INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL, VOL. 6, 805–823 (1996)

LOOPSHAPING FOR ROBUST PERFORMANCE

RICHARD D. BRAATZ

Department of Chemical Engineering, University of Illinois at Urbana-Champaign, 600 South Mathews Avenue, Box C–3, Urbana, 61801–3792.

MANFRED MORARI

Chemical Engineering, California Institute of Technology, Pasadena, CA 91125, U.S.A.

AND

SIGURD SKOGESTAD

Department of Chemical Engineering, University of Trondheim, N-7034 Trondheim, Norway

SUMMARY

Robust performance is said to be achieved if the performance specifications are met for all plants in a specified set. Classical loopshaping was developed decades ago to design for robust performance for single-loop systems with simple uncertainty and performance specifications. Specifications are often not so simple—multiple real parameter variations are not handled by classical loopshaping, for example. Also, it is important for multivariable systems that uncertainty may be present at different locations, for example, actuator uncertainty is located at the input of the plant whereas sensor uncertainty is located at the output of the plant. In this work classical loopshaping is extended to multiple parametric and unmodelled dynamic uncertainty descriptions, to multiple performance specifications, and to the design of decentralized controllers. The authors refer to this more general loopshaping technique as *robust loopshaping*. Robust loopshaping is applied to a coupled mass–spring problem studied by numerous researchers.

KEY WORDS loopshaping; robust performance; robust control; multivariable stability margin

1. INTRODUCTION

Loopshaping involves directly specifying a transfer function that parametrizes the controller based on magnitude bounds on the transfer function. These bounds are either necessary conditions or sufficient conditions so that the closed-loop system satisfies desired stability and performance specifications. Examples of transfer functions that parametrize the controller include the sensitivity $S=(I+PK)^{-1}$, the complementary sensitivity $H=PK(I+PK)^{-1}$, and the open-loop transfer function $L=PK$. The controller K is then calculated from the specified transfer function.

Advantages of loopshaping are: (1) the controller complexity (e.g. low-order, PID) can be directly specified, (2) decentralized controllers can be designed, and (3) the properties of interest to the engineer are often directly in terms of the designed loopshape.

This paper was recommended for publication by editor A. Sideris

CCC 1049–8923/96/080805–19

Received 17 March 1994
Revised 5 April 1995

The technique of loopshaping was introduced by Bode[3] for the design of feedback amplifiers. Doyle *et al.*[12] review classical loopshaping, where the system is single-input single-output (SISO), the uncertainty can be represented as a single complex Δ-block, and the sole performance specification is a bound on the peak of the closed-loop sensitivity. Uncertainty and performance specifications are often not so simple—control problems may involve multiple performance specifications, and uncertainty may be more conveniently described as real parameter variations. Also, it is important for multivariable systems that uncertainty may be present at different locations, for example, actuator uncertainty is located at the input of the plant whereas sensor uncertainty is located at the output of the plant.

In this work classical loopshaping is extended to multiple parametric and unmodelled dynamic uncertainty descriptions, to multiple performance specifications, and to the design of decentralized controllers. We refer to this more general loopshaping technique as *robust loopshaping*.

The basic ideas behind robust loopshaping are as follows. First, the uncertainty and performance specifications are written in terms of μ. Second, the uncertain system is parametrized in terms of a transfer function to be loopshaped. Third, necessary bounds and sufficient bounds on the Bode magnitude plot of this transfer function are calculated in terms of the achievement of robust performance. One of these bounds (the sufficient upper bound) was derived by Skogestad and Morari,[25,26] and in this paper we extend these results by deriving a new sufficient lower bound, as well as two new necessary bounds (upper and lower). These bounds are used in the subsequent loopshaping design. We show that robust loopshaping gives simple analytical expressions in the simple single-loop case. We then design a controller for a coupled mass–spring problem studied by numerous researchers.

2. BACKGROUND

Below we summarize robustness analysis via the structured singular value for analysing the stability and performance of uncertain systems, and provide formulas to parametrize uncertain systems in terms of transfer functions of interest, T. These parametrizations will be required in the calculation of the robust loopshaping bounds.

2.1. Robustness analysis

In practice the model is an inaccurate representation of the true process. To account for this plant/model mismatch, the true process is represented by a *set of plants*. The term *robust* is used to indicate that some property (e.g. stability or performance) holds for a set of possible plants as defined by the uncertainty description.

The uncertainty is modelled as norm-bounded perturbations (Δ_i) on the nominal system. Through weights each perturbation is normalized to be of size one

$$\|\Delta_i\|_\infty \equiv \sup_\omega \bar{\sigma}(\Delta_i) \leq 1 \tag{1}$$

where Δ_i is complex for representing unmodelled dynamics (treating Δ_i as complex is equivalent to treating it as a stable proper rational transfer function),[14,22] and real for representing parametric uncertainty. The perturbations, which may occur at different locations in the system, are collected in the block-diagonal matrix Δ_U (the U denotes uncertainty)

$$\Delta_U = \text{diag}\{\Delta_i\} \tag{2}$$

and the system is arranged to match the left block diagram in Figure 1. The interconnection matrix M in Figure 1 is determined by the nominal model (P), the size and nature of the

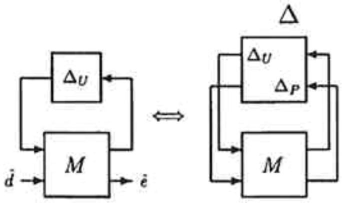

Figure 1. Robust performance and the $M-\Delta$ block structure

uncertainty, the performance specifications, and the controller (K). Without loss of generality we assume that each Δ_i is square.[18]

The *structured singular value* μ (also referred to as the *robustness margin*)[10,13,20,23,24] provides a necessary and sufficient test for whether a particular controller stabilizes the set of plants given by the uncertainty description, that is, whether the controller provides *robust stability*. Frequency-domain performance specifications can also be treated as complex uncertainty (the block Δ_P between *normalized disturbances* $\hat{d}$ and *normalized controlled variables*, ê, in Figure 1) and these blocks are included in the calculation of a larger μ problem which tests for *robust performance*. The necessary and sufficient test in both cases is

$$\text{system robustness} \Leftrightarrow \mu < 1 \text{ for all frequencies} \tag{3}$$

where μ is a function of M and the structure of the uncertainty Δ. For example, the test for robust stability is $\mu_{\Delta_U}(M_{11}) < 1$ and the test for robust performance is $\mu_\Delta(M) < 1$. For details see Reference 20.

In what follows we will need the definition of *linear fractional transformations* (LFTs). If we partition M to be compatible with Δ_U in Figure 1, then the transfer function between disturbances, $\hat{d}$, and controlled variables, $\hat{e}$, is given by the *linear fractional transformation* (LFT)

$$F_u(M, \Delta_U) = M_{22} + M_{21}\Delta_U(I - M_{11}\Delta_U)^{-1}M_{12} \tag{4}$$

The LFT $F_u(M, \Delta_U)$ is well-defined if and only if the inverse of $I - M_{11}\Delta_U$ exists. The subscript ‘u’ on F_u is used to denote that the *upper* loop of M is closed by Δ_U. If the *lower* loop had been closed instead, then the transfer function between inputs and outputs would be the LFT $F_l(M, \Delta_U) = M_{11} + M_{12}T(I - M_{22}\Delta_U)^{-1}M_{21}$.

It is a key idea that μ is a *general* analysis tool for determining robust stability and robust performance. Any system with uncertainty adequately modelled as in (1) and with frequency-domain performance specifications can be put into $M-\Delta$ form and robust stability and robust performance can be tested. Off-the-shelf software[2,9] calculates the M and Δ, given the transfer functions describing the system components, the performance specifications, and the location of the uncertainty. Upper and lower bounds for μ can be calculated in polynomial-time (the upper bound via transformation to a linear matrix inequality, and the lower bound via a power iteration) and are usually close.[2,29] The pitfalls in attempting to calculate μ exactly in the presence of real parametric uncertainties is discussed by Braatz *et al.*[8]

Parametrize uncertain system in terms of T

Here we show how to parametrize the uncertain system in terms of the transfer function T to be loopshaped.[25] For example, T could be the complementary sensitivity $H = PK(I + PK)^{-1}$, the sensitivity $S = (I + PK)^{-1}$, the open-loop transfer function $L = PK$, or the controller K.

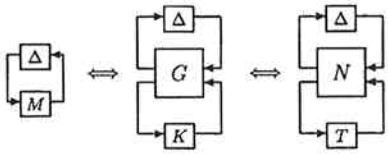

Figure 2. Equivalent representations of system M with perturbation Δ

Mathematically, we need to find an LFT in terms of T which describes M (see Figure 2). In many cases, this is done by inspection. When this is not possible, the equations given below (which are derived in Reference 5 but can be confirmed directly via the definition of the LFT) can be used.

To get an LFT in terms of T, begin with the interconnection structure in terms of G and K. The generalized plant G is determined by the nominal model, the location and magnitude of the uncertainties, and the performance specifications. The generalized plant G is found directly by rearranging the system's block diagram (the subroutine *sysic* does this in μ-tools).[2] We calculate N for $T=H$ (denoted as N^H) directly from G (so that $M=F_1(N^H,H)$):

$$N^H=\begin{bmatrix} G_{11} & G_{12}P^{-1} \\ G_{21} & 0 \end{bmatrix} \tag{5}$$

For $T=S$, L, and K, respectively, we have

$$N^S=\begin{bmatrix} G_{11}+G_{12}P^{-1}G_{21} & -G_{12}P^{-1} \\ G_{21} & 0 \end{bmatrix} \tag{6}$$

$$N^L=\begin{bmatrix} G_{11} & G_{12}P^{-1} \\ G_{21} & G_{22}P^{-1} \end{bmatrix} \tag{7}$$

$$N^K=G \tag{8}$$

A simple program can be written that calculates N^N, N^S, N^L, and N^K, given the transfer functions describing the system components, the performance specifications, and the location of the uncertainty blocks Δ_i.

3. ROBUST LOOPSHAPING BOUNDS

Controllers which satisfy robust performance can be designed via robust loopshaping. To perform robust loopshaping, the robust performance conditions are expressed as norm bounds on the transfer function T to be loopshaped.

Consider a system in $M-\Delta$ form as shown in Figure 2. The interconnection structures in Figure 2 are equivalent. The closed-loop transfer matrix M is written as a linear fractional transformation of the transfer function of interest, namely T. In the following let Δ_T represent an arbitrary transfer function with the same structure as T. We define the set of norm-bounded perturbations

$$\Delta_T^c \equiv \{\bar{\sigma}(\Delta_T) \leq c \mid \Delta_T \text{ has the same structure as } T\} \tag{9}$$

and its near-complement

$$\bar{\Delta}_T^c \equiv \{\bar{\sigma}(\Delta_T) \geq c \mid \Delta_T \text{ has the same structure as } T\} \tag{10}$$

The following theorems provide sufficient bounds and necessary bounds on the magnitude of the transfer function T for robust performance to be achieved. The basic idea of the theorems is to treat T as an unknown perturbation (Δ_T) and to use μ theory to either provide guarantees that the worst-case Δ_T satisfies the original robustness condition $\mu(M)<k$ (usually $k=1$), or guarantees that the best-case Δ_T cannot satisfy the robustness condition. More detailed existence conditions and proofs of all theorems in this paper are described in the Appendix. Remarks follow the theorems.

Theorem 3.1 (Sufficient upper bound for robustness)[25,26]

Let $M=F_l(N,T)$ and k be a given constant. Define

$$e(c)=\sup_{\Delta_T\in\Delta_T^c}\mu_\Delta(F_l(N,\Delta_T)) \tag{11}$$

Assume there exists c_T^{su} such that $e(c_T^{\mathrm{su}})=k$. Then $\mu_\Delta(M)<k$ if

$$\bar{\sigma}(T)<c_T^{\mathrm{su}} \tag{12}$$

Theorem 3.2 (Sufficient lower bound for robustness)

Let $M=F_l(N,T)$ and k be a given constant. Define

$$f(c)=\sup_{\Delta_T\in\bar{\Delta}_T^c}\mu_\Delta(F_l(N,\Delta_T)) \tag{13}$$

Assume there exists c_T^{sl} such that $f(c_T^{\mathrm{sl}})=k$. Then $\mu_\Delta(M)<k$ if

$$\bar{\sigma}(T)>c_T^{\mathrm{sl}}. \tag{14}$$

Theorem 3.3 (Necessary upper bound for robustness)

Let $M=F_l(N,T)$ and k be a given constant. Define

$$g(c)=\inf_{\Delta_T\in\bar{\Delta}_T^c}\mu_\Delta(F_l(N,\Delta_T)) \tag{15}$$

Assume there exists c_T^{nu} such that $g(c_T^{\mathrm{nu}})=k$. Then $\mu_\Delta(M)<k$ only if

$$\bar{\sigma}(T)<c_T^{\mathrm{nu}} \tag{16}$$

Theorem 3.4 (Necessary lower bound for robustness)

Let $M=F_l(N,T)$ and k be a given constant. Define

$$h(c)=\inf_{\Delta_T\in\Delta_T^c}\mu_\Delta(F_l(N,\Delta_T)) \tag{17}$$

Assume there exists c_T^{nl} such that $h(c_T^{\mathrm{nl}})=k$. Then $\mu_\Delta(M)<k$ only if

$$\bar{\sigma}(T)>c_T^{\mathrm{nl}} \tag{18}$$

Remark 3.5

Many parametrizations T exist for the controller K, for example, K can be parameterized by the sensitivity S, the complementary sensitivity H, the open-loop transfer function $L=PK$, or just the controller K. Controllers can also be designed via loopshaping the IMC filter F[12] or the IMC filter time constant λ.[17] Parametrizations for decentralized controllers are based on defining similar open-loop or closed-loop transfer functions as are used in the SISO case, except with the diagonal part of the plant replacing the plant P (details are provided in Reference 5).

Remark 3.6

When the necessary upper and the sufficient upper bounds are very close to each other, we have essentially a *necessary and sufficient upper* bound for robust performance in terms of $\bar{\sigma}(T)$. A similar statement holds for the necessary lower and sufficient lower bounds.

Remark 3.7

The bounds given by each theorem are the tightest bounds possible. For example, if we have a T_1 with $\bar{\sigma}(T_1)$ larger than c_T^{su} defined by Theorem 3.1, then there exists a T_2 with $\bar{\sigma}(T_2) = \bar{\sigma}(T_1)$ where T_2 does not meet robust performance.

Remark 3.8

The smallest upper and largest lower bounds are obtained when Δ_T is a repeated scalar block. This makes necessary bounds for robust performance more restrictive, and sufficient bounds easier to satisfy. This latter property encourages the use of repeated scalar Δ_T (i.e. assuming all loops are identical) when designing decentralized controllers for robust performance via loopshaping. When designing controllers to have failure tolerance properties, it can be useful to allow Δ_T to consist of independent 1×1 blocks when calculating sufficient bounds for robust stability. For further details on the design of decentralized controllers to satisfy failure/fault tolerance specifications, see Reference 5.

Remark 3.9

The norm bounds on different Ts can be combined over different frequency ranges. For example, for $T_1 = S$ and $T_2 = H$, robust performance is achieved if either of the conditions $\bar{\sigma}(S(j\omega)) < c_S^{\mathrm{su}}$ *or* $\bar{\sigma}(H(j\omega)) < c_H^{\mathrm{su}}$ is met for each ω.

Remark 3.10

Conditions for the existence of c_T which solves the equalities in the above theorems are given with sketches of the proofs in the Appendix. Braatz[5] describes in detail the interpretation of each of these existence conditions and when they are expected to hold when using c_T for loopshaping design of controllers.

Remark 3.11

Methods for calculating the bounds c_T given in the above theorems are provided by Braatz[5] and for brevity are not given in detail here. In brief, it can be shown via the Main Loop Theorem[20] that the sufficient upper bound c_T^{su} can be solved via a single μ calculation. As was discussed above, Δ_T is typically a repeated scalar block in the loopshaping design of SISO and decentralized controllers. In this case it can be shown via the Inverse LFT Theorem[22] that the sufficient lower bound c_T^{sl} can be solved via a single μ calculation.

The calculation of the necessary bounds is more difficult because these optimizations cannot be reparametrized as μ calculations. When Δ_T is repeated scalar (i.e. $\Delta_T = tI$), the optimizations (15) and (17) are over two parameters—the *gain* and *phase* of the repeated parameter t. By exploiting the monotonicity of $g(c)$ and $h(c)$, both of the two-parameter optimizations can be replaced by a *single* one-parameter optimization $m(c)$ over only the *phase* of t. The necessary lower bound and upper bound are given by the smallest and largest values of c which satisfy $m(c) = k$, respectively. One-parameter optimizations can be solved via line search. Another approach that is useful in the SISO case is to replace μ with its upper bound and convert the optimizations described by (15) and (17) into two coupled sets of linear matrix inequalities (LMIs)—one set of LMIs is over the D and G scales in the upper bound, the other set of LMIs is over the Youla parameter q. This conversion is standard in robust controller synthesis.[4] Each set of LMIs forms a convex optimization. The joint optimization is not convex, but the ad hoc approach of iteratively solving these optimizations has been found to work well in practice.

It is instructive to compare this iteration to that used in the most popular method for full robust controller synthesis, commonly referred to as μ-synthesis.[2,9] When calculating loopshaping bounds, the iteration is performed independently at each frequency, with the computations involved with each set of LMIs being well conditioned. In contrast, the μ-synthesis design procedure requires fitting stable transfer functions $D(s)$ and $G(s)$ to values for $D(j\omega)$ and $G(j\omega)$ at a discrete set of frequencies.[2] It is well-known that current robust control software[2,9] has difficulty in fitting these transfer functions accurately or even consistently. In fact, improving the fitting of these transfer functions is an area of active research.[1]

Remark 3.12

It is straightforward to derive alternative bounds in terms of $\underline{\sigma}(T) = 1/\bar{\sigma}(T^{-1})$ by parametrizing in terms of T^{-1} and using one of the inverse LFT lemmas.[5] We will not explore this further here.

4. CONTROLLER DESIGN VIA LOOPSHAPING

In robust loopshaping design the nominal transfer functions are specified directly based on necessary bounds and sufficient bounds for robust performance.

The control engineer has a choice whether to loopshape closed-loop (e.g. S or H) or open-loop (e.g. K or L) transfer functions. The advantage of loopshaping closed-loop transfer functions is that the properties of interest to the engineer are given by the closed-loop transfer functions. For example, the sensitivity specifies the capacity of the closed-loop system to reject disturbances at the output of the plant whereas the insensitivity of the output to measurement noise is specified by the complementary sensitivity. A simple form is usually chosen for S and H, and the controller is calculated via $K = P^{-1}HS^{-1}$.

The advantage of robust loopshaping open-loop transfer functions is that the controller complexity (e.g. PID, or low-order) is directly specified. This is difficult to do using other robust controller design methods. For example, the DK-iteration method[11] tends to give controllers of very high order, though the order can be somewhat reduced using model reduction.[15]

In either case separate conditions are used to guarantee nominal stability. These conditions depend on whether open-loop or closed-loop transfer functions are being loopshaped. When designing an SISO controller via loopshaping closed-loop transfer functions, nominal stability is guaranteed by pre-specifying parametrizations for S and H which are stable and satisfy the interpolation conditions[12] (see Section 5 for an example)

$$H(z_i) = 0 \text{ and } S(z_i) = 1 \text{ for all closed right half plane zeros } z_i \tag{19}$$

$$S(p_i) = 0 \text{ and } H(p_i) = 1 \text{ for all closed right half plane poles } p_i \tag{20}$$

The interpolation conditions are equivalent to the condition that the right half plane poles and zeros of the plant cannot be cancelled by the controller. These conditions are easy to satisfy when there are few right half plane poles and zeros; when there are more then the Internal Model Control (IMC) method can be used to stabilize the system, and the filter can be designed via loopshaping (for details see References 19 and 12). When loopshaping open-loop transfer functions, the phase of L is directly chosen to provide nominal stability (see Section 6 for an example).

A general advantage of *robust loopshaping* over other robust controller design methods is that *decentralized* controllers can be designed. Controllers can also be designed to meet specified gain and phase margins, multiple performance specifications, and failure and fault tolerance specifications.

5. ROBUST LOOPSHAPING AND CLASSICAL LOOPSHAPING

Classical loopshaping was developed decades ago by Bode[3] to design for robust performance for single-loop systems. Classical loopshaping bounds have been derived[12] for the case where the uncertainty can be represented as a single complex Δ-block, and the sole performance specification is an upper bound on the closed-loop sensitivity. Below we compare the robust loopshaping bounds with classical loopshaping bounds.

Formulation in terms of μ

Assume that we are interested in disturbance attenuation, then the performance condition is to keep the norm of the sensitivity function $\bar{\sigma}(S) = |S|$ small. If we let the frequency-dependent performance bound be $1/|wp|$, then robust performance is satisfied if $\bar{\sigma}(S) < 1/|wp|$ for all plants in the uncertainty description. Let the set of possible plants be given in terms of multiplicative uncertainty of magnitude $|wp|$ (see Figure 3). From inspection of Figure 3 we see that robust performance is satisfied if and only if $\mu_\Delta(M) < 1$ for all frequencies, where

$$M = \begin{bmatrix} w_O H & w_O H \\ w_P S & w_P S \end{bmatrix} \qquad \Delta = \begin{bmatrix} \Delta_O & \\ & \Delta_P \end{bmatrix} \tag{21}$$

and Δ_O and Δ_P are complex scalars representing the multiplicative uncertainty and performance, respectively. The generalized plant, G, is found from inspection to be

$$G = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} = \begin{bmatrix} 0 & 0 & w_O P \\ w_P & w_P & -w_P P \\ 1 & 1 & -P \end{bmatrix} \tag{22}$$

With G and Δ given, N^T can be calculated using (5)–(7), and the robust loopshaping theorems can be applied. For this problem the robust loopshaping bounds can be calculated analytically. For brevity we do not show the details of the derivation here (see Reference 5), but will only present the bounds and compare them with the classical loopshaping bounds. We will

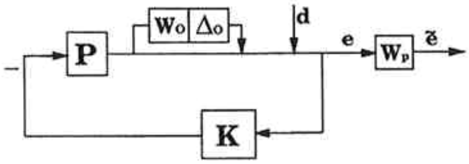

Figure 3. The plant with output uncertainty Δ_O of magnitude w_O. Robust performance is satisfied if $\bar{\sigma}(w_P(I+\hat{P}K)^{-1}) \leq 1$ for all Δ_O with $\|\Delta_O\|_\infty \leq 1$

assume that either $|w_O|$ or $|w_P|$ is less than one for each frequency, since if this condition is not met, then robust performance cannot be met for any controller.

Robust loopshaping bounds

(1) The sufficient upper bounds for $|H|$, $|S|$, and $|L|$ (bounds for K are immediately given by the bounds for L, since $|L| = |P| \cdot |K|$) are

$$\mu_\Delta(M) < 1 \Leftarrow |H| < c_H^{\text{su}} = \frac{1-|w_P|}{|w_O|+|w_P|} \quad \text{(for frequencies where } |w_P| < 1) \tag{23}$$

$$\mu_\Delta(M) < 1 \Leftarrow |S| < c_S^{\text{su}} = \frac{1-|w_O|}{|w_O|+|w_P|} \quad \text{(for frequencies where } |w_O| < 1) \tag{24}$$

$$\mu_\Delta(M) < 1 \Leftarrow |L| < c_L^{\text{su}} = \frac{1-|w_P|}{1+|w_O|} \quad \text{(for frequencies where } |w_P| < 1) \tag{25}$$

(2) The sufficient lower bound for $|L|$ is

$$\mu_\Delta(M) < 1 \Leftarrow |L| > c_L^{\text{sl}} = \frac{1+|w_P|}{1-|w_O|} \quad \text{(for frequencies where } |w_O| < 1) \tag{26}$$

The sufficient lower bound does not exist for closed loop transfer functions.

(3) The necessary upper bounds are

$$\mu_\Delta(M) < 1 \Leftarrow |L| < c_H^{\text{nu}} = \begin{cases} \dfrac{1-|w_P|}{|w_O|-|w_P|} & \text{(for frequencies where } |w_O| > 1) \\ \dfrac{1+|w_P|}{|w_O|+|w_P|} & \text{(for frequencies where } |w_O| \leq 1) \end{cases} \tag{27}$$

$$\mu_\Delta(M) < 1 \Leftarrow |S| < c_S^{\text{nu}} = \begin{cases} \dfrac{1-|w_O|}{|w_P|-|w_O|} & \text{(for frequencies where } |w_P| > 1) \\ \dfrac{1+|w_O|}{|w_P|+|w_O|} & \text{(for frequencies where } |w_P| \leq 1) \end{cases} \tag{28}$$

$$\mu_\Delta(M) < 1 \Rightarrow |L| < c_L^{\text{nu}} = \frac{1-|w_P|}{|w_O|-1} \quad \text{(for frequencies where } |w_O| > 1) \tag{29}$$

(4) The necessary lower bounds are

$$\mu_\Delta(M) < 1 \;\Rightarrow\; |H| > c_H^{\text{nl}} = \frac{1 - |w_P|}{|w_O| - |w_P|} \quad \text{(for frequencies where } |w_P| > 1) \tag{30}$$

$$\mu_\Delta(M) < 1 \;\Rightarrow\; |S| > c_S^{\text{nl}} = \frac{1 - |w_O|}{|w_P| - |w_O|} \quad \text{(for frequencies where } |w_O| > 1) \tag{31}$$

$$\mu_\Delta(M) < 1 \;\Rightarrow\; |L| > c_L^{\text{nl}} = \frac{|w_P| - 1}{1 - |w_O|} \quad \text{(for frequencies where } |w_P| > 1) \tag{32}$$

Numerical example: DC motor

The purpose of the following example is to familiarize the reader with the robust loopshaping bounds before tackling the more interesting benchmark problem given later in this paper.

Description. Assume the nominal transfer function is the double integrator

$$P(s) = \frac{1}{s^2} \tag{33}$$

This could describe a DC motor with negligible viscous damping. The nominal model, uncertainty description, and performance specifications for this example come from Reference 12.

We are interested in good tracking over a bandwidth of about 1. If $|S| < 1/|w_P|$, where

$$w_P = \frac{10}{s^3 + 2s^2 + 2s + 1} \tag{34}$$

then the tracking error is at most 10% over the desired closed-loop bandwidth. The true plant is assumed to have a time delay, which was covered by a multiplicative uncertainty of magnitude $|w_O|$ in Reference 12, where

$$w_O = \frac{0.21s}{0.1s + 1} \tag{35}$$

Closed-loop design. The three loopshaping bounds on H and S in (23), (24), (27), (28), (30), (31) are shown together with an example design (solid line) in Figure 4.

We see that the necessary upper bound on S and sufficient upper bound on S coincide at low frequencies. This is true in general when $|w_P| \gg 1 > |w_O|$, since in this case the bounds (24) and (28) both approach

$$\frac{1 - |w_O|}{|w_P|} \tag{36}$$

The necessary upper bound on H and sufficient upper bound on H coincide at high frequencies. This is true in general when $|w_O| \gg 1 > |w_P|$, since in this case the bounds (23) and (27) both approach

$$\frac{1 - |w_P|}{|w_O|} \tag{37}$$

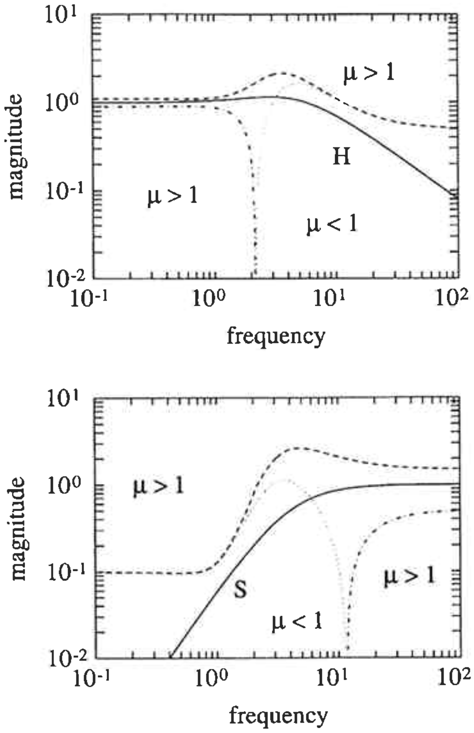

Figure 4. Loopshaping bounds on H and S for DC motor. The upper plot is for H and the lower plot is for S. The dashed lines are necessary upper bounds, the dashed and dotted lines are necessary lower bounds, the dotted lines are sufficient upper bounds, and the solid lines are H and S for an example design

When necessary bounds and sufficient bounds coincide, they become *necessary and sufficient* for $\mu < 1$. The distance between the necessary bounds and sufficient bounds at a given frequency measures the conservatism in the bounds.

Our design approach is to find an S that satisfies nominal stability, satisfies the necessary bounds on S and H for all frequencies, has the sufficient bound on S satisfied for low frequencies, and has the sufficient bound on H satisfied for the other frequencies so that $\mu < 1$ is guaranteed for all frequencies.

The following S satisfies nominal stability (interpolation condition (20))

$$S = \frac{s^2}{\lambda^2 s^2 + 2\lambda s + 1} \tag{38}$$

The complementary sensitivity is specified from $H = 1 - S$. From Figure 4 we see that the necessary bounds on S require a bandwidth (defined by -3 dB roll-off) between 2 and 30 whereas the necessary bounds on H require that a bandwidth between 1·5 and 20. The design shown in Figure 4 with $\lambda = 1/4$ satisfies these bandwidth requirements. The sufficient upper bound on H ensures that $\mu < 1$ for frequencies above 3 and the sufficient upper bound on S ensures that $\mu < 1$ for frequencies below 6; thus $\mu < 1$ for all frequencies. The controller corresponding to S and H,

$$K = (SP)^{-1}(1 - S) = -\frac{15}{16}s^2 + \frac{1}{2}s + 1 \tag{39}$$

is improper, so it is augmented with a second-order filter:

$$\hat{K} = \left(-\frac{15}{16}s^2 + \frac{1}{2}s + 1\right)\Big/\left(\frac{1}{100}s + 1\right)^2$$

Figure 5 is a plot of the structured singular value for the proper controller. The value for μ is less than one for all frequencies, as was implied by the satisfaction of the sufficient bound on S and/or H for each frequency.

The loopshaping bounds directly specify the relative ease or difficulty in meeting the stability and performance specifications. For example, the necessary bounds on S at high frequency are very lenient (far apart). This implies that the closed-loop system can maintain robust performance for much more uncertainty than was used to cover the time delay in the plant. This is confirmed by the structured singular value plot which is much less than 1 at high frequencies.

Open loop design. Using the same specifications as above, Braatz[5] designs an open-loop PD controller via loopshaping L (the details are omitted here for brevity). One interesting result was that the PD controller determining via loopshaping was very close to the μ optimal PD controller determined through extensive global search. This suggests that loopshaping can provide nearly optimal controllers, at least for single-loop systems. The pareto-optimallity of loopshaping controllers has been shown by Doyle *et al.*[12]

Comparison of robust loopshaping with classical loopshaping

For the above SISO problem, the robust loopshaping bounds for L agree with classical loopshaping.[12] Similarly, the sufficient bounds on the closed-loop transfer functions agree with those given by Doyle *et al.*;[12] however, their necessary bounds were incomplete. Knowing the necessary bounds at all frequencies provides the engineer with precise *a priori* bandwidth ranges which must be satisfied by the controller. The distance between the necessary and the sufficient bounds at a given frequency quantifies the conservatism of the bounds; thus the control engineer has some indication where the sufficient bounds can be violated without jeopardizing robust performance.

It is instructive to compare the robust loopshaping method with Quantitative Feedback Theory (QFT).[16] In QFT, loopshaping of the open-loop transfer function $L = PK$ is performed on a Nichols chart. In contrast, robust loopshaping can be performed with any of the common open-

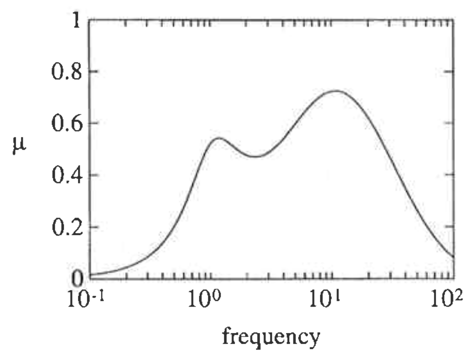

Figure 5. μ for robust performance for DC motor

and closed-loop transfer functions and is performed on a Bode plot. Until recently, calculating the loopshaping bounds used in QFT required gridding of the uncertainty set, which becomes computationally prohibitive as the number of parameters increases. Calculation of the robust loopshaping bounds does not involve gridding over the uncertainty set. Braatz has recently used ideas similar to those used in the development of robust loopshaping to remove the reliance of QFT on gridding of the uncertainty set.[6]

Uncertainty and performance specifications are often not as simple as in the above SISO problem—control problems may involve multiple performance specifications, and uncertainty may be more conveniently described as real parameter variations. Also, it is important for multivariable systems that uncertainty may be present at different locations, for example actuator uncertainty is located at the input of the plant whereas sensor uncertainty is located at the output of the plant. In such cases the loopshaping bounds cannot be derived analytically, but must be obtained numerically using Theorems 3.1–3.4. The following example shows how this is done for a specific SISO problem.

6. BENCHMARK PROBLEM; A COUPLED MASS–SPRING SYSTEM

We apply loopshaping to design a robust controller for an undamped pair of coupled masses with a noncolocated sensor and actuator. This simple problem captures many of the features of more complex aircraft and space structure vibration control problems, and numerous researchers have applied a variety of robust control design methodologies (see Reference 28 for a partial list) to it, most of which have had a very limited amount of success.[27] Braatz and Morari[7] used the DK-iteration method to design a controller which met all the specifications, though the design procedure required several iterations of input–output weights before arriving at an acceptable controller. Part of the difficulty in selecting input–output weights was that available *controller design* software[2,9] does not directly address real uncertainty. Available software can be used to *analyse* robustness with real uncertainty, and this software is used in the calculation of the robust loopshaping bounds. By using robust loopshaping instead of DK-iteration we avoid iterations over weights and instead directly specify the controller to meet the control requirements.

Problem description. Consider the two-mass/spring system in Figure 6, which is a generic model of an uncertain dynamical system with noncolocated sensor and actuator. The system is represented in state-space form as

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k/m_1 & k/m_1 & 0 & 0 \\ k/m_2 & -k/m_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1/m_1 \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1/m_2 \end{bmatrix} w \tag{40}$$

$$y = x_2 + v \tag{41}$$

$$z = x_2 \tag{42}$$

where x_1 and x_2 are the positions of body 1 and body 2, x_3 and x_4 are the velocities of body 1 and body 2, u is the control input acting on body 1, y is the sensor measurement, w is the disturbance acting on body 2, v is sensor noise, and z is the output to be controlled. The spring constant is denoted by k, the mass of body 1 by m_1, and the mass of body 2 by m_2.

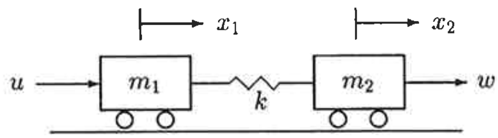

Figure 6. Coupled mass–spring system

The coupled spring–mass system is assumed to have negligible damping. The spring constant and masses are assumed to be uncertain. The actuator is located on body 1 while the sensor is located on body 2, i.e., the sensor and actuator are *noncolocated.* This makes the system much harder to control than in the colocated case.

Certain control specifications of the benchmark problem described in Reference 28 are given concretely, while other specifications were left to the control designer. Here we include additional practical requirements to make the problem more realistic (and hence more difficult), while slightly relaxing the settling time requirement to allow the specifications to be achievable.

Design specifications.

(i) The stability margin with respect to the three uncertain parameters m_1, m_2, k whose nominal values are $m_1 = m_2 = k = 1$ is required to be at least 30%.
(ii) For $w(t)$ = unit impulse at $t = 0$, the performance variable z must have a settling time of 20 seconds for the nominal system $m_1 = m_2 = k = 1$. The settling time is defined to be the time required for the output to reach and stay within 10% of its peak value.
(iii) The control system can tolerate Gaussian white noise with variance of 9×10^{-6}.
(iv) Because of finite actuator response time, the controller bandwidth must be $\leqslant$50 rad/s.
(v) The control input $u(t)$ is limited to $|u| \leqslant 1$.
(vi) The number of controller states should be $\leqslant$4.
(vii) The gain margin should be at least 2 and the phase margin at least 45°.

Though G can be determined analytically (using a similar development as in Reference 7), this is not required since off-the-shelf software builds G from the state-space equations (40)–(42) and the uncertainty specifications (i). We will plot the loopshaping bounds including only the robust stability requirement—the performance requirements will be met by directly specifying the open-loop transfer function. The plant P the open-loop transfer function L for an example design, and the robust stability loopshaping bounds calculated numerically using Theorems 3.1–3.4 are shown in the Bode magnitude plot in Figure 7 (a small amount of complex uncertainty was added to the real perturbation blocks to improve the numerical conditioning as described by Reference 21).

The nominal plant has two poles at $s = 0$ and a pair of poles at $s = \pm i\sqrt{2}$. The peak in the necessary lower bound around $\omega = \sqrt{2}$ reflects the fact that the controller is not allowed to cancel poles of any of the perturbed plants.

Usually a necessary upper bound exists at high frequencies which requires that $|L|$ roll off to satisfy robust stability. Notice that no necessary upper bound exists in Figure 7. The lack of a necessary upper bound and the predominance of the sufficient upper and lower bounds at both high and low frequencies illustrate that *robust* stability is completely determined by the behaviour of L at intermediate frequencies. It would have been difficult to learn this from a black box design method.

Now we design a controller which meets the specifications.

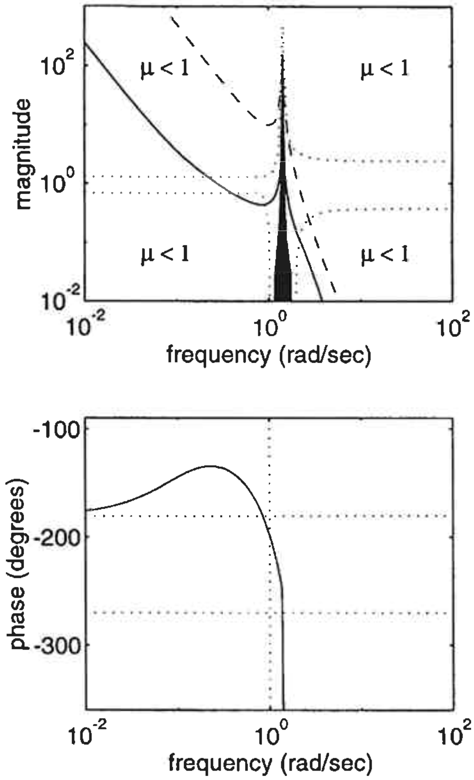

Figure 7. Bode plots with loopshaping bounds on $|L|$ for the benchmark problem. In the magnitude plot, the sufficient regions have dotted boundaries and the sole necessary region is shaded. The dashed line represents ten times the plant ($10P$) and the solid line represents L for an example design. The phase of L for the example design is shown in the lower plot. The plant P has phase of $-180°$ for $\omega < \sqrt{2}$ and $-360°$ for $\omega > \sqrt{2}$.

Open-loop design. We chose the following form for the controller:

$$K(s) = \frac{k(\tau s + 1)(\tau_z^2 s^2 - 2\xi_z \tau_z + 1)}{(\tau_p^2 s^2 + 2\xi_p \tau_p + 1)^2} \tag{43}$$

with $k = 0{\cdot}05$, $\tau = 10$, $\tau_z = 0{\cdot}6$, $\xi_z = 0{\cdot}5$, $\tau_p = 0{\cdot}4$, and $\xi_p = 0{\cdot}6$.

The crossover frequency (i.e. the frequency where the phase of the open-loop transfer function is $-180°$) must be less than $\omega = \sqrt{2}$ because the imaginary axis poles of the plant provide 180° phase lag at this frequency which is detrimental to nominal stability. Thus we need at least 45° phase lead for some frequency range below crossover to satisfy the phase margin specification—this suggests that the controller should have a left half plane (LHP) zero here. Choosing $\tau = 10$ provides enough phase so that the phase margin can be met, without providing too large a gain at crossover ($\omega \approx 1$) which gives poor gain margin (see Figure 7).

For plants with lightly damped or undamped poles, it is necessary to consider the phase margin *after crossover*. The phase must roll off rapidly at crossover to ensure at least a 45° margin. This rapid phase roll-off is provided by the RHP zeros. We choose a pair of zeros with mild damping ($\xi_z = 0{\cdot}5$) so we can get temporary gain roll-off just at crossover which gives