An LPV Approach to Active FTC of a Two-Link Manipulator

Ron J Patton, and Supat Klinkhieo

Department of Engineering, University of Hull, HU6 7RX, UK
(Tel: +44 (0)1482-46-5117; e-mail: r.j.patton@hull.ac.uk)

Abstract: This work is motivated by the challenge to develop an adaptive strategy for systems that are complex, have actuator faults and are difficult to control using linear methods. The novelty lies in combined use of LPV fault estimation and LPV fault compensation to meet active FTC performance requirements. The paper proposes a new design approach for systems which can be characterized via sets of LMIs and can be obtained using efficient interior-point algorithms. A polytopic LPV estimator is synthesized for generating actuator fault estimates used in an active FTC scheme to schedule the nominal system state feedback gain as a function of *fault effect factors*, thereby maintaining the system performance over a wide operating range within a proposed polytopic model. The method is demonstrated through a nonlinear two-link manipulator system with torque input faults at each joint.

Keywords: LPV Systems, Fault Tolerant Control, Fault estimation, Active FTC, Robotic Control

1. INTRODUCTION

There is a significant interest in the control of time-varying systems ([1] and [2]). LPV modelling methods have gained a great deal of interest, especially for applications related to vehicle, robust and aerospace control ([3] and [4]). The LPV approach is appealing when nonlinear plants can be modelled as time-varying systems with on-line measurable (or estimated) state-dependent parameters.

Bokor and Balas (2004) ([5]) introduced the concept of the use of fault detection filters for LPV systems and many other investigators have followed different aspects of this approach ([6], [7], [8] and [9]). Recently, the idea of extending the control approach using LPV to encompass Fault-tolerant Control (FTC) schemes has been the subject of a number of studies ([10], [11], [12], [13] and [14]).

Most FTC studies are based on LPV focus on *active* approaches and others on the passive fault tolerance. Active methods use control system adaption or reconfiguration (or both) subject to detectable faults, whilst passive FTC has no provision for active*ly reacting to a fault once it occurs* [15].

Ganguli, Marcos and Balas (2002) ([4]) use LPV ideas for the active FTC problem based on actuator faults in aircraft.

This paper proposes a new design of an active FTC and polytopic LPV estimator for systems which can be characterized via a set of LMIs and can be obtained using efficient interior-point algorithms ([17]). A polytopic LPV estimator is synthesized for providing actuator fault estimation which is used in an FTC scheme to schedule the state feedback gain. The gain is calculated using LMIs in the fault-free case in order to maintain the system performance over a wide operating range within a proposed polytopic model. The active FTC controller is a function of the fault effect factors as defined by [15] and [16] which can be derived on-line (in this case) from the residual vector of a polytopic LPV estimator mechanism. This work uses results from [17] and has mainly been motivated by:

(a) The work of Weng *et al* (2008) ([14]) on LPV fault estimation for rate bounded time-delay systems.

(b) The use of fault effect factors as described by Chen *et al* (1999) ([15]) and Chen and Patton (2001) ([16]).

The work [14] is limited only to fault estimation and does not include the full FTC problem, whilst the work of [15] and [16] pre-dates the development of the LPV approach to control and FTC in particular. The new contribution is the combined use of fault estimation and fault compensation for FTC within an LPV framework. The proposed method is demonstrated through a nonlinear two-link manipulator system with a fault in the torque inputs at each manipulator joint. The system can be represented by a polytopic model.

Section 2 overviews the LPV concept. Section 3 gives a statement of the mathematical problem to be solved. Section 4 details the polytopic LPV estimator design strategy that is to be used in the active FTC scheme. Section 5 describes the polytopic model structure of the two-link manipulator as a tutorial example, based on the LPV estimator theory. This example (with fault estimation) is used for active FTC design, via the polytopic LPV controller synthesis in Section 6. Section 7 gives concluding comments.

2. OVERVIEW OF LPV APPROACH

An LPV system is a mathematical description of the linear parameter-varying nature of a nonlinear system. LPV systems have state-space matrices that are fixed with some vector of varying parameters ([1] and [3]). From a practical point of view, a nonlinear system can be reduced to an LPV representation by using the linearization along trajectories of the parameters. In other words, the idea in LPV is to obtain smooth semi–linear models that can vary or be scheduled using a parameter, for example an altitude and/or speed of an aircraft, so that the LPV model will mimic the actual nonlinear plant ([14], [18] and [19]). Instead of choosing a combination of predefined linear models, the models change parametrically. The LPV model has the structure of a time-varying linear system with the parameter-dependent matrix quadruple $[A(\theta), B(\theta), C(\theta), D(\theta)]$,

where: $A(\theta)\in\Re^{nxn}$, $B(\theta)\in\Re^{nxm}$, $C(\theta)\in\Re^{pxn}$ and $D\in\Re^{pxm}$ as:

$$\dot{x}(t) = A(\theta)x(t) + B(\theta)u(t)$$
$$y(t) = C(\theta)x(t) + D(\theta)u(t) \tag{1}$$

θ is a vector of *smoothly* changing system parameters. An LPV system can also be reduced to a *Linear Time-Varying* (LTV) system with a given parameter trajectory and it can be reformulated as a *Linear Time-Invariant* (LTI) system with a given a constant trajectory [i.e. θ is a constant]. The LPV control is related to gain-scheduling ([1] and [17]) and motivated by the problem of designing multiple models but LPV controllers are set against the lack of performance and stability proofs for classical gain-scheduling ([2] and [4]). LPV controllers are dependent on system parametric changes and are not designed for all linearization points ([1] and [3]).

3. PROBLEM STATEMENT

Consider the LPV state space system described as follows:

$$\begin{aligned}\dot{x}_p(t) &= A_p(\theta)x_p(t) + B_p(\theta)u(t) + E_p(\theta)d(t) + F_p(\theta)f(t)\\ y_p(t) &= C_p(\theta)x_p(t) + D_p(\theta)u(t) + G_p(\theta)d(t) + H_p(\theta)f(t)\end{aligned} \tag{2}$$

$x_p(t) \in \Re^n$, $u(t) \in \Re^p$, $y_p(t) \in \Re^m$, and $d(t) \in \Re^q$ are the states, control inputs, outputs, and disturbances. $f(t) \in \Re^g$ is the fault vector where each element $i = 1, 2, \ldots, g$ corresponds to a specific fault. $\theta \in \Re^s$ is a time-varying parameter vector, and $A_p(\theta)$, $B_p(\theta)$, $C_p(\theta)$, $D_p(\theta)$, $E_p(\theta)$, $F_p(\theta)$, $G_p(\theta)$ and $H_p(\theta)$ are the matrices with appropriate dimensions. Assumptions applicable to (2) are ([17]):

- **(A.1)** The system (2) is stable.
- **(A.2)** The vector $\theta(t)$ varies in a polytope Θ with vertices $\theta_1, \theta_2, \ldots, \theta_r$ ($r = 2^s$), i.e.:

$$\begin{aligned}\theta(t) \in \Theta &:= Co\{\theta_1, \theta_2, \cdots, \theta_r\}\\ &= \left\{\sum_{i=1}^{r} \alpha_p^i \theta_i : \ \alpha_p^i \geq 0, \ \ \sum_{i=1}^{r} \alpha_o^i = 1\right\}\end{aligned} \tag{3}$$

- **(A.3)** The state-space matrices depend *affinely* on $\theta(t)$. System (2) is stable and assumed polytopic, i.e.:

$$\begin{pmatrix} A_p(\theta) & B_p(\theta) & E_p(\theta) & F_p(\theta)\\ C_p(\theta) & D_p(\theta) & G_p(\theta) & H_p(\theta)\end{pmatrix} \in Co\left\{\begin{pmatrix} A_p(\theta_i) & B_p(\theta_i) & E_p(\theta_i) & F_p(\theta_i)\\ C_p(\theta_i) & D_p(\theta_i) & G_p(\theta_i) & H_p(\theta_i)\end{pmatrix}, \ i = 1, \cdots, r\right\} \tag{4}$$

- **(A.4)** $C_p(\theta), D_p(\theta), G_p(\theta)$, and $H_p(\theta)$ are parameter independent, i.e.

$$C_p(\theta_i) = C_p, \quad D_p(\theta_i) = D_p,$$
$$G_p(\theta_i) = G_p, \quad H_p(\theta_i) = H_p, \quad i = 1, \ldots, r$$

4. THE POLYTOPIC LPV ESTIMATOR

We propose a structure fitting the objective of finding an estimator in order that the L_2-induced norm of the operator mapping $[u(t), d(t), f(t)]$ into the estimation error $e_f(t)$ is bounded by a scalar number γ, $\forall$ for all parameter trajectories. The LPV estimator design is given by ([14]):

$$\begin{aligned}\dot{x}_f(t) &= A_f(\theta)x_f(t) + B_f(\theta)\begin{pmatrix} u(t)\\ y_p(t)\end{pmatrix}\\ \hat{f}(t) &= C_f(\theta)x_f(t) + D_f(\theta)\begin{pmatrix} u(t)\\ y_p(t)\end{pmatrix}\end{aligned} \tag{5}$$

$u(t)$, $y_p(t)$ are defined by (1). $x_f(t) \in R^n$ $\hat{f}(t)$ is the estimate of fault $f(t)$. $A_f(\theta), B_f(\theta), C_f(\theta)$, and $D_f(\theta)$ are design matrices with appropriate dimensions. (5) is rewritten as:

$$\mathcal{F}(\theta) := \begin{pmatrix} A_f(\theta) & B_f(\theta)\\ C_f(\theta) & D_f(\theta)\end{pmatrix} \in Co\left\{\mathcal{F}_i = \begin{pmatrix} A_f(\theta_i) & B_f(\theta_i)\\ C_f(\theta_i) & D_f(\theta_i)\end{pmatrix}, \right\} \ i = 1, \cdots, r \tag{6}$$

The following is obtained by combining (2) with (5) and (6):

$$\begin{aligned}
\underbrace{\begin{bmatrix}\dot{x}_p(t)\\ \dot{x}_f(t)\end{bmatrix}}_{\dot{x}_{pf}(t)} &= \begin{bmatrix} A_p(\theta)x_p(t) + B_p(\theta)u(t) + E_p(\theta)d(t) + F_p(\theta)f(t)\\ A_f(\theta)x_f(t) + B_f(\theta)\begin{pmatrix} u(t)\\ C_p(\theta)x_p(t) + D_p(\theta)u(t) + G_p(\theta)d(t) + H_p(\theta)f(t)\end{pmatrix}\end{bmatrix}\\
&= \begin{bmatrix} A_p(\theta)x_p(t)\\ A_f(\theta)x_f(t) + B_f(\theta)\begin{pmatrix} 0\\ C_p(\theta)x_p(t)\end{pmatrix}\end{bmatrix} = \underbrace{\begin{bmatrix} A_p(\theta) & 0\\ B_f(\theta)\begin{pmatrix} 0\\ C_p(\theta)\end{pmatrix} & A_f(\theta)\end{bmatrix}}_{\overline{A}(\theta)}\underbrace{\begin{bmatrix} x_p(t)\\ x_f(t)\end{bmatrix}}_{x_{pf}(t)}\\
&\quad + \begin{bmatrix} B_p(\theta)u(t) + E_p(\theta)d(t) + F_p(\theta)f(t)\\ B_f(\theta)\begin{pmatrix} u(t)\\ D_p(\theta)u(t) + G_p(\theta)d(t) + H_p(\theta)f(t)\end{pmatrix}\end{bmatrix}\\
&= \underbrace{\begin{bmatrix} A_p(\theta) & 0\\ B_f(\theta)\begin{pmatrix} 0\\ C_p(\theta)\end{pmatrix} & A_f(\theta)\end{bmatrix}}_{\overline{A}(\theta)}\underbrace{\begin{bmatrix} x_p(t)\\ x_f(t)\end{bmatrix}}_{x_{pf}(t)}\\
&\quad + \underbrace{\begin{bmatrix}\begin{pmatrix} B_p(\theta) & E_p(\theta) & F_p(\theta)\end{pmatrix}\\ B_f(\theta)\begin{pmatrix} I & 0 & 0\\ D_p(\theta) & G_p(\theta) & H_p(\theta)\end{pmatrix}\end{bmatrix}}_{\overline{B}(\theta)}\underbrace{\begin{bmatrix} u(t)\\ d(t)\\ f(t)\end{bmatrix}}_{w_{udf}(t)}\\
&= \overline{A}(\theta)x_{pf}(t) + \overline{B}(\theta)w_{udf}(t)
\end{aligned} \tag{7}$$

$$\begin{aligned}
e_f(t) &= \hat{f}(t) - f(t)\\
&= C_f(\theta)x_f(t) + D_f(\theta)\begin{bmatrix} u(t)\\ y_p(t)\end{bmatrix} - f(t)\\
&= C_f(\theta)x_f(t) + D_f(\theta)\begin{bmatrix} 0\\ C_p(\theta)x_p(t)\end{bmatrix}\\
&\quad + \begin{bmatrix} D_f(\theta)u(t) - f(t)\\ D_f(\theta)D_p(\theta)u(t) + D_f(\theta)G_p(\theta)d(t) + D_f(\theta)H_p(\theta)f(t) - f(t)\end{bmatrix}\\
&= \underbrace{\begin{bmatrix} D_f(\theta)\begin{pmatrix} 0\\ C_p(\theta)\end{pmatrix} & C_f(\theta)\end{bmatrix}}_{\overline{C}(\theta)}\underbrace{\begin{bmatrix} x_p(t)\\ x_f(t)\end{bmatrix}}_{x_{pf}(t)}\\
&\quad + \underbrace{\begin{bmatrix} D_f(\theta) & 0 & -I\\ D_f(\theta)D_p(\theta) & D_f(\theta)G_p(\theta) & D_f(\theta)H_p(\theta) - I\end{bmatrix}}_{\overline{D}(\theta)}\underbrace{\begin{bmatrix} u(t)\\ d(t)\\ f(t)\end{bmatrix}}_{w_{udf}(t)}\\
&= \overline{C}(\theta)x_{pf}(t) + \overline{D}(\theta)w_{udf}(t)
\end{aligned} \tag{8}$$

$$\bar{A}(\theta)=\begin{bmatrix} A_p(\theta) & 0 \\ B_f(\theta)\underbrace{\begin{pmatrix}0\\ C_p(\theta)\end{pmatrix}}_{C_2(\theta)} & A_f(\theta)\end{bmatrix}$$
$$=\underbrace{\begin{bmatrix} A_p(\theta) & 0 \\ 0 & 0\end{bmatrix}}_{A_0(\theta)}+\underbrace{\begin{bmatrix} 0 & 0 \\ I & 0\end{bmatrix}}_{\mathcal{B}}\underbrace{\begin{bmatrix} A_f(\theta) & B_f(\theta) \\ C_f(\theta) & D_f(\theta)\end{bmatrix}}_{\mathcal{F}(\theta)}\underbrace{\begin{bmatrix} 0 & I \\ C_2(\theta) & 0\end{bmatrix}}_{\mathcal{C}} \tag{9}$$
$$= A_0(\theta)+\mathcal{B}\mathcal{F}(\theta)\mathcal{C}$$

$$\bar{B}(\theta)=\begin{bmatrix} \underbrace{\begin{pmatrix}B_p(\theta) & E_p(\theta) & F_p(\theta)\end{pmatrix}}_{B_1(\theta)} \\ B_f(\theta)\underbrace{\begin{pmatrix} I & 0 & 0\\ D_p(\theta) & G_p(\theta) & H_p(\theta)\end{pmatrix}}_{D_{21}(\theta)}\end{bmatrix}$$
$$=\underbrace{\begin{bmatrix} B_1(\theta) \\ 0\end{bmatrix}}_{B_0(\theta)}+\underbrace{\begin{bmatrix} 0 & 0 \\ I & 0\end{bmatrix}}_{\mathcal{B}}\underbrace{\begin{bmatrix} A_f(\theta) & B_f(\theta) \\ C_f(\theta) & D_f(\theta)\end{bmatrix}}_{\mathcal{F}(\theta)}\underbrace{\begin{bmatrix} 0 \\ D_{21}(\theta)\end{bmatrix}}_{\mathcal{D}_{21}} \tag{10}$$
$$= B_0(\theta)+\mathcal{B}\mathcal{F}(\theta)\mathcal{D}_{21}$$

$$\bar{C}(\theta)=\begin{bmatrix} D_f(\theta)\underbrace{\begin{pmatrix}0\\ C_p(\theta)\end{pmatrix}}_{C_2(\theta)} & C_f(\theta)\end{bmatrix}$$
$$=\underbrace{\begin{bmatrix} 0 & I\end{bmatrix}}_{\mathcal{D}_{12}}\underbrace{\begin{bmatrix} A_f(\theta) & B_f(\theta) \\ C_f(\theta) & D_f(\theta)\end{bmatrix}}_{\mathcal{F}(\theta)}\underbrace{\begin{bmatrix} 0 & I \\ C_2(\theta) & 0\end{bmatrix}}_{\mathcal{C}} \tag{11}$$
$$= \mathcal{D}_{12}\mathcal{F}(\theta)\mathcal{C}$$

$$\bar{D}(\theta)=\begin{bmatrix} D_f(\theta) & 0 & -I \\ D_f(\theta)D_p(\theta) & D_f(\theta)G_p(\theta) & D_f(\theta)H_p(\theta)-I\end{bmatrix}$$
$$=\begin{bmatrix} 0 & 0 & -I \\ 0 & 0 & -I\end{bmatrix}+D_f(\theta)\underbrace{\begin{bmatrix} I & 0 & 0 \\ D_p(\theta) & G_p(\theta) & H_p(\theta)\end{bmatrix}}_{D_{21}} \tag{12}$$
$$=\underbrace{\begin{bmatrix} 0 & 0 & -I \\ 0 & 0 & -I\end{bmatrix}}_{D_{11}}+\underbrace{\begin{bmatrix} 0 & I\end{bmatrix}}_{\mathcal{D}_{12}}\underbrace{\begin{bmatrix} A_f(\theta) & B_f(\theta) \\ C_f(\theta) & D_f(\theta)\end{bmatrix}}_{\mathcal{F}(\theta)}\underbrace{\begin{bmatrix} 0 \\ D_{21}\end{bmatrix}}_{\mathcal{D}_{21}}$$
$$= D_{11}(\theta)+\mathcal{D}_{12}\mathcal{F}(\theta)\mathcal{D}_{21}$$

Eqs (6) to (12) can be rewritten in the form:

$$\begin{aligned}\dot{x}_{pf}(t) &= \bar{A}(\theta)x_{pf}(t)+\bar{B}(\theta)w_{udf}(t)\\ e_f(t) &= \bar{C}(\theta)x_{pf}(t)+\bar{D}(\theta)w_{udf}(t)\end{aligned} \tag{13}$$

with $x_{pf}(t)=\begin{bmatrix}x_p^T(t) & x_f^T(t)\end{bmatrix}^T$, $w_{udf}(t)=\begin{bmatrix}u^T(t) & d^T(t) & f^T(t)\end{bmatrix}$

$$\bar{A}(\theta)=A_0(\theta)+\mathcal{B}\mathcal{F}(\theta)\mathcal{C},\quad \bar{B}(\theta)=B_0(\theta)+\mathcal{B}\mathcal{F}(\theta)\mathcal{D}_{21} \tag{14}$$
$$\bar{C}(\theta)=\mathcal{D}_{12}\mathcal{F}(\theta)\mathcal{C},\qquad \bar{D}(\theta)=D_{11}(\theta)+\mathcal{D}_{12}\mathcal{F}(\theta)\mathcal{D}_{21}$$

and $A_0(\theta)=\begin{pmatrix}A_p(\theta) & 0\\ 0 & 0\end{pmatrix}$, $B_0(\theta)=\begin{pmatrix}B_1(\theta)\\ 0\end{pmatrix}$, $\mathcal{B}=\begin{pmatrix}0 & 0\\ I & 0\end{pmatrix}$,

$$\mathcal{C}=\begin{pmatrix}0 & I\\ C_2(\theta) & 0\end{pmatrix},\quad \mathcal{D}_{21}=\begin{pmatrix}0\\ D_{21}(\theta)\end{pmatrix},\quad \mathcal{D}_{12}=\begin{pmatrix}0 & I\end{pmatrix},$$

$$B_1(\theta)=\begin{pmatrix}B_p(\theta) & E_p(\theta) & F_p(\theta)\end{pmatrix},\quad D_{11}(\theta)=\begin{pmatrix}0 & 0 & -I\end{pmatrix} \tag{15}$$

$$C_2(\theta)=\begin{pmatrix}0\\ C_p(\theta)\end{pmatrix},\qquad D_{21}(\theta)=\begin{pmatrix}I & 0 & 0\\ D_p(\theta) & G_p(\theta) & H_p(\theta)\end{pmatrix}$$

LPV approach to robust fault estimation

It can be verified that (13) is a polytopic system in accordance with **(A.2)-(A.4)**. ***Lemma 1*** can be used as an adaptation of the results from [17].

Lemma 1

For (13), the following statements are equivalent:

(1) L_2-induced norm of the operator mapping $w_{udf}(t)$ into $e_f(t)$ is bounded by a scalar number γ for all parameter trajectories $\theta(t)$ in the polytope Θ,

(2) There exists $X=X^T>0$ satisfying the system of LMIs:

$$\begin{pmatrix} X\bar{A}(\theta_i)+\bar{A}^T(\theta_i)X & X\bar{B}(\theta_i) & \bar{C}^T(\theta_i)\\ \bar{B}^T(\theta_i)X & -\gamma I & \bar{D}^T(\theta_i)\\ \bar{C}(\theta_i) & \bar{D}(\theta_i) & -\gamma I\end{pmatrix}<0 \quad i=1,\cdots,r \tag{16}$$

The main results of this work can be stated through ***Theorem 1*** which provides the conditions leading to the solution of (6).

Theorem 1 ([17])

Consider the LPV system in (2) with assumptions **(A.1)-(A.4)**. There exists a polytopic LPV estimator in (5) that can determine the solution of (6) *if* there exist matrices:

$0<R_o=R_o^T\in R^{n\times n}$, $0<S_o=S_o^T\in R^{n\times n}$ such that:

$$\begin{pmatrix}\mathcal{N}_R & 0\\ 0 & I\end{pmatrix}^T\begin{pmatrix} A_p(\theta_i)R_o+R_oA_p^T(\theta_i) & 0 & B_1(\theta_i)\\ 0 & -\gamma I & D_{11}\\ B_1^T(\theta_i) & D_{11}^T & -\gamma I\end{pmatrix}\begin{pmatrix}\mathcal{N}_R & 0\\ 0 & I\end{pmatrix}^T<0 \tag{17}$$
$$i=1,\cdots,r$$

$$\begin{pmatrix}\mathcal{N}_S & 0\\ 0 & I\end{pmatrix}^T\begin{pmatrix} S_oA_p(\theta_i)+A_p^T(\theta_i)S_o & S_oB_1(\theta_i) & 0\\ B_1^T(\theta_i)S_o & -\gamma I & D_{11}^T\\ 0 & D_{11} & -\gamma I\end{pmatrix}\begin{pmatrix}\mathcal{N}_S & 0\\ 0 & I\end{pmatrix}<0 \tag{18}$$
$$i=1,\cdots,r$$

$$\begin{pmatrix}R_o & I\\ I & S_o\end{pmatrix}\geq 0 \tag{19}$$

By *Lemma 1* [see (6)] and considering the notations in (14) and (15), there exists a polytopic LPV fault estimator (5) which solves the ***Lemma 1*** if:

$$\Psi(\theta_i)+U_x^T\mathcal{F}(\theta_i)V+V^T\mathcal{F}^T(\theta_i)U_x<0\quad i=1,\cdots,r \tag{20}$$

where: $$\Psi(\theta_i)=\begin{pmatrix} XA_0(\theta_i)+A_0^T(\theta_i)X & XB_0(\theta_i) & 0\\ B_0^T(\theta_i)X & -\gamma I & D_{11}^T\\ 0 & D_{11} & -\gamma I\end{pmatrix} \tag{21}$$

$$U_x=\begin{pmatrix}\mathcal{B}^TX & 0 & \mathcal{D}_{12}^T\end{pmatrix} \tag{22}$$
$$V=\begin{pmatrix}\mathcal{C} & \mathcal{D}_{21} & 0\end{pmatrix} \tag{23}$$

Once the matrices R_o and S_o are obtained, the LPV estimator of (5) can be constructed as follows:

Algorithm 1 ([17])

Step1. Use SVD to compute the full rank matrices M_o, N_o:

$$M_o N_o^T = I - R_o S_o \tag{24}$$

Step 2. Compute X as the unique solution of:

$$X\begin{pmatrix} I & R_o \\ 0 & M_o^T \end{pmatrix} = \begin{pmatrix} S_o & I \\ N_o^T & 0 \end{pmatrix} \tag{25}$$

Step 3. Compute $\mathcal{F}(\theta_i)$ by solving (20).

Step 4. Solve the polytopic LPV estimator:

$$\mathcal{F}(\theta) = \sum_{i=1}^{r} \alpha_p^i \mathcal{F}(\theta_i) \tag{26}$$

α_p^i is any solution of the convex decomposition problem:

$$\theta = \sum_{i=1}^{r} \alpha_p^i \theta_i \tag{27}$$

5. TWO-LINK ROBOT CASE STUDY

To illustrate the mathematical discussion above, a tutorial example of the actuator fault compensation problem is considered using a nonlinear simulation of the two-link manipulator/robot. The robot manipulators are familiar examples of position-controllable mechanical systems ([21]). Their nonlinear dynamics present a challenging control problem, since traditional linear control approaches do not easily apply. The objective is to model the nonlinear dynamics for a two-joint manipulator, so that the movement control, e.g. from one point to another on is facilitated.

Two-link Manipulator Dynamics

Three types of dynamic torques arise from the motion of a manipulator system: *Inertial*, *Centripetal*, and *Coriolis* torques ([20] and [21]). Inertial torques are proportional to acceleration of each joint in accordance with Newton's second law. Centripetal torques arise from the centripetal forces which constrain a body to rotate about a point.

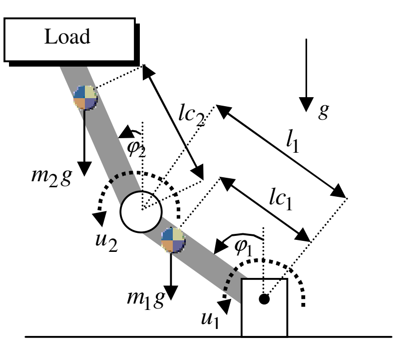

Fig. 1. Two-link manipulator structure

Centripetal torques are directed towards the centre of the uniform circular motion, and are proportional to the square of the velocity. Coriolis torques result from vertical forces derived from the interaction of two rotating links and are proportional to the product of the joint velocities of those links. For simplicity, the two-link robotic manipulator is considered to rotate in the vertical plane, and the equilibrium point is considered to be the upper vertical position, whose position can be described by a 2-vector $\varphi = (\varphi_1, \varphi_2)^T$ of joint angles, and whose actuator inputs consist of a 2-vector $u = (u_1, u_2)^T$ of torques applied at the manipulator joints as shown in Fig. 1. $\dot{\varphi}$, $\ddot{\varphi}$ denote the joint velocities and accelerations and the manipulator dynamics can be written in the more general form ([20], [21] and [22]) as:

$$\Xi(\varphi)\ddot{\varphi} + O(\varphi, \dot{\varphi})\dot{\varphi} + \mathcal{g}(\varphi) = u \tag{28}$$

$\Xi(\varphi) \in \Re^{2x2}$ is the SPD manipulator inertia tensor, the function $O(\varphi, \dot{\varphi})\dot{\varphi} \in \Re^2$ contains the Centripetal/Coriolis torques. $\mathcal{g}(\varphi) \in \Re^2$ are the gravitational torques. The following numerical example taken from ([21] and [22]) and modified here for the proposed design strategy in Section 4. The polytope model representation is now described.

The equations of motion are described by:

$$\begin{aligned}[m_1 lc_1^2 + m_2 l_1^2 + I_1]\ddot{\varphi}_1 + [m_2 l_1 lc_2 \cos(\varphi_1 - \varphi_2)\ddot{\varphi}_2 \\ + m_2 l_1 lc_2 \sin(\varphi_1 - \varphi_2)\dot{\varphi}_2^2 - [m_1 lc_1 + m_2 l_1] g \sin(\varphi_1) = u_1\end{aligned} \tag{29}$$

$$\begin{aligned}[m_2 l_1 lc_2 \cos(\varphi_1 - \varphi_2)\ddot{\varphi}_1 + [m_2 lc_2^2 + I_2]\ddot{\varphi}_2 \\ -[m_2 l_1 lc_2 \sin(\varphi_1 - \varphi_2)\dot{\varphi}_1^2 - m_2 g lc_2 \sin(\varphi_2) = u_2\end{aligned} \tag{30}$$

Parameters	***I₁***	***I₂***	***l₁***	***lc₁***	***lc₂***	***m₁***	***m₂***	***G***
Values	0.83	0.41	1	0.5	0.5	10	5	9.8
Units	Kg*m^2	Kg*m^2	m	m	m	Kg	Kg	m/s^2

Table 1: Parameter values of two-link manipulator system

I_1: Inertia of arm-1 and load, and I_2: Inertia of arm-2

l_1 : Distance between joint-1 and joint-2

lc_1: Distance of joint-1 from centre of mass arm-1

lc_2: Distance of joint-2 from centre of mass arm-2

m_1: Mass of arm-1 and load & m_2: Mass of arm-2

Note that in this study the quadratic terms $O(\varphi, \dot{\varphi})$ are not considered because they are not bounded. This is different from [23] in which the $O(\varphi, \dot{\varphi})$ term is taken into account in the robust control design for a two-link flexible manipulator. With this limitation (28) becomes:

$$\Xi(\varphi)\ddot{\varphi} + \mathcal{g}(\varphi) = u \tag{31}$$

where

$$\Xi(\varphi) = \begin{bmatrix} m_1 lc_1^2 + m_2 lc_1^2 + I_1 & m_2 l_1 lc_2 \cos(\varphi_1 - \varphi_2) \\ m_2 l_1 lc_2 \cos(\varphi_1 - \varphi_2) & m_2 lc_2^2 + I_2 \end{bmatrix},$$
$$\mathcal{g}(\varphi) = \begin{bmatrix} -[m_1 lc_1 + m_2 l_1] g \sin(\varphi_1) \\ -m_2 g lc_2 \sin(\varphi_2) \end{bmatrix} \tag{32}$$

The nonlinear term in $\Xi(\varphi)$ is clearly a bounded function:

$$\phi_1(\varphi) = \cos(\varphi_1 - \varphi_2) \tag{33}$$

where: $-1 \le \phi_1 \le 1$. Hence, $\Xi(\varphi)$ can be represented by a polytope whose vertices are defined by:

$$\Xi(\varphi) \in Co\{\Xi_1, \Xi_2\} \tag{34}$$

where $\Xi_1 = \begin{bmatrix} m_1 lc_1^2 + m_2 lc_1^2 + I_1 & m_2 l_1 lc_2 \\ m_2 l_1 lc_2 & m_2 lc_2^2 + I_2 \end{bmatrix}$,

$$\Xi_2 = \begin{bmatrix} m_1 lc_1^2 + m_2 lc_1^2 + I_1 & -m_2 l_1 lc_2 \\ -m_2 l_1 lc_2 & m_2 lc_2^2 + I_2 \end{bmatrix} \quad (35)$$

To facilitate a state-space formulation, the vector field $g(\varphi)$ with $\varphi \in \Re^2$ can be arranged in the form of $G^g(\varphi)\varphi$. The bounded function $\phi_2(\varphi)$ can now be defined as:

$$\sin(\varphi_1) = \left(\frac{\sin(\varphi_1)}{\varphi_1}\right)\varphi_1 = \phi_2(\varphi)\varphi_1 \quad (36)$$

where $-0.2 \le \phi_2(\varphi) \le 1$

From the boundedness of functions $\phi_2(\varphi)$ in terms of the angle φ, $G^g(\varphi)$ is considered as a polytope as follows:

$$G^g(\varphi) \in Co\left\{G_1^g, G_2^g, G_3^g, G_4^g\right\} \quad (37)$$

where:

$$G_1^g = \begin{pmatrix} 0.2[m_1 lc_1 + m_2 l_1]g & 0 \\ 0 & 0.2 m_2 g lc_2 \end{pmatrix}, \quad G_2^g = \begin{pmatrix} -[m_1 lc_1 + m_2 l_1]g & 0 \\ 0 & 0.2 m_2 g lc_2 \end{pmatrix}$$

$$G_3^g = \begin{pmatrix} 0.2[m_1 lc_1 + m_2 l_1]g & 0 \\ 0 & -m_2 g lc_2 \end{pmatrix}, \quad G_4^g = \begin{pmatrix} -[m_1 lc_1 + m_2 l_1]g & 0 \\ 0 & -m_2 g lc_2 \end{pmatrix}$$

The two-link system state space representation is defined as:

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} = \begin{bmatrix} \varphi_1(t) \\ \varphi_2(t) \\ \dot{\varphi}_1(t) \\ \dot{\varphi}_2(t) \end{bmatrix} \quad ; \quad W_b = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The state feedback LMI constraints, according to (29) and (30) are given by the following *descriptor system*: ([21] and [22]):

$$\begin{pmatrix} I & 0 \\ 0 & \Xi(\varphi) \end{pmatrix} \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \\ \dot{x}_4(t) \end{bmatrix} = \begin{pmatrix} 0 & I \\ -G^g(\varphi) & 0 \end{pmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + W_b\, u(t) \quad (38)$$

or the state space equation is presented as follows:

$$\dot{x}(t) = A(\varphi)x(t) + B(\varphi)u(t) \quad (39)$$

Let the matrix Π be a non-singular matrix given by:

$$\Pi = \begin{pmatrix} I & 0 \\ 0 & \Xi(\varphi) \end{pmatrix} \quad (40)$$

it then follows that:

$$A(\varphi) = \Pi^{-1}\begin{pmatrix} 0 & I \\ -G^g(\varphi) & 0 \end{pmatrix} \text{ and } B(\varphi) = \Pi^{-1} W_b$$

Actuator fault estimation

Consider a nominal time-varying model [depending smoothly on the angle φ] of the nonlinear dynamical system of (39), subject to actuator faults $F_a f_a(t)$, as follows:

$$\begin{aligned} \dot{x}(t) &= A(\varphi)x(t) + B(\varphi)u(t) + F_a f_a(t) \\ &= A_{ij}x(t) + B_i u(t) + F_a f_a(t) \quad (41) \\ & \qquad i = 1, 2 \quad j = 1, 2, \ldots, 4 \end{aligned}$$

F_a is the fault distribution matrix and the vectored signal f_a represents actuator faults. These gives rise to a polytopic controller with *8*-vertex systems. Therefore, the actuator fault estimate $\hat{f}_a(t)$ in system (41) can be implemented by using ***Algorithm 1*** and solved using the MATLAB© LMI toolbox in (24)-(27). The solution for $\gamma = 2.7550$, after 39 iterations. An estimator in (5) can be constructed by the ***Algorithm 1***. The polytopic LPV estimator parameters in (26) are:

$$\mathcal{F}(\theta) = \left(\begin{array}{cccc:cccc} -4.347E+02 & -4.365E+02 & -3.126E+02 & -3.982E+02 & 1.171E+06 & 1.359E+06 & -1.463E+06 & -1.667E+05 \\ -2.174E+02 & -2.727E+02 & -4.662E+02 & -3.387E+02 & 8.665E+05 & 1.018E+06 & -1.323E+06 & 6.172E+05 \\ 8.814E+02 & 5.495E+02 & -1.420E+03 & -1.098E+02 & -5.748E+05 & -5.788E+05 & -9.206E+05 & 4.979E+06 \\ 1.479E+02 & 1.170E+01 & -6.757E+02 & -2.271E+02 & 3.342E+05 & 4.078E+05 & -9.783E+05 & 1.827E+06 \\ \hdashline -5.344E-02 & -4.086E-02 & 1.115E-01 & -1.721E-02 & 7.742E+01 & 9.297E+01 & 1.193E+02 & -3.438E+02 \end{array}\right)$$

Figs. 2 & 3 show the result of the fault estimation, with a zero-mean Gaussian random disturbance $d(t)$ with variance 0.015. The polytopic system is simulated with scalar faults acting on the torque inputs at the manipulator joint-1 and 2, with the parameter trajectories of $\phi_1(\varphi)$, and $\phi_2(\varphi)$, and the actuator fault signals [i.e. $f_a(t) = col[f_{a1}(t), f_{a2}(t)]$ as shown. The fault signals are:

$$f_a(t) = col[2.50e10^{-2}\sin(0.5t), \quad 0.00] \text{ and}$$
$$f_a(t) = col[0.00, \quad 1.25e10^{-2}\sin(2t)],$$

Simulation results show that the designed polytopic LPV fault estimators provide very good estimation performance.

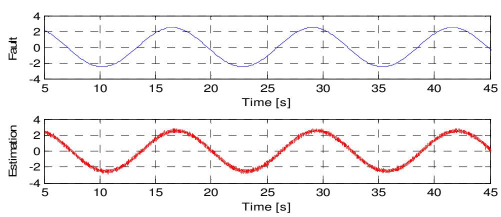

Fig. 2. The fault and its estimate using LPV estimator

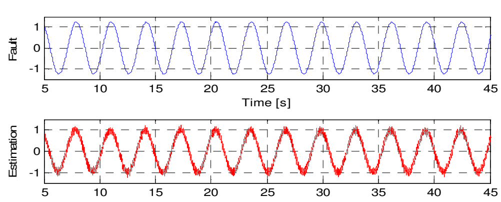

Fig. 3. The fault and its estimate using LPV estimator

After verifying the fault estimation performance the fault-free LPV controller design can be described as a basis for the Active FTC system.

6. POLYTOPIC LPV CONTROLLER DESIGN

The case study robot manipulator as described in Section 5 is used to illustrate various control polytopic modelling issues and FTC design. The control is developed for the nominal (fault-free) system (31). The objective is to compute the required actuator inputs to perform desired tasks (e.g. move the manipulated load to a desired position), given the measured system states, namely the vector φ of joint angles, and the vector $\dot{\varphi}$ of joint velocities.

Controller design for nominal (fault-free case)

Let a nominal state feedback control vector be $u_{nom}(t) = K_{lpv}x(t)$ [i.e. for the fault-free case], where $K_{lpv} \in \Re^{2x4}$ is controller gain matrix of the polytopic system to be designed. Before the nominal controller design can be completed it is first necessary to develop a stability condition that will be satisfied by all the LPV vertices. The nominal (fault-free) control system can be developed from (39) as:

$$\begin{aligned}\dot{x}(t) &= [A(\varphi) + B(\varphi)K_{lpv}]x(t) \\ &= [A_{ij} + B_i K_{lpv}]x(t) \quad i = 1, 2 \quad j = 1, 2, 3, 4\end{aligned} \qquad (42)$$

The following quadratic stability conditions are now defined at each vertex for the SPD Lyapunov matrix S_c:

$$(A_{ij} + B_i K_{lpv})S_c + S_c(A_{ij} + B_i K_{lpv})^T < 0 \quad i = 1, 2 \quad j = 1, 2, 3, 4 \quad (43)$$

Let $L_c = K_{lpv}S_c$, then $K_{lpv} = L_c S_c^{-1}$ and (43) is linear in L_c and K_{lpv}:

$$A_{ij}S_c + B_i L_c + S_c A_{ij}^T + L_c^T B_i^T < 0 \quad i = 1, 2 \quad j = 1, 2, 3, 4 \quad (44)$$

(43) & (44) lead to a polytopic controller with 8 vertex systems, with each system having *4* states, *2* inputs, and *2* outputs. S_c and L_c in (44) are computed using the MATLAB©LMI toolbox after 24 iterations and the controller can be constructed as:

$$K_{lpv} = \begin{pmatrix} -550.0309 & 0.0000 & -129.8775 & 0.0000 \\ 0.0000 & -135.2864 & 0.0000 & -39.2184 \end{pmatrix}$$

Fig. 4 shows that the polytopic LPV system remains stable with the movement from (20, 40) to (0, 0), although the quadratic terms, $O(\varphi, \dot{\varphi})$ are neglected in the model design.

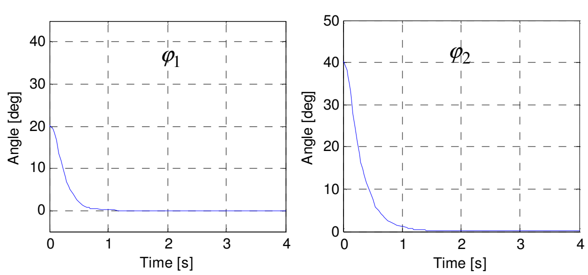

Fig. 4. The output responses of the nominal system

Design of control for active FTC scheme

From Section 5, the dynamic system of in (41) includes an additive description of the actuator faults. However, the faults can have a multiplicative effect in the system representation. A multiplicative actuator fault representation is defined as:

$$\dot{x}(t) = A_{ij}x(t) + B_i[I_m - \eta^a(t)]\,u(t) \quad i = 1, 2 \quad j = 1, 2, 3, 4 \qquad (45)$$

where η^a is the so-called *fault-effect factor*, and $\eta^a = diag[\eta_1^a, \eta_2^a, \ldots, \eta_m^a]$, and $0 \le \eta_i^a < 1$ represents a fault in the i^{th} actuator and $\eta_i^a = 0$ means that i^{th} actuator operates fault-free, whilst for $\eta_i^a > 0$ some degree of actuator fault effect occurs ([15] and [16]). (41) and (45) are equivalent to:

$$F_a f_a(t) = -B\eta^a(t)u(t) \qquad (46)$$

F_a is identical to the matrix B in the actuator fault case. The estimation of *fault-effect factor* $\hat{\eta}^a(t)$ is determined from the fault estimation $\hat{f}_a(t)$ provided by the fault estimator as described in Section 4. The adaptive active FTC scheme can be developed by considering the system with the actuator fault vector $f_a(t)$ described in (45) in terms of $\hat{\eta}^a(t)$, based on the nominal controller synthesized in Section 6 and achieved under the assumption that the fault effect factors η^a are provided by the estimator (5).

Theorem 2 From a design consideration consider the system in (45) with $i = 1, 2, \ldots, m$ actuator faults ($\eta^a \neq 0$) acting independently within each of the m vertex control systems with identical gain matrix K_{lpv} The new control (assuming non-zero fault effects) is given as:

$$\begin{aligned}u_{FTC}(t) &= \underbrace{[I - \hat{\eta}^a(t)]^+ K_{lpv}}_{K_{FTC}}x(t) \\ &= K_{FTC}(\hat{\eta}^a)x(t)\end{aligned} \qquad (47)$$

where $(I - \hat{\eta}^a)^+$ is the *Pseudo-Inverse* of $(I - \hat{\eta}^a)$ [see Fig. 5]. $K_{FTC}(\hat{\eta}^a)$ is the adaptive FTC controller gain, depending on the on-line estimation $\hat{\eta}^a$.

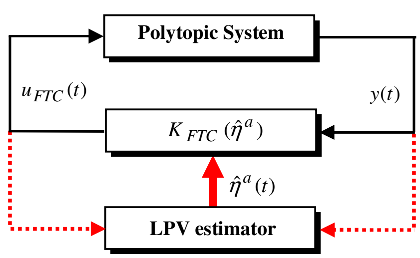

Fig. 5: Active fault-tolerant control scheme

Fig. 5 shows the active FTC structure, for the polytopic LPV system with exogenous disturbances and actuator faults as defined in (2). $K_{FTC}(\hat{\eta}^a)$ is the active FTC on-line adaptive gain matrix as in (47), and the $\hat{\eta}^a$ is an estimate of η^a, generated on-line using the polytopic LPV estimator (5).

Proof of Theorem 2: Consider (45) When an actuator fault occurs in a given vertex system, the controller of the complete polytope system is given by:

$$\dot{x}(t) = Ax(t) + B[I - \eta^a(t)]u(t) \qquad (48)$$

The new closed-loop LPV system is determined by substituting the new control law from (47) into the fault-corrupted system of (48), yielding:

$$
\begin{aligned}
\dot{x}(t) &= Ax(t) + B[I - \eta^{a}(t)]u_{FTC} \\
&= Ax(t) + B[(I - \eta^{a}(t)][(I - \hat{\eta}^{a}(t)]^{+} K_{lpv}x(t) \\
&= Ax(t) + Bu_{nom}(t)
\end{aligned} \quad (49)
$$

It can be seen that the term $(I - \eta^{a})$ acting on the system of (45) is removed by replacing u with u_{FTC} in (47). ■

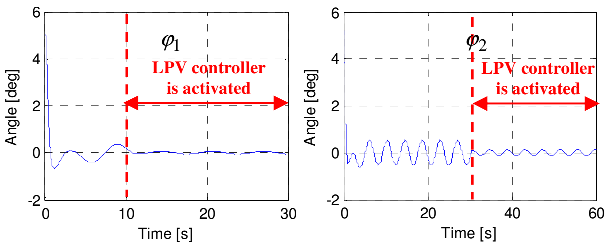

Fig.6. The control/output responses with active FTC [$f_{a1}(t) = 1.25e10^{-2}\sin(t)$ and $f_{a2}(t) = 2.50e10^{-2}\sin(1.5t)$]

Fig. 6 shows that if the polytopic LPV controller is employed under the assumption that the $\hat{\eta}^{a}$ can be estimated perfectly by the polytopic LPV estimator in (5), then the actuator fault can be compensated using new LPV control law in (47). It can be seen that the output performances of the angles; φ_1 and φ_2 soon return to their nominal/reference values with a very small amount of oscillation after LPV controller is activated at $t > 10$ and $t > 30$, respectively. This demonstrates very well the fault-tolerance of the LPV active FTC system.

7. CONCLUSION

This paper proposes a new strategy of an active FTC and polytopic LPV estimator for systems which can be implemented via a set of LMIs using efficient interior-point algorithms ([17]). A polytopic LPV estimator is synthesized for providing actuator fault estimates for use in an active FTC strategy to schedule predefined state feedback gains. The gains are calculated using LMIs for nominal and faulty cases in order to maintain the system performance over a wide operating range within according to a proposed polytopic model. The active FTC scheme is designed using the MATLAB©LMI toolbox, is a function of *fault effect factors* derived on-line from the polytopic LPV estimator. The scheme is investigated using a two-link manipulator with actuator faults acting on each torque input. Results show that the polytopic LPV estimator can follow the fault rapidly and effectively with robustness to disturbance signals, giving the system continued safe operation via the on-line FTC scheme.

ACKNOWLEDGEMENT

S Klinkhieo acknowledges PhD scholarship funding from the Synchrotron Light Research Institute (SLRI) under the Royal Thailand Government.

REFERENCES

[1] Leith D J and Leithead W E (2000), Survey of gain–scheduling analysis and design, *Int. J. Contol*, 73, 1001–1025

[2] Balas G J, (2002), Linear, parameter–varying control and its application to a turbo-fan engine, *Int. J. Robust & Nonlinear Control*, 12, 763–796.

[3] Wu F, (2001), A generalized LPV system analysis and control synthesis framework, *Int. J. Contol*, 74, 745–759.

[4] Ganguli S A, Marcos A and BalasG J, (2002), Reconfigurable LPV control design for Boeing 747-100/200 longitudinal axis, Proc. *ACC'2*, Anchorage, May 8-10, 3612–3617.

[5] Bokor J and Balas G, (2004),Detection filter design for LPV systems – a geometric approach, *Automatica*, 40, 511-518.

[6] Casavola A, Famularo D, Franze G and Patton R J (2008), A Fault Detection Filter Design Method for a class of Linear Time-Varying Systems, *16th IEEE Med. Conf. on Control & Automation*, Corsica.

[7] Casavola A, Famularo D, Franze G and Sorbara M, (2007), A fault detection filter design method for linear parameter-varying systems, *Proc. J. IMechE Part I*,865-874.

[8] Henry D and Zolghadri A, (2004), Robust fault diagnosis in uncertain Linear Parameter-Varying systems, Proc. SMC'2004, The Hague, NL.

[9] Zolghadri A, Henry D and Grenaille S, (2008), Fault diagnosis for LPV systems, *16th IEEE Med. Conf. on Control & Automation*, Corsica

[10] Shin J and Belcastro C M, (2006),Performance analysis on fault tolerant control system, *IEEE Trans. Control Sys. Tech.*, 14, 920–92.

[11] Rodrigues M, Theilliol M D, Aberkane S and Sauter D, (2007),Fault tolerant control design for polytopic LPV systems, *Inter. J. of App. Maths. & Comp. Science*, 17, 27–37.

[12] Rodrigues M, Theilliol D, and Sauter D, (2005), Design of an Active Fault-Tolerant control Polytopic Unknown Input Observer described by a Multi-model Representation, *CDC & ECC' 05*, 3815–3820, Seville.

[13] Weng Z, Patton R J and Cui P, (2007),Active fault-tolerant control of a double inverted pendulum, *J. of Sys. & Contr. Eng.*,. 221, (16), 895-904.

[14] Weng Z, Patton R J and Cui P, (2008),Integrated design of robust controller and fault estimator for linear parameter varying systems, *Proc. 17th IFAC World Congress*, Seoul, 4535-4539.

[15] J. Chen, R. J. Patton and Z. Chen, (1999),Active fault-tolerant flight Control systems design using the linear matrix inequality method, *Trans. Inst MC*, 21, (2/3).

[16] Chen J and Patton R J, (2001), Fault-Tolerant Control systems Design using the Linear Matrix Inequality Method, *ECC'01*, Porto.

[17] Apkarian P, Gahinet P and Becker G, (1995), Self-scheduled H∞ control of linear parameter-varying systems: a design example, *Automatica*, 31, 1251-1261.

[18] Packard A and Kantner M, (1992), Gain scheduling the LPV way, *Proc. of the CDC1992*, 1997-2001.

[19] Shamma J and Athans M, (1992), Gain scheduling: potential hazards and possible remedies, *IEEE Cont. Syst. Mag*, 12, (3), 101–107.

[20] McKerrow P J, *Introduction to Robotics*, (1991), Addison-Wesley.

[21] Hassen S, Crusca F and Abachi H, (2000),Modelling system for control studies – An overview, *ISCA 15th International conference on Computer & their Application*, Orlando.

[22] Kim J, (1997), Project Report: LMI optimization of Two-link Manipulator using Polytopic model and State feedback, *Department of Aerospace Eng.*, Inha University, South Korea

[23] Adams R J, Apkarian P, and Chretien J P, (1996), Robust control approaches for a two-link flexible manipulator, *3rd Inter. Conf. on Dynamics & Control of Structures in Space*, pp. 101–116, Cranfield, U.K