

Active fault-tolerant control design for Takagi-Sugeno fuzzy systems

Ł. DZIEKAN*, M. WITCZAK, and J. KORBICZ

Institute of Control and Computation Engineering, University of Zielona Góra, 50 Podgórna St., 65-246 Zielona Góra, Poland

Abstract. In this paper, a virtual actuator-based active fault-tolerant control strategy is presented. After a short introduction to Takagi-Sugeno fuzzy systems, it is shown how to design a fault-tolerant control strategy for this particular class of non-linear systems. The key contribution of the proposed approach is an integrated fault-tolerant control design procedure of fault identification and control within an integrated fault-tolerant control scheme. In particular, fault identification is implemented with the suitable state observer. While, the controller is implemented in such a way that the state of the (possibly faulty) system tracks the state of a fault-free reference model. Consequently, the fault-tolerant control stabilizes the possibly faulty system taking into account the input constraints and some control objective function. Finally, the last part of the paper shows a comprehensive case study regarding the application of the proposed strategy to fault-tolerant control of a twin-rotor system.

Key words: fault-tolerant control, state estimation; observers, fault diagnosis, convergence.

1. Introduction

Fault-Tolerant Control (FTC) [1] is one of the most important research directions underlying contemporary automatic control. FTC can also be perceived as an optimized integration of the advanced fault diagnosis [2, 3] and control [1] techniques. There is no doubt that the theory (and practice, as a consequence) of fault diagnosis and control is well-developed and mature for linear systems only [2]. There is also a number of different approaches that can be employed to settle the robustness problems regarding model uncertainty [3]. Such a robustness is especially important in practical implementations where various sources of uncertainty may be present, e.g. differences between various copies of a given component, time-varying properties, noise, external disturbances, etc. Another kind of solutions that may increase the performance of the FTC scheme is based on an appropriate scheduling of the control test signals in such a way to gain as much information as possible about the system being supervised [4]. Unfortunately, in spite of its originality, this technique can be used for linear systems only. In the light of the above discussion, it is clear that the development of new approaches for non-linear systems is justified. From this discussion, it is clear that FTC has become an important issue in modern automatic control theory and practice. This justifies the objectives of that aims at presenting an FTC for Takagi-Sugeno fuzzy systems.

In general, FTC systems are classified into two distinct classes [5]: passive and active. In passive FTC [6–9], controllers are designed to be robust against a set of presumed faults, therefore there is no need for fault detection, but such a design usually degrades the overall performance. In the contrast to passive ones, active FTC schemes, react to system components faults actively by reconfiguring control actions,

and by doing so the system stability and acceptable performance is maintained. To achieve that, the control system relies on the Fault Detection and Isolation (FDI) [2, 3, 10, 11] as well as an accommodation technique. Most of the existing works treat FDI and FTC problems separately. Unfortunately, perfect FDI and fault identification are impossible and hence there always is an error related to this process. Thus, there is a need for integrated FDI and FTC schemes for both linear and non-linear systems. One of the approach to the active fault tolerant control is control reconfiguration that creates dependable systems by means of the appropriate feedback control. It responds to severe component faults that open the control loop by on-line restructuring the controller [1]. It is possible to solve the reconfiguration problem by redesigning a new controller for the faulty system for every isolated fault. An optimal controller can be redesigned with the same optimization problem as in the nominal case as is shown in [12, 13], but the redesign step can become too complex for the large systems. Also if the controller is a human being, the replacement of the controller for reconfiguration implies the need for strong training efforts.

This paper relies on the idea of keeping the nominal controller in the loop, it avoids the implication of a complete controller redesign. By using a virtual actuator as introduced in [1, 14–16] it is possible to do so. It was also already demonstrated that the design of virtual actuators can be applied automatically after the fault has been detected. Prevention of a fault in one component (for example actuator) from causing system wide failures is the aim of control reconfiguration. The approach based on the use of virtual actuators is shown in Fig. 1, also it is worth mentioning that similar approach can be used for sensors faults, by using virtual sensors, but this is beyond the scope of this paper (see, e.g. [1]). The idea used

*e-mail: L.Dzieskan@issi.uz.zgora.pl

in this paper is to place a block between the controller output u and the input of all available actuators u_f . The goal of this block is to provide a signal which has the same effect as the broken actuator would have in the nominal system, therefore it is called a *virtual actuator*.

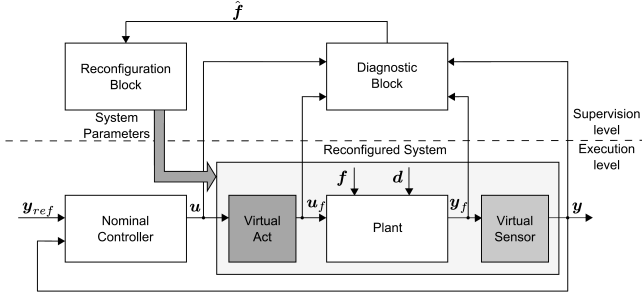


Fig. 1. Virtual actuators and virtual sensors in a fault tolerant control system

Successful application of the virtual actuator makes it possible to keep the behaviour of the faulty plant indistinguishable from that of the fault-free one [14]. The virtual actuator can be designed based on the open-loop behaviour, despite the fact that its purpose is to work in a closed-loop system. Therefore, the reconfiguration problem can be restated in terms of the open-loop system.

It might seem that this approach is more complex, but it has a considerable advantage over the redesign of the controller. Usually, the controller design is not an easy task and may be time consuming process, which involves several cycles of deriving and testing new controller parameters. Hence the redesign can be problematic to achieve on-line after the fault has been discovered. In this paper, the virtual actuator is used in addition to the original controller, which allows for the knowledge accumulated in designing the controller to be preserved and the reconfigured control loop can leverage on it. In comparison, the design of the virtual actuator itself is much more limited in the scope. As it presents fewer degrees of freedom it is easier to automate on-line in comparison to a complete controller redesign.

This paper is organised as follows. In Sec. 2 is presented background info about Takagi-Sugeno (T-S) fuzzy systems. In Sec. 3 an improved design technique for an integrated FTC and fault identification strategy for Takagi-Sugeno fuzzy systems is proposed that allows to include input constraints into FTC system. The proposed strategy is considerably less restrictive than the one presented in [17]. Indeed, the control matrix is not constant and the use of parameter-varying Lyapunov functions [18] is less conservative to the more usual approach of employing a common P matrix. subsequently the input constraints for the FTC and considered, which are followed by the regulator problem for T-S fuzzy system. The final part of the paper (Sec. 4) presents a comprehensive case study regarding the application of the proposed strategy to fault-tolerant control of a twin-rotor system.

2. Elementary background on T-S fuzzy systems

A non-linear dynamic system can be described in a simple way by a Takagi-Sugeno fuzzy model, which uses series of locally linearised models from the nonlinear system, parameter identification of an a priori given structure or transformation of a nonlinear model using the nonlinear sector approach (see, e.g. [2, 19, 20]). According to this model, a non-linear dynamic systems can be linearised around a number of operating points. Each of these linear models represents the local system behaviour around the operating point. Thus, a fuzzy fusion of all linear model outputs describes the global system behaviour. A T-S model is described by fuzzy IF-THEN rules which represent local linear I/O relations of the nonlinear system. It has a rule base of M rules, each having p antecedents, where i -th rule is expressed as

$$R^i: \text{IF } w_k^1 \text{ is } F_1^i \text{ and } \dots \text{ and } w_k^p \text{ is } F_p^i, \\ \text{then } \begin{cases} \mathbf{x}_{k+1} = \mathbf{A}^i \mathbf{x}_k + \mathbf{B}^i \mathbf{u}_k, \\ \mathbf{y}_k = \mathbf{C}^i \mathbf{x}_k, \end{cases} \quad (1)$$

in which $\mathbf{x}_k \in \mathbb{R}^n$ stands for the reference state, $\mathbf{y}_k \in \mathbb{R}^m$ is the reference output, and $\mathbf{u}_k \in \mathbb{R}^r$ denotes the nominal control input, also $i = 1, \dots, M$, F_j^i ($j = 1, \dots, p$) are fuzzy sets and $\mathbf{w}_k = [w_k^1, w_k^2, \dots, w_k^p]$ is a known vector of premise variables [2, 19].

Given a pair of $(\mathbf{w}_k, \mathbf{u}_k)$ and a product inference engine, the final output of the normalized T-S fuzzy model can be inferred as:

$$\begin{cases} \mathbf{x}_{k+1} = \sum_{i=1}^M h_i(\mathbf{w}_k) [\mathbf{A}^i \mathbf{x}_k + \mathbf{B}^i \mathbf{u}_k], \\ \mathbf{y}_k = \sum_{i=1}^M h_i(\mathbf{w}_k) \mathbf{C}^i \mathbf{x}_k, \end{cases} \quad (2)$$

where $h_i(\mathbf{w}_k)$ are normalized rule firing strengths defined as

$$h_i(\mathbf{w}_k) = \frac{\mathcal{T}_{j=1}^p \mu_{F_j^i}(w_k^j)}{\sum_{i=1}^M (\mathcal{T}_{j=1}^p \mu_{F_j^i}(w_k^j))} \quad (3)$$

and \mathcal{T} denotes a t -norm (e.g., product). The term $\mu_{F_j^i}(w_k^j)$ is the grade of membership of the premise variable w_k^j . Moreover, the rule firing strengths $h_i(\mathbf{w}_k)$ ($i = 1, \dots, M$) satisfy the following constraints

$$\begin{cases} \sum_{i=1}^M h_i(\mathbf{w}_k) = 1, \\ 0 \leq h_i(\mathbf{w}_k) \leq 1, \quad \forall i = 1, \dots, M. \end{cases} \quad (4)$$

3. FTC strategy for T-S fuzzy systems

Let us consider the following T-S reference model:

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k, \quad (5)$$

$$\mathbf{y}_{k+1} = \mathbf{C}_{k+1} \mathbf{x}_{k+1}, \quad (6)$$

with

$$\begin{aligned} \mathbf{A}_k &= \sum_{i=1}^M h_i(\mathbf{w}_k) \mathbf{A}^i, \\ \mathbf{B}_k &= \sum_{i=1}^M h_i(\mathbf{w}_k) \mathbf{B}^i, \\ \mathbf{C}_{k+1} &= \sum_{i=1}^M h_i(\mathbf{w}_{k+1}) \mathbf{C}^i \end{aligned}$$

for $i = 1, \dots, M$.

Let us also consider a possibly faulty T-S system described by the following equations:

$$\mathbf{x}_{f,k+1} = \mathbf{A}_k \mathbf{x}_{f,k} + \mathbf{B}_k \mathbf{u}_{f,k} + \mathbf{L}_k \mathbf{f}_k, \quad (7)$$

$$\mathbf{y}_{f,k+1} = \mathbf{C}_{k+1} \mathbf{x}_{f,k+1}, \quad (8)$$

with $\mathbf{L}_k = \sum_{i=1}^M h_i(\mathbf{w}_k) \mathbf{L}^i$. Where $\mathbf{x}_{f,k} \in \mathbb{R}^n$ stands for the system state, $\mathbf{y}_{f,k} \in \mathbb{R}^m$ is the system output, $\mathbf{u}_{f,k} \in \mathbb{R}^r$ denotes the system input, $\mathbf{f}_k \in \mathbb{R}^s$, ($s \leq m$) is the fault vector, and \mathbf{L}^i stands for its distribution matrices which are assumed to be known.

The main objective of this paper is to propose a control strategy which can be used for determining the system input $\mathbf{u}_{f,k}$ such that:

- the control loop for the system (7)–(8) is stable,
- $\mathbf{x}_{f,k+1}$ converges asymptotically to \mathbf{x}_{k+1} irrespective of the presence of the fault \mathbf{f}_k .

The subsequent part of this section shows the development details of the scheme that is able to settle such a challenging problem.

The crucial idea is to use the following control strategy:

$$\mathbf{u}_{f,k} = -\mathbf{S}_k \hat{\mathbf{f}}_k + \mathbf{K}_{1,k}(\mathbf{x}_k - \hat{\mathbf{x}}_{f,k}) + \mathbf{u}_k, \quad (9)$$

where $\hat{\mathbf{f}}_k$ is the fault estimate. Note that it is not assumed that $\mathbf{x}_{f,k}$ is available, i.e. an estimate $\hat{\mathbf{x}}_{f,k}$ can be used instead. Thus, the following problems arise:

- to determine $\hat{\mathbf{f}}_k$,
- to design $\mathbf{K}_{1,k}$ in such a way that the control loop is stable, i.e. the stabilisation problem.

3.1. Fault identification. Let us assume that the following rank condition is satisfied at any given moment¹

$$\text{rank}(\mathbf{C}_{k+1} \mathbf{L}_k) = \text{rank}(\mathbf{L}_k) = s. \quad (10)$$

This implies that it is possible to calculate

$$\begin{aligned} \mathbf{H}_{k+1} &= (\mathbf{C}_{k+1} \mathbf{L}_k)^+ = \\ &= [(\mathbf{C}_{k+1} \mathbf{L}_k)^T \mathbf{C}_{k+1} \mathbf{L}_k]^{-1} (\mathbf{C}_{k+1} \mathbf{L}_k)^T. \end{aligned} \quad (11)$$

By multiplying (8) by \mathbf{H}_{k+1} and then substituting (7), it can be shown that

$$\mathbf{f}_k = \mathbf{H}_{k+1}(\mathbf{y}_{f,k+1} - \mathbf{C}_{k+1} \mathbf{A}_k \mathbf{x}_{f,k} - \mathbf{C}_{k+1} \mathbf{B}_k \mathbf{u}_{f,k}). \quad (12)$$

Thus, if $\hat{\mathbf{x}}_{f,k}$ is used instead of $\mathbf{x}_{f,k}$ then the fault estimate is given as follows

$$\hat{\mathbf{f}}_k = \mathbf{H}_{k+1}(\mathbf{y}_{f,k+1} - \mathbf{C}_{k+1} \mathbf{A}_k \hat{\mathbf{x}}_{f,k} - \mathbf{C}_{k+1} \mathbf{B}_k \mathbf{u}_{f,k}). \quad (13)$$

and the associated fault estimation error is

$$\mathbf{f}_k - \hat{\mathbf{f}}_k = -\mathbf{H}_{k+1} \mathbf{C}_{k+1} \mathbf{A}_k (\mathbf{x}_{f,k} - \hat{\mathbf{x}}_{f,k}). \quad (14)$$

Unfortunately, the crucial problem with practical implementation of (13) is that it requires $\mathbf{y}_{f,k+1}$ and $\mathbf{u}_{f,k}$ to calculate $\hat{\mathbf{f}}_k$ and hence it cannot be directly used to obtain (9). To settle this problem, it is assumed that there exists a diagonal matrix α_k such that $\hat{\mathbf{f}}_k \cong \mathbf{f}_k = \alpha_k \hat{\mathbf{f}}_{k-1}$ and hence the practical form of (9) boils down to

$$\mathbf{u}_{f,k} = -\mathbf{S}_k \hat{\mathbf{f}}_k + \mathbf{K}_{1,k}(\mathbf{x}_k - \hat{\mathbf{x}}_{f,k}) + \mathbf{u}_k. \quad (15)$$

In most cases the matrix α_k should be equivalent to an identity matrix, i.e. it would simply mean an one time-step delay, which should have a negligible effect on the outcome. In cases where the fault behaviour is a linear one, it is possible to design the matrix α_k based on the previous changes of faults. In cases where faults changes in a nonlinear fashion and one time-step delay is unacceptable, one could try to predict the nature of the faults by using for example neural networks.

3.2. Stabilisation problem. By substituting (9) into (7), it can be shown that

$$\begin{aligned} \mathbf{x}_{f,k+1} &= \mathbf{A}_k \mathbf{x}_{f,k} - \mathbf{B}_k \mathbf{S}_k \hat{\mathbf{f}}_k + \mathbf{B}_k \mathbf{K}_{1,k} \mathbf{e}_k + \\ &+ \mathbf{B}_k \mathbf{K}_{1,k} \mathbf{e}_{f,k} + \mathbf{B}_k \mathbf{u}_k + \mathbf{L}_k \mathbf{f}_k, \end{aligned} \quad (16)$$

where $\mathbf{e}_k = \mathbf{x}_k - \mathbf{x}_{f,k}$ stands for the tracking error.

Let us assume that \mathbf{S}_k at any moment satisfies the following equality $\mathbf{B}_k \mathbf{S}_k = \mathbf{L}_k$, e.g. for actuator faults $\mathbf{S}_k = \mathbf{I}$. Thus

$$\begin{aligned} \mathbf{x}_{f,k+1} &= \mathbf{A}_k \mathbf{x}_{f,k} + \mathbf{L}_k (\mathbf{f}_k - \hat{\mathbf{f}}_k) + \\ &+ \mathbf{B}_k \mathbf{K}_{1,k} \mathbf{e}_k + \mathbf{B}_k \mathbf{K}_{1,k} \mathbf{e}_{f,k} + \mathbf{B}_k \mathbf{u}_k. \end{aligned} \quad (17)$$

Finally, by substituting (14) into (17) and then applying the result into $\mathbf{e}_{k+1} = \mathbf{x}_{k+1} - \mathbf{x}_{f,k+1}$ yields

$$\begin{aligned} \mathbf{e}_{k+1} &= (\mathbf{A}_k - \mathbf{B}_k \mathbf{K}_{1,k}) \mathbf{e}_k + \\ &+ (\mathbf{L}_k \mathbf{H}_{k+1} \mathbf{C}_{k+1} \mathbf{A}_k - \mathbf{B}_k \mathbf{K}_{1,k}) \mathbf{e}_{f,k}. \end{aligned} \quad (18)$$

where $\mathbf{e}_{f,k} = \mathbf{x}_{f,k} - \hat{\mathbf{x}}_{f,k}$ stands for the state estimation error.

3.3. Observer design. As was already mentioned, the fault estimate (13) is obtained based on the state estimate $\hat{\mathbf{x}}_{f,k}$. This raises the necessity for an observer design. Consequently, by substituting (12) into (7) it is possible to show that

$$\mathbf{x}_{f,k+1} = \bar{\mathbf{A}}_k \mathbf{x}_{f,k} + \bar{\mathbf{B}}_k \mathbf{u}_{f,k} + \bar{\mathbf{L}}_k \mathbf{y}_{f,k+1}, \quad (19)$$

where

$$\begin{aligned} \bar{\mathbf{A}}_k &= (\mathbf{I} - \mathbf{L}_k \mathbf{H}_{k+1} \mathbf{C}_{k+1}) \mathbf{A}_k, \\ \bar{\mathbf{B}}_k &= (\mathbf{I} - \mathbf{L}_k \mathbf{H}_{k+1} \mathbf{C}_{k+1}) \mathbf{B}_k, \\ \bar{\mathbf{L}}_k &= \mathbf{L}_k \mathbf{H}_{k+1}. \end{aligned}$$

¹It is not easy to guarantee that, unless matrices \mathbf{C}_k and \mathbf{L}_k are time invariant, i.e. $\mathbf{C}^1 = \mathbf{C}^i$ and $\mathbf{L}^1 = \mathbf{L}^i$ for all $i = 1, \dots, M$. However in real life cases, checking if rank condition is satisfied for every pair of matrices, i.e. $\text{rank}(\mathbf{C}^i \mathbf{L}^i) = \text{rank}(\mathbf{L}^i) = s$ for all $i = 1, \dots, M$ is usually sufficient.

Thus, the observer structure, which can be perceived as an unknown input observer (see, e.g. [3, 21]), is given by

$$\hat{\mathbf{x}}_{f,k+1/k} = \bar{\mathbf{A}}_k \hat{\mathbf{x}}_{f,k} + \bar{\mathbf{B}}_k \mathbf{u}_{f,k} + \bar{\mathbf{L}}_k \mathbf{y}_{f,k+1},$$

$$\hat{\mathbf{x}}_{f,k+1} = \hat{\mathbf{x}}_{f,k+1/k} + \mathbf{K}_{2,k+1} (\mathbf{y}_{f,k+1} - \mathbf{C}_{k+1} \hat{\mathbf{x}}_{f,k+1/k}), \quad (20)$$

Finally, the state estimation error can be written as follows:

$$\mathbf{e}_{f,k+1} = (\bar{\mathbf{A}}_k - \mathbf{K}_{2,k+1} \mathbf{C}_{k+1} \bar{\mathbf{A}}_k) \mathbf{e}_{f,k}. \quad (21)$$

3.4. Integrated design procedure. The main objective of this section is to summarise the presented results within an integrated framework for the development of fault identification and fault-tolerant control scheme. First, let us start with two crucial assumptions:

- the pair $(\bar{\mathbf{A}}_k, \mathbf{C}_{k+1})$ is detectable,
- the pair $(\mathbf{A}_k, \mathbf{B}_k)$ is stabilisable.

Under these assumptions, it is possible to design the matrices $\mathbf{K}_{1,k}$ and $\mathbf{K}_{2,k}$ in such a way that the extended error

$$\bar{\mathbf{e}}_k = \begin{bmatrix} \mathbf{e}_k \\ \mathbf{e}_{f,k} \end{bmatrix}, \quad (22)$$

described by

$$\begin{aligned} & \bar{\mathbf{e}}_{k+1} = \\ & = \begin{bmatrix} \mathbf{A}_k - \mathbf{B}_k \mathbf{K}_{1,k} & \mathbf{L}_k \mathbf{H}_{k+1} \mathbf{C}_{k+1} \mathbf{A}_k - \mathbf{B}_k \mathbf{K}_{1,k} \\ \mathbf{0} & \bar{\mathbf{A}}_k - \mathbf{K}_{2,k+1} \mathbf{C}_{k+1} \bar{\mathbf{A}}_k \end{bmatrix} \bar{\mathbf{e}}_k. \end{aligned} \quad (23)$$

converges asymptotically to zero.

It can be observed from the structure of (23) that the eigenvalues of the matrix are the union of those of $\mathbf{A}_k - \mathbf{B}_k \mathbf{K}_{1,k}$ and $\bar{\mathbf{A}}_k - \mathbf{K}_{2,k+1} \mathbf{C}_{k+1} \bar{\mathbf{A}}_k$. This clearly indicates that the design of the state feedback and the observer can be carried out independently (separation principle). So, let us start with the controller design with the corresponding tracking error defined by

$$\mathbf{e}_{k+1} = [\mathbf{A}_k - \mathbf{B}_k \mathbf{K}_{1,k}] \mathbf{e}_k = \mathbf{A}_0(\mathbf{h}(\mathbf{w}_k)) \mathbf{e}_k, \quad (24)$$

where

$$\mathbf{K}_{1,k} = \sum_{i=1}^M h_i(\mathbf{w}_k) \mathbf{K}_1^i$$

and the matrix $\mathbf{A}_0(\mathbf{h}(\mathbf{w}_k))$ belongs to a convex polytopic set defined as

$$\begin{aligned} \mathbb{A}_0 &= \left\{ \mathbf{A}_0(\mathbf{h}(\mathbf{w}_k)) : \sum_{i=1}^M h_i(\mathbf{w}_k) = 1, \quad 0 \leq h_i(\mathbf{w}_k) \leq 1 \right. \\ & \mathbf{A}_0(\mathbf{h}(\mathbf{w}_k)) = \sum_{i=1}^M \sum_{j=1}^M h_i(\mathbf{w}_k) h_j(\mathbf{w}_k) \mathbf{A}_{0,i,j}, \\ & \left. \mathbf{A}_{0,i,j} = \frac{1}{2} (\mathbf{A}^i - \mathbf{B}^i \mathbf{K}_1^j + \mathbf{A}^j - \mathbf{B}^j \mathbf{K}_1^i) \right\}. \end{aligned} \quad (25)$$

By adapting the general results of the work of [18], the following definition is introduced:

Definition 1. The tracking error described by (24) is robustly convergent to zero in the uncertainty domain (25) iff all eigenvalues of $\mathbf{A}_0(\mathbf{h}(\mathbf{w}_k))$ have magnitude less than one for all values of $\mathbf{h}(\mathbf{w}_k)$ such that $\mathbf{A}_0(\mathbf{h}(\mathbf{w}_k)) \in \mathbb{A}_0$.

Theorem 2. The tracking error described by (24) is robustly convergent to zero in the uncertainty domain (25) if there exist matrices $\mathbf{Q}_{i,j} \succ \mathbf{0}$, \mathbf{G}_1 , \mathbf{W}_j such that

$$\begin{bmatrix} \mathbf{G}_1 + \mathbf{G}_1^T - \mathbf{Q}_{i,j} & * \\ \mathbf{N}_{0,i,j} & \mathbf{Q}_{m,n} \end{bmatrix} \succ \mathbf{0}, \quad (26)$$

for all $i, m = 1, \dots, M$ and $j \geq i, n \geq m$, where

$$\mathbf{N}_{0,i,j} = \frac{1}{2} [(\mathbf{A}^i + \mathbf{A}^j) \mathbf{G}_1 - \mathbf{B}^i \mathbf{W}_j - \mathbf{B}^j \mathbf{W}_i].$$

Proof. See [18].

Finally, the design procedure boils down to solving the set of

$$\left[\frac{1}{2} M(1+M) \right]^2$$

LMI (26) and then determining $\mathbf{K}_1^i = \mathbf{W}_i \mathbf{G}_1^{-1}$. Since the controller design procedure is provided, then the observer synthesis procedure can be described. To tackle this problem, it is proposed to use a modified version of the celebrated Kalman filter, which can be described as follows:

$$\hat{\mathbf{x}}_{f,k+1/k} = \bar{\mathbf{A}}_k \hat{\mathbf{x}}_{f,k} + \bar{\mathbf{B}}_k \mathbf{u}_{f,k} + \bar{\mathbf{L}}_k \mathbf{y}_{f,k+1},$$

$$\mathbf{P}_{k+1/k} = \bar{\mathbf{A}}_k \mathbf{P}_k \bar{\mathbf{A}}_k^T + \mathbf{U}_k,$$

$$\mathbf{K}_{2,k+1} = \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T \left(\mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T + \mathbf{V}_{k+1} \right)^{-1},$$

$$\hat{\mathbf{x}}_{f,k+1} = \hat{\mathbf{x}}_{f,k+1/k} + \mathbf{K}_{2,k+1} (\mathbf{y}_{f,k+1} - \mathbf{C}_{k+1} \hat{\mathbf{x}}_{f,k+1/k}),$$

$$\mathbf{P}_{k+1} = [\mathbf{I} - \mathbf{K}_{2,k+1} \mathbf{C}_{k+1}] \mathbf{P}_{k+1/k},$$

where $\mathbf{U}_k = \delta_1 \mathbf{I}$ and $\mathbf{V}_k = \delta_2 \mathbf{I}$ with δ_1 and δ_2 sufficiently small positive numbers.

It is important to note that the Kalman filter is applied here for state estimation of a deterministic system (7)–(8) and hence \mathbf{U}_k and \mathbf{V}_k play the role of instrumental matrices only (see [3] and the references therein for more details).

3.5. Constraints on the control input. When the initial tracking error is known (i.e., the deviation of a faulty system state from a nominal system state), an upper bound on the norm of the control input $\hat{\mathbf{u}}_{f,k} = \mathbf{K}_1(\mathbf{x}_k - \mathbf{x}_{f,k})$ can be found as follows [22]. Let us assume that initial tracking error \mathbf{e}_0 lies in an ellipsoid of diameter γ , i.e., $\|\mathbf{e}_0\| \leq \gamma$, then the constraint on a control input described as follows $\|\hat{\mathbf{u}}_{f,k}\|_{\max} \triangleq \max_t |\hat{u}_{f,k}^t| \leq \lambda$ is enforced at all times if the LMIs

$$\begin{aligned} & \begin{bmatrix} \mathbf{X} & * \\ 0.5(\mathbf{W}_i^T + \mathbf{W}_j^T) & \mathbf{G}_1 + \mathbf{G}_1^T - \mathbf{Q}_{i,j} \end{bmatrix} \succeq \mathbf{0}, \\ & \mathbf{Q}_{i,j} \succeq \gamma^2 \mathbf{I}, \quad \text{diag}(\mathbf{X}) \preceq \lambda^2 \mathbf{I}, \end{aligned} \quad (27)$$

hold, where W_i, W_j satisfy conditions given by (26) for all $i = 1, \dots, M$ and $j \geq i$.

3.6. Regulator problem. In order to solve the regulator problem it is needed to find a state feedback controller such that the following objective function is minimized,

$$J_\infty = \sum_0^\infty \left(\hat{y}_{f,k}^T Q_R \hat{y}_{f,k} + \hat{u}_{f,k}^T R_R \hat{u}_{f,k} \right) \quad (28)$$

where $\hat{y}_{f,k} = y_{f,k} - y_k$, $Q_R \succeq 0$ and $R_R \succ 0$ are suitable weight matrices.

However system described in this paper is uncertain and thus only the upper bound of the objective function can be minimized. Therefore the following Theorem 3 only gives a sub-optimal solution for the regulator problem [18].

Theorem 3. The upper bound for the objective function (28) for initial tracking error e_0 lying in an ellipsoid of diameter γ can be obtained by solving the following LMI optimization problem

$$\min_{Q_{i,j}, G_1, W_i} \eta$$

subject to

$$\begin{bmatrix} G_1 + G_1^T - Q_{i,j} & * & * & * \\ N_{0,i,j} & Q_{m,n} & * & * \\ 0.5Q_R^{1/2}(C^i + C^j)G_1 & 0 & \eta I & * \\ 0.5R_R^{1/2}(W_i + W_j) & 0 & 0 & \eta I \end{bmatrix} \succ 0, \quad (29)$$

$$Q_{i,j} \succeq \gamma^2 I$$

for all $i, m = 1, \dots, M$ and $j \geq i, n \geq m$, where

$$N_{0,i,j} = \frac{1}{2}[(A^i + A^j)G_1 - B^i W_j - B^j W_i]$$

and local feedbacks gains are $K_1^i = W_i G_1^{-1}$.

Proof. See [18].

3.7. Implementation details. This section is presented to clarify the order of use of the equations and by doing so avoid some potential confusion about implementing the presented technique.

The initial stage is to compute virtual actuator gains K_1^i for all $i = 1, \dots, M$, by solving LMIs described by (26) or (29) (if a regulator is needed), also specific input constraints (27) can also be considered in either case if required. By doing so a virtual actuator is achieved presented on Fig. 2. Also if system matrices B^i, L^i, C^i are time invariant similar approach can be used to design observer matrices K_2^i instead of using a somewhat more complicated modified version of the Kalman filter. After initial design steps of a FTC system the following procedure must be executed in every time step.

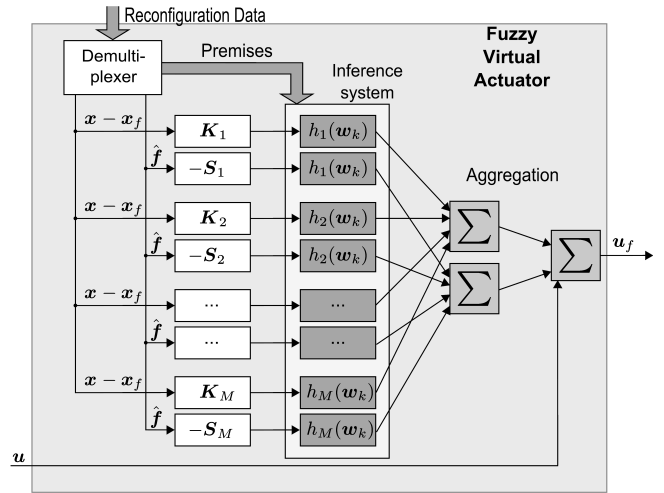


Fig. 2. Virtual actuator for Takagi-Sugeno fuzzy systems

The first step is to compute the virtual actuator output described by (15), to do so there is a need to use the current control input, difference between a previously estimated state of a possibly faulty system and a state of a nominal system, also because current fault estimate is not available there is a need to use $\hat{f}_k = \alpha_k \hat{f}_{k-1}$. As it was mentioned before, using one time step delay (i.e. $\alpha_k = I$) in most cases should be sufficient, otherwise one can use a linear approximation of the faults (not recommended for abrupt faults) for example $\alpha_k = \text{diag}(\alpha_k^1, \dots, \alpha_k^s)$, $\alpha_k^i = \hat{f}_{k-1}^i / \hat{f}_{k-2}^i, i = 1, \dots, s$ or some other nonlinear methods.

Afterwards it is possible to find output of a possibly faulty system achieved by using new control input (from virtual actuator) and also find reference system state using nominal input on (5). By doing so it is possible to compute estimate of the state of a possibly faulty system by using (20). And finally compute fault estimate by using (12).

4. Illustrative example

The selected non-linear system is based on the twin-rotor multiple input multiple output (MIMO) system (TRMS), being a laboratory set-up developed by Feedback Instruments Limited [24] for control experiments. Due to its high non-linearity, cross coupling between its two axes, and inaccessibility of some its outputs and states for measurements, the system is often perceived as a challenging engineering problem. Extensive research on modelling of such system can be found here [23] and in the references therein. The TRMS as shown in Fig. 3 is being driven by two d.c. motors. It has two propellers perpendicular to each other and joined by a beam pivoted on its base, so that it can rotate in such a way that its ends move on spherical surfaces. The joined beam can be moved by changing the input voltage of its motor, which controls the rotational speed of the propellers. The system is equipped with a pendulum counterweight fixed to the beam and it determines a stable equilibrium position. Additionally the system is balanced in such a way, that when the motors are switched off, the main rotor end of beam is lowered. In certain aspects the behaviour of the TRMS system resembles that of a helicopter [23]. For

example, there is a strong cross-coupling between the main rotor (collective) and the tail rotor, like a helicopter. However, the system is different from a helicopter in many ways, the main differences are: location of pivot point (midway between two rotors in TRMS vs main rotor head in helicopter), vertical control (speed control of main rotor vs collective pitch control), yaw control (tail rotor speed vs pitch angle of tail rotor blades) and lastly, cyclical control (none vs directional control).

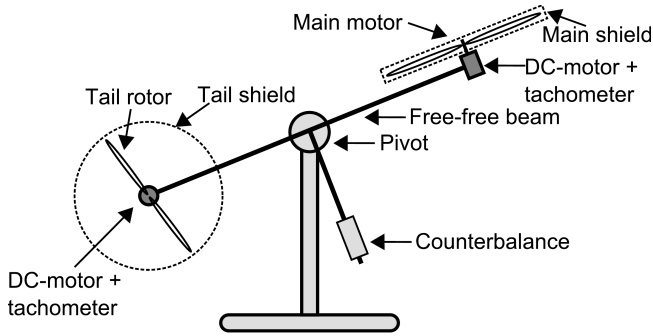


Fig. 3. Aero-dynamical model of the Twin Rotor MIMO System

The mathematical model of the TRMS can be described by a set of four non-linear differential equations with two linear differential equations and four non-linear functions [24]. Some of the parameters can be obtained from manual, whereas others should be obtained by experimentations, e.g., inertia, magnitudes of physical propeller, coefficients of friction and impulse force. The inputs of the system are defined by the input vector $\mathbf{u} = [\mathbf{u}_h, \mathbf{u}_v]^T$, where \mathbf{u}_h is the input voltage of the tail motor and \mathbf{u}_v is the input voltage of the main motor. Whereas the state vector is defined as $\mathbf{x} = [\Omega_h, \alpha_h, \omega_t, \Omega_v, \alpha_v, \omega_m]^T$, where Ω_h is the angular velocity around the vertical axis, α_h is the azimuth angle of beam, ω_t is the rotational velocity of the tail rotor, Ω_v is the angular velocity around the horizontal axis, α_v is the pitch angle of beam, ω_m is the rotational velocity of the main rotor. For the complete physical model of such a system refer here [23, 24].

A normalized T-S model, which approximates the non-linear TRMS system, is obtained by linearising system around five operating points [25]. The system can be described in the following way:

$$\begin{cases} \mathbf{x}_{k+1} = \sum_{i=1}^5 h_i(\alpha_{h,k}) [\mathbf{A}^i \mathbf{x}_k(t) + \mathbf{B}^i (\mathbf{u}_k - \mathbf{u}^i) + \mathbf{L}^i \mathbf{f}_k] \\ \mathbf{y}_k = \mathbf{C}^i \mathbf{x}_k + \mathbf{d}^i \end{cases} \quad (30)$$

The matrices \mathbf{A}^i , \mathbf{B}^i , \mathbf{C}^i , \mathbf{u}^i and \mathbf{d}^i (trend) are acquired by linearising the initial system around five points chosen in the operating range of the considered system, with the premise variable $\mathbf{w}_k = \alpha_{h,k}$ and membership functions as shown in Fig. 6. Five local models guarantee good approximation of the state of the real system by the T-S model in the operating range. The following numerical values, with the sampling time 50 ms, were used:

$$\mathbf{A}^1 = \begin{bmatrix} 0.9812 & -0.0105 & 0.1847 & 0 & 0 & 0 \\ 0 & 0.9657 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8780 & 0 & 0 & 0 \\ 0 & 0.0152 & -0.0254 & 0.9908 & -0.1718 & 0 \\ 0 & 0.0004 & 0.1367 & 0.0498 & 0.9957 & 0 \\ 0.0495 & 0.0276 & 0.0047 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{d}_1 = \begin{bmatrix} 0 \\ -0.9326 \\ 0 \end{bmatrix},$$

$$\mathbf{A}^2 = \begin{bmatrix} 0.9814 & -0.0103 & 0.1841 & 0 & 0.0004 & 0 \\ 0 & 0.9657 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8780 & 0 & 0 & 0 \\ 0 & 0.0200 & -0.0254 & 0.9908 & -0.1718 & 0 \\ 0 & 0.0005 & 0.1367 & 0.0498 & 0.9957 & 0 \\ 0.0495 & 0.0274 & 0.0046 & 0 & -0.0010 & 1 \end{bmatrix},$$

$$\mathbf{d}_2 = \begin{bmatrix} 0.1074 \\ -0.9257 \\ 64.1737 \end{bmatrix},$$

$$\mathbf{A}^3 = \begin{bmatrix} 0.9818 & -0.0098 & 0.1830 & 0 & 0.0007 & 0 \\ 0 & 0.9657 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8780 & 0 & 0 & 0 \\ 0 & 0.0405 & -0.0254 & 0.9908 & -0.1718 & 0 \\ 0 & 0.0010 & 0.1367 & 0.0498 & 0.9957 & 0 \\ 0.0495 & 0.0268 & 0.0045 & -0.0001 & -0.0020 & 1 \end{bmatrix},$$

$$\mathbf{d}_3 = \begin{bmatrix} 0.2146 \\ -0.9133 \\ 127.7300 \end{bmatrix},$$

$$\mathbf{A}^4 = \begin{bmatrix} 0.9826 & -0.0090 & 0.1809 & 0 & 0.0010 & 0 \\ 0 & 0.9657 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8780 & 0 & 0 & 0 \\ 0 & 0.0734 & -0.0254 & 0.9908 & -0.1717 & 0 \\ 0 & 0.0018 & 0.1367 & 0.0498 & 0.9957 & 0 \\ 0.0496 & 0.0256 & 0.0044 & -0.0001 & -0.0030 & 1 \end{bmatrix},$$

$$\mathbf{d}_4 = \begin{bmatrix} 0.3199 \\ -0.8895 \\ 189.7399 \end{bmatrix},$$

$$\mathbf{A}^5 = \begin{bmatrix} 0.9837 & -0.0079 & 0.1774 & 0 & 0.0013 & 0 \\ 0 & 0.9657 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8780 & 0 & 0 & 0 \\ 0 & 0.1126 & -0.0254 & 0.9908 & -0.1712 & 0 \\ 0 & 0.0028 & 0.1367 & 0.0498 & 0.9957 & 0 \\ 0.0496 & 0.0239 & 0.0043 & -0.0001 & -0.0039 & 1 \end{bmatrix},$$

$$\mathbf{d}_5 = \begin{bmatrix} 0.4211 \\ -0.8501 \\ 249.3018 \end{bmatrix},$$

Active fault-tolerant control design for Takagi-Sugeno fuzzy systems

$$B^1 = \begin{bmatrix} 0.0047 & -0.0003 \\ 0 & 0.0491 \\ 0.0469 & 0 \\ -0.0005 & 0.0004 \\ 0.0035 & 0 \\ 0.0001 & 0.0007 \end{bmatrix},$$

$$B^2 = \begin{bmatrix} 0.0047 & -0.0003 \\ 0 & 0.0491 \\ 0.0469 & 0 \\ -0.0005 & 0.0005 \\ 0.0035 & 0 \\ 0.0001 & 0.0007 \end{bmatrix},$$

$$B^3 = \begin{bmatrix} 0.0047 & -0.0002 \\ 0 & 0.0491 \\ 0.0469 & 0 \\ -0.0005 & 0.0010 \\ 0.0035 & 0 \\ 0.0001 & 0.0007 \end{bmatrix},$$

$$B^4 = \begin{bmatrix} 0.0046 & -0.0002 \\ 0 & 0.0491 \\ 0.0469 & 0 \\ -0.0005 & 0.0018 \\ 0.0035 & 0 \\ 0.0001 & 0.0006 \end{bmatrix},$$

$$B^5 = \begin{bmatrix} 0.0045 & -0.0002 \\ 0 & 0.0491 \\ 0.0469 & 0 \\ -0.0005 & 0.0028 \\ 0.0035 & 0 \\ 0.0001 & 0.0006 \end{bmatrix},$$

$$L^i = B^i, \forall i \in \{1, \dots, 4\},$$

$$C^1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 896.2360 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$C^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 894.1477 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$C^3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 879.0008 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$C^4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 851.0135 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$C^5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 810.7468 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$u^1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad u^2 = \begin{bmatrix} 0 \\ 0.05 \end{bmatrix},$$

$$u^3 = \begin{bmatrix} 0 \\ 0.1 \end{bmatrix}, \quad u^4 = \begin{bmatrix} 0 \\ 0.15 \end{bmatrix},$$

$$u^5 = \begin{bmatrix} 0 \\ 0.2 \end{bmatrix},$$

The reference input is defined by

$$u_h = u_{k,1} = 0,$$

$$u_v = u_{k,2} = \begin{cases} 0, & k < 4000 \\ 0.05, & 4000 \leq k < 7000 \\ 0.10, & k \geq 7000 \end{cases}$$

The actuators fault scenario, i.e. a decrease of the performance of the two rotors, is described as follows

$$f_{k,1} = \begin{cases} 0, & k < 7000 \\ 0.007 \sin(0.005k), & k \geq 7000 \end{cases}$$

$$f_{k,2} = \begin{cases} 0, & k < 5400 \\ 0.035 + 0.025 \sin(0.01k), & k \geq 5400 \end{cases}$$

The regulator for FTC system was designed with the following parameters $\gamma = 0.05$ and weighting matrices, based on Bryson's rule [26]

$$Q_R = \frac{1}{\gamma^2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R_R = \frac{1}{0.04} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Also due to the high transients of the systems in the initial phase, the FTC system was not enabled until 4000th iteration, but the control input were also at the neutral state (i.e., $u = [0, 0]^T$) at that time period.

Figure 4 presents the results achieved for the proposed FTC strategy (with $\alpha_k = I$). As a result, Fig. 4 clearly shows that the faults can be estimated with a very high accuracy (especially the second fault estimator). The estimator for the first fault presents some deviation of nominal value due to the abrupt changes in the reference input and modelling errors in the consequence of the high non-linearity of the system, particularly its high cross-coupling between the main rotor and the tail rotor. But after about 250 samples, in both faultless and faulty state, estimation achieves very high accuracy, even though the fault is time variable.

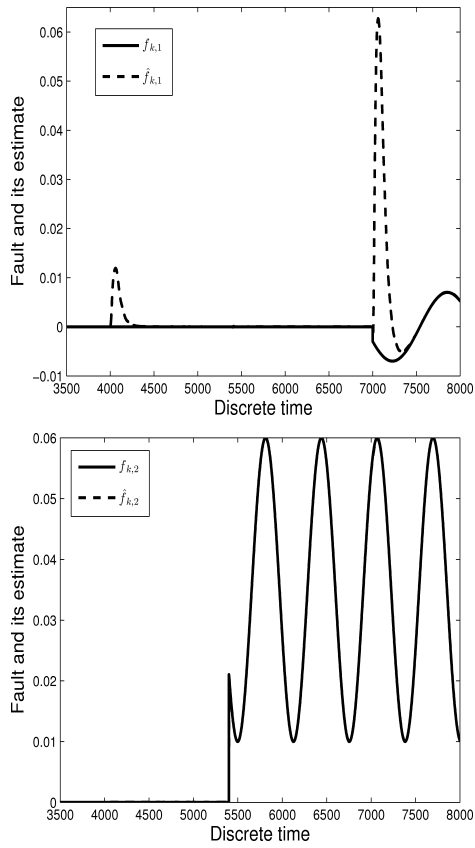


Fig. 4. Faults f_k and their estimates

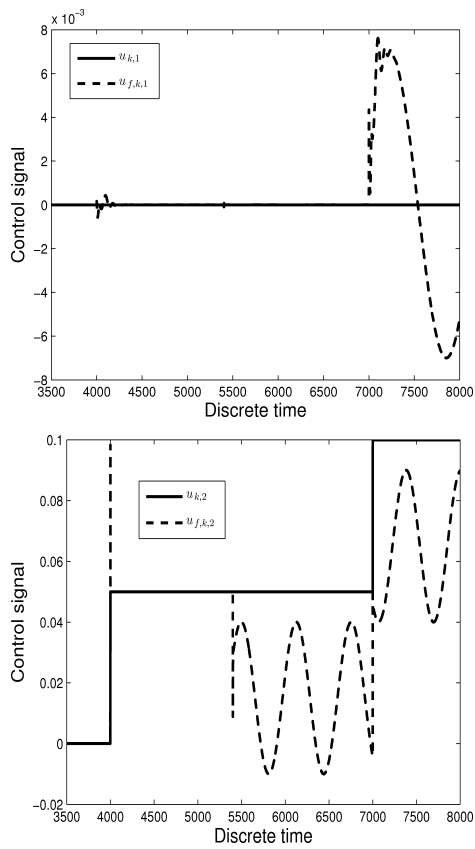


Fig. 5. Trajectories of u_k and $u_{f,k}$

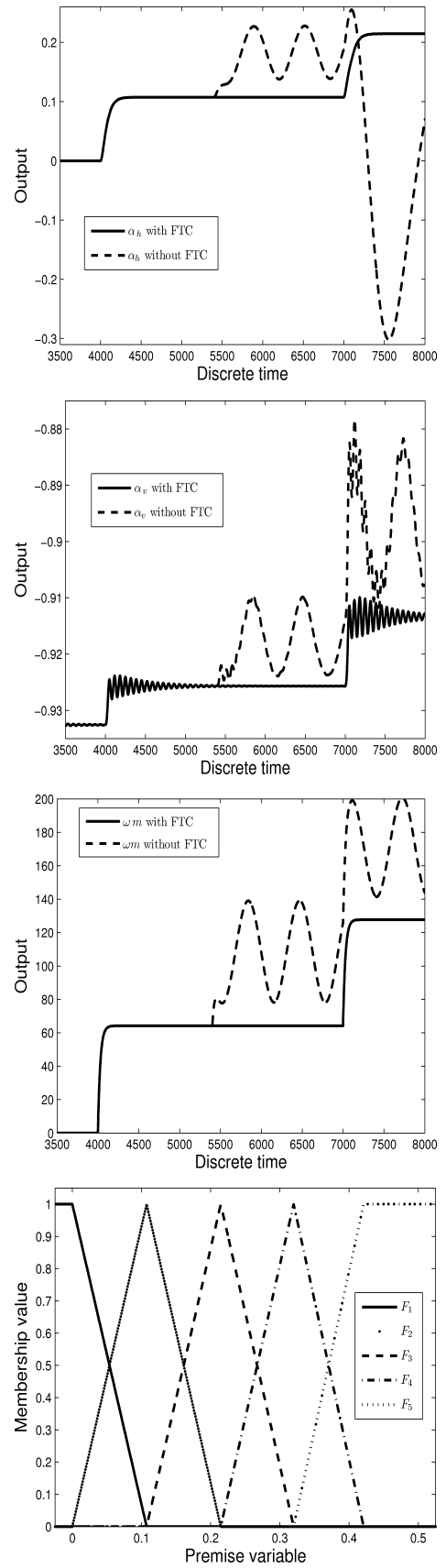


Fig. 6. Outputs of the system $y_{f,k}$ with FTC control disabled and enabled, also fuzzy sets used in Takagi-Sugeno model (lower right figure)

From Fig. 5 it can be observed that $\mathbf{u}_{f,k}$ is close to \mathbf{u}_k until the occurrence of the fault \mathbf{f}_k . After that time the control strategy $\mathbf{u}_{f,k}$ was changed. The only deviations from expected behaviour can be seen in the cases where the faults were overestimated, due to the reasons stated above.

The final conclusion is that the FTC is stabilizing the system with a high performance (Fig. 6), even in the presence of the faults the original trajectories are unchanged. This is because of the proposed control strategy for which $\mathbf{x}_{f,k}$ converges to \mathbf{x}_k and consequently $\mathbf{z}_k = \mathbf{y}_{f,k} - \mathbf{y}_k$ converges to zero. Whereas the system without FTC control deviates from original trajectories quite badly. Especially interesting is the fact that sometimes even small uncompensated fault, as a fault of the tail motor, can lead to catastrophic failure, which can be seen by the trajectories of the pitch angle of beam α_h .

5. Conclusions

In this paper, an active FTC strategy was proposed, which allows for on-line reconfiguration of the control after the occurrence of actuator faults. This approach was developed in the context of the Takagi-Sugeno fuzzy systems. The key contribution of the proposed approach is an integrated FTC design procedure of the fault identification and fault-tolerant control schemes. The design procedure also allows including input constraints into the FTC system. Fault identification is based on the use of an observer. Once the fault have been identified, the FTC controller is implemented as a state feedback controller. This controller is designed such that it can stabilize the faulty plant using Lyapunov theory and LMIs. The designed controller is called a virtual actuator, because it is a block which takes the signal for the broken actuators as input and produces the intended effects on the output of the plant by using the remaining actuators. Also design procedure for the regulator for T-S fuzzy systems was shown, that allows to minimize objective function similar to the LQR controller for the linear systems. A twin-rotor based illustrative example for the non-linear system described by T-S fuzzy models is provided that shows the effectiveness of the proposed FTC approach.

Acknowledgements. The work was financed as a research project with the science funds for years 2007-2010.

REFERENCES

- [1] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*, Springer-Verlag, New York, 2003.
- [2] J. Korbicz, J. Kościelny, Z. Kowalczyk, and W. Cholewa, *Fault Diagnosis. Models, Artificial Intelligence, Applications*, Springer-Verlag, Berlin, 2004.
- [3] M. Witczak, *Modelling and Estimation Strategies for Fault Diagnosis of Non-linear Systems*, Springer-Verlag, Berlin, 2007.
- [4] F. Delebecque, R. Nikoukah, and H. Rubio Scola, "Test signal design for failure detection: A linear programming approach", *Int. J. Applied Mathematics and Computer Science* 13 (4), 515–526 (2003).
- [5] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems", *Proc. IFAC Safeprocess Washington* 1, 65–276 (2003).
- [6] J. Chen, R. Patton, and Z. Chen, "An LMI approach to fault-tolerant control of uncertain systems", *Proc. IEEE Conf. on Decision and Control* 1, 175–180 (1998).
- [7] Y. Liang, D. Liaw, and T. Lee, "Reliable control of nonlinear systems", *IEEE Trans. Automatic Control* 45, 706–710 (2000).
- [8] F. Liao, J. Wang, and G. Yang, "Reliable robust flight tracking control: an lmi approach", *IEEE Trans. Control Syst. Techn.* 10, 76–89 (2002).
- [9] Z. Qu, C. Ihlefeld, J. Yufang, and A. Saengdeejing, "Robust fault-tolerant self-recovering control of nonlinear uncertain systems", *Automatica* 39, 1763–1771 (2003).
- [10] H. Li, Q. Zhao, and Z. Yang, "Reliability modeling of fault tolerant control systems", *Int. J. Applied Mathematics and Computer Science* 17 (4), 491–504 (2007).
- [11] M. Witczak, "Advances in model-based fault diagnosis with evolutionary algorithms and neural networks", *Int. J. Applied Mathematics and Computer Science* 16 (1), 85–99 (2006).
- [12] S. Kanev and M. Verhaegen, "Reconfigurable robust fault-tolerant control and state estimation", *Proc. 15th IFAC World Congress* 1, CD-ROM (2002).
- [13] J. Maciejowski, *Predictive Control with Constraints*, Prentice Hall, New York, 2002.
- [14] J. Lunze and T. Steffen, "Control reconfiguration after actuator failures using disturbance decoupling methods", *IEEE Trans. on Automatic Control* 51 (10), 1590–1601 (2006).
- [15] J.H. Richter and J. Lunze, "H-infinity-based virtual actuator synthesis for optimal trajectory recovery", *Proc. 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes* 1, 1587–1592 (2009).
- [16] J. H. Richter, S. Weiland, W. P. M. H. Heemels, and J. Lunze, "Decoupling-based reconfigurable control of linear systems after actuator faults", *Proc. 10th Eur. Control Conf.* 1, 2512–2517 (2009).
- [17] M. Witczak, L. Dziekan, V. Puig, and J. Korbicz, "Design of a fault-tolerant control scheme for takagi-sugeno fuzzy systems", *Control and Automation, 16th Mediterranean Conf.* 1, 280–285 (2008).
- [18] Q. Rong and G. Irwin, "LMI-Based controller design for discrete polytopic LPV systems", *Proc. European Control Conf.* 1, CD-ROM (2003).
- [19] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control", *IEEE Trans. Systems, Man and Cybernetics* 15 (1), 116–132 (1985).
- [20] K. Tanaka and H. O. Wang, *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*, Wiley-Interscience, London, 2001.
- [21] S. Hui and S. Zak, "Observer design for systems with unknown input", *Int. J. Applied Mathematics and Computer Science* 15 (4), 431–446 (2005).
- [22] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory, Studies in Applied Mathematics*, vol. 15, SIAM, New York, 1994.
- [23] A. Rahideh and M.H. Shaheed, "Mathematical dynamic modelling of a twin-rotor multiple input-multiple output system", *Proc. Institution of Mechanical Engineers, Part I, J. Systems and Control Engineering* 227, 89–101 (2007).

Ł. Dziekan, M. Witczak, and J. Korbicz

- [24] Feedback Instruments Limited, Crowborough, *Twin Rotor MIMO System Advanced Teaching Manual 1*, London, 1998.
- [25] S. Montes de Oca, V. Puig, M. Witczak, and J. Quevedo, "Fault-tolerant control of a two-degree of freedom helicopter using LPV techniques", *Proc. 16th Mediterranean Conference on Control and Automation 1*, 1204–1209 (2008).
- [26] G. Franklin, J. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, Prentice Hall, Upper Saddle River, 2002.