A State-Space Algorithm for Designing \mathcal{H}_{∞} Loop Shaping PID Controllers

A. Umut Genç

October 12, 2000

1 Introduction

Traditionally, many MIMO plants have been controlled by SISO controllers. In these designs the interaction between different channels have been ignored and only one input-output pair of the MIMO plant have been considered for each controller. Although SISO controllers may work satisfactorily for some MIMO plants, advances in performance can only be achieved through the use of a MIMO controller. There are several advanced synthesis techniques available in the literature for MIMO controller design. Such techniques produce optimal yet high order and unstructured controllers. Controller order can be reduced by available methods but the obtained controller is not necessarily optimal in terms of the robust stability margin.

The most popular and commonly used fixed-structure and fixed-order controllers are PIDs. Their simple and intuitive structure leads to easy and quick designs. Furthermore, they have been in use for many years and practical engineers have confidence in their operation. These advantages make PID controllers an attractive option for MIMO plants. Although there are diverse techniques for tuning PID controllers for SISO plants, there seems to be a lack of methods for tuning MIMO PIDs. Most of the research have focused on how to design multi-loop (diagonal) PID controllers for MIMO plants [9, 8, 7, 2], while few techniques have been proposed for designing MIMO PIDs for MIMO plants [13, 6].

This paper proposes a framework to design fixed-structure and fixed-order MIMO controllers in \mathcal{H}_{∞} loop shaping paradigm. The focus here is specifically on PID controllers due to their previously explained popularity. In addition, the particular use of \mathcal{H}_{∞} loop shaping paradigm is due to its efficiency and simplicity in synthesising optimal robust MIMO controllers. By introducing PID controllers into the \mathcal{H}_{∞} loop shaping paradigm, it is possible to design fixed-order, fixed-structure and at the same time optimal and robust MIMO controllers such as MIMO PIDs. The proposed algorithm can be applied to design PID controllers for any kind of plant and is not restricted to process control applications. A similar work has been presented in [11]. However, the problem has been formulated by using a modified Nehari extension problem unlike the BMI-based optimization approach taken in this paper. Furthermore, the pre-compensator W_1 in \mathcal{H}_{∞} loop shaping framework is restricted to be a diagonal fixed-structure transfer matrix in [11] whereas W_1 , W_1^{-1} are only restricted to belong to \mathcal{RH}_{∞} for the presented work.

2 Preliminaries and Notation

The following notation will be used throughout the paper: \mathbb{R} for the field of real numbers, \mathbb{R}_+ for field of strictly-positive real numbers, \mathcal{RH}_∞ for the real-rational subspace of \mathcal{H}_∞ , A^* for the complex conjugate transpose of matrix A, A > 0 for positive-definite matrix A, $\begin{pmatrix} Q & S \\ * & R \end{pmatrix}$ for $\begin{pmatrix} Q & S \\ S^* & R \end{pmatrix}$.

Lemma 2.1 (Bounded Real Lemma) Given a transfer function T(s) of (not necessarily minimal) realization $T(s) = C(sI - A)^{-1}B + D$. The following statements are equivalent:

- $\left\|C(sI-A)^{-1}B + D\right\|_{\infty} < \gamma$ and A is stable;
- there exists a symmetric positive definite solution X to the matrix equality:

$$\begin{pmatrix} XA + A^*X & XB & C^* \\ * & -\gamma I & D^* \\ * & * & -\gamma I \end{pmatrix} < 0$$

3 Overview of \mathcal{H}_{∞} Loop Shaping

 \mathcal{H}_{∞} loop shaping design procedure proposed by McFarlane and Glover is an efficient method to design robust controllers and has been applied to variety of practical control problems successfully [10]. In this framework designer shapes the open-loop scaled plant G with the pre-compensator W_1 and post-compensator W_2 as shown in Figure 1. Once the desired loop shape is achieved, the ∞ -norm of the transfer function matrix from disturbances d_1 and d_2 to the outputs z_1 and z_2 , is minimised over all stabilising controllers K_{∞} to obtain a desired value of γ ,

$$\gamma = \left\| T_{\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \to \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}} \right\|_{\infty} = \left\| \begin{bmatrix} K_{\infty} \\ I \end{bmatrix} (I - G_s K_{\infty})^{-1} \begin{bmatrix} G_s & I \end{bmatrix} \right\|_{\infty}$$
$$\geq \inf_{\text{stab}K_{\infty}} \left\| \begin{bmatrix} K_{\infty} \\ I \end{bmatrix} (I - G_s K_{\infty})^{-1} \begin{bmatrix} G_s & I \end{bmatrix} \right\|_{\infty} = \gamma_{opt} \tag{1}$$

where, $G_s(s)$ is the weighted plant, $K_{\infty}(s)$ is the controller.



Figure 1: \mathcal{H}_{∞} loop shaping setup

The inverse of γ is the so called robust stability margin $\epsilon(\epsilon \text{ is also referred as } b_{P,C})$. The robust stability margin, ϵ , can only take values between zero and unity. It is shown that achieved value of ϵ is an indicator of the success of the design procedure. If the value of ϵ is small it means that desired loop shape and the robust stability requirements can not be achieved simultaneously. In this case the designer should reshape the open-loop scaled plant.

The uncertainty model used in \mathcal{H}_{∞} loop shaping is the coprime factor uncertainty. It does not require a specific knowledge about the uncertainty itself and it captures both low and high frequency perturbations. Furthermore, the real plant and the nominal plant model do not have to have the same number of RHP poles and zeros.

3.1 McFarlane and Glover's Design Procedure

- 1. Scale the inputs and outputs of the nominal plant G_{nom} with pre and post scaling matrices S_1 and S_2 to give scaled plant $G = S_2 G_{nom} S_1$. The scaling is crucial to make sure that each input and output receives the same importance when considering sizes and directions of a MIMO system.
- 2. Shape the singular values of the nominal scaled plant G using a pre-compensator W_1 and/or a post-compensator W_2 to get the desired loop shape as shown in Figure 1. The weighted plant is defined as $G_s = W_2 G W_1$. W_1 and W_2 should be chosen such that G_s contains no hidden modes.
- 3. Minimise ∞ -norm of the transfer matrix $T_{\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \to \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}}$ over all stabilising controllers K_{∞} to give an optimal cost γ_{opt} as

$$\gamma_{opt} = \epsilon_{opt}^{-1} = \inf_{\text{stab}K_{\infty}} \left\| \begin{bmatrix} K_{\infty} \\ I \end{bmatrix} (I - G_s K_{\infty})^{-1} \begin{bmatrix} G_s & I \end{bmatrix} \right\|_{\infty}$$

This is the same as calculating the optimal robust stability margin. Check the achieved ϵ_{opt} ; ϵ_{opt} is a measure of how robust the desired loop shape is. If $\epsilon_{opt} < 0.25$ return to (2) and adjust W_1 and W_2 .

- 4. Synthesise a controller K_{∞} that achieves $\epsilon \leq \epsilon_{opt}$. Choose the position of K_{∞} in the loop, model reduce the controller, design the command pre-filters (if required).
- 5. Check the time simulations and frequency responses of the resulting closed loop system to verify robust performance. Reiterations may be required.

The final controller K is constructed by combining the \mathcal{H}_{∞} controller K_{∞} with the compensator matrices W_1 and W_2 (it is assumed that the scaling matrices S_1 and S_2 have been absorbed in the compensators description at this stage) such that

$$K = W_1 K_\infty W_2.$$

The theoretical basis for \mathcal{H}_{∞} loop shaping is that K_{∞} does not modify the desired loop shape significantly at low and high frequencies, if the achieved ϵ is not too small [14, Section 18.3]. Thus, shaping the open loop plant G corresponds to shaping the loop gains GK and KG. In contrast with conventional loop shaping, the control engineer does not need to shape of the phase G explicitly. It has been shown that a value of $\epsilon > 0.2 - 0.3$ is satisfactory, in the same way that a gain margin of ± 6 dB, and phase margin of 45° are for a SISO system [12, p. 21]. If ϵ is small, then the desired loop shape is incompatible with robust stability requirements and should be adjusted accordingly (note that calculation of ϵ is routine). It is shown in [14, Section 18.3] that all the closed loop objectives are guaranteed to have bounded magnitudes and bounds depend only on ϵ , W_1 , W_2 and G.

The most crucial part of the design procedure is to find the appropriate weighting matrices. The shape of the weights is determined by the closed loop design specifications. The general trends to be followed are high low frequency gain so that disturbance rejection at both the input and output of the plant, and output decoupling are achieved; low high frequency gain for noise rejection; and a smooth transition around the loop cross-over frequency, i.e. the loop gain should not decrease faster than 20dB/decade, in order to achieve desired robust stability, gain and phase margins, overshoot and damping [12, p. 27]. The fast settling time can be achieved with a high loop

cross-over frequency and a good ϵ . The rise time is set by the loop cross-over frequency. High low frequency gain can be achieved with proportional and integral filters placed in the pre-compensator W_1 . Low high frequency gain can be realized with low-pass filters placed in the post-compensator W_2 . Lead-lag filters placed in W_1 can provide the smooth transition around the loop cross-over frequency, if necessary.

4 Controller Structure

There are several PID controller structures used in practice. The following PID structure is adopted in this paper:

$$K_{PID_{ij}}(s) = k_{P_{ij}} + \frac{k_{I_{ij}}}{s} + \frac{k_{D_{ij}}}{\frac{s}{\tau} + 1}$$
(2)

where

 $K_{PID_{ij}}(s)$ is the ij^{th} element of the transfer function matrix $K_{PID}(s)$, $k_{P_{ij}} \in \mathbb{R}$ is the proportional gain of the ij^{th} element of $K_{PID}(s)$, $k_{I_{ij}} \in \mathbb{R}$ is the integral gain of the ij^{th} element of $K_{PID}(s)$, $k_{D_{ij}} \in \mathbb{R}$ is the derivative gain of the ij^{th} element of $K_{PID}(s)$, $\tau \in \mathbb{R}_+$ is the stop frequency of the approximate derivative action.

For such a PID controller there are $3 \times m^2 + 1$ parameters to be tuned for a plant with m inputs and m outputs (plant G is assumed to be square for simplicity).

A minimal state-space realization of $K_{PID}(s)$ can be obtained by using partial fraction expansion.

Equation (2) can be rewritten as

$$K_{PID_{ij}}(s) = \frac{(k_{I_{ij}} - k_{D_{ij}}\tau^2)s + k_{I_{ij}}\tau}{s^2 + \tau s} + (k_{P_{ij}} + k_{D_{ij}}\tau)$$
$$= \frac{\mathcal{K}_{D_{ij}}s + \mathcal{K}_{I_{ij}}}{s^2 + \tau s} + \mathcal{K}_{P_{ij}}.$$
(3)

Then,

$$K_{PID}(s) = \begin{bmatrix} \frac{\mathcal{K}_{D_{11}}s + \mathcal{K}_{I_{11}}}{s^2 + \tau s} + \mathcal{K}_{P_{11}} & \dots & \frac{\mathcal{K}_{D_{1m}}s + \mathcal{K}_{I_{1m}}}{s^2 + \tau s} + \mathcal{K}_{P_{1m}} \\ & \ddots & \ddots & \ddots \\ & \ddots & \ddots & \ddots \\ \frac{\mathcal{K}_{D_{m1}}s + \mathcal{K}_{I_{m1}}}{s^2 + \tau s} + \mathcal{K}_{P_{m1}} & \dots & \frac{\mathcal{K}_{D_{mm}}s + \mathcal{K}_{I_{mm}}}{s^2 + \tau s} + \mathcal{K}_{P_{mm}} \end{bmatrix}.$$
(4)

 $K_{PID}(s)$ has the following partial fraction expansion:

$$K_{PID}(s) = D_c + \frac{B_{c_1}}{s} + \frac{B_{c_2}}{s+\tau},$$
(5)

where,

$$D_{c} = \mathcal{K}_{P} = \begin{bmatrix} \mathcal{K}_{P_{11}} & \cdots & \mathcal{K}_{P_{1m}} \\ \cdot & \cdot & \cdot \\ \mathcal{K}_{P_{m1}} & \cdots & \mathcal{K}_{P_{mm}} \end{bmatrix},$$

$$B_{c_{1}} = \frac{\mathcal{K}_{I}}{\tau} = \begin{bmatrix} \frac{\mathcal{K}_{I_{11}}}{\tau} & \cdots & \frac{\mathcal{K}_{I_{1m}}}{\tau} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \frac{\mathcal{K}_{I_{m1}}}{\tau} & \cdots & \frac{\mathcal{K}_{P_{mm}}}{\tau} \end{bmatrix},$$

$$B_{c_{2}} = \mathcal{K}_{D} - \frac{\mathcal{K}_{I}}{\tau} = \begin{bmatrix} \mathcal{K}_{D_{11}} - \frac{\mathcal{K}_{I_{11}}}{\tau} & \cdots & \mathcal{K}_{D_{1m}} - \frac{\mathcal{K}_{I_{1m}}}{\tau} \\ \cdot & \cdot & \cdot \\ \mathcal{K}_{D_{m1}} - \frac{\mathcal{K}_{I_{m1}}}{\tau} & \cdots & \mathcal{K}_{D_{mm}} - \frac{\mathcal{K}_{P_{mm}}}{\tau} \end{bmatrix}.$$

Assuming,

 $rank B_{c_i} = m,$

then, a minimal realization for $K_{PID}(s)$ is given by

$$K_{PID}(s) = \begin{bmatrix} 0_{m \times m} & B_{c_1} \\ \hline \tau I_{m \times m} & B_{c_2} \\ \hline I_{m \times m} & I_{m \times m} & D_c \end{bmatrix}$$
$$= \begin{bmatrix} A_c & B_c \\ \hline C_c & D_c \end{bmatrix}$$
(6)

This minimal realization is also called Gilbert's realization. Note that C_c is a constant matrix for any PID controller of the specified structure for this minimal realization. If multi-loop PID controllers are required, the matrices B_{c_1} , B_{c_2} and D_c should be specified as diagonal matrices.

5 Proposed Setup

In this section a new setup will be introduced to design PID controllers in the \mathcal{H}_{∞} loop shaping framework. The main objective is to keep all the guidelines and guarantees of the \mathcal{H}_{∞} loop shaping paradigm as described in the Section 3 but to get a PID controller as the final controller. The new setup is shown in Figure 2. In this setup the controller K_{∞} has a particular structure

$$K_{\infty} = W_1^{-1} K_{PID},$$

where

 $W_1 \in \mathcal{RH}_{\infty}, W_1^{-1} \in \mathcal{RH}_{\infty}$, and K_{PID} is a PID controller as specified in the previous section.

This particular structure of K_{∞} will ensure that the final controller K will have the desired PID structure since

$$K = W_1 K_\infty W_2 = K_{PID} W_2.$$

Note that the final controller is a PID controller in series with the post-compensator W2. The post-compensator is used to reject the high frequency noise and this is a common practice in real



Figure 2: \mathcal{H}_{∞} loop shaping setup for tuning PID controllers

applications. If the measurements are noise free, the final controller will be a mere PID controller as W^2 can be chosen as unity.

Although K_{∞} is structured, it still retains the all the robustness and performance guarantees of a \mathcal{H}_{∞} loop shaping controller as long as a satisfactory ϵ is achieved. Furthermore, cancellation $W_1 W_1^{-1}$ does not pose any problems in terms of internal stability and robustness of the closed loop since

- both W_1 and W_1^{-1} are in \mathcal{RH}_{∞} , therefore no hidden modes involved in the cancellation;
- W_1 has no uncertainty involved in it although the plant is uncertain, which makes $W_1W_1^{-1} = I$ always (remember W_1 is not a part of the physical plant).

Thus, the PID controller designed using the proposed setup will have all the robustness and performance guarantees of a \mathcal{H}_{∞} loop shaping controller given that a satisfactory ϵ is achieved. The design problem can be written as an optimisation problem as follows:

$$\min_{\text{stab } K_{PID}} \left\| T_{\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \to \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}} \right\|_{\infty}$$
(7)

or equivalently,

$$\min_{\text{stab } K_{PID}} \gamma = \min_{\text{stab } K_{PID}} \left\| \begin{bmatrix} W_1^{-1} K_{PID} \\ I \end{bmatrix} (I - W_2 G K_{PID})^{-1} \begin{bmatrix} W_2 G W_1 & I \end{bmatrix} \right\|_{\infty}$$

This minimisation is equivalent to maximising the robust stability margin ϵ over all stabilising PID controllers. Optimisation in (7) can not be solved easily as it is non-convex in controller K_{PID} . However, a solution can be seek if the optimisation problem is posed in state-space using matrix inequalities.

6 Solving the Optimisation

The new setup can be transformed to an equivalent \mathcal{H}_{∞} formulation as shown in Figure 3. It shows that although W_1 and W_1^{-1} do not affect the feedback loop, they do affect the value of the



Figure 3: Equivalent \mathcal{H}_{∞} formulation

robust stability margin ϵ . The closed-loop performance is set by K_{PID} and W_2 . On the other hand W_1 affects the feedback loop indirectly through K_{PID} since the parameters of K_{PID} is shaped with respect to the weighted plant G_s which includes W_1 as well. Given appropriate state-space realization of each transfer matrix in Figure 3, a state-space realization for the transfer matrix from d_1 and d_2 to z_1 and z_2 can be obtained as

$$T_{\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \to \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}} = \begin{bmatrix} A_{cl} & B_{cl} \\ \hline C_{cl} & D_{cl} \end{bmatrix}$$

$$= \begin{bmatrix} \tilde{A}_1 & 0 & \tilde{B}_1 C_c & \tilde{B}_1 D_c C & 0 & \tilde{B}_1 D_c \\ 0 & A_1 & 0 & 0 & B_1 & 0 \\ 0 & 0 & A_c & B_c C & 0 & B_c \\ 0 & BC_1 & BC_c & A + BD_c C & BD_1 & BD_c \\ \hline \tilde{C}_1 & 0 & \tilde{D}_1 C_c & \tilde{D}_1 D_c C & 0 & \tilde{D}_1 D_c \\ 0 & 0 & 0 & C & 0 & I \end{bmatrix}$$

$$(8)$$

where

$$\tilde{G}_s(s) = W_2 G = \begin{bmatrix} A & B \\ \hline C & 0 \end{bmatrix},$$

$$K_{PID}(s) = \begin{bmatrix} A_c & B_c \\ \hline C_c & D_c \end{bmatrix},$$

$$W_1(s) = \begin{bmatrix} A_1 & B_1 \\ \hline C_1 & D_1 \end{bmatrix},$$

$$W_1^{-1}(s) = \begin{bmatrix} \tilde{A}_1 & \tilde{B}_1 \\ \hline \tilde{C}_1 & \tilde{D}_1 \end{bmatrix},$$

This state-space realization is convex in controller parameters. Now the optimisation problem in (7) can be posed in terms of matrix inequalities in state-space using the Bounded Real Lemma.

The optimisation (7) can be rewritten as:

$$\begin{array}{ccc}
& \min_{X,A_c,B_c,D_c} \gamma \\
& & \text{such that} \\
\begin{array}{ccc}
& & (9) \\
& & XA_{cl} + A_{cl}^* X & XB_{cl} & C_{cl}^* \\
& & & -\gamma I & D_{cl}^* \\
& & & & * & -\gamma I \end{array}\right) < 0, \qquad X > 0.
\end{array}$$

This is a BMI optimisation problem, i.e. it is convex in X and convex in controller parameters A_c , B_c and D_c but biconvex in all of them together. While it is straightforward to find at least one local minimum, global optimisation is hard in general. There are several methods proposed for solving the BMI optimisation problem in the literature [5, 4, 3, 2]. Most of these methods are still in their developing phases and can not be applied to practical problems easily. A simple and reliable way of getting an answer to the optimisation problem (9) is to solve it iteratively in a similar way to D-K iteration. In this method a local solution to the BMI optimisation problem is seek by alternately minimising the optimisation $\cos \gamma$ with respect X with controller parameters fixed and vice versa. This method has the advantage that readily available efficient interior-point methods can be used to solve the problem. The main disadvantage is that this approach is not guaranteed to converge to a stationary point due to the non-smoothness of the function [5]. However, experience showed that it works satisfactorily in practice.

Once the scaled open-loop plant is shaped with pre- and post-compensators to achieve the desired closed-loop properties, the \mathcal{H}_{∞} loop shaping PID controller can be designed as follows:

- 1. Given G, W_1 and W_2 , find an initial PID controller which stabilises the closed-loop. Obtain the state-space realization, A_{c_0} , B_{c_0} , C_c , D_{c_0} for the initial PID controller as described in section 4. Calculate the initial optimisation cost $\left\|T_{\begin{bmatrix} d_1\\ d_2\end{bmatrix} \rightarrow \begin{bmatrix} z_1\\ z_2\end{bmatrix}}\right\|_{\infty} \approx \gamma_0$ (γ_0 is a very tight upper bound for the cost). It is important that the initial optimisation cost γ_0 is not too large. Set $\gamma_{s_0} = \gamma_0$, i = 0.
- 2. i = i + 1.
- 3. Given $A_{c_{i-1}}$, $B_{c_{i-1}}$, $D_{c_{i-1}}$ and $\gamma_{s_{i-1}}$, calculate A_{cl} , B_{cl} , C_{cl} , D_{cl} using (8) and solve the following LMI feasibility problem for X_i ,

$$\begin{pmatrix} X_i A_{cl} + A_{cl}^* X_i & X_i B_{cl} & C_{cl}^* \\ * & -\gamma_{s_{i-1}} I & D_{cl}^* \\ * & * & -\gamma_{s_{i-1}} I \end{pmatrix} < 0, \qquad X_i > 0$$
(10)

This gives a positive definite matrix satisfying the closed-loop \mathcal{H}_{∞} norm condition for $K_{PID_{i-1}}$. As solution to the feasibility problem is not unique, the algorithm may converge to slightly different minimums from the same initial point every time it is run. There are $n \times (n+1)/2$ decision variables to be solved in the feasibility problem for a closed-loop with n states. This step constitutes the main computational burden of the optimization.

4. Given X_i , solve the following LMI minimisation problem

$$\begin{array}{c} \min_{A_{c_i}, B_{c_i}, D_{c_i}} \gamma_{s_i} \\ \text{such that} \end{array} \tag{11}$$

$$\begin{pmatrix} X_i A_{cl} + A_{cl}^* X_i & X_i B_{cl} & C_{cl}^* \\ * & -\gamma_{s_i} I & D_{cl}^* \\ * & * & -\gamma_{s_i} I \end{pmatrix} < 0,$$

If there is a solution to the LMI minimisation problem, go to Step 2. If there is no solution to the minimisation problem, go to the next step. Outputs of the minimisation are A_{c_i} , B_{c_i} , D_{c_i} , which describe the i^{th} PID controller, K_{PID_i} , and γ_{s_i} . There are $2 \times m^2 + 2$ decision variables to be solved in this step for a $m \times m$ plant.

5. Final values of τ , $\mathcal{K}_P, \mathcal{K}_I \mathcal{K}_D$ can be obtained from $A_{c_i}, B_{c_i}, D_{c_i}$ using the equation (5). Then, it is straightforward to calculate PID parameters using the equation (3).

Remark 1: The minimum value of γ_{s_i} at Step 4 can be very conservative in some cases since the X_i is fixed during the minimisation (11). Denote the actual optimisation cost achieved as γ_{a_i} , then following equality always holds

$$\left\| T_{\begin{bmatrix} d_1\\ d_2 \end{bmatrix} \to \begin{bmatrix} z_1\\ z_2 \end{bmatrix}} \right\|_{\infty} \approx \gamma_{a_i} \le \gamma_s$$

and the difference between γ_{a_i} and γ_{s_i} can be large. Despite

$$\gamma_{s_i} < \gamma_{s_j}, \qquad i > j$$

always holds,

$$\gamma_{a_i} < \gamma_{a_j} \qquad i > j$$

is not true necessarily. Hence, the algorithm may converge to a final γ_a which is not the best actual cost achieved during the optimisation. However, this problem can be overcome easily since it is possible to store all the outputs of the optimisation at each step.

Remark 2:The algorithm presented above is a descent algorithm, i.e. the value of γ_{s_i} is monotonically non-increasing as *i* increases and that the minimum cost γ_{s_i} obtained in Step 4 is less than $\gamma_{s_{i-1}}$. However, the above algorithm can not be guaranteed to converge to a local/global minimum [5]. Only monotonicity properties can be guaranteed.

Remark 3: The proposed framework is not restricted to PID controllers. K_{PID} can be replaced by any other fixed-structure fixed-order controller and the algorithm will still work.

6.1 Choosing an Initial PID controller

It is crucial to start the algorithm with a sensible initial stabilising controller. One way of choosing a sensible initial PID controller is by inspecting the weights of the pre-compensator W_1 . In general it is sufficient to place first or second order transfer functions as weights at the diagonals entries of W_1 . It is not difficult to transform these transfer functions to PI/PID transfer functions. Once the PID gains of the weights are known or the weights are approximated by PID weights, the initial PID controller can be chosen as a PID controller with the gains less than the PID gains of weights in the pre-compensator W_1 .

All these points will be more clear in the next section, when \mathcal{H}_{∞} loop shaping PID controllers are designed using the proposed algorithm for a number of case studies from the literature.

7 Design Examples

Example 1: The first example is taken from process control literature [7]. Consider a 24-tray tower separating methanol and water with the following transfer function model for controlling the temperature on the 4th and 17th trays

$$\begin{bmatrix} t_{17} \\ t_4 \end{bmatrix} = \begin{bmatrix} \frac{-2.2e^{-s}}{7s+1} & \frac{1.3e^{-0.3s}}{7s+1} \\ \frac{-2.8e^{-1.8s}}{9.5s+1} & \frac{4.3e^{-0.35s}}{9.2s+1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}.$$
 (12)

This is a stable plant with moderate time delays and interaction between its channels. It is desired to have the unity cross-over frequency at around 2 rad/s. The diagonal pre-compensator and post-compensator are chosen as follows:

$$W_{1} = \begin{bmatrix} \frac{5s+2}{s+0.001} & 0\\ 0 & \frac{5s+2}{s+0.001} \end{bmatrix}$$
$$W_{2} = \begin{bmatrix} \frac{10}{s+10} & 0\\ 0 & \frac{10}{s+10} \end{bmatrix}$$
(13)

Note that approximate PI weights are used in the pre-compensator to ensure almost zero steadystate error and a $w_c \approx 2$ rad/s. The time delays are realized in state-space by their 2^{nd} order Pade approximations. The singular values of the nominal and shaped plant are shown in Figure 4(a). $\epsilon_{opt} = 0.3607$ for the shaped plant for an unstructured \mathcal{H}_{∞} loop-shaping controller. This shows that desired loop-shape can be achieved along with robustness requirements. Since the open-loop plant is stable, it is not difficult to find a feasible initial stabilising PID controller. As the desired loop shape is defined by approximate PI weights, the initial PID parameters are chosen a bit lower than the PI parameters in W_1 as:

$$k_{P_0} = \begin{bmatrix} 3 & 0 \\ 0 & -3.5 \end{bmatrix}, \\ k_{I_0} = \begin{bmatrix} 0.5 & 0 \\ 0 & -0.6 \end{bmatrix}, \\ k_{D_0} = \begin{bmatrix} 0.01 & 0 \\ 0 & -0.01 \end{bmatrix}, \\ \tau_0 = 100.$$
(14)

This initial PID controller gives $\gamma_0 = 12.8$, which is a sensible initial cost. The weighted plant has 16 states, the feasibility problem (10) has 253 decision variables and minimisation (11) has 14 decision variables to solve at each step. Figure 4(b) shows how the algorithm converges to a satisfactory solution. The difference between the actual cost and synthetic cost is very small for this example. The best value of optimisation cost, $\gamma_s = 4.0582(\epsilon = 0.2464)$, is achieved by a PID controller described by

$$k_{P} = \begin{bmatrix} 2.4719 & -1.2098 \\ -1.1667 & -2.4766 \end{bmatrix},$$

$$k_{I} = \begin{bmatrix} 0.4657 & -0.31 \\ -0.2329 & -0.487 \end{bmatrix},$$

$$k_{D} = \begin{bmatrix} 0.0534 & -0.0072 \\ -0.015 & -0.0434 \end{bmatrix},$$

$$\tau = 16.61.$$
(15)

It took only 38 minutes to converge this solution ¹. In Figure 4(c) singular values of the suboptimal unstructured \mathcal{H}_{∞} loop shaping controller ($\epsilon = 0.3309$) and the PID controller ($\epsilon = 0.2464$) are compared. The PID controller approximates the unstructured controller quite well up to the cross-over frequency. While the unstructured controller can shape the loop around cross-over frequency optimally, the PID controller can only crudely approximates the optimal behaviour of

¹In all the examples the algorithm is run on a Pentium III PC



Figure 4: Process control example 1

the unstructured controller as it has very limited freedom. Furthermore, PID controller can not roll-off after cross-over like the unstructured controller since it is a proper controller. The rolloff in the singular values of PID controller after the cross-over frequency only due to the postcompensator W_2 . Finally, the time simulations of the closed-loop are performed for reference tracking in Figure 4(d). It is seen that both the suboptimal unstructured \mathcal{H}_{∞} loop shaping controller and \mathcal{H}_{∞} loop shaping PID controller perform in a similar manner.

Example 2: This example is taken from the μ Analysis and Synthesis Tool Box User's Guide [1]. A pitch axis controller will be designed for an experimental highly maneuverable airplane, HIMAT. The problem is posed as a robust performance problem, with multiplicative plant uncertainty at the plant input and plant output weighted sensitivity function as the performance criterion. An exact solution to this robust performance problem can be obtained via μ synthesis. However, a satisfactory controller can be designed easily by \mathcal{H}_{∞} loop shaping. The objective is to reject disturbances up to about 1 rad/s in the presence of substantial plant uncertainty above 100 rad/s.



Figure 5: HIMAT example

This can be achieved by placing the bandwidth of the loop shape around 10 rad/s and satisfying robustness requirements. Singular values of HIMAT is given in Figure 5(a).

Although the unity crossover frequency is approximately correct, the low frequency gain is too low. Therefore, an approximate PI pre-compensator with the following transfer function is introduced to boost the low frequency gain and give almost zero steady state error:

$$W_1 = \begin{bmatrix} \frac{s+1}{s+0.001} & 0\\ 0 & \frac{s+1}{s+0.001} \end{bmatrix}$$
(16)

For the purposes of this example a post-compensator is not necessary. The optimal robust stability margin $\epsilon_{opt} = 0.436$ for the weighted plant. The weighted plant has 6 states and there are 78 decision variables in the feasibility problem (10) and 14 decision variables in the minimisation (11). In order to start the optimisation, a feasible initial PID controller should be found. By inspecting

the weights in W_1 a feasible initial PID controller is chosen as follows:

$$k_{P_0} = \begin{bmatrix} 0.2 & 0 \\ 0 & -0.15 \end{bmatrix}, \\ k_{I_0} = \begin{bmatrix} 0.11 & 0 \\ 0 & -0.1 \end{bmatrix}, \\ k_{D_0} = \begin{bmatrix} 0.015 & 0 \\ 0 & -0.01 \end{bmatrix}, \\ \tau_0 = 100,$$
(17)

which gives $\gamma_0 = 14.3975$.

An \mathcal{H}_{∞} loop shaping PID controller can now be designed for shaped plant. Figure 5(b) shows how the algorithm converges to the final solution (It took only 27 seconds to converge this solution). γ_s achieves a minimum of 3.2416 ($\epsilon = 0.3085$) whereas γ_a has a minimum of 2.9151 ($\epsilon = 0.343$). Both solutions are quite satisfactory and PID parameters for $\epsilon = 0.3085$ are given as:

$$k_P = \begin{bmatrix} 1.3074 & -0.0601 \\ 1.3414 & -1.3123 \end{bmatrix},$$

$$k_I = \begin{bmatrix} 1.2729 & -0.0795 \\ 1.3609 & -1.2921 \end{bmatrix},$$

$$k_D = \begin{bmatrix} 0.0077 & 0.0043 \\ -0.0069 & -0.0039 \end{bmatrix},$$

$$\tau = 99.5724,$$

Figure 5(c) shows the singular values of the optimal unstructured \mathcal{H}_{∞} controller ($\epsilon = 0.436$) and the PID controller ($\epsilon = 0.3085$). The PID controller mimics the behaviour of the unstructured \mathcal{H}_{∞} loop-shaping controller at low frequencies and around crossover frequency. However, it can not roll-off like the unstructured controller at high frequencies. Finally, Figure 5(d) compares the μ achieved by the PID and unstructured \mathcal{H}_{∞} controller. The PID controller has better robust performance at low frequencies than the optimal \mathcal{H}_{∞} loop shaping controller.

Example 3: The plant considered in this example is a scaled-down model of the High Incidence Research Model (HIRM) developed by the Defence Evaluation and Research Agency (DERA) in Bedford. This linear time-invariant model of the plant has two inputs, roll and yaw thrusters, and 6 states. Two of the states are due to first order Pade approximation of a 0.05 s time delay at each input. The measured outputs that are used for feedback are roll and yaw angles. the nominal plant is unstable and non-minimum phase. The unstable pole, due to an unstable yaw mod, has a natural frequency of about 2.5 rad/s and RHP zero, due to the Pade approximation, has a natural frequency of about 40 rad/s. These restrict the closed-loop bandwidth of each channel to lie between 2.5 and 40 rad/s.

The desired closed-loop bandwidth is around 20 rad/s. With the following weights

$$W_{1} = \begin{bmatrix} 70\frac{s+2}{s+0.001}\frac{4s+15}{s+60} & 0\\ 0 & 70\frac{s+2}{s+0.001}\frac{4s+15}{s+60} \end{bmatrix}$$
$$W_{2} = \begin{bmatrix} \frac{100}{s+100} & 0\\ 0 & \frac{100}{s+100} \end{bmatrix}$$
(18)

the desired bandwidth, integral action (approximately), and high frequency noise rejection after 100rad/s are achieved. Such weights give a good ϵ_{opt} of 0.3253 for an unstructured \mathcal{H}_{∞} loop-shaping



Figure 6: HIRM example

controller. The weighted plant shown in Figure 6(a) has 12 states. The feasibility problem (10) has 210 decision variables and minimisation (11) has 14 decision variables to solve at each step. Although the scale of the problem is moderate, this is a more difficult plant to control with a RHP pole and zero. Each weight in the pre-compensator is composed of an approximate PI filter in series with a lead-lag filter. These weights can easily be approximated in PID form as

$$70\frac{s+2}{s}\frac{4s+15}{s+60} = 26.25 + \frac{35}{s} + \frac{4.23}{\frac{s}{60}+1}$$

Therefore, an initial PID controller can be chosen as

$$k_{P_0} = \begin{bmatrix} -5 & 0 \\ 0 & -6 \end{bmatrix}, \\ k_{I_0} = \begin{bmatrix} -7 & 0 \\ 0 & -8 \end{bmatrix}, \\ k_{D_0} = \begin{bmatrix} -2 & 0 \\ 0 & -3 \end{bmatrix}, \\ \tau_0 = 100.$$
(19)

This PID controller results in an initial cost of $\gamma_0 = 10.29$. The algorithm converges to a γ of $5.18(\epsilon = 0.19)$ as illustrated in Figure 6(b) and the final PID controller is given as

$$k_{P} = \begin{bmatrix} -6.9801 & 2.6003 \\ -7.8551 & -7.057 \end{bmatrix},$$

$$k_{I} = \begin{bmatrix} -11.1242 & 3.3332 \\ -2.8625 & -9.5152 \end{bmatrix},$$

$$k_{D} = \begin{bmatrix} -2.0006 & 0.3703 \\ 0.1794 & -3.2371 \end{bmatrix},$$

$$\tau = 98.5426.$$
(20)

This time it took 61 minutes to complete the 132 iterations of the algorithm. The singular values of a suboptimal unstructured controller ($\epsilon = 0.2983$) and the PID controller ($\epsilon = 0.19$) are compared in Figure 6(c). While, the PID controller perfectly matches the unstructured controller at low frequencies, its singular values are just an approximation of the singular values of the unstructured controller around cross-over frequency. Finally, Figure 6(d) shows the closed-loop responses of the unstructured and the PID controllers for unit step references. Both controller perform similarly even though the unstructured controller has much better robust stability margin.

Example 4: This is a well-known binary distillation plant from process control literature [13]:

$$G(s) = \begin{bmatrix} \frac{-12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21s+1} \\ \frac{-6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix}.$$
 (21)

The plant has strong interaction and significant time delays. It is desired to place the closedloop bandwidth around 0.5 rad/s. The time delays are realized in state-space by 3th order Pade approximations since a less order Pade approximation could not represent the plant phase around the desired cross-over frequency accurately. The compensators are chosen as

$$W_{1} = \begin{bmatrix} 0.5 \frac{s+0.08}{s+0.001} \frac{2s+0.5}{s+1} & 0\\ 0 & 0.5 \frac{s+0.08}{s+0.001} \frac{2s+0.5}{s+1} \end{bmatrix},$$
$$W_{2} = \begin{bmatrix} \frac{100}{s+100} & 0\\ 0 & \frac{100}{s+100} \end{bmatrix}.$$
(22)

As shown in Figure 7(a) the desired bandwidth is achieved with the above compensators. The optimal robust stability margin for the shaped plant is $\epsilon_{opt} = 0.4245$. A good ϵ is necessary for this plant due its long time delays. Initial PID controller can be chosen by inspecting the weights in



Figure 7: Process control example 2

the pre-compensator as in the previous example. The following initial PID gains,

$$k_{P_0} = \begin{bmatrix} -0.04 & 0 \\ 0 & 0.04 \end{bmatrix}, \\ k_{I_0} = \begin{bmatrix} -0.035 & 0 \\ 0 & 0.035 \end{bmatrix}, \\ k_{D_0} = \begin{bmatrix} -0.4 & 0 \\ 0 & 0.4 \end{bmatrix}, \\ \tau_0 = 5,$$
(23)

result in an initial optimisation cost, γ_0 , of 57.53. After 180 iterations the cost, γ converges to $3.07(\epsilon = 0.3252)$ as shown in Figure 7(b). Note that the weighted plant has 24 states, the feasibility problem (10) has 465 decision variables and minimisation (11) has 14 decision variables to solve at

each step (It took 7 hours and 4 minutes to complete 180 iterations). The final gains of the PID controller parameters are

$$k_{P} = \begin{bmatrix} -0.1343 & -0.0012\\ 0.077 & 0.0924 \end{bmatrix},$$

$$k_{I} = \begin{bmatrix} -0.0382 & 0.0087\\ 0.0187 & 0.0115 \end{bmatrix},$$

$$k_{D} = \begin{bmatrix} -0.1883 & -0.1015\\ 0.2222 & 0.1958 \end{bmatrix},$$

$$\tau = 5.5997.$$
(24)

The singular values of a suboptimal unstructured controller ($\epsilon = 0.39$) and the PID controller ($\epsilon = 0.3253$) is compared in Figure 7(c). Finally, the response of the \mathcal{H}_{∞} loop shaping controller is compared with the response of the autotuning MIMO PID controller designed in [13]. \mathcal{H}_{∞} loop shaping PID controller has faster settling times in both channels but the decoupling in y_2 channel is poorer.

So far in all the examples, it has been shown that the algorithm works and converges to a solution. However, the characteristics of this solution have not been specified. That is, whether or not a local/global minimum is reached or if not how close the solution is to a local/global minimum have not been investigated. With these questions in mind Example 4 will be reconsidered with the same weights as before, but this time with a different initial PID controller. The parameters of the new initial PID controller are chosen by inspecting the gains of the autotuning PID from [13] as

$$k_{P_0} = \begin{bmatrix} -0.17 & 0 \\ 0 & 0.06 \end{bmatrix},$$

$$k_{I_0} = \begin{bmatrix} -\frac{0.017}{4} & 0 \\ 0 & \frac{0.06}{4.25} \end{bmatrix},$$

$$k_{D_0} = \begin{bmatrix} -0.2 & 0 \\ 0 & 0.3 \end{bmatrix},$$

$$\tau_0 = 5.$$
(25)

Such an initial PID controller gives a $\gamma_0 = 4.44$. This time the algorithm converges to a minimum value of $\gamma = 2.76$ ($\epsilon = 0.36$) after 145 iterations as depicted in Figure 8(a). In order to assess the closeness of this value of γ to the global minimum of γ , the value of the global minimum must be calculated. Since this is not possible in a feasible time period, the global minimum $(\epsilon_{opt}^{-1} = (0.4245)^{-1} = 2.356)$ for the unstructured \mathcal{H}_{∞} loop shaping controller can be taken as a lower bound for the global minimum for \mathcal{H}_{∞} loop shaping PID controller. In this respect the achieved value of $\gamma = 2.76$ is very close to the worst case global minimum of 2.356. The final PID controller is given by

$$k_{P} = \begin{bmatrix} -0.2536 & -0.0399\\ 0.06 & 0.0502 \end{bmatrix},$$

$$k_{I} = \begin{bmatrix} -0.0643 & 0.0169\\ 0.0094 & 0.0052 \end{bmatrix},$$

$$k_{D} = \begin{bmatrix} -0.2025 & -0.0494\\ 0.0865 & 0.2537 \end{bmatrix},$$

$$\tau = 5.1543.$$
(26)



PID (-, $\epsilon = 0.32$), PID (.., $\epsilon = 0.36$)

Figure 8: Process control example 2 with a different initial controller

The singular values of the new PID controller is shown in Figure 8(b). The new controller ($\epsilon = 0.36$) has a better condition number around cross-over frequency in comparison to the previous PID ($\epsilon = 0.3253$). The reference tracking performances of all controllers are plotted in Figure 8(c). The new PID has faster response in y_1 channel but its rise time is slower than the others in y_2 channel.

Finally, this example clarifies two important properties of the proposed algorithm:

- It is possible to reach a better solution by trying different initial controllers.
- Although the algorithm is not guaranteed to converge to a local/global minimum, it may give a solution very close a local/global minimum.

8 Conclusion

This paper addresses the problem of designing a fixed-structure fixed-order \mathcal{H}_{∞} loop shaping controller. The problem has been formulated in particular for PIDs as they are the most popular fixed-structure and fixed-order controllers. The proposed algorithm which gives a sub-optimal solution is developed in state-space by using matrix inequalities. The main advantage of this algorithm is that a fixed-structure and fixed-order controller can be designed in a systematic way to achieve desired robustness and performance for MIMO plants. In particular, unstructured \mathcal{H}_{∞} loop shaping controllers can be approximated by PID controllers whenever there exists a stabilising PID controller for a given plant. Several examples have been solved using the proposed algorithm and it has been shown that satisfactory PID controllers can be obtained.

The proposed algorithm however has some significant shortcomings. Firstly, it is not guaranteed whether the solution will converge to a local/global minimum. Secondly, the chance of reaching a satisfactory solution depends highly on the initial controller chosen. Consequently, further research is necessary to overcome these problems.

References

- G.J. Balas, J.C. Doyle, K. Glover, A. Packard, and R. Smith. μ-Analysis and Synthesis Toolbox Users' Guide for Use with MATLAB. The Mathworks Inc., 1993.
- [2] J. Bao, J.F. Forbes, and P.J. McLellan. Robust Multiloop PID Controller Design: A Successive Semidefinite Programming Approach. Industrial & Engineering Chemistry Research, 38(9):3407–3419, 1999.
- [3] E.G. Collings, D. Sadhukhan, and L.T. Watson. Robust Controller Synthesis via Nonlinear Matrix Inequalities. In *Proceedings of the American Control Conference*, pages 67–71, June 1997.
- [4] K.C. Goh, M.G. Safonov, and G.P Papavassilopoulos. A Global Optimization Approach for the BMI Problem. In *Proceedings of the 33rd Conference on Decision and Control*, pages 2009–2014, December 1994.
- [5] K.C. Goh, L. Turan, M.G. Safonov, G.P. Papavassilopoulos, and J.H. Ly. Biaffine Matrix Inequality Properties and Computational Methods. In *Proceedings of the American Control Conference*, pages 850–855, June 1994.
- [6] J. Lieslehto, J.T. Tanttu, and H.N. Koivo. An Expert System for Multivariable Controller Design. Automatica, 29(4):953–968, 1993.
- [7] A.P. Loh, C.C. Hang, K. Quek, and V.U. Vasnani. Autotuning of Multiloop Proportional-Integral Controllers Using Relay Feedback. *Industrial & Engineering Chemistry Research*, 32:1102–1107, 1993.
- [8] A.P. Loh and V.U. Vasnani. Describing function matrix for Multivariable Systems and its Use in Multiloop PI Design. *Journal of Process Control*, 4(3):115–120, 1994.
- [9] L.W. Luyben. Simple Method for Tuning SISO Controllers in Multivariable Systems. Ind. Eng. Chem. Process Des. Dev., 25:654–660, 1986.
- [10] D. McFarlane and K. Glover. A Loop Shaping Design Procedure Using \mathcal{H}_{∞} Synthesis. *IEEE Transactions on Automatic Control*, 37(6):759–769, June 1992.

- [11] S. Miyamoto. Robust Control Design-A Coprime Factorization and LMI approach. PhD thesis, University of Cambridge, Department of Engineering, May 1998.
- [12] G. Papageorgiou. Robust Control System Design: \mathcal{H}_{∞} Loop Shaping and Aerospace Applications. PhD thesis, University of Cambridge, Department of Engineering, July 1998.
- [13] Q. Wang, B. Zhou, T. Lee, and Q. Bi. Autotuning of Multivariable PID Controllers from Decentralized Relay Feedback. Automatica, 33(3):319–330, 1997.
- [14] K. Zhou, J.C. Doyle, and K. Glover. Robust and Optimal Control. Prentice-Hall, 1996.