An extension of the LQG-LTR procedure

L. Ravanbod*† 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. Here an extension is discussed which avoids the usual loss of performance in LTR, and which is also applicable for non-minimum phase systems. It is also shown how the idea can be extended to other control structures. In particular, it is shown how PID controllers can be desensibilized with this new approach. The method is tested on several examples, including in particular the lateral flight control of an F-16 aircraft.

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

1 Introduction

It became apparent during the late 1960s that LQG controllers often lack robustness with regard to system uncertainty. In 1966 Kwakernaak [1] 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 [2], Doyle and Stein [3,4]. Even today LQG-LTR is still used by practitioners to desensibilize LQG controllers to enhance the robustness of a design.

Unfortunately, LTR has three main limitations. Firstly, the price for the enhanced robustness may be a considerable loss of performance. Secondly, LTR is limited to controllers with observer structure. And thirdly, its application to non-minimum phase systems is not obvious. Here we propose a new method, which avoids these difficulties. Our new approach can be cast as a constraint optimization program offering a trade-off between performance and robustness

$$(1)\qquad \begin{array}{ll} \text{minimize} & \mathcal{P}(K) \\ \text{subject to} & \mathcal{R}(K) \leq r \\ & K \text{ structured controller} \end{array}$$

where $\mathcal{P}(K)$ is the performance of the closed-loop system, expressed by an H_2 norm, while $\mathcal{R}(K)$ is the robustness, represented by a possibly frequency weighted H_∞ norm of the input or output

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

†L. Ravanbod-Hosseini (corresponding author) and D. Noll are with Université Paul Sabatier, Institut de Mathématiques, 118 route de Narbonne, 31062 Toulouse, France, Tel: 0033 5 61 55 86 26; LalehRavanbod@yahoo.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

sensitivity function $\|(I+KG)^{-1}\|_\infty$ or $\|(I+GK)^{-1}\|_\infty$. The crucial point is to choose the degree of robustness r in the constraint in such a way that a satisfactory compromise is achieved. As we shall show, for minimum phase systems and observer-based controllers, the LQG-LTR procedure allows to calibrate r in (1) in a natural way. The mixed H_2/H_∞-controller obtained by solving (1) is then as robust as the corresponding LQG-LTR controller, but has better performance.

In the case of non-minimum phase systems program (1) remains fully in effect. What needs to be modified is the LQG-LTR procedure, at least if one still wishes to use it to calibrate r. This can be done e.g. by working with frequency weighted sensitivity functions. For more details see [5] and special issue on loop transfer recovery of the International Journal of Robust and Nonlinear control, especially [6]. In [2] it is also 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.

For more general controller structures program (1) can be used in much the same way, but one needs a new way to calibrate the robustness parameter r in the constraint. We present a general method which provides a range $[r_*, r^*]$ in which the parameter r should be chosen. The validity of our method is tested for the PID controller structure.

The structure of the paper is as follows. In Sections 2 and 3 the essential features of LQG-LTR are recalled, presented for the case of the input loop breaking point. The improved LTR procedure 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, and a new procedure to calibrate r is introduced. Experiments are presented in Section 8.

2 Preparation

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

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

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

$$(3)\qquad K:\begin{bmatrix}\dot{x}_K\\ u\end{bmatrix}=\left[\begin{array}{c|c}A_K & B_K\\ \hline C_K & D_K\end{array}\right]\begin{bmatrix}x_K\\ y\end{bmatrix}$$

which stabilizes P in closed loop and minimizes the H_2 norm (cf. [7])

$$(4)\qquad \min_K \|T_{w_2\to z_2}(P,K)\|_2$$

of the closed-loop performance channel $w_2 \to z_2$. We call $\mathcal{P}(K)=\|T_{w_2\to z_2}(P,K)\|_2$ the performance of the closed-loop system. It is well known that the optimal solution K^* of (4) has observer-based structure

$$(5)\qquad K^*=\left[\begin{array}{c|c}A-B_2K_c-K_fC_2 & K_f\\ \hline -K_c & 0\end{array}\right],$$

and that K_f, K_c can be computed via AREs or LMIs [8]. In order to assure the existence of K^* we use standard assumption like (i) – (iv) on page 384 of [8], or (A1) – (A5) on page 387 of [9], which include stabilizability and detectability of the plant (2).

It is convenient to consider LQG control as a special case of H_2 synthesis. Following [9], consider the LQG problem

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

where w and v are white noise with covariance matrices W and V, respectively. Let $Q = Q^\top \succeq 0$ and $R = R^\top \succ 0$ and build a plant of form (2) by setting

$$P_{\mathrm{LQG}} = \left[\begin{array}{c|c|c} A & B_2 & B \\ \hline C_2 & 0 & D_{2u} \\ \hline C & D_{y2} & 0 \end{array}\right] = \left[\begin{array}{c|cc|c} A & \left(\Gamma W \Gamma^\top\right)^{1/2} & 0 & B \\ \hline Q^{1/2} & 0 & 0 & 0 \\ 0 & 0 & 0 & R^{1/2} \\ \hline C & 0 & V^{1/2} & 0 \end{array}\right]. \tag{6}$$

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

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

then LQG becomes a special case of H_2-synthesis in the sense that

$$J = E\left\{ \lim_{T\to\infty} \frac{1}{T} \int_0^T \left(x(t)^\top Q x(t) + u(t)^\top R u(t) \right) dt \right\} = \|F_l(P_{\mathrm{LQG}}, K^*)\|_2^2$$

for the LQG controller K^*. This confirms that the optimal LQG controller K^* has the observer structure (5). The plant P_{LQG} satisfies the standard assumptions for controller synthesis if $(A, \left(\Gamma W \Gamma^\top\right)^{1/2}, C)$ and $(A, B, Q^{1/2})$ are assumed stabilizable and detectable [9, 10].

3 Loop transfer recovery

This section continues with a rapid flashback on the LQG-LTR procedure [10, 11]. Using the embedding $P_{\mathrm{LQG}} \to P$, the situation is interpreted 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 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

$$P_{\mathrm{LQG}}(\rho) = \left[\begin{array}{c|cc|c} A & \left(\Gamma W \Gamma^\top\right)^{1/2} & 0 & B \\ \hline Q^{1/2} & 0 & 0 & 0 \\ 0 & 0 & 0 & R^{1/2} \\ \hline C & 0 & \rho^{1/2} V^{1/2} & 0 \end{array}\right], \tag{7}$$

the nominal case (6) being $\rho = 1$. In particular, $K^* = K(1)$. Explicitly

$$K(\rho) = -K_c \left(sI - (A - BK_c - K_f(\rho)C)\right)^{-1} K_f(\rho). \tag{8}$$

(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 [12]. 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) $\rho^{1/2} K_f(\rho) \to V^{-1/2}$ 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 (6).

Altogether the family of LTR controllers $K(\rho)$ in (8) represents a trade-off between performance (4) with respect to the original LQG plant (6), 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 (7). The procedure 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 quest addressed in this paper is now how to improve robustness $\|S(G, K)\|_\infty \to \|S_{\mathrm{LQ}}\|_\infty =: r_*$ just as in LTR, but at the same time *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 continue to use the notations

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

As was observed before, $\mathcal{R}(K(\rho)) \to r_* := \|S_{\mathrm{LQ}}\|_\infty$, while $\mathcal{P}(K(\rho)) \to \infty$ when $\rho \to 0$. Notice that r_* is the best robustness we can possibly achieve, so it serves as a lower bound for the parameter r in (1).

Let $r^* := \|S(G, K^*)\|_\infty = \mathcal{R}(K^*)$ be the robustness of the nominal H_2 (respectively LQG) controller K^*. As K^* is too sensitive with regard to $S(G, K)$, the value r^* is too large. So r^* is an upper bound for r. Now *every* intermediate value r with $r_* < r \leq r^* = \mathcal{R}(K^*)$, can be realized as $r = r(\rho) = \mathcal{R}(K(\rho))$ for some $\rho \in (0, 1]$. In other words, for every $r \in (r_*, r^*]$ we can find an LQG-LTR controller $K(\rho)$ which has precisely the robustness r.

Naturally, one aims at a compromise $r = r(\rho)$ somewhere in between the two extrema r_*, r^*. This is now where LQG-LTR has its limitations. Namely, 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 the lower bound $p^* = \mathcal{P}(K^*)$ and the upper bound $p_* = \infty$, and has to be accepted as such. *The present work claims* *that one can do better*. Having identified the appropriate robustness level $r = r(\rho) = \mathcal{R}(K(\rho))$ of the LTR controller $K(\rho)$, the following structured mixed H_2/H_∞ optimization program, a special instance of (1), is proposed.

$$(P_\rho) \quad \begin{array}{ll} \text{minimize} & \mathcal{P}(K) = \|T_{w_2\to z_2}(P,K)\|_2 \\ \text{subject to} & \mathcal{R}(K) = \|S(G,K)\|_\infty \le r(\rho) \\ & K \text{ has observer structure (5)} \end{array} \tag{9}$$

Its decision variable is $\mathbf{x} = (\text{vec}(K_c), \text{vec}(K_f))$. For the following, the solution of (9) is denoted as $K_{2,\infty}(\rho)$, indicating that a mixed H_2/H_∞ synthesis problem is solved. The robustness level $r(\rho) = \mathcal{R}(K(\rho))$ imposed in the constraint is taken to be the robustness level of the LQG-LTR controller (8) with parameter ρ. Program (9) is the key element of the following

---

1: **Initialize.** Synthesize 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 $r_* = \|S(G_{\text{LQ}}, K_c)\|_\infty$.
3: **Optimize.** For the current value ρ, 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.** If $K_{2,\infty}(\rho)$ is not sufficiently robust, use smaller ρ to get a smaller $r(\rho)$. If $K_{2,\infty}(\rho)$ 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 (9) 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. In particular, the optimal K_c, K_f are no longer characterized by AREs. Nonetheless $K_{2,\infty}(\rho)$ is an observer-based controller. Notice that without the structural constraint (5) the H_2/H_∞ program (9) has an infinite dimensional solution [13], which need not even be realizable. And even when realizability is imposed as the sole structural constraint, the optimal solution need not be observer-based.

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 from the graph of $100r(\rho)$ in Figure 2. Naturally, the ρ with $r(\rho) > r^*$ are of no use in algorithm 1. Similarly, for a given r only the largest ρ with $r = r(\rho)$ is of interest.

The central property of the solution $K_{2,\infty}(\rho)$ of (9) is the following

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

Proof: The first part of the statement claims that the constraint $\mathcal{R}(K) \le r(\rho)$ in (9) is active at the locally optimal solution $K_{2,\infty}(\rho)$. Suppose this is not the case, i.e., $\mathcal{P}(K_{2,\infty}(\rho)) < r(\rho)$.

Then $K_{2,\infty}(\rho)$ is also a local minimum of the unconstrained H_2 program (4). But program (4) is strictly convex and its unique global minimum is the LQG controller K^*. In particular, there are no other local minima, hence $K^* = K_{2,\infty}(\rho)$. This implies $r^* = \mathcal{P}(K^*) < \mathcal{P}(K(\rho)) = r(\rho)$. However, according to step 2 of algorithm 1, ρ is such that $r_* < r(\rho) \le r^*$ 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 (9), and that optimization is started at $K(\rho)$. This assures that the (locally) optimal solution $K_{2,\infty}(\rho)$ has a lower objective value $\mathcal{P}(K_{2,\infty}(\rho)) \le \mathcal{P}(K(\rho))$. □

Remark 4. Mixed H_2/H_∞-programs had originally been proposed by Haddad and Bernstein [14], who characterize the solution in the full-order case (in the absence of constraint (5)) by a system of coupled algebraic Riccati equations. A homotopy method is proposed to compute the solutions. The first numerically efficient way to solve (9) with the constraint (5) was presented in [15] and is based on nonsmooth optimization techniques. Tables 8.3 and 8.4 of [15] give a comparison between the method of Haddad and Bernstein and ours in cases where both are applicable. Notice that program (9) 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 $q \ge 0$ such that $K(0) = K^*$, and such that $K(q)$ now gets less sensitive as $q \to \infty$ [3]. 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 (6). The LQG-LTR controller is then obtained by an LQG synthesis for $P(q)$ and has the form

$$K(q) = \left[\begin{array}{c|c} A - B_2 K_c(q) - K_f C_2 & K_f \\ \hline -K_c(q) & 0 \end{array}\right] \tag{10}$$

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 $\|\widetilde{S}(G,K(q))\|_\infty \to \|\widetilde{S}_{\rm LQ}\|_\infty$, where $\widetilde{S}_{\rm LQ} = \left(I + C(sI - A)^{-1}K_f\right)^{-1} = (I + G_{\rm LQ}K_f)^{-1}$, which again has guaranteed margins as $q \to \infty$.

Remark 5. Notice that $K(q)$ is obtained by artificially increasing the cost term $x^\top Qx$ 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 [2] a variant is discussed, where in the cost term $x^\top Qx + \mu u^\top Ru$ 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. Fix $q > 0$ and compute $\widetilde{r}(q) = \|\widetilde{S}(G,K(q))\|_\infty$. Then solve the mixed H_2/H_∞ program

$$\begin{array}{ll} \text{minimize} & \mathcal{P}(K) = \|T_{w_2 \to z_2}(P,K)\|_2 \\ \text{subject to} & \mathcal{R}(K) = \|\widetilde{S}(G,K)\|_\infty \le \widetilde{r}(q) \\ & K \text{ observer-based} \end{array} \tag{11}$$

the solution being $K_{2,\infty}(q)$. 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 6. It is straightforward to propose an algorithm similar to algorithm 1 based on (11). The details are left to the reader.

6 Trade-off with performance certificate

There is a second approach to (9), 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 (4). We refer to p^* as the nominal performance. As soon as K^* is overly sensitive and lacks robustness, p^* is too small. Assuming that we are working with the sensitivity function $\mathcal{R}(K) = \|S(G, K)\|_\infty$, let us consider the following mixed H_∞/H_2 program

$$(12) \qquad (D_\alpha) \quad \begin{array}{ll} \text{minimize} & \mathcal{R}(K) = \|S(G,K)\|_\infty \\ \text{subject to} & \mathcal{P}(K) = \|T_{w_2 \to z_2}(P,K)\|_2 \leq (1+\alpha)p^* \\ & K \text{ has observer structure (5)} \end{array}$$

Here we accept a loss of $100\alpha\%$ in nominal performance p^*, and use this freedom to buy as much robustness as possible.

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 Karush-Kuhn-Tucker (KKT) solution of (P_ρ), where $r(\rho)$ is such that the LQG controller is not feasible for (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 KKT 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 KKT solution of (D_α), which is not a critical point of $\mathcal{R}$ alone and is more robust than the LQG controller. 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 KKT of $(P_{\rho(\alpha)})$. One has $r(\rho(\alpha)) = \mathcal{R}(K_{\infty,2}(\alpha))$.

Proof: 1) Let $K := K_{2,\infty}(\rho)$ be a KKT-point of (P_ρ), respectively, of (9). Then there exists a Lagrange multiplier $\lambda \geq 0$ and a Clarke subgradient $\Phi \in \partial\mathcal{R}(K)$ such that (see [16, Ch. 6])

$$(KKT)_\rho \qquad 0 = \nabla\mathcal{P}(K) + \lambda\Phi, \quad \lambda\left(\mathcal{R}(K) - r(\rho)\right) = 0, \quad \mathcal{R}(K) \leq r(\rho).$$

We argue that $\lambda > 0$. Suppose we had $\lambda = 0$. Then $\nabla\mathcal{P}(K) = 0$. By convexity of the LQG program K is then the unique minimum of $\mathcal{P}$, which means it is the LQG controller K^*. On the other hand, $\mathcal{R}(K) \leq r(\rho)$ by (KKT_ρ) which means the LQG controller K^* is feasible in (P_ρ). Since this was excluded by hypothesis, we have a contradiction, proving $\lambda > 0$.

Let us now compare this with the KKT-condition for program (D_α), that is, for (12). Notice that $\widetilde{K} := K_{\infty,2}(\alpha)$ is a KKT-point of (D_α) if there exists a subgradient $\Phi \in \partial\mathcal{R}(\widetilde{K})$ and a Lagrange multiplier $\mu \geq 0$ such that

$$(KKT)_\alpha \qquad 0 = \Phi + \mu\nabla\mathcal{P}(\widetilde{K}), \quad \mu(\mathcal{P}(\widetilde{K}) - (1+\alpha)p^*) = 0, \quad \mathcal{P}(\widetilde{K}) \leq (1+\alpha)p^*.$$

All we have to do now is tune α and μ such that K also satisfies (KKT_α). We simply let $\mu = 1/\lambda$, then the first equation of both conditions is the same. For the constraint, all we have to do is choose α such that $\mathcal{P}(K) = (1+\alpha)p^*$. This is possible, because as we have seen, K is not the LQG controller, hence it satisfies $\mathcal{P}(K) > p^*$. Therefore $\alpha(\rho) = \mathcal{P}(K)/p^* - 1$ as claimed.

2) Conversely, let $\widetilde{K} := K_{\infty,2}(\alpha)$ be a KKT-point of (D_α). Then condition (KKT_α) is satisfied. We argue that $\mu > 0$. Indeed, $\mu = 0$ gives $0 = \Phi \in \partial\mathcal{R}(\widetilde{K})$, which means that $\widetilde{K}$ is a critical point of $\mathcal{R}$. Since this was excluded by hypothesis, we must have $\mu > 0$.

Now we have to fix ρ and λ in such a way that $\widetilde{K}$ satisfies $(KKT)_\rho$. We simply put $\lambda = 1/\mu$, then the first equations is satisfied. For the constraint, let us put $\widetilde{r} := \mathcal{R}(\widetilde{K})$. Then $\widetilde{r} < r^* = r(1)$, as by hypothesis $\widetilde{K}$ is more robust than the LQG controller. Since $\widetilde{r} > r_*$, and since the curve $r(\rho)$ fills the interval $(r_*, r^*]$, there exists ρ such that $\widetilde{r} = r(\rho)$, hence $\mathcal{R}(\widetilde{K}) = r(\rho)$. This ρ is our $\rho(\alpha)$. $\square$

Remark 7. 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 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 a local minimum of the robustness function $\mathcal{R}$ alone is found. 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 local minima.

Remark 8. The LQG-LTR procedure encounters difficulties for non-minimum phase systems G. The target sensitivity function S_{LQ} can no longer be approached at all frequencies, and a weaker result of the form $S(G,K(\rho)) \to S_{\mathrm{LQ}}(I+E)$ for some frequency dependent error term E holds instead [5]. In this situation it may be advantageous to work with weighted sensitivity functions W_1SW_2 or $W_1\widetilde{S}W_2$ in order to preserve some of the properties of LTR in the minimum phase case, as proposed in [2]. In contrast, program (9), respectively it dual (12), do not really depend on G being minimum phase. For instance, in (12) we have still interest to minimize sensitivity as much as we can, non-minimum phase being just a warning that we might be less successful. In general we may decide to follow Athans [17] and use LTR despite the limitations of non-minimum phase, or we could use a robustness constraint of the form $\mathcal{R}(K) = \|W_1S(K)W_2\|_\infty \leq r$, respectively $\mathcal{R}(K) = \|W_1\widetilde{S}(K)W_2\|_\infty \leq r$, using a frequency weighted sensitivity function within a modified LTR procedure to calibrate $r(\rho)$. A third possibility is to use the method proposed in the next section to calibrate the robustness parameter r differently.

7 Extension to more general controller structures

In this section we propose an extension of algorithm 1 to general controller structures. In Section 8.1.2 this will be applied to controllers with PID structure.

A controller in state-space form (3) is called *structured* if the matrices A_K, B_K, C_K, D_K depend smoothly on a design parameter vector $\mathbf{x}$, that is

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

It is assumed 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. It is also expected that $n_K \ll n_x$. Full order controllers are *en abus de langue* referred to as *unstructured*.

A first controller structure was already encountered, 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:

$$K_{\mathrm{pid}}(\mathbf{x}) = \left[\begin{array}{cc|c} 0 & 0 & R_i \\ 0 & -\tau I_{m_2} & R_d \\ \hline I_{m_2} & I_{m_2} & D_K \end{array}\right], \tag{13}$$

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 $\epsilon > 0$) is typically added in parameter space.

Armed with this, the following algorithm is proposed

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

$$\begin{array}{ll} \text{minimize} & \mathcal{P}(\mathbf{x}) = \|T_{w_2 \to z_2}(P, K(\mathbf{x}))\|_2 \\ \text{subject to} & K(\mathbf{x}) \text{ internally stabilizing} \end{array} \tag{14}$$

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 structured H_∞-optimal controller $K(\mathbf{x}_\infty)$ by solving

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

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

3: **Optimize.** For the current $r \in [r_*, r^*]$, solve the following structured mixed H_2/H_∞ program

$$\begin{array}{ll} \text{minimize} & \mathcal{P}(\mathbf{x}) = \|T_{w_2 \to z_2}(P, K(\mathbf{x}))\|_2 \\ \text{subject to} & \mathcal{R}(\mathbf{x}) = \|S(G, K(\mathbf{x}))\|_\infty \leq r \\ & K(\mathbf{x}) \text{ internally stabilizing} \end{array} \tag{16}$$

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 1 is that LTR is no longer available to calibrate the procedure. Instead, the lower bound r_* is computed in step 2, based on a structured H_∞-synthesis with objective $\mathcal{R}$. This can be obtained via the matlab function `hinfstruct` [18]. The mixed H_2/H_∞-program is solved via [15], using the matlab function `fmincon` [19] as a presolver.

In order to solve (16) efficiently, the solution $\mathbf{x}^*$ of step 1, or the solution $\mathbf{x}_\infty$ of step 2, can be used as starting points. It is also possible to obtain a starting point $\mathbf{x}_r$ by stopping the minimization in (15) at the moment when $\mathcal{R}(\mathbf{x}_r) \leq r$ is activated. This feature is indeed available in the matlab function `hinfstruct` [18]. The controller $K(\mathbf{x}_r)$ is then a favorable initial guess in (16), because it already satisfies the constraint. The result extending Proposition 1 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 2 and used as initial guess in solving program (16). Then the locally optimal solution $K(\mathbf{x}_{2,\infty}(r))$ of (16) is at least as robust as $K(\mathbf{x}_r)$, and has better H_2 performance.*

Proof: The first statement says $\mathcal{R}(K(\mathbf{x}_{2,\infty}(r))) \leq r = \mathcal{P}(K(\mathbf{x}_r))$ which is clear, because a locally optimal solution is also feasible.

The second statement follows from the fact that $\mathbf{x}_r$ is used as initial guess. Then a descent method will produce a locally optimal solution, which has better performance than $K(\mathbf{x}_r)$. $\square$

Remark 9. Notice that solutions to (14), (15), and (16) may no longer be computed by algebraic Riccati equations or LMIs. While (14) can be solved by smooth optimization technique, see e.g. [20], programs (15) and (16) 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 [21], and also [22–24], can be cited. A recent implementation is `hinfstruct` in [18], which is based on [21]. Constrained programs like (16) are discussed in [15, 25]. General mathematical background is given in [26, 27]. A recent approach to combine nonsmooth techniques with classical nonlinear programming techniques is discussed in [28].

Remark 10. In algorithm 2 we assume that P is stabilizable and detectable. However, we need to be able to stabilize the plant internally with a controller $K(\mathbf{x})$ of the imposed structure. Interestingly, deciding whether or not such controllers exists is NP-complete for many practical structures like PID, reduced-order, static, decentralized controller; see [29]. Practical ways to compute a stabilizing $K(\mathbf{x})$ are discussed in [30].

8 Numerical experiments

In this section we present three studies in which the proposed trade-off based on mixed structured H_2/H_∞-control is tested. In each study performance of the nominal system is evaluated in the H_2-norm, which is optimized subject to a constraint on the controller structure (observer-based, respectively, PID). In the first and second study the input sensitivity function, S, and in the third the output sensitivity function, $\tilde{S}$, is used to assess robustness.

8.1 Mass-spring system

Our first study uses the mass-spring system [10] of Figure 1, which can be considered as a prototype of a flexible system. Considering the positions and the velocities of the two mass as the states $x = [x_1 \; x_2 \; \dot{x}_1 \; \dot{x}_2]^T$, the state space representation is:

$$\dot{x} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k}{m_1} & \frac{k}{m_1} & -\frac{f}{m_1} & \frac{f}{m_1} \\ \frac{k}{m_2} & \frac{-k}{m_2} & \frac{f}{m_2} & \frac{-f}{m_2} \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m_1} \\ 0 \end{bmatrix} u$$

$$y = [0\ 1\ 0\ 0]$$

8.1.1 LQG-LTR

According to algorithm 1 the procedure starts with a nominal LQG synthesis. The nominal LQG controller K_{LQG} is obtained with the covariance matrices $W = BB^\top$ and $V = 1$, while $Q = C^\top C$ and $R = 1$; see [10]. This results in $(K_f, K_c) = ([0.94\,0.06\,0.97\,0.75], [1.49\,1.93\,0.13\,1.87])$ with performance $p^* = \mathcal{P}(K_{\text{LQG}}) = 3.99$. Following algorithm 1, the LTR procedure is now applied to generate a curve $(K_f(\rho), K_c)$. This is done by keeping W fixed and letting $V = \rho I = \rho \to 0$, which corresponds to using the input sensitivity function $\mathcal{R}(K) := S(K) = (I + KG)^{-1}$ as robustness index. The LQG-LTR procedure is compared to our H_2/H_∞ trade-off model (9) of section 4. Figure 2 compares performance and robustness of the different controllers. The graph $r(\rho)$ represents the robustness of both the LTR and the H_2/H_∞ controller, which are matched through the constraint in program (9). As can be seen, performance is considerably improved without degrading robustness.

The parametric robustness of the LTR and the mixed H_2/H_∞-controller have also been compared when mass m_2 and spring coefficient k undergo changes around their nominal values, k^0 and m_2^0, in the square $(k^0 \pm 30\%k^0, m_2^0 \pm 30\%m_2^0)$. Figure 3 compares the stability regions for $\rho = 0.001$. The performance $\mathcal{P}(K(.001)) = 27.85$ of the LTR controller corresponds to a degradation of $\alpha = 597\%$ of the nominal performance $p^* = 3.99$. Image **(c)** shows what the mixed H_2/H_∞ controller $K_{2,\infty}(\rho)$ achieves at the same $\rho = .001$. On top of having significantly better performance $\mathcal{P}(K_{2,\infty}(.001)) = 4.23$, corresponding to $\alpha = 6\%$, it has also better parametric robustness. Figure 4 displays the relative performance $\frac{\mathcal{P}(G,K)-\mathcal{P}(G^0,K)}{\mathcal{P}(G^0,K)}$ for the controllers K of Figure 3 when the same variation of the nominal parameters is considered. Since LQG and LQG-LTR controllers are both not stabilizing over the entire square, their graphs are restricted to their closed-loop stability regions. As can be seen, the mixed controller $K_{2,\infty}(\rho)$ performs best with regard to this criterion over the square.

8.1.2 H_2-optimal PID controller

In this section a desensibilized H_2-optimal PID controller is searched for the mass-spring system. As LTR is no longer available, the procedure follows algorithm 2, which starts by computing the solution of the nominal program (14) for the structure (13). The H_2-optimal PID controller $K_{\text{pid},2}$ has $p^* = \mathcal{P}(K_{\text{pid},2}) = 12.61$ and $r^* = \mathcal{R}(K_{\text{pid},2}) = 17.23$. Continuing with algorithm 2, program (15) for the structure (13) is solved, which provides the most robust PID controller with regard to the sensitivity function S. This robustified PID has performance $p_* = \mathcal{P}(K_{\text{pid},\infty}) = 152.8$, which is clearly degraded ($p^* \ll p_*$), while naturally $r_* = \mathcal{R}(K_{\text{pid},\infty}) = 6.39$ is improved ($r_* < r^*$). Finally, the compromise is achieved by solving program (16), which it is initialized with $K_{\text{pid},\infty}$. Several choices $r \in [r_*, r^*]$ were tested, and finally $r = 17$ was chosen, because it achieved parametric robustness of $K_{\text{pid},2,\infty}$ over the 40% square of variation in m_2, k. Comparison with the two other PIDs is made in Figure 5, where it can be seen that the mixed controller shows the best trade-off between performance and robustness (in the sense of the input sensitivity and parametric robustness).

8.2 Lateral flight control of an F-16 aircraft

In our last study the improved LTR procedure was applied to lateral flight control of an F-16 aircraft. The nonlinear F-16 lateral model was linearized using the F-16 simulation program [31].

The high fidelity model is evaluated at altitude $h = 4575\,m$ and velocity $v = 152.5\,m/s$, considering steady wings-level flight conditions for trimming. The state variables are side slip angle β, bank angle ϕ, roll rate p and yaw rate r. Using a 6-DOF flat-earth, body-axis aircraft model:

$$\begin{aligned}\dot{\Phi} &= \tfrac{\cos\gamma_0}{\cos\theta_0}\, p_s + \tfrac{\sin\gamma_0}{\cos\theta_0}\, r_s \\ \dot{\beta} &= \tfrac{Y_\beta}{V}\,\beta + \tfrac{Yr}{V}\, r_s + \tfrac{g\cos\theta_0}{V}\,\Phi - r_s\end{aligned}$$

$$\begin{aligned}\dot{p_s} &= L_\beta\beta + L_p p_s + L_r r_s + \delta_l(p_s, r_s) + L_{\delta_a}(\beta, \delta_a) + L_{\delta_r}(\beta, \delta_r) \\ \dot{r_s} &= N_\beta\beta + N_p p_s + N_r r_s + \delta_n(p_s, r_s) + N_{\delta_a}(\beta, \delta_a) + N_{\delta_r}(\beta, \delta_r)\end{aligned}$$

where

$$p_s = p\,\cos\alpha_0 + r\,\sin\alpha_0$$

$$r_s = r\,\cos\alpha_0 - p\,\sin\alpha_0,$$

δ_r and δ_n are incremental rolling and yawing moment due to p_s and q_s. L_{δ_a}, L_{δ_r}, N_{δ_a} and N_{δ_r} are rolling and yaw moments due to aileron and rudder deflections. θ_0, γ_0 and α_0 are the trimmed pitch angle, angle of attack and side slip where $\alpha_0 = \theta_0 - \gamma_0$. For more details see [32] and [33].

8.2.1 Performance channel

As in [34], 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 goal of the study is to make the bank angle ϕ follow a reference command r_ϕ, while simultaneously keeping the side slip 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 [34]. 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 $K(s)$.

8.3 LTR procedure

In this study LTR recovery at the output breaking point is used, i.e., robustness is measured via the output sensitivity function $\widetilde{S}$, and an observer-based controller is computed. Using $V = I_2$ and $W = \mathrm{diag}([0.1\ 0.1\ 0.1\ 0.1\ 0\ 0\ 10\ 10]) * 100$, 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). 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 $\overline{\sigma}$ and $\underline{\sigma}$ of the loop transfer functions of LQ (target) with those of LQG and LTR($\rho = 1e - 10$). In other words, the singular values of $C(sI - A)^{-1}K_f$ are compared with the singular values of $G(s)K_c(\rho)(sI - (A + BK_c(\rho) + K_f^\top C))^{-1}K_f$ for $\rho = 1$ and $\rho = 1e - 10$. As can be seen, forcing $\rho \to 0$ brings the singular values of the LTR controller near those of the target. In addition, this also drives the system output responses 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 $\mathcal{R}_{\text{LQ}} = \|(I + GK_{\text{LQ}}(\rho))^{-1})\|_\infty$. This proves that LTR with recovery at the output breaking point is not a monotone procedure either.

8.3.1 Mixed synthesis

In order to overcome the loss of performance of the LTR controller, we apply algorithm 1, where in program (16) the output sensitivity function $\widetilde{S}$ replaces S. An appropriate parameter range is $\rho \in [10^{-4}\ 10^{-1.3}]$, where 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 via Proposition 1. A substantial improvement in performance can be observed.

The efficiency of this new method is checked by considering changes of the flight parameters. $h = h_0 \pm \Delta h$ and $v = v_0 \pm \Delta v$ are considered with $\Delta h = 305\, m$ and $\Delta v = 7.625\, m/s$, the nominal flight point being $h_0 = 4575\ m$ and $v_0 = 152.5\ m/s$. The LTR controller and the corresponding mixed 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 neighboring flight points around the nominal flight point. The diagram in Figure 10 (c) shows the improvement in performance obtained with the mixed controller.

9 Conclusion

We have used mixed H_2/H_∞ synthesis with structured control laws to obtain a quantified trade-off between performance and robustness. Within the class of observer-based controllers our method 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 grant *Survol* from Fondation de Recherche pour l'Aéronautique et l'Espace (FNRAE) and grant *Technicom* from Fondation d'Entreprise EADS.

References

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

[2] Stein G., Athans M. "The LQG-LTR Procedure for Multivariable Feedback Control Design." *IEEE Transaction on Automatic Control*, 1987, AC-32, pp. 105-114.

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

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

[5] Zhang Z., Freudenberg J.S. "Loop Transfer Recovery for non-minimum phase plants." *IEEE Transaction on Automatic Control*, 1990, vol. 35, no. 5, 547-553.

[6] Saeki M. "Loop recovery via H_∞-modified complementary sensitivity recovery for non-minimum phase plants."*International Journal of Robust and Nonlinear Control, special issue on Loop Transfer Recovery.*(1995).

[7] Boyd S., Barratt C.H. "Linear controller design. Limits of performance"(Prentice Hall, 1991).

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

[9] Skogestad, S., Postlethwaite I. "Multivariable Feedback Control: Analysis and Design"(Wiley, September 2005).

[10] Alazard D., Cumer C., Apkarian P., Gauvrit M., Ferreres G. "Robustesse et Commande Optimale" (CEPADUES Edition, 2000).

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

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

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

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

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

[16] Clarke, F., H. "Optimization and Nonsmooth Analysis. SIAM Classics in Applied Mathematics, 1990.

[17] Athans M. "A tutorial on the LQG/LTR method". Proc. Amer. Control Conf., 1986, Seattle.

[18] The Robust Control Toolbox, R(2010)b, The MathWorks Inc., Natick, MA.

[19] The Optimization Toolbox, R(2010)b, The MathWorks Inc., Natick, MA.

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

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

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

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

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

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

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

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

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

[29] Blondel V., Tsitsiklis J. "NP-hardness of some linear control design problems". *SIAM Journal on Control and Optimization*, 1997, vol. 35, no. 6, pp. 2118 -2127.

[30] Bompart V., Apkarian P., Noll D. "Nonsmooth techniques for stabilizing linear systems." *American Control Conference,* New York, Conference Proceedings, 2007, pp. 1245-1250.

[31] Russell R.S. "Nonlinear F-16 simulation using Simulink and Matlab." Technical Report, Department of Aerospace Engineering and Mechanics, University of Minnesota. (2003)

[32] Vo H., Seshagiri S. "Robust Control of F-16 Lateral Dynamics." *International Journal of Aerospace and Mechanical Engineering*, 2008, vol. 2, no. 2, 80-85.

[33] Young A., Cao C., Hovakimyan N. "Adaptive approach to nonaffine control design for a aircraft applications." 2006, *AIAA Guidance*, Keystone, Colorado.

[34] Stevens B., Lewis F. "Aircraft control and simulation"(New York, Wiley, 1992).

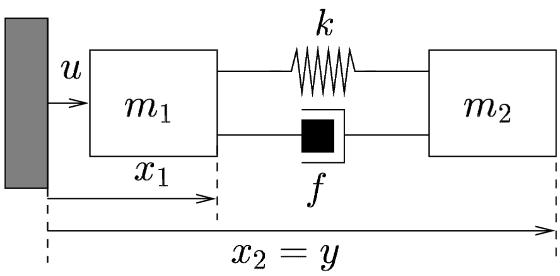

FIGURE 1: Mass-spring system. Nominal data are $m_1 = m_2 = 0.5$ kg, $k = 1$ N/m, $f = 0.0025$ Ns/m. Measured output is $y = x_2$, control force u acts on m_1.

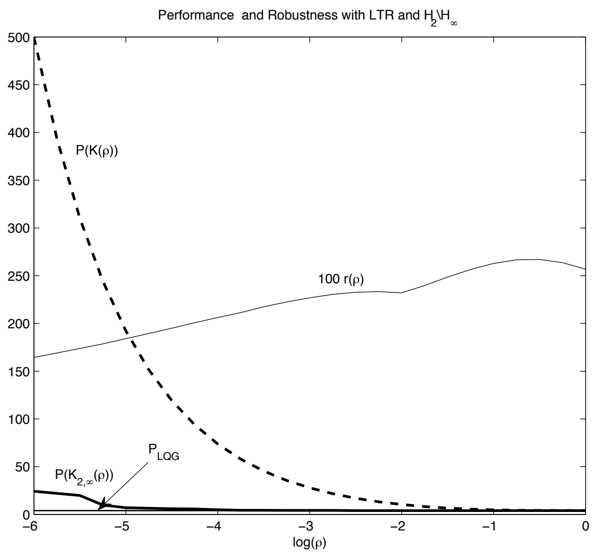

FIGURE 2: LQG-LTR study. Performance of $K(\rho)$ and $K_{2,\infty}(\rho)$ in logarithmic scale. Lower bound is the performance of the nominal LQG controller. The curve $100r(\rho)$ shows the robustness profile over the same abscissa. As a by-product, it can be seen that LTR is not a monotone procedure.

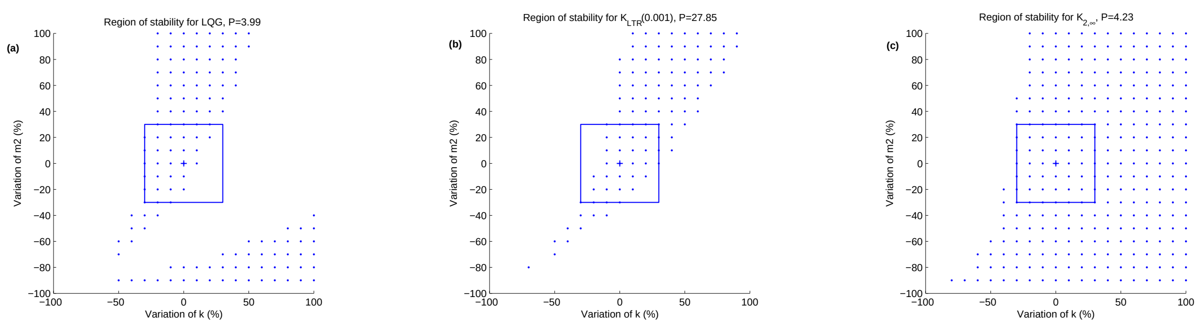

FIGURE 3: LQG-LTR study: stability regions as a function of the parameters variation.

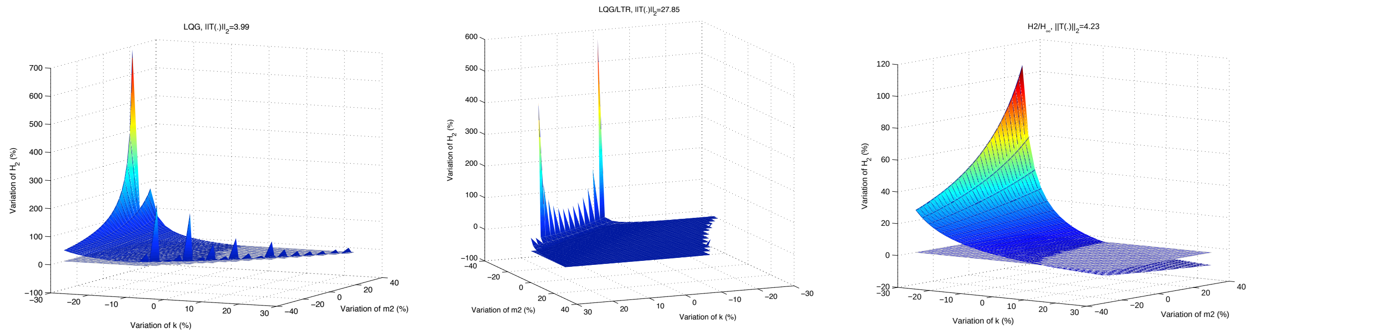

FIGURE 4: LQG-LTR study. Each graph shows relative performance as a function of the parameters variation. Left: LQG controller, middle: LQG-LTR controller, right: mixed H_2/H_∞.

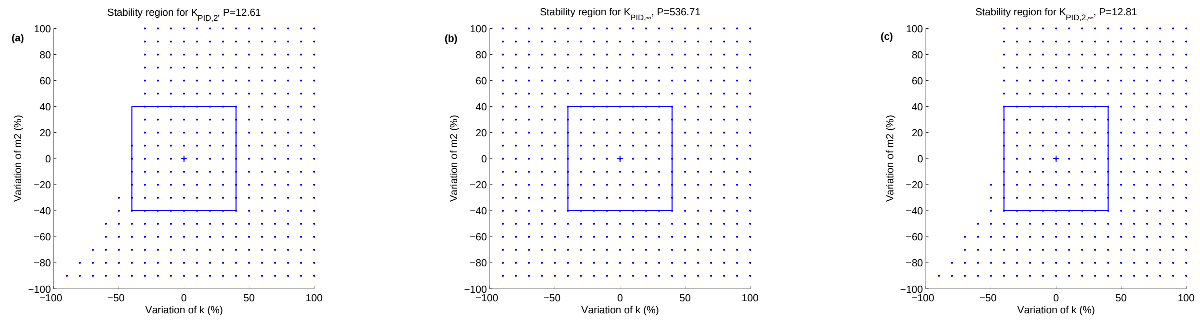

FIGURE 5: PID study. Stability regions as a function of the parameters variation.

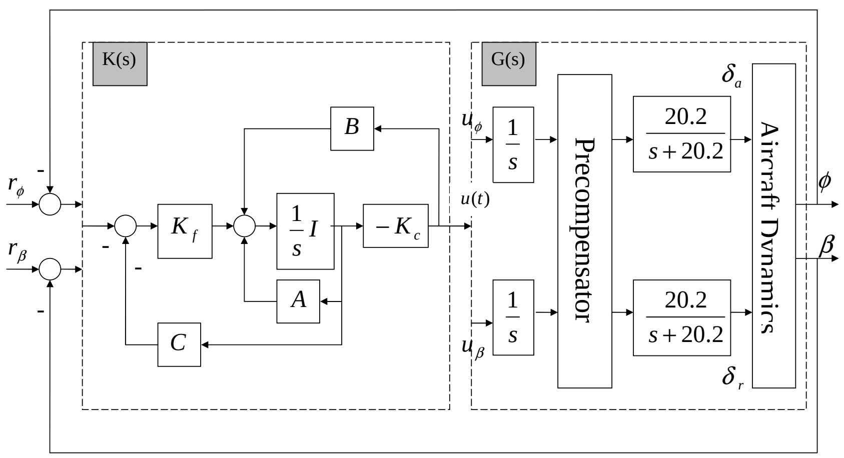

FIGURE 6: Model of F16 aircraft lateral control system and the observer structure.

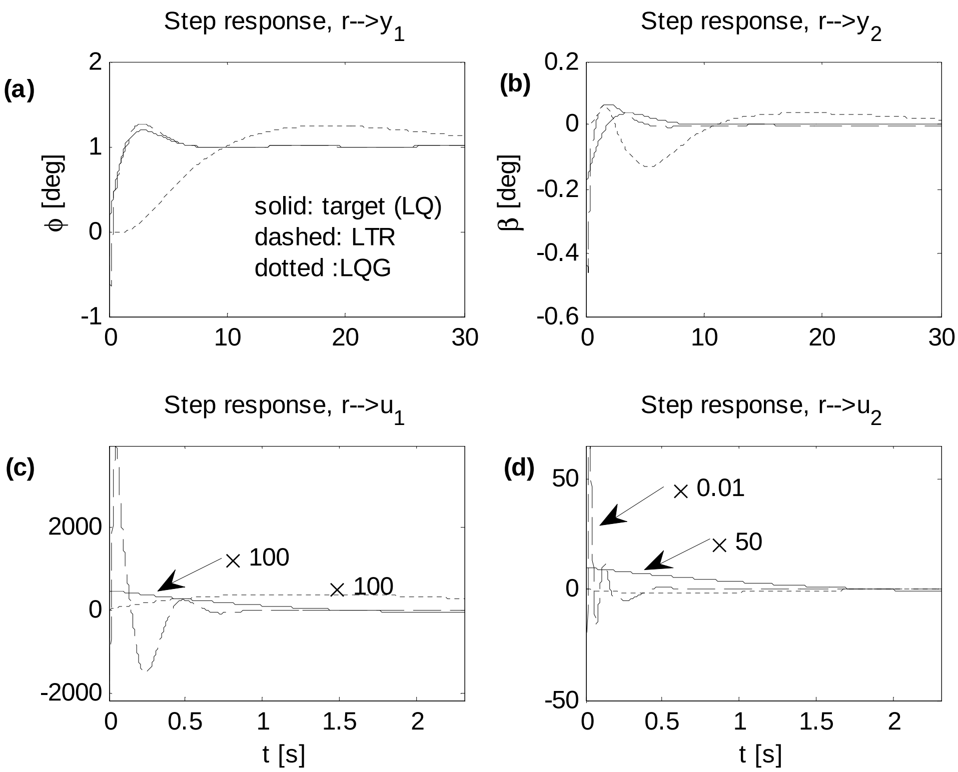

FIGURE 8: Step responses with different controllers.

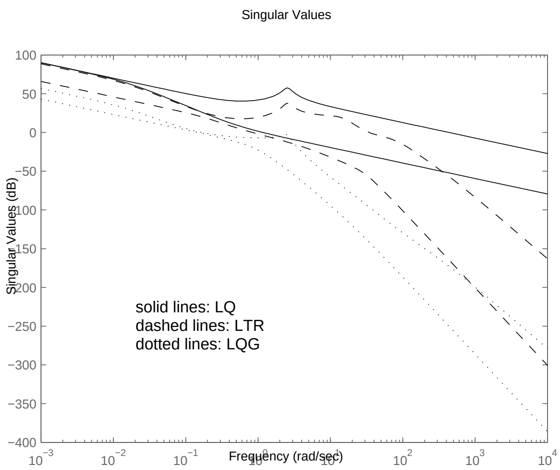

FIGURE 7: Singular values of the loop transfer function $L(s) = G(s)K(s)$ for LQ, LQG and LTR controller.

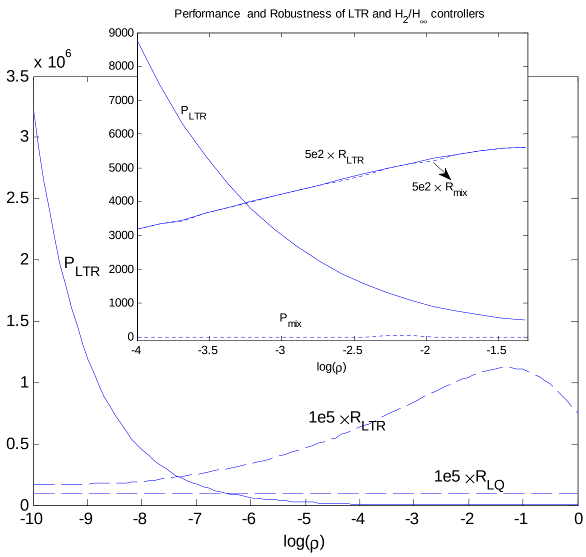

FIGURE 9: F-16 study. Comparison of performance of LTR and H_2/H_∞ controller when robustness according to $\mathcal{R}(K) = \|\widetilde{S}(G,K)\|_\infty$ is used.

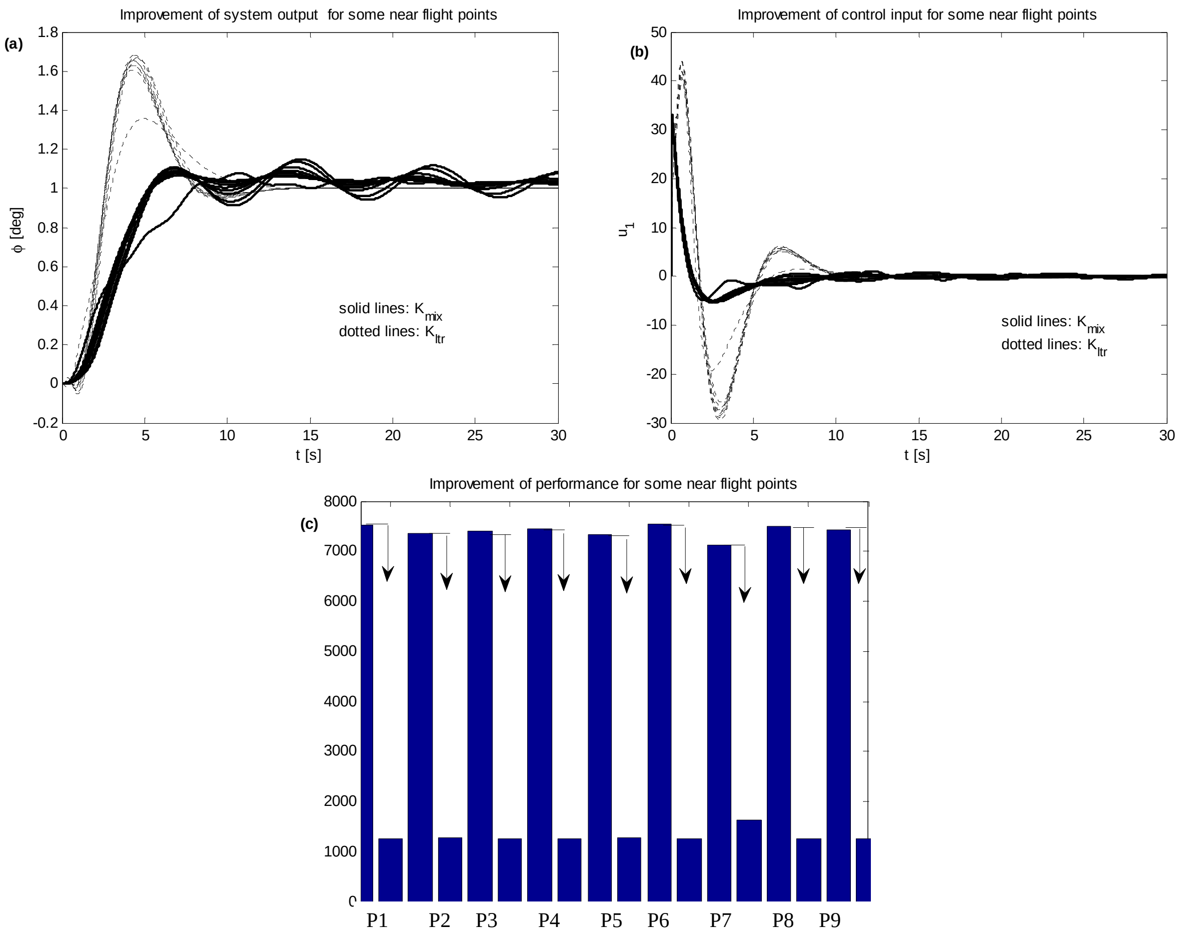

FIGURE 10: Comparing LTR and H_2/H_∞ controllers against the system variations, (a) and (b) step responses, (c) performance.