On a generalization of the LTR procedure

L. Ravanbod-Hosseini, D. Noll, P. Apkarian

Abstract— **The LQG/LTR procedure is a classical means to desensibilize a system in closed-loop with respect to disturbances and system uncertainty. We discuss an extension of LTR which can be applied in more general situations. As an example, we show that PID controllers can be robustified with this approach.**

Keywords: LQG/LTR, observer-based control, PID control.

I. Introduction

LQG/LTR is a classical method to enhance the robustness of LQG controllers [20]. It is often used by practitioners to desensibilize the LQG design in situations where a lack of robustness occurs. Unfortunately, LTR has two limitations. On the one hand, the enhanced robustness often leads to a considerable loss of performance. And secondly, LTR is clearly limited to controllers with observer structure. Here we propose a method which can overcome these restrictions. We indicate in particular how it can be brought to work for observer-based controllers and for PID controllers.

II. Preparation

A. H_2 form of LQG control

LQG will be considered as a special case of H_2 synthesis. We build a plant [1] (see also the appendix):

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

where W and V are the covariance matrices of state and output noises, Q and R are the weighting matrices of the state and input, A, B and C are the state space representation matrices. Then, for any controller $u = K(s)y$, the LQG performance objective is recovered from the H_2 set-up as $\|T_{w_2\to z_2}(P_{\rm lqg}, K)\|_2^2$. We let $G(s) = C(sI-A)^{-1}B$.

B. Parametrized controllers

We consider controllers in state-space form

$$K: \begin{bmatrix} \dot{x}_K \\ u \end{bmatrix} = \left[\begin{array}{c|c} A_K(\theta) & B_K(\theta) \\ \hline C_K(\theta) & D_K(\theta) \end{array}\right] \begin{bmatrix} x_K \\ y \end{bmatrix} \quad (1)$$

where the matrices $A_K(\theta)$ etc. depend smoothly on a design parameter θ varying in some parameter space $\mathbb{R}^n$. The transfer function is $K(s) = C_K(\theta)(sI - A_K(\theta))^{-1}B_K(\theta) + D_K(\theta)$. Typical examples are observer-based controllers

$$K_{\rm obs}(\theta) = \left[\begin{array}{c|c} A - BK_c - K_fC & K_f \\ \hline -K_c & 0 \end{array}\right], \quad (2)$$

where $\theta = ({\rm vec}(K_c), {\rm vec}(K_f))$, and PID controllers,

$$K_{\rm pid}(\theta) = \left[\begin{array}{cc|c} 0 & 0 & R_i \\ 0 & -\tau I & R_d \\ \hline -I & -I & -D_K \end{array}\right] \quad (3)$$

where $\theta = (\tau, {\rm vec}(R_i), {\rm vec}(R_d), {\rm vec}(D_K))$. We shall write $K(\theta)$ if we address both cases and possibly other situations.

C. Loop transfer recovery

In loop transfer recovery (LTR) controller or observer gains are parametrized as functions of a scalar parameter. One may for instance increase the covariance of the state noise W using a parameter $q > 0$ as in [9], or reduce the covariance of the output noise V through a parameter $\rho > 0$ as in [1]. In both cases, the excellent robustness margins of LQ will be recovered asymptotically as $q \to \infty$, respectively, as $\rho \to 0$. Here we concentrate on the case $V = \rho V_0$ with $\rho \to 0$. Then K_f becomes a function $K_f(\rho)$ of ρ, while K_c remains fixed. It is well-known that

$$\lim_{\rho\to 0} \rho^{\frac{1}{2}} K_f(\rho)) \quad \to \quad BV_0^{-\frac{1}{2}}$$

which implies:

$$\lim_{\rho\to 0} K_{\rm ltr}(\rho)(s)G(s) \quad \to \quad K_c(sI-A)^{-1}B$$

so that the loop transfer function $K_{\rm ltr}(\rho)(s)G(s)$ approaches the corresponding LQ loop transfer function. Here the LQG/LTR controller

$$K_{\rm ltr}(\rho) = \left[\begin{array}{c|c} A - BK_c - K_f(\rho)C & K_f(\rho) \\ \hline -K_c & 0 \end{array}\right].$$

is the LQG controller of a modified plant $P_{\rm lqg}(\rho)$, where $V = \rho V_0$ with $P_{\rm lqg} = P_{\rm lqg}(1)$. Performance of $K_{\rm ltr}(\rho)$ with respect to the original plant $P_{\rm lqg}$ therefore degrades as $\rho \to 0$, while robustness is improved. Our concern is how to avoid this loss of performance, while benefitting of the improved robustness properties.

D. Performance and Robustness

If we consider $P_{\rm lqg}$ as a special case of the plant

$$P_2: \begin{bmatrix} \dot{x} \\ \hline z_2 \\ y \end{bmatrix} = \left[\begin{array}{c|cc} A & B_2 & B \\ \hline C_2 & 0 & D_{2u} \\ C & D_{y2} & 0 \end{array}\right] \begin{bmatrix} x \\ \hline w_2 \\ u \end{bmatrix},$$

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

and if $K(\theta)$ stabilizes P_2 internally in closed loop, then we call

$$\mathcal{P}(\theta) = \|T_{w_2 \to z_2}(P_2, K(\theta))\|_2$$

the performance of $K(\theta)$ respectively θ.

The LTR procedure based on ρ is related to the closed-loop sensitivity function $S(G,K) = (I + K(s)G(s))^{-1}$, respectively its H_∞-norm $\|S(G,K)\|_\infty$, as a measure of robustness. To capture more general cases, we consider a plant

$$P_\infty : \begin{bmatrix} \dot{x} \\ \hline z_\infty \\ y \end{bmatrix} = \left[\begin{array}{c|cc} A & B_\infty & B \\ \hline C_\infty & D_\infty & D_{\infty u} \\ C & D_{y\infty} & 0 \end{array}\right] \begin{bmatrix} x \\ \hline w_\infty \\ u \end{bmatrix}.$$

If $K(\theta)$ stabilizes P_∞ internally in closed-loop, then we call

$$\mathcal{R}(\theta) = \|T_{w_\infty \to z_\infty}(P_\infty, K(\theta))\|_\infty$$

the robustness measure of $K(\theta)$ or θ. The case $\mathcal{R}(K_c, K_f) = \|S(G, K_{\rm obs}(K_c, K_f))\|_\infty$ of LQG/LTR discussed above corresponds to the choices $B_2 = B_2(W)$, $C_2 = C_2(Q)$, $D_{2u} = D_{2u}(R)$, $D_{y2} = D_{y2}(V)$, $B_\infty = B$, $C_\infty = 0$, $D_\infty = D_{\infty u} = I$, $D_{y\infty} = 0$. It will be convenient to re-write the LTR controller as $K_{\rm ltr}(\rho) = K_{\rm obs}(\theta_\rho)$, where $\theta_\rho = (K_c, K_f(\rho))$.

III. Improved LQG/LTR procedure

In the case $V = \rho V_0$ outlined above we can now propose the following extension of the LQG/LTR procedure.

Algorithm 1. Algorithm for improved LQG/LTR

1: **Initialize.** Synthesize the nominal LQG controller $K_{\rm obs}(\theta^*)$ and compute its robustness $r^* = \mathcal{R}(\theta^*) = \|S(G, K_{\rm obs}(\theta)^*)\|_\infty$. A lower bound is $r_* = \|(I + K_c(sI - A)^{-1})\|_\infty$. If $K_{\rm obs}(\theta^*)$ is sufficiently robust, quit. Otherwise continue.
2: **Calibrate.** Use LQG/LTR procedure and compute $K_{\rm ltr}(\rho) = K_{\rm obs}(\theta_\rho)$ such that robustness $r(\rho) := \|S(G, K_{\rm ltr}(\rho))\|_\infty = \mathcal{R}(\theta_\rho)$ is satisfactory.
3: **Optimize.** Solve the following mixed H_2/H_∞ optimization program

$$\begin{array}{ll} \min & \mathcal{P}(\theta) = \|T_{w_2 \to z_2}(P_2, K_{\rm obs}(\theta))\|_2 \\ \text{s. t.} & \mathcal{R}(\theta) = \|T_{w_\infty \to z_\infty}(P_\infty, K_{\rm obs}(\theta))\|_\infty \le r(\rho) \end{array} \quad (4)$$

using $K_{\rm ltr}(\rho) = K_{\rm obs}(\theta_\rho)$ as initial guess. The locally optimal solution is $K_{\rm obs}(\theta^\sharp)$.
4: **Evaluate.** Check whether $K_{\rm obs}(\theta^\sharp)$ offers an acceptable compromise between performance and robustness. 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.

The key to the understanding of this approach is program (4), whose solution has the following property.

Proposition 1: The mixed H_2/H_∞ controller $K_{\rm obs}(\theta^\sharp)$ is as robust as the LQG/LTR controller $K(\rho)$ in the sense that $\|S(G, K_{\rm obs}(\theta_\rho))\|_\infty = \|S(G, K_{\rm ltr}(\rho))\|_\infty$, or equivalently, $\mathcal{R}(K_{\rm obs}(\theta^\sharp)) = \mathcal{R}(K_{\rm ltr}(\rho))$. At the same time it has better H_2 performance than the LTR controller, i.e., $\mathcal{P}(\theta^\sharp) \le \mathcal{P}(\theta_\rho)$.

Proof: Notice that the LTR controller θ_ρ is a feasible point for program (4). Unless θ_ρ is already optimal, in which event we get equality, the optimal solution $\theta^\sharp$ therefore necessarily has a better performance $\mathcal{P}(\theta^\sharp) \le \mathcal{P}(\theta_\rho)$. ■

Notice that the mixed H_2/H_∞ program could be solved approximatively using the matlab function `hinfmix` or `h2hinfsyn` of [23]. A more rigorous approach is [3], where an algorithm to compute locally optimal solutions is presented. It is also possible to solve (4) via constrained programming based on the matlab function `fmincon` [22].

The outlined procedure is practical in so far as it improves over LTR even when we do not succeed in solving program (4) until optimality. Namely, as soon as we can provide an algorithmic step away from θ_ρ such that $\mathcal{P}$ is decreased and $\mathcal{R}$ is not increased, we will already obtain an improvement over θ_ρ.

Remark 1: We have observed in our experiments that optimization of the performance index $\mathcal{P}$, while not exceeding the robustness index $\mathcal{R}$ provided by LTR, very often improves the parametric robustness of the design.

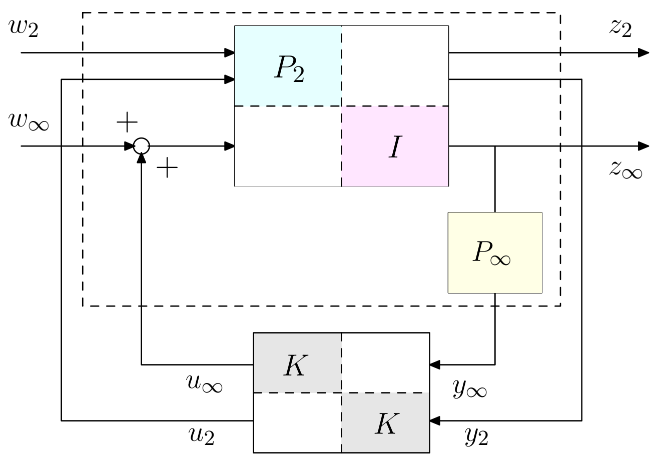

Fig. 1. LFT scheme for mixed synthesis.

IV. Extension to PID controllers

In this section we outline a similar procedure which can provide a substrate for LTR in the frame of PID control.

Algorithm 2. Algorithm for robustified H_2 PID control.

1: **Initialize.** Synthesize nominal H_2 PID controller $K_{\rm pid}(\theta^*)$ and compute its robustness $r^* = \mathcal{R}(\theta^*) = \|S(G, K_{\rm pid}(\theta^*))\|_\infty$. If $K_{\rm obs}(\theta^*)$ is sufficiently robust, quit. Otherwise continue.

2: **Calibrate.** Compute lower bound r_* by solving the PID H_∞-synthesis program

$$\begin{array}{ll} \text{minimize} & \mathcal{R}(\theta) = \|S(G, K_{\rm pid}(\theta)\|_\infty \\ \text{subject to} & K_{\rm pid}(\theta) \text{ internally stabilizing} \end{array} \tag{5}$$

The solution is θ_* and the lower bound is $r_* = \mathcal{R}(\theta_*)$. Pick r with $r_* < r < r^*$.

3: **Optimize.** Solve the following mixed H_2/H_∞ PID-optimization program

$$\begin{array}{ll} \min & \mathcal{P}(\theta) = \|T_{w_2 \to z_2}(P_2, K_{\rm pid}(\theta))\|_2 \\ \text{s. t.} & \mathcal{R}(\theta) = \|T_{w_\infty \to z_\infty}(P_\infty, K_{\rm pid}(\theta))\|_\infty \leq r \end{array} \tag{6}$$

using either $K_{\rm pid}(\theta^*)$ or $K_{\rm pid}(\theta_*)$ as initial guess. The locally optimal solution is $K_{\rm pid}(\theta^\sharp)$.

4: **Evaluate.** Check whether $K_{\rm pid}(\theta^\sharp)$ offers an acceptable compromise between performance and robustness. If it is not sufficiently robust use smaller $r \in (r_*, r^*)$. If it is too robust and not sufficiently performing, use larger r. Then go back to step 3.

The difference with Algorithm I is that we no longer have LTR at our disposition to calibrate parameter r in step 3. This is why we compute the lower bound r_* in step 2. The latter can be obtained via the matlab function `hinfstruct` [23]. The mixed H_2/H_∞-program can again be solved via [3] or using the matlab function `fmincon`. An interesting alternative to program (6) is presented in [12] in the case of a H_∞ performance objective.

As we have seen in section III, the LTR procedure provides a trajectory of controllers $K_{\rm ltr}(\rho)$ going from good performance in tandem with bad robustness ($\rho = 1$) all the way to good robustness in tandem with bad performance ($\rho \to 0$). This is useful as it allows to calibrate program (4) via $r = r(\rho)$. In the case of PID control (or any other controller parametrization $K(\theta)$) we need a substitute. Instead of solving the H_∞ program (5) to optimality r^*, we can dispense with the optimization as soon as a controller θ_r satisfying $\mathcal{R}(\theta_r) \leq r$ is found. This is indeed a feature available in `hinfstruct`. Controller $K_{\rm pid}(\theta_r)$ then offers good robustness and can then be used to initialize (6) more efficiently. It has a property similar to the LTR controller:

Proposition 2: Let $K_{\rm pid}(\theta_r)$ be the controller obtained by minimizing the H_∞ norm until a value $\leq r$ is obtained. Then the mixed H_2/H_∞ PID controller $K_{\rm pid}(\theta^\sharp)$ computed in (6) is as robust as $K(\theta_r)$ but has better H_2 performance. $\square$

V. Computing the nominal PID H_2-controller

The nominal PID H_2-controller in (5) is computed via smooth local optimization methods. In our tests we have used the matlab function `fmincon`, but other choices are possible. This requires computation of derivatives of $\mathcal{P}(\theta)$. We have the following general procedure. We augment the plant P_2 in such a way that K becomes static:

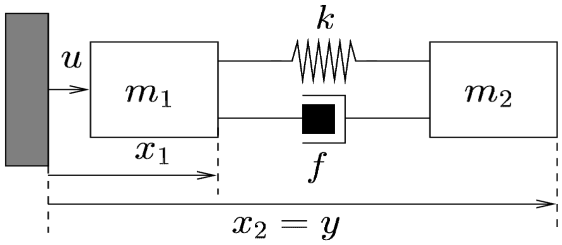

Fig. 2. Mass-spring system. Nominal data are $m_1 = m_2 = 0.5$kg, $k = 1$N/m, $f = 0.0025$Ns/m. Measured output is $y = x_2$, control force u acts on m_1.

$$A^{\rm aug} = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}, B_2^{\rm aug} = \begin{bmatrix} B_2 \\ 0 \end{bmatrix}, B^{\rm aug} = \begin{bmatrix} 0 & B \\ I & 0 \end{bmatrix},$$

$$C_2^{\rm aug} = \begin{bmatrix} C_2 & 0 \end{bmatrix}, C^{\rm aug} = \begin{bmatrix} 0 & I \\ C & 0 \end{bmatrix},$$

$$D_{2u}^{\rm aug} = \begin{bmatrix} 0 & D_{2u} \end{bmatrix}, D_{y2}^{\rm aug} = \begin{bmatrix} 0 & D_{y2} \end{bmatrix}.$$

Writing A for $A^{\rm aug}$, etc., we use [18, Thm. 3.2] to compute $\nabla_K \mathcal{P}(K) = 2\left[B^\top X + D_{2u}^\top \mathcal{C}_2(K)\right] YC^\top + 2B^\top X \mathcal{B}_2(K) D_{y2}^\top$, where X solves the Riccati equation

$$\mathcal{A}(K)^\top X + X\mathcal{A}(K) + \mathcal{C}_2(K)^\top \mathcal{C}_2 = 0$$

and Y solves the Riccati equation

$$\mathcal{A}(K)Y + Y\mathcal{A}(K)^\top + \mathcal{B}_2(K)\mathcal{B}_2(K)^\top = 0$$

and where

$$\mathcal{A}(K) = A + BKC, \mathcal{B}_2(K) = B_2 + BKD_{y2},$$
$$\mathcal{C}_2(K) = C_2 + D_{2u}KC.$$

Now expanding $K(\theta + d\theta) = K(\theta) + \sum_{\nu=1}^p K_\nu(\theta) d\theta_\nu + \mathcal{O}(\|d\theta\|^2)$, we apply the chain rule to get the following formula

$$\frac{\partial P(\theta)}{\partial \theta_\nu} = 2\,\mathrm{Trace}\Big(\left[B^\top X + D_{2u}^\top \mathcal{C}_2(K)\right] YC^\top + B^\top X \mathcal{B}_2(K) D_{y2}^\top \Big)^\top \frac{\partial K(\theta)}{\partial \theta_\nu}.$$

For example, in SISO PID control, we obtain

$$\frac{\partial K_{\rm pid}(\theta)}{\partial R_i} = \left[\begin{array}{cc|c} 0 & 0 & 1 \\ 0 & 0 & 0 \\ \hline 0 & 0 & 0 \end{array}\right], \quad \frac{\partial K_{\rm pid}(\theta)}{\partial R_d} = \left[\begin{array}{cc|c} 0 & 0 & 0 \\ 0 & 0 & 1 \\ \hline 0 & 0 & 0 \end{array}\right],$$

$$\frac{\partial K_{\rm pid}(\theta)}{\partial D_K} = \left[\begin{array}{cc|c} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \hline 0 & 0 & -1 \end{array}\right], \quad \frac{K_{\rm pid}(\theta)}{\partial \tau} = \left[\begin{array}{cc|c} 0 & 0 & 0 \\ 0 & -1 & 0 \\ \hline 0 & 0 & 0 \end{array}\right],$$

and similar formulae occur in the LQG or H_2 case.

VI. Numerical Experiments

We consider the mass-spring system in Fig. 2, which can be considered as a prototype of a flexible system. We have

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{m_1} & -\frac{f}{m_1} & \frac{k}{m_1} & \frac{f}{m_1} \\ 0 & 0 & 0 & 1 \\ \frac{k}{m_2} & \frac{f}{m_2} & \frac{-k}{m_2} & \frac{-f}{m_2} \end{bmatrix}, B = \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}, D = 0$$

For LQG, covariance matrices of state and output noise are $W = 1$ and $V = 1$, state and input weighting matrices are $Q = C^T C$, $R = I$.

Performance $\mathcal{P}(K_{\rm ltr}(\rho))$ of the LTR controller $K_{\rm ltr}(\rho)$ and performance $\mathcal{P}(K_{\rm obs}(\theta^\sharp))$ of the mixed controller $K_{\rm obs}(\theta^\sharp)$ in logarithmic scale are shown in Fig. 3 (displayed as $\mathcal{P}(K(\rho))$ respectively $\mathcal{P}(K_{2,\infty}(\rho))$). Lower bound, P_{LQG}, is the performance of the nominal LQG controller, which is too low here as the nominal controller $K_{\rm obs}(\theta^*)$ has excellent performance, but lacks robustness. The curve $100r(\rho)$ shows the robustness level of $K_{\rm ltr}(\rho) = K_{\rm obs}(\theta_\rho)$ and $K_{\rm obs}(\theta^\sharp)$ over the same abscissa $\log(\rho)$. As we can see, $K_{\rm obs}(\theta^\sharp)$ is as robust as $K_{\rm ltr}(\rho)$, but has considerably better performance than $K_{\rm ltr}(\rho)$. (The curve $100r(\rho)$ also shows that LTR is not monotone as a rule.)

In order to assess the robustness of the various controllers from a different point of view we compare the stability regions of the LQG controller $K_{\rm obs}(\theta^*)$, the LTR controller $K_{\rm ltr}(\rho)$ and the mixed controller $K_{2,\infty}(\theta^\sharp) = K_{\rm obs}(\theta^\sharp)$ for $\rho = 1e-3$ under the hypothesis that the system has two uncertain parameters m_2 and k. Fig. 4 displays the square of 30% variation in m_2 and k about the nominal values $m_2^{\rm nom}$ and $k^{\rm nom}$, with a dot indicating that the controller continues to stabilize G for that parameter variation $m_2 = m_2^{\rm nom} + \Delta m_2$, $k = k^{\rm nom} + \Delta k$. The square is only covered for the mixed controller $K_{\rm obs}(\theta^\sharp)$.

Fig. 5 illustrates the stability region for the PID study. Here we consider 40% variations in m_2 and k. **(d)** shows nominal H_2 optimal PID controller $K_{\rm pid}(\theta^*)$, which is not parametrically robust over the 40% square. Its robustness $r^* = 17.23$ gives the upper bound. The H_∞ optimal PID controller $K_{\rm pid}(\theta_*)$ shown in **(e)** gives the lower robustness bound $r_* = 4.49$. Its performance $\mathcal{P}(K_{\rm pid}(\theta_*)) = 536.71$ is degraded, and its robustness region is too large, indicating its conservatism. Finally, the mixed H_2/H_∞ PID controller $K_{\rm pid}(\theta^\sharp)$ obtained via (6) is shown in **(f)**. Its stability region covers the 40% square at $r = 17.0$, and the fact that the instability region is tangent to the square indicates that no unnecessary conservatism is produced.

VII. Conclusion

LQG/LTR is a classical procedure which desensibilizes the LQG design in order to improve its robustness. We have shown that the loss of performance in LTR can be considerably reduced if mixed H_2/H_∞ programming is used to optimize H_2 performance subject to a robustness constrained. We have further proposed an similar procedure

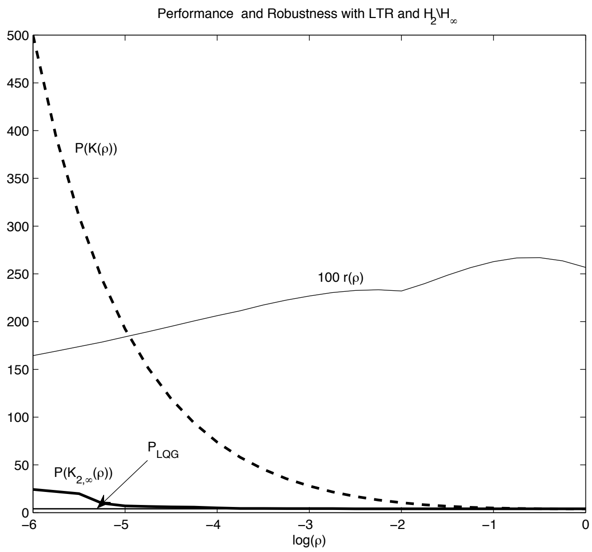

Fig. 3. LQG/LTR study. Decreasing $\rho \to 0$ leads to a dramatic increase of $\mathcal{P}(K_{\rm ltr}(\rho))$ (dashed curve), compared to which the degradation of performance of the mixed H_2/H_∞ controller is only mild. The curve $100r(\rho)$ versus $\log(\rho)$ shows the gain of robustness as $\rho \to 0$.

to compute robustified H_2 optimal PID controllers. Posterior testing shows that our procedure also improves the parametric robustness of the design.

Acknowledgement

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

VIII. Appendix

For the convenience of the reader we recall the classical passage from LQG to H_2 synthesis. Consider the following LQG problem:

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

where w and v are the white noise with covariance matrices W and V, respectively. We denote by Q and R the state and input weighting matrices used to define LQ criterion. Then we build the following plant:

$$\begin{bmatrix} \dot{x} \\ \hline z_2 \\ \hline y \end{bmatrix} = P \begin{bmatrix} x \\ \hline w_2 \\ \hline u \end{bmatrix}$$

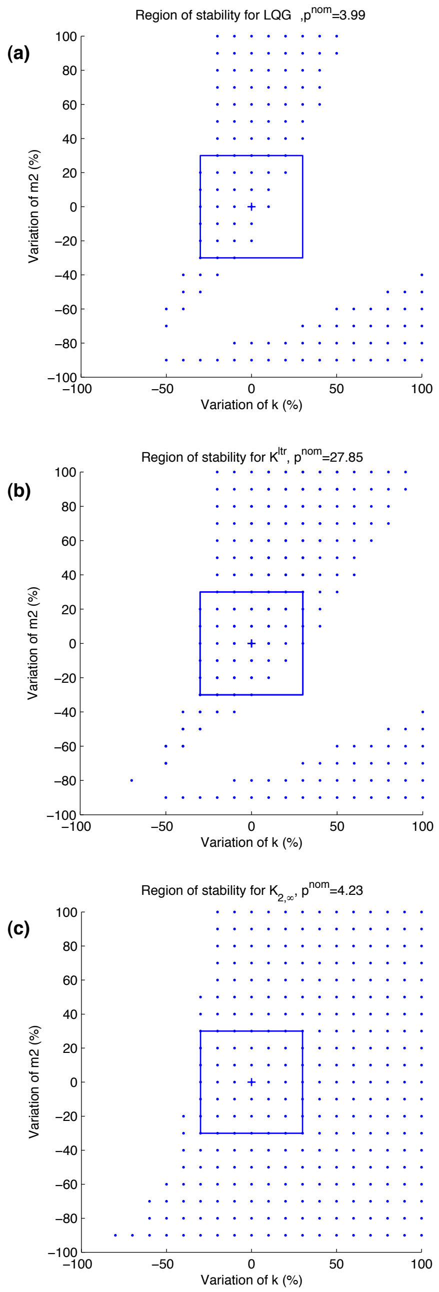

Fig. 4. Stability regions for observer based controllers.

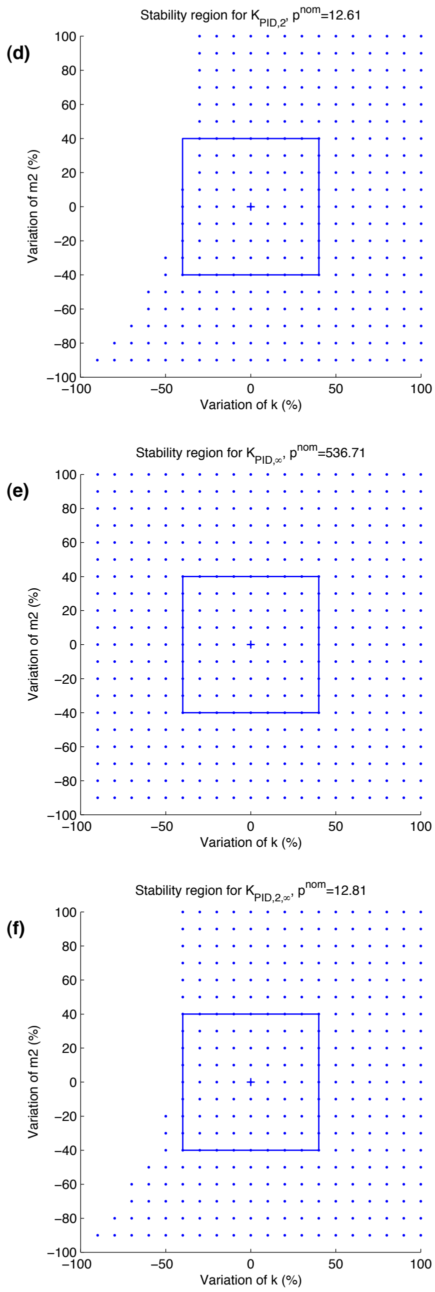

Fig. 5. Stability regions for PID controllers.

where

$$P = \left[\begin{array}{c|c|c} A & B_2 & B \\ \hline C_2 & 0 & D_{2u} \\ \hline C & D_{y2} & 0 \end{array}\right] =$$

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

The original inputs ξ, η and outputs x, u of LQG are encoded as w_2 and z_2 and recovered from the relations

$$\left[\begin{array}{c} \xi \\ \eta \end{array}\right] = \left[\begin{array}{cc} W^{1/2} & 0 \\ 0 & V^{1/2} \end{array}\right] w_2, \quad z_2 = \left[\begin{array}{cc} Q^{1/2} & 0 \\ 0 & R^{1/2} \end{array}\right] \left[\begin{array}{c} x \\ u \end{array}\right].$$

Then we have:

$$\|T_{w_2 \to z_2}(P, K)\|_2^2 = \lim_{T \to \infty} \mathbb{E} \int_0^T \left(x^T Q x + u^T R u\right) \, dt.$$

It is well-known that the LQG controller is

$$K(s) = -K_c(sI - A + BK_c + K_f C)^{-1} K_f$$

with $K_f = P_f C^\top V^{-1}$, $K_c = R^{-1} B^\top P_c$ and P_f, P_c solutions of the algebraic Riccati equations

$$P_f A^\top + A P_f - P_f C^\top V^{-1} C P_f + \Gamma W \Gamma^\top = 0$$

and

$$P_c A + A^\top P_c - P_c B R^{-1} B^\top P_c + C^\top Q C = 0.$$

Moreover (cf. [1]), if $W = I$, $V = \rho V_0$, $\Gamma = B$, and $C(sI - A)^{-1}$ minimum phase, and if the number of inputs equals the number of outputs, then

$$\lim_{\rho \to 0} K(s)C(sI - A)^{-1}B = -K_c(sI - A)^{-1}B$$

because $P_c \to 0$ and $P_f \to 0$ as $\rho \to 0$. One refers to loop transfer recovery because in the limit the loop transfer function $-K_c(sI - A)^{-1}B$ of the LQ controller is obtained. According to Doyle [9] the latter has guaranteed gain margin $\in [0.5, \infty)$ and phase margin $\in [-60^0, 60^0]$. For additional information see [1], [20], [9], [16], [6].

REFERENCES

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

[2] P. APKARIAN, D. NOLL. *Nonsmooth H_∞ synthesis*. IEEE Trans. Autom. Control, vol. 51, 2006, pp. 71 - 86.

[3] 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.

[4] 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.

[5] 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.

[6] B.M. CHEN, A. SABERI, P. SANNUTI. *Loop transfer recovery for general non-minimum phase non-strictly proper system. Part 1: analysis. Part 2: design*. Control Theory and Advanced Technology, vol. 8, no. 1, 1992, pp. 59 – 144.

[7] S.M. DJOUADI, C.D. CHARALAMBOUS, D.W. REPPERGER. *A convex programming approach to the multiobjective H_2/H_∞ problem*. Proceedings of the American Control Conference, may 2002, Anchorage, Alaska, pp. 4315 – 4320.

[8] J.C. DOYLE. *Structured uncertainty in control system design*. Proc. 24th Conference on Decision and Control, Ft. Lauderdale, Fl, 1985, pp. 260 – 265.

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

[10] 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.

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

[12] C.-M. FRANSSON, T. WIK, B. LENNARTSON, M. SAUNDERS, P.-O. GUTMAN. *Nonconservative robust control: optimized and constrained sensitivity functions*. IEEE Transactions on Control System Technology, vol. 17, no. 2, 2009, pp. 298 – 308.

[13] S.M. HINDI, C.D. HASSIBI, S.P. BOYD. *Multiobjective H_2/H_∞ optimal control via finite dimensional Q-parametrization and linear matrix inequalities*. Proc. ACC 1998, pp. 3244 – 3249.

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

[15] M. M. MONAHEMI, J. B. BARLOW, D. P. O'LEARY. *Design of reduced-order observers with precise loop transfer recovery*. Journal of Guidance, Control, and Dynamics, vol. 15, no. 6, 1992, pp. 1320 – 1324.

[16] H.H. NIEMANN, P. SOGAAD-ANDERSEN, J. STOPSTRUP. *Loop transfer recovery: analysis and design for general observer architectures*. Int. Journal of Control, vol. 53, no. 5, 1991, pp. 1177 – 1203.

[17] E. POLAK, Y. WARDI, *A nondifferential optimization algorithm for the design of control systems subject to singular value inequalities over the frequency range*. Automatica, vol. 18, 1982, pp. 267 - 283.

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

[19] A. SABERI, BEN M. CHEN, P. SANNUTI. *Loop transfer recovery: analysis and design*. Communication and Control Engineering Series, Springer Verlag 1993.

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

[21] K. ZHOU. Essentials of robust control. Prentice Hall, 1998.

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

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