Evolutionnary Fixed-Structure LPV/LFT Controller Synthesis for Multiple Plants

Philippe Feyel
Safran Electronics & Defense, Massy, France

Keywords: LPV/LFT Synthesis, H_∞ Synthesis, µ Synthesis, Fixed-Structure, Multiple Plants, Small Gain Theorem, Evolutionary Computation, Robust Control, Adaptive Control.

Abstract: This paper proposes to address the problem of fixed-structure gain-scheduled LPV/LFT controllers for plants with time-varying measurable and time invariant unmeasurable uncertainties. Due to the complexity of merging µ-technics with LPV/LFT approach, an alternative presented here consists in computing robust fixed-structure LPV/LFT controllers using the multiple plants framework instead of µ-technics. The complexity of this optimization problem is tackled with global evolutionary optimization. This paper shows that this approach is quite efficient and very simple to implement. The algorithm has been tested on the pendulum in the cart academic example.

1 INTRODUCTION

Since few decades H_∞ synthesis has proved to be a powerful tool to compute robust controllers due to the merge of the small gain theorem and the concept of standard form for control (Zhou *et al.,* 1996). Lots of applications can be found in literature and more recently in the structured framework (Apkarian *et al,* 2000).

When the structure of uncertainties is well-known and more especially in block-diagonal form (which is systematic when modelling the plant using the LFT framework), some more complex but less-conservative extensions have been developed leading to more performing robust controllers.

If the uncertainties are bounded but unknown, one can compute robust controllers using the µ-synthesis technics (Young *et al.,* 1990), which is based on the structured singular value concept. µ-synthesis is often solved using sub-optimal heuristics such as D-K iteration, D-G-K iteration, etc. More recently the use of modern optimization technics such as (Apkarian, 2011) or (Feyel *et al.*, 2014a) allows the computation of µ-optimal structured controllers.

When the structured uncertainty is bounded and measured, one can use the LPV approach which consists in enforcing the searched controller to have the same varying parameters dependency as the plant to be controlled. The stability along parameters trajectories is ensured using the small gain theorem. The controller can be computed using either the polytopic framework (Apkarian *et al.,* 1993) or the LFT modelling framework (Apkarian *et al.,* 1995) which appears to be less conservative and more general. More recently the use of modern optimization technics allows the computation of LPV/LFT optimal structured controllers (Shi *et al.,* 2010).

In this work we are interested in computing a robust fixed-structure LPV/LFT controller. The term *robust* means here that the structured uncertainty is partially known as in Figure 1 where Δ_2 denotes the unknown part of the uncertainty block and $\Delta_1(t)$ denotes the well-known part. Because the LFT modelling framework is common to the LPV/LFT technic and µ-synthesis technic, some works have tried to directly merge those two technics leading generally to the non-convex synthesis problem of computing the LPV/LFT controller and some corresponding augmented scalings (Apkarian *et al.,* 1995) (Blue *et al.*, 1997); thus the problem is usually addressed using some sub-optimal heuristics (DeVito *et al.*, 2010) but without guaranty of the global optimality of the computed LPV controller.

As an alternative to µ-synthesis, the multiple plants H_∞-synthesis has emerged and has proved to be a good compromise between the complexity of

Feyel, P.
Evolutionnary Fixed-Structure LPV/LFT Controller Synthesis for Multiple Plants.
DOI: 10.5220/0006858101560165
In *Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2018) - Volume 1*, pages 156-165
ISBN: 978-989-758-321-6

the µ-synthesis technic and the conservatism of the non-structured uncertainty framework (Apkarian, 2002), (Feyel *et al*, 2014b). Based on the same idea, we propose in this work to extend the LPV/LFT synthesis approach to the multiple plants framework; the main idea is to use the multiple plants framework instead of µ-technics to make the LPV/LFT controller be robust against unknown uncertainties.

The paper is composed as follows: in section 2 we recall the basics of the LPV/LFT controller synthesis problem and the multiple plants extension is proposed. In Section 3, the perturbed differential evolution algorithm in described in order to solve the LPV/LFT problem described in section 4. Finally an example showing the efficiency of the method is proposed in section 5

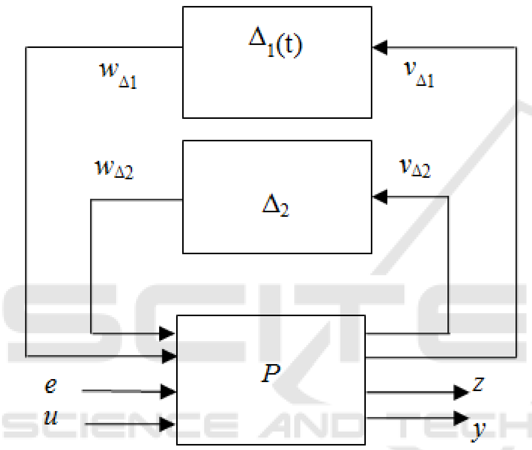

Figure 1: The LFT modelling framework with well-known (Δ_1(t)) and unknown (Δ_2) uncertainty blocks.

2 THE LPV/LFT CONTROLLER SYNTHESIS PROBLEM

2.1 Notation

First we consider the system in LFT form depicted in Figure 2a where $\Delta(t)$ represent the varying uncertainties. $(z,e)\in R^{n_z\times n_e}$ refers to the performance channels. $y\in R^{n_y}$ are the measures and $u\in R^{n_u}$ is the control signal. P is a LTI plant and we have:

$$\begin{pmatrix} z \\ y \end{pmatrix} = F_u\left(P,\Delta(t)\right)\begin{pmatrix} e \\ u \end{pmatrix} \quad (1)$$

The uncertainty block $\Delta(t)$ is assumed to be block-diagonal structured:

$$\Delta(t) = blockdiag\left(\theta_1(t)I_{r_1},\dots,\theta_{n_\Delta}(t)I_{r_{n\Delta}}\right) \quad (2)$$

where $r_i>1$ when the i^{th} time varying parameter $\theta_i(t)$ is repeated in $\Delta(t)$ and:

$$r = \sum_{i=1}^{n_\Delta} r_i \quad (3)$$

Thus $(v_\Delta,w_\Delta) \in R^{r\times r}$ refers to the uncertainty channels.

We define $\underline{\Delta}$ as the set of matrices with the same structure as $\Delta(t)$:

$$\underline{\Delta} := \left\{\begin{matrix} diag\left(\theta_1(t)I_{r1},\dots,\theta_{n_\Delta}(t)I_{r_{n_\Delta}}\right), \\ \theta_i(t)\in R \end{matrix}\right\}. \quad (4)$$

2.2 Principle of LPV Synthesis for Multiple Plants

2.2.1 Basics of the LPV/LFT Controller Synthesis Problem

To ensure the stability and the performance in the LPV framework, we seek a controller with the same parametric dependence as the system to be controlled. Then the controller will be adjusted depending on the evolution of time varying parameters, which are supposed to be measured or estimated. Thus the controller is a LPV system with the following form (Apkarian *et al,* 1995):

$$u = F_l\left(K,\Delta(t)\right)y \quad (5)$$

where K is LTI. Considering the Figure 2b, the close-loop between z and e is written:

$$T\left(P,K,\Delta\right)= F_l\left(F_u\left(P,\Delta(t)\right),F_l\left(K,\Delta(t)\right)\right). \quad (6)$$

The problem is to find a LTI controller K which:

- internally stabilizes the closed-loop $T(P,K,\Delta)$ for all uncertainties such as $\gamma^2\Delta^T(t)\Delta(t)\le 1$,
- $\left\|T\left(P,K,\Delta\right)\right\|_2 \le \gamma$, where $\|\ \|_2$ is the L_2-induced norm.

Following the same idea as in (Apkarian *et al,* 1995), the two schemes depicted in Figure 2b and Figure 3 are strictly equivalent.

Introducing the new outputs to survey $v_k \in R^r$, the new exogenous inputs $w_k \in R^r$, the new measures $y_k \in R^r$ and the new control signals $u_k \in R^r$, an augmented plant P_a(s) can be defined in (7).

$$\begin{pmatrix} v_K \\ v_\Delta \\ z \\ y \\ y_K \end{pmatrix} = \begin{pmatrix} 0 & 0 & I_r \\ 0 & P(s) & 0 \\ I_r & 0 & 0 \end{pmatrix} \begin{pmatrix} w_K \\ w_\Delta \\ e \\ u \\ u_K \end{pmatrix} = P_a(s) \begin{pmatrix} w_K \\ w_\Delta \\ e \\ u \\ u_K \end{pmatrix} \qquad (7)$$

Thus, the LPV synthesis problem can be viewed as a more classical performance robustness one applied to the nominal plant $P_a(s)$ towards the uncertainty block $diag(\Delta(t),\Delta(t))$.

In the following, the repeated structure is noted $\underline{\Delta}\oplus\underline{\Delta}$:

$$\underline{\Delta}\oplus\underline{\Delta} := \left\{ blockdiag\left(\Delta(t),\Delta(t)\right) : \Delta(t) \in \underline{\Delta} \right\} \qquad (8)$$

Now we consider the set of scalings positive definite associated with the structure $\underline{\Delta}$:

$$L_{\underline{\Delta}} = \left\{ L > 0 : L\Delta = \Delta L,\ \forall \Delta \in \underline{\Delta} \right\} \subset \mathbb{R}^{r\times r} \qquad (9)$$

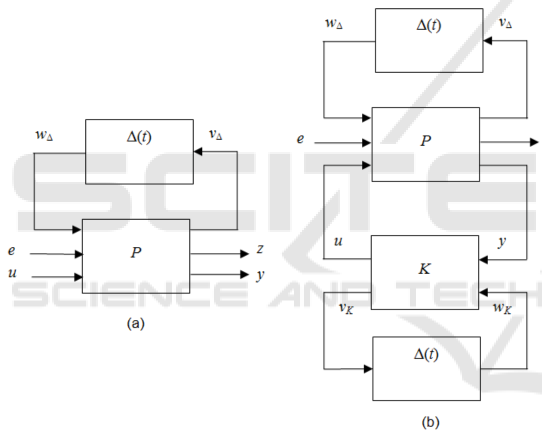

Figure 2: The LFT/LPV model (a) and the LPV closed-loop scheme (b).

The set of scalings positive definite commuting with the structure $\underline{\Delta}\oplus\underline{\Delta}$ is then defined by:

$$L_{\underline{\Delta}\oplus\underline{\Delta}} = \left\{ \begin{array}{l} \begin{pmatrix} L_1 & L_2 \\ L_2^T & L_3 \end{pmatrix} > 0 \\ \\ L_1, L_3 \in L_{\underline{\Delta}} \text{ and} \\ L_2\Delta = \Delta L_2,\ \forall \Delta \in \underline{\Delta} \end{array} \right\} \qquad (10)$$

According to (Apkarian *et al,* 1995), $F_l(K,\Delta(t))$ is a γ-suboptimal gain-scheduled H∞-controller if there exists a scaling $L \in L_{\underline{\Delta}\oplus\underline{\Delta}}$ and a LTI control structure K such that the nominal closed-loop system $F_l(P_a,K)$ is internally stable and satisfies:

$$\left\| \begin{pmatrix} L^{1/2} & 0 \\ 0 & I_{n_z\times n_z} \end{pmatrix} F \begin{pmatrix} L^{-1/2} & 0 \\ 0 & I_{n_e\times n_e} \end{pmatrix} \right\|_\infty < \gamma, \qquad (11)$$

$$F = F_l\left(P_a(s), K(s)\right)$$

Assuming that such a controller exists and that:

$$K := \begin{pmatrix} A_K & B_K \\ C_K & D_K \end{pmatrix} := \begin{pmatrix} A_K & \begin{bmatrix} B_{K_y} & B_{K_\Delta} \end{bmatrix} \\ \begin{bmatrix} C_{K_y} \\ C_{K_\Delta} \end{bmatrix} & \begin{bmatrix} D_{K_{yy}} & D_{K_{y\Delta}} \\ D_{K_{\Delta y}} & D_{K_{\Delta\Delta}} \end{bmatrix} \end{pmatrix} \qquad (12)$$

Then the gain-scheduled controller $F_l(K,\Delta(t))$ has the state-space implementation (13).

$$K(\Delta) := F_l\left(K, \Delta(t)\right) := \begin{pmatrix} A_K' & B_K' \\ C_K' & D_K' \end{pmatrix}$$

$$\begin{aligned} A_K' &= A_K + B_{K_\Delta} M_\Delta C_{K_\Delta} \\ B_K' &= B_{K_y} + B_{K_\Delta} M_\Delta D_{K_{\Delta y}} \\ C_K' &= C_{K_y} + D_{K_{y\Delta}} M_\Delta C_{K_\Delta} \\ D_K' &= D_{K_{yy}} + D_{K_{y\Delta}} M_\Delta D_{K_{\Delta y}} \\ M_\Delta &= \Delta(t)\left(I - D_{K_{\Delta\Delta}}\Delta(t)\right)^{-1} \end{aligned} \qquad (13)$$

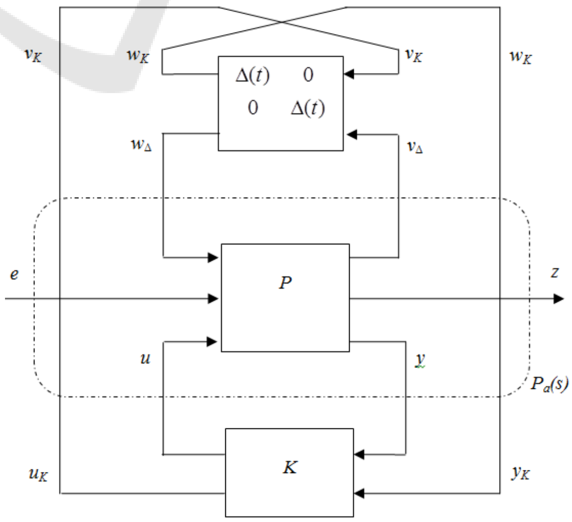

Figure 3: An equivalent LPV closed-loop scheme.

2.2.2 The Proposed Multiple Plants Extension

As said in introduction, we propose to use the multiple plants framework instead of µ-technics to make the LPV/LFT controller be robust against unknown uncertainties. The unknown uncertainties allow us to define a set of m plants with equation (14).

$$\wp = \{P_i(s),\ i = 1,\dots,m\} \quad (14)$$

As depicted in Figure 4 and according to the previous paragraph, $F_l(K,\Delta(t))$ is a robust γ-suboptimal gain-scheduled H_∞-controller if there exists a scaling $L \in L_{\Delta\oplus\Delta}$ and a LTI control structure K such that each closed-loop system $F_l\left(P_{a_i}(s),K\right)$ is internally stable and satisfies:

$$\max_{i=1,\dots,m}\left[\left\|\begin{pmatrix} L^{1/2} & 0 \\ 0 & I_{n_z\times n_z}\end{pmatrix} F_i \begin{pmatrix} L^{-1/2} & 0 \\ 0 & I_{n_e\times n_e}\end{pmatrix}\right\|_\infty\right] < \gamma \quad (15)$$
$$F_i = F_l\left(P_{a_i}(s), K(s)\right)$$

where each $P_{a_i}(s)$ is defined by equation (7) with $P_i(s)$ instead of $P(s)$.

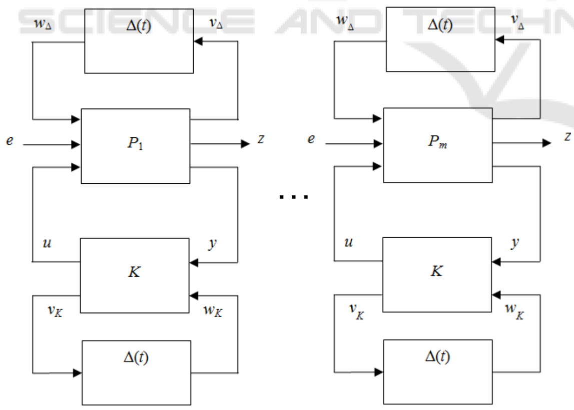

Figure 4: The multiple plants LPV/LFT controller synthesis problem.

Due to the complexity of the posed problem, we propose to use the evolutionary algorithm described below to solve it.

3 THE PERTURBED DIFFERENTIAL EVOLUTION (PDE) ALGORITHM

The Differential Evolution (DE) algorithm (Storm *et al*, 1995) is a recent metaheuristic which belongs to the class of evolutionary algorithms (like genetic algorithms for instance). Such a stochastic algorithm is helpful for minimizing a nonlinear function $f(X)$ whose gradient cannot be computed (for instance if $f(X)$ is not differentiable) so that classical methods cannot be used. Here the only requirement is the capability of evaluating function $f(X)$, called the fitness. As a main drawback of such a stochastic algorithm, the result of the optimization problem has to be considered in a statistical way but a good solution (that is near the global optimum) is often found. The reader can have a good introduction of metaheuristic methods in (Dréo *et al*, 2006).

3.1 Description

To introduce the DE algorithm, we consider the problem of finding X_{opt} so that $X_{opt} = \underset{X\in\Psi}{\operatorname{argmin}}\left(f(X)\right)$ where $\Psi \subset R^n$ is the search space for X.

In the following, rnd(x,y) designed a random value generated by a uniform distribution on the interval [x, y].

A description of the classical DE algorithm is as follows.

Step 0: Initialisation

Construct an initial population P with N individuals:

$$P = \{X_1,\dots,X_i,\dots,X_N\}. \quad (16)$$

Each individual is defined by its n genes:

$$X_i = (x_{i1},\dots,x_{ij},\dots,x_{in})^{\mathrm{T}}. \quad (17)$$

Thus n is also the problem dimension.

Assuming that $x_j \in \left[\underline{x}_j, \overline{x}_j\right]$, the j^{th} component of the i^{th} individual is randomly chosen in its definition interval:

$$x_{ij}^{(0)} = \mathrm{rnd}\left(\underline{x}_j, \overline{x}_j\right)_{ij} \quad (18)$$
$$\text{with } i = 1,\dots,N \text{ and } j = 1,\dots,n$$

Now the k^{th} iteration consists of the three following steps.

Step 1: Mutation

Usually two mutation schemes can be considered, the rand one and the best one, as defined in (19).

$$\begin{aligned} &\text{Rand} \rightarrow U_i^{(k+1)} = X_{a_i}^{(k)} + F\left(X_{b_i}^{(k)} - X_{c_i}^{(k)}\right) \\ &\text{Best} \rightarrow U_i^{(k+1)} = X_{best}^{(k)} + F\left(X_{a_i}^{(k)} - X_{b_i}^{(k)}\right) \end{aligned} \quad (19)$$

where $X_{a_i}, X_{b_i}, X_{c_i}$ are different and randomly chosen in the population and X_{best} is the best individual (with respect to the fitness) since the beginning of optimisation. $F \in [0, 2]$ is a number called the mutation factor. At each iteration, N mutants U_i are defined according to one of these rules.

Note that usually the rand scheme encourages diversity whereas the best scheme encourages fast convergence, but very often to a suboptimal solution. That is why we propose here to use the mutation scheme depicted in (20) which is a merge of the two previous schemes and known as the "DE/rand-to-best/1" mutation scheme.

$$\begin{aligned} \text{Rand to Best} \rightarrow U_i^{(k+1)} &= X_{a_i}^{(k)} + F\left(X_{b_i}^{(k)} - X_{c_i}^{(k)}\right) \\ &+ F'\left(X_{best}^{(k)} - X_{a_i}^{(k)}\right) \end{aligned} \quad (20)$$

Where $F' \in [0,2]$ is another mutation factor.

Finally to avoid stagnation during the optimization process, we decide to perturb the mutant obtained in (20) by the rule (21).

$$\begin{aligned} U_i^{(k+1)} &= U_i^{(k+1)} + \text{rnd}(-1,1) X_{d_i} \\ &\text{if } \text{rnd}(0,1) < p_f \end{aligned} \quad (21)$$

Where p_f is the probability of perturbation and X_{d_i} is different from $X_{a_i}, X_{b_i}, X_{c_i}$ and randomly chosen in the population.

Step 2: Cross-over

By crossing-over $U_i^{(k+1)}$ with $X_i^{(k)}$, a new individual $V_i^{(k+1)}$ is generated with genes defined as follows:

$$v_{ij}^{(k+1)} = \begin{cases} u_{ij}^{(k+1)} & \text{if } \text{rnd}(0,1)_{ij} \le C_r \text{ or if } j = j_i \\ x_{ij}^{(k)} & \text{else} \end{cases} \quad (22)$$

where $C_r \in [0.1, 0.9]$ is the crossing-rate and j_i an integer number randomly chosen in $\{1,2,...,N\}$. As the population moves towards its bounds, the bounce-back method can be used to generate vectors that will be located even closer the bounds (Storm *et al,* 1995).

Step 3: Selection

The population is updated with individuals defined by:

$$X_i^{(k+1)} = \begin{cases} V_i^{(k+1)} & \text{if } f\left(V_i^{(k+1)}\right) \le f\left(X_i^{(k)}\right) \\ X_i^{(k)} & \text{else} \end{cases} \quad (23)$$

If the stopping criterion is satisfied, then return best solution found so far, otherwise go to step 1.

The stopping criterion can be:

- a maximum number of iterations,
- the convergence of the algorithm which is detected when all individuals tend to be similar and centered around the best one, that is when (24) is verified.

$$\frac{\max_{i=1,\dots,N} \left\| X_i^{(k+1)} - X_{best}^{(k+1)} \right\|}{\left\| \overline{x} - \underline{x} \right\|} < \varepsilon \quad (24)$$

3.2 Settings for Control Problems

One of the main advantages of this heuristic approach is its low number of tuning parameters. In this work we use the classical values given by (Storm *et al,* 1995) for F and C_r:

- The mutation factor: $F = 0.75$,
- The cross-over rate: $C_r = 0.8$,

A good balance between exploration and convergence is achieved by enforcing: $F' = 1 - F = 0.25$.

The mutant is perturbed with a probability $p_f = 0{,}025$.

The convergence threshold is set by default to: $\varepsilon = 0{,}1\%$.

Although a population size between $5n$ and $10n$ is generally advised, we use the same idea as (Clerc, 2012) by setting the size of the population with the following rule:

$$N = \text{floor}\left(10 + \sqrt{n}\right) \quad (25)$$

The number of iterations depends on the problem and will be specified later and is not very sensitive.

4 EVOLUTIONNARY FIXED-STRUCTURE LPV/LFT SYNTHESIS FOR MULTIPLE PLANTS

4.1 Notations

Given the controller K with state-space (12) and the plant $P_a(s)$ defined in (26).

$$P_a := \left(\begin{array}{c|c:c} A_a & B_{1a} & B_{2a} \\ \hline C_{1a} & D_{11a} & D_{12a} \\ \hdashline C_{2a} & D_{21a} & D_{22a} \end{array}\right) \tag{26}$$

The closed-loop $F_l(P_a,K)$ has the following state-space representation:

$$\begin{aligned}
&F_l\left(P_a(s),K(s)\right)=\begin{pmatrix} A_{cl} & B_{cl} \\ C_{cl} & D_{cl}\end{pmatrix}\\
&A_{cl}=\begin{pmatrix} A_a+B_{2a}M_{1a}D_kC_{2a} & B_{2a}M_{1a}C_k \\ B_kM_{2a}C_{2a} & A_k+B_kM_{2a}D_{22a}C_k\end{pmatrix}\\
&B_{cl}=\begin{pmatrix} B_{1a}+B_{2a}M_{1a}D_kD_{21a} \\ B_kM_{2a}D_{21a}\end{pmatrix}\\
&C_{cl}=\left(C_{1a}+D_{12a}M_{1a}D_kC_{2a} \quad D_{12a}M_{1a}C_k\right)\\
&D_{cl}=D_{11a}+D_{12a}M_{1a}D_kD_{21a}\\
&M_{1a}=\left(I-D_kD_{22a}\right)^{-1},\ M_{2a}=\left(I-D_{22a}D_k\right)^{-1}
\end{aligned} \tag{27}$$

We note $\lambda(A)$ the set of eigenvalues of A. Then the close-loop defined in (27) is internally stable iff (28) holds.

$$\max\left(\text{real}\left(\lambda\left(A_{cl}\right)\right)\right)<0 \tag{28}$$

4.2 Fitness Function Definition and Optimization Scheme

As said before, we propose in that work to use evolutionary computation to find simultaneously a structured optimal controller K and optimal scalings by solving the following optimization problem:

$$\begin{gathered}
\min_{L_1,L_2,L_3,K}\left[\max_{i=1,\ldots,m}\gamma_i\right]\\
\textit{subject to}\\
L\in \mathbf{L}_{\Delta\oplus\Delta}\\
F_l\left(P_{a_i},K\right)\in \mathrm{R}H_\infty,\ i=1,\ldots,m
\end{gathered} \tag{29}$$

With:

$$\begin{aligned}
&\gamma_i=\left\|\begin{pmatrix} L^{1/2} & 0 \\ 0 & I_{n_z\times n_z}\end{pmatrix}F_i\begin{pmatrix} L^{-1/2} & 0 \\ 0 & I_{n_e\times n_e}\end{pmatrix}\right\|_\infty\\
&F_i=F_l\left(P_{a_i}(s),K(s)\right)
\end{aligned} \tag{30}$$

where each $P_{a_i}(s)$ is defined by equation (7) with $P_i(s)$ instead of $P(s)$.

The problem (29) can be rewritten:

$$\min_{X\in\left[\underline{x};\overline{x}\right]} f(X) \tag{31}$$

The unknown X stands for the coefficients of the state-space matrices (A_k, B_k, C_k, D_k) of K and for those of the scalings L_j (j=1,2,3) which are symmetric (thus only the upper part of matrices are searched) and have to commute with Δ.

Note that:

$$\begin{gathered}
M \text{ symetric non positive definite}\\
\Leftrightarrow\\
-\min_i\left(\lambda_i(M)\right)\geq 0.
\end{gathered} \tag{32}$$

Thus:

$$\begin{gathered}
L\notin \mathbf{L}_{\Delta\oplus\Delta}\\
\Leftrightarrow\\
-\min\left(\min_i\lambda_i(L_1),\min_i\lambda_i(L_3),\min_i\lambda_i(L)\right)\geq 0.
\end{gathered} \tag{33}$$

The fitness function $f(X)$ is given by the evaluation of the following flowchart:

- Build (A_k, B_k, C_k, D_k) from X,
- Build each A_{cl_i} according to (27) for i=1,…,m,
- Build each L_j (j=1,2,3) from X,
- Build scaling L from each L_j according to (10),
- Evaluate the global spectral abscissa of the set of closed-loop plants:

$$\overline{\lambda}=\max_{i=1,\ldots,m}\left[\max\left(\text{real}\left(\lambda\left(A_{cl_i}\right)\right)\right)\right]. \tag{34}$$

- According to (33) evaluate:

$$\underline{\lambda}=-\min\left(\min_i\lambda_i(L_1),\min_i\lambda_i(L_3),\min_i\lambda_i(L)\right) \tag{35}$$

- If max($\overline{\lambda}$, $\underline{\lambda}$) ≥ 0, evaluate:

$$f(X)=\max\left(\overline{\lambda},\underline{\lambda}\right), \tag{36}$$

- Else :
  - Build each $F_l\left(P_{a_i}(s),K\right)$ according to (30),

- Compute each γ_i according to (30),
- Evaluate :

$$f(X) = -\frac{1}{\max\limits_{i=1,\ldots,m} \gamma_i}. \quad (37)$$

As done in (Feyel *et al*, 2014a) and (Feyel *et al*, 2014b) to increase the rate of convergence, we use the tridiagonal form for the state-matrix A_k (which limits the number of unknowns) and a transformation of the search space interval is performed to increase the sensitivity of the algorithm.

5 EXAMPLE

5.1 Specification

The proposed method has been tested for the pendulum in the cart depicted in the Figure 5. The system, denoted H, can be modelled by the following simplified equations:

$$H := \begin{cases} L\,\dot{i}(t) + R\,i(t) + \varphi\,\omega(t) = u(t) \\ J(t)\dot{\omega}(t) + f\,\omega(t) + d(t) = \varphi\,i(t) \\ \dot{x}_c(t) = \dfrac{r}{N}\omega(t) \\ \ddot{x}_c(t) + \dot{v}_l(t) + \alpha\,\dot{\phi}(t) + g\,\phi(t) = 0 \\ v_l(t) = l(t)\dot{\phi}(t) \end{cases} \quad (38)$$

Where:

- $i(t)$, $u(t)$: current, voltage in the motor;
- $\omega(t)$: rotation speed of the motor;
- $x_c(t)$: position of the cart;
- $\varphi(t)$: angle of the pendulum;
- $d(t)$: disturbance torque;

Definitions and nominal values of parameters are given in Table 2; $\varphi(t)$ and $x_c(t)$ are measured with measurement gain k_x and k_φ.

The specification required is:

- Tracking the reference defined in Figure 7,
- No steady-state error, overshoot lower than 0,01 m, $|\varphi(t)| < 0,1$ rad, time response less than 6s,
- $|u(t)| < 15$ V,
- Good stability margins.

5.2 Standard H∞ Synthesis

We consider the H∞-control scheme depicted in Figure 6; it is easy to verify that the following weights satisfy the specification with the nominal plant.

$$\begin{gathered} W_1(s) = \frac{500s + 866}{1000s + 0,866} \\ W_2(s) = \frac{99,5s + 200}{0.995s + 2000} \\ W_3 = 0,01;\ W_4 = 2;\ W_5 = 0,1;\ W_6 = 1 \end{gathered} \quad (39)$$

Thus a 3 DOF controller with order 7 is obtained. The corresponding position response of the cart is given in Figure 7.

5.3 Standard Polytopic LPV/LFT Synthesis

In fact two measurable parameters are time varying in the following intervals:

- $l(t) \in [0,1 ; 0,3]$, the length of the pendulum;
- $J(t) \in [3.10^{-6} ; 50.10^{-6}]$, the inertia of the motor.

Typical trajectories for $J(t)$ and $l(t)$ are given in Figure 8. As we can see in Figure 9, those variations are disturbing the cart time response and the previous standard H∞ controller doesn't succeed in stabilizing the time varying plant. Assuming $J(t)$ and $l(t)$ measurable, we compute a gain-scheduled controller first using the polytopic approach (Apkarian *et al,* 1993) so that the specification remains satisfied in spite of parameters variations. Thus we pose as varying parameters:

$$p_1 = l(t)^{-1},\ p_2 = J(t)^{-1} \quad (40)$$

So that the polytope is defined by:

$$\begin{gathered} 3{,}33\ \text{m}^{-1} < p_1 < 10\ \text{m}^{-1} \\ 0{,}02.10^6\ \text{kg}^{-1}.\text{m}^{-2} < p_2 < 0{,}333.10^6\ \text{kg}^{-1}.\text{m}^{-2} \end{gathered} \quad (41)$$

And:

$$\begin{gathered} p_1 = \theta_1\,\underline{p}_1 + (1-\theta_1)\,\overline{p}_1;\ p_2 = \theta_2\,\underline{p}_2 + (1-\theta_2)\,\overline{p}_2 \\ \theta_1 = \frac{\overline{p}_1 - p_1(t)}{\overline{p}_1 - \underline{p}_1}\ \ ;\ \ \theta_2 = \frac{\overline{p}_2 - p_2(t)}{\overline{p}_2 - \underline{p}_2} \end{gathered} \quad (42)$$

The polytopic LPV plant is thus on the form:

$$\begin{aligned} H_{LPV} &= \theta_1\theta_2\,H\left(\underline{p}_1, \underline{p}_2\right) + (1-\theta_1)\theta_2\,H\left(\overline{p}_1, \underline{p}_2\right) \\ &+ \theta_1(1-\theta_2)\,H\left(\underline{p}_1, \overline{p}_2\right) + (1-\theta_1)(1-\theta_2)\,H\left(\overline{p}_1, \overline{p}_2\right) \end{aligned} \quad (43)$$

As we can see in Figure 9, the LPV plant is thus stabilized.

5.4 Robust Fixed-structure LPV/LFT Synthesis

In addition of the previous (measured) parameters variations, the friction coefficient f is also uncertain (and unmeasured) and can vary in the interval $f \in \left[0\ ;13.10^{-5}\right] N.(m.s^{-1})^{-1}$. As we can see in Figure 10, the cart response is really disturbed and even the LPV polytopic previous controller doesn't succeed in stabilizing the cart. Thus we propose to use our multiple-plants approach to compute a more robust LPV controller. We proceed in two steps:

- The first one consists in modelling the LPV plant using the LPV/LFT modelling framework,
- The second one consists in defining a set of the previous LPV plants to take into account the unmeasurable uncertainty.

5.4.1 LPV/LFT Plant Modelling

The first step consists in finding a LFT model for the uncertain system. For that purpose, we define:

$$\begin{aligned} \tilde{J}(t) = J(t)^{-1} = \tilde{J}_0 + \delta_{\tilde{J}}(t)\tilde{J}_1,\ \left|\delta_{\tilde{J}}(t)\right| \le 1, \\ \tilde{l}(t) = l(t)^{-1} = \tilde{l}_0 + \delta_{\tilde{l}}(t)\tilde{l}_1,\ \left|\delta_{\tilde{l}}(t)\right| \le 1. \end{aligned} \quad (44)$$

with:

$$\begin{aligned} \tilde{J}_1 &= 0.5\left(J_{\min}{}^{-1} - J_{\max}{}^{-1}\right) \\ \tilde{J}_0 &= 0.5\left(J_{\min}{}^{-1} + J_{\max}{}^{-1}\right) \\ \tilde{l}_1 &= 0.5\left(l_{\min}{}^{-1} - l_{\max}{}^{-1}\right) \\ \tilde{l}_0 &= 0.5\left(l_{\min}{}^{-1} + l_{\max}{}^{-1}\right) \end{aligned} \quad (45)$$

A state-space representation of the uncertain system with state-vector $X(t)=(i(t),\omega(t),x_c(t),\varphi(t),v_l(t))^T$, is easily obtained:

$$\begin{pmatrix} \dot{X} \\ v_\Delta \\ x_c \\ \varphi \end{pmatrix} = \left(\begin{array}{c|ccc} A & B_\delta & B_d & B_u \\ \hline C_\delta & D_{\delta\delta} & D_{\delta d} & D_{\delta u} \\ \hline C_x & D_{x\delta} & D_{xd} & D_{xu} \\ C_\varphi & D_{\varphi\delta} & D_{\varphi d} & D_{\varphi u} \end{array}\right) \begin{pmatrix} X \\ w_\Delta \\ d \\ u \end{pmatrix} \quad (46)$$

$$v_\Delta = \Delta(t) w_\Delta$$

Where the (measurable) uncertainty block is structured as follows:

$$\Delta(t) = diag\left(\delta_{\tilde{J}}(t), \delta_{\tilde{J}}(t), \delta_{\tilde{l}}(t), \delta_{\tilde{l}}(t)\right) \quad (47)$$

5.4.2 Multiple Plants LPV/LFT Modelling

We propose do define a set of 3 LPV/LFT plants based on (46) by considering the minimal, nominal and maximal values of the friction coefficient (48).

$$f = \left\{0\ ; 6,5.10^{-5}\ ; 13.10^{-5}\right\} N.(m.s^{-1})^{-1} \quad (48)$$

5.4.3 Robust LPV/LFT Controller Synthesis

Now we can compute a 3 DOF controller K with (for instance) order 4 and symmetric scalings L_j (j=1,2,3) commuting with Δ. Thus each of them has the following block diagonal structure:

$$L_j = \begin{pmatrix} l_{11}{}^{\{j\}} & l_{12}{}^{\{j\}} & 0 & 0 \\ l_{12}{}^{\{j\}} & l_{22}{}^{\{j\}} & 0 & 0 \\ 0 & 0 & l_{33}{}^{\{j\}} & l_{34}{}^{\{j\}} \\ 0 & 0 & l_{34}{}^{\{j\}} & l_{44}{}^{\{j\}} \end{pmatrix},\ j = 1,2,3 \quad (49)$$

Note that searching for diagonal scalings may lead to too conservative results; but it would be the good choice if uncertainties were all different (that is not repeated) in Δ(t).

After proceeding to 10 runs (20000 iterations per run) on an Intel Core i7-3740 QM – 2.7 GHz processor and Matlab 2016b, best results are given in Table 1. We can see that our approach is effective for a reasonable computing time.

We implement the gain-scheduled controller according to (13) in Simulink and proceed to a temporal simulation using references for $J(t)$ and $l(t)$ defined in Figure 8. As we can see in Figure 10, the cart response is more robust with our LPV/LFT (only order 4) controller than with the LPV (full-order) polytopic controller for different values of the friction coefficient. Our LPV/LFT controller is stabilizing and performing for the different f values whereas the LPV full-order polytopic controller doesn't stabilize the plant for f = 0. This makes our robust fixed-structure LPV/LFT synthesis successful.

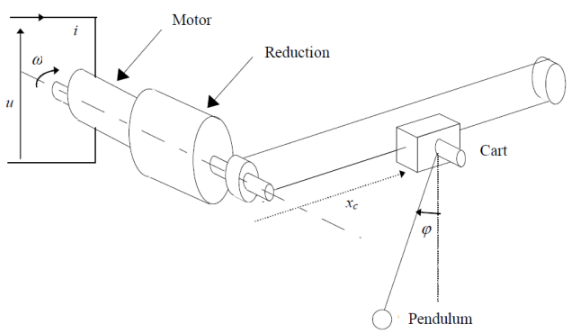

Figure 5: Pendulum in a cart.

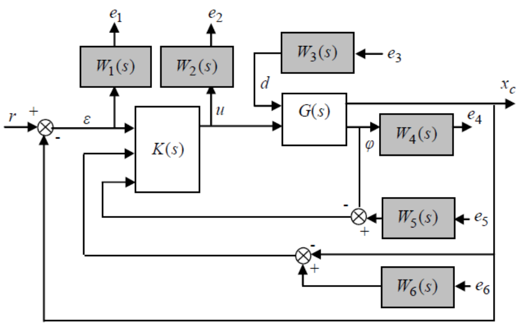

Figure 6: H∞ control scheme.

Table 1: Results for multiple plants LPV/LFT synthesis.

γ_{best}	γ_{mean}	γ_{std}	t_{CPU}/run (min)
3,39	29,82	25,6	257

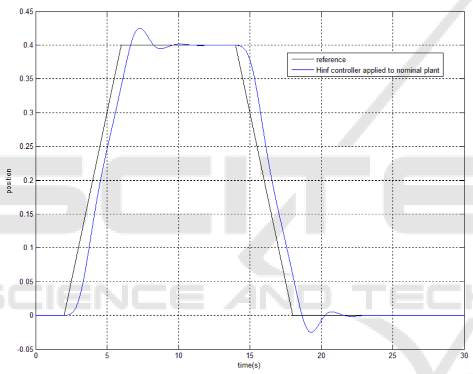

Figure 7: H∞ controller response without uncertainty.

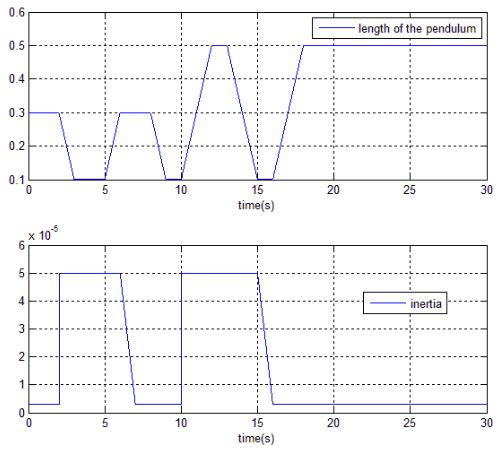

Figure 8: Evolution laws for inertia and length of the pendulum.

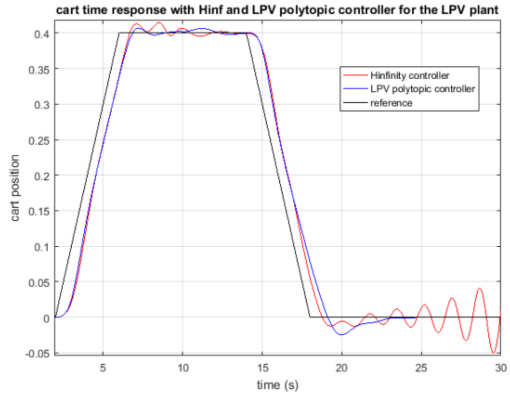

Figure 9: Position reference of the cart and cart responses when considering the LPV plant.

Table 2: Nominal values of parameters.

Symbol	Signification	value	Unit
R	Resistance of the motor	2,3	Ω
L	Inductance of the motor	1.10^{-3}	H
ϕ	Electromagnetic constant	0,0162	U.S.I.
f	Friction coefficient	$6{,}5.10^{-5}$	$N/m.s^{-1}$
r	Radius of the pulley	0,022	m
N	Gear reduction	17	-
α	Friction coefficient of the pendulum	0,3	$m.s^{-1}$
G	Weight acceleration	9,81	$m.s^{-2}$
L	Length of the pendulum	0,275	m
J	Inertia of the motor	5.10^{-6}	$kg.m^2$
k_x	Gain of position sensor	39,77	$V.m^{-1}$
k_φ	Gain of angular sensor	4,77	$V.rad^{-1}$

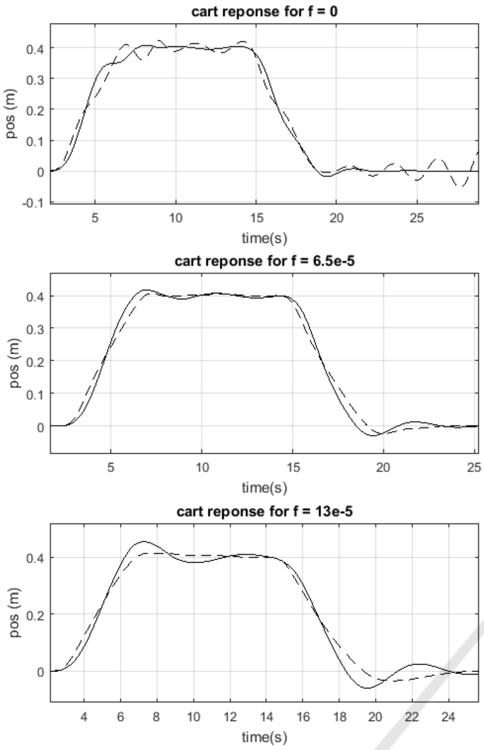

Figure 10: Cart response for different f values: polytopic approach (--) and proposed approach (-).

6 CONCLUSIONS

We developed in this paper a method for computing robust fixed-structure LPV/LFT controllers for plants with time-varying (measurable) and time invariant (unmeasurable) uncertainties. By using the multiple plants framework instead of μ-technics, we achieved to determine a performing and robust LPV controller against the unknown uncertainties using evolutionary computation. Future works deal with the reduction of computation time.

REFERENCES

Moore, R., Lopes, J., 1999. Paper templates. In *TEMPLATE'06, 1st International Conference on Template Production*. SCITEPRESS.

Zhou, K., Doyle, J. C., Glover, K., 1996. *Robust and Optimal Control*. Prentice-Hall.

Apkarian, P., Noll, D., 2002, Nonsmooth H_∞ synthesis. In *Journal of class files,* vol.1, n°11.

Young, P., Doyle, J., 1990, Computation of with real and complex uncertainties. In *Proceedings of the 29th IEEE Conference on Decision and Control*, pp. 1230-1235.

Apkarian, P., 2011. non-smooth μ-synthesis. In: *Int. Journal of Robust and Nonlinear Control*, vol. 21, Issue 13, pp 1493-1508.

Apkarian, P., Biannic, J.M., Gahinet, P., 1993. Gain Scheduled H_∞ Control of a Missile via Linear Matrix Inequalities. In: *Journ. of Guidance Control and Dyn.*, vol. 18, no. 3, pp. 532-538.

Apkarian, P., Gahinet, P., 1995. A Convex Characterization of Gain-Scheduled H_∞ Controllers. In: *IEEE Transations on Automatic Control*, Vol.40, N° 5, pp 853-864.

Shi, Y., Tuan, H. D., Apkarian, P., 2010, Nonconvex Spectral Optimization Algorithms for Reduced-Order H_∞ LPV-LFT controllers, In: *Int. J. Robust. Nonlinear Control*.

Blue, P. A., Banda, S. S., 1997. D-K iteration with optimal scales for systems with time-varying and time invariant uncertainties. In: *Proceedings of the American Control Conference, Albuquerque, New Mexico*.

De Vito, D., Kron, A., De Lafontaine, J., Lovera, M., 2010. A Matlab toolbox for LMI-based analysis and synthesis of LPV/LFT self-scheduled H∞ control systems. In : *2010 IEEE International Symposium on Computer-Aided Control System Design.* pp 1397-1402.

Apkarian. P., 2002. Tuning controllers for multiple models or operating conditions. In: *http://pierre.apkarian.free.fr.*

Storm, R., Price, K., 1995. Differential Evolution – A simple and efficient adaptive scheme for global optimization over continuous spaces. In: *Technical Report TR95 -012, International Computer Science Institute, Berkeley, California.*

Dréo, J., Pétrowski, A., Siarry, P., Taillard, E., 2006. *Metaheuristics for hard optimization*, Springer-Verlag, Berlin.

Clerc, M., 2012. Standard Particle Swarm Optimization, In : *http://clerc.maurice.free.fr/pso.*

Feyel, P., Sandou, G., Duc, G., 2014a. Evolutionnary Fixed-Structure Mu-synthesis. In: *IEEE Symposium Series on Computational Intelligence 2014*, Orlando, Florida, USA.

Feyel, P., Sandou, G., Duc, G., 2014b. Evolutionnary Fixed-H∞ synthesis for multiple plants. In: *2014 IEEE Multi-Conference on Systems and Control, Antibes, France*.