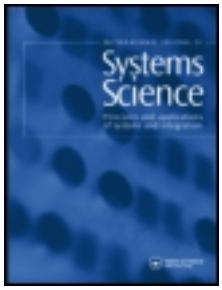
Active fault detection and isolation of discrete-time linear time-varying systems: a set-membership approach

Tabatabaeipour, Mojtaba

Published in:
International Journal of Systems Science

Link to article, DOI:
10.1080/00207721.2013.843213

Publication date:
2013



International Journal of Systems Science
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tsys20

Active fault detection and isolation of discrete-time linear time-varying systems: a set-membership approach

Seyed Mojtaba Tabatabaeipour[ab]
[a] Department of Electrical Engineering, Technical University of Denmark, Lyngby, Denmark
[b] Department of Electronic Systems, Aalborg University, Aalborg, Denmark
Published online: 07 Oct 2013.

To cite this article: Seyed Mojtaba Tabatabaeipour , International Journal of Systems Science (2013): Active fault detection and isolation of discrete-time linear time-varying systems: a set-membership approach, International Journal of Systems Science, DOI: 10.1080/00207721.2013.843213

To link to this article: http://dx.doi.org/10.1080/00207721.2013.843213

International Journal of Systems Science, 2013
http://dx.doi.org/10.1080/00207721.2013.843213

Active fault detection and isolation of discrete-time linear time-varying systems: a set-membership approach

Seyed Mojtaba Tabatabaeipour[a,b,*]

[a]*Department of Electrical Engineering, Technical University of Denmark, Lyngby, Denmark;* [b]*Department of Electronic Systems, Aalborg University, Aalborg, Denmark*

(*Received 2 August 2012; accepted 19 August 2013*)

Active fault detection and isolation (AFDI) is used for detection and isolation of faults that are hidden in the normal operation because of a low excitation signal or due to the regulatory actions of the controller. In this paper, a new AFDI method based on set-membership approaches is proposed. In set-membership approaches, instead of a point-wise estimation of the states, a set-valued estimation of them is computed. If this set becomes empty the given model of the system is not consistent with the measurements. Therefore, the model is falsified. When more than one model of the system remains un-falsified, the AFDI method is used to generate an auxiliary signal that is injected into the system for detection and isolation of faults that remain otherwise hidden or non-isolated using passive FDI (PFDI) methods. Having the set-valued estimation of the states for each model, the proposed AFDI method finds an optimal input signal that guarantees FDI in a finite time horizon. The input signal is updated at each iteration in a decreasing receding horizon manner based on the set-valued estimation of the current states and un-falsified models at the current sample time. The problem is solved by a number of linear and quadratic programming problems, which result in a computationally efficient algorithm. The method is tested on a numerical example as well as on the pitch actuator of a benchmark wind turbine.

Keywords: fault detection and isolation; fault diagnosis

1. Introduction

In modern industrial systems there is an increasing demand on performance, safety, and reliability. A fault in the system might degrade the performance of the system or eventually lead to the loss of its functionality or stability. Some severe faults or propagation of non-severe faults might result in hazardous events. Therefore, fault detection and isolation (FDI) is of crucial importance in modern industrial systems. In real applications, noise, uncertainties, and model differences are always present. To ensure the reliability and performance of an FDI method, it is important to make sure that it is robust to uncertainties and noise but simultaneously sensitive to faults. An FDI method that possesses this property is called robust.

Robust FDI methods are broadly classified into two classes: residual signal based and set-membership based. In the robust residual signal based fault detection, a residual signal is generated and its value is checked against a threshold. When the value of the residual signal becomes greater than the threshold, a fault is detected. For the method to be robust the residual signal must be insensitive to uncertainties and sensitive to faults, which is usually a difficult and challenging problem. Choosing the appropriate threshold is also an important and a challenging task. Among the most known residual based approaches are unknown input observers (Chen and Patton, 1999), eigenstructure assignment (Patton and Chen, 1991a), and structured parity equations (Patton and Chen, 1991b). In the set-membership approaches, the noise, disturbance, and uncertainties are assumed to be unknown but in given bounded sets. Then, a set of states or parameters consistent with the model of the system, past measurements, and bounds on the noise and uncertainties is computed for the system. If the current measurement is not consistent with any of the members of this set, a fault is detected. In the control literature, these approaches are known as set-membership, or error-bounded methods. For a review of set-membership approaches see Puig (2010) and Ingimundarson, Bravo, Puig, Alamo, and Guerra (2009). Relevant to this line of research are the methods in Olaru, De Doná, and Seron (2010) and Seron, Zhou, De Doná, and Martinez (2008) where positive invariant sets are used to obtain a set characterisation of faulty and healthy behaviour of residual signals. The advantage of these methods is that the online computation of sets is avoided. The problem of designing an excitation signal that guarantees FDI by separation between the invariant sets corresponding to faulty and healthy behaviours is investigated in Stoican, Olaru, Seron, and De Doná (2012).

Set-membership approaches are either state space based or parameter space based. At each iteration, the set of states

*Email: setaba@.elektro.dtu.dk

or parameters that are consistent with the past measurements, the model of the system, and the bounds on the uncertainties, disturbance, and noise is calculated as a closed set. If the set of states or the parameters consistent with a new measurement does not intersect with this set, a fault is detected. Therefore, in these approaches there is no need for threshold design. Also, when the given bounds on the uncertainties, noise, and disturbance are realistic, the approach does not generate any positive false alarm. The disadvantage of the method is that due to the propagation of uncertainties and over-approximations required in the set computations, it is possible for a measurement to be consistent with more than one faulty model or with the model of the normal and faulty system. Therefore, the fault would remain hidden or non-isolated.

In PFDI methods, it is possible that because of the low excitation signal or regulatory actions of the controller, the fault remains hidden or un-isolated in the normal operation. A remedy to this problem is to use an auxiliary excitation signal that is added to the input signal to excite the system on a periodic basis or at critical times to detect and isolate faults that would remain hidden or un-isolated otherwise. These solutions are called active FDI (AFDI) in the literature. In AFDI, an input is generated that excites the system with the aim of detecting and isolating the fault and then based on the observed output the condition of the system is determined. In recent years, there has been a considerable attention to the area of AFDI, see papers (Campbell and Nikoukhah, 2004; Niemann, 2006) and references therein. In Campbell and Nikoukhah (2004) dynamic optimisation is used to find the smallest auxiliary input that guarantees fault detection. The auxiliary input is pre-computed and then applied to the system in the online operation. During the detection period, measurements from the system are not used to update the pre-computed input. The test is stopped if the fault is detected before the end of period. This result is extended in Nikoukhah, Campbell, Savkin, and Selmic (2005) for uncertain sampled data systems using a multi-model framework. The multi-model approach is extended for detection of incipient faults in Nikoukhah and Campbell (2008) and to include the possibility of having a-priori information about the initial condition (Nikoukhah and Campbell, 2006). The proposed method in Campbell and Nikoukhah (2004) is extended for non-linear systems in Andjelkovic, Sweetingham, and Campbell (2008). Niemann and Poulsen (2005) and Niemann (2006) present a method for active diagnosis of parametric faults in closed loop systems based on YJKB parameterisation. In Stoustrup and Niemann (2010), two methods are proposed such that instead of generating an auxiliary signal the controller is altered. In the first method, the observer part of the controller is changed between sequences of observers each sensitive to one or a set of faults such that the continuity and stability of the transition is preserved. In the second method, the controller is changed such that the faulty system becomes unstable. All of the aforementioned approaches consider linear systems. In Tabatabaeipour, Ravn, Izadi-Zamanabadi, and Bak (2009a), an active fault diagnosis method for linear hybrid systems in discrete time based on reach set computation for faulty and normal systems is proposed. The results are extended to automatic sensor assignment in Tabatabaeipour, Izadi-Zamanabadi, Bak, and Ravn (2009b). The problem is reformulated in Tabatabaeipour (2010) as a mixed integer optimisation problem for active diagnosis of a hybrid system using the mixed logical dynamical framework. All of the above methods consider the problem in the open loop configuration. Esna Ashari, Nikoukhah, and Campbell (2012) study the effect of feedback on active fault detection with only one faulty model and show that when the norm of the auxiliary signal is considered to be the cost function to be minimised, linear feedback cannot reduce the cost considering the worst case of uncertainty.

In this paper, we consider the problem of AFDI formulated for a state space based set-membership FDI approach. A set-membership approach is used for PFDI of the system. The set of states consistent with the normal model and the models of the system subject to different faults are computed. When one of the sets becomes empty, the corresponding model is falsified. It is expected that after a while only one of the models remains un-falsified, i.e., is compatible with the measurement. When more than one model is compatible with the measurements, the AFDI comes into the picture. It receives the set-valued estimation of the states form the set-membership FDI approach as an input. Then, based on the models of the system (faulty and normal), it generates an input sequence that guarantees fault detection and isolation in a finite time horizon. The contribution of this paper includes the following:

(1) A new AFDI method based on set-membership PFDI approaches is proposed that finds the optimal input signal which guarantees detection and isolation of faults in a finite time horizon. The criterion for optimality is considered to be the norm of the input.
(2) The information about the initial condition of the normal system and faulty system subject to different faults is given as bounded sets where these sets might be different.
(3) The input can be updated at each sample time in a decreasing receding horizon manner using the measurement information. The input is updated at each iteration using the new available information from the set-membership PFDI algorithm which is the set-valued estimation of the states and unfalsified models at the current time.

This paper is organised as follows. In Section 2, preliminaries and basic notions that are used throughout the paper are introduced. Then, passive set-membership FDI is

explained in Section 3. The AFDI problem and the proposed method are given in Section 4. Simulation results of testing the method on a numerical example as well as on the pitch actuator of a benchmark wind turbine are given in Section 5. Finally, conclusions are given in Section 6.

2. Preliminaries

In this section basic definitions and notations used throughout the paper are given. Given two sets $\mathcal{X} \in \mathbb{R}^n$ and $\mathcal{Y} \in \mathbb{R}^n$, the Minkowski sum of them is defined as $\mathcal{X} \oplus \mathcal{Y} = \{x + y : x \in \mathcal{X}, y \in \mathcal{Y}\}$. A convex polytope P is the convex combination of its vertices. The polytope P with r vertices $v^i \in \mathbb{R}^n$ is the set:

$$P = \left\{ \sum_{i=1}^{r} \alpha^i v^i | v^i \in \mathbb{R}^n, \alpha^i \in \mathbb{R}, \alpha^i \geq 0, \sum_{i=1}^{r} \alpha^i = 1 \right\}. \quad (1)$$

P can also be represented by the non-empty intersection of a finite set of half-spaces. In this case the polytope P is represented by:

$$P = \{x | Hx \leq K\}. \quad (2)$$

The above representation is called the H-representation. Zonotopes are a special class of convex polytopes. A zonotope is the Minkowski sum of a finite number of line segments. A zonotope is represented by:

$$Z = \left\{ z \in \mathbb{R}^n | z = c + \sum_{i=1}^{p} x_i g_i, -1 \leq x_i \leq 1 \right\}. \quad (3)$$

Here, c is the centre of zonotope and g_i's are called generators.

Given the set $\mathcal{M} = \{M_0, M_1, \ldots, M_n\}$, its cardinality which is the number of its elements is denoted by $|\mathcal{M}|$. The index set of $\mathcal{M}$ is denoted by $\mathcal{I}$ which is a set that gathers indices of the elements of $\mathcal{M}$, i.e., $\mathcal{M} = \cup_{i \in \mathcal{I}} M_i$.

3. Set-membership fault detection and isolation

In set-membership approaches, instead of a point-wise estimation of states, at each sampling time, the set of states that are consistent with the current measurement, a given model of the system, the initial condition set, the bounds on the disturbance and noise, and the input-output sequence up to the current sample time is calculated. As long as this set is not empty, the corresponding model is valid and as soon as the set becomes empty it is falsified. Falsification of a given dynamic model means that the given model is not compatible with the observed input and output of the system. Therefore, if the given model is representing the model of the normal system, its falsification is equivalent to the detection of a fault. This is shown in Figure 1. The same procedure is used for fault isolation. Given $\mathcal{M} = \{M_0, \ldots, M_{n_f}\}$ representing models of the system with no fault and subject to fault $f_1, \ldots, f_{n_f}$, when M_0 is falsified, a fault is detected. Then, the method is applied to the un-falsified models until only a specific model, namely M_j remains un-falsified. At that point, the fault f_j is isolated. This is illustrated by an example in Figure 2. In the rest of the paper, we make the following assumptions:

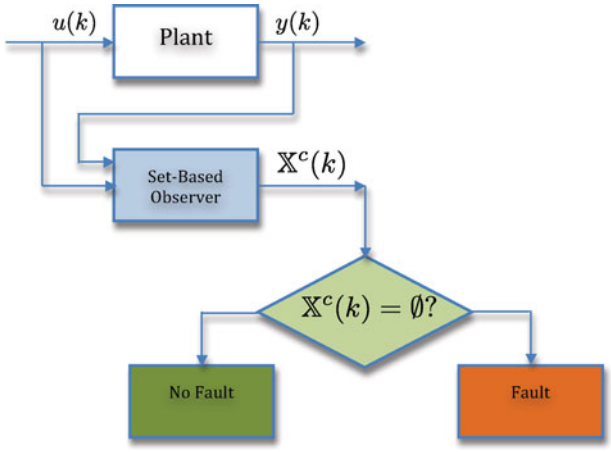


Figure 1. Basic structure of set-membership FD approach.

Assumption 1: *The models of the system subject to different faults $M_1, \ldots, M_{n_f}$ are known a priori.*

Assumption 2: *During the fault detection and isolation period the fault is persistent.*

3.1. Set-membership model falsification

In this section, it is explained how the set-membership approaches are used for falsification of a given model of a system. The following given linear time-varying model of the system is considered:

$$M_i : \begin{cases} x_i(k+1) = A_i(k)x_i(k) + B_i(k)u(k) + w(k), \\ y_i(k) = C_i(k)x_i(k) + v(k), \\ v(k) \in \mathcal{V}_i, w(k) \in \mathcal{W}_i, \\ x_i(0) \in \mathcal{X}_{i_0}, \end{cases} \quad (4)$$

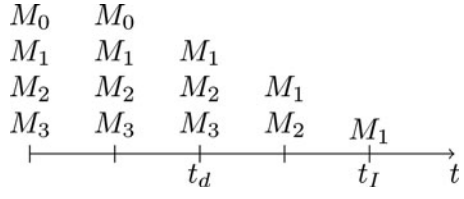


Figure 2. Example of fault detection and isolation using model falsification; each column shows the un-falsified models, t_d : detection time, t_I : isolation time.

where $x_i(k) \in \mathbb{R}^n$ is the state, $y_i(k) \in \mathbb{R}^m$ is the output, $u(k) \in \mathbb{R}^p$ is the input, $w(k) \in \mathbb{R}^n$ is disturbance, and $v(k) \in \mathbb{R}^m$ is noise. It is assumed that the noise and disturbance are unknown but bounded, i.e., $w(k) \in \mathcal{W}_i$ and $v(k) \in \mathcal{V}_i$. Moreover, it is assumed that the initial condition is given in a compact set $x_i(0) \in \mathcal{X}_{i_0}$. Also the input is assumed to be constrained in a compact polyhedral set, i.e., $u(k) \in \mathcal{U}$.

At each iteration, the set of states that are consistent with the given model M_i, the input and output of the system, the initial condition, and the bounds on the noise and disturbance are computed. This set is denoted by $\mathcal{X}_i^c(k)$. Computation of $\mathcal{X}_i^c(k)$ consists of two steps: a prediction step and a correction step. At the prediction step, having $\mathcal{X}_i^c(k-1)$, $u(k)$, based on the dynamic of the system and bounds on the disturbance, the set of all the states reachable from $\mathcal{X}_i^c(k-1)$ denoted by $\mathcal{X}_i^p(k)$ is computed. $\mathcal{X}_i^p(k)$ is defined by:

$$\mathcal{X}_i^p(k) = \{z = A_i(k-1)x + B_i(k-1)u + w | x \in \mathcal{X}_i^c(k-1), \; u = u(k-1), w \in \mathcal{W}_i\}, \tag{5}$$

which is computed as:

$$\mathcal{X}_i^p(k) = A_i(k-1)\mathcal{X}_i^c(k-1) \oplus \{B_i(k-1)u(k-1)\} \oplus \mathcal{W}_i. \tag{6}$$

This set is then corrected using the information available from the current measurement $y(k)$. Given the current measurement $y(k)$, the set of all states that are consistent with it is given by:

$$\mathcal{X}_i^y(k) = \{x \in \mathbb{R}^n : \exists v \in \mathcal{V}_i \text{ such that } C_i(k)x + v = y(k)\}. \tag{7}$$

Since the noise is additive here, we have:

$$\mathcal{X}_i^y(k) = \{x : C_i(k)x \in \{y(k)\} \oplus (-\mathcal{V}_i)\}. \tag{8}$$

The set of states compatible with the model and the measurement is consistent with both the prediction and the measurement which means that it is the intersection of the predicted set and the measurement consistent set:

$$\mathcal{X}_i^c(k) = \mathcal{X}_i^p(k) \cap \mathcal{X}_i^y(k). \tag{9}$$

The prediction and correction steps to find $\mathcal{X}_i^c(k)$ are depicted in Figure 3. As long as $\mathcal{X}_i^c(k)$ is not empty, this means that the model M_i is not falsified. But, when $\mathcal{X}_i^c(k)$ becomes empty, the input/output sequence can no longer be explained by M_i. Therefore, M_i is falsified. The overall algorithm for fault detection is given in the Algorithm 1. The model falsification algorithm can also be viewed as a set-valued observer (SVO) where $\mathcal{X}_i^c(k)$ is a set-valued estimation of the state, see Rosa, Casau, Silvestre, Tabatabaeipour, and Stoustrup (2012) and references therein.

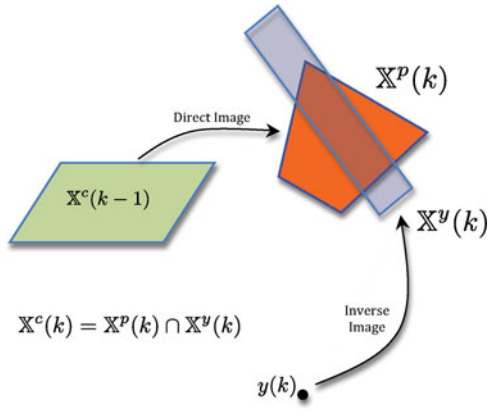


Figure 3. Calculation of the corrected consistent set.

Algorithm 1 Set-membership model falsification

Given $M_i : A_i(k), B_i(k), C_i(k), \mathcal{X}_{i_0}, \mathcal{V}_i, \mathcal{W}_i$
$k \leftarrow 0$, $\mathcal{X}_i^c(k) \leftarrow \mathcal{X}_{i_0}$
while $\mathcal{X}_i^c(k) \neq \emptyset$ **do**
  $k \leftarrow k+1$
  Given $u(k)$, find the prediction set:
  $\mathcal{X}_i^p(k) \leftarrow A_i(k-1)\mathcal{X}_i^c(k-1) \oplus \{B_i(k-1)u(k-1)\} \oplus \mathcal{W}_i$
  Given $y(k)$, find $\mathcal{X}_i^y(k)$
  $\mathcal{X}_i^y(k) \leftarrow \{x : C_i(k)x \in y(k) \oplus (-\mathcal{V}_i)\}$
  $\mathcal{X}_i^c(k) \leftarrow \mathcal{X}_i^p(k) \cap \mathcal{X}_i^y(k)$
  if $\mathcal{X}_i^c(k) = \emptyset$ **then**
    M_i is falsified
  end if
end while

For fault detection and isolation given $\mathcal{M} = \{M_0, \ldots, M_{n_f}\}$, a set valued observer for each model of the system is used. Then, a model M_i is falsified when the corresponding $\mathcal{X}_i^c$ becomes empty. The overall algorithm for model falsification using the set-membership approach is given in Algorithm 2. The algorithm is initiated at k_0 and runs until k_f. At each time step, for all models in $\mathcal{M}$ the set $\mathcal{X}_i^c$ is updated. If for model M_i, the set $\mathcal{X}_i^c$ becomes empty, it is falsified and therefore it is excluded from the set $\mathcal{M}$. The index of the model is also removed from the index set. After updating $\mathcal{M}$ and $\mathcal{I}$, the algorithm is repeated with the un-falsified models.

To implement the Algorithm 1 or 2, a specific set representation must be used. Several set representations are proposed in the literature, including ellipsoids, polytopes, intervals, parallelotopes, and zonotopes. Each representation

has its own benefits and downsides, see Alamo, Bravo, and Camacho (2005) and references therein. The representation that is used must be efficient concerning the operations that must be performed in the algorithm. Here, the operations are: affine transformation, Minkowski sum, and intersection. Ellipsoids are very simple to represent; however they are not closed under the Minkowski sum and intersection. Polytopes are closed under all the three required operations. Also, using polytopic representation results in exact computation of the sets, however the drawback of this representation is its high computational complexity. Zonotopes are closed under the affine transformation and the Minkowski sum and they offer low time and memory complexity. Although zonotopes are not closed under intersection but are a computationally efficient method for over-approximating the intersection of a strip and zonotopes are proposed in the literature that can be used here, see Alamo et al. (2005). In this paper we use zonotopes for PFDI. But during the AFDI horizon, because we need an exact computation of these sets, as it will be explained later, polytopic representation is used.

Algorithm 2 Model falsification using set-membership approach

$(\mathcal{M}, \mathcal{I}, \{\mathcal{X}_i^c\}_{i\in\mathcal{I}}) = \text{unfalsified}(\mathcal{M}, \mathcal{X}_i(k_0), \mathcal{V}_i, \mathcal{W}_i, k_0, k_f)$
Given $\mathcal{M}, \mathcal{X}_i(k_0), \mathcal{V}_i, \mathcal{W}_i, k_0, k_f$
$k \leftarrow k_0$, $\mathcal{X}_i^p(k) \leftarrow \mathcal{X}_i(k_0)$, $\mathcal{X}_i^c(k) \leftarrow \mathcal{X}_i(k_0)$
while $k \leq k_f$ **do**
  Get $y(k)$, $u(k)$
  for $j = 1$ to $|\mathcal{M}|$ **do**
    $i \leftarrow \mathcal{I}_j$
    $\mathcal{X}_i^p(k+1) \leftarrow A_i(k)\mathcal{X}_i^c(k) \oplus \{B_i(k)u(k)\} \oplus \mathcal{W}_i$
    $\mathcal{X}_i^y(k) \leftarrow \{x | C_i(k)x = y(k) \oplus (-\mathcal{V}_i)\}$
    $\mathcal{X}_i^c(k) \leftarrow \mathcal{X}_i^p(k) \cap \mathcal{X}_i^y(k)$
    if $\mathcal{X}_i^c(k) = \emptyset$ **then**
      $\mathcal{M} \leftarrow \mathcal{M} \setminus M_i, \mathcal{I} \leftarrow \mathcal{I} \setminus i$
    end if
  end for
  $k \leftarrow k + 1$
end while

4. Active fault detection and isolation

Because of the presence of noise and uncertainty or due to a small excitation signal or regulatory actions of the controller, it might happen that the input and output sequence is compatible with more than one model of the system. Consequently, when the Algorithm 2 is used, more than one model of the system remains un-falsified. Therefore, it is not possible to distinguish between the un-falsified models of the system. It must be pointed out that this problem is not only limited to set-membership approaches and it might happen in any PFDI method. A solution to this problem is to use AFDI methods to improve the distinguishability between un-falsified models by exciting the system using an auxiliary input signal. In AFDI, an input signal with the aim of fault detection and isolation is produced and injected into the system. Then, the output of the system is measured and based on the measured input/output, it is determined whether the system is in the normal or a faulty condition. In case the system is in the faulty condition, it is desirable to determine the faulty condition that the system is in. The input signal must be designed such that based on the observation it is possible to distinguish between the un-falsified models.

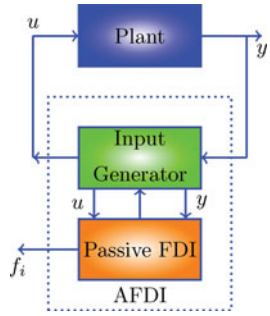


Figure 4. General structure of an AFDI module.

Figure 4 depicts the general structure of an AFDI method. The AFDI consists of an input generator module and a PFDI module. The input generator produces an input sequence that is injected into the system and then the PFDI module decides about the condition of the system by observing the output sequence.

The AFDI problem can be stated as follows:

Problem 1: Active fault detection and isolation problem: Given the set $\mathcal{M} = \{M_0, \ldots, M_{n_f}\}$ describing dynamical models of the system with no fault and subject to the faults $\{f_1, \ldots, f_{n_f}\}$ respectively, a set of initial states for each model, i.e., $\{\mathcal{X}_0(k_0), \ldots, \mathcal{X}_{n_f}(k_0)\}$, find a sequence of inputs $\mathbf{U}_{N_d}(k_0) = \{u(k_0) \ldots u(k_0 + N_d - 1)]$ such that the observation sequence $\mathbf{Y}_{N_d}(k_0 + 1) = \{y(k_0 + 1), \ldots y(k_0 + N_d)\}$ can only be produced by a unique $M_j, j \in \{0, 1, \ldots, n_f\}$.

In other words, the output sequences produced by applying the generated input sequence to the system in different conditions must be distinguishable. This means that we are searching for an input sequence, $\mathbf{U}_{N_d}(k_0)$, such that:

$$\forall x_i(k_0) \in \mathcal{X}_i(k_0), v_i(k) \in \mathcal{V}_i, w_i(k) \in \mathcal{W}_i : \exists \bar{k}, k_0 < \bar{k} \leq k_0 + N_d, \text{ such that: } y_i(k) \neq y_j(k), \forall i, j \in 0, \ldots, n_f, i \neq j, \quad (10)$$

where

$$x_i(k+1) = A_i(k)x_i(k) + B_i(k)u(k) + w_i(k), \quad y_i(k) = C_i(k)x_i(k) + v_i(k). \quad (11)$$

If such an input sequence exists, then we can look for the optimal solution, where optimality can be interpreted in different senses. The problem can be formulated as a feasibility test problem as follows:

$$\min_{\mathbf{U}_{N_d}(k_0)} \mathbf{1} \tag{12}$$
$$s.t. \begin{cases} x_i(k_0) \in \mathcal{X}_i(k_0), \\ x_i(k+1) = A_i(k)x_i(k) + B_i(k)u(k) + w_i(k), \\ y_i(k) = C_i(k)x_i(k) + v_i(k), \\ v_i(k) \in \mathcal{V}_i, \\ w_i(k) \in \mathcal{W}_i, \\ u(k) \in \mathcal{U}, \\ i \in \mathcal{I}, \\ k = k_0, \dots k_0 + N_d, \\ y_i(k) - y_j(k) \neq 0, i, j \in \mathcal{I}, i \neq j, \text{for some } \bar{k}, \\ \quad k_0 < \bar{k} \leq k_0 + N_d, \end{cases}$$

where N_d is the AFDI horizon and $\mathcal{I}$ is the index set of $\mathcal{M}$, i.e., $\mathcal{M} = \cup_{i\in\mathcal{I}} M_i$. This problem is in general non-convex. Assume that an input sequence $\mathbf{U}_N(k_0)$ is given. For a given input sequence if there exist a noise and disturbance sequence and an initial condition such that the following problem is feasible, then we cannot guarantee that the models would be distinguishable during the horizon:

$$\min_{\mathbf{U}_N(k_0)} \mathbf{1} \tag{13}$$
$$s.t. \begin{cases} x_i(k_0) \in \mathcal{X}_i(k_0), \\ x_i(k+1) = A_i(k)x_i(k) + B_i(k)u(k) + w_i(k), \\ u(k) = u(k), \\ y_i(k) = C_i(k)x_i(k) + v_i(k), \\ v_i(k) \in \mathcal{V}_i, \\ w_i(k) \in \mathcal{W}_i, \\ i \in \mathcal{I}, \\ k = k_0, \dots k_0 + N, \\ y_{\mathcal{I}_\ell}(k) - y_{\mathcal{I}_{\ell+1}}(k) = 0, 0 \leq \ell \leq |\mathcal{I}|, \\ k = k_0 + 1, \dots k_0 + N, \end{cases}$$

where $\mathcal{I}_\ell$ denotes the ℓth element of the set $\mathcal{I}$. But, infeasibility of the above problem means that there does not exist a $x(k_0) \in \mathcal{X}_0$, a noise and disturbance sequence $v_i(k) \in \mathcal{V}_i$, $w_i(k) \in \mathcal{W}_i$ such that all models produce the same output sequence, i.e., $y_i(k) - y_j(k) = 0$, $i, j \in \mathcal{I}$, $i \neq j$. In other words, at least two of the models produce difference output for some $\bar{k}$, $k_0 < \bar{k} \leq k_0 + N$ regardless of the realisation of the initial condition, noise, and disturbance, which means that at least two of the models are distinguishable. Therefore, to solve (12) we use a divide and conquer strategy. We look for input sequences that render (13) infeasible. If such an input sequence exists, this means that at the end of sequence, as it will be shown in the sequel, at least one of the models is falsified. Therefore, the size of the problem is reduced by one. Consequently, the problem can be solved by repeatedly applying this method.

In the following we prove that if (13) is infeasible, using the set-membership approach results in falsification of at least one of the models.

Theorem 1: *Given the set $\mathcal{M} = \{M_0, \dots, M_{n_f}\}$ describing dynamical models of the system with no fault and subject to the faults $\{f_1, \dots, f_{n_f}\}$ respectively, a set of initial states for each model, i.e., $\{\mathcal{X}_0(k_0), \dots, \mathcal{X}_{n_f}(k_0)\}$, and an input sequence $\mathbf{U}_N(k_0)$ that yields the output sequence $\mathbf{Y}_N(k_0+1) = \{y(k_0+1), \dots y(k_0+N)\}$, if the problem* (13) *is infeasible, then using Algorithm* 2 *with the output sequence $\mathbf{Y}_N(k_0+1)$ generated by applying $\mathbf{U}_N(k_0)$ to the system, at least one of the models is falsified.*

Proof: The theorem is proved using a *reductio ad absurdum* argument. Infeasibility of (13) means that:

$$\begin{aligned} &\nexists x_i^0(k_0) \in \mathcal{X}_i(k_0), v_i^0(k) \in \mathcal{V}_i, w_i^0(k) \in \mathcal{W}_i, : \\ &y_i(k) = y_j(k), \forall i, j \in 0, \dots, n_f, i \neq j, \\ &\qquad k = k_0 + 1, \dots, k_0 + N. \end{aligned} \tag{14}$$

Now, assume the conclusion of the theorem is not true meaning that none of the models are falsified during the period. This means that $\mathcal{X}_i^c(k) \neq \emptyset$ which is equal to:

$$\begin{aligned} &\forall i \in 0, \dots n_f, \exists x_i^1(k_0) \in \mathcal{X}_i(k_0), v_i^1(k) \in \mathcal{V}_i, w_i^1(k)\mathcal{W}_i : \\ &\forall k \in k_0 + 1, \dots, k_0 + N, \exists x_i^1(k) : \\ &y(k) = C_i(k)x_i^1(k) + v^1(i) \wedge x_i^1(k) = A_i(k)x_i^1(k-1) \\ &\qquad + B_i(k)u(k-1) + w^1(k-1). \end{aligned} \tag{15}$$

Therefore:

$$\begin{aligned} &\forall k \in k_0 + 1, \dots, k_0 + N, \exists x_i^1(k_0) \in \mathcal{X}_i(k_0), \\ &\quad v_i^1(k) \in \mathcal{V}_i, w_i^1(k)\mathcal{W}_i : y(k) = y_i(k). \end{aligned} \tag{16}$$

which means that:

$$\begin{aligned} &\forall l, m, l \neq m :, \forall k \in k_0 + 1, \dots, k_0 + N : \exists x_i^1(k_0) \in \mathcal{X}_i(k_0), \\ &\quad v_i^1(k) \in \mathcal{V}_i, w_i^1(k)\mathcal{W}_i : \\ &y(k) = y_l(k), y(k) = y_m(k) \rightarrow y_l(k) = y_m(k). \end{aligned} \tag{17}$$

This is in contradiction to (14), because at least when $x_i^0(k_0) = x_i^0(k_0)$, $v_i^0(k) = v_i^1(k)$, $w_i^0(k) = w_i^1(k)$, it is contradicted. This implies a contradiction that proves the theorem. □

In the proof no over-approximation is taken into account. Therefore, to implement the set-membership model falsification algorithm in the AFDI horizon we must use polytopes.

In fact, to falsify at least one of the models as in (13), it is not necessary to include all the un-falsified models in the optimisation problem. If we arbitrarily choose $\xi, \xi' \in \mathcal{I}$ and

find an input that renders the following problem infeasible:

$$\min_{\mathbf{U}_N(k_0)} \mathbf{1} \tag{18}$$
$$s.t. \begin{cases} x_i(k_0) \in \mathcal{X}_i(k_0), \\ x_i(k+1) = A_i(k)x_i(k) + B_i(k)u(k) + w_i(k), \\ u(k) = u(k), \\ y_i(k) = C_i(k)x_i(k) + v_i(k), \\ v_i(k) \in \mathcal{V}_i, \\ w_i(k) \in \mathcal{W}_i, \\ i \in \{\xi, \xi'\}, \\ k = k_0, \dots k_0 + N, \\ y_\xi(k) - y_{\xi'}(k) = 0, \\ k = k_0 + 1, \dots k_0 + N, \end{cases}$$

then we can guarantee that at least one of the models is falsified. This is stated in the following theorem.

Theorem 2: *Given the set $\mathcal{M} = \{M_0, \dots, M_{n_f}\}$ describing dynamical models of the system with no fault and subject to the faults $\{f_1, \dots, f_{n_f}\}$, respectively, a set of initial states for each model, i.e., $\{\mathcal{X}_0(k_0), \dots, \mathcal{X}_{n_f}(k_0)\}$, and an input sequence $\mathbf{U}_N(k_0)$ that yields the output sequence $\mathbf{Y}_N(k_0+1) = \{y(k_0+1), \dots y(k_0+N)\}$, if the problem* (18) *with ξ, ξ' chosen arbitrarily form $\mathcal{I}$, is infeasible, then using Algorithm* 2 *with the output sequence $\mathbf{Y}_N(k_0+1)$ generated by applying $\mathbf{U}_N(k_0)$ to the system, at least one of the models M_ξ or $M_{\xi'}$ is falsified.*

Proof: Assume that the correct model is $\xi^\star$. Then three cases are possible:

1. $\xi = \xi^\star$: In this case according to Theorem 1, the model M_ξ, would be falsified.
2. $\xi' = \xi^\star$: In this case according to Theorem 1, the model $M_{\xi'}$, would be falsified.
3. $\xi \neq \xi^\star$ and $\xi' \neq \xi^\star$: Assume, *ad absurdum*, that in this case neither M_ξ nor M'_ξ is falsified. This means that:

$$\begin{aligned} &\forall i \in \{\xi, \xi'\}, \forall k \in k_0 + 1, \dots, k_0 + N, \\ &\quad \exists x_i^1(k_0) \in \mathcal{X}_i(k_0), v_i^1(k) \in \mathcal{V}_i, \\ &\quad w_i^1(k) \in \mathcal{W}_i : y(k) = y_i(k), \end{aligned} \tag{19}$$

which means that:

$$\begin{aligned} &\forall k \in k_0 + 1, \dots, k_0 + N, \forall i \in \{\xi, \xi'\}, \\ &\quad \exists x_i^1(k_0) \in \mathcal{X}_i(k_0), v_i^1(k) \in \mathcal{V}_i, \\ &\quad w_i^1(k) \in \mathcal{W}_i : y(k) = y_\xi(k), \\ &\quad y(k) = y_{\xi'}(k) \rightarrow y_\xi(k) = y_{\xi'}(k). \end{aligned} \tag{20}$$

This in contradiction to the infeasibility of (18) that requires:

$$\begin{aligned} &\nexists x_\xi^0(k_0) \in \mathcal{X}_\xi(k_0), v_\xi^0(k) \in \mathcal{V}_\xi, \\ &\quad w_\xi^0(k) \in \mathcal{W}_\xi \text{ and } x_{\xi'}^0(k_0) \in \mathcal{X}^{\xi'}(k_0), v_{\xi'}^0(k) \in \mathcal{V}_{\xi'}, \\ &\quad w_{\xi(k)'}^0 \in \mathcal{W}_{\xi'} : y_\xi(k) = y_{\xi'}(k), \\ &\quad \forall k = k_0 + 1, \dots, k_0 + N. \end{aligned} \tag{21}$$

Therefore, at least M_ξ or M'_ξ is falsified. □

To find an input sequence that renders (18) infeasible, it is assumed that the horizon for each subproblem is N. We need to find the feasible region of the following optimisation problem.

$$\mathcal{J}_{\xi,\xi'}(\mathbf{U}_N(k_0)) = \min \mathbf{1} \tag{22}$$
$$s.t. \begin{cases} x_i(k) \in \mathcal{X}_i(k_0), \\ x_i(k+1) = A_i(k)x_i(k) + B_i(k)u(k) + w_i(k), \\ y_i(k) = C_i(k)x_i(k) + v_i(k), \\ v_i(k) \in \mathcal{V}_i, \\ w_i(k) \in \mathcal{W}_i, \\ k = k_0, \dots, k_0 + N, i \in \{\xi, \xi'\}, \\ y_\xi(k) - y_{\xi'}(k) = 0 \\ k = k_0 + 1, \dots, k_0 + N, \\ \mathbf{U}_N \in \mathcal{U}^N. \end{cases}$$

Then, we search for an $\mathbf{U}_N(k_0)$ that is not inside the feasible region; hence rendering (18) infeasible. The feasible region of the above problem is a polytope denoted here by $\mathcal{G}(\xi, \xi')$ (see Appendix A for a description of the feasible region). To find input sequences that are not inside the feasible region we project $\mathcal{G}(\xi, \xi')$ on the input space. Then, we find an input that lies outside the projection. For now, assume that we can find a $\mathbf{U}_N(k_0)$. Then, we apply the input sequence and observe the output. As a result, at least one of the models must be falsified. For the remaining un-falsified models we repeat the algorithm. It is possible that, given $\mathcal{X}_i^c$ and constraints on the input, some of the models are not distinguishable from each other. In this case, the algorithm must be repeated until all the falsifiable models are falsified.

To find a $\mathbf{U}_N(k_0)$ that is outside the feasible region of (13), the following approach is used. Let us assume that the projection of the feasible region of (22) on the $\mathbf{U}_N$ space is given by the polytope:

$$P(\xi, \xi') = \{\mathbf{U}_N | H(\xi, \xi')\mathbf{U}_N \leq K(\xi, \xi')\}, \tag{23}$$

Moving the facets of $P(\xi, \xi')$ outward by vector $\epsilon[\,\|h^1\|_2, \|h^2\|_2, \cdots \|h^r\|_2\,]$, where ϵ is a small positive

number, we get the following polytope:

$$P^e(\xi,\xi') = \{\mathbf{U}_N | H(\xi,\xi')\mathbf{U}_N \leq K(\xi,\xi') + \epsilon\big[\, \|h^1\|_2, \|h^2\|_2, \dots \|h^r\|_2 \,\big]\}, \quad (24)$$

where $\|h^i\|_2$ is the 2-norm of the *ith* row of H. It is clear that $P(\xi,\ \xi') \subset P^e(\xi,\ \xi')$. Let us define $K^e = K(\xi,\xi') + \epsilon[\, \|h^1\|_2, \|h^2\|_2, \dots \|h^r\|_2\,]$ and denote its *ith* element by k^{e^i}. Therefore, any point in the set $\{\cup_{i=1}^r \mathbf{U}_N | h^i(\xi,\xi')\mathbf{U}_N \geq k^{e^i}(\xi,\xi')\}$ lies outside of the set $P(\xi,\xi')$. Therefore, to find the optimal input sequence we solve the following problem:

$$\min_{\mathbf{U}_N} \mathbf{U}_N^T Q \mathbf{U}_N \quad (25)$$
$$s.t. \begin{cases} \cup_{i=1}^r h^i(\xi,\xi')\mathbf{U}_N \geq k^{e^i}(\xi,\xi') \\ \mathbf{U}_N \in \mathcal{U}^N. \end{cases}$$

The above optimisation problem is not convex since the set $\{\cup_{i=1}^r \mathbf{U}_N | h^i(\xi,\xi')\mathbf{U}_N \geq k^{e^i}(\xi,\xi')\}$ is non-convex. However, this set is actually a union of r convex sets: $\bigcup_{i=1}^r \{h^i \mathbf{U}_N \geq k^{e^i}\}$. Hence, the above optimisation problem is solved by first solving the following quadratic optimisation problem:

$$\min_{\mathbf{U}_N^i} \mathbf{U}_N^{i^T} Q \mathbf{U}_N^i \quad (26)$$
$$s.t. \begin{cases} h^i(\xi,\xi')\mathbf{U}_N^i \geq k^{e^i}(\xi,\xi') \\ \mathbf{U}_N^i \in \mathcal{U}^N. \end{cases}$$

for $i = 1, \dots, r$ and then finding the minimum of the solutions. We can also accommodate linear inequality constraints on inputs.

The overall algorithm is given in Algorithm 3. The sets $\mathcal{M}^l$ in the outer while loop are used to keep track of the un-falsified models. If the un-falsified models at two iterations are the same, this means that these set of models are not distinguishable from each other given the set of initial conditions and constraints on the input. In this case the algorithm terminates. Therefore, the set $\mathcal{M}^0$ is set as an empty set to initiate the algorithm ($\mathcal{M}^1 \neq \mathcal{M}^0$). At each iteration, the first two models from $\mathcal{M}^l$ are chosen. If (25) is feasible, $\mathbf{U}_N$ is applied to the system and after N steps either M_ξ or $M_{\xi'}$ is falsified. The falsified model is excluded from $\mathcal{M}^l$ and the procedure is repeated with the next two elements of $\mathcal{M}^l$. In this way all pairs are checked. At each iteration, in case (25) is infeasible M_ξ is deleted form $\mathcal{M}^l$ and put in $\mathcal{M}^{l+1}$ which gathers the set of un-falsified models at iteration l. (In case only M_ξ, $M_{\xi'}$ remain in $\mathcal{M}^l$ they are both added to $\mathcal{M}^{l+1}$.)

Algorithm 3 requires finding the feasible region of the optimisation problem in (22) online each time when $\mathcal{X}_i^c(k_0)$ is updated. It is possible to avoid this step by eliminating the constraints on the initial conditions, i.e., $x_i(k_0) \in \mathcal{X}_i^c(k_0)$. Therefore, the feasible region is described by:

Algorithm 3 Active fault detection and isolation algorithm

Given $\mathcal{M}$, $\mathcal{X}_i^c(k_0)$, $\mathcal{V}_i$, $\mathcal{W}_i$, N
$\mathcal{M}^1 \leftarrow \mathcal{M}$, $\mathcal{M}^0 \leftarrow \emptyset$, $l \leftarrow 1$
while $\mathcal{M}^l \neq \mathcal{M}^{l-1}$ **do**
  while $|\mathcal{M}^l| > 1$ **do**
    $\mathcal{I} \leftarrow \{j | M_j \in \mathcal{M}^l\}$, $\xi \leftarrow \mathcal{I}_1$, $\xi' \leftarrow \mathcal{I}_2$
    Use (22) to find $\mathcal{G}(\xi,\xi')$, $P(\xi,\xi') \leftarrow$ *projection*$(\mathcal{G}(\xi,\xi'), \mathbf{U}_N)$
    Calculate $P^e(\xi,\xi')$ using (24)
    if (25) is feasible **then**
      Find the input sequence $\mathbf{U}_N(k_0)$ by solving (25)
      Apply $\mathbf{U}_N$ to the system
      $(\mathcal{M}^l, \mathcal{X}_i^c(k_0+N)) =$ unfalsified$(\mathcal{M}^l, \mathcal{X}_i^c(k_0), \mathcal{V}_i, \mathcal{W}_i, k_0, k_0+N)$
      $k_0 \leftarrow k_0 + N$
    else if $|\mathcal{M}^l| > 2$ **then**
      $\mathcal{M}^{l+1} \leftarrow \mathcal{M}^{l+1} \cup M_\xi$, $\mathcal{M}^l \leftarrow \mathcal{M}^l \setminus M_\xi$
    end if
  end while
  $\mathcal{M}^{l+1} \leftarrow \mathcal{M}^{l+1} \cup \mathcal{M}^l$
  $l \leftarrow l + 1$
end while

$$\mathcal{G}(\xi,\xi') : \begin{cases} x_i(k+1) = A_i(k)x_i(k) + B_i(k)u(k) + w_i(k), \\ y_i(k) = C_i(k)x_i(k) + v_i(k), \\ v_i(k) \in \mathcal{V}_i, \\ w_i(k) \in \mathcal{W}_i, \\ k = k_0, \dots, k_0+N, i \in \{\xi,\xi'\}, \\ y_\xi(k) - y_{\xi'}(k) = 0, \\ k = k_0+1, \dots, k_0+N, \mathbf{U}_N \in \mathcal{U}^N. \end{cases} \quad (27)$$

Then, in the online operation, when we get $\{\mathcal{X}_i^c(k_0)\}_{i\in\{\xi,\xi'\}}$ from the model falsification algorithm, the intersection of $\mathcal{G}(\xi,\xi')$ and $\mathcal{U}^N \times \mathcal{X}_\xi^c(k_0) \times \mathcal{X}_\xi^c(k_0)$ is obtained by adding the constraints on the initial states. If the intersection is not empty, for any point $\{x_\xi(k_0)\} \times \{x_{\xi'}(k_0)\}$ in this intersection, there is at least one input sequence $\mathbf{U}_N$ that makes (22) feasible. The projection of this polytope on the $\mathbf{U}_N$ is computed. Using a similar argument as before, we need to find an input sequence lying outside the projection. Applying this input to the system falsifies at least one of the models. We exclude the falsified models from the set of un-falsified models and repeat the algorithm. Since we do not know a priori, the order of the models that would be falsified, we need to find the feasible region in (27) for all unordered pair from $\mathcal{M}$ and store them in a look-up table. Then, at each iteration, having the updated $\mathcal{M}$, we recall the corresponding feasible region, namely $P(\xi,\xi')$ from the look-up table. This is summarised in the Algorithm 4. In the algorithm *projection*$(P, \mathbf{U}_N)$ denotes projection of the polytope P on the input sequence dimensions $\mathbf{U}_N$ and k_{afd} denotes the time that the AFDI algorithm is initiated. For

example, when by using Algorithm 2 more than one model remains un-falsified for a certain time instant, the algorithm is initiated.

Algorithm 4 AFDI algorithm with offline computation of P^e

Given $\mathcal{M}, \mathcal{I}, \mathcal{X}_i(0), \mathcal{V}_i, \mathcal{W}_i, N$
OFFLINE
for all unordered pair $\{\xi, \xi'\} \in \mathcal{I}$ **do**
  Use (27) to find $\mathcal{G}(\xi, \xi')$
  Store $P(\xi, \xi')$ in the look-up table $\mathcal{LU}$
end for
ONLINE
if $k = k_{afd}$ **then**
  Get $\mathcal{M}, \mathcal{I}, \mathcal{X}_i^c(k_0)$ from Algorithm 2, $k_0 \leftarrow k_{afd}$
end if
$\mathcal{M}^1 \leftarrow \mathcal{M}, \mathcal{M}^0 \leftarrow \emptyset, l \leftarrow 1$
while $\mathcal{M}^l \neq \mathcal{M}^{l-1}$ **do**
  while $|\mathcal{M}^l| > 1$ **do**
    $\mathcal{I} \leftarrow \{j | M_j \in \mathcal{M}^l\}, \xi \leftarrow \mathcal{I}_1, \xi' \leftarrow \mathcal{I}_2$
    Recall $P(\xi, \xi')$ from the look-up table $\mathcal{LU}$
    $\mathcal{P} \leftarrow \mathcal{U}^N \times \mathcal{X}_\xi^c(k_0) \times \mathcal{X}_{\xi'}^c(k_0)$
    $\mathcal{P} \leftarrow \mathcal{P} \cap P$
    $P \leftarrow projection(\mathcal{P}, U_N)$
    Use (24) to calculate $P^e(\xi, \xi')$
    if (25) is feasible **then**
      Find the input sequence $\mathbf{U}_N$ by solving (25)
      while $\{M_\xi, M_{\xi'}\} \in \mathcal{M}^l$ **do**
        Apply the $k'th$ element of $\mathbf{U}_N$ to the system
        $(\mathcal{M}^l, \mathcal{X}_i^c(k_0+1)) \leftarrow$ unfalsified$(\mathcal{M}, \mathcal{X}_i^c(k_0), \mathcal{V}_i, \mathcal{W}_i,$
          $k_0, k_0+1)$
        $k_0 \leftarrow k_0 + 1$
      end while
    else if $|\mathcal{M}^l| > 2$ **then**
      $\mathcal{M}^{l+1} \leftarrow \mathcal{M}^{l+1} \cup M_\xi, \mathcal{M}^l \leftarrow \mathcal{M}^l \setminus M_\xi$
    end if
  end while
  $\mathcal{M}^{l+1} \leftarrow \mathcal{M}^{l+1} \cup \mathcal{M}^l$
  $l \leftarrow l + 1$
end while

Note that it might happen that one of the models is falsified before the termination of the horizon, i.e., before N steps. Therefore, in the Algorithm 4, if this happens, the procedure of applying $\mathbf{U}_N$ is stopped and the algorithm exit the innermost while loop.

It is also possible to apply Algorithm 4 in a decreasing receding horizon fashion where at each iteration the input sequence is updated. At each iteration $\mathbf{U}_N$ is obtained. Then, only the first element of it, $u(k_0)$, is applied to the system. Then, we observe the input and output of the system to check if a model is falsified. Based on this result $\mathcal{M}$ is updated. Then, the length of the horizon is decreased by 1, i.e., $N_0 \leftarrow N_0 - 1$ and the whole procedure is repeated. The advantage of this method is that at each iteration the input is updated based on the set-valued estimation of the states using the set-valued observers which means that at each step the input is recalculated based on the measurement information. The algorithm is given in Algorithm 5.

Algorithm 5 AFDI: The decreasing receding horizon approach

Given $\mathcal{M}, \mathcal{I}, \mathcal{X}_i(0), \mathcal{V}_i, \mathcal{W}_i, N$
OFFLINE
For all unordered pair $\{\xi, \xi'\} \in \mathcal{I}$:
Use (27) with $N_0 = 1, \ldots, N$ to find $\mathcal{G}(\xi, \xi', N_0)$,
Store $P(\xi, \xi', N_0)$ in the look-up table $\mathcal{LU}$
ONLINE
if $k = k_{afd}$ **then**
  $k_0 \leftarrow k_{afd}$
  Get $\mathcal{M}, \mathcal{I}, \mathcal{X}_i^c(k_0)$ from Algorithm 2
end if
$\mathcal{M}^1 \leftarrow \mathcal{M}, \mathcal{M}^0 \leftarrow \emptyset, l \leftarrow 1$
while $\mathcal{M}^l \neq \mathcal{M}^{l-1}$ **do**
  while $|\mathcal{M}^l| > 1$ **do**
    $\mathcal{I} \leftarrow \{j | M_j \in \mathcal{M}^l\}, \xi \leftarrow \mathcal{I}_1, \xi' \leftarrow \mathcal{I}_2, N_0 \leftarrow N$
    Recall $P(\xi, \xi', N_0)$ from the look-up table $\mathcal{LU}$
    $\mathcal{P} \leftarrow \mathcal{U}^{N_0} \times \mathcal{X}_\xi^c(k_0) \times \mathcal{X}_{\xi'}^c(k_0)$
    $\mathcal{P} \leftarrow \mathcal{P} \cap P$
    $P \leftarrow projection(\mathcal{P}, U_{N_0})$, Use (24) to calculate $P^e(\xi, \xi')$
    if (25) is feasible **then**
      while $\{M_\xi, M_{\xi'}\} \in \mathcal{M}^l$ **do**
        Find the input sequence $\mathbf{U}_{N_0}$ by solving (25)
        Apply the first element of $\mathbf{U}_{N_0}$ to the system
        $(\mathcal{M}^l, \mathcal{X}_i^c(k_0+1)) \leftarrow$ unfalsified$(\mathcal{M}, \mathcal{X}_i^c(k_0), \mathcal{V}_i, \mathcal{W}_i,$
          $k_0, k_0+1)$
        if $N_0 > 1$ **then**
          $N_0 \leftarrow N_0 - 1$
          Recall $P(\xi, \xi', N_0)$ from the look-up table $\mathcal{LU}$
          $\mathcal{P} \leftarrow \mathcal{U}^{N_0} \times \mathcal{X}_\xi^c(k_0) \times \mathcal{X}_{\xi'}^c(k_0)$
          $\mathcal{P} \leftarrow \mathcal{P} \cap P$
          $P \leftarrow projection(\mathcal{P}, U_{N_0})$, Use (24) to calculate
            $P^e(\xi, \xi')$
        end if
      end while
    else if $|\mathcal{M}^l| > 2$ **then**
      $\mathcal{M}^{l+1} \leftarrow \mathcal{M}^{l+1} \cup \mathcal{M}^l$
    end if
  end while
  $\mathcal{M}^{l+1} \leftarrow \mathcal{M}^{l+1} \cup \mathcal{M}^l$
  $l \leftarrow l + 1$
end while

The algorithms explained so far, eliminate the incorrect models sequentially. In the following we propose a method to falsify the incorrect models simultaneously. This means that the inputs sequence must lie outside the projection of the feasible region of (22) on the input space, i.e., $P(\xi, \xi')$ for all unordered pairs of $\{\xi, \xi'\} \in \mathcal{I}$. Therefore, the optimisation problem that must be solved is:

$$\min_{\mathbf{U}_N} \mathbf{U}_N^T Q \mathbf{U}_N \tag{28}$$
$$s.t. \begin{cases} \cup_{i=1}^r h^i(\xi, \xi') \mathbf{U}_N \geq k^{e^i}(\xi, \xi'), \forall \{\xi, \xi'\} \in \mathcal{I} \\ \mathbf{U}_N \in \mathcal{U}^N. \end{cases}$$

The inequality constraints obviously form a non-convex set, but we can solve the problem by dividing it into

convex subproblems, finding the minimum for each subproblem and then finding the minimum of the solutions of subproblems. This is shown in Algorithm 6. In the algorithm $\mathcal{N} = \binom{|\mathcal{M}|}{2}$ denotes the number of subsets $\{\xi, \xi'\}$ of $\mathcal{M}$. For each $1 \leq \ell \leq \mathcal{N}$, H_ℓ, K_ℓ^e denotes the corresponding matrices for the expanded polytopes of feasible region, r_ℓ denotes the number of rows of H_ℓ, h_ℓ^j denotes the *jth* rows of the H_ℓ and $k_\ell^{e^j}$ denotes the *jth* element of the K_ℓ^e. The region outside each expanded feasible region $\mathcal{P}^{e^\ell}$ is given as $\cup_{j=1}^{r_\ell} \mathcal{H}_\ell^j$, where each $\mathcal{H}_\ell^j$ is a halfspace given as: $\mathcal{H}_\ell^j = \{\mathbf{U}_N | h_\ell^j \mathbf{U}_N \geq k_\ell^{e^j}\}$. Therefore, the area outside all feasible regions is given by:

Algorithm 6 Solving (28) by sequential convex optimisation

for $i_1 = 1$ to r_1 **do**
  ⋮
  for $i_\mathcal{N} = 1$ to $r_\mathcal{N}$ **do**
    if (31) is feasible **then**
      $j \leftarrow j + 1$
      $\mathfrak{U}_j = \arg\min \Gamma(i_1, \ldots, i_\mathcal{N})$
    end if
  end for
  ⋮
end for
if $j = 0$ **then**
  The problem is infeasible.
else
  $\mathbf{U}_N = \min\{\mathfrak{U}_\ell | 1 \leq \ell \leq j\}$
end if

$$\left(\bigcup_{i_1=1}^{\mathcal{N}} \left(\mathcal{H}_1^{i_1}\right)\right) \bigcap \cdots \bigcap \left(\bigcup_{i_\mathcal{N}=1}^{\mathcal{N}} \left(\mathcal{H}_\mathcal{N}^{i_\mathcal{N}}\right)\right), \tag{29}$$

which is equal to:

$$\bigcup_{i_1=1}^{\mathcal{N}} \cdots \bigcup_{i_\mathcal{N}=1}^{\mathcal{N}} \left(\mathcal{H}_1^{i_1} \bigcap \cdots \bigcap \mathcal{H}_\mathcal{N}^{i_\mathcal{N}}\right). \tag{30}$$

Consequently, the following subproblem must be solved for all possible combinations of $i_1, \ldots, i_\mathcal{N}$:

$$\Gamma(i_1, \ldots, i_\mathcal{N}) = \min_{\mathbf{U}_N} \mathbf{U}_N^T Q \mathbf{U}_N \tag{31}$$
$$s.t. \begin{cases} h_1^{i_1} \mathbf{U}_N \geq k_1^{e^{i_1}} \\ \vdots \\ h_\mathcal{N}^{i_\mathcal{N}} \mathbf{U}_N \geq k_\mathcal{N}^{e^{i_\mathcal{N}}} \\ \mathbf{U}_N \in \mathcal{U}^N \end{cases}$$

Remark 1: In this way it is possible to construct an input sequence that simultaneously falsifies the incorrect models and by checking the feasibility of the method one can guarantee the existence of such an input and guarantee fault detection an isolation in N steps.

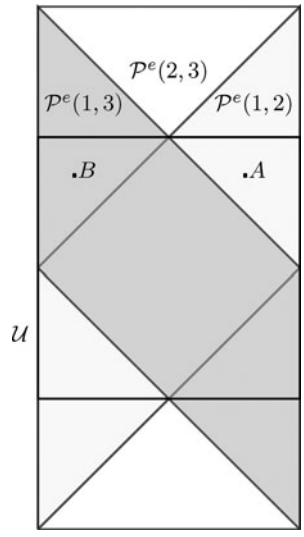


Figure 5. $\mathcal{P}^e$ for all pairs in Remark 3.

Remark 2: A good choice for N is the smallest N such that the optimisation problem in (31) is feasible. This would require solving the problem several times online. Note that N might change depending on the models that are still unfalsified. If this method is not applicable, another option is to choose an N based on the offline analysis of the feasible regions $\mathcal{G}(\xi, \xi')$. One could choose a griding of the feasible region in the $x_\xi(k_0) \times x'_\xi(k_0)$ space, and then find a good choice of N for each area of the grid. In the online operation based on the location of the $\mathcal{X}_1^c(k) \times \mathcal{X}_2^c(k)$ in the grid, the corresponding N is used.

Remark 3: There might be cases that it is not possible to falsify all the modes simultaneously, i.e., the optimisation problem is infeasible, but it is possible to detect and isolate the fault using the sequential method of Algorithm 4. To exemplify this remark consider an example with models M_1, M_2, M_3. Assume that the correct model is M_1. Figure 5 shows $\mathcal{P}^e$ for all unordered pairs and the set $\mathcal{U}$ that denotes the constraints on the input. As can be seen from the figure, finding an input that falsifies all the models simultaneously without violating the constraints on input is impossible. However, it is possible to first apply the input denoted by point A which falsifies M_3 and then apply the input B which falsifies M_2.

5. Examples

5.1. *Example 1*

In this example, we use a simple system with two faults to demonstrate the algorithm. We consider the following example:

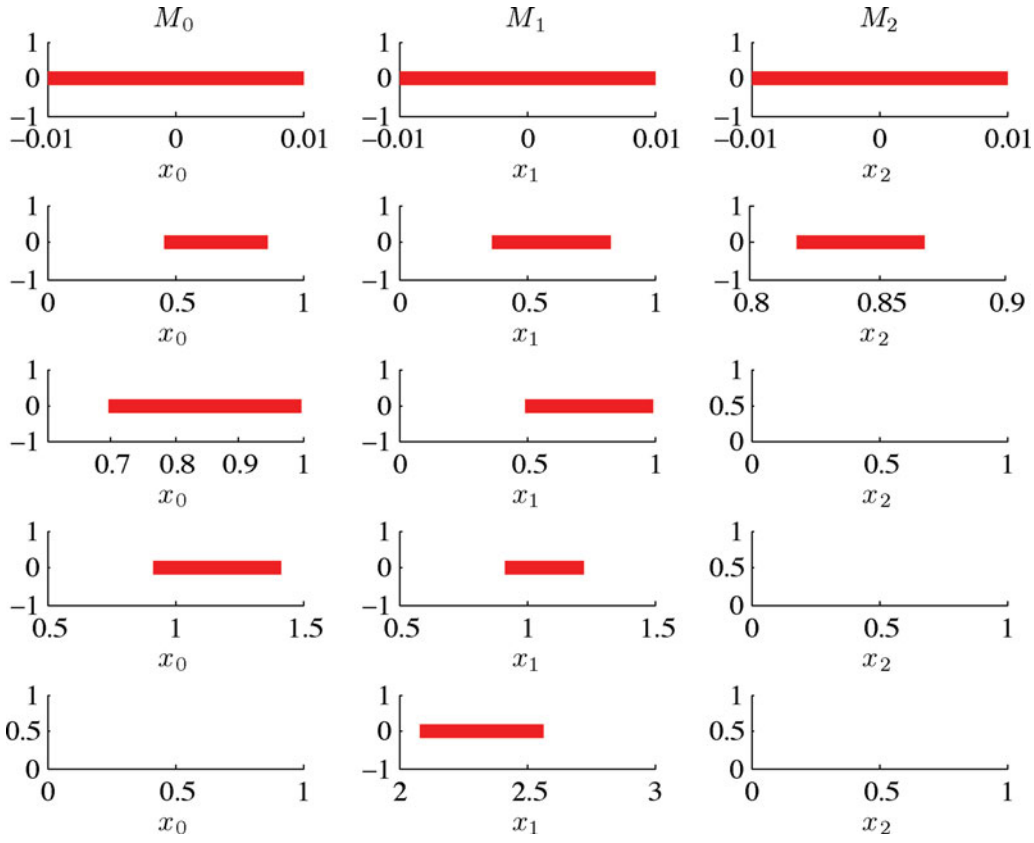


Figure 6. Evolution of $\mathcal{X}^c(k)$ for example 1 in different conditions over time. Columns show the system in different conditions: nominal system M_0, system subject to the pressure drop fault M_1, and subject to air content increase M_2. Each row represents a sample time.

$$M_0 : x_0(k+1) = 0.5x_0(k) + u(k) + w(k), \quad (32)$$

$$y_0(k) = x_0(k) + v(k),$$
$$M_1 : x_1(k+1) = 0.4x_1(k) + 0.8u(k), \quad (33)$$

$$y_1(k) = x_1(k) + v(k),$$
$$M_2 : x_2(k+1) = 0.7x_2(k) + 1.5u(k), \quad (34)$$
$$y_2(k) = x_2(k) + v(k),$$

where M_0 denotes the fault-free system, M_1 denotes the system subject to fault f_1, and M_2 denotes the system subject to fault f_2. The initial state is assumed to be in $\mathcal{X}_{i_0} = [-0.01, 0.01]$, $i = 0, 1, 2$. Also, we assume that $v(k) \in [-0.25, 0.25]$, $w(k) \in [-0.25, 0.25]$, and $\mathcal{U} = [-5, 5]$. The detection horizon, N, is chosen to be five.

We use Algorithm 4 to find the input sequence. It is assumed that the system is subject to fault f_1. Figure 6 shows an example of the evolution of the $\mathcal{X}_i^c$ over time. Rows represent sample times and each column represents the system in a condition. After two samples M_2 is falsified and after four samples M_0 is falsified; hence the fault f_1 is detected and isolated correctly.

In this simulation, for the sake of demonstration we look for a constant input in the AFDI horizon that can isolate faults. The obtained input sequence is [0.7167 0.7167 2.4562 2.4562]. This enables us to demonstrate the steps of the algorithm in three-dimensional plots after M_2 is falsified. First, we use (27) and find $P(0, 1)$ which is shown in Figure 7 in red. At time step 3, the initial state of $x_0 \in [0.698, 0.998]$ and $x_1 \in [0.498, 0.998]$. Then, the intersection of P^e and $\mathcal{U} \times \mathcal{X}_0^c \times \mathcal{X}_1^c = [-5, 5] \times [0.698, 0.998] \times [0.498, 0.998]$ is found as shown in Figure 7 in purple. This intersection is projected on u which results in $P^e = [-2.8437, 2.4562]$. From this it is obvious that $u = 2.4562$ is the optimal input. As $N = 5$, and input is assumed to be constant, applying this input guarantees isolation in five samples. However, in the simulation in Figure 6 the fault is isolated in only two samples due to noise and disturbance realisation in this example. To investigate the effect of the bounds on the noise and disturbance on the results we apply Algorithm 5 to the above example where we perform 100 Monte Carlo simulations. The noise and disturbance are generated randomly with a uniform distribution over the given interval: $[v] = [-v_M, v_M]$ and $[w] = [-w_M, w_M]$. Tables 1 and 2 show the effect of increasing the bounds on the noise and disturbance on the

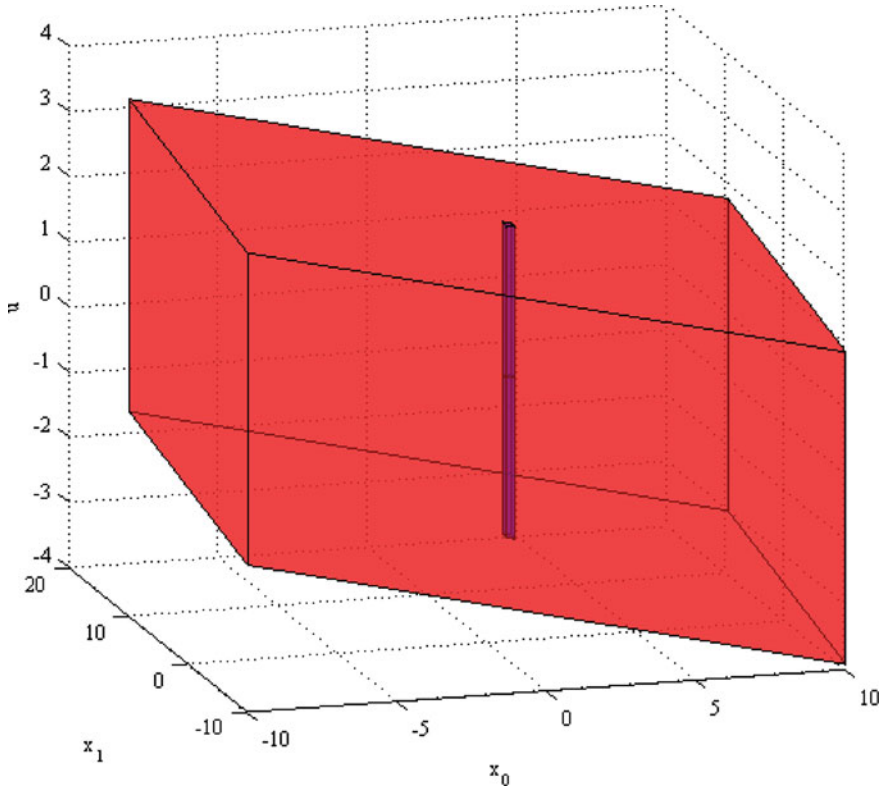


Figure 7. Expanded feasible region: P^e (red), and its intersection with $\mathcal{U} \times \mathcal{X}_0^c(3) \times \mathcal{X}_1^c(3)$ (purple).

Table 1. Effects of noise of the input.

v_M	$mean(T)$	$mean(\frac{\|u\|}{T})$
0.1	4.21	0.1793
0.2	4.26	0.298
0.3	4.32	0.4543
0.4	4.61	0.516
0.5	4.9	0.6593

energy and time required for FDI. As it is expected when the bounds increase, both the T and $\frac{\|u\|}{T}$ increase, which means that for the system with bigger bounds on the noise and disturbance we need more energy in the auxiliary input and also it takes more time to detect and isolate the fault.

Table 2. Effect of disturbance on the input with v = 0.5.

w_M	$mean(T)$	$mean(\frac{\|u\|}{T})$
0	4.9	0.6593
0.1	5.25	1.0151
0.2	5.32	1.3756
0.3	5.49	1.6652
0.4	5.81	1.8873
0.5	6.01	2.1938

5.2. *Example 2*

In this example, we demonstrate how the proposed method can be used for fault diagnosis in the pitch system of a wind turbine. A wind turbine converts a part of the kinetic energy of the wind into electrical energy. The wind turbine model consists of four parts: blade and pitch systems, drive train, generator and converter, and the controller, see Figure 8. The rotational torque due to the airflow of the wind on the rotor blades rotates the rotor. As a result, a part of wind energy is converted to mechanical energy. The drive train transmits this mechanical energy to the generator, and the generator converts the mechanical energy to electrical energy. To control the power, we can manipulate the pitch angle of the blades or the rotational speed of the rotor. The rotational speed of the generator or the rotor is controlled by the generator torque. The generator is fully coupled with a converter, which provides us with the ability to control the generator torque. The blade's pitch is controlled through the pitch actuators. The pitch actuator that is considered in this paper is a hydraulic pitch actuator. The hydraulic circuit of the pitch system is depicted in Figure 9. The flow to the two sides of the cylinder is controlled by a proportional valve. The proportional valve is controlled by a proportional feedback of the piston position error. The closed loop dynamic of the pitch system can be approximated by a second-order

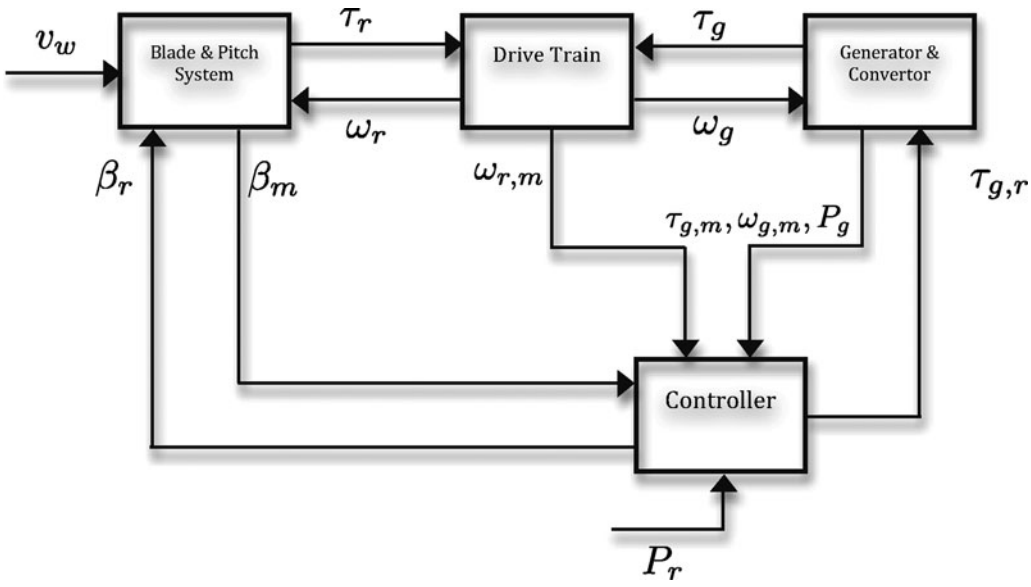


Figure 8. Block diagram of a horizontal axis wind turbine model.

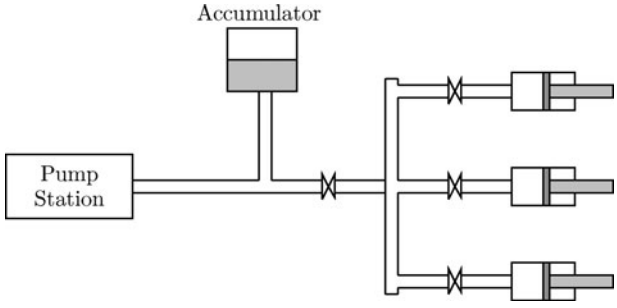


Figure 9. The hydraulic pitch system with three actuators.

system as:

$$\frac{\beta(s)}{\beta_r(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}, \tag{35}$$

where β_r is the pitch reference, see Odgaard, Stoustrup, and Kinnaert (2009) for the description of the benchmark model as well as parameters of the model. The nominal values of ξ and ω_n are given in Table 3 as ξ_0, ω_{n0}.

Faults:

Two kinds of faults for the pitch actuator are considered in this work. The first one is a pressure drop in the actuator, which changes the pressure to 50% of its nominal value. As a result of this fault, the nominal values of ξ_0, ω_{n0} change to ξ_1, ω_{n1}. The second fault is an increase of the air content in the oil in the actuator. In the normal condition, the air in the hydraulic oil is 7%. As a result of this fault, the air constant increases to 15%. This changes ξ_0, ω_{n0} to ξ_2, ω_{n2}. Parameters of the nominal and the faulty system are given in Table 3.

Table 3. Parameters of the pitch system in the normal and faulty conditions.

Fault	Parameters
No Fault	$\xi_0 = 0.6$, $\omega_{n0} = 11.11$
Pressure drop	$\xi_1 = 0.45$, $\omega_{n1} = 5.73$
Air content increase	$\xi_2 = 0.9$, $\omega_{n2} = 3.42$

Simulation results:

Now, we demonstrate how the proposed method can be used for fault diagnosis of the pitch actuator system. The set-membership methods are used in Tabatabaeipour, Odgaard, and Bak (2012a), Tabatabaeipour, Odgaard, Bak, and Stoustrup (2012b) and Casau, Rosa, Tabatabaeipour, Silvestre, and Stoustrup (2012) for fault detection of a benchmark wind turbine proposed in Odgaard et al. (2009). The fault scenarios considered in this paper are also considered in the benchmark problem. When the reference signal is constant for a period of time or in situations that the excitation of the system because of the change of the reference signal is small, the faults cannot be detected or isolated. Note that this is not a rare case as in the low wind speed the pitch reference is always kept at 0. In this situation it is useful to use AFDI to excite the pitch system periodically to check the condition of the actuator. We use the proposed AFDI method to find the optimal excitation signal for fault detection. We choose a sample time of $0.01s$. It is assumed that the measurement noise is in the interval $[-v_\beta, v_\beta]$. We would investigate the effect of bounds on the noise on the input.

Figure 10 shows the result of model falsification algorithm to the pitch actuator system. For implementation of the algorithm we have used zonotopes. The interested reader is referred to Tabatabaeipour et al. (2012b) for details of implementation. M_0 denotes the nominal system,

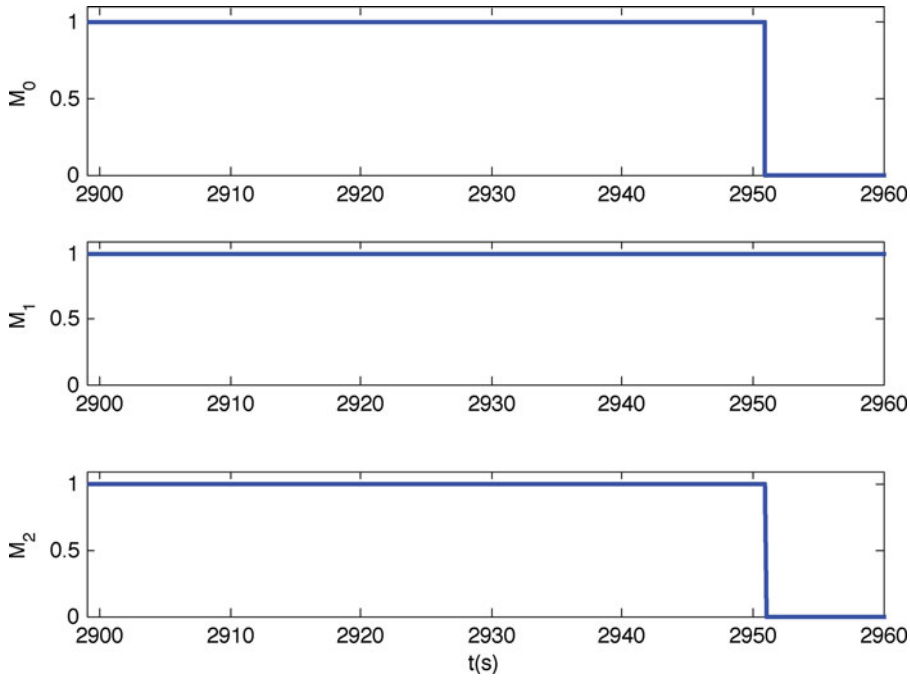


Figure 10. Simulation result of the model falsification algorithm using zonotopes.

M_1 the system subject to pressure drop faults, and M_2 the system subject to air content increase. $M_i = 1$ denotes that the *ith* model is not falsified and $M_i = 0$ denotes that the *ith* model is falsified. At $t = 2900s$ the pressure drop fault occurs. As can be seen from Figure 10, all three models are valid for more than 5000 samples after occurrence of the fault. The reason for this delay is that the system is not excited enough in this period as it can be seen in Figure 11, which shows β_r in this period.

For faster fault detection and isolation the proposed method can be applied when more than one model is valid for a period of time. Here, we apply the proposed AFDI method at $t = 2905$. Figure 12 shows an example of the evolution of the $\mathcal{X}_i^c$ over time. Rows represent sample times and each column represents the system in a condition. In this simulation v_β is chosen to be 0.5. At $t = 605.01$, $\mathcal{X}_i^c$'s are initialised using the result form of the model falsification algorithm. They are over-approximated by boxes. After three steps, $\mathcal{X}_2^c$ becomes empty and M_2 is falsified. Then, $\mathcal{M} = \{M_0, M_1\}$ and the input is updated based on this new information. The input is applied to the system and after three steps, M_0 is also falsified and the fault is isolated.

Table 4. Effect of noise on the input.

v_β	$mean(T)$	$mean(\frac{\|u\|}{T})$
0.5	5.8	10.08
0.6	6.06	12.16
0.7	5.32	14.55
0.8	6.53	15.11
0.9	6.81	17.72
1	6.94	19.26

Table 4 shows the average isolation time and 2-norm of the input for different values of the bound on the noise. For each average value 100 Monte Carlo simulations are performed. As it is expected when the level of noise increases,

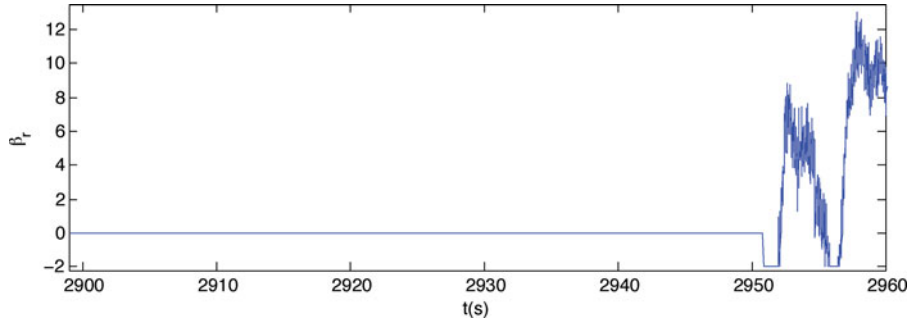


Figure 11. Reference signal for the pitch system.

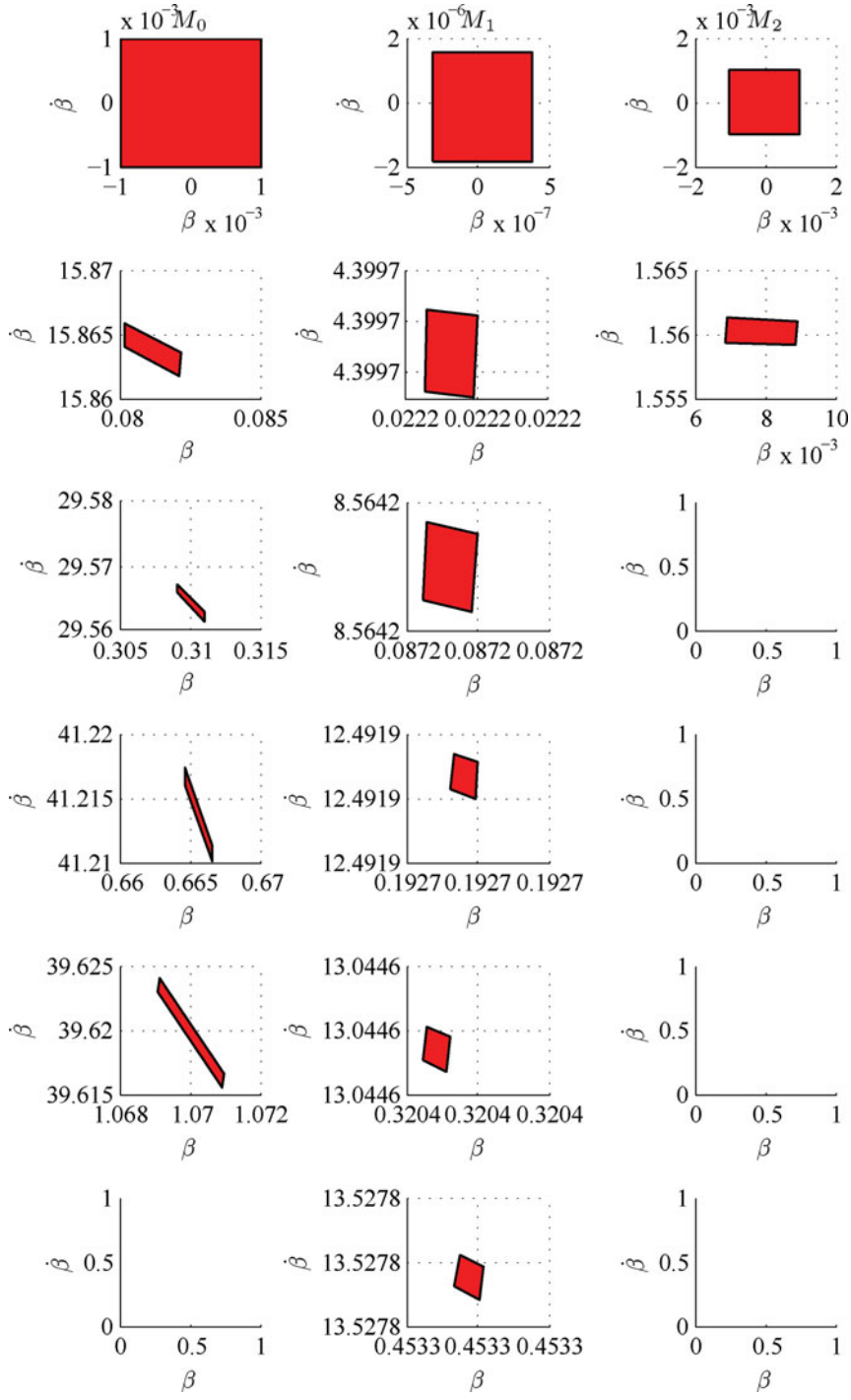


Figure 12. Evolution of $\mathcal{X}_i^c(k)$ for the pitch system in different conditions over time. Columns show the system in different conditions: nominal system M_0, the system subject to the pressure drop fault M_1, and subject to air content increase M_2. Each row represents a sample time.

the AFDI time as well as the average energy required for detection and isolation increases.

6. Conclusion

In this paper a new method for AFDI for linear time-varying systems is proposed. The method is based on using set-membership approaches for PFDI. It is assumed that noise and disturbance are unknown but bounded. Using set-membership PFDI method, the set of models that are not compatible with the input/output sequence and bounds on the noise and disturbance is falsified. When more than one model is compatible with the input/output and bounds, the AFDI is used to distinguish between un-falsified models.

The AFDI receives the set-valued estimation of the states for each model and calculates an optimal input sequence that guarantees fault detection and isolation in a finite time horizon. The algorithm uses the information about the unfalsified models and the set-valued state estimation forms the set-membership PFDI and updates the input at each iteration in a decreasing receding horizon manner. The method is demonstrated through a numerical example as well as on the pitch system of a benchmark wind turbine.

Notes on contributor

S.M. Tabatabaeipour received his PhD degree in control and automation from the Aalborg University, Denmark, in 2010. He was a post-doctoral researcher at the Section on Automation and Control, Department of Electronic systems, Aalborg University. He is currently a post-doctoral researcher at automation and control group at Technical University of Denmark. His research interests include switched systems, non-linear control, fault diagnosis and fault-tolerant control.

References

Alamo, T., Bravo, J., and Camacho, E. (2005), 'Guaranteed State Estimation by Zonotopes', *Automatica*, 41(6), 1035–1043.

Andjelkovic, I., Sweetingham, K., and Campbell, S. (2008), 'Active Fault Detection in Nonlinear Systems Using Auxiliary Signals', in *American Control Conference*, Seattle, WA, pp. 2142–2147.

Campbell, S., and Nikoukhah, R. (2004), *Auxiliary Signal Design for Failure Detection*, Princeton, NJ: Princeton University Press.

Casau, P., Rosa, P., Tabatabaeipour, S.M., Silvestre, P., and Stoustrup, J. (2012), 'Fault Detection and Fault Tolerant Control of Wind Turbines Using Set-Valued Observers', in *Proceedings of the 8th IFAC international Symposium on Fault Detection, Supervision, and Safety for Technical Processes*, August, Mexico City, Mexico, pp. 120–125.

Chen, J., and Patton, R. (1999), *Robust Model-Based Fault Diagnosis for Dynamic Systems*, Norwell, MA: Kluwer Academic Publishers.

Esna Ashari, A., Nikoukhah, R., and Campbell, S. (2012), 'Effects of Feedback on Active Fault Detection', *Automatica*, 48(5), 866–872.

Ingimundarson, A., Bravo, J.M., Puig, V., Alamo, T., and Guerra, P. (2009), 'Robust Fault Detection Using Zonotope-Based Set-Membership Consistency Test', *International Journal of Adaptive Control and Signal Processing*, 23(4), 311–330.

Niemann, H.H. (2006), 'A Setup for Active Fault Diagnosis', *IEEE Transactions on Automatic Control*, 51(9), 1572–1578.

Niemann, H.H., and Poulsen, N.K. (2005), 'Active Fault Diagnosis in Closed-Loop Systems', in *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, pp. 448–453.

Nikoukhah, R., and Campbell, S.L. (2006), 'Auxiliary Signal Design for Active Failure Detection in Uncertain Linear Systems With a Priori Information', *Automatica*, 42(2), 219–228.

Nikoukhah, R., and Campbell, S. (2008), 'On the Detection of Small Parameter Variations in Linear Uncertain Systems', *European Journal of Control*, 14(2), 158–171.

Nikoukhah, R., Campbell, S.L., Savkin, A., and Selmic, R. (2005), 'A Multi-Model Approach to Failure Detection in Uncertain Sampled-Data Systems', *European Journal of Control*, 11(3), 255–268.

Odgaard, P., Stoustrup, J., and Kinnaert, M. (2009), 'Fault Tolerant Control of Wind Turbines: A Benchmark Model', in *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, June–July, Barcelona, Spain: IFAC, pp. 155–160.

Olaru, S., De Doná, J.A., and Seron, M.M. (2010), 'Positive Invariant Sets for Fault Tolerant Multisensor Control Schemes', *International Journal of Control*, 83(12), 2622–2640.

Patton, R., and Chen, J. (1991a), 'Robust Fault Detection of Jet Engine Sensor Systems Using Eigenstructure Assignment', in *AIAA Guidance, Navigation and Control Conference*, New Orleans, LA, pp. 1666–1675.

Patton, R., and Chen, J. (1991b), 'A Review of Parity Space Approaches to Fault Diagnosis', in *Proceedings of 1st IFAC Symposium on Fault Detection, Supervision, and Safety of Technical Processes*, September, Baden-Baden, Germany, pp. 65–81.

Puig, V. (2010), 'Fault Diagnosis and Fault Tolerant Control Using Set-Membership Approaches: Application to Real Case Studies', *International Journal of Applied Mathematics and Computer Science*, 20(4), 619–635.

Rosa, P.A.N., Casau, P., Silvestre, C., Tabatabaeipour, S.M., and Stoustrup, J. (2012), 'A Set-Valued Approach to FDI and FTC: Theory and Implementation Issues', in *Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, August, Mexico City, Mexico.

Seron, M.M., Zhou, X.W., De Doná, J.A., and Martinez, J.J. (2008), 'Multisensor Switching Control Strategy With Fault Tolerance Guarantees', *Automatica*, 44(1), 88–97.

Stoican, F., Olaru, S., Seron, M.M., and De Doná, J.A. (2012), 'Reference Governor Design for Tracking Problem With Fault Detection Guarantees', *Journal of Process Control*, 22(2), 829–836.

Stoustrup, J., and Niemann, H.H. (2010), 'Active Fault Diagnosis by Controller Modification', *International Journal of Systems Science*, 41(8), 925–936.

Tabatabaeipour, S. (2010), *Fault Diagnosis and Fault Tolerant Control of Hybrid Systems*, Aalborg: Aalborg University.

Tabatabaeipour, S., Izadi-Zamanabadi, R., Bak, T., and Ravn, A.P. (2009b), 'Automatic Sensor Assignment of a Supermarket Refrigeration System', in *IEEE Multi-Conference on Control Applications, (CCA) & Intelligent Control, (ISIC)*, July, pp. 1319–1324.

Tabatabaeipour, S.M., Odgaard, P.F., and Bak, T. (2012a), 'Fault Detection of a Benchmark Wind Turbine Using Interval Analysis', in *Proceedings of the American Control Conference*, June, Montreal, Canada, pp. 4387–4392.

Tabatabaeipour, S.M., Odgaard, P.F., Bak, T., and Stoustrup, J. (2012b), 'Fault Detection of Wind Turbines With Uncertain Parameters: A Set-Membership Approach', *Energies*, 5(7), 2224–2248.

Tabatabaeipour, S., Ravn, A.P., Izadi-Zamanabadi, R., and Bak, T. (2009a), 'Active Fault Diagnosis of Linear Hybrid Systems', in *Proceedings of 7th IFAC Symposium on Fault Detection, Supervision, and Safety of Technological Processes*, June–July, Barcelona, Spain, pp. 211–216.

Appendix. Description of feasible regions

In this appendix a description of the polytope of feasible region of (22) is given. For simplicity of notation we consider models M_1 and M_2. The symbol $\mathbf{Z}_N^i(k_0+n)$ denotes the vector $\left[\, z_i^T(k_0+n) \cdots z_i^T(k_0+N) \,\right]^T$. The feasible region is described by:

$$\mathbf{Y}_N^1(k_0+1) = \mathbf{Y}_N^2(k_0+1), \tag{A.1}$$

where:

$$\mathbf{Y}_N^i(k_0+1) = \mathcal{Q}_i x_i(k_0) + \mathcal{R}_i \mathbf{U}_{N-1}(k_0) + \mathcal{S}_i \mathbf{W}_{N-1}^i(k_0) + \mathbf{V}_N^i(k_0+1), \tag{A.2}$$

where

$$\mathbf{Q}_i = \begin{bmatrix} C_i(k_0+1)A_i(k_0) \\ \vdots \\ C_i(k_0+N)\prod_{j=1}^{N} A_i(k_0+N-j) \end{bmatrix}, \tag{A.3}$$

$$\mathcal{R}_i = \begin{bmatrix} C_i(k_0+1)B_i(k_0) & 0 & \cdots & 0 \\ C_i(k_0+2)A_i(k_0+1)B_i(k_0) & C_i(k_0+2)B_i(k_0+1) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ C_i(k_0+N)(\prod_{j=1}^{N-1} A_i(k_0+N-j))B_i(k_0) & C_i(k_0+N)(\prod_{j=1}^{N-1} A_i(k_0+N-2))B_i(k_0+1) & \cdots & C_i(k_0+N)B_i(k_0+N-1) \end{bmatrix}, \tag{A.4}$$

$$\mathcal{S}_i = \begin{bmatrix} C_i(k_0+1) & 0 & \cdots & 0 \\ C_i(k_0+2)A_i(k_0+1) & C_i(k_0+2) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ C_i(k_0+N)(\prod_{j=1}^{N-1} A_i(k_0+N-j)) & C_i(k_0+N)(\prod_{j=1}^{N-1} A(k_0+N-2)) & \cdots & C_i(k_0+N) \end{bmatrix}. \tag{A.5}$$

Then, (A.1) is rewritten as:

$$\mathcal{Q}_1 x_1(k_0) + \mathcal{R}_1 \mathbf{U}_{N-1}(k_0) + \mathcal{S}_1 \mathbf{W}_{N-1}^1(k_0) - \mathcal{Q}_2 x_2(k_0) - \mathcal{R}_2 \mathbf{U}_{N-1}(k_0) - \mathcal{S}_2 \mathbf{W}_{N-1}^2(k_0) = \mathbf{V}_N^2(k_0+1) - \mathbf{V}_N^1(k_0+1). \tag{A.6}$$

But since it is assumed that $v_i(k) \in \mathcal{V}_i$, then we have:

$$v_2(k) - v_1(k) \in \mathcal{V}_2 \oplus (-\mathcal{V}_1).$$

If $\mathcal{V}_1 \oplus (-\mathcal{V}_2)$ is given as $H_v v \le K_v$, then:

$$\mathcal{H}_v \left(\begin{bmatrix} \mathcal{Q}_1 & -\mathcal{Q}_2 \end{bmatrix} \begin{bmatrix} x_1(k_0) \\ x_2(k_0) \end{bmatrix} + \begin{bmatrix} \mathcal{R}_1 & -\mathcal{R}_2 \end{bmatrix} \mathbf{U}_{N-1}(k_0) + \begin{bmatrix} \mathcal{S}_1 & -\mathcal{S}_2 \end{bmatrix} \begin{bmatrix} \mathbf{W}_{N-1}^1(k_0) \\ \mathbf{W}_{N-1}^2(k_0) \end{bmatrix} \right) \le \mathcal{K}_v, \tag{A.7}$$

where $\mathcal{H}_v = \mathrm{diag}(H_v, \ldots, H_v)$ and $\mathcal{K}_v = \left[\, K_v^T \ \ldots \ K_v^T \,\right]^T$. Overall, the polytope of feasible region is given by:

$$\begin{cases} (A.7) \\ x_i(k_0) \in \mathcal{X}_i^c(k_0),\ i = 1, 2 \\ w_i(k) \in \mathcal{W}_i,\ i = 1, 2,\ k = k_0, \ldots, k_0 + N - 1 \end{cases}. \tag{A.8}$$