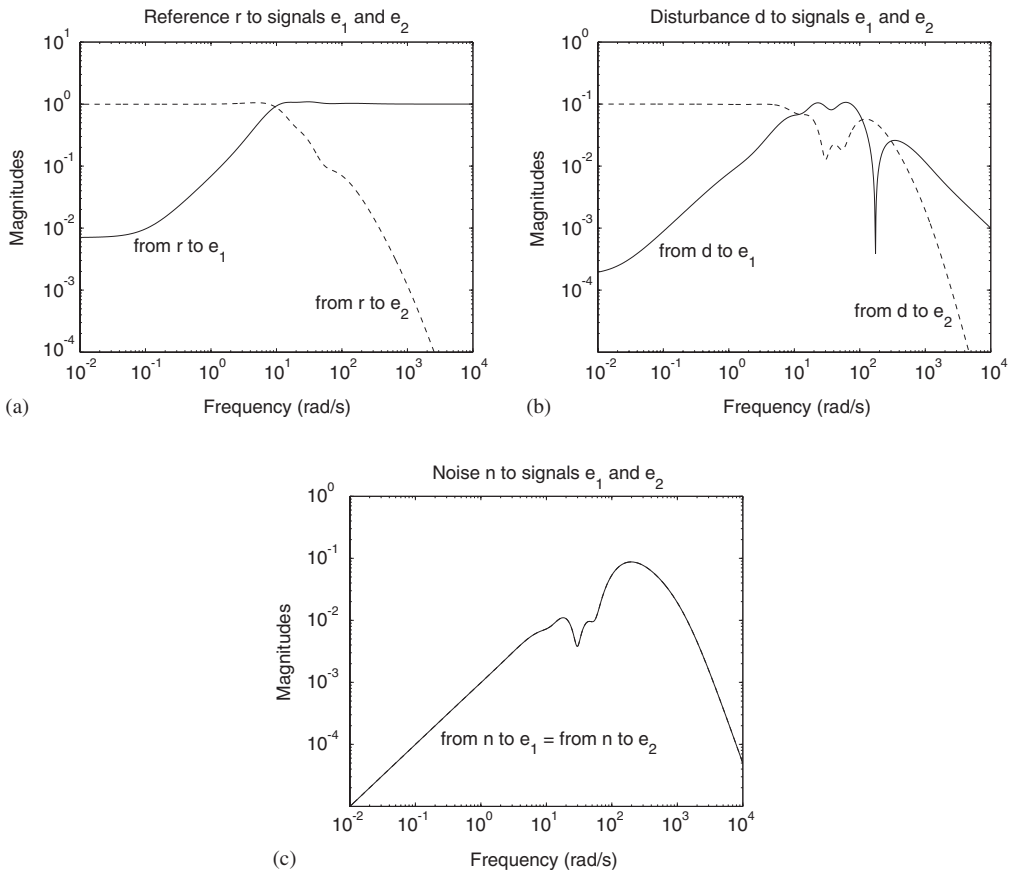

Figure 11. Nominal magnitude plots for frequency responses from r, d and n to e_1 and e_2: (a) from r to e_1 and e_2; (b) from d to e_1 and e_2; and (c) from n to e_1 and e_2.

rejected from both signals e_1 and e_2 (curves are exactly on top of each other), and noise has only an effect in the mid-frequencies around 200 rad/s.

9. CONCLUSIONS

The problem of maximizing performance weights in the frequency range of interest, subject to the existence of an internally stabilizing controller that guarantees robust performance with respect to these maximized weights, was posed as an optimization problem in Section 2. However, this optimization problem was difficult to compute. Thus, a computationally tractable optimization problem was formulated in Section 4 which had tighter and hence more restrictive constraints than the original problem. A pointwise approximation of this optimization problem was then given in Section 5. Reduction of conservativeness in this latter optimization problem

required D-scales to be computed from yet another optimization problem, that was unfortunately interlinked with the chosen optimization problem. Consequently, an iterative procedure was proposed in Section 6 to take care of this. Both these optimization problems were written as LMI problems.

Some important advantages of this algorithm over existing methods for robust performance μ-synthesis design (e.g. D–K iterations) are given here: (a) Performance weights, which maximize some cost function that captures the desired closed-loop performance, are synthesized simultaneously with an internally stabilizing controller by one systematic algorithm; (b) The controller synthesized by this algorithm immediately guarantees robust performance with respect to these maximized weights. Also, incompatible performance weights can never be obtained by this algorithm, as the performance weights obtained must be feasible to the optimization's constraint; (c) This optimization usually gives a final closed-loop μ-curve that is flat across frequency and very close to unity, reflecting the fact that the robust performance level has been maximized; and (d) 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 and gives the designer a quick indication of what robust performance level is attainable.

The proposed algorithm does, however, suffer from some disadvantages: (a) The Laguerre-like parametrization $Q(s) = \check{Q}B(s)$ usually causes high-order controllers to be synthesized (which will have to be model reduced for a practical design); (b) 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 (even though the algorithm proposed here allows for a much denser analysis grid than the synthesis grid); and (c) Maximization of performance weights occurs only in the frequency range $[\omega_L, \omega_H]$. Fortunately, a variant of the optimization problem proposed in this paper that admits a state-space solution has been also investigated by the author in References [21, 24].

In summary, this article addresses the μ-synthesis robust performance problem from a conceptually different point of view to what is nowadays standard practice.

APPENDIX A: PROOF OF LEMMA 2

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

(a) There exists a $\Theta^{\star}_{k,0} \in \mathscr{D}$ and a $v_{W,k} \in \mathbb{R}^n$ satisfying

$$\begin{bmatrix} \begin{pmatrix} (\Theta^{\star}_{k,0})^{-1} & 0 \\ 0 & \mathrm{diag}(v_{W,k}) \end{pmatrix} & \mathscr{F}_l(T(j\omega_k^{\mathrm{a}}), \check{Q}B(j\omega_k^{\mathrm{a}})) \\ * & \begin{pmatrix} (\Theta^{\star}_{k,0}) & 0 \\ 0 & I_m \end{pmatrix} \end{bmatrix} > 0$$

(b) There exists a $\Theta_{k,0}^{\bigstar} \in \mathscr{D}$ and a $v_{W,k} \in \mathbb{R}_+^n$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^{\bigstar})^{-1} & H_{11}(j\omega_k^{\mathrm{a}}) & H_{12}(j\omega_k^{\mathrm{a}}) \\ * & (\Theta_{k,0}^{\bigstar}) & 0 \\ * & * & I_m \end{bmatrix} > \begin{bmatrix} 0 \\ H_{21}(j\omega_k^{\mathrm{a}})^* \\ H_{22}(j\omega_k^{\mathrm{a}})^* \end{bmatrix} \mathrm{diag}(v_{W,k})^{-1} [0 \;\; H_{21}(j\omega_k^{\mathrm{a}}) \;\; H_{22}(j\omega_k^{\mathrm{a}})] \tag{A1}$$

where

$$\begin{bmatrix} H_{11}(j\omega_k^{\mathrm{a}}) & H_{12}(j\omega_k^{\mathrm{a}}) \\ H_{21}(j\omega_k^{\mathrm{a}}) & H_{22}(j\omega_k^{\mathrm{a}}) \end{bmatrix} := \begin{bmatrix} T_{11}(j\omega_k^{\mathrm{a}}) & T_{12}(j\omega_k^{\mathrm{a}}) \\ T_{21}(j\omega_k^{\mathrm{a}}) & T_{22}(j\omega_k^{\mathrm{a}}) \end{bmatrix} + \begin{bmatrix} T_{13}(j\omega_k^{\mathrm{a}}) \\ T_{23}(j\omega_k^{\mathrm{a}}) \end{bmatrix} \times \check{Q}B(j\omega_k^{\mathrm{a}})[T_{31}(j\omega_k^{\mathrm{a}}) \;\; T_{32}(j\omega_k^{\mathrm{a}})]$$
$$= \mathscr{F}_l(T(j\omega_k^{\mathrm{a}}), \check{Q}B(j\omega_k^{\mathrm{a}}))$$

The equivalence (a) $\Leftrightarrow$ (b) follows from a standard Schur Complement Lemma [1] applied around the term 'diag($v_{W,k}$)' appearing in (a).

(c) There exists a $\Theta_{k,0}^{\bigstar} \in \mathscr{D}$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^{\bigstar})^{-1} & H_{11}(j\omega_k^{\mathrm{a}}) & H_{12}(j\omega_k^{\mathrm{a}}) \\ * & (\Theta_{k,0}^{\bigstar}) & 0 \\ * & * & I_m \end{bmatrix} > 0$$

Note that (b) $\Rightarrow$ (c) trivially follows from inequality (A1), 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 right side of inequality (A1) sufficiently small to also imply that (b) is true.

(d) There exists a $\rho \in \mathbb{R}_+$ and a $\Theta_{k,0}^{\bigstar} \in \mathscr{D}$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^{\bigstar})^{-1} & H_{11}(j\omega_k^{\mathrm{a}}) & H_{12}(j\omega_k^{\mathrm{a}}) \\ * & (\Theta_{k,0}^{\bigstar}) & 0 \\ * & * & \rho I_m \end{bmatrix} > 0 \tag{A2}$$

The equivalence (c) $\Leftrightarrow$ (d) can be seen through the application of the congruence transformation $\mathrm{diag}(\sqrt{\rho} I_r, 1/\sqrt{\rho} I_r, 1/\sqrt{\rho} I_m)$ on inequality (A2).

(e) There exists a $\rho \in \mathbb{R}_+$ and a $\Theta_{k,0}^{\bigstar} \in \mathscr{D}$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^{\bigstar})^{-1} & H_{11}(j\omega_k^{\mathrm{a}}) \\ * & (\Theta_{k,0}^{\bigstar}) \end{bmatrix} > \frac{1}{\rho} \begin{bmatrix} H_{12}(j\omega_k^{\mathrm{a}}) \\ 0 \end{bmatrix} [H_{12}(j\omega_k^{\mathrm{a}})^* \quad 0] \tag{A3}$$

The equivalence (d) $\Leftrightarrow$ (e) follows from a standard Schur Complement Lemma [1] applied around the (3,3)-element of inequality (A2).

(f) There exists a $\Theta_{k,0}^{\bigstar} \in \mathscr{D}$ satisfying

$$\begin{bmatrix} (\Theta_{k,0}^{\bigstar})^{-1} & H_{11}(j\omega_k^{\mathrm{a}}) \\ * & (\Theta_{k,0}^{\bigstar}) \end{bmatrix} > 0$$

Note that (e) $\Rightarrow$ (f) trivially follows from inequality (A3), whereas (e) $\Leftarrow$ (f) is because given that (f) is true there always exists a sufficiently large $\rho \in \mathbb{R}_+$ that makes the right side of inequality (A3) sufficiently small to also imply that (e) is true. This last condition is in fact Statement (ii) in the Lemma (since $H_{11}(j\omega_k^a) = T_{11}(j\omega_k^a) + T_{13}(j\omega_k^a)\check{Q}B(j\omega_k^a)T_{31}(j\omega_k^a)$) and hence the proof is complete. □

ACKNOWLEDGEMENTS

A part of this work was supported by an ARC Discovery-Projects Grant (DP0342683) and National ICT Australia Ltd. National ICT Australia Ltd. is funded through the Australian Government's *Backing Australia's Ability* initiative, in part through the Australian Research Council. Preliminary parts of this paper have also been presented at the 38th IEEE Conference on Decision and Control, Phoenix, AZ, U.S.A. (see Reference [25]).

REFERENCES

1. Boyd S, El Ghaoui L, Feron E, Balakrishnan V. *Linear Matrix Inequalities in System and Control Theory*. SIAM Studies in Applied Mathematics, vol. 15. Society for Industrial and Applied Mathematics: Philadelphia, PA, 1994.
2. El Ghaoui L, Niculescu S-I. Advances in linear matrix inequality methods in control. *Advances in Design and Control*. SIAM: Philadelphia, PA, 2000.
3. Skelton RE, Iwasaki T, Grigoriadis K. *A Unified Algebraic Approach to Linear Control Design*. Taylor and Francis: London, 1998.
4. Vandenberghe L, Boyd S. Semidefinite programming. *SIAM Review* 1996; **38**(1):49–95.
5. Jovik I, Lennartson B. On the choice of criteria and weighting functions of an $\mathscr{H}_2/\mathscr{H}_\infty$-design problem. *IFAC Symposia Series—Proceedings of the 13th Triennial World Congress*, San Francisco, CA, U.S.A., 1996; 373–378.
6. Lanzon A. Weight selection in robust control: an optimisation approach. *Ph.D. Thesis*, University of Cambridge, U.K., October 2000. Download from http://rsise.anu.edu.au/~alanzon.
7. Stoughton RM. Formulation of an improved set of weighting functions for $\mathscr{H}_\infty$ control of flexible beam-like systems. *Proceedings of the American Control Conference*, San Diego, CA, U.S.A., May 1990; 1745–1751.
8. Green M, Limebeer DJN. *Linear Robust Control*. Prentice-Hall: Englewood Cliffs, NJ, 1995.
9. Skogestad S, Postlethwaite I. *Multivariable Feedback Control: Analysis and Design*. Wiley: New York, 1998.
10. Zhou K, Doyle JC, Glover K. *Robust and Optimal Control*. Prentice-Hall: Englewood Cliffs, NJ, 1996.
11. Doyle JC. Structured uncertainty in control systems design. *Proceedings of the 24th IEEE Conference on Decision and Control*, Ft. Lauderdale, FL, U.S.A., 1985; 260–265.
12. Doyle JC, Packard A, Zhou K. Review of LFTs, LMIs and μ. *Proceedings of the 30th IEEE Conference on Decision and Control*, U.K., 1991; 1227–1232.
13. Lin J-L, Postlethwaite I, Gu D-W. μ–K iteration: a new algorithm for μ synthesis. *Automatica* 1993; **29**(1):219–224.
14. Chang C-Y, Yang C-D, Lin CE. A new iteration scheme for robust performance problem: E–K iterations. *Proceedings of the American Control Conference*, Baltimore, MD, U.S.A., 1994; 2809–2813.
15. Rotea MA, Iwasaki T. An alternative to the D–K iteration? *Proceedings of the American Control Conference*, Baltimore, MD, U.S.A., 1994; 53–57.
16. Fan MKH, Tits AL. A measure of worst-case $\mathscr{H}_\infty$ performance and of largest acceptable uncertainty. *Systems and Control Letters* 1992; **18**(6):409–421.
17. Lanzon A. Simultaneous synthesis of weights and controllers in $\mathscr{H}_\infty$ loop-shaping. *Proceedings of the 40th IEEE Conference on Decision and Control*, vol. 1, Orlando, FL, U.S.A., December 2001; 670–675.
18. Lanzon A, Bombois X, Anderson BDO. On weight adjustments in $\mathscr{H}_\infty$ control design. *Proceedings of the European Control Conference*, Cambridge, U.K., September 2003.
19. Helmersson A. Methods for Robust gain scheduling. Linköping Studies in Science and Technology, *Dissertations*, vol. 6. Linköping University: Sweden, 1995.
20. Packard A, Doyle J. The complex structured singular value. *Automatica* 1993; **29**(1):71–109.
21. Lanzon A, Cantoni MW. On the formulation and solution of robust performance problems. *Automatica* 2003; **39**(10):1707–1720.

22. Haggarty R. *Fundamentals of Mathematical Analysis* (2nd edn). Addison-Wesley: Reading, MA, 1994.
23. Gahinet P, Nemirovski A, Laub AJ, Chilali M. *LMI Control Toolbox*. The MathWorks, Inc.: Natick, MA, 1995 (for use with MATLAB).
24. Lanzon A, Cantoni MW. A state-space algorithm for the simultaneous optimisation of performance weights and controllers in μ-synthesis. *Proceedings of the 39th IEEE Conference on Decision and Control*, vol. 1, Sydney, Australia, December 2000; 611–616.
25. Lanzon A, Richards RJ. A frequency domain optimisation algorithm for simultaneous design of performance weights and controllers in μ-synthesis. *Proceedings of the 38th IEEE Conference on Decision and Control*, vol. 5, Phoenix, AZ, U.S.A., December 1999; 4523–4528.

