A LYAPUNOV APPROACH TO $H_2$ ITERATIVE ADJUSTMENT FOR FIXED STRUCTURE CONTROLLERS

Ph. Mouyon*, Y. Losser†, C. Cumer‡

* ONERA/DCSD, BP 4025, 31055 Toulouse Cedex, France, fax: +33562252564, mouyon@cert.fr
† SUPAERO, 10 av. E. Belin, 31400 Toulouse, France, losser@cert.fr
‡ ONERA/DCSD, BP 4025, 31055 Toulouse Cedex, France, cumer@cert.fr

**Keywords:** $H_2$ problems. Control law optimization. Adjustment tools. Aircraft control laws.

**Abstract**

The adjustment of fixed structure controllers becomes a major issue in the development of complex control systems. Indeed, for long term project, the controller structure is often fixed at an early stage while the tuning of controller parameters remains possible all along the project. It is also a key point whenever multi-objective design problems are tackled.

The objective of the fixed structure controllers $H_2$ adjustment is to adjust selected gains of a given control law in order to reduce the $H_2$ norm of a transfer function representative of some performance criterium while minimizing all other changes in the closed loop behavior.

This paper deals with methodological aspects of $H_2$ adjustment. We present the development of a Lyapunov approach based on the classical Lyapunov computation of $H_2$ norms. An analytic sensitivity analysis is carried out that leads to an efficient gradient like search. Furthermore we show that fast and accurate numerical procedures may be derived allowing to work with large scale models often encountered in adjustment aeronautical applications.

1 Introduction

1.1 Motivation

Our main concern is in flight control system applications for large passenger transportation aircraft. The aircraft control law design is seldom a direct operation. Generally multiple stages are necessary to obtain a satisfactory result. When the design conditions are modified, to start again the work since the beginning is not necessarily the most effective way. Thus the need for a retuning of the control laws may arise all along the development of an aircraft, each time the design model or the specifications evolve.

Obviously the availability of efficient adjustment tools is of particular importance in the end of the project where delays may have highly negative economic consequences on the overall project. At this stage the structure of the controller is frozen, and the adjustment procedures only intend to adapt some controller parameter so that every control law requirements are satisfied. Thus the development of adjustment tools dedicated to fixed structure controllers becomes a major issue in the development of flight control systems.

Among various objectives, adjustment of $H_2$ norms is often required because a lot of design objectives express as $H_2$ criteria. Adjustment tools allow to compensate for differences between the aircraft design model and the actual aircraft model that includes all the more recent knowledge of the aircraft behavior. This later model often includes a lot of modes neglected at the design step, but may also differ because of new available flight test identification results.

Another need for $H_2$ adjustment tools results from the fact that a lot of $H_2$ constraints are just roughly taken into account for when designing the control law, or even not at all. For example an accurate evaluation of the aircraft behavior in the presence of turbulence wind in terms of servomechanisms fatigue, mechanical loads undergone by the aircraft structure or passengers comfort may leads to the adjustment of the control law gains.

1.2 Methodological aspects

We begin to show in section 2 that the adjustment problem may be seen has a decentralized static output feedback design problem. Fixed order and fixed structure design problems are known to be generally nonconvex. For example existence and uniqueness of stabilizing controllers of a given order or structure is still an open question [1].

Several heuristic procedures have been proposed to solve fixed order design problems. They mostly rely on numerical optimization [2] and thus may be used in the adjustment context. Indeed, even if our objective is not to find an $H_2$ optimal solution, iterative techniques dedicated to the minimization of $H_2$ norms provide a good basis for the development of $H_2$ adjustment procedures.

Proposed approaches are often based on a two-stage optimization process: $V = K$ iterations [2], alternating convex projection methods [3] [4], dual iterations [5]. Each stage is a (quasi) convex optimization problem set up within the linear matrix inequality (LMI) framework. However optimization problems that involve large scale models are not currently tractable by LMI solvers.

We deal here with such large scale problems. The question of stabilization is bypassed assuming that an initial feasible point has been yet found. We just have to preserve stability. Thus our objective is to develop efficient numerical procedures allowing...
to adjust selected gains of a given control law in order to 
- reduce the $H_2$ norm of a transfer function representative of 
some performance criterium or constraint,
- preserve stability,
- and also minimize all other changes in the closed loop behavior.

The satisfaction of this two later points simply results here from 
the fact that we develop iterative small gain corrections 
methods. At each step the adjustment can be stopped if 
unexpected changes appear. A more accurate processing of these 
constraints would be interesting but is out of the scope of the 
present paper.

One approach to controller $H_2$ adjustment for such high 
dimensional systems is through the use of Lyapunov solvers. 
This is the way followed in this paper.

In the first section we develop a sensitivity analysis of $H_2$ criterium 
based on Lyapunov theory. We show that the sensitivity 
with respect to controller gains expresses as a function of 
the matrix solutions of a set of coupled Lyapunov equations. 
This analytic expression of the sensitivity is then used in section 
3 where an $H_2$ adjustment procedure is proposed. 
Such ideas may be seen as a reminiscence of [6] [7] where more 
complex mixed $H_2/H_\infty$ design problems are studied. The main 
advantage of this Lyapunov approach is that numerically efficient 
Lyapunov solvers exist. This lead us to show by numerical 
experiments that, for each adjustment iteration, a linearly 
increasing CPU time with respect to the model order may be 
achieved (section 4).

It is worth to be pointed out that another Lyapunov based approach 
amay be followed. As a matter of fact the $H_2$ optimal static output feedback may be characterized by a set of bilinear 
matrix equations. Alternating projection methods may be then 
applied. This yields to an iterated resolution of Lyapunov equation sets. In this spirit, the Kleinman algorithm [8] dedicated to 
the $H_2$ optimal state feedback case is known to converge. 
The extension to the static output feedback case is under study.

As an alternative strategy for addressing $H_2$ optimal problems 
subject to architecture constraints, homotopic technique have 
been also proposed. Related works may be found in [9] [10] 
Such techniques might be also very interesting within the 
adjustment context.

Finally, note that direct synthesis approaches is out off 
the scope of the present paper, since we follow here a multiple 
steps design approach.

### 2 $H_2$ norm sensitivity analysis

#### 2.1 Modeling dedicated to adjustment

We consider an $M-\Delta$ representation of a closed system model 
as depicted on figure (1). The matrix $\Delta = K$ involves the controller gains. Some of them are to be adjust in order to decrease 
the $H_2$ norm of the closed-loop transfer $G$ from $w$ to $e$. Such a

A structured gain is often called decentralized static output feed-
back and captures a large class of controller architectures [12].

The closed-loop state space representation writes as:

$$\dot{x} = (A + BKC_w)x + Buw$$
$$e = (C_e + D_{eu}KC_y)y$$

(1)

The $A$ matrix includes the open loop system dynamic, the 
controller dynamic and the perturbation model dynamic. Thus $u$ 
and $y$ on figure (1) are not the true system input and output, but 
 fictitious signals that are equivalent to control and measurement 
signals for the $M-\Delta$ representation.

We assume that the $H_2$ norm between $w$ and $e$ is finite whatever 
is the feedback gain $K$. This implies that $D_{uw}$, $D_{wy}$ and $D_{yw}$ are all zero matrices.

#### 2.2 Computation of the $H_2$ sensitivity

We denote by $J$ the squared $H_2$ norm of $G$: $J = \|G\|^2_2$. The classical computation of $J$ within the Lyapunov approach is first 
recalled.

$$J = \text{trace} \left\{ (C_e + D_{eu}KC_y)X(C_e + D_{eu}KC_y)^T \right\}$$

(2)

where $X$ satisfies the Lyapunov equation

$$(A + BKC_y)X + X(A + BKC_y)^T + B_uB_{wu}^T = 0$$

(3)

Within the stochastic framework, if $w$ is a white noise process 
with power spectral density equal to 1, then the definite non 
negative matrix $X$ is the covariance of the state variable $x$.

Since all closed loop matrices are linear functions of the matrix 
gain $K$ it is always possible to shift them towards zero. Thus 
we can assume that the initial value of $K$ is zero.

The sensitivity matrix of $J$ with respect to $K$ is a matrix denoted 
$S = \partial J/\partial K$ with the same dimensions as $K$. It depends on $K$. 
Its value at the initial point $K = K_0 = 0$ is given by the fol-
lowing lemma (2.1). However it must be pointed out that this 
lemma only applies to non zero terms of $S$. Since we consider 
structured feedback, if $K_{i,j}$ is fixed then obviously $S_{i,j} = 0$. 

![Figure 1: $M-\Delta$ representation of a closed system](image_url)
Lemma 2.1 The general term of the sensitivity matrix \( S \) is given by:

\[
S_{ij} = \text{tr} \{ C_i W_{ij} C_j^T + D_i C_j X C_i^T + C_i X (D_i C_j)^T \}
\]

where \( X \geq 0 \) and \( W_{ij} \) satisfy the set of Lyapunov equations:

\[
AX + XA^T + B_o B_o^T = 0 \quad \text{and} \quad AW_{ij} + W_{ij} A^T + B_i C_j X + X (B_j C_i)^T = 0
\]

and where \( B_i \) (resp. \( D_i \)) stands for the \( i \)th column of \( B \) (resp. \( D_c \)) and \( C_j \) stands for the \( j \)th row of \( C \).

Proof 2.2 The dimension of \( K \) is \( m \times p \). We denote by \( e_{i,m} \) the \( i \)th vector of the natural basis of \( \mathbb{R}^m \). We consider a variation of \( K \) with norm \( \rho \) and which is zero everywhere except at row \( i \) and column \( j \). This means that \( K \) writes:

\[
K = \rho e_{i,m} e_{i,j}^T
\]

We have:

\[
B_i = B_0 e_{i,m} \quad C_j = e_{j,m} C_i \quad D_i = D_{eu} e_{i,m}
\]

Then, from equations (2) and (3), \( J \) expresses as a function of \( \rho \):

\[
(A + \rho B_i C_j) X + X (A + \rho B_i C_j)^T + B_0 B_0^T = 0
\]

\[
J(\rho) = \text{tr} \{ (C_i + \rho D_i C_j)^T X (C_i + \rho D_i C_j)^T \}
\]

The derivative of \( J \) with respect to this parameter \( \rho \) is \( K_{i,j} \) is the general term of the sensitivity matrix:

\[
S_{ij} = \frac{dJ}{d\rho} = \text{tr} \{ (C_i + \rho D_i C_j)^T X (C_i + \rho D_i C_j)^T \}
\]

where \( W_{ij} \) is the matrix defined by \( W_{ij} = dX / d\rho \).

In order to evaluate this later matrix we compute the derivative of the Lyapunov equation satisfied par \( X \) (equation 3). Its derivative with respect to \( \rho \) writes as:

\[
B_i C_j X + (A + \rho B_i C_j) W_{ij} + W_{ij} (A + \rho B_i C_j)^T + X (B_j C_i)^T = 0
\]

Finally taking \( \rho = 0 \) yields to the expected lemma result.

### 3 An \( H_2 \) adjustment procedure

#### 3.1 Maximal sensitivity descent

The behavior of \( J \) about the initial point \( K_0 = 0 \) is described at the first order by:

\[
J \approx J_0 + \text{tr} \{ S^T (K - K_0) \}
\]

where \( J_0 \) is the value when \( K = K_0 \). If a gradient algorithm were used to decrease \( J \), with length \( \rho \), then on should take:

\[
K = K_0 - \rho \frac{S}{\text{tr} \{ S^T S \}}
\]

However our objective is to reduce \( J \) but without any other important change. This constraint led us not to use a gradient search algorithm but a descent algorithm along the direction of maximal sensitivity.

Let us consider a Singular Value Decomposition of \( S : S = U \Sigma V^T \). We can write:

\[
K = K_0 + U R V^T
\]

Then we have:

\[
J - J_0 = \text{tr} \{ V \Sigma^T U^T U R V^T \} = \text{tr} \{ V \Sigma^T R V^T \} = \text{tr} \{ R \Sigma^T \} = \sum_i \sigma_i
\]

Let us remark that \( \|K - K_0\|_2 = \|U R V^T\|_2 = \|R\|_2 \). Thus if this norm is fixed and equal to \( \rho \), the maximal decrease of \( J \) is then achieved with

\[
K = K_0 - \rho \frac{u_1 v_1^T}{\sigma_1}
\]

where \( u_1 \) and \( v_1 \) are the left and right singular vectors associated to the largest singular value \( \sigma_1 \) of \( S \).

The first order variation of \( J \) is \( J = J_0 - \rho \). This suggests the \( H_2 \) adjustment procedure described hereafter:

**Procedure**

1. Compute the current closed loop model in order to shift \( K \) towards \( K_0 = 0 \).
2. Compute \( dK^* = u_1 v_1^T / \sigma_1 \), the direction of variation for \( K \) maximizing the sensitivity of \( S \) about \( K_0 \).
3. Search for an optimal \( \rho \) so that the gain variation \( -\rho dK^* \) yields to a minimal \( J \) value, while keeping the closed loop stable.
4. Iterate if required.

Before each step of the algorithm, the previously computed \( K \) matrix is included in the system closed loop model. Then the sensitivity is estimated, and a \( K \)-variation is proposed along the maximal sensitivity direction. A priori a one dimensional search algorithm may be used to solve step (iii). Furthermore, in order to ensure that the iterations stay inside the stability domain, a logarithmic barrier functional may be used in the spirit of interior point methods. However this is a theoretical solution. Simpler and more intuitive processes, as proposed below, may be as efficient and far less time consuming.

#### 3.2 Optimization over an adaptive mesh

The above procedure involves a constraint one dimensional optimization that may be tackled from different angles. We first note that with \( \rho = \alpha_0 \), the new algorithm step \( \alpha \) is just equal to the expected relative decrease of \( J \). Within the adjustment context a typical maximal value for the relative criterion decrease is \( \alpha_{\text{max}} = 10\% \). And a typical minimal value is \( \alpha_{\text{min}} = 1\% \), which corresponds to the minimal criterion decrease that is worth to do. Thus it seems to be reasonable to constraint the one dimensional \( \rho \)-optimization to the range \( J_0 \times [\alpha_{\text{min}}, \alpha_{\text{max}}] \).
In order to speed up the procedure we propose to just test several $\rho$ values over the previously defined range and to jump to the best stable point. We currently use the following mesh for $\alpha$:

$$\alpha \in \alpha_{\text{max}} \times \{0.1, 0.3, 0.7, 1\}$$ (9)

Such a very simple algorithm works quite well when applied to our physical aeronautical problems. When iterated, the adjustment steps produce a sequence of decreasing criterium values. It stops when the relative variation of the criterium is less than $\alpha_{\text{min}}$.

However in some cases it stops prematurely. As a matter of fact, since no continuous penalty function is used to prevent from instability, the current point may converge towards the stability domain barrier. Then it may happens that none of the tested points of the mesh satisfy the stability constraint, and the procedure stops even if the current value of $K$ is far from being locally optimal.

To prevent from such a behavior, the range of the mesh (parameter $\alpha_{\text{max}}$) is adapted. At the beginning of each iteration the value of $\alpha_{\text{max}}$ is initialized at 10%. Then it is reduced (multiplication by 0.9) until there exists at least one stable point in the mesh.

One can think that the procedure may be used to minimize a $H_2$ criterium under fixed structure constraints and stability constraint. However it is worth to be pointed out that even if the proposed procedure always converges, it may not always lead to a local optimum. As a matter of fact, even with an adaptive mesh, the descent direction $K^*$ may not be admissible with respect to the stability constraint. Thus we do not really solve the fixed structure $H_2$ optimization problem under stability constraint.

4 Numerical aspects

4.1 Accuracy

From a numerical point of view, our procedure mainly relies on the resolution of a set of Lyapunov equation, and a singular value decomposition. At each step the number of Lyapunov equations to be solved is equal to the number of free parameters in $K$ plus the number of points of the mesh. Since an important characteristic of the state space models involved in our aeronautical applications is their high dimension, then a lot of care must be paid to the choice of the Lyapunov solver. Indeed even the criterium calculus may be ill-conditioned. And accurate solutions of all the involved Lyapunov equations are not easy to found.

Within such large systems it is recommended to use efficient numerical routines such as the ones of the SLICOT package (http://www.win.tue.nl/niconet/niconet.html). That is what we do. The adjustment procedure was thus first developed under MATLAB, with an embedded SLICOT Lyapunov solver. This led us to verify the good behavior of the proposed adjustment procedure.

As regards the SVD, it must be pointed out that it only applies to a matrix whose dimension is equal to that of $K$. An accurate solution is thus easy to found.

4.2 Computation cost

As regards the computation cost, the first adjustment routine was developed without any special attention dedicated to the computation cost. The average step duration was less than 1.4 s on a SPARC station IV, when a 25 dimensional model was used. In order to increase the size of the models to which the procedure applies a more adapted MATLAB routine has then been developed.

It relies on the fact that at each step, all the Lyapunov equations to be solved involve the same $A$ matrix. Classical Lyapunov solvers use a schur factorization followed by a Gauss pivoting method applied to solve a set of linear triangular equations. Consequently the computational cost of the schur factorization needs not to be repeated. Such a routine yields to an average step duration which is equal to 0.3 s.

With this new algorithm, numerical experiments show that the average CPU time duration of each adjustment iteration increases linearly when the size of the aircraft state space model is greater than about 130 (figures 2). Above this limit each iteration increases of about 0.1s per state. These results were obtained with reduced state space models computed from a unique initial aircraft structural model. For that purpose a reduction procedure dedicated to aircraft structural model reduction has been developed.

5 Application

5.1 An academic example

A first order system has been used to check the behavior of the adjustment algorithm. It is described on figure (3). The
This preliminary study uses a Dryden turbulent wind model. The mechanical load outputs we dealt with here are: 1. the vertical bending moment at the wing root, 2. the vertical bending moment at the tail root, 3. the torsion moment at a forward fuselage point, 4. the torsion moment at a rear fuselage point.

This table (2) brings to the fore the balance between the two bending moments. If the bending moment at the wing root is reduced by 10 percent (first column), the bending moment at the tail root is increased by 20 %. On the other hand (second column) if this later moment is minimized down to 60 %, then the former increases by 34 %. As regards the torsion fuselage moments, the table depicts the fact that the minimization of these moments may result in a very large increase of the bending moments (columns 3 and 4). Thus, in order to achieve a reasonable adjustment, these two criteria must be completed with wing and tail bending terms.

The adjustment algorithm is obviously not aimed to minimize an $H_2$ criterion. It is a tool that may help an engineer to tune a given control law, in a multi-objective and highly constraint framework. This example demonstrates that the proposed procedure is able to deal with high dimensional models as far as adjustment of $H_2$ norms is concerned.

6 Conclusion

In this paper, we propose to use a gradient-like algorithm to solve the iterative $H_2$ adjustment problem. An analytical expression of the sensitivity function is first derived, and an adjustment algorithm based on a maximal sensitivity descent is then described. Numerical experiments show that the procedure is well suited to solve $H_2$ adjustment for large scale systems.

<table>
<thead>
<tr>
<th>Moment</th>
<th>Adjustment case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bending Wing root</td>
<td>-10  34  46  10</td>
</tr>
<tr>
<td>Bending Tail root</td>
<td>20 -60 500 1398</td>
</tr>
<tr>
<td>Torsion Forward fuselage</td>
<td>9  78 -40 48</td>
</tr>
<tr>
<td>Torsion Rear fuselage</td>
<td>-2  66 -4  -64</td>
</tr>
</tbody>
</table>

Table 2: Load power variations (%)
Many other tools may be used to tackle the adjustment problem. We must emphasize on the LMI approach, which provides us with a very interesting framework for the iterative design, as illustrated in the references hereafter. For example a method based on LMI computation of the $H_2$ norm is developed in [13]. However, at the time being, none of these approaches is numerically accurate and efficient enough to be applied within the context of high dimension models.

7 Acknowledgments
Authors want to thank AIRBUS industry and DPAC/SPAe for fruitful cooperation and for supporting this work.

References