

An LFT approach to robust gain scheduling

Jean-François Magni

ONERA-CERT DCSD

2, av Edouard Belin, BP 4025, F-31055 Toulouse Cedex, France

Tel.: +33/(0)5.62.25.29.20, Fax: +33/(0)5.62.25.25.64

magni@cert.fr

Abstract—Traditionally, scheduled controllers are obtained by interpolation of a bank of linear feedback gains. The strategy adopted here is quite different as the controller is directly designed in scheduled form. The key idea consists of designing the feedback gain in LFT (Linear Fractional Transformation) form, the system to be controlled being itself in LFT form. Using an observer-based scheduled feedback and the corresponding Q -parameterization, robust control design can be treated by alternating μ -analysis for worst case identification and multimodel control for simultaneous treatment of the worst cases.

I. INTRODUCTION

This paper illustrates feedback gain synthesis in LFT form. An LFT model can be viewed to some extent as a symbolic representation involving only rational operations. If a system depending on some parameters is given in symbolic form, it is natural to attempt to design a feedback law in symbolic form which would naturally lead to a scheduled feedback gain. However, such laws are not practically implemented on account of the numerical difficulties that would be encountered. LFT modeling and then, feedback design in LFT form does not present these drawbacks for several reasons. The main reason is that most computation can be performed off-line in the LFT case, in addition, several tools are available in order to reduce significantly the complexity (see [1], [2], [3]).

Using the object-oriented LFRT Toolbox [3], some matrix operations relative to matrices (addition, multiplication, inversion, concatenation, conjugation, real and imaginary parts, kernel computation) can also be applied to LFT objects. Therefore, deriving a feedback gain in LFT form becomes straightforward. Indeed, any classical algorithm involving only matrix operations that can also be applied to LFT objects, is potentially an algorithm for LFT-scheduled gain design because it suffices to apply the considered algorithm to the LFT version of the matrices describing the system. Eigenstructure assignment ([4]) satisfies the above requirement, it is this technique that will be considered in this paper.

Most alternative contributions to gain scheduling (see [5], [6] and the references therein) consider *a posteriori* interpolation or realization of feedback gains. Interpolation becomes cumbersome when there are too many parameters (3 or more). On the contrary, LFT-scheduled gains become

simpler when there are many parameters. The LPV approach is an alternative approach to design LFT-scheduled gains ([7], [8], [9]). The LPV approach gives some stability guarantees but leads to conservative control design. In contrast the technique proposed in this paper avoids conservatism but stability with respect to neglected nonlinear effects must be checked *a posteriori*.

In this paper, we shall assume that the system to be controlled is in LFT form. Usually, the LFT model is a continuum of linearized models. It can also be a quasi-LPV model (e.g., the missile LPV-model in [5]). We shall use extensively μ -analysis because this analysis tool is efficient and often non-conservative. However, using μ -analysis it must be assumed that the scheduling parameters vary slowly. The parameters of the LFT model will be divided into two groups

- the parameters that can be measured: these parameters will be used for scheduling
- the uncertain parameters: for these parameters robustness will be considered.

The proposed algorithm can be sketched as follows. *First*, a nominal observer-based controller is designed by eigenstructure assignment ([4]). Note that this initial feedback law might reproduce a feedback gain that has been designed using any alternative technique ([4], page 284). *Then*, this controller is transformed to an LFT-scheduled gain by replacing the nominal state-space matrices by their LFT form. This step is discussed in §III. Note that the assigned eigenvalues can also be LFTs. *The third step* consists of improving the robustness of the scheduled controller. The system and its controller being in LFT form, μ -analysis can be used to detect the worst cases (lower bound of μ) and the worst pole (i.e., the pole of the worst case that crosses the imaginary axis). Then it will suffice to move to the left the worst poles in order to flatten the μ curve. This can be done because eigenstructure assignment can easily be adapted to multimodel control. We shall alternate μ -analysis for worst case identification and multimodel control for treating together all the worst cases (see [10]). Q -parameterization (for Luenberger observer-based controller) is used here for a better control of the nominal performance during the iterations for robustness improvement. The details

are given in §IV.

Designing LFT-scheduled gains requires some attention in order to avoid high complexity controllers, in addition, “on line” computation of an LFT requires the inversion of a matrix depending on scheduling parameters, this inversion must remain feasible (well-posedness). These two problems specific to LFT-scheduled gains are considered all along this paper.

II. NOTATIONS AND GENERALITIES

Only real parameter variations are considered. The parameters are denoted $\delta_1, \dots, \delta_r$. Δ stands for the matrix with diagonal blocks as follows:

$$\Delta = \text{Diag}\{\delta_1 I_{n_1}, \dots, \delta_r I_{n_r}\} \quad (1)$$

It is assumed that the parameters are normalized, *i.e.*, $-1 \leq \delta_i \leq 1$, $i = 1, \dots, r$. The matrix Δ is divided into two diagonal sub-matrices: Δ_m corresponding to the parameters that can be measured, and Δ_u corresponding to the uncertain parameters.

$$\Delta = \begin{bmatrix} \Delta_m & 0 \\ 0 & \Delta_u \end{bmatrix} \quad (2)$$

The size k of the matrices I_k varies with each LFT object considered in this paper. *In order to alleviate notation, we shall use Δ, Δ_m and Δ_u in all cases regardless of the block sizes.* The complexity of the LFT objects considered in this paper will be defined as the size of the square matrix Δ . We shall speak of “ Δ -complexity”.

Modelling systems from equations of physics leads to symbolic representations of the form $y = G(s, \Delta)u$. Modelling this system in LFT form consists of finding a multi-dimensional realization as depicted in Figure 1. The toolbox of [3] can be used for automatic realizations from symbolic forms.

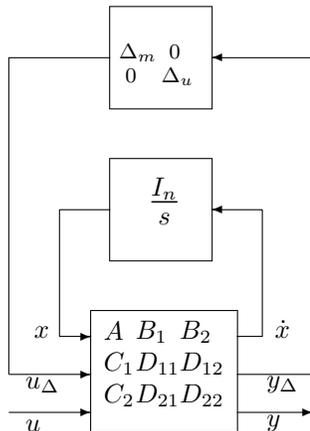


Fig. 1. LFT representation of a linear uncertain system

In this paper we shall use the following state-space representation

$$\begin{cases} \dot{x} = A(\Delta)x + B(\Delta)u \\ y = C(\Delta)x + D(\Delta)u \end{cases}$$

where x is the state vector, y the output vector and u the input vector. For a fixed value of Δ : $A(\Delta) \in \mathbb{R}^{n \times n}$,

$B(\Delta) \in \mathbb{R}^{n \times m}$, $C(\Delta) \in \mathbb{R}^{p \times n}$ and $D(\Delta) \in \mathbb{R}^{p \times m}$. Using the operator \mathcal{F}_u defined as follows:

$$\mathcal{F}_u \left(\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \Delta \right) = M_{21}\Delta(I - M_{11}\Delta)^{-1}M_{12} + M_{22}$$

after reordering the sub-matrices of Figure 1, clearly:

$$\begin{bmatrix} A(\Delta) & B(\Delta) \\ C(\Delta) & D(\Delta) \end{bmatrix} = \mathcal{F}_u \left(\begin{bmatrix} D_{11} & C_1 & D_{12} \\ B_1 & A & B_2 \\ D_{21} & C_2 & D_{22} \end{bmatrix}, \Delta \right) \quad (3)$$

In the LFT model that will be used for gain scheduling in §III the uncertain parameters are frozen to their nominal value. This system can be extracted from (3) as it suffices to give to Δ_u the numerical value corresponding to the nominal case and to close the Δ_u -loop. The corresponding system and LFT-scheduled controller will be denoted

$$\begin{bmatrix} A(\Delta_m) & B(\Delta_m) \\ C(\Delta_m) & D(\Delta_m) \end{bmatrix}; u(s) = K(\Delta_m) y(s) \quad (4)$$

where

$$K(\Delta_m) = K_{22} + K_{21}\Delta_m(I - K_{11}\Delta_m)^{-1}K_{12} \quad (5)$$

for some matrices K_{11} , K_{12} , K_{21} and K_{22} (note that we shall use observer-based controllers, therefore, we shall not consider directly dynamic LFT-scheduled gain of the form $K(s, \Delta_m)$).

For implementation of such a feedback gain, we must check that $(I - K_{11}\Delta_m)$ can be inverted for all admissible values of Δ_m (*i.e.* for all admissible operating point). It is well known that this matrix can be inverted for all Δ_m structured as in (1) and such that $\|\Delta_m\| < 1$ if and only if

$$\mu(K_{11}) < 1 \quad (6)$$

this is a *well-posedness* condition (see [11], [12] for the definition and computation of μ) that will be considered in §III-D. It is natural to define the *well-posedness radius* as

$$\mathcal{R}_{wp} \left(\mathcal{F}_u \left(\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \Delta \right) \right) = \frac{1}{\mu(M_{11})} \quad (7)$$

Similarly, the *non-singularity radius* can be defined as the well-posedness radius of the inverse of an LFT (if M_{22} is singular the non-singularity radius is equal to zero):

$$\mathcal{R}_{ns} \left(\mathcal{F}_u \left(\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \Delta \right) \right) = \frac{1}{\mu(M_{11} - M_{12}M_{22}^{-1}M_{21})} \quad (8)$$

III. LFT GAIN SCHEDULING

In this section is described an algorithm for observer-based control design. The details can be found in [4]. It is presented in such a way that all used operations can be applied either to matrices or to LFTs (LFT operations are described in Chapter 9 of [3]). Therefore, this algorithm can be used either in the matrix or in the LFT case but in the second case there are some specificities discussed in §III-C.

We prefer to detail Luenberger observers rather than observers with Kalman filter structure, because Luenberger observers offer more flexibility, in particular, it is not necessary to observe all the states (in addition observers with Kalman filter structure are special cases of Luenberger observers).

A. Observer design

Let us consider the system of Equation (4) in which the dependency with respect to Δ_m is omitted. An n_o -order Luenberger observer is a triple (U, T, Π) with $U \in \mathbb{R}^{n_o \times n}$, $T \in \mathbb{R}^{n_o \times p}$, $\Pi \in \mathbb{R}^{n_o \times n_o}$ and Π stable, satisfying

$$UA + TC = \Pi U \quad (9)$$

The corresponding observer is defined by

$$\dot{\hat{\mathbf{z}}} = \Pi \hat{\mathbf{z}} - T\mathbf{y} + (UB + TD)\mathbf{u} \quad (10)$$

in which $\hat{\mathbf{z}} \in \mathbb{R}^{n_o}$ is an estimate of $U\mathbf{x}$ and the estimation error dynamic is given by $\dot{\hat{\mathbf{z}}} - \dot{\mathbf{z}} = \Pi(\hat{\mathbf{z}} - \mathbf{z})$

Procedure 3.1: Design procedure of an n_o -order observer.

- 1) Choose n_o desired observation dynamics π_i , $i \in \{1 \dots n_o\}$ and n_o sets of matrices $e_i \in \mathbb{R}^p$, $e_i \neq 0$, $E_i \in \mathbb{R}^{n \times p}$ and $F_i \in \mathbb{R}^{n \times p}$.
- 2) Compute (u_i, t_i) , $i = 1, \dots, n_o$ satisfying:

$$\begin{bmatrix} u_i & t_i \end{bmatrix} = \begin{bmatrix} 0 & e_i \end{bmatrix} \begin{bmatrix} A - \pi_i I & E_i \\ C & F_i \end{bmatrix}^{-1} \quad (11)$$

- 3) Build U , T and Π as follows

$$U = \begin{bmatrix} u_1 \\ \vdots \\ u_{n_o} \end{bmatrix}; T = \begin{bmatrix} t_1 \\ \vdots \\ t_{n_o} \end{bmatrix}; \Pi = \text{Diag}\{\pi_1 \dots \pi_{n_o}\} \quad (12)$$

Note that n_o can be much less than the number of states (for minimum order observer-based control design) or much larger (redundancy for multimodel control design). It remains to connect the corresponding observer to the original system as depicted in Figure 2. The choice of the matrices e_i , E_i and F_i is a classical modal control problem.

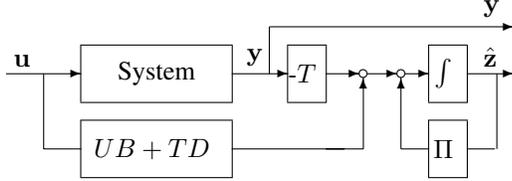


Fig. 2. System with observer to be used for feedback design.

The state-space form of the system of Figure 2 is:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\hat{\mathbf{z}}} \\ \mathbf{y} \\ \dot{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} A & 0 \\ -TC & \Pi \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \hat{\mathbf{z}} \end{bmatrix} + \begin{bmatrix} B \\ UB \end{bmatrix} \mathbf{u} \quad (13)$$

$$\begin{bmatrix} \mathbf{y} \\ \dot{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} C & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \hat{\mathbf{z}} \end{bmatrix} + \begin{bmatrix} D \\ 0 \end{bmatrix} \mathbf{u}$$

The following lemma states that the transfer function matrix between the input and the estimating signal $\hat{\mathbf{z}}$ is equal to the one between the input and the corresponding estimated signal \mathbf{z} . It is a form of the ‘‘Separation Principle’’. See for example [4] for a proof. This well-known result will be used for Q -parameterization in § IV-A.

Lemma 3.2: The transfer function matrix between \mathbf{u} and $\hat{\mathbf{z}}$ (see System (10)) is equal to $U(sI - A)^{-1}B$, which is also the transfer function matrix between \mathbf{u} and \mathbf{z} . The eigenvalues of Π are uncontrollable.

In the next section (Procedure 3.3), the considered feedback law has the following form: $\mathbf{u} = K_y \mathbf{y} + K_z \hat{\mathbf{z}}$. From the ‘‘separation principle’’ (Lemma 3.2), it is clear that the system of Equation (13) is equivalent to the following one:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} A \\ C \\ U \end{bmatrix} \mathbf{x} + \begin{bmatrix} B \\ D \\ 0 \end{bmatrix} \mathbf{u} \quad (14)$$

Note that in the standard observer approach, the observer design and the controller design (state feedback) are independent. In the proposed design procedure, the observer must be designed first, then the observed variables are considered as additional measurements used for designing an output feedback controller.

B. Observer-based control design

After having connected the observer (U, T, Π) of Algorithm 3.1 as shown in Figure 2, applying the ‘‘separation principle’’ we obtain the system of Equation (14). The following output feedback design procedure is dual to some extent to Procedure 3.1. Let us denote (14) as follows

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \tilde{\mathbf{y}} \end{bmatrix} = \begin{bmatrix} A \\ \tilde{C} \end{bmatrix} \mathbf{x} + \begin{bmatrix} B \\ \tilde{D} \end{bmatrix} \mathbf{u} \quad (15)$$

in which $\tilde{C} \in \mathbb{R}^{(p+n_o) \times n}$ and $\tilde{D} \in \mathbb{R}^{(p+n_o) \times m}$

Procedure 3.3: Design procedure for eigenstructure assignment. Eigenstructure assignment by output feedback relative to System (15) can be performed as follows :

- 1) Choose $n_c \leq \min(n, p + n_o)$ desired closed-loop eigenvalues λ_i and the corresponding matrices $e_i \in \mathbb{R}^m$, $e_i \neq 0$, $E_i \in \mathbb{R}^{m \times n}$ and $F_i \in \mathbb{R}^{m \times m}$.
- 2) Compute (v_i, w_i) , $i = 1, \dots, n_c$ satisfying:

$$\begin{bmatrix} v_i \\ w_i \end{bmatrix} = \begin{bmatrix} A - \lambda_i I & B \\ E_i & F_i \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ e_i \end{bmatrix} \quad (16)$$

- 3) Let us denote

$$V = [v_1 \dots v_{n_c}], W = [w_1 \dots w_{n_c}],$$

$$\Lambda = \text{diag}[\lambda_1 \dots \lambda_{n_c}]$$

Compute K satisfying:

$$K[\tilde{C}V + \tilde{D}W] = W \quad (17)$$

Usually, e_i , E_i and F_i are chosen in order to satisfy some decoupling properties ([4]). It is sometimes interesting to consider $E_i = 0$ and $F_i = I$, in this case $w_i = e_i$ and (16) becomes

$$v_i = (A - \lambda_i I)^{-1} B w_i \quad (18)$$

In this case, the vectors w_i can be chosen from values assigned by an alternative design technique (for example LQ or \mathcal{H}_∞). It remains to discuss the choice of n_c that is the number of assigned poles. Generally, pole assignment techniques must assign a number n_c of poles equal to the number of dominant modes. For computing K , especially in the LFT case, we shall assume that n_o is chosen so that $n_c = p + n_o$ in order to have a square matrix that can be inverted. In that way

$$K = W(\tilde{D}V + \tilde{C}W)^{-1} \quad (19)$$

For Q -parameterization (§IV-A), the intersection between the row span of U and C must not reduce to zero. In this case, it will be assumed that Algorithm 3.1 is applied in two steps in order to add rows to the matrix U .

C. Problems specific to LFT-scheduled gains

As mentioned in the introduction of this section, for designing a scheduled feedback gain in LFT form, it suffices to replace the matrices A, B, C, D by $A(\Delta_m), B(\Delta_m), C(\Delta_m), D(\Delta_m)$. The complete procedure consists of applying Algorithm 3.1 to $A(\Delta_m), B(\Delta_m), C(\Delta_m), D(\Delta_m)$ with n_o (observer order) such that $p + n_o$ is equal to the number of dominant poles n_c . Then, connect the observer as shown in Figure 2 and apply Algorithm 3.3 to the corresponding system of Equation (14). By the way, the matrices V, W, U and T become also LFTs respectively denoted $V(\Delta_m), W(\Delta_m), U(\Delta_m)$ and $T(\Delta_m)$. Note that λ_i and π_i might also be in LFT form.

However, there are two problems that do not appear in the matrix case:

- the size of the Δ -matrix of the LFT gain might become very large. See [3], § 7.1.2 for a description of several tricks to be used for avoiding too large Δ -matrices.
- a well-posedness condition must be satisfied (see (6)), this issue is addressed in §III-D. Note that in the matrix case, the inversions appearing in Algorithms 3.1 and 3.3 are generically feasible ([13]). In the LFT case we don't accept any degeneracy in the unit ball, even in an algebraic subspace of zero measure. Checking nondegeneracy in the unit ball is a problem of wellposedness.

D. Well-posedness

The next result gives an *a priori* well-posedness guarantee in the state-feedback / single-input case.

Lemma 3.4: If the single-input pair $(A(\Delta_m), B(\Delta_m))$ is uniformly controllable in the unit ball, all LFT-scheduled state feedback gains assigning distinct poles have an equivalent form that is well-posed in the unit ball.

Proof. Note that well-posedness depends on the ways the gain is computed, this is why “an equivalent form” is mentioned in the formulation of the lemma. There exists a simple direct proof of this statement however the proof of Proposition 3.7 can also be invoked because, assuming that the system is controllable in the unit ball, the well-posedness radius in (25) is known to be larger than one (from classical pole assignment theory).

Corollary 3.5: If the single-input pair $(A(\Delta_m), B(\Delta_m))$ is uniformly controllable in the unit ball and if the single-output pair $(A(\Delta_m), C(\Delta_m))$ is uniformly observable in the unit ball, all LFT-scheduled observer-based feedback laws with Kalman filter structure assigning distinct poles admit an equivalent form that is well-posed in the unit ball.

In the proposed design algorithm 3.3, there are two kinds of operations that might reduce the well-posedness radius of the gain. The first group of operations is the inversion in (16)-(18). Lemma 3.7 below shows that we can ignore

this problem. The second kind of operations is the inversion appearing in (21), for this second problem we have just a partial understanding, see Lemma 3.4 and Corollary 3.5.

Lemma 3.6: The feedback $K(\Delta_m)$ gain of Equation (19)

$$K(\Delta_m) = W(\Delta_m) \left(\begin{bmatrix} C(\Delta_m) & D(\Delta_m) \\ U(\Delta_m) & 0 \end{bmatrix} \begin{bmatrix} V(\Delta_m) \\ W(\Delta_m) \end{bmatrix} \right)^{-1} \quad (20)$$

can also be computed as follows: $K(\Delta_m) = -$

$$\begin{bmatrix} I_{p-m} & 0 \\ 0 & I_m \end{bmatrix} \left[\begin{bmatrix} C(\Delta_m) & D(\Delta_m) \\ U(\Delta_m) & 0 \end{bmatrix} \begin{bmatrix} V(\Delta_m) \\ W(\Delta_m) \end{bmatrix} \middle| \begin{bmatrix} 0 \\ I_m \end{bmatrix} \right]^{-1} \begin{bmatrix} I_{\tilde{p}} \\ 0 \end{bmatrix} \quad (21)$$

Proposition 3.7: Using formula (21) rather than (20), the invertibility condition for computing v_{i,Δ_m} and w_{i,Δ_m} arising in Equations (16) (or (18)) can be ignored.

Proof. We shall compute the feedback gain in two ways. First, the natural way for which $V(\Delta_m)$ and the $W(\Delta_m)$ are considered individually as in Equation (20) and then the improved way based on Equation (21). Let

$$\begin{bmatrix} \tilde{C}(\Delta_m)V(\Delta_m) & \tilde{D}(\Delta_m)W(\Delta_m) \\ 0 & W(\Delta_m) \end{bmatrix} = \mathcal{F}_u \left(\left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21V} & M_{22V} \\ M_{21W} & M_{22W} \end{array} \right], \Delta_m \right) \quad (22)$$

- *Natural way.* From the upper part of (22)

$$\tilde{C}(\Delta_m)V(\Delta_m) + \tilde{D}(\Delta_m)W(\Delta_m) = \mathcal{F}_u \left(\left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21V} & M_{22V} \end{array} \right], \Delta_m \right)$$

and from the lower part

$$W(\Delta_m) = \mathcal{F}_u \left(\left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21W} & M_{22W} \end{array} \right], \Delta_m \right)$$

Then from (20), using the LFT inversion and the multiplication formulas ([3]) we obtain $K(\Delta_m) =$ having the following LFT structure:

$$\left[\begin{array}{cc|c} M_{11} & -M_{12}M_{22V}^{-1}M_{21} & M_{12}M_{22V}^{-1} \\ 0 & M_{11} - M_{12}M_{22V}^{-1}M_{21V} & M_{12}M_{22V}^{-1} \\ \hline M_{21W} & -M_{22W}M_{22V}^{-1}M_{21V} & M_{22W}M_{22V}^{-1} \end{array} \right] \quad (23)$$

So, the well-posedness of this form of the gain is limited by the values of Δ_m such that

$$\det \left(I - \begin{bmatrix} M_{11} & -M_{12}M_{22V}^{-1}M_{21} \\ 0 & M_{11} - M_{12}M_{22V}^{-1}M_{21V} \end{bmatrix} \begin{bmatrix} \Delta_m & 0 \\ 0 & \Delta_m \end{bmatrix} \right) = 0$$

We recognize here two conditions: $\det(I - M_{11}\Delta_m) \neq 0$ and $\det(I - M_{11} + M_{12}M_{22V}^{-1}M_{21V}\Delta_m) \neq 0$. So the well-posedness radius satisfy $\mathcal{R}_{wp}(K(\Delta_m)) =$

$$\min \left(\mathcal{R}_{wp} \left[\begin{bmatrix} V(\Delta_m) \\ W(\Delta_m) \end{bmatrix} \right], \mathcal{R}_{ns}(\tilde{C}(\Delta_m)V(\Delta_m) + \tilde{D}(\Delta_m)W(\Delta_m)) \right) \quad (24)$$

- *Using Equation (21).* From (22)

$$\left[\begin{bmatrix} \tilde{C}(\Delta_m) & \tilde{D}(\Delta_m) \\ 0 & I_m \end{bmatrix} \begin{bmatrix} V(\Delta_m) \\ W(\Delta_m) \end{bmatrix} \middle| \begin{bmatrix} 0 \\ 0 \\ I_m \end{bmatrix} \right] = \mathcal{F}_u \left(\left[\begin{array}{c|c} M_{11} & M_{12} & 0 \\ \hline M_{21V} & M_{22V} & 0 \\ M_{21W} & M_{22W} & I \end{array} \right], \Delta_m \right)$$

after inversion the following LFT form is obtained:

$$\left[\begin{array}{c|c} M_{11} - M_{12}M_{22V}^{-1}M_{21V} & M_{12}M_{22V}^{-1} & 0 \\ \hline M_{21V}M_{22V}^{-1} & M_{22V}^{-1} & 0 \\ M_{21W} - M_{22W}M_{22V}^{-1}M_{21V} & -M_{22W}M_{22V}^{-1} & I \end{array} \right]$$

so:

$$\mathcal{R}_{\text{wp}}(K(\Delta_m)) = \tilde{\mathcal{R}}_{\text{ns}} \left(\tilde{C}(\Delta_m)V(\Delta_m) + \tilde{D}(\Delta_m)W(\Delta_m) \right) \quad (25)$$

to be compared to (24) ■.

In the observer case the matrices $[U(\Delta_m) \ T(\Delta_m)]$ must be well-posed, this point is discussed in ([3], § 8.2.2).

Conclusion on well-posedness. From Proposition 3.7 and from the above remark, the inversions involved in the computation of the vectors $v_i(\Delta_m)$, $w_i(\Delta_m)$, $u_i(\Delta_m)$ and $t_i(\Delta_m)$ are not troublesome. Concerning the inversion problem of $\tilde{C}(\Delta_m)V(\Delta_m) + \tilde{D}(\Delta_m)W(\Delta_m)$ we have the result of Corollary 3.5 but the validity of this result is restricted to the SISO case. However, it is always possible to check well-posedness *a posteriori* using (6).

Note that the fundamental difference between the SISO and MIMO cases is that in the SISO case, Corollary 3.5 extends *exact* properties of standard systems. A similar result in the MIMO case would extend *generic* properties (see, e.g. [13]). In other words, this problem would consist of checking the exclusion of an algebraic subspace of zero measure (genericity) from the unit ball (with non-zero measure). Up to our knowledge, the best technique available for that purpose is μ -analysis (so, we are back to the *a posteriori* test of Equation (6)).

IV. ROBUSTNESS

It is assumed that an observer-based LFT-scheduled gain has been designed as detailed in the previous section. This controller depends on Δ_m and it has been computed fixing the uncertain parameters of Δ_u to their nominal values. The controller being in LFT form, the closed-loop system is itself an LFT model so, can be analyzed by μ -analysis (more precisely by skew- μ -analysis, see §IV-B).

In order to preserve at least the stability of the scheduled gain for all admissible variations of Δ_m and Δ_u nominal, we shall use Q -parameterization. Here again, Q -parameterization involves matrix operations that can be applied to LFT objects, therefore, the scheduled nature of the feedback gain is considered without additional difficulties. The parameter Q will be designed as a multimodel feedback gain.

The overall robustness improvement technique can be sketched as follows. First, the system with its LFT-scheduled gain is adapted to Q -parameterization (§IV-A). Then, for designing Q , we shall alternate worst cases identification (by skew- μ -analysis) and multimodel design relative to the worst cases, the objective being to flatten the skew- μ curve (§IV-B).

A. Q -parameterization for Luenberger observers

Q -parameterization consists of adding a stable feedback gain Q between two points of a regulation scheme with

stability preservation. In order to obtain this effect, it suffices to identify an input point (denoted \mathbf{u}_Q) and an output point (denoted \mathbf{y}_Q) so that the transfer function matrix from \mathbf{u}_Q to \mathbf{y}_Q is equal to zero. Then,

$$\mathbf{u}_Q = Q \mathbf{y}_Q$$

defines a new design parameter (Q) that preserves nominal stability. This property is easily checked, for example, consider an input signal \mathbf{u}_P (or \mathbf{u}) and an output signal \mathbf{y}_P (or \mathbf{y}) we have

$$\begin{bmatrix} \mathbf{y}_P \\ \mathbf{y}_Q \end{bmatrix} = \begin{bmatrix} T_{\mathbf{u}_P \rightarrow \mathbf{y}_P} & T_{\mathbf{u}_Q \rightarrow \mathbf{y}_P} \\ T_{\mathbf{u}_P \rightarrow \mathbf{y}_Q} & T_{\mathbf{u}_Q \rightarrow \mathbf{y}_Q} \end{bmatrix} \begin{bmatrix} \mathbf{u}_P \\ \mathbf{u}_Q \end{bmatrix}$$

with $\mathbf{u}_Q = Q \mathbf{y}_Q$ and $T_{\mathbf{u}_Q \rightarrow \mathbf{y}_Q} = 0$ we have

$$\mathbf{y}_P = (T_{\mathbf{u}_P \rightarrow \mathbf{y}_P} + T_{\mathbf{u}_Q \rightarrow \mathbf{y}_P} Q T_{\mathbf{u}_P \rightarrow \mathbf{y}_Q}) \mathbf{u}_P$$

in which all transfers are stable.

Q -parameterization for the observers of §III-A (note that standard observers are a special case of Luenberger observers with $Q_z = 0$ and $U \in \mathbb{R}^{n \times n}$). In all observer-based control system, the transfer function matrix between the plant input and the observation error is equal to zero. In the Luenberger-like observer of Figure 3, the observed signal is $\mathbf{z} = U\mathbf{x}$ and its estimate is the vector $\hat{\mathbf{z}}$. Therefore, the transfer from the plant input \mathbf{u} to $U\mathbf{x} - \hat{\mathbf{z}}$ is equal to zero.

Unfortunately, the signal $U\mathbf{x} - \hat{\mathbf{z}}$ is not available from measurements, so, we must find in the vector \mathbf{y} some components of $U\mathbf{x}$. For that purpose, let us define two matrices Q_y et Q_z of maximal rank so that

$$\begin{bmatrix} Q_y & Q_z \end{bmatrix} \begin{bmatrix} C \\ U \end{bmatrix} = 0 \quad (26)$$

(assuming that there exist non-zero matrices Q_y and Q_z). Let us check that the signal $Q_z(\hat{\mathbf{z}} - U\mathbf{x})$ is available from measurements. $Q_z(\hat{\mathbf{z}} - U\mathbf{x}) = Q_z\hat{\mathbf{z}} + Q_zC\mathbf{x}$ because $Q_yC + Q_zU = 0$. As $C\mathbf{x} = \mathbf{y} - D\mathbf{u}$: $Q_z(\hat{\mathbf{z}} - U\mathbf{x}) = Q_z\hat{\mathbf{z}} + Q_z(\mathbf{y} - D\mathbf{u})$ with $Q_u = -Q_yD$, $Q_z(\hat{\mathbf{z}} - U\mathbf{x}) = Q_z\hat{\mathbf{z}} + Q_y\mathbf{y} + Q_u\mathbf{u}$. Finally, $Q_z(\hat{\mathbf{z}} - U\mathbf{x})$ is a component of the estimation error that is available from measurements because it depends on \mathbf{y} , \mathbf{u} , and $\hat{\mathbf{z}}$. This remark justifies Figure 3 in which

$$\mathbf{y}_Q = Q_y\mathbf{y} + Q_u\mathbf{u} + Q_z\hat{\mathbf{z}}; \quad \mathbf{u}_Q = \mathbf{u} \quad (27)$$

Q -parameterization for LFT-scheduled observer-based feedback. The computation required for Q -parameterization reduces to solving (26). Equation (26) can be written in an equivalent way that only involves classical operations that are compatible with LFTs:

$$\begin{bmatrix} Q_y(\Delta_m) & Q_z(\Delta_m) \end{bmatrix} = [0 \ I] \begin{bmatrix} C(\Delta_m) & E \\ U(\Delta_m) & F \end{bmatrix}^{-1} \quad (28)$$

The choice of E and F must optimize the well-posedness radius of $[Q_y(\Delta_m) \ Q_z(\Delta_m)]$. It can be done by choosing E and F as the orthogonal complement at $\Delta_m = 0$.

In the next section, the feedback Q will be designed in order to improve the robustness with respect to the uncertain variations of Δ_u .

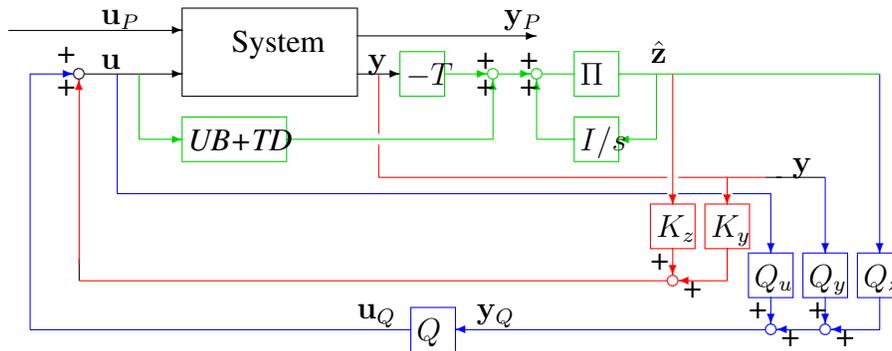


Fig. 3. Illustration of the Q-parameterization for Luenberger observers

B. Algorithm for robustness improvement

For robustness improvement of the LFT-scheduled control law, it is suggested to use a technique similar to the one proposed in [10]. This technique consists of using in alternation:

- μ -analysis for identifying worst cases and worst poles. The worst cases can be identified from the lower bound of μ at the peak values of the μ -curve. The corresponding worst poles are on the imaginary axis at the frequencies corresponding to the peaks.
- synthesis of the feedback gain Q by a multimodel approach. The models treated together are the worst cases identified using μ -analysis. Reducing the peak values of the μ -curve consists of shifting to the left the worst poles of the worst cases.

More details are given below as a series of comments.

It is worth noting that the proposed algorithm is feasible because the scheduled gain is in LFT form, therefore the closed-loop system remains in LFT form, and then, μ -analysis can be applied. The main difference with respect to the technique of [10] is that here, there are two kinds of parameters: Δ_m and Δ_u . As the feedback is scheduled with respect to Δ_m , robustness analysis is only relevant for variations of Δ_u while Δ_m remains in the unit ball. Therefore, what we called μ -analysis is in fact skew- μ -analysis. The toolbox [14] makes available the required tools for skew- μ -analysis. This toolbox emphasises in particular μ -analysis with respect to *real* uncertain parameters.

The observer designed using Procedure 3.1 is used first for LFT-scheduled gain design (see (21)) and then for Q-parameterization (see (26)). In the first case U must be a row complement of C but in the second case, some redundancy is required. In fact, Procedure 3.1 can be used only once with redundancy in view of solving (26), but when Procedure 3.3 is considered, only a selection of rows of U is taken into account.

Several multimodel design techniques are proposed in [4]. Note that Procedure 3.3 can be used for designing Q (the system (16) must be replaced by the transfer function matrix from u_Q to y_Q of Figure 3). This design procedure is compatible with multimodel design because it consists of linear constraints (Equation (17)) applied to the gain being

computed, keeping in mind that for each assignment the matrices “ A, B, C, D ” might be relative to different models.

Both steps of the proposed robustness improvement technique (skew- μ -analysis and multimodel design of Q) can be organized in an automatic algorithm. Unfortunately, the lower bound computation of μ is not very reliable, therefore it is better to iterate manually and check the validity of worst cases at each step. Usually two or three iterations suffice for reducing significantly the peak values of the skew- μ curve.

REFERENCES

- [1] P. Lambrechts, J. Terlouw, S. Bennani, and M. Steinbuch. Parametric uncertainty modeling using LFTs. *In Proc. American Control Conference, San Francisco, CA*, pages 267–272, 1993.
- [2] C. Beck and J. Doyle. A necessary and sufficient minimality condition for uncertain systems. *IEEE Transactions on Automatic Control*, AC-44:1802–1813, october 1999.
- [3] J.F. Magni. Linear Fractional Representations with a Toolbox: Modelling, order reduction, gain scheduling, Version 1.3. Technical Report TR 6/08162 DCSD, ONERA-CERT: www.cert.fr/dcsd/idco/perso-Magni/booksandtb.html, July 2004.
- [4] J.F. Magni. *Robust Modal Control with a Toolbox for use with MATLAB*. Kluwer Academic/Plenum Publishers, April 2002.
- [5] W.J. Rugh and J.S. Shamma. Research on gain scheduling. *Automatica*, 36(10):1401–1425, 2000.
- [6] D.J. Leith and W.E. Leithead. Survey of gain-scheduling analysis and design. *Int. J. Control*, 73:1001–1025, 2000.
- [7] P. Apkarian and P. Gahinet. A convex characterisation of gain-scheduled H_∞ controller. *IEEE Transactions on Automatic Control*, 5(40):853–863, 1995.
- [8] A. Packard. Gain-scheduling via linear fractional transformations. *Systems & Control Letters*, 22:79–92, 1994.
- [9] A. Helmersson. μ -synthesis and lft gain scheduling with mixed uncertainties. *In Proceedings of the 3rd European Control Conference*, volume 1, pages 153–158, Rome, Italy, Sep 1995.
- [10] J.F. Magni, Y. Le Gorrec, and C. Chiappa. A multimodel-based approach to robust and self-scheduled control design. *In Proc. 37th I.E.E.E. Conf. Decision Contr., Tampa, Florida*, pages 3009–3014, 1998.
- [11] J.C. Doyle. Analysis of feedback systems with structured uncertainties. *IEE Proceedings, Part D, Control Theory and Applications*, 129(6):212–250, 1982.
- [12] P.M. Young and J.C. Doyle. Computation of μ with real and complex uncertainties. *in Proceedings Conference on Decision and Control*, pages 1230–1235, 1990.
- [13] C. Champetier and J.F. Magni. On eigenstructure assignment by gain output feedback. *SIAM J. Contr. Optimiz.*, 29(4):848–865, July 1991.
- [14] G. Ferreres and J.M. Biannic. Skew Mu Toolbox (smt): improvements and additional tools. Web publication, ONERA-DCSD: http://www.cert.fr/dcsd/idco/perso/Biannic/SMT_v2.pdf, October 2004.