Robust Modal Control: A technique for designing restricted complexity controllers

Presented by Carsten DÖLL[2]

Written by Jean-François MAGNI[2]

Results obtained by Caroline CHIAPPA[1,2]

[1] **SUPAERO**

[2] **ONERA – DCSD**

E-mail: “name”@cert.fr

Grenoble January 15 2003

Contents

ONERA

1- Introduction - What is Modal Control ?

The techniques presented in this tutorial are detailed in the book: ***Robust Modal Control with a Toolbox for Use with Matlab***, J.F. MAGNI (RMCT book).

Modal Control is a set of techniques based on eigenvalue / eigenvector assignment.

Basic eigenstructure assignment was introduced in 1975-1976 by KIMURA and MOORE. Design objectives were limited to pole assignment and to decoupling based on eigenvector selection. (All alternative techniques were based on non-convex optimization.)

For about 20 years:

- no significant improvements of theory
- numerous applications in the aeronautical field (Airbus A320, A340... military aircrafts).

These applications demonstrated a natural degree of robustness of modal control.

1- Introduction - Design objectives

Recently, with the extension to dynamic controllers, new kinds of objectives became feasible:

- multimodel control design (therefore the denomination Robust Modal Control became justified).
- frequency domain constraints (DC-gain, band-stop properties, distance w.r.t. given controller)
- structured controller design (constant entries, band-stop entries,...)
- re-design of dynamic controllers

The last item is the key idea for order reduction presented in this tutorial.

In fact, we re-design a simpler version of a given controller by re-assigning closed-loop dominant eigenstructure (plus robustness improved by multimodel assignment, feedback gains structured, frequency domain constraints...)

1- Introduction - Eigenvector selection

During the long period in which eigenstructure assignment was used without improvements, eigenvector selection was based on decoupling ideas $\Rightarrow$ applications were limited to aircraft / rotorcraft control.

When decoupling is not relevant, we suggest to select eigenvectors

- using minimum energy pole placement (not treated here, see RMCT book)
- using projection of eigenvectors
  - projection of open-loop eigenvectors. (preserves natural behaviour of the system, preserve natural open-loop pole dispersion $\rightarrow$ robustness).
  - projection of closed-loop eigenvectors (re-design of a given controller).

1- Introduction - Corresponding feedback gains

Re-design of a given controller with multimodel ideas, structured gain, frequency domain constraints requires a lot of degrees of freedom. So, constant feedback gains cannot be used. In order to introduce additional degrees of freedom it is suggested:

- **To use a dynamic gain:**
  - **Transfer matrix approach (developed in this tutorial)**
  - **Observer-based approach (see RMCT book for details). This approach offers better control of assigned poles but is less flexible for applications.**
- **To design directly a scheduled gain (see RMCT book for details and DÖLL: *Asian Control Conf. 2000* for an application).**

Contents

2- Robust Modal Control - Notations

System: (A, B, C, D).

Bank of systems: (A_i, B_i, C_i, D_i).

Continuum of systems (LFT): $(A(\Delta), B(\Delta), C(\Delta), D(\Delta))$

Eigenvalues: λ_i

Eigenvectors: v_i

Input directions: w_i

2- Robust Modal Control - Assignment with dynamic feedback

Standard eigenstructure assignment. **Let** (λ_i, v_i, w_i) **be such that**

$$\begin{bmatrix} A-\lambda_i I & B \end{bmatrix} \begin{bmatrix} v_i \\ w_i \end{bmatrix} = 0$$

λ_i, v_i **are assigned by the non-dynamic gain** K **if and only if**

$$K\ (Cv_i + Dw_i) = w_i$$

Eigenstructure assignment with dynamic feedback. **Let** (λ_i, v_i, w_i) **be such that**

$$\begin{bmatrix} A-\lambda_i I & B \end{bmatrix} \begin{bmatrix} v_i \\ w_i \end{bmatrix} = 0$$

λ_i, v_i **are assigned by the dynamic gain** $K(s)$ **if and only if**

$$K(\lambda_i)\ (Cv_i + Dw_i) = w_i$$

2- Robust Modal Control - Multimodel assignment

In the previous formula, the same model (A,B,C,D) is considered for each assignment of (λ_i, v_i). Clearly, if we have a bank of models (or an LFT description of the system), for each assignment a different model (A_i, B_i, C_i, D_i) can be considered.

Multimodel eigenstructure assignment with dynamic feedback. Let (λ_i, v_i, w_i) be such that

$$\begin{bmatrix} A_i - \lambda_i I & B_i \end{bmatrix} \begin{bmatrix} v_i \\ w_i \end{bmatrix} = 0$$

λ_i, v_i are assigned by the dynamic gain $K(s)$ if and only if

$$K(\lambda_i)\,(C_i v_i + D_i w_i) = w_i$$

The sequel of this presentation is devoted to illustrate how this result can be used.

2- Robust Modal Control - Feedback gain computation

Dynamic feedback so that:

$$K(\lambda_i)\ (C_i v_i + D_i w_i) = w_i$$

Observer approach: **This approach is not detailed here. It consists of translating the above constraints to linear constraints relative to the matrices defining an observer in Luenberger form. These observers might have less states than those of "minimum order" or have more states than the system has, because they are not used for "signals observation" but for providing additional degrees of freedom. With this approach, all poles are under control (that is not true using the next approach).**

Transfer matrix approach: **The dynamic feedback is defined as follows:**

$$K(s) = \begin{bmatrix} \frac{b_{011}+b_{111}s+\ldots+b_{q11}s^q}{a_{011}+a_{111}s+\ldots+a_{q11}s^q} & \ldots & \frac{b_{01p}+b_{11p}s+\ldots+b_{q1p}s^q}{a_{01p}+a_{11p}s+\ldots+a_{q1p}s^q} \\ \ldots & \ldots & \ldots \\ \frac{b_{0m1}+b_{1m1}s+\ldots+b_{qm1}s^q}{a_{0m1}+b_{1m1}s+\ldots+a_{qm1}s^q} & \ldots & \frac{b_{0mp}+b_{1mp}s+\ldots+b_{qmp}s^q}{a_{0mp}+a_{1mp}s+\ldots+a_{qmp}s^q} \end{bmatrix}$$

ONERA

2- Robust Modal Control - Feedback gain computation

The degrees of freedom are the numerators and denominators. We have decided to fix *a priori* the denominators (non-dominant modes that can be ignored to some extent). Therefore the degrees of freedom are the numerator coefficients:

$$\Xi = [b_{011} \dots b_{qmp}]$$

Assignment constraints: It is obvious that

$$K(\lambda_i)\ (C_i v_i + D_i w_i) = w_i$$

becomes a linear constraint relative to Ξ.

Criterion: Usually, dynamic feedback introduces too many degrees of freedom. Therefore a criterion must be defined, *e.g.*

$$J = \sum_k \|K(j\omega_k)\|_2 \quad \text{or} \quad J = \sum_k \|K(j\omega_k) - K_{\textbf{ref}}(j\omega_k)\|_2$$

These criteria can be shown to be quadratic w.r.t. Ξ.

$\Longrightarrow$ **The computation of $K(s)$ is a Linear Quadratic Programming (LQP) Problem.**

Grenoble January 15 2003

ONERA

2- Robust Modal Control - Feedback gain computation

Comment 1. It is possible, to some extent, to preserve the linearity of assignment constraints when denominators coefficients are also considered as degrees of freedom. But then limitations for gain structuring and controller order reduction arise.

Comment 2. The denominator poles are weakly controllable, so, are not dominant modes. As it will be shown in the illustrative examples we only deal with dominant modes. In the proposed structure of $K(s)$ the denominators can be viewed as filters permitting us to use derivatives of the measurements (pseudo-derivatives).

2- Robust Modal Control - Feedback gain computation

Comment 3. For most systems "(A,B,C,D)" we have $D=0$ and $CB=0$. In this case, using the trace operator, it is straightforward to show that the sum of all closed-loop poles is constant and equal to the sum of the eigenvalues of A and of the denominator roots of $K(s)$ (modulo a minimal realization). This remarks can be used for selecting the denominators of $K(s)$ having in mind the global amount of dominant pole shifting towards the left (this amount being compensated by non-dominant pole motion to the right).

Comment 4. With some design objectives (not considered here): LQP $\rightarrow$ LMI (see LE GORREC: *PhD-thesis No. 257 SUPAERO, 1998*). But computational efficiency is at the moment so much reduced that we prefer to restrict the domain of possible design objectives in order to preserve the LQP nature of the problem.

2- Robust Modal Control - Feedback gain computation

Provisional conclusion.

Up to now, we have shown how to compute a dynamic feedback gain that assigns several (λ_i, v_i) in a multimodel setting.

It remains to:

- **explain how to select the (λ_i, v_i). See Section 3.**
- **introduce structuring and frequency domain constraints. See Section 4.**
- **explain how to select the models in a multimodel design (or re-design) procedure. See Section 5.**

Contents

1- Introduction

2- Robust Modal Control: Generalities

3- Controller order reduction

- *Projection of open-loop/closed-loop eigenvectors*
- *Identification of dominant modes*
- *Proposed algorithm for controller order reduction*
- *Illustrative example (Anderson and Liu problem)*

4- Structured feedback design with an application

5- Multimodel improvement

6- MATLAB tools

7- Conclusion

ONERA

3- Controller order reduction - Projections

It is possible

- **to design directly low complexity controllers**
- **to consider a given controller (*e.g.* μ-synthesis) and to reduce its complexity.**

This presentation emphasizes the second point.

In both cases, the best way for eigenvector selection is to use projection. In the first case we propose to project open-loop eigenvectors, in the second case, we suggest to project the eigenvectors corresponding to dominant closed-loop modes.

Assignable eigenvectors satisfy: $\begin{bmatrix} A_i - \lambda_i I & B_i \end{bmatrix} \begin{bmatrix} v_i \\ w_i \end{bmatrix} = 0$

In other words:

$$\begin{bmatrix} v_i \\ w_i \end{bmatrix} = \begin{bmatrix} V_i \\ W_i \end{bmatrix} \xi_i \quad \textbf{where} \quad \textbf{Im} \begin{bmatrix} V_i \\ W_i \end{bmatrix} \textbf{ = Ker} \begin{bmatrix} A_i - \lambda_i I & B_i \end{bmatrix}$$

3- Controller order reduction - Projections

Let v_{id} denote the desired eigenvector.

This eigenvector must be projected onto the subspace Im(V_i):

$$\xi_i = (V_i^T V_i)^{-1} V_i^T v_{id} \rightarrow v_i = V_i \xi_i \text{ and } w_i = W_i \xi_i$$

Summary :

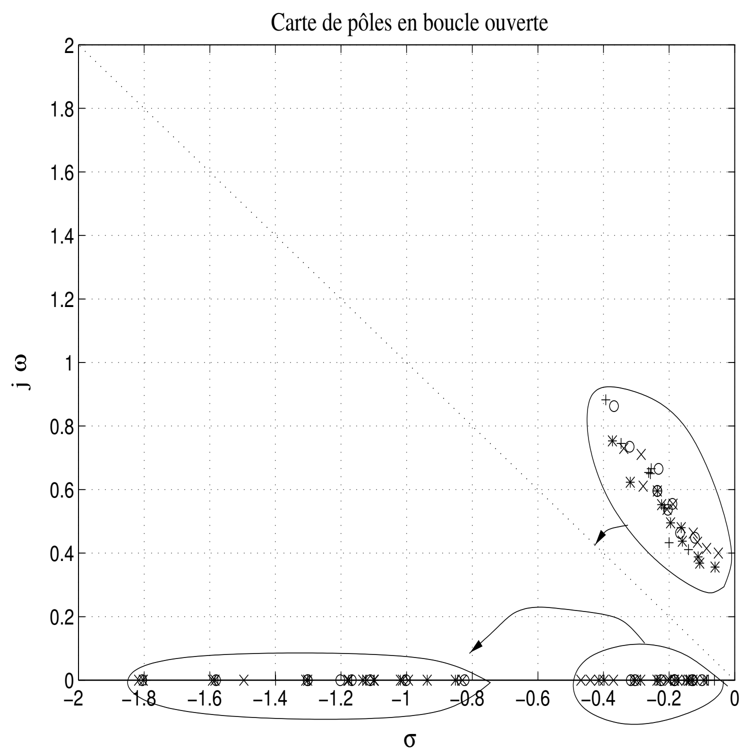
- Select the desired vector v_{id}. (This point will be treated in the next slides)
- Compute the matrix of eigenvectors that can be assigned (V_i).
- Compute ξ_i as above and then $v_i = V_i \xi_i$ and $w_i = W_i \xi_i$
- It remains to solve an LQP problem for obtaining $K(s)$.

3- Controller order reduction - Projections and robustness

The first order variation of an open-loop eigenvalue is given by:

$$\Delta\lambda_{i,ol} = u_{i,ol}\,\Delta A\,v_{i,ol}$$

Systems present normally a certain natural robustness.



Grenoble January 15 2003

3- Controller order reduction - Projections and robustness

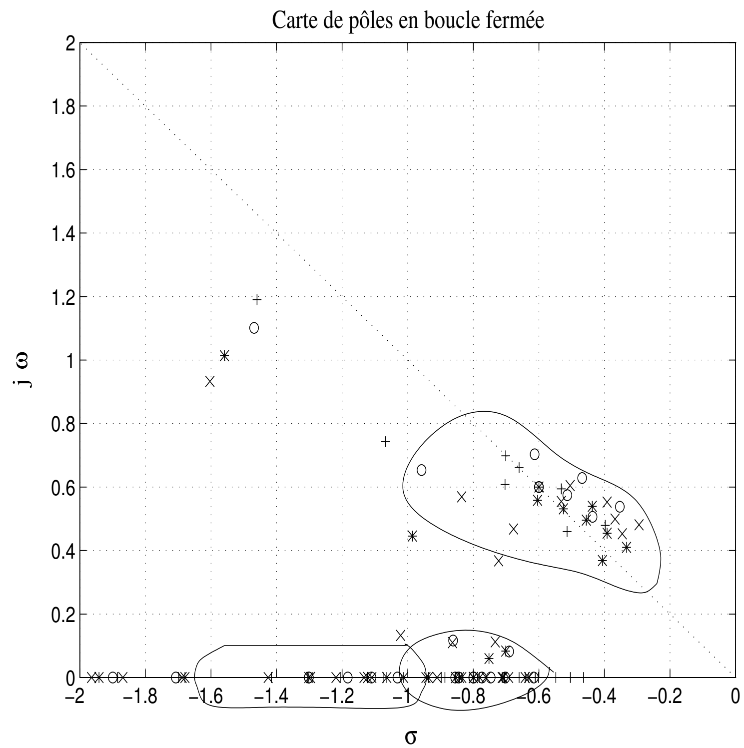
In closed loop, the eigenvalue variation becomes:

$$\Delta\lambda_{i,cl} = u_{i,cl}\,\Delta\hat{A}\,v_{i,cl}$$

For **robustness**, it would be desirable that:

$$\Delta\lambda_{i,ol} = \Delta\lambda_{i,cl}$$

Supposing $\Delta\hat{A} = \Delta A$, it would be necessary to choose $u_{i,cl}$ and $v_{i,cl}$ similar to $u_{i,bo}$ and $v_{i,bo}$. $\Longrightarrow$ **Projection**



ONERA

3- Controller order reduction - Modal decomposition

Assume we have a dynamic feedback. In this case the state-space $\mathcal{X}'$ is the direct sum of the system state-space $\mathcal{X}$ plus controller state-space $\mathcal{X}_c$:

$$\mathcal{X}' = \mathcal{X} \oplus \mathcal{X}_c$$

The eigenvectors become

$$v_i' = \begin{bmatrix} v_i \\ v_{ic} \end{bmatrix} \quad \textbf{where} \quad v_i \in \mathcal{X} \quad \textbf{and} \quad v_{ic} \in \mathcal{X}_c$$

We are going to justify that only v_i in v_i' is of interest.

3- Controller order reduction - Modal decomposition

Consider a system controlled by a dynamic feedback and a controlled output $z = Ex$. $z(t)$ becomes:

$$z = \begin{bmatrix} E & \mathbf{0} \end{bmatrix} \begin{bmatrix} x \\ x_c \end{bmatrix} \quad \text{where} \quad \begin{bmatrix} \dot{x} \\ \dot{x}_c \end{bmatrix} = \begin{bmatrix} * & * \\ * & * \end{bmatrix} \begin{bmatrix} x \\ x_c \end{bmatrix} + \begin{bmatrix} * \\ * \end{bmatrix} u$$

The modal decomposition of $z(t)$ is

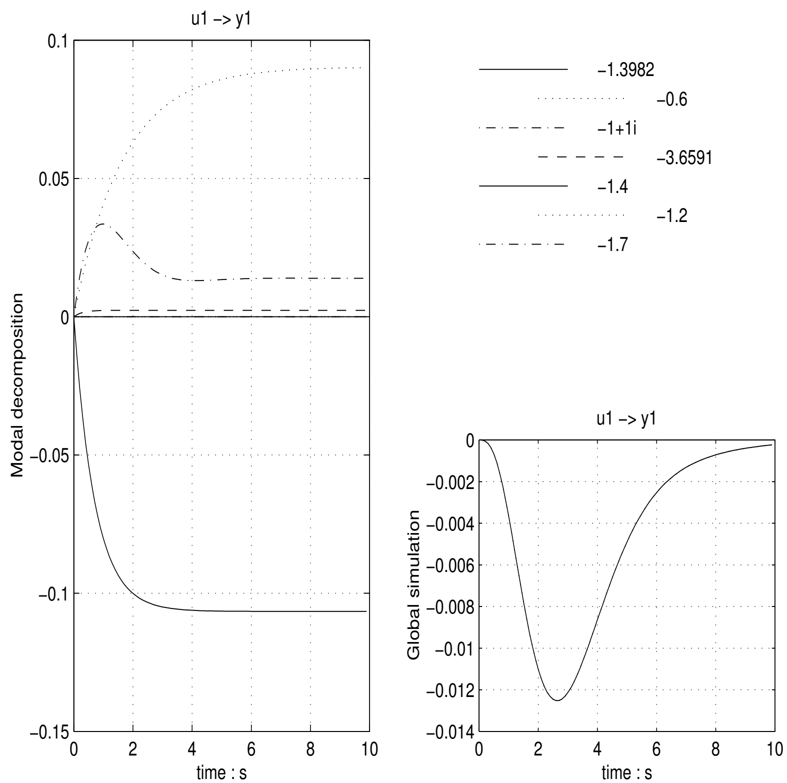
$$z(t) = \begin{bmatrix} E & 0 \end{bmatrix} \sum_i \begin{bmatrix} v_i \\ v_{ci} \end{bmatrix} \int_0^t e^{\lambda_i (t-\tau)} [u_i \;\; u_{ic}] B' u(\tau) d\tau$$

From which it is clear that the signals in $\mathcal{X}_c$ are not “seen” in z.

Conclusion: when a dynamic controller is reproduced, it suffices to re-assign the upper part of the eigenvectors (v_i’s) corresponding to dominant poles.

ONERA

3- Controller order reduction - Dominant poles analysis



First technique for identifying dominant modes

Modal simulation: on the left are plotted separately the components of

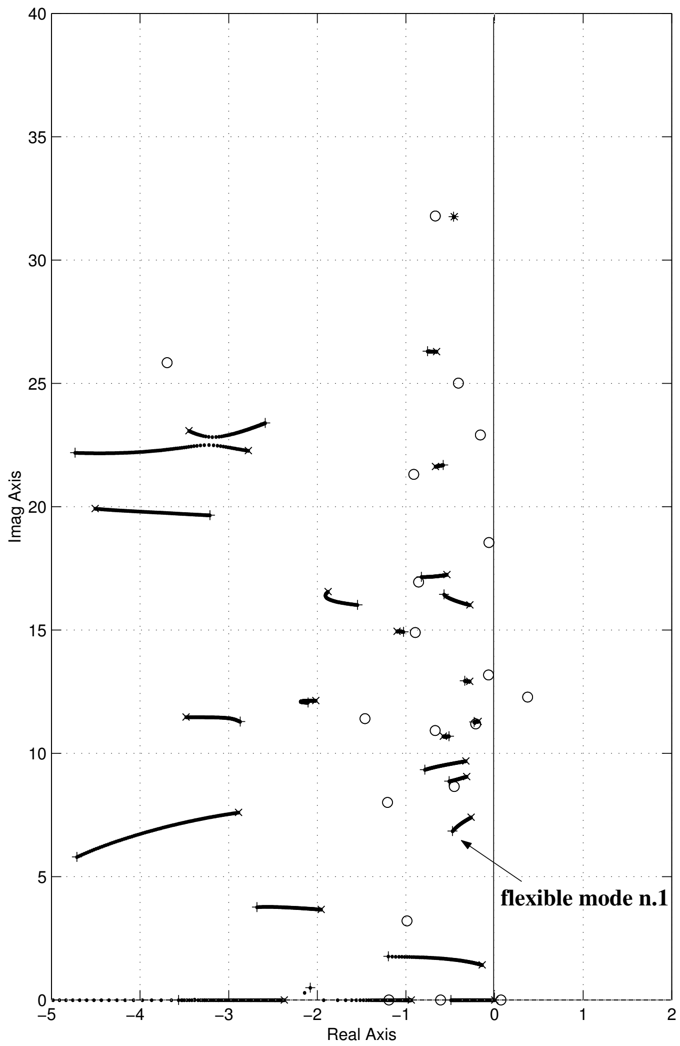
$$\sum_i Ev_i \int_0^t e^{\lambda_i(t-\tau)}[u_i \ \ u_{ic}]B'u(\tau)d\tau$$

Here, there are 2 or 3 dominant modes.

By analyzing several signals, the set of relevant dominant poles can be identified.

ONERA

3- Controller order reduction - Dominant poles analysis



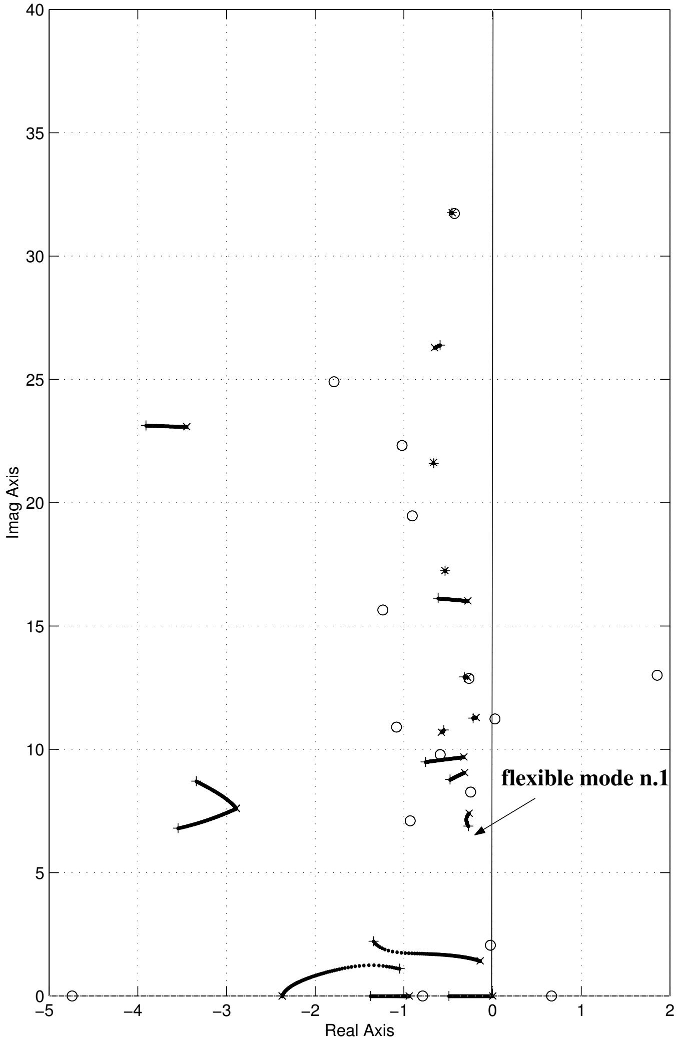
Second technique for identifying dominant modes

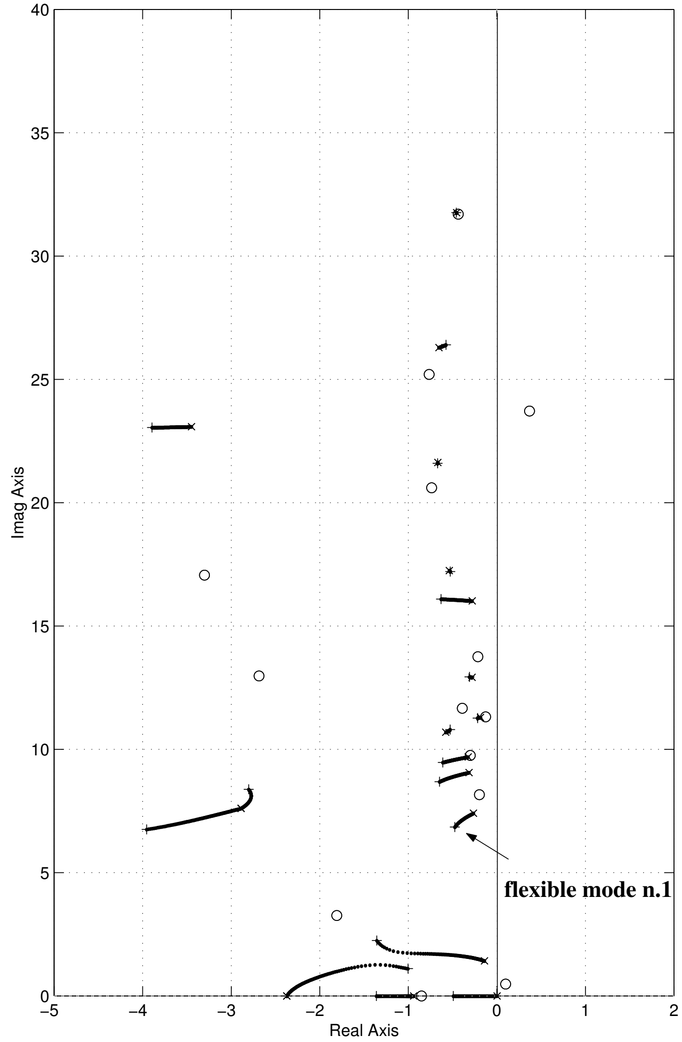
Open-loop dominant poles are usually easy to identify (from physics or modal simulation).

Let $K_0(s)$ be a given dynamic feedback. In some cases considering the root locus for $\rho K_0(s)$ ($\rho \in [0, 1]$) permits the designer to identify dominant poles.

The branches from open-loop dominant poles to closed-loop poles are likely to lead to dominant poles in closed loop.

3- Controller order reduction - Dominant poles analysis





ONERA

3- Controller order reduction - Algorithm

Algorithm for controller order reduction. Let $K_0(s)$ be a given dynamic controller.

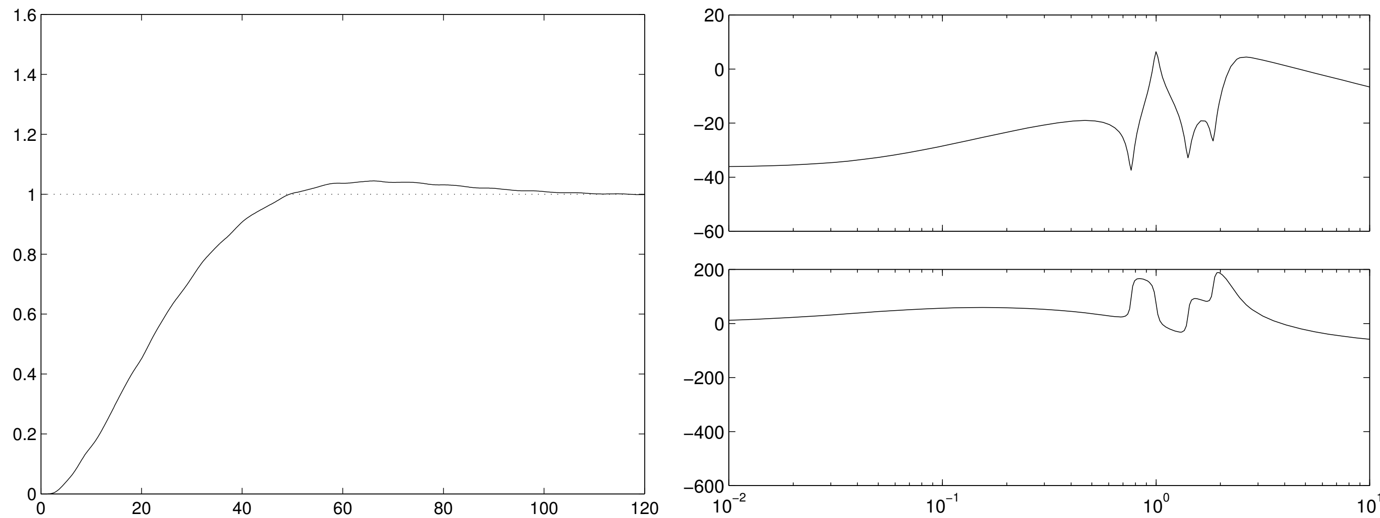
- Identify the **the closed-loop dominant modes**.
- Compute the **upper part** v_{i0} of the eigenvectors corresponding dominant closed-loop poles.
- **Project** these vectors on feasible eigenvector subspaces Im$V_i \rightarrow v_i, w_i$.
- Select the **denominators of** $K(s)$ (form poles of $K_0(s)$). **It is at this step that the complexity of** $K_0(s)$ **can be considerably reduced** because only 1, 2 or 3 poles are sufficient in most cases.
- **Solve the LQP problem** (criterion = distance from $K_0(s)$)

It remains to show how gain structure constraints can be added (Section 4) and how multimodel design can be used to improve robustness (Section 5).

3- Controller order reduction - Illustrative example

"Controllers reduction : Concepts and Approaches" O. ANDERSON, Y. LIU *IEEE Transactions on automatic control 1989*

The plant to be controlled is a four-disk system, 8 states, SISO. The initial controller $K_0(s)$ = LQG controller, 8th order.



Step and Frequency responses of LQG initial controller

ONERA

3- Controller order reduction - Modal analysis of $K_0(s)$

	Poles of	Measure of dominance
$K_0(s)$	$-1.5202+0.6509i$	3.2904
	$-0.9410+1.6906i$	-3.1861
	$-0.0226+0.9993i$	-0.0963
	$-0.3507+2.2429i$	0.0072
closed loop	$-0.0477+0.0473i$	65.9021
	$-0.0157+0.7649i$	-0.3447
	$-0.0278+1.4097i$	-0.1274
	$-0.0374+1.8496i$	0.0484
	$-0.9902+1.6291i$	-0.0019
	$-1.4165+0.5720i$	0.0012
	$-0.3567+2.2640i$	0.0001
	$-0.0229+0.9993i$	-0.0000

3- Controller order reduction - Reduced orders (7th to 5th)

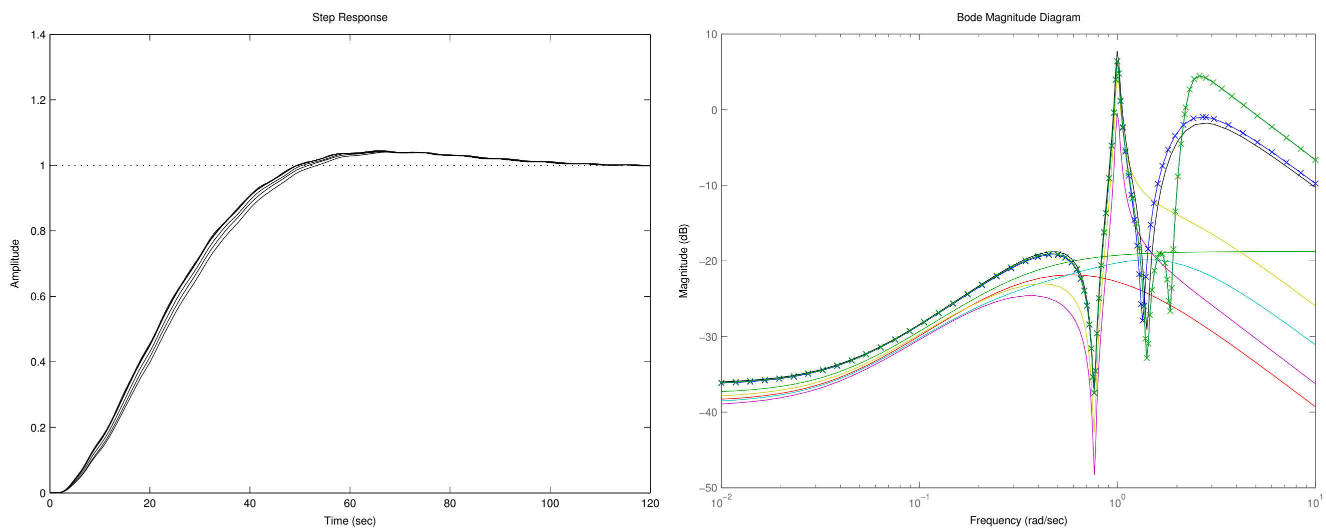
Controller order	Poles of $K(s)$ selected from $K_0(s)$	Re-assigned closed-loop dominant modes
7	$-1.5202+0.6509i$ $-0.9410+1.6906i$ $-0.0226+0.9993i$ -0.3	$-0.0477+0.0473i$ $-0.0157+0.7649i$ $-0.0278+1.4097i$
6	$-1.5202+0.6509i$ $-0.9410+1.6906i$ $-0.0226+0.9993i$	$-0.0477+0.0473i$ $-0.0157+0.7649i$ $-0.0278+1.4097i$
5	$-1.5202+0.6509i$ $-0.0226+0.9993i$ -0.3	$-0.0477+0.0473i$ $-0.0157+0.7649i$

3- Controller order reduction - Reduced orders (4th to 1st)

Controller order	Poles of $K(s)$ selected from $K_0(s)$	Re-assigned closed-loop dominant modes
4	$-0.0226+0.9993i$ -0.3 -1	$-0.0477+0.0473i$ $-0.0157+0.7649i$
3	$-1.5202+0.6509i$ -0.3	$-0.0477+0.0473i$
2	-0.3 -1	$-0.0477+0.0473i$
1	-0.3	$-0.0477+0.0473i$

3- Controller order reduction - Results

- Stable until 1th order reduction.
- Same step response for all order reductions
- Frequency domain responses degraded with order reduction ;-)



Contents

ONERA

Structured feedback design - gain structure

Example: in order to avoid interaction between autopilot and the structural modes of an aircraft, a controller should be designed having the following specific structure (two inputs, four measurements). LE GORREC et al, *"Structured gain design..."*, CESA'98.

- $K_{11}(s) = 0$ — **Constant entry**
- $K_{12}(s) = \alpha_1 \frac{1+0.013s+0.0083s^2}{1+0.13s+0.0083s^2}$ — **Proportional to bandstop filter**
- $K_{13}(s) = \frac{\alpha_2+\alpha_3 s}{1+0.1976s+0.0176s^2}$ — **Proportional / pseudo-derivative**
- $K_{14}(s) = \frac{\alpha_4}{1+0.1s}$ — **Filtered proportional gain**
- $K_{21}(s) = \frac{\alpha_5}{1+0.2s}$ — **Filtered proportional gain**
- $K_{22}(s) = \frac{\alpha_6+\alpha_7 s}{1+0.2252s+0.0253s^2}$ — **Proportional / pseudo-derivative**
- $K_{23}(s) = \frac{\alpha_8+\alpha_9 s}{1+0.2252s+0.0253s^2}$ — **Proportional / pseudo-derivative**
- $K_{24}(s) = \frac{\alpha_{10}}{1+0.15s}$ — **Filtered proportional gain**

Structured feedback design - gain structure

The degrees of freedom are $\alpha_1 \dots \alpha_{10}$.

It is straightforward to transform these structural constraints into linear constraints w.r.t. the vector Ξ (vector of all numerator coefficients of $K(s)$). So the LQP nature is preserved.

Example 1: in $K_{13}(s) = \frac{\alpha_2+\alpha_3 s}{1+0.1976s+0.0176s^2}$ the coefficient of s^2 is equal to zero, which correspond to one entry of Ξ set to zero.

Example 2: in $K_{12}(s) = \alpha_1 \frac{1+0.013s+0.0083s^2}{1+0.13s+0.0083s^2}$ the corresponding coefficients of Ξ satisfy a linear constraint:

$$\begin{bmatrix} \Xi_k & \Xi_{k+1} & \Xi_{k+2} \end{bmatrix} \begin{bmatrix} -0.013 & 0 \\ 1 & -0.0083 \\ 0 & 0.013 \end{bmatrix} = 0$$

Structured feedback design - frequ. domain constraints

The considered constraints are applied at a finite set of frequencies $\{\omega_1, \ldots, \omega_q\}$. It is possible to consider

$$\|K(j\omega_i)\| < \textbf{bound}$$

but the LQP nature is lost ($\rightarrow$ LMI). So we restrict this kind of constraints to coefficients of $K(s)$ and use real and imaginary parts.

Example 1 . Let $K_0(s)$ be a reference gain.

$$-\textbf{bound} < \Re(K_0(i,j)(j\omega_i) - K(i,j)(j\omega_i)) < \textbf{bound}$$

$$-\textbf{bound} < \Im(K_0(i,j)(j\omega_i) - K(i,j)(j\omega_i)) < \textbf{bound}$$

Example 2 . DC gain larger than a given bound

$$K(i,j)(0) > \textbf{bound}$$

These constraints are linear w.r.t. Ξ, so preserve the LQP nature of the computation of $K(s)$.

Contents

Multimodel improvement - principle

Multimodel improvement consists of identifying “worst cases” plus “worst poles” (analysis) and to improve the corresponding control law by adding a constraint to the set of existing ones (multimodel design).

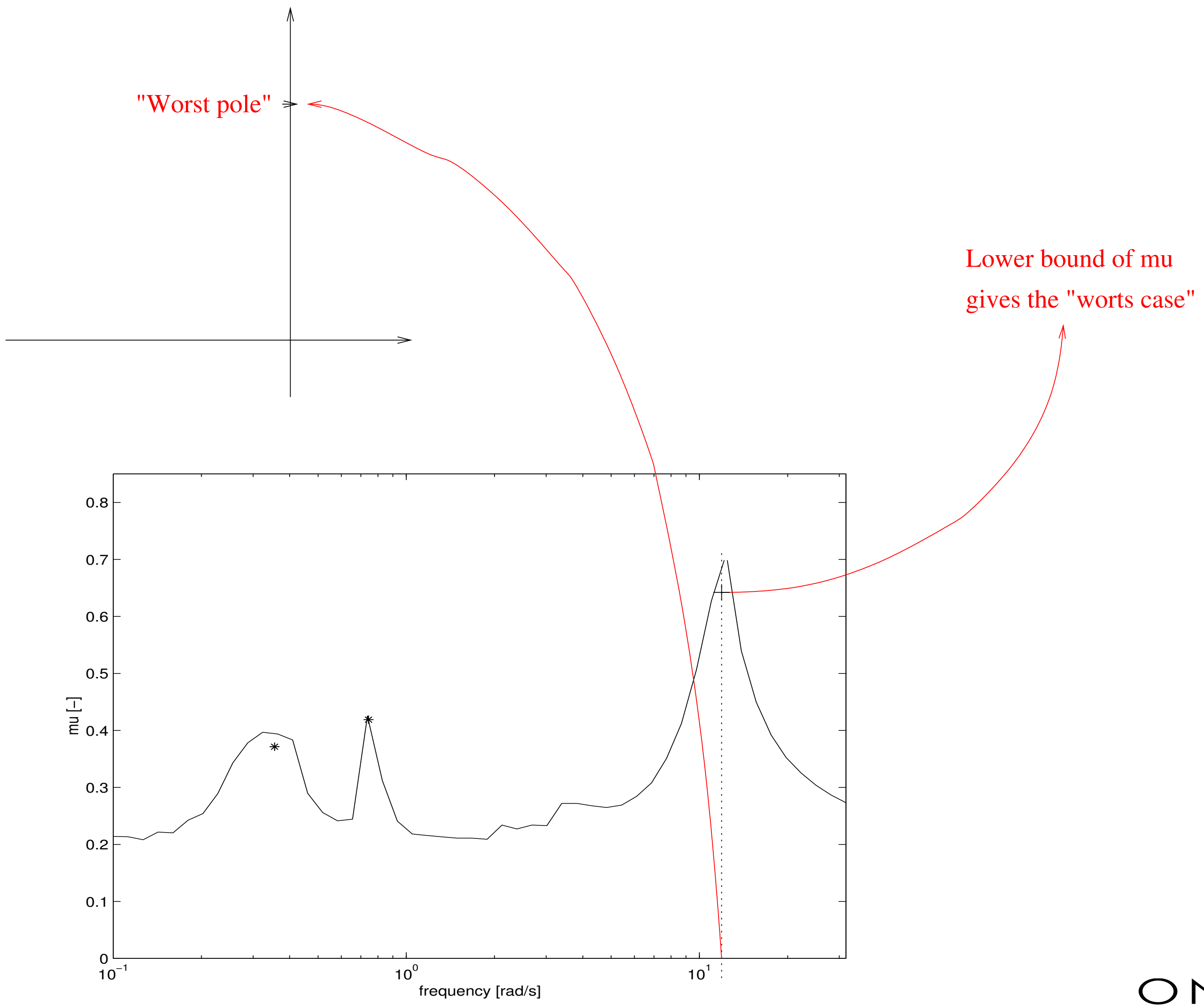
Analysis and multimodel improvement is performed iteratively. At each step a new “worst case” is added $\rightarrow$ multimodel nature.

Analysis can be performed

- considering a bank of linearized models
- using μ-analysis (provided that the uncertain system can be modeled as an LFT)

We shall develop the second point.

Multimodel improvement - use of μ-analysis

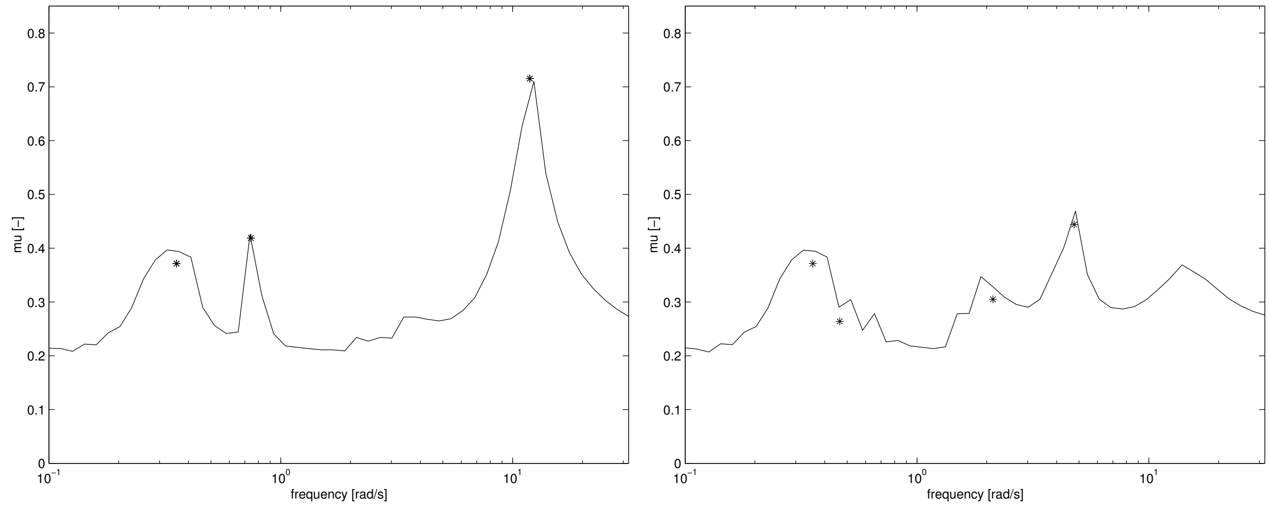


ONERA

Multimodel improvement - illustrative example

CHIAPPA et al, *"Improvement of the robustness of an autopilot"*, CESA'98. Original controller designed using μ-synthesis (BENNANI, GARTEUR'98), order = 18 .

- **First step: reduction to order 4 (similar technique as for ANDERSON and LIU problem).**
- **Second step: improvement by pushing down the μ peak. A multimodel constraint is added to the constraints used for re-design at order 4. (The step responses are similar to original ones).**



Real μ-analysis of original and reduced/improved controller

ONERA

Contents

RMCT Toolbox

The RMCT book is published with a MATLAB toolbox. Five functions permit the user to implement all the techniques presented in this tutorial.

- **str_cstr: gain structure definition**
- **eig_cstr: eigenstructure assignement constraints**
- **add_cstr: frequency domain constraints**
- **ktf_crit: criterion definition**
- **fb_dyn: dynamic feedback computation (LQP)**

Principle:

- **The first three functions plug linear constraints into the matrix CSTR. The 2nd and 3rd function can be invoked as many times as desired (*e.g.* once per model to be treated).**
- **The 4th function defines the criterion and plug it into the matrix CRIT. For composite criteria, this function can be invoked several times.**
- **fb = fb_dyn(CSTR,CRIT)**

RMCT Toolbox: Anderson and Liu problem

The functions for defining constraints have a lot of options that cannot be detailed here. We just give the MATLAB sequency of commands used for solving Anderson and Liu problem. Let us denote fb0 the initial LQG controller.

```
% Feedback structure: 1 × 1, den = poles of fb0
        den = [-1.52+0.65j,-0.3];
        CSTR = str_cstr(1,1,den,1);

% Constraints for preservation of dominant modes:
% pol = dominant poles in closed loop = re-assigned poles
        pol = -0.477+0.0473i;
        CSTR = eig_cstr(CSTR,sys,pol,'p',pol,fb0);

% Quadratic criterion (distance from fb0 at frequencies in omeg)
        omeg = [0 0.1 0.5 1 2 10];
        CRIT = kft_crit(CSTR,fb0,omeg);

% Feedback computation
        fb = fb_dyn(CSTR,CRIT);
```

ONERA

Contents

ONERA