Expert Systems with Applications 39 (2012) 10630–10638

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Fault tolerant control using a fuzzy predictive approach

L.F. Mendonça [a,b,*], J.M.C. Sousa [a], J.M.G. Sá da Costa [a]

[a] Instituto Superior Técnico, Technical University of Lisbon, Dept. of Mechanical Engineering, CIS/IDMEC-LAETA, Av. Rovisco Pais, 1049-001 Lisbon, Portugal
[b] Escola Superior Náutica Infante D. Henrique, Dept. of Marine Engineering, Av. Eng. Bonneville Franco, 2770-058 Lisbon, Portugal

ARTICLE INFO

Keywords:
Fault tolerant control
Fault detection
Fault isolation
Fuzzy decision making
Fuzzy optimization
Model predictive control

ABSTRACT

This paper proposes the application of fault-tolerant control (FTC) using fuzzy predictive control. The FTC approach is based on two steps, fault detection and isolation (FDI) and fault accommodation. The fault detection is performed by a model-based approach using fuzzy modeling and fault isolation uses a fuzzy decision making approach. The information obtained on the FDI step is used to select the model to be used in fault accommodation, in a model predictive control (MPC) scheme. The fault accommodation is performed with one fuzzy model for each identified fault. The FTC scheme is used to accommodate the faults of two systems a container gantry crane and three tank benchmark system. The fuzzy FTC scheme proposed in this paper was able to detect, isolate and accommodate correctly the considered faults of both systems.

1. Introduction

Complexity of technical processes is increasing continuously. One consequence of this increase is that safety the reliability become important system requirements. The complexity of these systems increases when the fault probability increases. This is the main reason why control systems include automatic supervision of process control to detect and isolate faults as early as possible and to perform fault accommodation.

FTC can be performed by passive methods or by active methods. Passive methods make use of robust control techniques to ensure that a closed-loop system remains insensitive to certain faults using constant controller parameters and without use of on-line fault information (Zhang & Jiang, 1985). In active methods, a new control system is redesigned using desirable properties of performance and robustness in the system without faults. Active fault-tolerant controllers are generally variable in their structure. Active approaches are divided into two main types of methods: projection based methods and on-line automatic controller redesign methods (Patton, 1997). The reconfiguration includes the selection of a new control configuration where alternative input and output signals are used (Blanke, Kinnaert, Lunze, & Staroswiecki, 2003). On the other hand, fault accommodation adapts the controller parameters to the dynamical properties of the faulty plant. A simple but well established way of fault accommodation is based on predesigned controllers, each of which selected off-line for a specific fault (Blanke et al., 2003). Fault accommodation involves the detection and isolation of faults, and taking appropriate control actions that eliminate or reduce the effect of faults and maintains the control.

The use of model predictive control to deal with fault accommodation is relatively natural and straightforward, considering the representation of both faults and control objectives (Maciejowski & Jones, 2003). MPC with additional flexibility is obtained using fuzzy sets in the objective function. The fuzzy sets theory provides ways of representing and dealing with flexible or soft criteria. The fuzzy objective function used in MPC includes goals and the constraints. The optimal trade-off amongst fuzzy goals and fuzzy constraints is determined by maximizing simultaneously the satisfaction of the optimization goals and the constraints (Sousa & Kaymak, 2001).

The FDI approach used in this paper uses one fuzzy model representing the normal state of the system and one fuzzy model for each fault that can occur in a given system. The faults are detected and isolated based on these fuzzy models (Mendonça, Sousa, & Sá da Costa, 2009). A fuzzy decision making (FDM) approach is used to isolate the faults. When a fault is isolated, fault accommodation is performed by using the respective faulty model. This paper proposes a fault tolerant control scheme, where the faulty model is used in a fuzzy MPC scheme. This control technique can be a highly efficient approach to perform fault accommodation (Gopinathan, Boskovic, Mehra, & Rago, 1998).

This paper is organized as follows. Next section presents fault tolerant control. The architecture for fault tolerant control proposed in this paper is presented in Section 3. Predictive control is presented in Section 4. This paper presents two application examples, the container gantry crane in Section 5 and the three tank benchmark process in Section 6. Finally, some conclusions are drawn in Section 7.

* Corresponding author at: Instituto Superior Técnico, Technical University of Lisbon, Dept. of Mechanical Engineering, CIS/IDMEC-LAETA, Av. Rovisco Pais, 1049-001 Lisbon, Portugal.

E-mail address: mendonca@dem.ist.utl.pt (L.F. Mendonça).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.02.094

2. Fault tolerant control

FTC can be motivated by different purposes, as the improvement of safety and efficiency in industrial processes. The main design challenges of FTC are: the number of possible faults and their diagnosability; the system reconfigurability, and the global stability of the system (Blanke, Frei, Kraus, Patton, & Staroswiecki, 2000).

2.1. State-of-art

Fault tolerant control can be classified into two types: passive approaches (Chen & Patton, 1999) and active approaches (Steffen, 2005).

Active fault tolerant control approach uses the FDI information to make the on-line controller reconfiguration or model selection (Chandler, Pachter, & Mears, 1995). In Patton and Klinkhieo (2009), a new approach to fault compensation for FTC using fault estimation is presented, where the faults acting in a dynamic system are estimated and compensated within an adaptive control scheme with required stability and performance robustness. The development of a novel FTC design method is presented in Guenaba, Webera, Theilliola, and Zhangb (2011), which incorporates both reliability and dynamic performance of the faulty system in the design of a FTC. Another possible approach is to use all the information given by FDI to improve the ability of on-line controller reconfiguration (Polycarpou & Helmicki, 1995).

The fuzzy logic approach in FTC is used in Lopez-Toribio, Patton, and Daley (2000) where Takagi–Sugeno (TS) fuzzy models are used in fault tolerant control of non-linear systems. In Ichtev, Hellendoorn, Babuška, and Mollov (2002), multiple TS fuzzy models are used in fault tolerant model predictive control. When MPC is used in FTC, some faults can be accommodate modifying the constraints in the MPC problem definition (Maciejowski, 2002). The use of MPC increases the degree of fault tolerance under certain conditions, when the fault is not detected. Thus, MPC in fault tolerant control provides a suitable implementation architecture and increases the system capability to accommodate the faults.

In order to overcome the limitations of conventional control, new controllers are being used which are capable of tolerating component malfunctions. Complex control applications require a capability for accommodating faults in the controlled industrial process. Fault accommodation involves the detection and isolation of faults, and taking appropriate control actions that eliminate or reduce the effect of the faults and maintains the control. The method used in this paper is an active approach.

2.2. FDI in fault tolerant control

A system that includes the capacity of detecting, isolating and identifying faults is called a fault diagnosis and isolation system (Chen & Patton, 1999). During the years, many research has been carried out using analytical approaches, based on quantitative models. The idea is to generate signals that reflect inconsistencies between normal and faulty system operation, and detect and isolate the faults. Such signals, the residuals, are usually generated using analytical approaches, such as observers, parameter estimation or parity equations. Early detection and isolation of abrupt and incipient faults can be achieved using a model-based approach, which processes all measured variables, using either qualitative or quantitative modeling. The use of fuzzy logic for fault detection and isolation in industrial processes is presented in Koscielny and Syfertm (2003). Optimized fuzzy models have been used with success in model based FDI (Mendonça et al., 2009).

The use of FDI in fault tolerant control is very important in the active way of achieving fault-tolerance, by detect and isolate the faults. After the fault indication by FDI, the system can then be reconfigured or restructured. In some cases, a pre-calculated controller will be activated, or the parameters of the controller will be changed according the real time diagnostic provided by the FDI. Next section presents the architecture of FTC proposed in this paper.

3. Architecture for fault tolerant control

This paper proposes a simple architecture for fault tolerant control. This approach is based on two steps: the first performs fault detection and isolation, and the second performs fault accommodation. The two steps are depicted in Fig. 1, and are denoted as FDI and FTC.

3.1. Fault detection and isolation

The fault detection and isolation approach is showed in Fig. 1 in the block called FDI. In this FDI approach, the multidimensional input, $\mathbf{u}$, of the system enters both the process and a model (observer) in normal operation. The vector of residuals $\boldsymbol{\varepsilon}$ is defined as

$$\boldsymbol{\varepsilon} = \mathbf{y} - \hat{\mathbf{y}}, \tag{1}$$

where $\mathbf{y}$ is the output of the system and $\hat{\mathbf{y}}$ is the output of the model in normal operation. When any component of ε is bigger than a certain threshold, the system detects faults. In this case, n observers (models), one for each fault, are activated, and n vectors of residuals are computed. Each residual i, with $i = 1,\ldots,n$, is computed as

$$\boldsymbol{\varepsilon}_{\mathbf{F}_i} = \mathbf{y} - \hat{\mathbf{y}}_{\mathbf{F}_i}, \tag{2}$$

where $\hat{\mathbf{y}}_{\mathrm{F}_i}$ is the output of the observer for the fault i. The residuals $\varepsilon_{\mathrm{F}_1},\ldots,\varepsilon_{\mathrm{F}_n}$ are evaluated, and the fault or faults detected are the outputs of the FDI system. The fault isolation is performed by evaluating *fuzzy decision factors*, which are built based on residuals. The fuzzy fault isolation used in this paper is based on fuzzy decision making (FDM) (Mendonça et al., 2009; Mendonça, Sousa, & Sá da Costa, 2006b). In this approach, a membership function $\mu_{\varepsilon_{ij}}$ is derived for each residual ε_{ij}. The membership functions used in this paper are trapezoidal because they revealed to be the most appropriate to describe the residuals in a simple and effective way. The membership functions spread is obtained experimentally based on the maximum and minimum variations of the residuals. The core of the membership functions indicates the possible isolation of a fault, i.e. if ε_{ij} is zero, then the membership degree $\mu_{\varepsilon_{ij}}$ should be one. The core is also determined experimentally and is a small interval around zero in order to accommodate process noise, disturbances and model-plant mismatches. Note that this method to derive membership functions is common in various fuzzy approaches (Mendonça, Sousa, & Sá da Costa, 2004). The m membership functions $\mu_{\varepsilon_{i1}},\ldots,\mu_{\varepsilon_{im}}$ must be aggregated using a conjunction operator, which assures that a fault is isolated only when all the residuals ε_{ij} are close to zero. The aggregation can be given by

$$\gamma_i = t(\mu_{\varepsilon_{i1}},\ldots,\mu_{\varepsilon_{im}}), \tag{3}$$

where t is a triangular norm, as e.g. the minimum operator. An example of γ_i for two outputs is shown in Fig. 2. Let $\gamma_i(k) \in [0,1]$, $i = 1,\ldots,n$, be called a *fuzzy decision factor*. These values are computed at each time instant k. A vector of fuzzy decision factors can be computed as:

$$\Gamma(k) = [\gamma_1(k)\ \gamma_2(k) \cdots \gamma_n(k)], \tag{4}$$

i.e., one fuzzy decision factor for each fault. A fuzzy decision factor $\gamma_i(k)$ is high only if all the residuals are close to zero.

In order to isolate a fault i, the value of $\gamma_i(k)$ must be higher than a *threshold* T, which must be close to one. Note that the threshold T

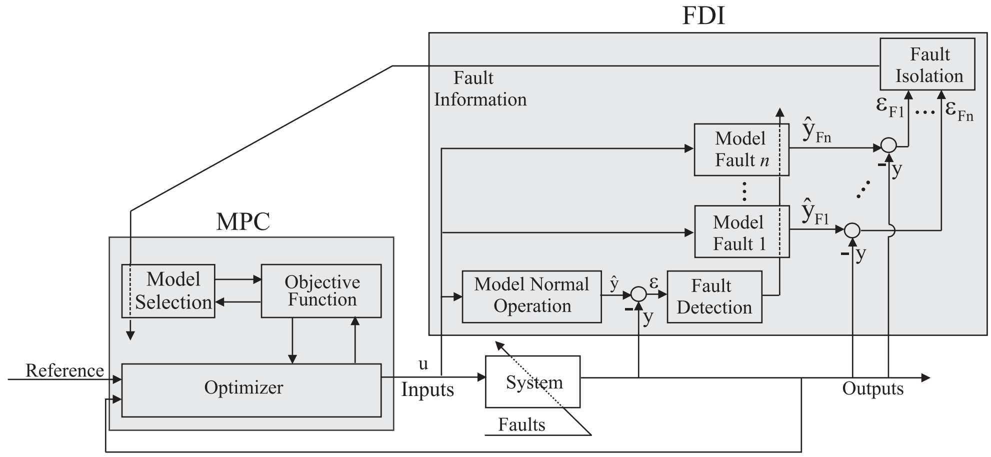

Fig. 1. Fault tolerant control scheme proposed in this paper.

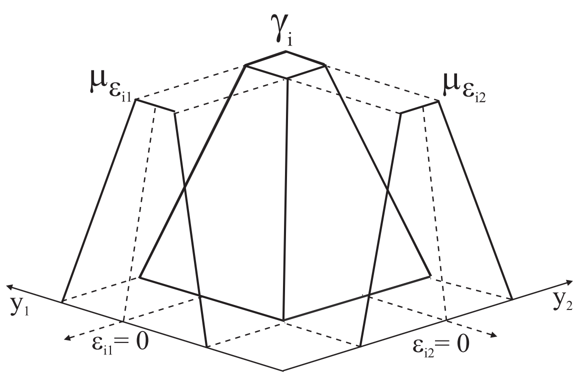

Fig. 2. Residual evaluation of fuzzy model of fault *i*.

is equal for all the faults, because the fuzzy decision factors are already normalized in the interval [0,1]. The threshold is obtained experimentally and defines the regions of fault and no fault. Note that several $\gamma_i(k)$ can be above the threshold at a certain time k. Therefore, a fault i is isolated only when the remaining faults are below T. However, even if only one fault is above the threshold at a certain time instant, this can occur due to noise or model errors. Therefore, our approach considers that a fault $i \in \{1,\ldots,n\}$ is only isolated when

$$\begin{cases} \gamma_i > T \text{ and} \\ \gamma_l < T \ \forall l \neq i, \quad \text{for } t_k \text{ consecutive instants}, \end{cases} \tag{5}$$

i.e., when γ_i is above the threshold T and the remaining γ_l decision factors are below the same threshold for t_k consecutive time instants. The fuzzy isolation scheme used in this paper is summarized in the next algorithm:

(1) Build the membership functions $\mu_{\varepsilon_{ij}}$ for each residual i, $i = 1,\ldots,n$, and for each output j, with $j = 1,\ldots,m$, based on simulated or experimental values of the residuals ε_{ij}.
(2) Define the threshold value T and the number of consecutive time instants to isolate a fault t_k.
(3) REPEAT at each time instant k,
  (a) Compute the residuals ε;
  (b) Calculate all the fuzzy decision factors γ_i by aggregating the residuals using (3).
  (c) Compare the fuzzy decision factors with the threshold T.
(4) UNTIL the conditions defined in (5) are satisfied.

3.2. Fuzzy fault tolerant control

The FTC structure proposed in this paper was shown in Fig. 1. The presented architecture uses FDI and MPC. The FDI approach was presented in Section 3.1. The MPC is very useful in FTC, because it allows a different control specification for the faulty models, in order to have minimal losses when the system is working in a faulty mode. Furthermore, the control action can take into account a time interval (prediction horizon). Also the receding horizon principle allows at each time instant to assess the situation by taking into account any change in the fault status to apply the best control action (Ichtev et al., 2002).

The FTC scheme proposed in this paper uses a multiple model selection approach, where a fuzzy model for the process running in normal operation and one model for each one of the faults are used. The use of fuzzy set theory in MPC support the FTC proposed approach because sometimes, it is impossible to model nonlinear systems by analytical equations. The use of fuzzy models increase the capability of proposed FTC architecture to work with systems without complete information and noisy. The key advantage of fuzzy logic is that it enables the system behavior to be described by "if-then" relations.

MPC has also been demonstrated as a highly efficient approach to failure accommodation (Gopinathan et al., 1998). The fault accommodation means to adapt the controller parameters to the dynamical properties of the faulty plant. A simple but well established way of fault accommodation is based on predesigned controllers, each of which has been selected off-line for a specific fault (Blanke et al., 2003). Next section presents some characteristics of MPC.

4. Predictive control

Predictive control is probably the advanced control scheme most frequently used in industry. Its advantages are the use of an objective function and the ability to control complicated processes. Predictive control is closely related to decision making. The objective function can be seen as the simultaneous satisfaction of (soft) goals and (soft) constraints in multistage decision making. This technique has been applied to control by several authors (Sousa & Kaymak, 2002). When fuzzy criteria is used in the objective function, fuzzy optimization is the most obvious technique to deal with the optimization problem in fuzzy predictive control. Next section resents a brief description of classical predictive control and subsequently, fuzzy predictive control is presented.

4.1. Classical objective functions

In predictive control of multivariable systems, the output values $\hat{\mathbf{y}}(k+i),\ i=1,\ldots,H_p$, depend on the states of the process at the current time k and on the future control signals $\mathbf{u}(k+j)$, $j=1,\ldots,H_c$, where H_c is the control horizon. For multivariable systems the objective function can be represented by

$$J = \mathbf{e}^T R \mathbf{e} + \Delta\mathbf{u}^T Q \Delta\mathbf{u}, \tag{6}$$

where the first term in (6) accounts for the minimization of the output errors, the second term represents the minimization of the control effort, and R and Q are weighting matrices. Note that these parameters have two functions: they normalize the different outputs and inputs of the system, and vary the importance of the two different terms in the objective function (6) over the time steps.

4.2. Fuzzy objective functions

One of the main issues in MPC is the optimization technique applied to derive the control actions. When fuzzy criteria is used in the objective function, the criteria has some flexibility that can be exploited for improving the optimization objective (Sousa & Kaymak, 2002).

Predictive control using fuzzy goals and fuzzy constraints can be defined as a fuzzy decision making problem. Let G_i, with $i=1,\ldots,q$, be a fuzzy goal characterized by its membership function μ_{G_i}, which is a mapping from the space of the goal G_i to the interval $[0,1]$. Let also C_l, $l=1,\ldots,r$ be a fuzzy constraint characterized by its membership function μ_{C_l}, mapping the space of the constraint C_l to the same interval $[0,1]$. The fuzzy goals G_i and the fuzzy constraints C_l can be defined for the domain of the control actions, system outputs, state variables or for any other convenient domain. Each fuzzy goal G_i and each fuzzy constraint C_l constitute a *decision criterion* ζ_j, $j=1,\ldots,T$, where $T=q+r$ is the total number of goals and constraints. Each criterion is defined in the domain Φ_j, $j=1,\ldots,T$, which can be any of the various domains used in control. In order to solve the optimization problem in low computational time, the optimization problem is defined in a discrete control space with a finite number of control alternatives.

Fuzzy criteria are aggregated in the control environment. Assume that a policy π is defined as a sequence of control actions for the entire prediction horizon in MPC, H_p:

$$\pi = \mathbf{u}(k),\ldots,\mathbf{u}(k+H_p-1), \quad \pi\in\Omega, \tag{7}$$

where the control actions belong to a set of alternatives Ω. In the general case, all the criteria must be applied at each time step i, with $i=1,\ldots,H_p$. Thus a criterion ζ_{ij} denotes that the criterion j is considered at time step $k+i$, with $i=1,\ldots,H_p$, and $j=1,\ldots,T$. Further, let $\mu_{\zeta_{ij}}$ denote the membership value that represents the satisfaction of this decision criterion after applying the control actions $\mathbf{u}(k+i)$. The total number of decision criteria for the decision problem is thus given by $\widetilde{T} = T\cdot H_p$. The confluence of goals and constraints is performed by aggregating the membership values $\mu_{\zeta_{ij}}$. The membership value μ_π for the control sequence π is obtained using the aggregation operator $\circledast$ to combine the decision criteria, i.e.

$$\begin{aligned}&\mu_\pi = \mu_{\zeta_{11}} \circledast \cdots \circledast \mu_{\zeta_{1q}} \circledast\\ &\mu_{\zeta_{1(q+1)}} \circledast \cdots \circledast \mu_{\zeta_{1T}} \circledast\\ &\vdots\\ &\mu_{\zeta_{H_p(q+1)}} \circledast \cdots \circledast \mu_{\zeta_{H_pT}}.\end{aligned} \tag{8}$$

In this equation, the aggregation operator $\circledast$ combines the goals and the constraints. Various types of aggregation operations can be used as decision functions for expressing different decision strategies using the well-known properties of these operators. Parametric triangular norms can generalize a large number of t-norms, and can control the degree of compensation between the different goals and constraints. Usually, parametric t-norms depend only on one parameter, which makes them easier to tune when compared to weighted t-norms. On the other hand, they are not so general as the weighted approaches (Kaymak & Sousa, 2003). The translation of each goal and each constraint for a given policy π to a membership value avoids the specification of the criteria in a large dimensional space. The decision criteria in (8) should be satisfied as much as possible, which corresponds to the maximum value of the overall decision. Thus, the optimal sequence of control actions π^* is found by the maximization of μ_π:

$$\pi^* = \underset{\mathbf{u}(k),\ldots,\mathbf{u}(k+H_{p-1})}{\arg\max}\ \mu_\pi. \tag{9}$$

Because the membership functions for the fuzzy criteria can have an arbitrary shape, and because of the nonlinearity of the decision function, the optimization problem (9) is usually non-convex. To deal with the increasing complexity of the optimization problem, a proper optimization algorithm must be chosen. One possibility is to use, for instance, a branch-and-bound algorithm (Mendonça et al., 2004).

This paper uses one approach where preference for different constraints and goals can be specified by the decision-maker and the difference in the preference for the constraints is represented by a set of associated weight factors as proposed in Mendonça, Sousa, Kaymak, and Sá da Costa (2006). Next section presents the heuristic used to obtain the weight factors.

4.3. Weight selection in fuzzy aggregation

The weight factors represent the relative importance of various constraints and objectives with respect to one another. The general assumption is that the higher the weight of a particular constraint, the larger its importance on the aggregation result. Hence, the final optimization result will be closer to the more important constraints. If the objective is more important, the constraints will be relaxed to a larger degree in order to increase the objective function. The user can specify preferences regarding the outcome of the optimization by changing the weight factors (Mendonça, Sousa, & da Costa, 2006a). Knowing how to combine the different weights in the weighted aggregation function, it is now very important to choose properly the values of the weights for each criterion. The used algorithm is summarized as follows:

(1) Initialize all the weight factors to one, and evaluate the control performance using the corresponding objective function.
(2) Decrease each of the $\widetilde{T}$ weight factors to 0.5 one by one. Evaluate the performance, and order the criteria, where the first is the one that improved the performance of the system most. When the number of criteria $\widetilde{T}$ is very high, a simplification can be made. In this case, reduce simultaneously a certain criterion for the entire prediction horizon H_p. This is similar to evaluate simultaneously each column in (8). The number of iterations is then reduced from $\widetilde{T} = T\times H_p$ to T. Thus, instead of evaluating each weight associated with the criterion ζ_{ij}, the same weight is assumed for the criterion ζ_j, i.e. the criterion is considered constant for the entire prediction horizon.
(3) For each criterion, ζ_{ij} or ζ_j depending on the choice in Step 2, reduce the weight factor to 0.25 and check if the control performance is better. If this is the case, reduce further the weight to 0.125. The weight that yields the best performance is chosen as the weight factor for that criterion.
(4) When all the criteria have been evaluated, the best combination of weights is determined, and should be used for the system.

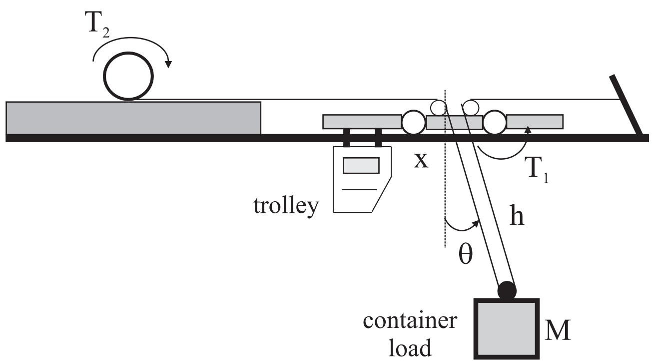

Fig. 3. Schematic picture of the container gantry crane.

Next sections present the application of the FTC scheme proposed in this paper to a container gantry crane system and to a three tank benchmark system. These processes are complex and difficult to control. Therefore, they are two good systems to test the effectiveness of the proposed approach.

5. Application example: container gantry crane

5.1. Description of the process

The approach presented in this paper is applied to a container gantry crane, which is shown in Fig. 3. This process consists of a bridge girder on portal legs from which a trolley system is suspended. The trolley can travel along the bridge girder that stretches over the container ship and part of the quay for loading and unloading the ship. A hoisting mechanism, consisting of a spreader suspended from the trolley by means of hoisting cables, is used for grabbing and hoisting the container. The control goal is to position the trolley at a desired horizontal location x, with a rope length h, while the swing θ of the load is damped so that the container can be positioned accurately, see Fig. 3. The simulation model of the gantry crane is implemented using the Lagrangian of the system (Sakawa & Shindo, 1982), considering also the models of the electric motors, and the viscous friction. The parameters of the model are from one crane at the port of Rotterdam. The simulation presented uses the prediction horizon equal to 5, and the control horizon equal to 3. The input of the crane motors is in the interval [−460 and 460] V. The trolley can reach a maximum velocity of 3.2 ms^{-1} for a maximum load of 53 ton. The crane construction is assumed to be stiff, and the maximum acceleration is 0.8 ms^{-2}.

The predictive control structure applied includes measurement noise and system disturbances, which have values similar to the ones in the real system.

5.2. FDI results

The FTC scheme proposed in this paper, which was presented in Fig. 1, was applied to the model of a container gantry crane from the port of Rotterdam. The faults considered in this paper are, $F1$ length of the rope sensor fault and $F2$ horizontal location sensor fault. Fault $F1$ is a fault in the sensor measuring the length of the rope. The result of this fault is an increase in the readings of the length of the rope. Further, the other fault, $F2$, is a fault in the sensor measuring the horizontal location of the trolley. This fault is simulated by an increase in the readings of the position sensor.

In order to measure the accuracy of the fuzzy models, this paper uses the variance accounted for (VAF) and the root mean square (RMS). The identification data used to build the valve model in normal operation contains 2000 samples. The same number of samples was used to identify each fault used in the simulation. A fuzzy model was identified for the model in normal operation. This model has two inputs, which are the torque of the motors T_1 and T_2 in Fig. 3. The outputs of the model are the horizontal position x, the length of the rope h and the swing angle θ.

Table 1
Accuracy of fuzzy models for the process with and without faults.

Faults	VAF			RMS		
	x	h	θ	x	h	θ
No fault	98.6	97.1	52.7	3.3	1.1	3.8
$F1$	91.8	95.1	45.0	1.8	3.5	3.1
$F2$	99.9	98.8	65.5	0.7	3.1	2.3

Table 1 presents the obtained results when the process is without fault and with faults. The presented performance values are obtained for each one of output variables, x, h and θ. In general, the fuzzy models present good accuracy when the system is with or without faults. The swing angle has some oscillation and consequently its VAF is small.

The FDI step is made considering the scheme presented in Fig. 1. The time detection (t_{td}) and the time isolation (t_{ti}) of the faults are presented in Table 2. The two faults $F1$ and $F2$ are correctly detected and isolated. The fault isolation time is determined when the residuals are close to zero.

Table 2
Detection and isolation performance.

Faults	$t_{td}(s)$	$t_{ti}(s)$
$F1$	36	39
$F2$	37	40

5.3. Fuzzy FTC results

Fault accommodation was performed initially using the classical MPC. Fuzzy FTC is also used to accommodate the container gantry crane faults. Sections 5.3.1 and 5.3.2 present the obtained results for fault $F1$ and for fault $F2$, respectively.

5.3.1. Fuzzy FTC for fault F1

Fault $F1$ was simulated to start 35 s after the beginning of the trajectory. When the weighted criteria is applied, the difference in the preference for the constraints is represented by a set of associated weight factors. The vector of the weights is given by $w^T = [w_x w_h w_\theta]$, where w_x, w_h and w_θ are the weights for the horizontal position x, length of the rope h and the swing angle θ, respectively. Table 3 shows the control results when the fault $F1$ occurs, using the Yager t-norm with the weight selection algorithm proposed in Section 4.3. The control results consist of the normalized sum squared error between the references and the outputs of the system after the fault isolation time are presented in Table 2. The error using the classical predictive controller with weights $R_x = 1.0$, $R_h = 1.0$ and $R_\theta = 1.0$ is taken as 1 (100%), and it serves as the normalization to be compared with the errors using Yager t-norm with the weights presented in Table 3. The absolute error values obtained using the classical controller are $e_h = 8.57$ for the rope length, $e_x = 2.31$ for the horizontal location, and $e_\theta = 0.0028$ for the swing angle. The best result is obtained at Step 4 of weighted fuzzy predictive control. The results of the simulations using classical MPC controller and Step 4 in Table 3 are depicted in Fig. 4. Both controllers present good control performance for the three controlled variables. However the position error and the swing angle decrease when weighted fuzzy MPC is applied. The best value for the swing angle, which is the most

Table 3
Normalized errors using Yager t-norm with various weight combinations for the fault F1.

	R_x	R_h	R_θ	e_x	e_h	e_θ
Classical	1.0	1.0	1.0	1	1	1
Fuzzy	w_x	w_h	w_θ			
1	1.0	1.0	1.0	0.52	0.95	0.5
2	1.0	1.0	0.5	1.04	1.02	1.32
3	1.0	0.5	1.0	1.05	0.95	0.96
4	**0.5**	**1.0**	**1.0**	**0.28**	**0.94**	**0.39**
5	0.25	1.0	1.0	0.4	0.95	0.5
6	0.5	0.5	1.0	0.66	0.94	0.42
7	0.5	1.0	0.5	0.51	0.97	0.46

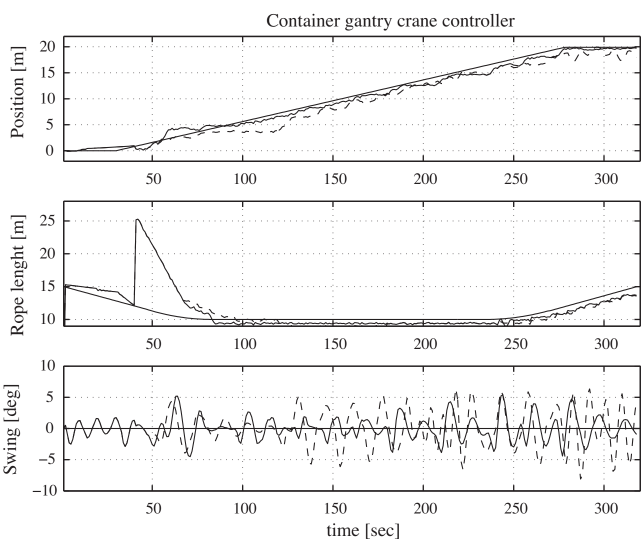

Fig. 4. Fault $F1$ accommodation. Solid — fuzzy, $w_x = 0.5$, $w_h = 1.0$, $w_\theta = 1.0$ and dashed — classical, $R_x = 1.0$, $R_h = 1.0$, $R_\theta = 1.0$.

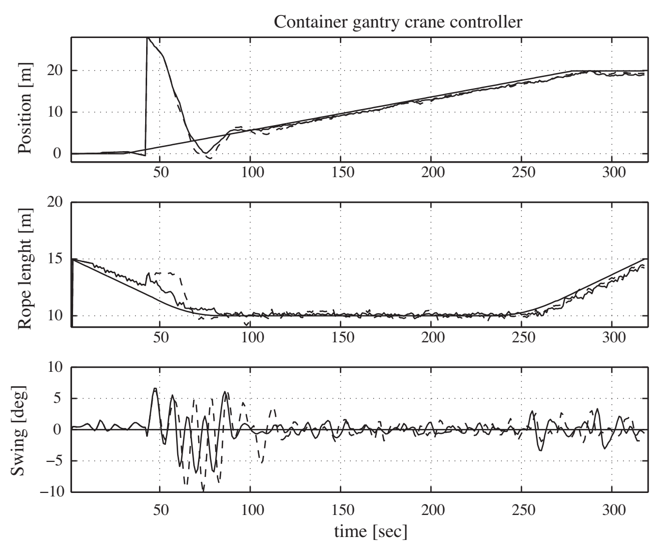

Fig. 5. Fault $F2$ accommodation. Solid — Fuzzy, $w_x = 1.0$, $w_h = 0.5$, $w_\theta = 0.25$ and dashed — classical, $R_x = 20.0$, $R_h = 3.0$, $R_\theta = 3.0$.

sensitive controlled variable, is obtained with the Yager t-norm operator using the weights $w_x = 0.5$, $w_h = 1.0$ and $w_\theta = 1.0$, which has a maximum absolute value of 5.3°. When classical MPC is used the maximum absolute value of the swing angle is 8.1°. This simulation example shows clearly that the best results are obtained with fuzzy objective function with different values for the weights. The fuzzy FTC scheme proposed in this paper was able to detect, isolate and accommodate correctly the fault $F1$. The behavior of fault $F1$ can be observed in both the length of the rope and in the swing angle. It has almost no effect on the horizontal position. Despite the large intensity of fault $F1$, as it is clearly seen in Fig. 4, the final objective of this process, i.e. the ship loading and unloading, is made correctly.

5.3.2. Fuzzy FTC for fault F2

Fault $F2$ was simulated to start 35 s after the beginning of the trajectory. Table 4 shows the control results when the fault $F2$ occurs, using the Yager t-norm with the various weight factors selected from algorithm presented in Section 4.3. Also with fault $F2$ the error using classical predictive control, presented in Table 4 is taken as 1 (100%), and it serves as the normalization. The best results obtained with the classical controller are $e_h = 0.58$ for the rope length, $e_x = 29.98$ for the horizontal location, and $e_\theta = 0.0017$ for the swing angle, with weights $R_x = 20.0$, $R_h = 3.0$ and $R_\theta = 3.0$. The best result obtained when weighted optimization is used is obtained at Step 8.

Table 4
Normalized errors using Yager t-norm with various weight combinations for the fault F2.

	R_x	R_h	R_θ	e_x	e_h	e_θ
Classical	20.0	3.0	3.0	1	1	1
Fuzzy	w_x	w_h	w_θ			
1	1.0	1.0	1.0	1.07	0.7	0.76
2	1.0	1.0	0.5	1.02	0.68	0.64
3	1.0	0.5	1.0	1.36	0.39	1.11
4	0.5	1.0	1.0	0.98	0.89	0.82
5	1.0	1.0	0.25	1.07	0.36	0.64
6	1.0	1.0	0.125	1.21	0.27	0.94
7	0.5	1.0	0.25	1.0	0.74	1.11
8	**1.0**	**0.5**	**0.25**	**0.99**	**0.36**	**0.58**
9	1.0	0.25	0.25	1.53	0.29	1.05

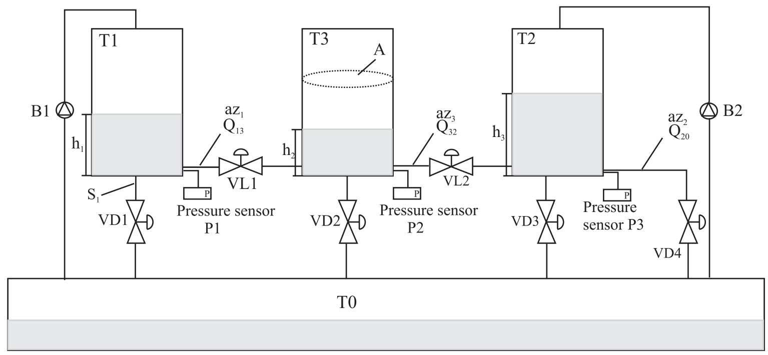

Fig. 6. Flowsheet of the experimental three tank process.

The results of the simulations using classical MPC controller and Step 8 in Table 4 are depicted in Fig. 5. Both the controllers present good control performance for the three controlled variables. However the rope length error and the swing angle decrease when weighted fuzzy MPC is applied. The best value for the swing angle is obtained with weights $w_x = 1.0$, $w_h = 0.5$ and $w_\theta = 0.25$, which has a maximum absolute value of 6.9°. When the classical approach is used, the maximum absolute value of swing angle is 10.1°. This simulation example shows clearly that the best results are obtained using a fuzzy objective function with different values for the weights. Also with the fault *F*2 the fuzzy FTC scheme proposed was able to detect, isolate and accommodate correctly the fault. Fault *F*2 degrades the control behavior of the trolley horizontal position, the rope length and of the swing angle. When this fault is active, the ship loading and unloading is again made correctly, as it was shown for fault *F*1. Fig. 5 shows that the three controlled variables are largely affected up to 40 s, time at which the fault is isolated. At this time, the model used in MPC is replaced, and the system behavior improves clearly. The position error reduces its value and it is close to zero at 90 s. The rope length is almost immediately settled to the reference at 75 s. Finally, the swing angle is largely increased when the fault occurs. However, when the fault is accommodated, the angle is very small, showing the good performance of the proposed fuzzy FTC scheme. Concluding, the FTC system proposed in this paper was able to detect, isolate and accommodate the two sensor faults considered.

6. Application example: three tank benchmark process

6.1. Description of the process

Another process used in this paper is an experimental benchmark which presents the transport of fluids in chemical processes. Many of the faults of chemical processes occur in the level of transport of fluids and raw materials, leaks, clogs, valve blockages and sensor faults. Fig. 6 presents the flowsheet of the used benchmark process (Dolanc, Juricic, Rakar, Petrovcic, & Vrancic, 1997). It consists of three tanks, T1, T2 and T3 connected with flow paths, which serve to supply water from the reservoir T0. The pump B1 supply the tank T1 and the pump B2 supply the tank T2. The supply voltage of pump B1 and the supply voltage of pump B2 denote the input signals. The liquid levels of T1, h_1 and the liquid levels of T2, h_2 denote the output signals. Valves VD1, VD2 and VD3 are on–off valves. The purpose of these valves is mainly to implement a real fault, i.e. the leakage of the tank T1 and/or T2, respectively. The capacity of the reservoir T0 is much larger than the capacity of the tanks so that its level is practically constant during the operation.

6.2. FDI results

The proposed fuzzy FTC scheme was applied to a model the process. Two faults are considered in this paper. Fault *F*1 is a leak in the tank T1. The result of this fault is a decrease in the liquid level of the tank T1. Further, another fault, *F*2, is a leak in the tank T2. The faults intensities considered are 100% and 75%, because when small faults intensities are considered the system controller is able to accommodate the fault effects.

The identification data used to build the valve model in normal operation contains 2000 samples. The same number of samples was used to identify each considered fault. A fuzzy model was identified for the model in normal operation. This model has two inputs, which are the supply voltage of the pumps *B*1 and *B*2. The outputs of the model are the liquid level of tank T1, h_1, the liquid level of tank T2, h_2, and the liquid level of tank T3, h_3.

Table 5 presents the modeling results when the process is without fault and with faults. The presented performance values are obtained for each one of the output variables, h_1, h_2 and h_3. In general, the fuzzy models present good accuracy when the system is with or without faults. The FDI step is made considering the scheme presented in Fig. 1. Faults *F*1 and *F*2 occur at 110 s. The performance of fault detection and fault isolation is presented in Table 6. The presented performance indices are presented in Bartys, Patton, Syfert, de las Heras, and Quevedo (2006). The indices, detection time t_{td} and isolation time t_{ti} are used to evaluate the performance of the proposed FDI architecture. The two faults *F*1 and *F*2 are correctly detected and isolated. When the fault intensities are decreased, the detection time t_{td} and the isolation time t_{ti} increase for both faults. Note however, that the obtained values of t_{td} and t_{ti} remain small.

6.3. Fuzzy FTC results

The fault accommodation of the three tank process was performed using the classical MPC and the fuzzy FTC. The obtained

Table 5
Accuracy of fuzzy models for process without faults and with faults.

Faults	VAF			RMS		
	h_1	h_2	h_3	h_1	h_2	h_3
No fault	99.8	99.3	98.8	0.002	0.003	0.020
F1	99.4	99.6	98.7	0.002	0.003	0.010
F2	99.2	89.3	93.2	0.002	0.120	0.001

Table 6
Detection and isolation performance.

Faults	$t_{td}(s)$	$t_{ti}(s)$	Fault intensity (%)
*F*1	9	11	100
*F*2	8	9	100
*F*1	10	12	75
*F*2	9	10	75

Table 7
Normalized errors using Yager t-norm with various weight combinations for the faults F1 and F2.

	*F*1				*F*2			
Class.	R_{h_1}	R_{h_2}	e_{h_1}	e_{h_2}	R_{h_1}	R_{h_2}	e_{h_1}	e_{h_2}
	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Fuzzy	w_1	w_2	e_{h_1}	e_{h_2}	w_1	w_2	e_{h_1}	e_{h_2}
1	1.0	1.0	0.98	1.17	1.0	1.0	0.98	0.90
2	1.0	0.5	1.06	1.07	0.5	1.0	1.01	0.89
3	1.0	0.25	0.98	0.63	0.25	1.0	1.0	0.87
4	1.0	0.05	1.03	0.54	**0.05**	**1.0**	**0.95**	**0.82**
5	**1.0**	**0.01**	**0.95**	**0.53**	0.01	1.0	0.98	0.86

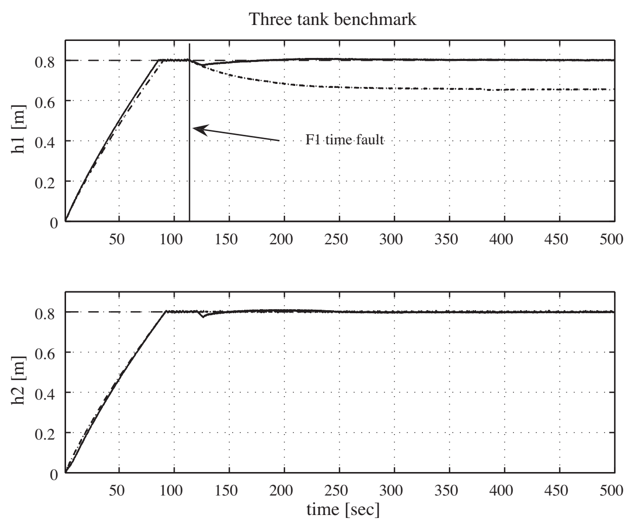

Fig. 7. Fault *F*1 accommodation (weighted fuzzy objective function). Solid-fault with weighted accommodation, dashed-reference, dashdot-fault without accommodation.

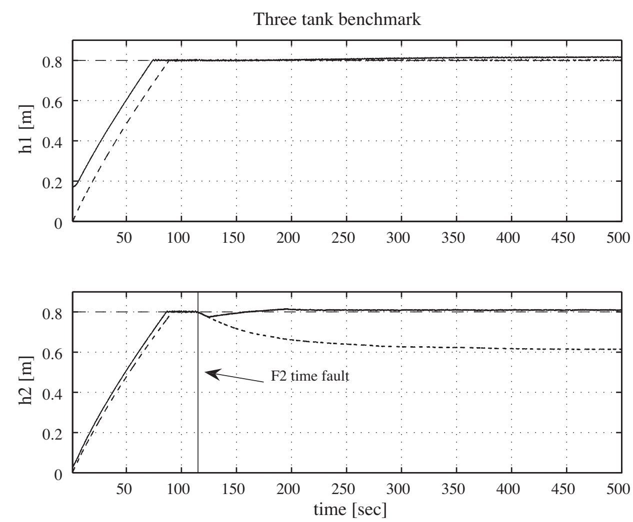

Fig. 8. Fault *F*2 accommodation (weighted fuzzy objective function). Solid-fault with weighted accommodation, dashed-reference, dashdot-fault without accommodation.

results for the accommodation of faults *F*1 and *F*2 are presented in Section 6.3.1 and Section 6.3.2, respectively.

6.3.1. Fuzzy FTC for fault F1

The fault accommodation is made considering the two outputs of process, h_1 and h_2. The approach proposed in this paper with weighted fuzzy MPC was applied to the system.

The weights w_1 and w_2 are for h_1 and h_2, respectively. The control performance is measured using the normalized sum squared error between the references and the outputs of the system after fault isolation. Table 7 shows the control results when the faults *F*1 and *F*2 occur, using the algorithm for weight selection described in Section 4.3. The error using the classical predictive controller with weights $R_{h_1} = 1.0$ and $R_{h_2} = 1.0$ is taken as 1 (100%), and it serves as the normalization to be compared with the errors using the Yager t-norm with the weights presented in Table 7. The absolute error values obtained using the classical controller are $e_{h_1} = 0.0065$ for the liquid level of tank T1 and $e_{h_2} = 0.0088$ for the liquid level of tank T2. The best result is obtained at Step 5 of the weighted fuzzy predictive control for fault *F*1 and at step 4 for fault *F*2.

The fuzzy FTC scheme proposed in this paper was able to detect, isolate and accommodate correctly the faults *F*1 and *F*2. The fault behavior of fault *F*1 and fault *F*2 can be observed in the liquid level of tank T1 and tank T2, respectively. Note that when one fault occurs, the fuzzy model in normal operation is substituted by the fuzzy model of the isolated fault. This faulty fuzzy model is used in the weighted fuzzy MPC scheme to derive the proper control actions.

The experimental results for fault *F*1 using fuzzy predictive control (weights presented in Step 5, Table 7) are depicted in Fig. 7. The controller presents good control performance for the two controlled variables. The error in the liquid level of tank T1 and the error in the liquid level of tank T2 decrease when weighted fuzzy MPC is applied. The best values for both controlled variables, are obtained with the weights $w_1 = 1.0$ and $w_2 = 0.01$. This experimental example shows clearly that the best results are obtained using a fuzzy objective function with different values for the weights.

6.3.2. Fuzzy FTC for fault F2

Considering the fault *F*2 also the error using the classical predictive control, presented in Table 7 is taken as 1 (100%), and it serves as the normalization. The best results obtained with the classical controller are $e_{h_1} = 0.0115$ for the liquid level of tank T1, and $e_{h_2} = 0.0121$ for the liquid level of tank T2, with weights $R_{h_1} = 1.0$, and $R_{h_2} = 1.0$. The best result is obtained at Step 4 with the weighted optimization. The experimental results using weighted fuzzy objective functions (weights presented in step 4, Table 7) are depicted in Fig. 8.

The controller presents good control performance for the two controlled variables. However, the error of h_1 and of h_2 decrease when weighted fuzzy MPC is applied.

Fig. 8 shows that the fault is isolated at 121 s. At this time, the model used in MPC is replaced, and the system behavior improves clearly. The liquid level error reduces its value and it is close to zero. Concluding, the FTC system proposed in this paper was able to detect, isolate and accommodate the two faults considered.

7. Conclusions

This paper proposes a weighted fuzzy FTC scheme to accommodate faults. The FTC approach is based on two steps: fault detection and isolation, and fault accommodation. In the first step, the FDI scheme is based on fuzzy models for both normal operation and for faulty operation, and on a fuzzy decision making approach. The fault isolation is performed by evaluating fuzzy decision factors that are built based on residuals. In the second step, the fault accommodation is made using weighted fuzzy MPC. The fuzzy models that were identified for the FDI step are now used in the weighted fuzzy MPC control scheme. The proposed approach is applied to a container gantry crane and to an experimental three tank process and shown its ability to detect, isolate and accommodate the faults.

Future research can consider the extension of the proposed FTC scheme to a larger number of faults, including incipient, intermittent or other types of faults.

Acknowledgments

This work is partially supported by the project POCI/EME/59522/2004, co-sponsored by FEDER, Programa Operacional Ciência e Inovação 2010, FCT, Portugal.

References

Bartys, M., Patton, R., Syfert, M., de las Heras, S., & Quevedo, J. (2006). Introduction to the DAMADICS actuator FDI benchmark study. *Control Engineering Practice, 14*, 577–596.

Blanke, M., Frei, C., Kraus, F., Patton, R. J., & Staroswiecki, M. (2000). What is fault-tolerant control? In *Proceedings of IFAC symposium on fault detection supervision and safety for technical processes, Budapest* (Vol. 1, pp. 40–51).

Blanke, M., Kinnaert, M., Lunze, J., & Staroswiecki, M. (2003). *Diagnosis and fault-tolerant control*. Berlin, Heidelberg: Springer-Verlag.

Chandler, P., Pachter, M., & Mears, M. (1995). System-identification for adaptive and reconfigurable control. *Journal of Guidance Control and Dynamics, 18*(3), 516–524.

Chen, R., & Patton, R. (1999). *Robust model-based fault diagnosis for dynamic systems*. Boston, MA: Kluwer Academic Publishers.

Dolanc, G., Juricic, D., Rakar, A., Petrovcic, J., & Vrancic, D. (1997). *Three-tank benchmark test*. Technical report COPL007R, Copernicus project CT94-0237. Jozef Stefan Institute.

Gopinathan, M., Boskovic, J. D., Mehra, R. K., & Rago, C. (1998). A multiple model predictive scheme for fault-tolerant flight control design. In *Proceedings of the 37th IEEE conference decision and control, Tampa, FL* (Vol. 2, pp. 1376–1381).

Guenaba, F., Webera, P., Theilliola, D., & Zhangb, Y. M. (2011). Design of a fault tolerant control system incorporating reliability analysis and dynamic behaviour constraints. *International Journal of Systems Science, 42*(1), 219–233.

Ichtev, A., Hellendoorn, J., Babuška, R., & Mollov, S. (2002). Fault-tolerant model-based predictive control using multiple takagi-sugeno fuzzy models. In *Proceedings of the IEEE international conference on fuzzy systems, FUZZ-IEEE'02* (Vol. 1(12–17), pp. 346–351).

Kaymak, U., & Sousa, J. M. (2003). Weighted constraint aggregation in fuzzy optimization. *Constraints, 8*(1), 61–78.

Koscielny, J. M., & Syfertm, M. (2003). Fuzzy logic applications to diagnostics of industrial processes. In *Preprints of the 5th IFAC symposium on fault detection, supervision and safety for technical processes, SAFEPROCESS'2003, Washington, USA* (pp. 771–776) .

Lopez-Toribio, C. J., Patton, R. J., & Daley, S. (2000). Takagi–Sugeno fuzzy fault-tolerant control of an induction motor. *Neural Computing & Applications, 9*, 19–28.

Maciejowski, J. M. (2002). *Predictive control with constrains*. Prentice-Hall.

Maciejowski, J. M., & Jones, C. N. (2003). MPC fault-tolerant flight control case study: Flight 1862. In *IFAC symposium SAFEPROCESS'2003* (pp. 121–125).

Mendonça, L. F., Sousa, J. M. C., & Sá da Costa, J. M. G. (2004). Optimization problems in multivariable fuzzy predictive control. *International Journal Approximate Reasoning, 36*(3), 199–221.

Mendonça, L. F., Sousa, J. M. C., & Sá da Costa, J. M. G. (2006a). Fault tolerant control using fuzzy MPC. In *Proceedings of Safeprocess 2006, 6th IFAC symposium on fault detection, supervision and safety of technical processes, Beijing, China* (pp. 1501–1506).

Mendonça, L. F., Sousa, J. M. C., & Sá da Costa, J. M. G. (2006b). Fault isolation using fuzzy model-based observers. In *Proceedings of Safeprocess 2006, 6th IFAC symposium on fault detection, supervision and safety of technical processes* (pp. 781–786).

Mendonça, L. F., Sousa, J. M. C., Kaymak, U., & Sá da Costa, J. M. G. (2006). Weighted goals and constraints in fuzzy predictive control. *Journal of Intelligent & Fuzzy Systems, 17*(5), 517–532.

Mendonça, L. F., Sousa, J. M. C., & Sá da Costa, J. M. G. (2009). An architecture for fault detection and isolation based on fuzzy methods. *Expert System with Applications, 36*(2), 1092–1134.

Patton, R. J. (1997). Fault-tolerant control systems: The 1997 situation. In *Proceedings of the 3rd IFAC symposium Kingston Upon Hull, UK* (pp. 26–28).

Patton, R. J., & Klinkhieo, S. (2009). An adaptive approach to active fault-tolerant control. *The Open Automation and Control Systems Journal, 2*, 54–61.

Polycarpou, M. M., & Helmicki, A. J. (1995). Automated fault detection and accommodation: A learning systems approach. *IEEE Transactions on Systems Man and Cybernetics, 25*(11), 1447–1458.

Sakawa, Y., & Shindo, Y. (1982). Optimal control of container cranes. *Automatica, 3*(18), 257–266.

Sousa, J. M. C., & Kaymak, U. (2001). Model predictive control using fuzzy decision functions. *IEEE Transactions on Systems Man and Cybernetics Part B: Cybernetics, 31*(1), 54–65.

Sousa, J. M. C., & Kaymak, U. (2002). *Fuzzy decision making in modeling and control*. World Scientific.

Steffen, T. (2005). *Control reconfiguration of dynamic systems*. Berlin, Heidelberg: Springer-Verlag.

Zhang, Y., & Jiang, J. (1985). Design issues for fault-tolerant restructurable aircraft control. In *Proceedings of 24th CDC, Fort Lauderdale, USA* (pp. 900–905).