Self-scheduling controller for a launcher in atmospheric ascent

Saussié David∗,∗∗ Gianluigi Baldesi∗,∗∗∗ Carsten Döll****
Caroline Bérard∗,****

∗ SUPAERO, Toulouse, France
** Electrical engineering department, École Polytechnique de Montréal, Canada
*** Dip. di Informatica e Sistemistica, Via Eudossiana18, 00184 Rome, Italy
**** ONERA, DCSD, Toulouse, France*

Abstract: This paper describes the synthesis of a self-scheduled controller for a launcher vehicle. The problem consists in designing a control law which will be valid on the atmospheric ascent trajectory, from time 5 s to time 85 s after takeoff, while ensuring decoupling performances for roll rates between 0°/s and 30°/s. Eigenstructure assignment has been retained in its multi model approach. After introducing the problem itself and the launcher model, the theory related to self-scheduling synthesis is reviewed before its application on the launcher is described. A first performance analysis will be carried out to validate the method and the given results.

1. INTRODUCTION

As a standard practice, the control design for a launch vehicle is based on decoupled dynamic models. Several Thrust Vector Control (TVC) law designs, which assume that an action on the pitch plane produces no effect on the yaw plane, can be easily found in the literature (Alazard et al. [2003]). Coupling effects (Sadray and Colgreb [2005]) in the dynamics can be due to the external forces, but also to the state variables for the velocity linear components and the angular rate components. As Roux and Cruciani [2007] have presented, there can be cases where a negligible roll rate can be foreseen. In order to keep valid the assumption of two uncoupled axes, a dedicated roll control system is added to the initial control design to reduce the roll rate. This additional subsystem has an important effect on the mass performance of the whole launch vehicle since it is added on the first stage.

In this paper the synthesis of a self-scheduled controller for a launcher vehicle having a no negligible roll rate is presented. The coupled dynamics are decoupled for roll rates between 0°/s and 30°/s during the complete atmospheric flight using an eigenstructure assignment technique without the need of an additional roll control system. The (Mu-μ) iteration based on a worst-case analysis and multi-model eigenstructure assignment has given interesting results.

2. ROBUST MODAL CONTROL

2.1 Eigenstructure assignment

We briefly present here the classical eigenstructure assignment method that will be used in the first step to find the initial controller $K_0$ for the nominal plant. Let us consider a linear system $(A_0, B_0, C_0, D_0)$ with $n$ states, $m$ inputs and $p$ outputs.

Proposition 1. Magni [2002] The triplet $T_i = (\lambda_i, v_i, w_i)$ satisfying

$$[A_0 - \lambda_i I \ B_0] \begin{bmatrix} v_i \\ w_i \end{bmatrix} = 0$$  \hfill (1)

is assigned by the static feedback $K_0$ if and only if

$$K_0 \begin{bmatrix} Cv_i + Dw_i \end{bmatrix} = w_i$$  \hfill (2)

The input directions $w_i$ and right eigenvectors $v_i$ associated with the closed loop eigenvalue $\lambda_i$ can be fixed by various methods. They can be chosen with decoupling objectives:

$$[A_0 - \lambda_i I \ B_0] \begin{bmatrix} v_i \\ w_i \end{bmatrix} = 0$$  \hfill (3)

with for example $E = e_j$ and $F = [\cdots \cdots]$ if the $j^{th}$ mode must not excite the $i^{th}$ state. More on the subject can be found in Magni et al. [1998].

2.2 Multi-model modal control

Multi-model eigenstructure assignment Magni et al. [1998] is done by simultaneously assigning triplets $T_i$ for several models which reduces to solve a set of equality constraints of type (2). The choice of the models to treat with and the triplets to assign is determined by an analysis of the stability and/or performance robustness. The proposed iterative procedure is called (Mu-μ)-iteration for Multimodel-worst case analysis.

Procedure: (Mu-μ)-iteration

Step A.1 — Elaborate a first initial design on a nominal model. All kinds of synthesis methods can be applied at this step ($H_\infty$ control, LQG optimal control , μ-synthesis, etc.). In the case of initial non-modal approaches, look for eigenstructure assignment having the same characteristics as the initial controller. Here, we will assign an eigenstructure which stems from a LQ design.
Step B.1 — Here, proceed with a classical multi-model analysis of the pole map and time-responses for worst case performance analysis. Real \(\mu\)-analysis as proposed in Packard and Doyle [1993], Magni and Döll [1997] could be applied at this stage if an LFT was available. That is where the name (Mu-\(\mu\)) stems from. If the initial design is satisfactory for all models or all values of uncertainties, then stop. Otherwise identify the worst-case model, determine its critical triplet \(T_i\) and continue with Step B.2.

Step B.2 — Improve the behaviour of the worst-case model by replacing the triplet \(T_i\) by \(T_i^*\) respecting the specifications while preserving the properties of all models treated before. You will need additional degrees of freedom which can be introduced by using dynamic controllers or self-scheduled controllers, see §2.3. Return to Step B.1.

Remark: See Magni [1999] for some general rules on multi-model eigenstructure assignment, for example to avoid incompatible assignments we should treat models as ‘far’ as possible from each other in the considered parameter space and/or relax some constraints on models treated before.

2.3 Multi-model modal self-scheduling

Classical gain-scheduling is typically done by interpolating \(a\) posteriori the linear controllers obtained for several models. But, because of structure, the gain-scheduling problem can be difficult to tackle. Multi-model modal control handles this task by choosing \(a\) priori the interpolation formula for the controller gain. This choice can be guided by physical constraints or previous experiments. Let us for example take a scheduling w.r.t measurable parameter \(\delta\) and an interpolation formula

\[
K_s(\delta) = K_0 + \delta K_1 + \delta^2 K_2
\]

(4)

The synthesis of this controller can then be tackled using the following Proposition 2 (Magni [2002]).

Proposition 2. The determination of such a self-scheduled controller is equivalent to the synthesis of a multi-model modal controller

\[
K = [K_0 \quad K_1 \quad K_2]
\]

with respect to the augmented system

\[
\begin{bmatrix}
A, B, \begin{pmatrix} C \delta C \\ \delta^2 C \end{pmatrix}, \begin{pmatrix} D \\ \delta D \\ \delta^2 D \end{pmatrix}
\end{bmatrix}
\]

(6)

As it can be seen, the problem boils down to increasing the number of outputs of the original system \((A, B, C, D)\) from \(p\) to \(3p\). The augmentation of the output number offers the additional degrees of freedom necessary for the simultaneous resolution of some linear constraints of type (2) at Step B.2 of the (Mu-\(\mu\))-iterations.

For aeronautical applications, refer Döll et al. [2001] and Constant et al. [2002].

3. APPLICATION TO THE LAUNCHER

Let us apply this method to the launcher atmospheric ascent problem.

3.1 Launcher model

Launch vehicle dynamics are generally described by ‘short-period’ equations of motion during the atmospheric flight. Indeed, in this particular flight phase, the main constraint for a launcher is to minimize the angle of attack, which generates a lift force acting on the lateral direction of vehicle. Therefore no important manoeuvres are commanded. Although the ‘short-period’ equations are the results of a linearization process, the lateral dynamics on pitch and yaw axes can be coupled by the vehicle roll rate. In fact, when the launch vehicle has a not negligible roll rate, an action on the pitch plane produces an effect also in the yaw plane. Neglecting the elastic behaviour of the whole system and the actuation dynamics, the model is just characterized by the following equations (Greensite [1970]):

Lateral forces equations:

\[
\begin{align*}
\dot{v} &= -r u_0 + p w - \frac{1}{m} \rho v^2 \frac{S_R}{2} \int \frac{\partial C_N(q)}{\partial q} \beta(q) \text{d}q + g \cos \theta _0 \cdot \Delta \theta - \frac{T_c}{m} \Delta \sigma \\
\dot{w} &= -p w + u_0 w - \frac{1}{m} \rho v^2 \frac{S_R}{2} \int \frac{\partial C_N(q)}{\partial q} a(q) \text{d}q - g \sin \theta _0 \cdot \Delta \theta - \frac{T_c}{m} \Delta \varepsilon
\end{align*}
\]

Moment equations:

\[
I_{y\gamma} \ddot{\gamma} = -(I_{xx} - I_{zz}) p \omega - I_{zz} \Delta \varepsilon = \frac{1}{2} \rho v^2 \frac{S_R}{2} \int \frac{\partial C_N(q)}{\partial q} (\eta_{y\gamma} - \eta) q(q) \text{d}q
\]

\[
I_{zz} \ddot{\zeta} = -I_{yy} - I_{xx} p \omega - I_{zz} \Delta \varepsilon = \frac{1}{2} \rho v^2 \frac{S_R}{2} \int \frac{\partial C_N(q)}{\partial q} (\eta_{zz} - \eta) \beta(q) \text{d}q
\]

The kinematic equations are \(\Delta \dot{\phi} = q\) and \(\Delta \dot{\psi} = \frac{1}{\cos \epsilon_0} \cdot r\) where \(l_c\) is the distance between the centre of mass and the pivot point; the \(T_c\) is the swivelled thrust; \((\rho v^2 S_R)/2\) is the dynamic pressure; \(S_R\) is the reference area. The angle of attack \(\alpha\) and the side-slip angle \(\beta\) can be expressed as follows:

\[
\begin{align*}
\alpha(q) &= \frac{w + w_{wind}}{u_0} - \frac{\eta y - \eta}{U_0} q \\
\beta(q) &= \frac{v + v_{wind}}{U_0} + \frac{\eta y - \eta}{U_0} r
\end{align*}
\]

The states are the lateral launcher velocities \((v, w)\), angular velocities \((q, r)\), yaw and pitch angles \((\psi, \theta)\) with \(X = [v \ r \ \psi \ w \ q \ \theta]^T\) and the inputs are the two nozzles deflections \(U = [\Delta \sigma, \Delta \varepsilon]^T\).

Using the above equations of motion, different linear time-invariant (LTI) models can be derived for different instants of a flight trajectory by assuming the frozen parameters including the vehicle roll rate \(p\) in order to investigate the effect of the roll rate on the system dynamics. Therefore, 36 different models were generated for 9 different flight instants (from 5s called Model 1 to 85s called Model 9 with an increment of 10 seconds), each of them for 4 roll rates \(p_0 = (0, 10, 20, 30)^{\circ}/s\). Figure 1 shows how the open-loop poles evolve with the roll rate \(p_0\) on model 6. It illustrates that the open loop launcher is naturally unstable. For \(p_0 = 0^{\circ}/s (\Delta)\), the pair of the real positive pole at \(+1.7\) and the real negative pole at \(-1.85\) determine the unstable pitch motion. You can also identify a pole at the origin representing the integration of the pitch rate \(q\) to the pitch angle \(\theta\) and one slightly unstable pole at \(+0.01\) associated to the lateral drift \(w\). You find more or less the same poles (\(\Delta\)) for the unstable yaw motion, the yaw angle \(\psi\) and the lateral drift \(v\). Both axes are naturally decoupled. With non-zero \(p_0\) however, \(i.e. +\), the real poles for pitch and yaw motion combine eachother to coupled complex conjugated pole pairs.
3.2 Specifications

The controlled outputs are the pitch angle \( \theta \) and the yaw angle \( \psi \). A performance objective for the control law is to follow a unit step in \( \theta \) (resp. in \( \psi \)) with a 90\% settling time \( T_s \) inferior to 1.8\,s, an overshoot less than 10\% and no steady-state errors.

The decoupling objective is as follows: a command on \( \theta \) or \( \psi \) has to limit the coupling effects.

At last, the controller must be robust to any roll rate variations between 0°/s and 30°/s and be effective all along the takeoff phase between time 5\,s and time 85\,s. In order to be robust to non modelised high-frequency dynamics or flexible effects, the gain margins should remain between 0 dB and 3 dB, and the delay margins greater than 40ms.

3.3 Controller architecture

The available measurements are \( v, r, \psi, w, q, \theta \). We add two integral terms on \( \psi \) and \( \theta \) in order to ensure zero steady-state errors on these variables and perturbation robustness. With 8 measurements, we are able to assign 8 poles, see Eq. (2). With two inputs \( \Delta \sigma \) and \( \Delta \xi \), there are 16 gains \( K_{ij} \) as degrees of freedom:

\[
K = \begin{bmatrix}
K_{11} & K_{12} & K_{13} & K_{14} & K_{15} & K_{16} & K_{17} & K_{18} \\
K_{21} & K_{22} & K_{23} & K_{24} & K_{25} & K_{26} & K_{27} & K_{28}
\end{bmatrix}
\]  

allowing furthermore the axe decoupling following Eq. (3).

3.4 Eigenstructure assignment for the nominal model \( M_0 = \) Model 6 with \( p_0 = 0°/s \)

\( \theta \) and \( \psi \) are naturally decoupled. \( \lambda_1 \), \( \lambda_{2,3} \) and \( \lambda_4 \) are belonging to \( \theta \) and \( \lambda_5 \) and \( \lambda_6 \) to \( \psi \). During the design Step A.1, the low frequency poles \( \lambda_1 \) and \( \lambda_5 \) are chosen near low-frequency zeros in order to compromise non desirable zero effects. \( \lambda_{2,3} \) and \( \lambda_{6,7} \) satisfy the settling time criterion of 1.8\,s. Their damping is 0.7 for an overshoot less than 10\%. The fast eigenvalues \( \lambda_4 \) and \( \lambda_8 \) are chosen faster than the complex poles, but not too fast in order to keep gains \( K_{ij} \) small. See \( M_0 \) in Tab. 1 for more details. The eigenstructure assignment boils down to a pole placement following Eq. (2). This single model controller leads for sure to good time behaviour in terms of settling time and overshoot for the nominal model, but time response analysis during Step B.1 identifies immediately that it does not ensure decoupling on the other models (it was not taken into account during synthesis) and that oscillations appear for higher roll rates \( p_0 \geq 20°/s \). Model 6 with \( p_0 = 20°/s \) is identified as the first worst-case model \( M_1 \).

3.5 Search for a suitable eigenstructure for Model \( M_1 = \) Model 6 with \( p_0 = 20°/s \)

The choice of closed loop eigenvalues and eigenvectors is not as straightforward as the coupling depends on the roll rate \( p \). In order to find a suitable eigenstructure on \( M_1 \), it is first decided to treat \( p \) independently from \( M_0 \), i.e. return to Step A.1 instead of continuing with Step B.2. A similar pole placement as for model \( M_0 \) with coherent supplementary constraints (3) to decouple \( \theta \) and \( \psi \) does not give expected results. A trial and error approach did not lead to the selection of adequate eigenvalues and eigenvectors.

In order to solve the issue, a LQR design (Kalman [1960]) is then considered with \( R = diag(0.001, 0.001) \) and \( Q = diag(0, 0.1, 0.1, 1.0, 0.1, 0.1, 1.0, 0.1) \) where \( f \) and \( \psi \) are the more ponderated states. The LQR controller performed quite well placing the closed loop poles as follows:

\[
\begin{align*}
\lambda_{1,2} &= -0.0459 \pm 0.3490i \\
\lambda_{3,4} &= -1.3527 \pm 1.1534i \\
\lambda_{5,6} &= -1.5915 \pm 1.2870i \\
\lambda_{7,8} &= -64.7749 \pm 0.34i
\end{align*}
\]

The complex pair of eigenvalues \( \lambda_{1,2} \) seems to be critical in the decoupling effect and is located near a multivariable zero. As multivariable zeros are not as well understood as SISO zeros, our former study did not point that out. This complex value must be preserved and calculated by an LQR design for each considered model. It can not be decoupled as for \( M_0 \) and is hence affected either to \( \theta \) or to \( \psi \), here to \( \theta \). Finally, an eigenstructure affecting \( \lambda_{1,2}, \lambda_{3,4}, \lambda_5 \) to \( \theta \) and \( \lambda_{6,7} \) to \( \psi \) is assigned. The values for \( \lambda_i \neq \lambda_{1,2} \) are the same as those for \( M_0 \) in order to satisfy settling time and overshoot criteria while keeping gains \( K_{ij} \) as small as possible. See \( M_1 \) in Tab. 1. Figure 2 shows that all requirements concerning settling time, overshoot and decoupling are satisfied for \( M_1 \).

Table 1. Eigenstructure assignments for the three synthesis models

<table>
<thead>
<tr>
<th>Model</th>
<th>Poles</th>
<th>Closed loop</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M_0 )</td>
<td>#6</td>
<td>( p_0 = 0°/s )</td>
</tr>
<tr>
<td>( M_1 )</td>
<td>#6</td>
<td>( p_0 = 20°/s )</td>
</tr>
<tr>
<td>( M_2 )</td>
<td>#6</td>
<td>( p_0 = 20°/s )</td>
</tr>
</tbody>
</table>
Having identified the suitable eigenstructure assignments for models with zero and non-zero roll rate $p_0$, it is possible to continue the (Mu-$\mu$)-iterations stopped at the end of §3.4, i.e. with Step B.2. $M_0$ and $M_1$ can be treated simultaneously, if $p$ is introduced as a scheduling parameter. The total roll rate $p$ is normalized as $p = 15 + 15\delta p$ and $\delta p \in [-1, 1]$. The controller is then scheduled as follows:

$$K = K_0 + K_1 \delta p$$

The self-scheduled controller is synthetized by assigning the eigenvalues of the augmented system:

$$\begin{bmatrix} A_{\Delta}, B_{\Delta}, [C_{\Delta}, \delta p C_{\Delta}], [D_{\Delta}, \delta p D_{\Delta}] \end{bmatrix}$$

with $K = [K_0 \ K_1]$. $K$ has now $2 \times 16 = 32$ degrees of freedom which can be used to assign two models following Eq. (2). The two former eigenstructure for $M_0$ and $M_1$ of Tab.1 are assigned simultaneously on the augmented system. When dealing with $M_0$, the augmented system (10) is taken with $\delta p = -1$ whereas $\delta p = 1/3$ for $M_1$.

A quick study of the time responses (Step B.1) reveals that this scheduled controller performs very well for models 1 to 6 at any roll rates in $[0^\circ/s, 30^\circ/s]$ (Fig.3). Nevertheless, the behaviour tends to be quite oscillatory and not satisfactory for models 7 to 9 with rates above $10^\circ/s$ (Fig. 4). Depending on the time, the performances of this scheduled controller are hence not acceptable. Model 9 at $p_0 = 20^\circ/s$ is identified as a second worst case model $M_2$.

### 3.7 Self scheduled controller w.r.t. $p$ and $t$

In order to be able to treat simultaneously models $M_0$, $M_1$ and $M_2$, 16 new degrees of freedom have to be added to Eq. (2) in Step B.2. The idea is now to add the time as a supplementary scheduling parameter. The so-scheduled controller has the form:

$$K = K_0 + K_1 \delta p + K_2 \delta t$$

The total time is normalized as $t = 55 + 30\delta t$ and $\delta t \in [0, 1]$ (for model 6, $\delta t = 0$ and for model 9, $\delta t = 1$). The self-scheduled controller is synthetized by assigning the eigenvalues of the augmented system:

$$\begin{bmatrix} A_{\Delta}, B_{\Delta}, [C_{\Delta}, \delta p C_{\Delta}], [D_{\Delta}, \delta p D_{\Delta}] \end{bmatrix}$$

In addition to the former eigenstructure assignments for $M_0$ and $M_1$, the assignment for Model 9 with $p_0 = 20^\circ/s$ has to be chosen. The low-frequency poles $\lambda_{1,2}$ are again determined by an LQR approach as for $M_1$ for the same reasons. The other eigenvalues are chosen as for $M_0$ in order to respect settling time and overshoot criteria while keeping gains $K_{ij}$ small. Again $\lambda_{1,2}$, $\lambda_{3,4}$ and $\lambda_5$ are affected to $\theta$; the remaining 3 to $\psi$. See $M_2$ in Tab. 1. Figure 5 shows the result of the time-responses analysis in Step B.1. They are now quite satisfactory for all models and roll rates. It is possible to stop the (Mu-$\mu$)-iterations at this point.

Nevertheless, there are still light oscillations and the decoupling could still be improved. One way to improve this behaviour could be to continue the (Mu-$\mu$)-iterations by also treating Model 1 at $t = 5$ s as a third worst-case model $M_3$ during synthesis. This would mean to add an additional term $K_3$ to the controller of Eq. (11) with...
an additional parameter $\delta t$ (for example $\delta t^2$) in order to create the necessary degrees of freedom for the control design. Here, a simpler way was chosen.

3.8 Improved self scheduled controller w.r.t. p and t

By speeding up the fast poles of Models 6 and 9, $p_0 = 20^\circ/s$ (instead of $\lambda_{5,8} = -6$, we take $\lambda_{5,8} = -20$ on Model 6 and $\lambda_{5,8} = -30$ on Model 9), one can obtain much better time-responses (Fig 6). But an overshoot can be observed on $\theta$ time-responses of Models 1 and 2. As these models were not considered in the eigenstructure assignment, they are not entirely satisfactory but still remain acceptable. Nevertheless, gains are higher compared to the former scheduled controller, see Tab. 2 for the values of $K_0$, $K_1$ and $K_2$. Moreover in presence of a flexible model, such big gains may not give as good results. This will be treated in future work on a flexible model of the launcher. Frequency constraints can be added in the synthesis in order to limit the action of the controller on specific frequencies and/or additional terms can be added to Eq. (11) in order to increase the degrees of freedom.

Figure 5. Time responses due to a unit step command in $\theta_c$ (up) and in $\psi_c$ (down) with the final scheduled controller

Figure 6. Time responses due to a unit step command in $\theta_c$ (up) and in $\psi_c$ (down) with the final improved scheduled controller

Figure 7. Pitch control loop SISO margins

Figure 8. Yaw control loop SISO margins

4. PRELIMINARY ROBUST ANALYSIS

We propose a brief study of the robustness of our improved controller. First SISO margins are calculated, then MIMO margins.

4.1 SISO margins

Gain, phase and delay margins are found with the improved controller. For the pitch control loop, Figure 7 shows that the gain margins are good compared to the required 3dB margin. The delay margins are rather poor at $p_0 = 30^\circ/s$, but the specification of 40 ms is satisfied for all models except of Model 1 whose delay margin violates with 25 ms the requirement. Considering the yaw control loop (Fig. 8), all margins are good for any case.

With increasing roll rate $p_0$, gain and phase margins are improving which seems surprising, but for greater roll rates ($50^\circ/s$ and above), they are decreasing. Nevertheless, SISO margins are quite optimistic and MIMO margins must be calculated.

4.2 MIMO margins

In the SISO approach, there is only one control loop that is disturbed at the same time. Hence the margins found before may not reflect the robustness of the system. The
Table 2. Improved scheduled controller matrices

\[
K_0 = \begin{bmatrix} 0.0008448 & 2.9753 & 5.8158 & -7.5389 & 0.000003257 & -0.03935 & -0.002092 \\
0 & 0.03938 & 0 & 0 & 0.0008568 & 2.9754 & 7.7014 \end{bmatrix}
\]

more time for computation. In our case, only 3 models needed to be treated at the same time. A LFT model is under study in order to perform a \( \mu \)-analysis that will confirm our results.

ACKNOWLEDGEMENTS

We thank Piero D. Resta (VEGA LV Avionics Chief Eng., from ESA/IPT) for useful discussions and advices, and Sabrina Labbad for obtaining the main results.

REFERENCES


