Abstract: The missile pitch-axis autopilot design is revisited using a new and recently available LPV control technique. The missile plant model is characterized by an LFT representation. The synthesis task is conducted by exploiting new capabilities of the LPV method: a set of $H_2$/$H_{\infty}$ criteria is considered and different Lyapunov and scaling variables are used for each channel/specification. The method is shown to provide additional flexibility to tradeoff conflicting and demanding performance and robustness specifications for the missile while preserving the practical advantage of previous single-objective LPV methods.

Keywords: missile autopilots, LPV synthesis, LFT, mixed $H_2$/$H_{\infty}$, multi-channel control, gain scheduling.
For the LPV plant (1) the gain-scheduling control channel. denitions, the pair \( (\Delta_1, \Delta_2, \ldots, \Delta_L) \) \( \geq 0 \),

\[
\Delta := \sum_{i=1}^{L} \alpha_i \Delta_i, \quad \sum_{i=1}^{L} \alpha_i = 1,
\]

where \( \alpha_i \geq 0 \) are the polytopic coordinates of \( \Delta \). Polytopic coordinates are computed in real time as functions of the scheduling variables (Section 4) and can be exploited by the controller. According to our denitions, the pair \( (w_{\Delta}, z_{\Delta}) \) is the gain-scheduling channel.

For the LPV plant (1) the gain-scheduling control problem consists in seeking an LPV controller \( K \) with LFT structure.

\[
\begin{bmatrix}
A_K & B_{K1} & B_{K2} \\
C_{K1} & D_{K11} & D_{K12} \\
C_{K2} & D_{K21} & D_{K22}
\end{bmatrix}
\begin{bmatrix}
x_k(k+1) \\
u(k) \\
z_k(k) \\
w_k(k)
\end{bmatrix} =
\begin{bmatrix}
A_X & B_{X1} & B_{X2} \\
C_{X1} & D_{X11} & D_{X12} \\
C_{X2} & D_{X21} & D_{X22}
\end{bmatrix}
\begin{bmatrix}
x_k(k) \\
u(k) \\
z_k(k) \\
w_k(k)
\end{bmatrix}
\]

The assumptions (5) and (6) mean that:

- the time-varying parameter \( \theta \) is valued in a hyper-rectangle \( \mathcal{P}_\Theta \) of \( \mathbb{R}^q \), with
  \[
  \mathcal{P}_\Theta := \{\Theta_1, \ldots, \Theta_L\},
  \]
  where the \( \Theta_i \) are the vertices of \( \mathcal{P}_\Theta \);
- \( \Delta \) and \( \theta \) have the same polytopic coordinates \( \{\alpha_i\} \);
- \( L = 2^r \) and \( N = \sum_{i=1}^{r} s_i \).

Hereafter, \( i (= 1, \ldots, L) \) indexes the vertices \( \Theta_i \) and \( \Delta_i \), \( j (= 1, 2, \ldots) \) indexes the channels and specications, and \( l (= 1, \ldots, r) \) indexes the parameters.

It is shown in Reference (Apkarian et al., 2000) that sufficient conditions for the existence of a solution to the multi-objective LPV control problem can be written as an LMI program. The general synthesis scheme is described below.

**Algorithm 2.1. Controller synthesis**

**Step 1:** Define the following general non symmetric decision variables which are common to all specications and channels (Table 1):

- the set \( S_i \) of general slack variables; the set \( K_v \) of transformed controller variables, whose dimensions must be de ned in accordance to the controller dimensions; and the set \( \Delta_{Kv} \) of scheduling function coefcients.

**Step 2:** For each \( H_2 \)-channel, de ne the set \( H_{2v} \) of the following symmetric decision variables:

- Lyapunov variables \( (X_{2j}) \): scaling variables \( (Q_{2j}) \) and \( R_{2j} \); and a performance variable \( (v_j) \).

**Step 3:** For each \( H_{\infty} \)-channel, de ne the set \( H_{\infty v} \) of the following symmetric decision variables:

- a Lyapunov variable \( (X_{\infty j}) \): scaling variables \( (Q_{\infty j}) \) and \( R_{\infty j} \); and a performance variable \( (v_j) \).

**Step 4:** For each channel/specication, construct the LMI constraint system derived in Appendix A of (Apkarian et al., 2000) and represented here by the simple notations below:

- \( H_2 \) performance:
  \[
  \mathcal{L}_{H_2}(S, K_v, \Delta_{Kv}, H_{2v}, \Delta_i P_j) < 0
  \]

- \( H_\infty \) performance:
  \[
  \mathcal{L}_{H_\infty}(S, K_v, \Delta_{Kv}, H_{\infty v}, \Delta_i P_j) < 0
  \]

where \( P_j \) is the set of state-space matrices representing the LPV plant (1) with only the channel/specication \( (w_j, z_j) \) under consideration.

**Step 5:** (LMI optimization problem) - Minimize a specic performance variable \( v_j \) or \( v_i \) subject to the LMI constraints (8)-(9), fixing the remaining performance variables at some adequate set of values; or simply compute a feasible solution to the LMI constraints (8)-(9).

**Step 6:** As described in (Apkarian et al., 2000), compute the LPV controller data (4) as functions of
the decision variables (Table 1) obtained in Step 5. Note that the set $K$ (bold notation) does not represent the set of controller data. The controller gain-scheduling function is determined by
\[
\Delta_K(\Delta) := \sum_{i=1}^{L} \alpha_i \Phi_i,
\]
where the $\Phi_i$ can be computed off line as functions of the decision variables.

3. DISCRETIZATION

While genuine extensions of the foregoing method to the continuous-time case remain challenging, it can be indirectly applied to continuous plants with the help of a formal bilinear transformation.

Consider a continuous-time system
\[
\begin{align*}
\dot{x}(t) &= \dot{A}x(t) + \dot{B}\xi(t) \\
\psi(t) &= \dot{C}x(t) + \dot{D}\xi(t)
\end{align*}
\]  
(11)

A corresponding discrete-time state-space realization is obtained as
\[
\begin{bmatrix}
A & B \\
C & D
\end{bmatrix} = \mathcal{F}_u \left( \begin{bmatrix}
\dot{A} & \dot{B} \\
\dot{C} & \dot{D}
\end{bmatrix}, \begin{bmatrix} I & \sqrt{2I} \\
\sqrt{2I} & I
\end{bmatrix} \right)
\]
(12)

where $\mathcal{F}_u$ is the customary notation for upper LFT. A transformation from the discrete-time domain to the continuous-time domain can be obtained similarly:
\[
\begin{bmatrix}
\dot{A} & \dot{B} \\
\dot{C} & \dot{D}
\end{bmatrix} = \mathcal{F}_u \left( \begin{bmatrix}
A & B \\
C & D
\end{bmatrix}, \begin{bmatrix} -I & \sqrt{2I} \\
\sqrt{2I} & -I
\end{bmatrix} \right)
\]
(13)

Supposing that $\xi(t) := [w^T(t), w^T(t), u^T(t)]^T$ and $\psi(t) := [z^T(t), z^T(t), y^T(t)]^T$ in Equation (11), a corresponding discrete-time system in the form (1) is readily obtained by applying the bilinear transformation (12). Since $H_\infty$ problems are properly posed in continuous time, the methodology described in the previous section can be applied without restrictions to the transformed system. For the $H_\infty$ performance index $J$ to be well defined in continuous time, the state-space data must be such that the closed-loop feedthrough term of the channel/specification $j$ is zero. Without imposing restrictions to the controller, this is achieved with $\hat{D}_{11j} = 0$ and either $\hat{D}_{12j} = 0$ or $\hat{D}_{11j} = 0$ and $\hat{D}_{21j} = 0$.

Once the LFT discrete-time controller has been computed, one can use the transformation (13) to recover the corresponding continuous-time controller. It is worth mentioning that only the LTI components of the LFT plant and of the LFT controller are modified by bilinear transformations, whereas the $\Delta$- and $\Delta_K$-blocks remain unchanged.

4. POLYTOPIC COORDINATES

Modern flight control systems undergo highly maneuverable trajectories which requires a fast controller update. Controllers designed through general LPV/gridding techniques (Wu et al., 1995b; Apkarian and Adams, 1998) show little conservatism but require more complex on-line computations at the gain-scheduling level. Contrarily, LFT/LPV controllers are often more conservative but their favorable LFT structure offers obvious advantages in this respect. In comparison with the single-objective $H_\infty$ LFT/LPV control methods (Packard, 1994; Apkarian and Gahinet, 1995), the foregoing mixed $H_\infty$ multi-objective approach allows to consider a richer class of scheduling functions (10), instead of replicating the parameter block of the plant ($\Delta_K := \Delta$). This is another factor which reduces conservatism and that is immediately penalized by an increase in complexity of on-line computations. Fast algorithms for the calculation of polytopic coordinates should therefore be utilized in order to overcome this difficulty.

For a parameter evolving in a hyper-rectangle, barycentric coordinates can be directly and quickly computed by ratio of hyper-volumes. The following algorithm describes a procedure for the computation of polytopic coordinates to general hyper-rectangles (6) with vertices in Equation (7):

Algorithm 4.1. Computation of polytopic coordinates

Step 1: Given a parameter $\theta := (\theta_1, \ldots, \theta_r)^T$, compute its normalized coordinates
\[
\bar{\theta}_i := \frac{\theta_i - \theta_i}{(\theta_i - \theta_i)}, \quad l = 1, \ldots, r.
\]

Step 2: For each vertex $\Theta_i$, $i = 1, \ldots, L$, compute the corresponding polytopic coordinates
\[
\alpha_i = \prod_{l=1}^{r} \bar{\theta}_l, \quad \text{where}
\]
\[
\bar{\theta}_l = \begin{cases} 
\theta_l & \text{if } \theta_l \text{ is a coordinate of } \Theta_i \\
1 - \theta_l & \text{if } \theta_l \text{ is a coordinate of } \Theta_i 
\end{cases}
\]

Then, computing polytopic coordinates from measured rectangular coordinates is not a costly procedure. It can be readily performed on line through simple operations basically consisting in ($r$) scalar normalizations and ($Lr - L$) scalar multiplications.

5. MISSILE CONTROL PROBLEM

In this section, we apply the technique presented in Sections 2-4 to a realistic missile gain-scheduling autopilot problem. This problem consists in controlling a missile to track commanded normal acceleration $\eta(t)$, by generating a commanded tail fin deflection $\delta(t)$.

The nonlinear pitch-axis missile model and actuator dynamics are available in References (Reichert, 1992; Nichols et al., 1993). They involve the angle of attack $\alpha(t)$, the pitch-rate $q(t)$ and the tail deflection angle $\delta(t)$ and its derivative $\dot{\delta}(t)$. Normal acceleration $\eta(t)$
Table 1. Decision variables

<table>
<thead>
<tr>
<th>Set</th>
<th>Variables</th>
<th>Dimension</th>
<th>Number of scalar variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_0$</td>
<td>$U, V_{ij}, W_{ij}$</td>
<td>$n \times n$</td>
<td>$N \times N$</td>
</tr>
<tr>
<td>$K_w$</td>
<td>$A_{K_w}, B_{K_w}, C_{K_w}, D_{K_w}$</td>
<td>$n \times n$</td>
<td>$(n + N)(m_1 + p_2) + m_2p_2$</td>
</tr>
<tr>
<td>$A_{K_v}$</td>
<td>$A_{K_v}, i = 1, \ldots, L$</td>
<td>$N \times N$</td>
<td>$N^{\alpha}$</td>
</tr>
<tr>
<td>$H_{z}$</td>
<td>$X_{zj}, Z_j, Q_{zj}, R_{1j}, R_{2j}$</td>
<td>$2n \times 2n$</td>
<td>$n(2n + 1)$</td>
</tr>
<tr>
<td>$H_{w}$</td>
<td>$X_{wj}, Q_{wj}, R_{wj}$</td>
<td>$2n \times 2n$</td>
<td>$n(2n + 1)$</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
W_{ij} &= W_{ij}^T \quad \text{for } i, j \in [0, L], \\
G(s) &= G(s)^T, \\
\Delta(\theta) &= \Delta(\theta)^T, \\
\theta_0(t) &= \theta_0(t)^T.
\end{align*}
\]

The precompensator $W_i(s)$ is used to achieve the command shaping. The weighting functions $W_i(s)$ and $W_e(s) := \text{diag}(W_i^T(s), 0.01)$ penalize the tracking error and $W_e(s)$ incorporates bounds on the norm of unmodeled dynamics and also reflects magnitude restriction on the control signal.

Hence, the specifications above can be met by a controller $K(s)$ together with its scheduling function $\Delta_K(\theta_0(t)), |\theta_0(t)| = \frac{|a(t)|}{15} \leq 1$, which:

- minimize the $L_2$-induced gain $\gamma_M$ of the operator mapping $z_M$ into $w_M$,
- maintain the variance of $z_\varepsilon$ due to the disturbance $\eta_c$ below an appropriate bound $\gamma_s$,
- guarantee an upper bound $\gamma_u$ on the $L_2$-induced gain of the operator mapping $\eta_c$ into $z_u$.

for all trajectories $\alpha(t) \in [-30, 30]$ degrees.

Then, this problem can be solved by running Algorithm 2.1 and consists in finding an adequate compromise between three conflicting objectives over the entire operating range: one $H_z$ and two $H_w$ specifications. Note that such a problem cannot be solved by earlier LPV methodologies for plants described by LFT representations.

The discrete-time synthesis plant $P(z)$ and the final continuous-time controller $K(s)$ are computed through bilinear transformations, respectively from $P(s)$ and the designed $K(z)$, as indicated in Section 3. The continuous-time synthesis plant $P(s)$ is readily obtained from the connections in Figure 2 and incorporates the missile model $G(s)$ and the weighting functions, $W_i(s), W_e(s)$, and $W_e(s)$. These frequency-dependent weights have been tuned by performing a few trials-and-errors of synthesis and simulations for the nominal plant. That is, an LTI plant model obtained from the linearization about the point $\theta = [0, 0]^T, \Delta = 0$. The adopted frequency shapes for the filters are fairly standard and their numerical data are available in the Reference (Pellanda et al., 2001).

We adopt the closed-loop control structure depicted in Figure 2. The LFT missile model $F_0(G(s), \Delta(\theta))$ is derived in the full version of this paper (Pellanda et al., 2001). In order to utilize the approach discussed in this paper, we express the performance objectives by choosing appropriate weighting functions.

5.1 Performance objectives and control structure

The performance and robustness specifications for the closed-loop system are similar to those detailed in References (Wu et al., 1995a; Nichols et al., 1993). Our aim is to maintain robust stability over the entire operating range, $\alpha \in [-30, 30]$ degrees and $M \in [2, 4]$, and to track step commands in $\eta_c$ with time constant no more than 0.35 s, maximum overshoot of 10%, steady-state error less than 1% and an adequate high-frequency roll-off for noise attenuation and withstand neglected high frequency dynamics and flexible modes. In order to avoid saturation of the actuator, the maximum tail deflection rate for 1g step command in $\eta_c$ should not exceed $25^\circ/s$.

We adopt the closed-loop control structure depicted in Figure 2. The LFT missile model $F_0(G(s), \Delta(\theta))$ is derived in the full version of this paper (Pellanda et al., 2001). In order to utilize the approach discussed in this paper, we express the performance objectives by choosing appropriate weighting functions.

\[
\begin{align*}
\text{Table 1. Decision variables}
\end{align*}
\]
5.2 Results and simulations

In order to put in light the potentials of our multi-channel LPV synthesis method and to allow comparisons, we have considered two designs. The first LPV controller, \( K_1(s) \) and \( \Delta K_1(\theta_a) \), has been synthesized considering \( M \) as a constant (= 3); the second one, \( K_2(s) \) and \( \Delta K_2(\theta_a) \), considers \( M \) as a bounded uncertain parameter \((M \in [2, 4])\). In short, we have used the following strategy to compute these controllers:

- \( K_1(s) \) and \( \Delta K_1(\theta_a) \):
  - Starting with a small value \( v_c \), synthesize controllers which minimize the \( H_{\infty} \) performance \( \gamma_a \) subject to a \( H_2 \) constraint \( \sqrt{v_c} \).
  - Through successive relaxations in \( v_c \), find a reasonable compromise between these objectives. To check out when a good balance has been achieved, perform non-stationary \((\alpha(t))\) and nonlinear simulations for \( M = 3 \) and evaluate the closed-loop performance in the time domain.

- \( K_2(s) \) and \( \Delta K_2(\theta_a) \):
  - As mentioned in the previous subsection and analogously to \( K_1(s) \), minimize \( \gamma_a \) subject to the constraints \( \gamma_a \) and \( v_c \).
  - Starting with the final values \( \gamma_a \) and \( v_c \) obtained in designing \( K_1(s) \), relax them alternately in order to find an adequate balance between the three objectives.

Numerical data for \( K_2(s) \) and its scheduling function \( \Phi_2 \)'s are provided in the Reference (Pellanda et al., 2001).

Nonlinear simulation results for fixed values of \( M \) are displayed on Figure 3. Figure 4 shows nonlinear simulations for time-varying \( M(t) \). The input is a sequence of step commanded acceleration \( \eta_c \) whose amplitudes have been chosen such that the parameter \( \alpha \) covers most of the scheduling range, thus inducing significant variations in the aerodynamic coefficients. The Mach number time trajectory has been generated as in References (Nichols et al., 1993; Wu et al., 1995a). As theoretically expected, all performance objectives are met for all considered trajectories when \((K_2, \Delta K_2)\) is employed for controlling the system. In contrast, the desired closed-loop behavior is satisfied only at the central point \((M = 3)\) for \((K_1, \Delta K_1)\). We recall that \((K_2, \Delta K_2)\) has been computed in order to ensure robustness with respect to variations in the Mach number through an extra \( H_{\infty} \) constraint on the Mach channel \( M \). From the later result, the advantages of using this multi-objective LPV synthesis method became evident.

6. CONCLUSIONS

We have discussed a multi-objective/channel \( H_2/H_{\infty} \) LPV control technique for the design of a missile autopilot over a broad range of operating conditions in both the angle of attack and the Mach number. The proposed method provides additional flexibility to handle various and stringent specifications attached to the missile problem while maintaining the same operational simplicity as earlier single-objective LPV techniques:

- the missile nonlinearities are captured through the use of an LFT representation,
- different channels are defined to translate tracking performance, control limitation and robustness properties,
- balancing the different design requirements is carried out in a very natural way within the proposed design framework and conservatism is kept reasonable thanks to the use of different Lyapunov and scaling variables for each channel/specification,
- besides, we describe simple schemes to control the controller scheduling function and show how all these manipulations carry over the continuous-time case.

The determination of a full genuine continuous-time methodology remains, however, challenging and future research should be oriented in this direction. Also, we think that other practical reasons might dictate the use of observer-based LPV controllers. This is a delicate and seemingly untouched topic that will be considered in a future study.

7. REFERENCES


Fig. 3. Nonlinear closed-loop response for fixed $M$: $(K_1, \Delta K_1)$ on the left and $(K_2, \Delta K_2)$ on the right

Fig. 4. Nonlinear closed-loop response using $(K_2, \Delta K_2)$ for time-varying $M$


