HAL
open science

An extension of the linear quadratic Gaussian-loop transfer recovery procedure

Laleh Hosseini-Ravanbod, Dominikus Noll, Pierre Apkarian

To cite this version:

Laleh Hosseini-Ravanbod, Dominikus Noll, Pierre Apkarian. An extension of the linear quadratic Gaussian-loop transfer recovery procedure. IET Control Theory and Applications, 2012, 6 (14), pp.2269 - 2278. 10.1049/iet-cta.2011.0308 . hal-01868671

HAL Id: hal-01868671

https://hal.science/hal-01868671v1

Submitted on 17 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An extension of the LQG-LTR procedure

L. Ravanbod-Hosseini*† D. Noll *† P. Apkarian*‡

Abstract

The LQG/LTR procedure is a classical method to desensibilize a system in closed loop with respect to disturbances and system uncertainty. We discuss an extension which avoids the usual loss of performance in LTR. We also show how the idea can be extended to other control structures. In particular, we show how PID controllers can be desensibilized with this new approach.

Keywords: LQG/LTR, observer-based control, PID control, mixed H_2/H_∞ synthesis, trade-off, structured control law.

1 Introduction

It became apparent during the late 1960s that LQG controllers often lack robustness with regard to system uncertainty. In 1966 Kwakernaak [17] proposed *loop transfer recovery* (LTR) as a means to overcome this deficit in practical situations. LTR was later re-discovered and popularized in a series of papers by Stein and Athans [23], Doyle and Stein [14, 15]. Even today LQG/LTR is still used by practitioners to desensibilize LQG controllers to enhance the robustness of a design.

Unfortunately, LTR has two main limitations. On the one hand, the price for the enhanced robustness may be a considerable loss of performance. And secondly, LTR is limited to controllers with observer structure. Here we propose a new method which avoids these shortcomings. We demonstrate its use for observer-based controllers and also for multi-variable PID controllers.

In our approach a trade-off between performance robustness is formalized as a multi-objective optimization program. We propose to optimize performance in closed loop under a constraint on the H_∞ norm of the (input or output) sensitivity function, combined with a structural constraint on the controller. This has the merits of the LTR controller, reduced sensitivity, but with only a slight loss over nominal performance. Our approach is much in the spirit of [23], where it is already shown that a similar trade-off between sensitivity and complementary sensitivity can be cast as an optimization problem over the Hardy space of stable transfer functions with 2-norm, i.e. an H_2-optimization problem, which under some restrictions can be solved by LQG/LTR.

The structure of the paper is as follows. In Sections 2 and 3 we recall the essential features of LQG/LTR, presented for the case of the input loop breaking point. The improved LTR procedure

*Supported by grants from Fondation EADS and Fondation de Recherche pour l'Aéronautique et l'Espace

†L. Ravanbod-Hosseini and D. Noll are with Université Paul Sabatier, Institut de Mathématiques, 118 route de Narbonne, 31062 Toulouse, France; `laleh.hosseini@math.univ-toulouse.fr; noll@mip.ups-tlse.fr`

‡P. Apkarian is with ONERA, Department of Systems Control and Flight Dynamics, 2 av. Edouard Bélin, 31055 Toulouse, France; `Pierre.Apkarian@onera.fr`

for this case is presented in Section 4. Section 5 briefly discusses LTR at the output loop breaking point. Section 6 gives a dual mathematical programming approach, where the roles between performance and robustness in the trade-off are changed. More general controller structures are discussed in Section 7. Experiments are presented in Sections 8 and 9.

2 Preparation

Let us briefly recall the set-up for H_2-synthesis. Given an open-loop plant in state-space form

$$P : \left[\begin{array}{c} \dot{x} \\ \hline z_2 \\ y \end{array}\right] = \left[\begin{array}{c|cc} A & B_2 & B \\ \hline C_2 & 0 & D_{2u} \\ C & D_{y2} & 0 \end{array}\right] \left[\begin{array}{c} x \\ \hline w_2 \\ u \end{array}\right], \tag{1}$$

the goal of H_2 synthesis is to find a dynamic output feedback controller in state space form

$$K : \left[\begin{array}{c} \dot{x}_K \\ u \end{array}\right] = \left[\begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array}\right] \left[\begin{array}{c} x_K \\ y \end{array}\right] \tag{2}$$

which stabilizes P in closed loop and minimizes the H_2 norm (cf. [11])

$$\min_K \|T_{w_2 \to z_2}(P, K)\|_2 \tag{3}$$

of the closed-loop performance channel $w_2 \to z_2$. We refer to $\mathcal{P}(K) = \|T_{w_2 \to z_2}(P, K)\|_2$ as the performance of the closed-loop system. It is well known that the optimal solution K^* of (3) has observer-based structure

$$K^* = \left[\begin{array}{c|c} A - B_2 K_c - K_f C_2 & K_f \\ \hline -K_c & 0 \end{array}\right], \tag{4}$$

and that K_f, K_c can be computed via AREs or LMIs [13].

It is convenient to consider LQG control as a special case of H_2 synthesis. Consider the following LQG problem:

$$G_{\text{LQG}} : \begin{cases} \dot{x} = Ax + Bu + \Gamma w \\ y = Cx + v \end{cases}$$

where w and v are white noise with covariance matrices W and V, respectively. We denote by Q and R the state and input weighting matrices used to define the LQG criterion and build a plant of form (1) by setting

$$P_{\text{LQG}} = \left[\begin{array}{c|c|c} A & B_2 & B \\ \hline C_2 & 0 & D_{2u} \\ \hline C & D_{y2} & 0 \end{array}\right] = \left[\begin{array}{c|cc|c} A & \left(\Gamma W \Gamma^\top\right)^{1/2} & 0 & B \\ \hline Q^{1/2} & 0 & 0 & 0 \\ 0 & 0 & 0 & R^{1/2} \\ \hline C & 0 & V^{1/2} & 0 \end{array}\right], \tag{5}$$

where the original inputs v, w and outputs x, u of LQG are encoded as w_2 and z_2 and recovered from the relations

$$\begin{bmatrix} w \\ v \end{bmatrix} = \begin{bmatrix} W^{1/2} & 0 \\ 0 & V^{1/2} \end{bmatrix} w_2, \qquad z_2 = \begin{bmatrix} Q^{1/2} & 0 \\ 0 & R^{1/2} \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}.$$

As a consequence the optimal LQG controller then has the observer structure (4).

3 Loop transfer recovery

We continue with a rapid flashback on the LQG/LTR procedure [1, 22]. Using the embedding $P_{\mathrm{LQG}} \to P$, we interpret the situation in the context of H_2 optimal control.

Along with its excellent performance $p^* = \mathcal{P}(K^*) = \|T_{w_2 \to z_2}(P, K^*)\|_2$, the optimal LQG controller K^* may be highly sensitive and therefore lack robustness with respect to system uncertainty. This is where the LQG/LTR procedure sets in. In its so-called input-sensitivity form it provides a one-parameter family of observer-based controllers

$$K(\rho) = \left[\begin{array}{c|c} A - B_2 K_c - K_f(\rho) C_2 & K_f(\rho) \\ \hline -K_c & 0 \end{array}\right],$$

indexed by $0 < \rho \leq 1$, such that

(i) $K(\rho)$ is the LQG controller of the modified LQG plant

$$P_{\mathrm{LQG}}(\rho) = \left[\begin{array}{c|cc|c} A & \left(\Gamma W \Gamma^\top\right)^{1/2} & 0 & B \\ \hline Q^{1/2} & 0 & 0 & 0 \\ 0 & 0 & 0 & R^{1/2} \\ \hline C & 0 & \rho^{1/2} V^{1/2} & 0 \end{array}\right], \tag{6}$$

the nominal case (5) being $\rho = 1$. In particular, $K^* = K(1)$. Explicitly

$$K(\rho) = -K_c \left(sI - (A - BK_c - K_f(\rho)C)\right)^{-1} K_f(\rho). \tag{7}$$

(ii) As $\rho \to 0$, the LTR controller $K(\rho)$ gets less and less sensitive in so far as the H_∞ norm of the LQG sensitivity function $S(G, K(\rho)) = (I + K(\rho)G)^{-1}$ approaches the H_∞ norm of the so-called target sensitivity function $S_{\mathrm{LQ}} = (I + K_c G_{\mathrm{LQ}})^{-1}$, which has provable good gain and phase margins [16]. Here

$$G(s) = C(sI - A)^{-1}B, \qquad S_{\mathrm{LQ}} = \left(I + K_c(sI - A)^{-1}B\right)^{-1} = (I + K_c G_{\mathrm{LQ}})^{-1}.$$

(iii) $\rho^{1/2} K_f(\rho) \to \Gamma V^{-1/2}$ as $\rho \to 0$, so $K(\rho)$ has no limit in controller space as $\rho \to 0$. In consequence, performance of $K(\rho)$ degrades in the sense that $\mathcal{P}(K(\rho)) = \|T_{w_2 \to z_2}(P, K(\rho))\|_2 \to \infty$ as $\rho \to 0$, where P is the nominal plant (5).

Altogether the family of LTR controllers $K(\rho)$ in (7) represents a trade-off between performance (3) with respect to the original LQG plant (5), and robustness with respect to the input sensitivity function $S(G, K) = (I + KG)^{-1}$. Each $K(\rho)$ is conveniently obtained by solving a modified LQG synthesis program based on (6). The procedure presented leaves K_c fixed and adapts the Kalman filter gain $K_f(\rho)$ to the noise level ρV.

Remark 1. A variant of the described LTR procedure is obtained by fixing $V = V_0$ and letting $W = W_0 + \rho^{-1} BB^T$, where W_0 is nominal.

The central theme is now the following: We wish to improve robustness $\|S(G, K)\|_\infty \to \|S_{\mathrm{LQ}}\|_\infty =: r_*$ just as in LTR, but at the same time we want to avoid the loss of performance $\mathcal{P}(K(\rho)) \to \infty$ caused by the LTR controller.

4 Improved LQG/LTR procedure

In order to emphasize the terms performance and robustness, we will use the notation

$$\mathcal{P}(K) = \|T_{w_2 \to z_2}(P, K)\|_2, \quad \mathcal{R}(K) = \|S(G, K)\|_\infty.$$

As we have seen, $\mathcal{R}(K(\rho)) \to r_* := \|S_{\mathrm{LQ}}\|_\infty$, while $\mathcal{P}(K(\rho)) \to \infty$ when $\rho \to 0$. Let $r^* = \mathcal{R}(K^*)$ be the robustness level of the nominal H_2 controller K^*. Since K^* is too sensitive with regard to $S(G, K)$, we know that the value r^* is too large. Fortunately, for every intermediate value r with $r_* < r \leq r^* = \mathcal{R}(K^*)$, there exists some $\rho \in (0, 1]$ such that $r = r(\rho) = \mathcal{R}(K(\rho))$.

Naturally, we seek a compromise $r = r(\rho)$ somewhere in between these two extrema r_*, r^*. This is where LTR has its limitation. It can only propose to *stop* at some $K(\rho)$ where $r = r(\rho)$ is as desired, but it can then no longer influence the corresponding performance $p(\rho) = \mathcal{P}(K(\rho))$. The value $p(\rho) := \mathcal{P}(K(\rho))$ is just somewhere in between $p^* = \mathcal{P}(K^*)$ and $p_* = \infty$ and has to be accepted as such. *But we can do better.* Having identified the appropriate robustness level $r = r(\rho) = \mathcal{R}(K(\rho))$ of the LTR controller $K(\rho)$, we consider the mixed H_2/H_∞ optimization program

$$(8) \qquad (P_\rho) \quad \begin{array}{ll} \text{minimize} & \mathcal{P}(K) = \|T_{w_2 \to z_2}(P, K)\|_2 \\ \text{subject to} & \mathcal{R}(K) = \|S(G, K)\|_\infty \leq r(\rho) \\ & K \text{ has observer structure (4)} \end{array}$$

whose decision variable is $\mathbf{x} = (\mathrm{vec}(K_c), \mathrm{vec}(K_f))$. For short, we go beyond LTR and optimize performance $\mathcal{P}$ subject to the robustness constraint $\mathcal{R} \leq r(\rho)$. For the following, we will denote the solution of (8) as $K_{2,\infty}(\rho)$, indicating that a mixed H_2/H_∞ controller synthesis problem is solved, where the robustness level $r(\rho) = \mathcal{R}(K(\rho))$ is imposed as a constraint, $K(\rho)$ being the LQG/LTR controller (7) with parameter ρ. Program (8) is the key element of the following

Algorithm I

1: **Initialize.** Synthesize the nominal LQG controller K^* and compute its robustness $r^* = \mathcal{R}(K^*) = \|S(G, K^*)\|_\infty$. If r^* is small enough, meaning that K^* is sufficiently robust, then quit. Otherwise continue.

2: **Calibrate.** Compute LTR controller $K(\rho)$ so that robustness $r(\rho) := \|S(G, K(\rho))\|_\infty < r^*$ is small enough. A lower bound is $r_* = \|S(G_{\mathrm{LQ}}, K_c)\|_\infty$.

3: **Optimize.** Solve mixed H_2/H_∞ program $(P_{r(\rho)})$, using $K(\rho)$ as initial guess. The locally optimal solution is $K_{2,\infty}(\rho)$.

4: **Evaluate.** Check whether $K_{2,\infty}(\rho)$ is acceptable. If it is not sufficiently robust use smaller ρ to get a smaller $r(\rho)$. If it is too robust and not sufficiently performing, use larger ρ to get a larger $r(\rho)$. Then go back to step 3.

Remark 2. Notice that in (8) the Kalman gain K_f and the state feedback gain K_c are optimized *simultaneously.* The principle of separation of observation and control is no longer valid here. In particular, the optimal K_c, K_f are no longer characterized by AREs. Nonetheless $K_{2,\infty}(\rho)$ is an observer-based controller.

Remark 3. The fact that the $r(\rho)$ cover the range $(r_*, r^*]$ does not mean that $r(\rho) \in (r_*, r^*]$ for all ρ. Typically, for ρ close to the nominal value 1 it may happen that $r(\rho) > r^*$. This means LTR

is not a monotone procedure, as can be seen e.g. in Figure 3. Naturally, the ρ with $r(\rho) > r^*$ are of no use in Algorithm I.

The central property of the solution $K_{2,\infty}(\rho)$ of (8) is the following

Proposition 1. *The optimal H_2/H_∞ controller $K_{2,\infty}(\rho)$ computed in step 3 of Algorithm I is as robust as the LTR controller $K(\rho)$: $\|S(G, K_{2,\infty}(\rho))\|_\infty = \|S(G, K(\rho))\|_\infty$, but it has better performance: $\mathcal{P}(K_{2,\infty}(\rho)) \leq \mathcal{P}(K(\rho))$.*

Proof: Notice that the constraint $\mathcal{R} \leq r(\rho)$ in (8) must be active, because otherwise $K_{2,\infty}(\rho)$ would be an unconstrained local minimum of the nominal H_2 program (3). But program (3) is in fact strictly convex and its unique solution is the LQG controller K^*. Hence we would have $r^* = \mathcal{P}(K^*) < \mathcal{P}(K(\rho)) = r(\rho)$. However, according to step 2 of algorithm I, ρ is such that $r_* < r(\rho) \leq r^*$ and values ρ with $r(\rho) > r^*$ are not considered. This shows that the constraint is active.

The second claim, the improvement of the performance, is due to the fact that $K(\rho)$ is a feasible point in (8), and that we start optimization at $K(\rho)$. This assures that the (locally) optimal solution has a lower objective value $\mathcal{P}(K_{2,\infty}(\rho)) \leq \mathcal{P}(K(\rho))$. □

Remark 4. What we have shown is that $K_{2,\infty}(\rho)$ is always preferable to the LQG/LTR controller $K(\rho)$ with regard to the chosen criteria $\mathcal{P}, \mathcal{R}$. Therefore the LTR controller should not be accepted as the final solution of the process, but $K(\rho)$ remains a very useful tool to calibrate the procedure, and it also serves as starting point for the optimization program (8).

Remark 5. Graphical visualization. One can compare the LTR controller $K(\rho)$ and the mixed controller $K_{2,\infty}(\rho)$ graphically as shown in Figures 3 and 9. We display $\mathcal{P}(K(\rho))$ and $\mathcal{P}(K_{2,\infty}(\rho))$ over the same abscissa $\log(\rho)$. The matched robustness level of the two controllers $r(\rho) = \mathcal{R}(K(\rho)) = \mathcal{R}(K_{2,\infty}(\rho))$ is also displayed over the same logarithmic scale.

Remark 6. It is well-known that the optimal H_2 controller K^* has the observer-based structure (4) if no additional control law specifications are imposed. In that sense the observer-based structure is *optimal* for H_2 control. This is no longer the case in (8). If we wish to obtain an observer-based solutions, we have to impose (4) as a constraint, as we do. Otherwise the H_2/H_∞ program has an infinite dimensional solution [12], which need not even be realizable. And even when we impose realizability as the sole structural constraint, the optimal solution need not be observer-based.

Remark 7. Mixed H_2/H_∞-programs had originally been proposed by Bernstein and Haddad [10], who characterize the solution in the full-order case (in the absence of constraint (4)) by a system of coupled algebraic Riccati equations. A homotopy method is proposed to compute the solutions. The first numerically efficient way to solve (8) with the constraint (4) was presented in [8] and is based on nonsmooth optimization techniques. Notice that program (8) is no longer convex due to the structural constraint on K.

5 Other LTR procedures

There exists a dual LTR procedure, which generates a family $K(q)$ of LQG controllers parametrized by $q \geq 0$ such that $K(0) = K^*$, and such that $K(q)$ now gets less sensitive as $q \to \infty$. Consider

the deformed LQG system

$$P(q): \left[\begin{array}{c|cc|c} A & \left(\Gamma W \Gamma^\top\right)^{1/2} & 0 & B \\ \hline Q^{1/2}(q) & 0 & 0 & 0 \\ 0 & 0 & 0 & R^{1/2} \\ \hline C & 0 & V^{1/2} & 0 \end{array}\right],$$

where $Q(q) = Q + qC^\top C$, and $q = 0$ corresponds to the nominal case (5). The LQG/LTR controller is then obtained by an LQG synthesis for $P(q)$ and has the form

$$K(q) = \left[\begin{array}{c|c} A - B_2 K_c(q) - K_f C_2 & K_f \\ \hline -K_c(q) & 0 \end{array}\right] \tag{9}$$

where now K_f is fixed and $K_c(q)$ tuned. Limiting results now hold with respect to the output sensitivity function $\widetilde{S}(G, K) = (I + GK)^{-1}$. Namely $\|\widetilde{S}(G, K(q))\|_\infty \to \|\widetilde{S}_{\mathrm{LQ}}\|_\infty$, where $\widetilde{S}_{\mathrm{LQ}} = \left(I + C(sI - A)^{-1} K_f\right)^{-1} = (I + G_{\mathrm{LQ}} K_f)^{-1}$, which again has guaranteed good margins as $q \to \infty$.

Remark 8. Notice that $K(q)$ is obtained by artificially increasing the cost term $x^\top Q x$ in the LQG objective, replacing the nominal Q by $Q + q\, C^\top C$. As $q \to \infty$ increases, this obviously forces the trajectories $x(t)$ to decay faster to 0 as $t \to \infty$, hence a gain in robustness. In [23] a variant is discussed, where in the cost term $x^\top Q x + \mu u^\top R u$ the parameter μ is driven to zero.

The new type of controller $K_{2,\infty}(q)$ associated with the family $K(q)$ is constructed as follows. Let $\widetilde{r}(q) = \|\widetilde{S}(G, K(q))\|_\infty$, solve the mixed H_2/H_∞ program

$$\begin{array}{ll} \text{minimize} & \mathcal{P}(K) = \|T_{w_2 \to z_2}(P, K)\|_2 \\ \text{subject to} & \mathcal{R}(K) = \|\widetilde{S}(G, K)\|_\infty \leq \widetilde{r}(q) \\ & K \text{ observer based} \end{array} \tag{10}$$

and let $K_{2,\infty}(q)$ be the solution of (10). The link between the dual LQG/LTR controller $K(q)$ and its associated H_2/H_∞ controller $K_{2,\infty}(q)$ is the following

Proposition 2. *The mixed H_2/H_∞ controller $K_{2,\infty}(q)$ is as robust as the LQG/LTR controller $K(q)$ in the sense that $\|\widetilde{S}(G, K_{2,\infty}(q))\|_\infty = \|\widetilde{S}(G, K(q))\|_\infty$, but it has better performance.* □

Remark 9. It is straightforward to propose an algorithm similar to algorithm I based on (10). We leave the details to the reader.

6 Trade-off with performance certificate

There is a second approach to (8), which can be interpreted as setting aside some of the good performance in order to buy some robustness. Suppose the unconstrained H_2 program has $p^* = \mathcal{P}(K^*)$, where K^* solves (3). We call p^* the nominal performance. As soon as K^* is overly sensitive and lacks robustness, we expect p^* to be too small. Assuming that we are working with the sensitivity function $\mathcal{R}(K) = \|S(G, K)\|_\infty$, we can therefore consider the following mixed H_∞/H_2 program

$$\begin{array}{lll} & \text{minimize} & \mathcal{R}(K) = \|S(G, K)\|_\infty \\ (D_\alpha) & \text{subject to} & \mathcal{P}(K) = \|T_{w_2 \to z_2}(P, K)\|_2 \leq (1 + \alpha) p^* \\ & & K \text{ has observer structure (4)} \end{array} \tag{11}$$

where we accept a loss of $\alpha\%$ in nominal performance p^* and use this freedom to buy as much robustness as possible. Let $K_{\infty,2}(\alpha)$ be a locally optimal solution of (D_α).

It turns out that there is a close relationship between programs (P_ρ) and (D_α).

Proposition 3. *Let $K_{2,\infty}(\rho)$ be a locally optimal solution of (P_ρ). Then there exists $\alpha = \alpha(\rho)$ such that $K_{2,\infty}(\rho) = K_{\infty,2}(\alpha(\rho))$, i.e., $K_{2,\infty}(\rho)$ is also a locally optimal solution of a suitable program $(D_{\alpha(\rho)})$. One simply has to set $\alpha(\rho) := [\mathcal{P}(K_{2,\infty}(\rho)) - p^*]/p^*$.*

Conversely, let $K_{\infty,2}(\alpha)$ be a locally optimal solution of (D_α), where the constraint $\mathcal{P} \leq (1+\alpha)p^*$ is active. Then $K_{\infty,2}(\alpha) = K_{2,\infty}(\rho(\alpha))$ for a suitable $\rho = \rho(\alpha)$, i.e., $K_{\infty,2}(\alpha)$ is also a locally optimal solution of $(P_{\rho(\alpha)})$. One simply puts $\rho(\alpha) = \mathcal{R}(K_{\infty,2}(\alpha))$.

Proof: It suffices to compare the necessary optimality conditions of both programs. □

Remark 10. While programs (P_ρ) and (D_α) are at least locally in one-to-one correspondence via $\rho \mapsto \alpha(\rho)$ and $\alpha \mapsto \rho(\alpha)$, it is beneficial to have both casts at our disposition. For instance, in some cases it may be easier to calibrate the value α, i.e. the accepted loss of performance, than to guess an appropriate ρ in (P_ρ). On the other hand, LTR can be used more directly to calibrate the procedure in the primal approach based on P_ρ. Notice, however, a difference between (D_α) and (P_ρ). In (D_α) it may happen that the constraint $\mathcal{P} \leq (1+\alpha)p^*$ is inactive. In that case we have found a local minimum of the robustness function $\mathcal{R}$ alone. This is possible, because the H_∞-program $\min\{\|S(G,K)\|_\infty : K \text{ observer based}\}$ is not a convex program and may therefore have locally optimal solutions.

7 Extension to more general controller structures

In this section we propose a variant of algorithm I which applies to more general controller structures. In Section 8.2 we will apply it to PID controllers.

We call a controller in state-space form (2) *structured* if the matrices A_K, B_K, C_K, D_K depend smoothly on a design parameter vector $\mathbf{x}$, that is

$$A_K = A_K(\mathbf{x}), B_K = B_K(\mathbf{x}), C_K = C_K(\mathbf{x}), D_K = D_K(\mathbf{x}).$$

We assume that $\mathbf{x}$ varies in some parameter space $\mathbb{R}^n$, or in a constrained subset of $\mathbb{R}^n$. Here $n = \dim(\mathbf{x})$ is typically smaller than $\dim(K) = n_K^2 + m_2 n_K + p_2 n_K + m_2 p_2$, where m_2 is the number of inputs, p_2 the number of outputs, n_K the order of K. We also expect $n_K \ll n_x$, even though this is not formally imposed, so that the full order controller is admitted. We usually refer to full order controllers as *unstructured*.

We have already encountered a controller structure, namely observer-based controllers, where $\mathbf{x} = (\mathrm{vec}(K_c), \mathrm{vec}(K_f)) \in \mathbb{R}^{n_x m_2 + n_x p_2}$. Other useful controller structures are for instance reduced-order controllers ($n_K \ll n_x$), decentralized, or PID controllers. For PIDs the structure is:

$$K_{\mathrm{pid}}(\mathbf{x}) = \left[\begin{array}{cc|c} 0 & 0 & R_i \\ 0 & -\tau I_{m_2} & R_d \\ \hline I_{m_2} & I_{m_2} & D_K \end{array}\right], \tag{12}$$

where $\mathbf{x} = (\tau, \mathrm{vec}(R_i), \mathrm{vec}(R_d), \mathrm{vec}(D_K))$ has $\dim(\mathbf{x}) = 3m_2p_2 + 1$, and a constraint $\tau \geq \epsilon$ (for some small $\epsilon > 0$) is typically added in parameter space.

Armed with this, we can propose the following

Algorithm II

1: **Nominal synthesis.** Compute the structured optimal H_2 controller $K(\mathbf{x}^*)$ by solving the nominal structured H_2 problem

$$\begin{array}{ll} \text{minimize} & \mathcal{P}(\mathbf{x}) = \|T_{w_2\to z_2}(P, K(\mathbf{x}))\|_2 \\ \text{subject to} & K(\mathbf{x}) \text{ internally stabilizing} \end{array} \tag{13}$$

Evaluate its sensitivity $r^* = \mathcal{R}(K(\mathbf{x}^*)) = \|S(G, K(\mathbf{x}^*))\|_\infty$. If r^* is small enough, meaning that $K(\mathbf{x}^*)$ is sufficiently robust, then quit. Otherwise continue and keep r^* as upper bound.

2: **Lower bound.** Compute the structured H_∞-optimal controller $K(\mathbf{x}_\infty)$ by solving

$$\begin{array}{ll} \text{minimize} & \mathcal{R}(\mathbf{x}) = \|S(G, K(\mathbf{x}))\|_\infty \\ \text{subject to} & K(\mathbf{x}) \text{ internally stabilizing} \end{array} \tag{14}$$

Keep $r_* = \mathcal{R}(K(\mathbf{x}_\infty))$ as lower bound. Choose $r \in [r_*, r^*]$.

3: **Optimize.** Solve the following structured mixed H_2/H_∞ program

$$\begin{array}{ll} \text{minimize} & \mathcal{P}(\mathbf{x}) = \|T_{w_2\to z_2}(P, K(\mathbf{x}))\|_2 \\ \text{subject to} & \mathcal{R}(\mathbf{x}) = \|S(G, K(\mathbf{x}))\|_\infty \le r \\ & K(\mathbf{x}) \text{ internally stabilizing} \end{array} \tag{15}$$

The locally optimal solution is $K(\mathbf{x}_{2,\infty}(r))$.

4: **Evaluate.** Check whether $K(\mathbf{x}_{2,\infty}(r))$ offers an acceptable compromise between performance and robustness. If it is not sufficiently robust, choose a smaller $r \in [r_*, r^*]$. If it is too robust and lacks performance, use larger $r \in [r_*, r^*]$. Then loop back to step 3.

The difference with algorithm I is that we no longer have LTR at our disposition to calibrate the procedure. Instead we compute the lower bound r_* in step 2 based on a structured H_∞-synthesis with objective $\mathcal{R}$. This can be obtained via the matlab function `hinfstruct` [26]. The mixed H_2/H_∞-program can be solved via [8] or using the matlab function `fmincon` [25].

In order to solve (15) efficiently, we can use the result $\mathbf{x}^*$ of step 1 or $\mathbf{x}_\infty$ of step 2. It is also possible to stop the minimization in (14) at the first $\mathbf{x}_r$ where $\mathcal{R}(\mathbf{x}_r) \le r$. This feature is indeed available in the matlab function `hinfstruct` [26]. We may then use the controller $K(\mathbf{x}_r)$ as initial guess in (15). The result extending Proposition 3 is the following

Proposition 4. *Suppose $\mathbf{x}_r$ with $\mathcal{R}(\mathbf{x}_r) = r$ is obtained as intermediate solution in step 2 of algorithm II. If $\mathbf{x}_r$ is used as initial guess in program (15), then the locally optimal solution $K(\mathbf{x}_{2,\infty}(r))$ of (15) is as robust as $K(\mathbf{x}_r)$, but has better H_2 performance.* □

Remark 11. Notice that solutions to (13), (14), and (15) can no longer be computed by algebraic Riccati equations or LMIs. While (13) can be solved by smooth optimization technique, see e.g. [20], programs (14) and (15) are non-smooth and require specific bundle techniques. (BMI solvers could at least in principle be used, but they suffer from the presence of Lyapunov variables, which lead to numerical trouble). For nonsmooth H_∞ synthesis we refer to [3], and also [2, 5, 6]. An implementation is `hinfstruct` in [26]. Constrained programs like (15) are discussed in [8, 9].

General mathematical background is given in [18, 19]. A recent approach to combine nonsmooth techniques with classical nonlinear programming techniques is discussed in [4].

8 Numerical Experiments

In this section we present two studies in which we test our trade-off models via local optimization techniques. In each study performance of the nominal system is optimized subject to the structural constraint observer-based, respectively, PID. Performance refers to the H_2 norm. The working hypothesis is that nominal synthesis is not sufficiently robust. In order to decide what sufficiently robust means, we consider in each study two uncertain parameters of the system matrix. We expect a valid solution to be robustly stable with respect to $\alpha\%$ variation in these parameters (where $\alpha = 30$ for LQG, and $\alpha = 40$ for H_2-PID). We want best *nominal* performance subject to *parametrically robust* stability in this sense. In both studies the input sensitivity function S is used to robustify the system.

8.1 LQG/LTR

Our first study uses the mass-spring system [1] of Figure 1, which can be considered as a prototype of a flexible system. We compare the LQG/LTR procedure to our H_2/H_∞ trade-off model (8) of section 4, i.e., we follow Algorithm I.

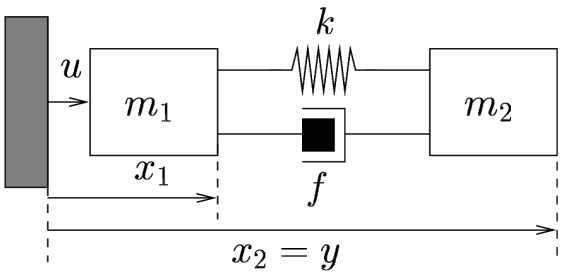

FIGURE 1: Mass-spring system. Nominal data are $m_1 = m_2 = 0.5$kg, $k = 1$N/m, $f = 0.0025$Ns/m. Measured output is $y = x_2$, control force u acts on m_1.

We start with a nominal LQG synthesis, where the covariance matrices $W = BB^\top$ and $V = 1$ are chosen, while $Q = C^\top C$ and $R = 1$; see [1]. In order to decide whether a controller is sufficiently robust, we pose the following parametric robustness goal:

- Any acceptable controller K should stabilize the system for all parameter variations of $\alpha\%$ in k and m_2 about the nominal values, ($\alpha = 30$ for LQG, $\alpha = 40$ for PID), and should optimize the nominal performance within the given structure.

As can be seen in Figure 2, **(c)**, this goal is missed for the nominal LQG controller $(K_f, K_c) = ([0.94\,0.06\,0.97\,0.75], [1.49\,1.93\,0.13\,1.87])$, whose performance is $\mathcal{P}(K_2) = 3.99$. We therefore use the LTR procedure to generate a curve $(K_f(\rho), K_c)$, which consists in keeping W fixed, and letting $V = \rho I = \rho \to 0$. This corresponds to using the input sensitivity function $\mathcal{R}(K) := S(K) = (I + KG)^{-1}$ as robustness index. The LQG/LTR controller $K(\rho)$ is used to calibrate $r(\rho) = S(G, K(\rho))$,

and for each $r(\rho)$ an optimal H_2/H_∞ controller $K_{2,\infty}(\rho)$ is synthesized using program (8). That means

$$\mathcal{R}(K(\rho)) = \mathcal{R}(K_{2,\infty}(\rho)) = r(\rho), \qquad \mathcal{P}(K_{2,\infty}(\rho)) < \mathcal{P}(K(\rho)),$$

so that $K_{2,\infty}(\rho)$ is better than $K(\rho)$. Figure 3 shows the evolution of the curves $\mathcal{P}(K(\rho))$, $\mathcal{P}(K_{2,\infty}(\rho))$ and $r(\rho)$ as $\rho \to 0$.

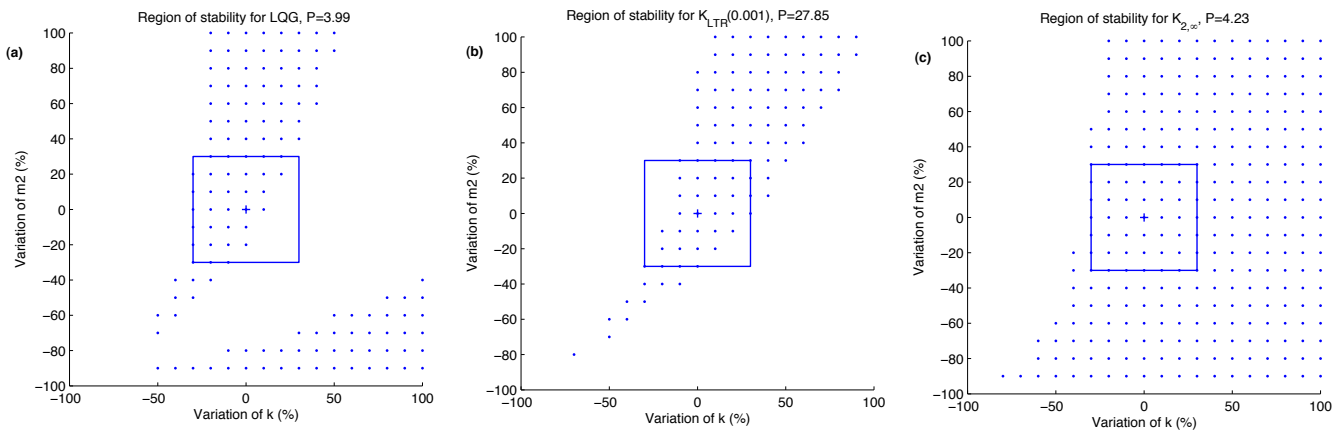

FIGURE 2: LQG/LTR study. Image **(a)** shows the stability region of the nominal H_2 controller K_2 with performance $\mathcal{P}(K_2) = 3.99$ and evident lack of robustness. Image **(b)** shows the stability region of the LTR controller at $\rho = .001$. Its performance $\mathcal{P}(K(.001)) = 27.85$ is degraded. Image **(c)** shows what the mixed H_2/H_∞ controller achieves at the same $\rho = .001$. On top of having significantly better performance $\mathcal{P}(K_{2,\infty}(.001)) = 4.23$, it has also better parametric robustness than the LTR controller.

In a second step we now identify the value $r = r(\rho)$, which gives closed-loop stability for all parameter values $k + \Delta k$, $m_2 + \Delta m_2$ over the square $|\Delta k| \leq \alpha\% k$, $|\Delta m_2| \leq \alpha\% m_2$. It turns out that in this respect $K_{2,\infty}(\rho)$ is way better than the LTR controller $K(\rho)$. For the mixed controller we achieve parametric robustness at $K_{2,\infty}(0.001)$, whereas $K(\rho)$ is still not parametrically robust at $\rho = 10^{-10}$. Nonetheless, we can observe that the region of parametric robustness of LQG/LTR

controller $K(\rho)$ also increases as ρ decreases.

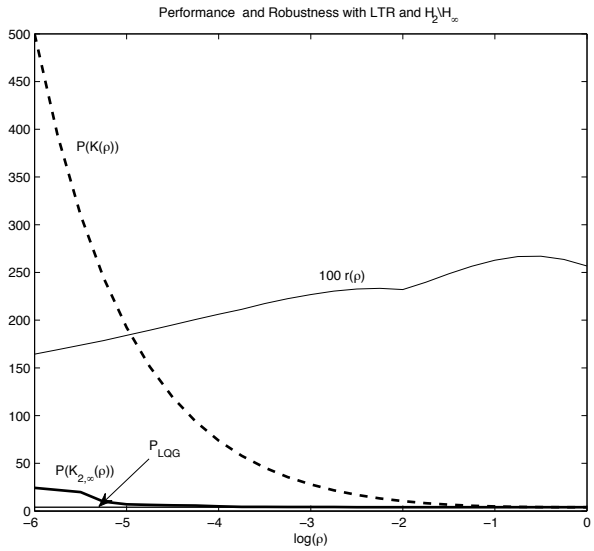

FIGURE 3: LQG/LTR study. Performance of $K(\rho)$ and $K_{2,\infty}(\rho)$ in logarithmic scale. Lower bound is the performance of the nominal LQG controller. The curve $100r(\rho)$ shows the robustness level over the same abscissa. As a by-product, we see that LTR is not a monotone procedure.

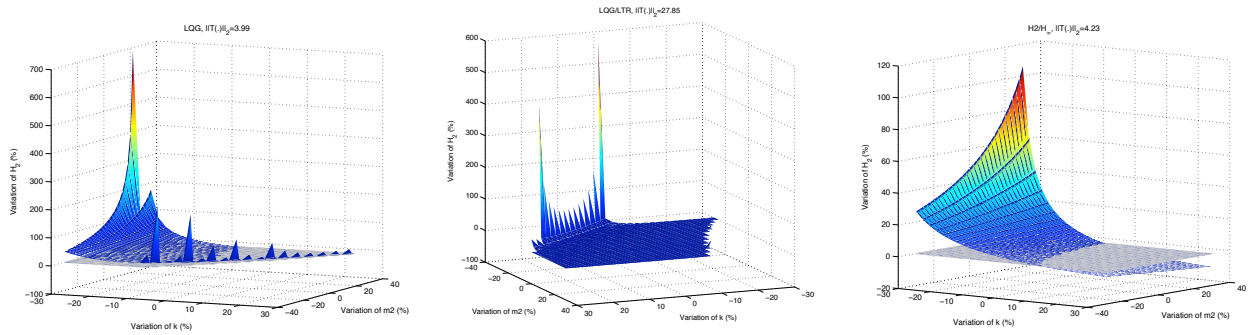

FIGURE 4: LQG/LTR study. Each graph shows relative performance $\frac{\mathcal{P}(G(k,m_2),K)-\mathcal{P}(G,K)}{\mathcal{P}(G,K)}$ for different controllers K plotted over the square $\Omega = (k^0 \pm 30\% k^0, m_2^0 \pm 30\% m_2^0)$. Left: LQG controller, middel: LQG/LTR controller, right: mixed H_2/H_∞.

8.2 H_2-optimal PID controller

In our second study we design a H_2 optimal PID controller for the mass-spring system of Figure 1 by solving the nominal program

$$\begin{array}{ll} \text{minimize} & \mathcal{P}(K) = \|T_{w\to z}(G,K)\|_2 \\ \text{subject to} & K \text{ a PID of the form (12)} \\ & K \text{ closed-loop stabilizing} \end{array} \tag{16}$$

which is a specific instance of (13). Using local optimization methods we obtain a H_2 locally optimal PID controller $K_{\text{pid},2}$ within the class of internally stabilizing PID controllers. Inspecting the same parameter variations as before, we see that $K_{\text{pid},2}$ fails to stabilize the system within the 40%-square, so we use a robustness constraint as outlined in Section 7. Since LTR is no longer available in this case, we use Algorithm II for the trade-off.

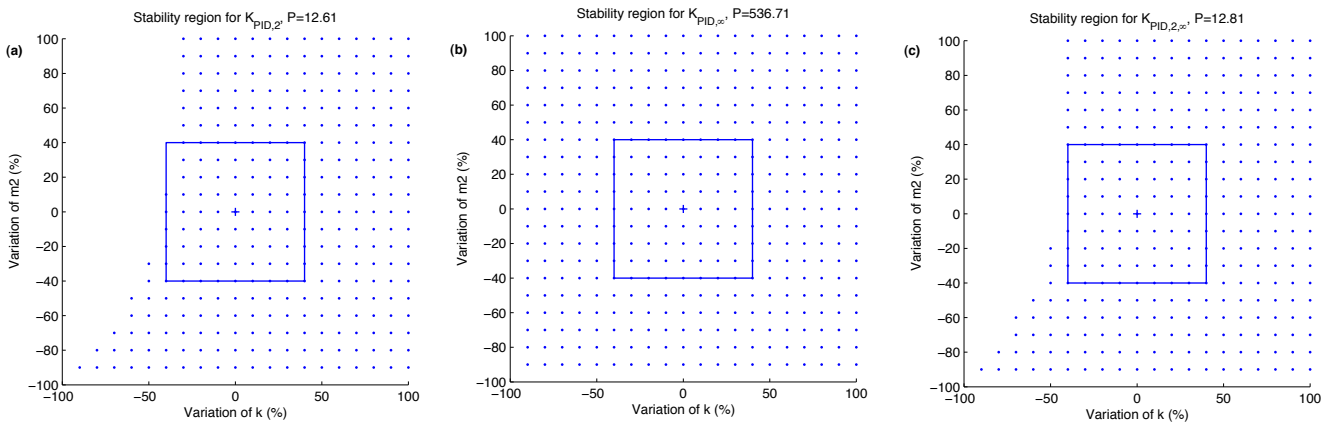

FIGURE 5: PID study. **(a)** shows nominal H_2 optimal PID controller $K_{\text{pid},2}$, which is not parametrically robust over the 40% square. Its robustness $r^* = 17.23$ it too large and gives the upper bound. The H_∞ optimal PID shown in **(b)** gives the lower robustness bound $r_* = 6.39$. Its performance $\mathcal{P}(K_{\text{pid},\infty}) = 152.8$ is degraded. The mixed H_2/H_∞ PID controller shown in **(c)** is limiting parametrically robust for $r = 17.0$.

The solution $K_{\text{pid},\infty}$ of (14) is the H_∞ optimal PID for the sensitivity objective $S(K) = (I + KG)^{-1}$. Knowing that $K_{\text{pid},2}$ performs too well and lacks robustness, we compute $r_* = \|S(K_{\text{pid},\infty})\|_\infty = 6.39$, $r^* = \|S(K_{\text{pid},2})\|_\infty = 17.23$, and then choose $r = 17.0$ in between r_* and r^*. Mixed H_2/H_∞ PID control then gives the solution $K_{\text{pid},2,\infty}$, which is as robust as $K_{\text{pid},\infty}$, but has better performance. Naturally, $K_{\text{pid},2,\infty}$ is less performing than $K_{\text{pid},2}$, but is as robust as we require, in addition to performing best subject to this constraint. Here the choice $r = 17.0$ achieves parametric robustness over the square. The result is shown in Figure 5.

9 Lateral flight control of an F-16 aircraft

In our last study we apply the improved LTR procedure to lateral flight control of an F-16 aircraft. The nonlinear F-16 lateral model was linearized using the F-16 simulation program [21]. The high fidelity model is evaluated at altitude $h = 15000\,ft$ and velocity $v = 500\,ft/s$, considering Steady Wings-Level Flight condition for trimming. The state variables are sideslip β, bank angle Φ, roll rate p and yaw rate r.

9.1 Performance channel

As in [24], state variables δ_a and δ_r representing deflection of aileron and rudder actuators are included in the model each with approximate transfer function $20.2/(s + 20.2)$. The objective of the study is to make the bank angle Φ follow a reference command r_Φ, while simultaneously keeping the sideslip angle β as close to $r_\beta = 0$ as possible. The plant has $u = [u_\Phi\ u_\beta]$ as control input and $y = [\Phi\ \beta]$ as measured output and is of type-0 with constant steady state error. To eliminate this error, the dynamics are augmented by integrators in each control channel. Moreover, to balance the singular values at dc, the system was augmented again by the inverse of the dc gain of the system [24]. The overall state vector including aircraft state variables, actuators and integrators is then

$$x = [\beta\ \Phi\ p\ r\ \delta_a\ \delta_r\ \epsilon_\Phi\ \epsilon_\beta].$$

The model for synthesis is shown in Figure 6, $G(s)$. In this figure the precompensator block represents the inverse of the dc gain. This figure also demonstrates the observer structure $K(s)$.

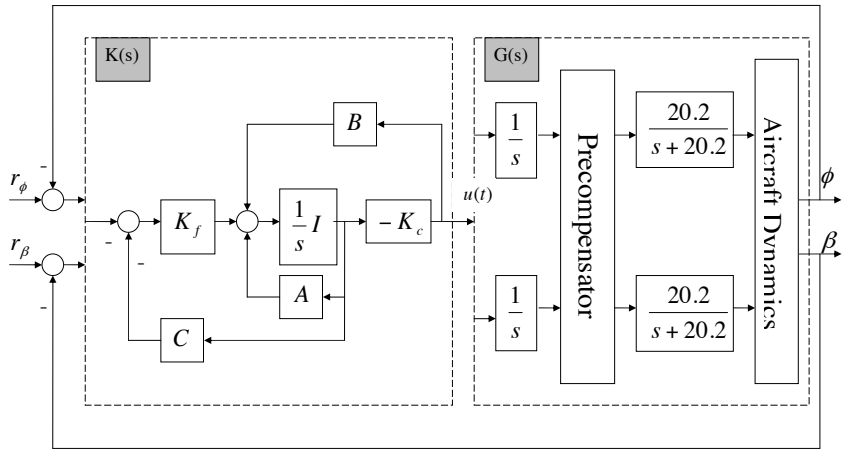

FIGURE 6: Model of F16 aircraft lateral control system and the observer structure.

9.2 LTR procedure

In this study LTR recovery at the output breaking point is used, i.e., robustness is measured via the output sensitivity function $\widetilde{S}$. Using

$$V = I_2 \text{ and } W = \text{diag}([0.1\ 0.1\ 0.1\ 0.1\ 0\ 0\ 10\ 10]) * 100,$$

we first fix the Kalman gain K_f such that the target loop gain $C(sI - A)^{-1}K_f$ has the desired performance. That this goal is achieved can be seen in the singular value plot in Figure 7, and through the step responses of Figure 8 (solid lines).

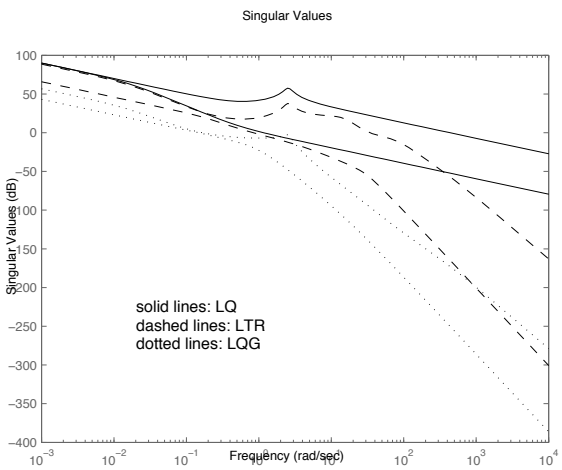

FIGURE 7: Singular values of the loop transfer function $L(s) = K(s)G(s)$ for LQ, LQG and LTR controller.

LTR is now applied with $Q = C^\top C$ and $R = \rho I_2$, where $\rho \to 0$, and $K_c(\rho)$ is tuned. With $q = 1/\rho$ this corresponds to the case discussed in Section 5. Figure 7 compares the singular values

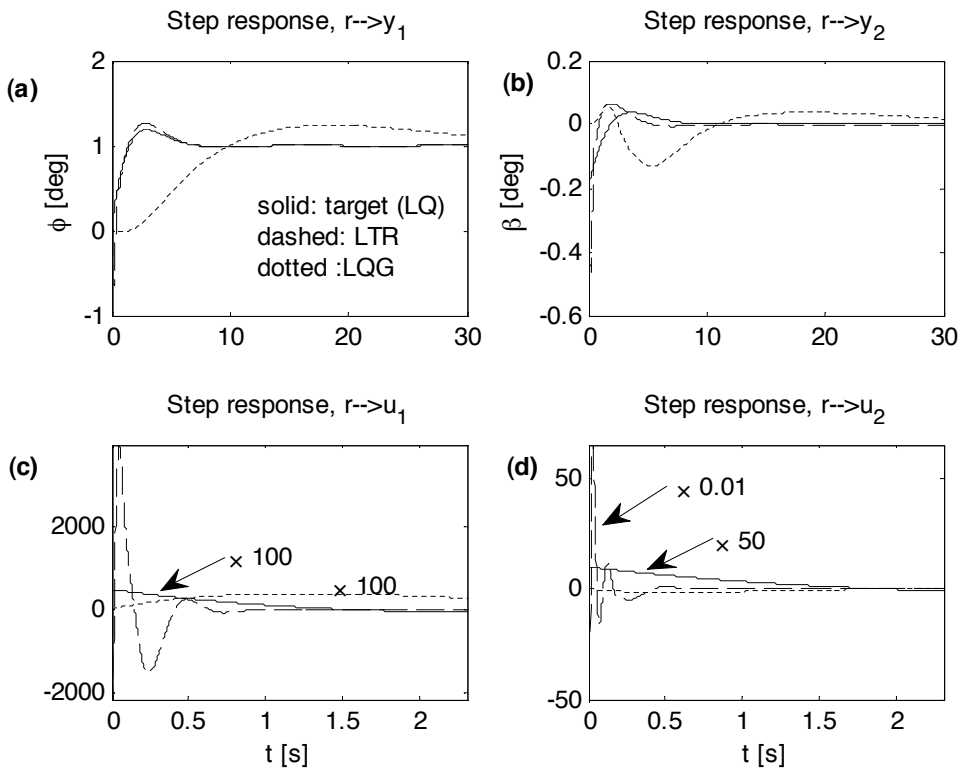

FIGURE 8: Step responses with different controllers.

$\overline{\sigma}$ and $\underline{\sigma}$ of the LQ target with those of LQG and LTR (where $C(sI-A)^{-1}K_f$ is matched with the LQG controller ($\rho = 1$) and the LTR controller $K_c(\rho)(sI-(A+BK_c(\rho)+K_f^\top C))^{-1}K_f$ at $\rho = 10^{-10}$. As can be seen, forcing $\rho \to 0$ brings the singular values of the LTR controller near those of the target. In addition, this drives the system output responses (with LTR controller) toward the model responses of the target, as shown in Figure 8 (a) and (b). In Figure 8 (c) and (d), the control input signals of LQ, LQG and LTR are compared. Unfortunately, the LTR controller causes a large control input, which results in a large (degraded) performance. This loss of performance increases with ρ^{-1} as Figure 9 shows. In the same figure the robustness index $\mathcal{R}_{\text{LTR}} = \|\widetilde{S}\|_\infty = \|(I+GK_{\text{LTR}}(\rho))^{-1})\|_\infty$ is displayed. As can be seen, at the beginning (going from right to left) $\mathcal{R}$ increases and then decreases before stabilizing around $\|(I+GK_{\text{LQ}}(\rho))^{-1})\|_\infty$. This proves that LTR with recovery at the output breaking point is not a monotone procedure either.

9.3 Mixed synthesis

We use the analogue of program (15) with $\widetilde{S}$ replacing S and for $\rho \in [10^{-4}\ 10^{-1.3}]$. In this range robustness $\mathcal{R}$ decreases monotonically with ρ, while performance $\mathcal{P}$ increases. Figure 9 compares performance after matching robustness of the H_2/H_∞ and LTR controllers. A substantial improvement in performance can be observed.

We check the efficiency of our method by considering changes of the flight parameters. We consider $h = h_0 \pm \Delta h$ and $v = v_0 \pm \Delta v$, with $|\Delta h| \leq 1000\, ft$ and $|\Delta v| \leq 25\, ft/s$, the nominal flight point being $h_0 = 15000\ ft$ and $v_0 = 500\ ft/s$. The LTR controller and the corresponding mixed

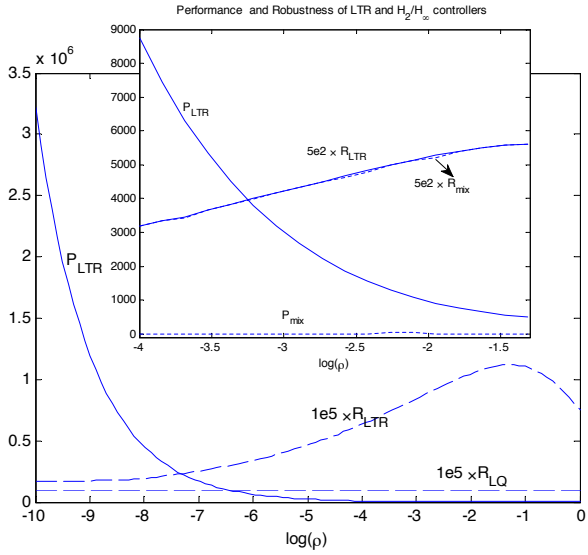

FIGURE 9: F-16 study. Comparison of performance of LTR and H_2/H_∞ controller when robustness according to $\mathcal{R}(K) = \|\widetilde{S}(G,K)\|_\infty$ are matched.

controller are evaluated at $\rho = 1.438e-4$. Figure 10 (a) and (b) compares the first output and the first control input of the 8 neighbouring flight points around the nominal flight point. The diagram in Figure 10 (c) shows the improvement in performance obtained with the mixed controller.

10 Conclusion

We have used mixed H_2/H_∞ synthesis for structured control laws to obtain a quantified trade-off between performance and robustness in feedback control design. Within the class of observer-based controllers this leads to an improvement of the LQG/LTR procedure. The latter is still useful to calibrate and initialize the procedure. For other controller structures a different idea is used to calibrate the mixed program. The new method was applied to a mass-spring benchmark example and also to lateral flight control of an F-16 aircraft. Experiments indicate that the new technique can also be useful to enhance the parametric robustness of a design. In our tests the achieved degree of parametric robustness was satisfactory.

Acknowledgement

This work was supported by research grant *Survol* from Fondation de Recherche pour l'Aéronautique et l'Espace (FNRAE) and *Technicom* from Fondation d'Entreprise EADS.

References

[1] D. ALAZARD, C. CUMER, P. APKARIAN, M. GAUVRIT, G. FERRERES. *Robustesse et Commande Optimale*. CEPADUES Edition, 2000.

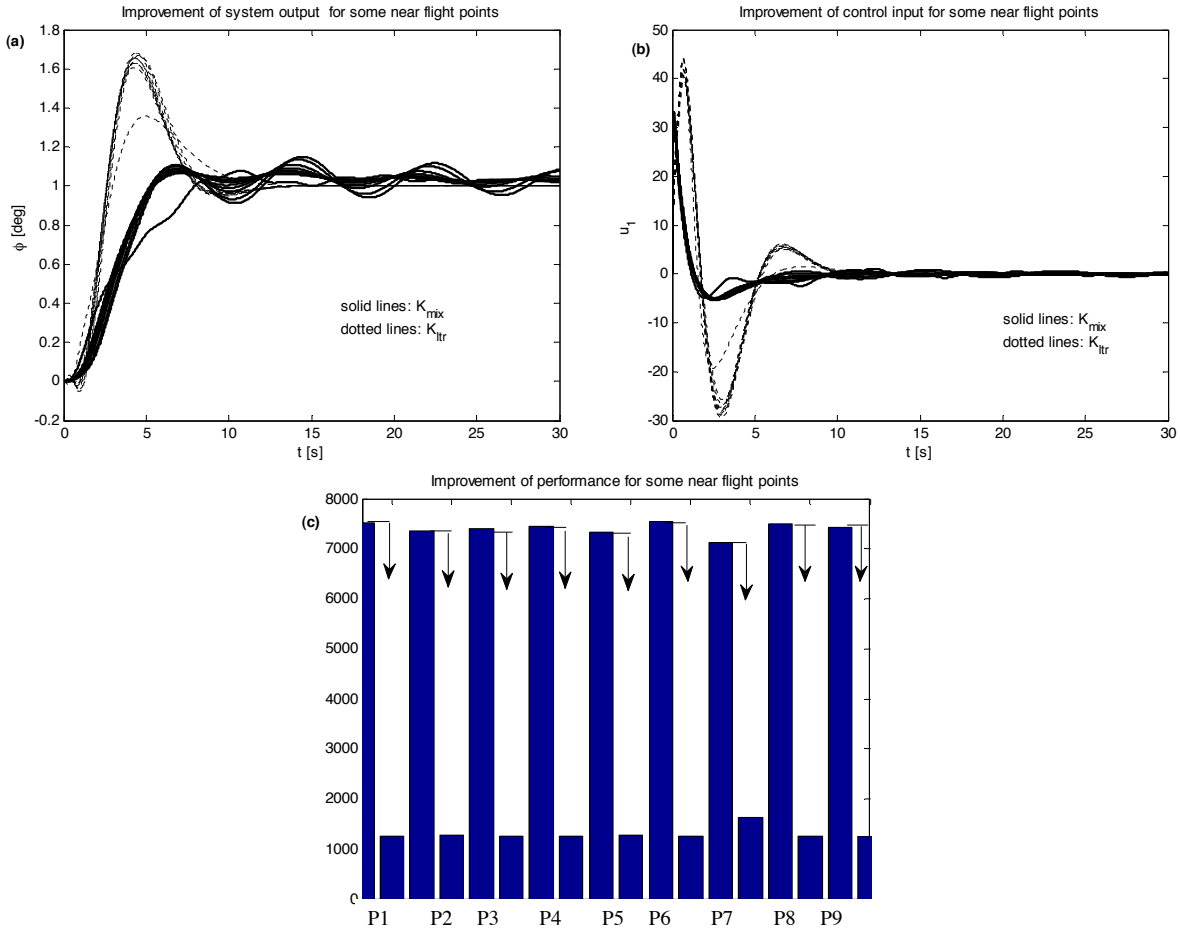

FIGURE 10: Comparing LTR and H_2/H_∞ controllers against the system variations, (a) and (b) step responses, (c) performance.

[2] P. Apkarian, D. Noll, *Nonsmooth optimization for multidisk H_∞ synthesis.* European J. Control, vol. 12, no. 3, 2006, pp. 229 - 244.

[3] P. Apkarian, D. Noll, *Nonsmooth H_∞ synthesis.* IEEE Trans. Autom. Control, vol. 51, 2006, pp. 71 - 86.

[4] P. Apkarian, L. Ravanbod-Hosseini, D. Noll. *Time domain constrained H_∞ synthesis.* Int. J. of Rob. and Nonlin. Contr., vol. 21, no. 2, 2011, pp. 197 – 217.

[5] P. Apkarian, D. Noll, O. Prot. *A trust region spectral bundle method for nonconvex eigenvalue optimization.* SIAM Journal on Optimization, vol. 19, no. 1, 2008, pp. 281 – 306.

[6] P. Apkarian, D. Noll, O. Prot. *A proximity control algorithm to minimize nonsmooth and nonconvex semi-infinite maximum eigenvalue functions.* Journal of Convex Analysis, vol. 16, no. 3 & 4, 2009, pp. 641 – 666.

[7] P. Apkarian, D. Noll, O. Prot. *A non-smooth IQC method for robust synthesis.* Proc. 46th IEEE CDC, New Orleans, 2007, pp. 824 – 829.

[8] P. Apkarian, D. Noll, A. Rondepierre. *Mixed H_2/H_∞ control via nonsmooth optimization.* SIAM Journal on Control and Optimization, vol. 47, no. 3, 2008, pp. 1516-1546.

[9] P. Apkarian, D. Noll, A. Simões. *A nonsmooth progress function algorithm for frequency shaping control design.* IET Control Theory & Applications, vol. 2, no. 4, 2008, pp. 323 - 336.

[10] D.S. Bernstein, W.M. Haddad. *LQG control with an H_∞ performance bound: A Riccati equation approach.* IEEE Trans. Autom. Control, vol. 34, no. 3, 1989, pp. 683 – 688.

[11] S. Boyd, C.H. Barratt. Linear controller design. Limits of performance. Prentice Hall 1991.

[12] S.M. Djouadi, C.D. Charalambous, D.W. Repperger. *On multiobjective H_2/H_∞ optimal control.* Proc. Amer. Contr. Conf., Arlington, june 2001, pp. 4091 – 4096.

[13] J. Doyle, K. Glower, P.P. Kargonekar, B. Francis. *State-space solutions to the standard H_2 and H_∞ control problems.* IEEE Trans. Autom. Control, vol. 34, no. 8, 1989, pp. 831 – 847.

[14] J. Doyle, G. Stein, *Robustness with observers*, IEEE Transactions on Autom. Control, vol. AC-24, 1979, pp. 607 – 611.

[15] J. Doyle, G. Stein. *Multivariable feedback design: concepts for a classical/modern synthesis.* IEEE Transactions on Automatic Control, vol. AC-26, no. 1, 1981, pp. 4 – 16.

[16] J. Doyle. *Guaranteed margins for LQG regulators.* IEEE Trans. on Auto. Control, vol. 23, no.4, 1978, pp. 756-757.

[17] H. Kwakernaak. *Optimal low-sensitivity linear feedback systems.* Automatica, vol. 5, 1969, pp. 279 – 285.

[18] D. Noll, P. Apkarian, *Spectral bundle method for nonconvex maximum eigenvalue functions: first-order methods.* Math. Programming, Series B, vol. 104, 2005, pp. 701 - 727.

[19] D. Noll, O. Prot, A. Rondepierre. *A proximity control algorithm to minimize nonsmooth and nonconvex functions.* Pacific Journal of Optimization, vol. 4, no. 3, 2008, pp. 569-602.

[20] T. Rautert, E. Sachs. *Computational design of optimal output feedback controllers.* SIAM J. Optim., vol. 7, no. 3, 1997, pp. 837 – 852.

[21] R.S. Russell. *Nonlinear F-16 simulation using Simulink and Matlab.* Technical Report, Department of Aerospace Engineering and Mechanics, University of Minnesota. 2003. `http://www.aem.umn.edu/people/faculty/balas/darpa_sec/SEC.Software.html`

[22] A. Saberi, Ben M. Chen, P. Sannuti. *Loop transfer recovery: analysis and design.* Communication and Control Engineering Series, Springer Verlag 1993.

[23] G. Stein, M. Athans. *The LQG/LTR Procedure for Multivariable Feedback Control Design.* IEEE Transaction on Automatic Control, AC-32, (1987), pp. 105114.

[24] B. Stevens, F. Lewis. *Aircraft control and simulation.* New York: Wiley 1992.

[25] The Optimization Toolbox, The MathWork Inc., Natick, MA.

[26] The Robust Control Toolbox, The MathWorks Inc., Natick, MA.