Preprints of the 2012 20th Mediterranean Conference on
Control & Automation (MED), Barcelona, Spain, July 3-6,
2012

FTC Design for Polytopic LPV Systems subject to Actuator Saturations

Damiano Rotondo, Fatiha Nejjari and Vicenç Puig

Abstract— **In this paper, a Fault Tolerant Control (FTC) scheme using virtual actuators for discrete-time polytopic Linear Parameter Varying (LPV) systems subject to actuator saturations is proposed. The plant with the faulty actuator is augmented to take into account the saturations and modified adding the virtual actuator block that masks the fault. The elements of the FTC control loop are designed using Linear Matrix Inequalities (LMIs) in order to achieve the Pole Placement and the H_2 norm specifications. To assess the performance of the proposed approach an aeronautical application is used.**

I. INTRODUCTION

Fault Tolerant Control allows to maintain performance and stability in the presence of faults [1]. Accommodation capability of a control system depends on many factors [2] such as severity of the fault, robustness of the nominal system and mechanisms that introduce redundancy in sensors and/or actuators. From the point of view of the control strategies, the literature considers two main groups of techniques: the *active* and the *passive* (see [3] for a review). The *passive FTC techniques* are control laws that take into account the fault as a system perturbation. Thus, within certain margins, the control law has inherent fault tolerant capabilities, allowing the system to cope with the fault presence. On the other hand, the *active FTC techniques* consist in adapting the control law using some information about the fault. With such information, some automatic adjustments in the control loop are done after the fault appearance so as to satisfy the control objectives with minimum performance degradation.

Recently, virtual actuators for linear systems have been proposed as a fault accommodation approach (see, e.g. [1]). The main idea of this FTC method is to reconfigure the control loop such that the nominal controller could be still used without need of retuning it. The plant with the faulty actuator is modified adding the virtual actuator block that masks the actuator fault and allows the controller to see the same plant as before the fault. This technique, originally proposed for LTI systems, is extended in this work to non-linear systems that can be approximated by an LPV model. The main advantage of LPV models is that they allow applying powerful linear design tools to complex non-linear models [4]. The LPV theory is mainly used for designing controllers for non-faulty systems, but recently it has also been used for active FTC [5].

In all controlled systems, actuator capacity is limited by physical constraints and limitations of the actuator. Results of saturation on the control loop are performance degradation, large overshoot or possible instability in spite of the satisfactory performance predicted from a linear design. While analysis of systems, including saturated actuators, is relatively easy, the controller synthesis problem in presence of input non-linearity is a much more involved task. In [6], a systematic anti-windup control synthesis approach for systems with actuator saturation is provided within an LPV design framework. The advantage of this approach is that it directly utilizes saturation indicator parameters to schedule accordingly the parameter-varying controller.

In this paper, LPV virtual actuators are applied to non-linear systems described by an LPV model so as to give fault tolerance. This work is an extension of [7], where LPV virtual actuators and LPV nominal controller are designed using LMI regions, as proposed in [8], without taking into account the actuator saturations in the design step. Moreover, the closed-loop eigenvalues are put in a desired region of the complex plane, but no optimal criterion is used to choose the exact position of the poles. In this paper, H_2 performance [9] is used to enforce additional conditions in the controller design so as to give some robustness against random disturbances and, at the same time, to assure the overall stability in presence of actuator saturations.

The paper is organized as follows: *Section II* states the control problem and presents the faults and the LPV virtual actuator strategy to be used to provide fault tolerance to the system. *Section III* presents some theory that will be used in this paper for the controller design. The polytopic approximation of an LPV system and the polytopic version of the LMIs that will be used for control design are presented in *Section IV*. In *Section V* the overall FTC scheme design is provided. *Section VI* presents results of the proposed approach in the application example. Finally, in *Section VII*, conclusions and further research paths are presented.

II. LPV CONTROL PROBLEM STATEMENT

A. LPV System and Faults Definition

Let us consider the following discrete-time LPV system in state-space form including actuator saturation non-linearities, faults and H_2 output equation as follows:

$$x_f(k+1) = A(\theta(k))x_f(k) + B_w w(k) + B_f(\gamma_k)\sigma(u(k)) + \Gamma f_u(k) \quad (1)$$

$$y(k) = Cx_f(k) \quad (2)$$

$$z(k) = C_z x_f(k) + D_{zu}\sigma(u(k)) \quad (3)$$

This work has been funded by the Spanish MINECO through the project CYCYT SHERECS (ref. DPI2011-26243), by the European Commission through contract i-Sense (ref. FP7-ICT-2009-6-270428) and by UPC through the grant FPI-UPC E-01104.

Damiano Rotondo, Fatiha Nejjari and Vicenç Puig are with Advanced Control Systems Group (SAC), Universitat Politècnica de Catalunya (UPC), Rambla de Sant Nebridi, 11, 08222 Terrassa, Spain. e-mail: vicenc.puig@upc.edu

978-1-4673-2529-5/12/$31.00 ©2012 IEEE

where $x_f(k) \in \mathbb{R}^{n_x}$ denotes the state vector, $u(k) \in \mathbb{R}^{n_u}$ denotes the control inputs, $w(k) \in \mathbb{R}^{n_w}$ are the exogenous inputs, $y(k) \in \mathbb{R}^{n_y}$ are the measured outputs and $z(k) \in \mathbb{R}^{n_z}$ are the output signals related to the H_2 performance of the control system. All system matrices have the appropriate dimensions. $\sigma(u(k))$ is the input saturation non-linearity. $\theta(k)$ is the system vector of time-varying parameters of dimension n_θ that changes with the operating point. This vector is scheduled by some measured system variables $p(k)$ that can be estimated using some known function $\theta(k) = f(p(k))$, named scheduling function. $f_u(k) \in \mathbb{R}^{n_u}$ denotes the additive actuator faults and $\Gamma \in \mathbb{R}^{n_x \times n_u}$ denotes the fault distribution matrix. The multiplicative actuator faults are embedded in the matrix $B_f(\gamma_k)$ as follows:

$$B_f(\gamma_k) = B_u diag(\gamma_1(k), \gamma_2(k), \ldots, \gamma_{n_u}(k)), \ 0 \leq \gamma_i(k) \leq 1 \quad (4)$$

γ_i represents the effectiveness of the i^{th} actuator, such that the extreme values $\gamma_i = 0$ and $\gamma_i = 1$ represent a total failure of the i^{th} actuator and a non-faulty i^{th} actuator, respectively. B_u is the nominal input matrix.

Hereafter, the assumption that $\Gamma = B_f$ is done as usually done in most of the literature. Cases different from this case could be handled by adding more complexity to the mathematic formulation.

B. Control Strategy and Objectives

The LPV system is controlled by a state feedback control with a tracking reference input as proposed in [10]. The feedback control law can be expressed as follows:

$$u_c(k) = u_r(k) + K(\theta(k))(\hat{x}(k) - x_r(k)) \quad (5)$$

where the state reference $x_r(k)$ and the feedforward control action $u_r(k)$ correspond to an equilibrium point for the reference $r(k)$.

Notice that the estimated state $\hat{x}(k)$ is used because $x(k)$ is assumed not to be available. Consequently, an LPV state observer is used to provide such state estimation.

The LPV controller matrix $K(\theta(k))$ should be designed in order to satisfy the following specifications:

- The system should maintain its stability in case of actuator saturations
- Pole placement in a region of the complex plane,
- Bounds on the H_2 cost,

Moreover, some Fault Tolerant Control strategy is needed so as to assure that the overall system can cope with the fault occurrence.

C. Virtual Actuators FTC Strategy

The concept of virtual actuator introduced in [1] was extended in [7] to non-linear systems that can be approximated by an LPV model. The main idea of this FTC technique is to reconfigure the *faulty plant* such that the nominal controller could still be used without need of retuning. The plant with the faulty actuator is modified adding the virtual actuator block that masks the fault and allows the controller to see the same plant as before the fault.

If the following rank condition is satisfied:

$$rank\,(B_f(\gamma_k)) = rank\begin{pmatrix} B_u & B_f(\gamma_k) \end{pmatrix} \quad (6)$$

the reconfiguration structure can be expressed as:

$$u(k) = N_v(\gamma_k) u_c(k) - f_u(k) \quad (7)$$

where $u_c(k)$ is the controller output and matrix $N_v(\gamma_k)$ is given by:

$$N_v(\gamma_k) = B_f(\gamma_k)^{\dagger} B_u \quad (8)$$

where $B_f(\gamma_k)^{\dagger}$ is the pseudo-inverse of $B_f(\gamma_k)$.

Cases where (6) is not satisfied should be described through values of the matrix B^* such that the following condition holds:

$$B^* = B_f(\gamma_k) N_v(\gamma_k) \quad (9)$$

In such cases, the reconfiguration structure is expressed by:

$$u(k) = N_v(\gamma_k)\left(M_v(\vartheta_k) x_v(k) + u_c(k)\right) - f_u(k) \quad (10)$$

where $M_v(\vartheta_k)$ is the gain of the LPV virtual actuator while the virtual actuator state x_v is calculated as:

$$x_v(k+1) = (A(\vartheta_k) - B^* M_v(\vartheta_k))\, x_v(k) + (B_u - B^*)\, u_c(k) \quad (11)$$

Notice that the matrix B^* does not depend on γ_k because the matrix $N_v(\gamma_k)$ eliminates the effects of partial faults.

When the actuator fault appears, the LPV virtual actuator reconstructs the system input vector $u_f(k)$ from the output of the nominal controller $u_c(k)$, taking into account the fault occurrence. The faulty plant and the LPV virtual actuator are called the reconfigured LPV plant which is connected to the nominal LPV controller. If the reconfigured LPV plant behaves like the nominal plant, the loop consisting of the reconfigured plant and the LPV controller behaves like the nominal closed-loop system.

In order to apply the virtual actuator technique, an estimation of the faults $\hat{\gamma}_k$ and $\hat{f}_u(k)$ is needed. In this paper, the fault estimation is obtained by applying a method based on the Recursive Least Squares with a forgetting factor [11]. This method allows to estimate the multiplicative and additive faults in (1) and starts from the hypothesis that it is possible to find a state of the system that is directly influenced by the faulty actuator. More details about this method can be found in [7].

III. LMI DESIGN

A. LMI Pole Placement Design

In [8], an LMI approach for the design by pole placement constraints is described. The main motivation for seeking pole clustering in specific regions of the complex plane is that, by constraining the eigenvalues λ to lie in a prescribed region, stability can be guaranteed and some desired transient response performances can be ensured. A subset D of the complex plane is called an LMI region if there exist a symmetric matrix $\rho = [\rho_{kl}] \in \mathbb{R}^{m \times m}$ and a matrix $\beta = [\beta_{kl}] \in \mathbb{R}^{m \times m}$ such that

$$D = \{\xi \in \mathbb{C} : f_D(\xi) < 0\} \quad (12)$$

978-1-4673-2529-5/12/$31.00 ©2012 IEEE

where the characteristic function $f_D(\xi)$ is given by:

$$f_D(\xi) = \left[\rho_{kl} + \beta_{kl}\xi + \beta_{kl}\bar{\xi}\right]_{l\leq k,\, l\leq m} \tag{13}$$

$f_D(\xi)$ takes values in the space of $m \times m$ Hermitian matrices. Using Gutman's theorem for LMI regions [12], pole location in a given LMI region can be characterized in terms of the $m \times m$ block matrix

$$\begin{array}{rcl} M_D(A,X) & := & \rho \otimes X + \beta \otimes (AX) + \beta^T \otimes (AX)^T \\ & = & \left[\rho_{kl} X + \beta_{kl} AX + \beta_{lk} XA^T\right]_{1\leqslant k,l\leqslant m} \end{array} \tag{14}$$

According to [8], there exists a state-feedback controller in the form $u(k) = Kx(k)$ such that it places the closed-loop poles in some LMI region D with characteristic function (13) if, and only if, there exists $X_D > 0$ such that $M_D(A + B_uK, X_D) < 0$

By means of the auxiliary variable $L := KX$, this matrix inequality becomes an LMI that can be solved using convex optimization techniques:

$$\begin{cases} \left[\alpha_{kl} X + \beta_{kl} U(X,L) + \beta_{lk} U(X,L)^T\right] < 0 \\ U(X,L) := AX + B_u L \end{cases} \tag{15}$$

B. LMI H_2 Design

The H_2 performance of the closed-loop transfer function T from the input w to the output z (see Eq. (1)-(3)) is useful to handle random disturbances and is defined as [9]:

$$\|T\|_2^2 := \frac{1}{2\pi} \int_{-\pi}^{\pi} Tr\left(T(e^{j\omega})^H T(e^{j\omega})\right) d\omega \tag{16}$$

and corresponds to the asymptotic variance of the output z when the system is driven by white noise w.

According to [13], there exists a state-feedback controller in the form $u(k) = Kx(k)$ such that the inequality $\|T\|_2^2 < \mu$ holds if, and only if, the LMIs:

$$trace(W) < \mu \tag{17}$$

$$\begin{bmatrix} W & C_z X + D_{zu} L \\ (\cdot)^T & X \end{bmatrix} > 0 \tag{18}$$

$$\begin{bmatrix} X & AX + B_u L & B_w \\ (\cdot)^{\mathrm{T}} & X & 0 \\ (\cdot)^{\mathrm{T}} & (\cdot)^{\mathrm{T}} & I \end{bmatrix} > 0 \tag{19}$$

hold, where the symmetric matrices X and W and the matrix $L = KX$ are the variables.

C. Anti-windup LPV Controller Design

The saturation function $\sigma(u)$ specifies the limited actuator capacity on the control input u. The saturation is supposed to be a decoupled, sector-bounded, static actuator non-linearity with a constant saturation limit u_i^{MAX} in the i-th channel, that is:

$$\sigma(u_i) = \begin{cases} u_i & if \quad |u_i| < u_i^{MAX} \\ sign(u_i) u_i^{MAX} & if \quad |u_i| \geq u_i^{MAX} \end{cases} \tag{20}$$

for $i = 1, 2, \ldots, n_u$. Following the idea proposed in [6], the following saturation scheduling parameters are introduced:

$$\varsigma_i(u_i) = \frac{\sigma(u_i)}{u_i} \quad for \quad i = 1, 2, \ldots, n_u \tag{21}$$

and $\varsigma_i(0) = 1$; thus $\varsigma_i(u_i) \in (0,1]$. Hence, the variable ς_i defines the level of saturation of the i-th actuator at each instant of time. The objective is to design parameter varying controllers that are scheduled on the basis of both the operating condition parameter vector θ and the saturation indicator parameter vector ς.

Let us define the following saturation matrix operator:

$$\Sigma = diag(\varsigma_1, \varsigma_2, \ldots, \varsigma_{n_u}) \tag{22}$$

where ς_i is given by (21). Clearly, $\Sigma = I$ represents the situation that all actuators are in their linear regimes. Otherwise, the values on the diagonal elements of Σ will reflect the status of each saturated actuator. Now, equations (1) and (3) can be rewritten in the form:

$$x_f(k+1) = A(\theta(k))x_f(k) + B_w w(k) + \tilde{B}_f(\gamma_k)u(k) + \Gamma f_u(k) \tag{23}$$

$$z(k) = C_z x(k) + \tilde{D}_{zu} u(k) \tag{24}$$

where the matrices $\tilde{B}_f(\gamma_k) = B_f(\gamma_k)\Sigma(\varsigma(k))$ and $\tilde{D}_{zu} = D_{zu}\Sigma(\varsigma(k))$ are parameter-dependent. In this case, as stated by [14], some problems arise in the computation of a solution for some LMIs expressed in a polytopic way, as those used in this paper, because they will need an infinite number of constraints and are therefore not easily tractable.

In [15], it is proposed to filter the control input u so as to solve this difficulty. Specifically, a new control input $\tilde{u}$ is defined and the system (23)-(24) is augmented with:

$$s(k+1) = A_s s(k) + B_s \tilde{u}(k) \tag{25}$$

$$u(k) = C_s s(k) \tag{26}$$

In this way, the overall system takes the form:

$$\begin{bmatrix} x_f(k+1) \\ s(k+1) \end{bmatrix} = \begin{bmatrix} A(\theta(k)) & B_f(\gamma_k)\Sigma(\varsigma(k))C_s \\ 0 & A_s \end{bmatrix} \begin{bmatrix} x_f(k) \\ s(k) \end{bmatrix} + \begin{bmatrix} B_w \\ 0 \end{bmatrix} w(k) + \begin{bmatrix} 0 \\ B_s \end{bmatrix} \tilde{u}(k) + \begin{bmatrix} \Gamma \\ 0 \end{bmatrix} f_u(k) \tag{27}$$

$$z(k) = \begin{bmatrix} C_z & D_{zu}\Sigma(\varsigma(k))C_s \end{bmatrix} \begin{bmatrix} x(k) \\ s(k) \end{bmatrix} \tag{28}$$

IV. POLYTOPIC LMI CONDITIONS

A. Polytopic Approximation

In this work, the kind of LPV systems considered are those whose time-varying parameter vector $\theta(k)$ varies within a polytope Θ. Each polytope vertex corresponds to a particular value of the scheduling variable $\theta(k)$. In other words,

$$A(\theta(k)) \in Co(A_j, j = 1, \ldots, N) := \sum_{j=1}^{N} \alpha_k^j(\theta(k)) A_j \tag{29}$$

with $\alpha_k^j(\theta(k)) \geq 0$ and $\sum_{j=1}^{N} \alpha_k^j(\theta(k)) = 1$. Because of this property, this type of LPV systems is referred to as polytopic [15]. A common approach to design a controller and an observer for an LPV system is to approximate it by a polytopic LPV system [15]. The simplest polytopic approximation relies on bounding each LPV parameter by an interval (*bounding box* approach). Such a box can be found by considering every possible combination of minima and maxima of the elements of $\theta(k)$ over the allowed range of variation of the measured system variables $p(k)$.

B. Polytopic Pole Placement Conditions

The LMI pole placement condition (15) can be easily extended to systems described by a polytopic state-space model. The problem consists in computing state-feedback gains K_j, one for each vertex of the polytope (29), and a single Lyapunov matrix $X > 0$ such that $M_D(A_j + B_u K_j, X) < 0$ for $j = 1, \ldots, N$. The use of a single Lyapunov matrix over the entire operating range guarantees D-stability for all state-space matrices in the polytope (29). The LMIs (15) are particularized for an LMI region consisting of the intersection between the circle of radius r and center $(-q, 0)$ and the vertical strip defined by the extreme values S_{min} and S_{max}:

$$\left\{ \begin{array}{l} diag(Q_1, Q_2, Q_3) < 0 \\ X > 0 \end{array} \right. \tag{30}$$

$$Q_1 = \left(\begin{array}{cc} -rX & qX + A_j X + B_u L_j \\ qX + XA_j^T + L_j^T B_u^T & -rX \end{array} \right) \tag{31}$$

$$Q_2 = S_{min} X - \frac{1}{2} \left(A_j X + XA_j^T + B_u L_j + L_j^T B_u^T \right) \tag{32}$$

$$Q_3 = -S_{max} X + \frac{1}{2} \left(A_j X + XA_j^T + B_u L_j + L_j^T B_u^T \right) \tag{33}$$

Once these LMIs are solved, the gains K_j can be determined by $K_j = L_j X^{-1}$.

C. Polytopic H_2 Conditions

The H_2 norm conditions (17)-(19) can be extended to polytopic LPV systems. In this case, the problem consists in computing state-feedback gains K_j, one for each vertex of the polytope (29), and symmetric matrices $X > 0$ and W such that:

$$trace(W_j) < \mu \tag{34}$$

$$\left[\begin{array}{cc} W_j & C_{zj} X + D_{zu} L_j \\ (\cdot)^T & X \end{array} \right] > 0 \tag{35}$$

$$\left[\begin{array}{ccc} X & A_j X + B_u L_j & B_w \\ (\cdot)^{\mathrm{T}} & X & 0 \\ (\cdot)^{\mathrm{T}} & (\cdot)^{\mathrm{T}} & I \end{array} \right] > 0 \tag{36}$$

Once these LMIs are solved, the gains K_j can be determined by $K_j = L_j X^{-1}$.

Notice that in (35) the matrix C_{zj}, that is the output matrix of (3), is allowed to vary in a polytopic way. This does not results in computation problems as those described in [14], as C_{zj} only appears multiplied by the unique matrix X and such a relaxation is needed in order to deal with the design of a H_2 controller for the augmented system (27)-(28).

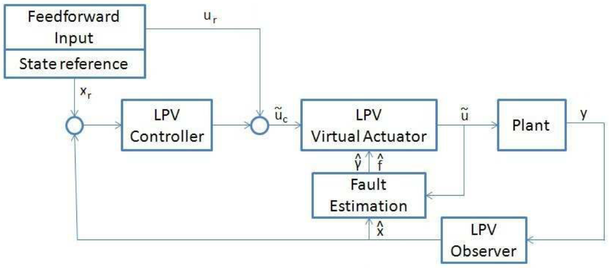

Fig. 1. FTC Scheme using Virtual Actuators

V. OVERALL FTC SCHEME DESIGN

A. The overall FTC system

The overall FTC scheme results as in Fig. 1.

In (27)-(28), the filter matrices are chosen as $B_s = I - A_s$ and $C_s = I_s$, where I_s is the identity matrix of order n_u. This selection assures that $u_\infty = \tilde{u}_\infty$ at steady state and the results obtained in the case without saturation are valid for the case with saturation.

The nominal augmented system is controlled with a law as (5):

$$\tilde{u}_c(k) = u_r(k) + K(\theta(k), \varsigma(k)) \left(\left[\begin{array}{c} \hat{x}(k) - x_r(k) \\ 0 \end{array} \right] \right) \tag{37}$$

where the estimated state $\hat{x}(k)$ is provided by an LPV state observer as in [7]. Notice that there is no need to consider the augmented system for state estimation purposes, as the value of $s(k)$ in (25) is always available. Lastly, the virtual actuator reconfiguration structure is obtained from (10)-(11), by applying these equations to the augmented system:

$$\tilde{u}(k) = N_v(\gamma_k) \left(M_v(\theta(k)) x_v(k) + \tilde{u}_c(k) \right) - f_u(k) \tag{38}$$

$$x_v(k+1) = \left(A(\theta(k)) - B^* M_v(\theta(k)) \right) x_v(k) + (B - B^*)\, \tilde{u}_c(k) \tag{39}$$

B. Polytopic LPV Equations

Using the *bounding box* approximation, equations (27), (28), (37) and (39) can be expressed in a polytopic way as follows:

$$\begin{aligned} \left[\begin{array}{c} x_f(k+1) \\ s(k+1) \end{array} \right] = \sum_{j=1}^{N} \alpha_k^j(\theta(k), \varsigma(k)) \left[\begin{array}{cc} A_j & B_f \Sigma_j \\ 0 & A_s \end{array} \right] \left[\begin{array}{c} x_f(k) \\ s(k) \end{array} \right] \\ + \left[\begin{array}{c} B_w \\ 0 \end{array} \right] w(k) + \left[\begin{array}{c} 0 \\ I - A_s \end{array} \right] \tilde{u}(k) + \left[\begin{array}{c} \Gamma \\ 0 \end{array} \right] f_u(k) \end{aligned} \tag{40}$$

$$z(k) = \sum_{j=1}^{N} \alpha_k^j(\theta(k), \varsigma(k)) \left[\begin{array}{cc} C_{zj} & D_{zu} \Sigma_j \end{array} \right] \left[\begin{array}{c} x(k) \\ s(k) \end{array} \right] \tag{41}$$

$$\tilde{u}_c(k) = u_r(k) + \sum_{j=1}^{N} \alpha_k^j(\theta(k), \varsigma(k)) K_j \left[\begin{array}{c} \hat{x}(k) - x_r(k) \\ 0 \end{array} \right] \tag{42}$$

$$\tilde{x}_v(k) = \sum_{j=1}^{N} \alpha_k^j(\theta(k), \varsigma(k)) (A_j - B^* M_j) x_v(k) + (B - B^*) \tilde{u}_c \tag{43}$$

The polytopic equations are scheduled through non-negative $\alpha_k^j(\theta(k),\varsigma(k)), j = 1,\dots,N$ such that $\sum_{j=1}^{N}\alpha_k^j(\theta(k),\varsigma(k)) = 1$. There are several ways to calculate α_k^j. Here, they are calculated via barycentric combination of vertices as suggested by [15].

C. Polytopic LMI-based Design

The main objective of this subsection is to summarize the design procedure of the *LPV Controller* and *LPV Virtual Actuator*. The design implies obtaining:

- matrices K_j of (42) in order to guarantee closed-loop stability of the original system assuming that the pair $(A_{aug}(\theta(k)), B_{aug})$ is stabilizable for all $\theta(k) \in \Theta$, where A_{aug} and B_{aug} are the matrices of the augmented system,
- matrices M_j of (43) in order to guarantee the stability of the LPV virtual actuator assuming that the pair $(A(\theta(k)), B^*)$ is stabilizable for all $\theta(k) \in \Theta$. This problem must be solved for each possible B^*.

Under these assumptions, it is possible to design the matrices K_j and M_j using LMI techniques.

The *polytopic LPV controller* is designed by finding matrices L_j and W_j and a single matrix $X > 0$, such that (30)-(36) hold. Then, the gains K_j can be determined by $K_j = L_j X^{-1}$.

Defining an H_2 norm equation for the virtual actuator, a similar problem can be solved to obtain the gains M_j, by replacing the matrices in a proper way.

VI. APPLICATION EXAMPLE

A. Description of Twin-Rotor MIMO System

The Twin-Rotor MIMO System (TRMS) is a laboratory setup developed by Feedback Instruments Limited for control experiments. The system is perceived as a challenging engineering problem due to its high non-linearity, cross-coupling between its two axes, and inaccessibility of some of its states through measurements. The TRMS mechanical unit has two rotors (the main and the tail, both driven by DC motors) placed on a beam together with a counterbalance whose arm with a weight at its end is fixed to the beam at the pivot and it determines a stable equilibrium position. The beam can rotate freely both in the horizontal and vertical planes. The system input vector is $u = [u_h,\ u_v]^T$ where u_h is the input voltage of the tail motor and u_v is the input voltage of the main motor. The system states vector is $x = [\omega_h,\ \Omega_h,\ \theta_h,\ \omega_v,\ \Omega_v,\ \theta_v - \theta_v^0]^T$ where $\omega_{h/v}$ is the rotational velocity of the tail/main rotor, $\Omega_{h/v}$ is the angular velocity around the horizontal/vertical axis, $\theta_{h/v}$ is the yaw/pitch angle of beam and θ_v^0 is the value of the pitch angle corresponding to the equilibrium position with $u_h = 0$ and $u_v = 0$.

B. The TRMS LPV model

The mathematical model of the TRMS is given by a set of non-linear differential equations that can be found in [16]. This model has been validated with the equipment of the TRMS and, following the approach described in [17], it has been reshaped and brought to an LPV representation in [18]:

$$\begin{bmatrix} \omega_h(k+1) \\ \Omega_h(k+1) \\ \theta_h(k+1) \\ \omega_v(k+1) \\ \Omega_v(k+1) \\ \theta_v(k+1) - \theta_v^0 \end{bmatrix} = A(\vartheta_k) \begin{bmatrix} \omega_h(k) \\ \Omega_h(k) \\ \theta_h(k) \\ \omega_v(k) \\ \Omega_v(k) \\ \theta_v(k) - \theta_v^0 \end{bmatrix} + B(p_k) \begin{bmatrix} u_h(k) \\ u_v(k) \end{bmatrix} \tag{44}$$

where

$$A(\vartheta_k) = \begin{bmatrix} a_{11}(p_k) & 0 & 0 & 0 & 0 & 0 \\ a_{21}(p_k) & a_{22}(p_k) & a_{23}(p_k) & a_{24}(p_k) & a_{25}(p_k) & a_{26}(p_k) \\ 0 & T_s & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{44}(p_k) & 0 & 0 \\ a_{51}(p_k) & a_{52}(p_k) & 0 & a_{54}(p_k) & a_{55} & a_{56}(p_k) \\ 0 & 0 & 0 & 0 & T_s & 1 \end{bmatrix}$$

$$B(p_k) = \begin{bmatrix} b_{11} & 0 \\ 0 & b_{22}(p_k) \\ 0 & 0 \\ 0 & b_{44} \\ b_{51} & 0 \\ 0 & 0 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

with $\vartheta_k = [\vartheta_k^1,\ \dots\vartheta_k^{12}]^T = [a_{11}(p_k),\ a_{21}(p_k),\ a_{22}(p_k),$ $a_{23}(p_k), a_{24}(p_k), a_{25}(p_k), a_{26}(p_k), a_{44}(p_k), a_{51}(p_k), a_{52}(p_k),$ $a_{54}(p_k),\ a_{56}(p_k)]^T$.

The model is scheduled with $p_k = [\hat{\omega}_h(k),\ \hat{\Omega}_h(k),\ \hat{\theta}_h(k),$ $\hat{\omega}_v(k),\ \hat{\theta}_v(k)]^T$, using the available state estimations provided by the state observer. Since the scheduling vector p_k contains some state variables of the system, the model is said to be *quasi-LPV*[1]. To implement the control and virtual actuator approach for the TRMS, it is necessary to obtain a polytopic representation of the system. This is done by means of the *bounding box approach*.

C. Design specifications

The *LPV controller* is designed so as to place the closed-loop poles of the nominal augmented system in the right half of the semicircle described by the parameter $q = -0.8$, $r = 0.2$, $S_{min} = 0.8$ and $S_{max} = 1$.

The filter (25) is chosen to be deadbeat, that is:

$$A_s = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad B_s = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{45}$$

In (1), the exogenous input matrix B_w is chosen as:

$$B_w = \begin{pmatrix} 0.001 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.001 & 0 & 0 \end{pmatrix}^T \tag{46}$$

Notice that the input matrix B_u is:

$$B_u = \begin{pmatrix} 0.0163 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.009 & 0 & 0.0038 & 0 & 0 \end{pmatrix}^T \tag{47}$$

so the exogenous input matrix has been chosen in such a way that the exogenous inputs affect the system less than the control inputs, but with a similar order of magnitude ($b_u^{11} \cong 16 b_w^{11}$ and $b_u^{42} \cong 4 b_w^{42}$).

The H_2 performance bound μ has been initially set to a high value ($\mu = 1$) and has been successively reduced to

---

[1] Notice that b_{22} in (44) depends on p_k. However, it has been seen that its variations are small enough to approximate it by its mean value. Then, $B(p_k)$=B.

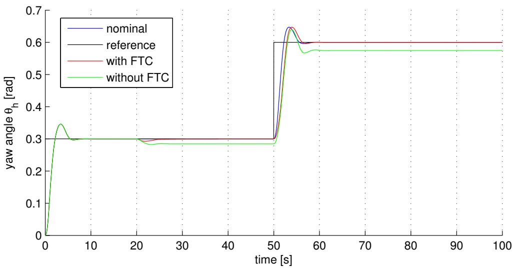

Fig. 2. TRMS yaw angle θ_h (simulation).

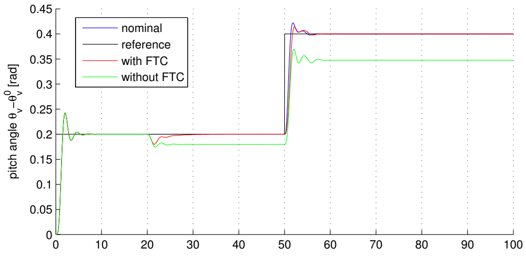

Fig. 3. TRMS pitch angle $\theta_v - \theta_v^0$ (simulation).

lower values trying to maintain the feasibility of the LMIs used for the design. A solution to the LMIs has been found by using the YALMIP toolbox [19].

Due to the lack of actuator redundancy, the system loses its controllability in the event of total loss of an actuator. The fault tolerance can be achieved only with respect to partial faults. In this case, the rank condition (6) is satisfied and the virtual actuators reduce to a static block. Hence, there is no need to design the dynamical part of the virtual actuator.

D. Results

The fault scenario presents the closed-loop behavior when the following change of the set-point is introduced:

$$\theta_h^{ref} = \begin{cases} 0.3, & t < 50s \\ 0.6, & t \geq 50s \end{cases}, \theta_v^{ref} - \theta_v^0 = \begin{cases} 0.2, & t < 50s \\ 0.4, & t \geq 50s \end{cases} \quad (48)$$

The actuator behavior in the fault scenario is defined as:

$$[\gamma_h, \gamma_v] = \begin{cases} [1,1], & \text{for } t < 20s \\ [0.5,0.5], & \text{for } t \geq 20s \end{cases} \quad (49)$$

The bound on the H_2 performance was reduced up to $\mu = 0.1$. Fig. 2 and Fig. 3 present the response of the system both when the FTC strategy acts (red line) and when such a strategy is not implemented (green line). These responses are compared with the response of the system in no-fault situation (blue line) and with the angle references (black line). In both cases, the response in fault conditions is degraded and the system is not able to follow the angle reference. However, the addition of the FTC strategy results in an improvement of the performances of the control system, that at steady-state behaves as in the non-faulty conditions.

VII. CONCLUSIONS

This paper has proposed an FTC strategy using *LPV Virtual Actuators* for non-linear systems subject to actuator saturations that can be approximated by an LPV model. The overall loop consists of the nominal LPV controller, an LPV state observer and the LPV virtual actuator. These are designed using polytopic LPV techniques to achieve pole placement in LMI regions and a H_2 performance, applied to an augmented system so as to take into account the saturation non-linearity in the actuators.

The potential and performances of the proposed approach have been demonstrated in an application to a two degree of freedom helicopter, and the results have been presented in a simulated environment.

REFERENCES

[1] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*. Springer-Verlag Berlin Heidelberg, 2006.

[2] M. R. D. Theilliol and D. Sauter, “Fault tolerant control design for switched systems,” in *2nd IFAC Conference on Analysis and Design of Hybrid Systems*, 2006.

[3] Y. Zhang and J. Jiang, “Bibliographical review on reconfigurable fault-tolerant control systems,” *Annual Reviews in Control*, vol. 32, no. 2, pp. 229–252, December 2008.

[4] J. Shamma and J. Cloutier, “Gain Scheduled Missile Autopilot Design using Linear Parameter Varying Transformations,” *AIAA Journal of Guidance, Control, and Dynamics*, vol. 16, no. 2, pp. 256–263, 1993.

[5] M. Rodrigues, D. Theilliol, S. Aberkane, and D. Sauter, “Fault Tolerant Control Design for Polytopic LPV Systems,” *International Journal of Applied Mathematics and Computer Science*, vol. 17, no. 1, pp. 27–37, 2007.

[6] F. Wu, K. Grigoriadis, and A. Packard, “Anti-windup controller design using linear parameter-varying control methods,” *International Journal of Control*, vol. 73, no. 12, pp. 1104–1114, 2000.

[7] D. Rotondo, F. Nejjari, and V. Puig, “Fault Estimation and Virtual Actuator FTC Approach for LPV Systems,” in *8th IFAC Symposium SAFEPROCESS, Mexico City, Mexico*, 2012.

[8] M. Chilali and P. Gahinet, “H_∞ Design with Pole Placement Constraints: An LMI Approach,” *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 358–367, 1996.

[9] C. W. Scherer, P. Gahinet, and M. Chilali, “Multi-objective output feedback control via LMI optimization,” *IEEE Transactions on Automatic Control*, vol. 42, pp. 896–911, 1997.

[10] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*, 3rd ed. Addison Wesley Longman, 1997.

[11] L. Ljung, *System Identification: Theory for the User*. Prentice Hall Information and System Sciences Series, 1987.

[12] E. I. J. S. Gutman, “A general theory for matrix root clustering in subregions of the complex plane,” *IEEE Transactions on Automatic Control*, vol. AC-26, pp. 853–863, 1981.

[13] M. D. Oliveira, J. Geromel, and J. Bernussou, “Extended H2 and Hinf norm characterizations and controller parametrizations for discrete-time systems,” *International Journal of Control*, vol. 75, no. 9, pp. 666–679, 2002.

[14] G. Becker, A. Packard, D. Philbrick, and G. Balas, “Control of parametrically-dependent linear systems: a single quadratic Lyapunov approach,” in *Proceedings of the American Control Conference, San Francisco, CA, USA*, 1993, pp. 2795–2799.

[15] P. Apkarian, P. Gahinet, and G. Becker, “Self-scheduled H_∞ Control of Linear Parameter-Varying Systems: A Design Example,” *Automatica*, vol. 31, no. 9, pp. 1251 – 1261, 1995.

[16] A. Rahideh and M. H. Shaheed, “Mathematical Dynamic Modelling of a Twin-Rotor Multiple Input-Multiple Output System,” *IMechE Journal of Systems and Control Engineering*, vol. 221, no. 1, pp. 89–101, 2007.

[17] A. Kwiatkowski, M. T. Boll, and H. Werner, “Automated Generation and Assessment of Affine LPV Models,” *Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA, USA*, pp. 6690–6695, 2006.

[18] F. Nejjari, D. Rotondo, V. Puig, and M. Innocenti, “LPV Modelling and Control of a Twin Rotor MIMO System ,” in *19th IEEE Mediterranean Conference on Control and Automation*, 2011.

[19] J. Löfberg, “YALMIP : A Toolbox for Modeling and Optimization in MATLAB,” in *Proceedings of the CACSD Conference, Taipei, Taiwan*, 2004.