Active fault diagnosis of linear hybrid systems

Tabatabaeipour, Seyed Mojtaba; Ravn, Anders P.; Izadi-Zamanabadi, Roozbeh; Bak, Thomas

Published in:
Elsevier IFAC Publications / IFAC Proceedings series

Publication date:
2009

Document Version
Publisher's PDF, also known as Version of record

Proceedings of the 7th IFAC Symposium on
Fault Detection, Supervision and Safety of Technical Processes
Barcelona, Spain, June 30 - July 3, 2009

Active Fault Diagnosis of Linear Hybrid Systems

S. Tabatabaeipour * **A.P. Ravn** ** **R. Izadi-Zamanabadi** ***
T. Bak *

* *Department of Electronic Systems, Aalborg University, DK-9220, Denmark, (e-mail: {smt,tba}@es.aau.dk)*
** *Department of Computer Scince, Aalborg University, DK-9220, Denmark, (e-mail: apr@cs.aau.edu)*
*** *Central R&D - RA-DP, Danfoss A/S, DK-6470, Denmark, (e-mail:roozbeh@danfoss.com)*

Abstract: A method for active fault diagnosis of linear discrete time hybrid systems is presented. The algorithm generates appropriate test signals that can be used for sanity check during system commissioning or later in the normal phase to detect faults which are impossible to detect by means of passive diagnosis methods because of regulatory actions of the controller. The algorithm is illustrated on a two tank benchmark example.

1. INTRODUCTION

In complex large control systems there are many components with strong interaction between them. Hence the overall system depends crucially on the individual performance of the components. Therefore a fault in a single component may degrade the overall performance of the system and may even leads to unacceptable loss of system functionality. Thus fault diagnosis is of crucial importance in automatic control of complex large systems.

There are two main approaches to fault diagnosis: active and passive. In the passive approach the diagnoser observes the input and output of the system and based on the measured I/O decides whether a fault has occurred or not. Most of the available methods for fault diagnosis are of this kind.

In active fault diagnosis the diagnoser generates a test signal which excites the system to decide whether the system represents the normal behaviour or the faulty behaviour and if possible decides which faulty behaviour occurs. The test signal should be designed such that it affects the overall system as little as possible although enough to make fault diagnosis possible. The advantage of the active approach is in the operating points where the normal system and faulty system represents the same behaviour. Under such circumstances it is possible to detect faults faster by active diagnosis. Active fault diagnosis can also be used to provide sanity check in the commissioning phase by generating an appropriate test signal.

Modelling of complex systems are captured by hybrid system theory, which has been subject of intensive research in recent years, for an overview see Antsaklis and Koutsoukos [2003]. Generally speaking, a hybrid system is a dynamical system with both continuous and discrete behaviours and non-trivial interaction between continuous evolutions and discrete transitions.

Fault diagnosis of hybrid systems has been investigated recently, for a survey one can look at Mohammadi et al. [2007], Narasimhan and Biswas [2007], Zhao et al. [2005]. A class of approaches for diagnosis of hybrid systems uses discrete/temporal abstraction of the continuous dynamics Lunze [2000]. In Mohammadi et al. [2007], the diagnoser uses a discrete event abstraction of the system and the continuous dynamics information is used when it becomes necessary. In Zhao et al. [2005], the authors use a Petri net abstraction for dealing with continuous behaviour of hybrid systems. In Narasimhan and Biswas [2007] a model based diagnosis method based on a hybrid bond graph modelling framework is proposed. Particle filtering methods are another class of methods for diagnosis of hybrid systems; Koutsoukos et al. [2003], Hofbaur and Williams [2002].

All of the aforementioned approaches are in the area of passive diagnosis. In Campbell and Nikoukhah [2004], Niemann [2006] the problem of active diagnosis for linear system using an auxillary signal for fault detection is investigated. The results of Campbell and Nikoukhah [2004] is extended for nonlinear systems in Andjelkovic et al. [2008] using linearization and also a direct optimization approach. In the field of discrete event systems, some approaches have been proposed for active diagnosis. Active diagnosis of DES is studied in Lin [1994] and input sequence for diagnosis is computed. Sampath et al. [1998] studied the active diagnosis problem of DES as a supervisory control problem.

To the knowledge of the authors there is no research considering directly active fault diagnosis of hybrid systems. In this paper an active fault diagnosis method for diagnosis of linear hybrid system in discrete time is proposed. The idea is based on reach set computations for the faulty and the normal system. For both systems, those states that the system can reach in forthcoming steps considering all possible excitations are considered. Reach sets are computed as long as the faulty system and the normal system have

the same reach sets. But as soon as they represent different sets the algorithm terminates and selects a point which uniquely belongs to one of the sets. Then the optimal input for reaching the selected point is calculated and injected to the system. If the system reaches the selected point then it is in the corresponding mode, otherwise it is in the other mode.

This paper is organized as follows. An outline of the approach and some preliminaries are given in Section 2. Section 3 describes the algorithm. In Section 4 the proposed algorithm is applied to the two tank benchmark example. Section 5 provides conclusions and topics for future investigation.

2. OUTLINE OF THE METHOD

Most model-based diagnostic methods follow the same principle Blanke et al. [2006]. They observe a sequence of measured input and output of the system and decide whether the measured I/O pair is consistent with the model that describes the behaviour of the system. If the consistency is not confirmed a fault is detected.

Suppose that the current observed I/O pair is A or B as depicted in Fig. 1. The set $\mathbf{B}_0$ represents the normal behaviour of the system and the set $\mathbf{B}_1$ represents the behaviour of the system subject to the fault f_1. As long as A or B belong uniquely to the sets $\mathbf{B}_0$ or $\mathbf{B}_1$, the diagnoser can decide whether the system is in its normal operation or subject to a fault. The ambiguity arises when the observed data is the I/O pair C, which belongs to the area where the normal behaviour and the faulty behaviour of the system overlap. In this case, the diagnoser can not distinguish if the system is subject to the fault f_1 or in the normal operation. The main idea of active fault diagnosis is to exert an input signal to the system to move C to an area which belongs uniquely either to the set $\mathbf{B}_0$ or $\mathbf{B}_1$.

The active diagnosis algorithm in this paper assumes that we have a model of the normal and the faulty system. From current state, we predict the behaviour of the system at future time steps considering all possible inputs and using both models. We then find the first time step that the faulty and the normal system have different behaviours. Now consider the set holding these different behaviours. We choose one of them, e.g. belonging to future behaviour of the normal system. Then we find an optimal input sequence that will drive the system to a state corresponding to the selected behaviour and apply it to the system. If the output of the system reaches the corresponding output of the selected behaviour, then the system is in the normal mode otherwise it is faulty.

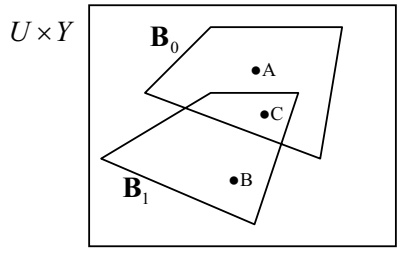

Fig. 1. System behaviour

In order to make this idea precise, we define following terms.

Definition 1. (Hybrid Automaton). A *hybrid automaton,* $\mathcal{H}$ is a collection $\mathcal{H} = (Q, X, U, Y, Init, f, h, Inv, E, G, J)$ where,

- Q is a set of finite discrete modes, $Q = \{q_1, q_2, \ldots, q_m\}$,
- X is a finite set of continuous state variables,
- U is a finite collection of input variables,
- Y is a finite collection of output variables,
- $Init \subset Q \times X$ is a set of initial states,
- $f : Q \times X \times U \to \mathbb{R}^n$ is a vector field,
- $h : Q \times X \times U \to Y$ is an output map,
- $Inv : Q \to 2^{X \times U}$ assigns to each $q \in Q$ an invariant set $Inv(q) \subseteq X \times U$,
- $E \subset Q \times Q$ is a set of discrete transitions,
- $G : E \to 2^{X \times U}$ assigns to each $e = (q, q') \in E$ a guard $g(e) \subset X \times U$,
- $J : E \times X \times U \to 2^X$ is a jump function that assigns a jump set $J(e, x, u) \subseteq X \times U$ to each pair $e \in E$ and $x \in g(e)$.

In the case of linear hybrid systems the vector field f_q is represented by a linear difference equation: $x_{i+1} = A_{q_i} x_i + B_{q_i} u_i$ and and the output map is of the form $y_{i+1} = C_{q_i} x_i + D_{q_i} u_i$.

The tuple $(q, x, u, y) \in Q \times X \times U \times Y$ is called a point of $\mathcal{H}$, $(q, x) \in Q \times X$ is called the state of $\mathcal{H}$, $u \in U$ the input and $y \in Y$ is called the output of $\mathcal{H}$. Also we refer to $(u, y) \in U \times Y$ as an observation of $\mathcal{H}$.

Definition 2. (Execution). An execution of a hybrid automaton is a sequence $\chi = (\sigma_0, \ldots, \sigma_i, \sigma_{i+1}, \ldots)$ where $\sigma_0 = (q_0, x_0, u_0, y_0)$, $\sigma_i = (q_i, x_i, u_i, y_i)$ and $\sigma_{i+1} = (q_{i+1}, x_{i+1}, u_{i+1}, y_{i+1})$ such that:

- Initial condition $(q_0, x_0) \in Init$,
- Continuous evolution: for all i, $q_i = q_{i+1}$, $(x_{i+1}, u_{i+1}) \in Inv(q_i)$:
$$x_{i+1} = A_{q_i} x_i + B_{q_i} u_i$$
$$y_{i+1} = C_{q_i} x_i + D_{q_i} u_i$$
- Transition: for all i, $e = (q_i, q_{i+1}) \in E, (x_i, u_i) \in G(e) : x_{i+1} \in J(e, x_i, u_i), (x_{i+1}, u_{i+1}) \in Inv(q_{i+1})$

For modelling of faults in hybrid systems two types of faults can be considered: discrete faults and continuous faults. Discrete faults can be considered as a new mode or location in a hybrid system. Here continuous faults are also modelled as a new mode as in Mohammadi et al. [2007]. It is supposed that events that describe transitions from a normal location to a faulty location are unobservable events. The system can be in a normal condition N or a faulty condition F where each condition is a subset of Q. A condition set $K = \{N, F_1, \ldots, F_p\}, p > 1$ is a set of conditions that is a complete partition of the mode set Q.

For every condition $\kappa \in K$, the corresponding dynamical system, Σ_κ, is denoted by:

$$\Sigma_\kappa = \{\kappa, X, U, Y, Init, f, Inv, E_\kappa, G, J\}$$

where $E_\kappa = \{e = (q, q') \,|\, q \in \kappa, q' \in \kappa\}$ and $Init_\kappa \subset \kappa \times X$.

3. THE PROPOSED ALGORITHM

The diagnoser is a system that gives us an estimate $\hat{\kappa}(k)$ of the current system condition $\kappa(k)$. A passive diagnoser

recieves a sequence of observations $\langle(u(k-m), y(k-m)), \ldots, (u(k), y(k))\rangle$ as input and generetes an estimate of the current condition $\hat{\kappa}(k)$ as output. The excitation signal or the input comes from the controller.

In active diagnosis the diagnoser generates an input sequence $\langle u(k+1), \ldots, u(k+m')\rangle$, applies it to the system and observes the output sequence $\langle y(k+1), \ldots, y(k+m')\rangle$ to determine the system condition. The output of the diagnoser could be an estimate of the current condition of the system $\hat{\kappa}(m')$ or the condition of the system for some finite transition into the past $\hat{\kappa}(m'-k')$. So, the active diagnosis problem can be defined as follows:

Problem 3. (Active diagnosis problem). Given a hybrid automaton $\mathcal{H}$, find a sequence of input $\langle u(0), \ldots, u(m)\rangle$ such that the condition $\kappa(0)$ is determined by observing the sequence $\langle y(0), \ldots, y(m)\rangle$.

If the input sequence exists, then we can ask for the optimal one, where optimality can be interpreted in different senses. The algorithm that is proposed in this paper looks for the shortest sequence of the inputs that can diagnose the system.

Here, it is supposed that an observer-based passive diagnoser for the hybrid system is designed which gives us the initial state of the system. For detailed description of this diagnoser, the interested reader is referred to Balluchi et al. [2002] and Mohammadi et al. [2007]. But briefly, the diagnoser consists of a bank of observers, each one designed for a discrete mode q_i of the hybrid system. The inputs of the observers are a sequence of observations (u, y). Based on the output of the observers, a residual vector $\rho = \{r_1, \ldots r_m\}$ is generated. A zero residual r_i shows that the corresponding mode, q_i, is consistent with the input and output sequence. If the current state of the system, $(q(k), x(k))$, is determined uniquely then the condition is also determined . A problem arises when both the faulty mode and the normal mode are recognized as consistent modes with the I/O sequence. This is because these two modes have indistinguishable executions, where indistinguishable executions are defined as follows.

Definition 4. (Indistinguishability). Given a hybrid system $\mathcal{H}$ and $\delta \in \mathbb{N}$, modes q and q' are indistinguishable on the time interval $[i, i+\delta]$ if there exist executions $\chi = (\sigma_i, \ldots, \sigma_{i+\delta})$ and $\chi' = (\sigma'_i, \ldots, \sigma'_{i+\delta})$, where the corresponding continuous outputs are equal.

This problem may happen very often. Consider a simple hybrid system with two discrete modes q_1 an q_2 and a switch between them (like an on/off valve) which forces the system to switch between these two modes. If the switch is stuck in one position, say, such that the mode q_1 is active, then the faulty system has exactly the same properties as the mode q_1. Therefore, the faulty mode and q_1 are indistinguishable. An advantage of our method is that there is no need for modelling efforts to make these two modes distinguishable.

3.1 The Algorithm for a system with one faulty mode

In this subsection, the proposed algorithm is described for a system with one faulty mode. Therefore, the condition set is $\{N, F\}$. The possibility for expansion of the method for more than one faulty mode is discussed in the next subsection.

The idea of the proposed method is to find two executions χ_1 and χ_2 respectively from the system in normal conditon, Σ_N, and the faulty system, Σ_F, which are distinguishable. This task is done by finding all possible outputs that both systems could reach in the future time steps considering all admissible inputs and starting from the given initial state. As soon as they represent different outputs then the required executions are found and the algorithm terminates.

To find all possible outputs that a system could reach in the future, reach set of the system and the corresponding output should be computed.

Definition 5. (Reach Set). *Reach Set* of a hybrid automata $\mathcal{H}$ at time k denoted by $Reach_k(\mathcal{H}, \mathcal{X}(0), \mathcal{U})$ is the set of all states $(q, x) \in Q \times X$ that are reachable by a given hybrid automata $\mathcal{H}$ at time step k, starting from any initial state $x(0) \in \mathcal{X}(0)$ and with all possible inputs $u \in \mathcal{U}$.

As described in Algorithm 1, the reach set of both the normal system, $\mathcal{R}_{N_k}$, and the faulty system, $\mathcal{R}_{F_k}$, are calculated for time k. It is assumed that the area of tolerable performance is given by the set $\mathcal{T}$. The area of untolerable performance is excluded from the reach sets. The corresponding outputs are denoted by $Y(\mathcal{R}_{N_k})$ and $Y(\mathcal{R}_{F_k})$. If the set $\Delta_k = (Y(\mathcal{R}_{N_k}) \cup Y(\mathcal{R}_{F_k}))\backslash(Y(\mathcal{R}_{N_k}) \cap Y(\mathcal{R}_{F_k}))$ is not empty then there exist distinguishable executions in the time interval $[0, k]$. The set Δ_k is called the *discriminating set*. Now, there are two possible ways to determine the condition of the system. Assume that at time $k = K_{max}$ the discriminating set $\Delta_{K_{max}} \neq \emptyset$. It can be assumed that the system at time 0 is in the Normal condition. We choose a point which uniquely belongs to the future behaviour of the normal system i.e $\tilde{y}(K_{max}) \in (\Delta_{K_{max}} \cap Y(\mathcal{R}_{N_{K_{max}}}))$. After choosing the point, the optimal input to reach $\tilde{y}(K_{max})$ is computed and applied to the system. If $y(K_{max}) = \tilde{y}(K_{max})$ then system is in the normal condition otherwise it is in the faulty condition. Since the termination of the algorithm is not guaranteed, K_{max} may not exist. For practical applications a bound β on K_{max} is set. If the algorithm does not terminate after β steps, it is recognized as indiagnosable by this method.

Another strategy is to assume that $\kappa(0) = F$ and choose $\tilde{y}(K_{max}) \in (\Delta_{K_{max}} \cap Y(\mathcal{R}_{F_{K_{max}}}))$. If $y(K_{max}) = \tilde{y}(K_{max})$ then the system is in faulty condition otherwise it is the normal condition. In Algorithm 1, the first strategy is chosen.

In the case of linear systems, having the convex polyhedral of $\mathcal{X}(0)$,$\mathcal{U}$, the reach set can be computed as:

$$Reach(\Sigma, \mathcal{X}(0), \mathcal{U}) = A\mathcal{X}(0) \oplus B\mathcal{U}, \qquad (1)$$

where $\oplus$ is the geometric or Minkowski sum. The first part considers the effect of the autonomous part of the system, $x(k+1) = A_q x(k)$, which is computed as mapping of the convex set $\mathcal{X}(0)$ through the matrix A. Because the mapping of a convex set by a linear transformation yields a convex set, the resulting set is also convex. Similarly, in the second part of (1) the effect of input is computed by mapping the set $\mathcal{U}$ by matrix B which again results in a convex set. Finally, the reach set is computed as

Table 1. Active Fault Diagnosis

Algorithm 1
Given $x_0, \beta, \Sigma_N, \Sigma_F, (\Sigma_N \neq \Sigma_F)$
Find condition κ
$k = 0, I = x_0, \mathcal{R}_{N_0} = \mathcal{R}_{F_0} = x_0$
Repeat
$\mathcal{R}_{N_k} = Reach(\Sigma_N, \mathcal{R}_{N_{k-1}}, U)$
$\mathcal{R}_{F_k} = Reach(\Sigma_F, \mathcal{R}_{F_{k-1}}, U)$
$\mathcal{R}_{N_k} = \mathcal{R}_{N_k} \cap \mathcal{T}$
$\mathcal{R}_{F_k} = \mathcal{R}_{F_k} \cap \mathcal{T}$
$I = Y(\mathcal{R}_{N_k}) \backslash Y(\mathcal{R}_{F_k})$
$k = k + 1$
Until $(I \neq \emptyset \vee k > \beta - 1)$
$K_{max} = k$
IF $I = \emptyset$
The fault F is undiagnosable
Else
Solve the optimization problem
$\min_{\mathbf{u}_{K_{max}}} J(\mathbf{x}_{K_{max}}, \mathbf{u}_{K_{max}}, \mathbf{y}_{K_{max}})$
s.t. $\begin{cases} \Sigma_N \\ x_o = x_0, x_f \in Y^{-1}(I) \end{cases}$
Apply $\mathbf{u}_{K_{max}}$ **to the system**
IF $y_{K_{max}} \in Y(I)$ **Then** $\kappa = N$ **Else** $\kappa = F$

the Minkowski sum of these two sets. For computational reasons, the representation used for the reach set and input set consists of sets which are closed under linear transformation and Minkowski sum such as polytopes, ellipsoids or zonotopes Girard et al. [2006].

In the case of hybrid systems, as is shown in Algorithm 2 in Table. 2, enabled transitions should also be considered and the corresponding jump functions should be applied. Note that in general the reach set could be nonconvex and disconnected *i.e.* a finite union of p disconnected convex polytopes. In this case it is enough to apply the above algorithm to each polytope separately and at the end calculate the union of results.

The cost function $J(\mathbf{x}_k, \mathbf{u}_k, \mathbf{y}_k)$ is the same as the cost function for the controller, which can have the following from:

$\Sigma_{k=0}^{K_{max}} \|y(t+k) - r(k)\| + \|u(t+k) - u_r(k)\| + \|x(t+k) - x_f\|$,

where $r(k)$ is the output reference signal, $u_r(k)$ the input reference signal and x_f is the final desired state.

Table 2. Reach Set Computation

Algorithm 2
Given $\mathcal{H}, \mathcal{R}_k, \mathcal{U}$,
$\mathcal{R} = \emptyset, \mathcal{R}_G = \emptyset$
$Q_R = \{q | (q, x) \in \mathcal{R}_k \}$
For all $q \in Q_R$
$X = \{x | (q, x) \in \mathcal{R}_k\}$
$X_q = X \cap Inv(q)$
$\mathcal{R}_X = A_q X_q \oplus B_q \mathcal{U}$
$E_q = \{e | e = (q, q') \in E\}$
For all $e \in E_q$
$G_e = X \cap g(e)$
$G_{e_{trans}} = execute\_transition(G_e)$
$\mathcal{R}_{G_e} = A_{q'} G_{e_{trans}} \oplus B_{q'} \mathcal{U}$
$\mathcal{R}_G = \mathcal{R}_G \cup \mathcal{R}_{G_e}$
End
$\mathcal{R} = \mathcal{R} \cup \mathcal{R}_G \cup \mathcal{R}_{G_e}$
End
$Reach_{k+1}(\mathcal{H}, \mathcal{R}_k, \mathcal{U}) = \mathcal{R}$

In the above formulation we have assumed that the system is in the normal condition and therefore the optimization problem is solved by constraining the variables to the hybrid dynamic of the normal system Σ_N. In the case which the system is in the faulty condition and it is not possible to remain in the area of required performance, it is required that the system will remain in a region of tolerable performance. Suppose that the area of tolerable performance is described by the polytope $\mathcal{T} = \{x \in \mathbb{R}^n | \mathcal{P}x \leq \mathcal{M}\}$.To ensure that system states will still remain in the polytope of the tolerable performance, the following constraints should be added to the optimization problem: $\{\mathcal{P}x(i) \leq \mathcal{M}\}_{i=k+1}^{k+K_{max}}$.

Since we have supposed that there exist an observer that gives us the current state at each time, this new information can be used in the algorithm. Suppose that at time $k-1$ the algorithm starts with x_{k-1}. At time k, the information that the diagnoser is using for predicting the behaviour of the system at time $k+1$ is the polytope $Reach_k(\Sigma, x_{k-1}, U)$. While the information from the observer for the current state is more exact. So based on this information, the diagnoser can predict the future behaviour of the system more precisely. Therefore, the overall algorithm can be described as follows. At each time step the output of the observer is given as the input to the main algorithm as described in Algorithm 1. When the optimal input sequence $u(k), \ldots, u(k + K_{max})$ is computed only the first element,$u(k)$, is applied to the system. The overall procedure repeats until $K_{max} = 1$, which means that only in one step it is possible to find the point that uniquely belongs to the normal predicted behaviour of the system. The diagnoser applies the optimal input to the system and then the status of the system can be determined. The modified version of the algorithm is more computationally demanding but it can diagnose the fault faster because it also uses available information from the observer.

3.2 Extension for more than one faulty mode

The algorithm can be extended as follows when there is more than one faulty mode. At first, the algorithm tries to choose a state that its corresponding output uniquely belongs to one of the sets $Y(\mathcal{R}_{\Sigma_\kappa k}), \kappa \in K$. Then the optimal input for driving the system to the chosen state is applied to the system. If the system could reach the target state then it is in the condition κ. Otherwise if the current output is consistent with just one of the modes then the corresponding condition is the system condition. But if it is consistent with more than one mode, then the same procedure should be repeated for these modes starting from the new initial condition.

4. EXAMPLE

The proposed method is tested on the two tank system depicted in Fig. 2. The system consists of two cylindrical tanks with cross sectional area A. These two tanks are connected together by two pipes at the bottom and at level h_v. The flows through the pipes, denoted by $Q_{12}V_{12}$ and $Q_{12}V_1$, are controlled using two on/off valves V_{12} and V_1. There is a flow Q_1 through a pump to tank 1 which is a continuous input. Dynamical equations of the system is described as follows.

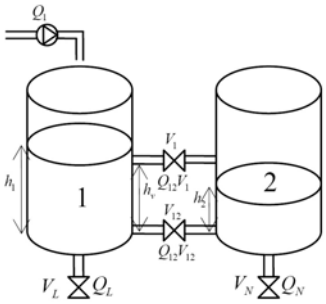

Fig. 2. Two-tank system

$$\dot{h}_1 = \frac{1}{A}(Q_1 - Q_{12}V_{12} - Q_{12}V_1 - Q_L), \quad (2)$$

$$\dot{h}_2 = \frac{1}{A}(Q_{12}V_{12} + Q_{12}V_1 - Q_N), \quad (3)$$

where h_1 and h_2 denote the levels of tanks 1 and 2 respectively. The flow $Q_{12}V_{12}$ is described as:

$$Q_{12}V_{12} = V_{12}k_{12}sign(h_1 - h_2)\sqrt{2g\,|h_1 - h_2|},$$

where g is the gravity constant and k_{12} is a constant. similarly $Q_L = V_L k_L \sqrt{2gh_1}$ and $Q_N = V_N k_N \sqrt{2gh_2}$. The flow trough valve V_1 is described by:

$$Q_{12}V_1 = V_1 k_1 sign(max\{h_v, h_1\} - max\{h_v, h_2\}) \sqrt{|2g(max\{h_v, h_1\} - max\{h_v, h_2\})|}$$

In order to apply the reach set computation algorithm to the above system, the dynamic of the system should be described as a discrete time linear hybrid system. This task is done in three steps. First, four discrete modes corresponding to four combinations of binary inputs are generated. In each of these modes the governing equations are obtained by substituting the corresponding values of binary inputs. The system switches between these four discrete modes based on the binary input vector $V = [V_{12}, V_1]$. Then, the nonlinear relation $\sqrt{x}$ is approximated by a straight line x. The resulting equations are piecewise affine. Finally, differential equations 2, 3 are discretized in time by Euler approximation $\dot{h}_i(t) \approx \frac{h_i(t+1) - h_i(t)}{Ts}$, where T_s is sample time.

To compute $\mathcal{R}_k$ from $\mathcal{R}_{k-1}$, all possible binary and continuous inputs must be considered. Algorithms 2 considers all possible continuous inputs. To consider the effect of all possible binary inputs, for every corresponding discrete mode, the reach set is computed via algorithm 2 and $\mathcal{R}_k$ is obtained by calculating the union of the results.

The proposed active fault diagnosis algorithm is used for sanity check of the valve V_1. A stuck ON fault is considered in V_1 and the algorithm is used to generate the shortest test signal sequence to diagnose this fault. Nine different scenarios as shown in Table. 3 are considered. In each scenario, a binary input is used or fixed during the diagnosis to 0 or 1.

Fig. 3 and 4 show the results for scenario 1 where both discrete inputs are used. In order to make the difference between $Y(\mathcal{R}_{N_k})$ and $Y(\mathcal{R}_{F_k})$ observable, the discriminating set in algorithm 1 is changed to $I = Y(\mathcal{R}_{N_k}) \backslash (Y(\mathcal{R}_{F_k}) \oplus \mathcal{B}(0,d))$, where $\mathcal{B}(0,d)$ is a box defined as $\mathcal{B}(0,d) = \{x \in \mathbb{R}^2 | 0 \leq x_i \leq d\}$. The algorithm terminates after $k = 5$ steps. $Y(\mathcal{R}_{N_5})$ and $Y(\mathcal{R}_{F_5}) \oplus \mathcal{B}(0, 0.01)$ are shown in Fig. 3. The set I consists of two polytopes shown in Fig. 4. One of these polytopes(the grey one here) is considered as the target set and then the input to reach the target set is computed and applied to the system. The resulting output and the expected output of the system are depicted in Fig. 4. As it can be seen the result of the diagnosis algorithm is that the system is faulty.

As it is shown in Table 3, scenarios 4, 5, 6 are not applicable. Because V_1 is fixed as 1 and therefore the model of the normal system becomes exactly the same as the model of the faulty system.Moreover, it shows that using a valve as a free input variable causes more computational complexity than fixing it. The reason is that the main source for the computational complexity of the algorithm is nonconvexity of the reach set which is casued by either crossing a gaurd (h_v here) or a switching input. It should be noted that however using both valves is the most computationally demanding scenario but for

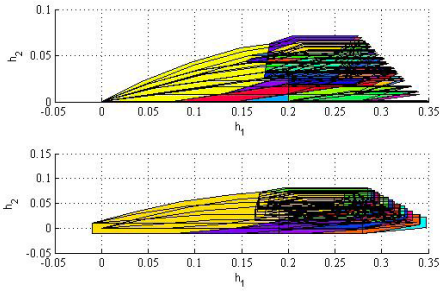

Fig. 3. Top:Reach set of the normal system at $k = 5$:($\mathcal{R}_{N_5}$), Bottom:Reach set of the faulty system at $k = 5$ added by $\mathcal{B}(0, 0.01)$:($\mathcal{R}_{F_5} \oplus \mathcal{B}(0, 0.01)$).

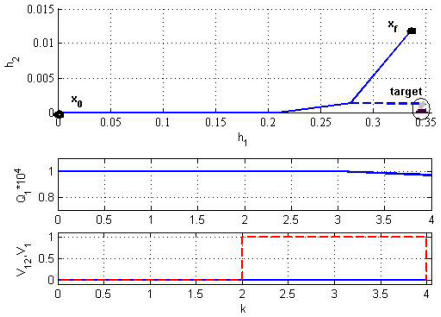

Fig. 4. Top:Output of the system(solid), expected output of the system (dashed) and discriminating set (target), Middle:continuous input Q_1, Bottom:discrete inputs:V_1(dashed), V_{12}(solid)

Table 3.

Scenario	V_1	V_{12}	K_{max}	CPU time (sec)
1	x	x	5	8.9
2	x	1	6	0.87
3	x	0	5	0.71
4	1	x	NA	-
5	1	1	NA	-
6	1	0	NA	-
7	0	x	5	4.5
8	0	0	5	0.37
9	0	1	6	0.54

this scenario the algorithm will find the shortest input sequence for diagnosis while by fixing valves it may not find the shortest sequence e.g. as it is the case in scenarios 2, 9.

Fig. 5 demonstrates the case where the faulty and the normal system exhibit same dynamic behaviour. In this example a model predictive controller is designed for the two tank system. Fig. 5 shows the simulation of the closed loop system. As one can see the control variable V_1 is manipulated such that the output of the system in the normal condition and in the faulty one is exactly the same. In this situation if a stuck ON fault happens, no passive diagnoser would be able to diagnose it, while the active diagnoser proposed here is capable of detecting this fault. Our active diagnoser was started at $t = 200$ sec. and the result is shown in Fig. 6.

5. CONCLUSION AND FUTURE WORKS

This paper presented an approach for active fault diagnosis of hybrid systems based on reach sets computation of both the normal and the faulty modes. The proposed method can be used for sanity check of the system at the commissioning phase and also periodically during normal operation for faster detection of faults or detection of faults when it is impossible to detect them by a passive diagnoser.

During the diagnosis it is assumed that the system is in the normal mode of the operation. To ensure that if the system is faulty, it will remain in the tolerable performance region, the optimization problem is solved subject to constraints describing polytope of the tolerable area. It might happen that the optimization become infeasible by these constraints. This issue is subject to future investigations.

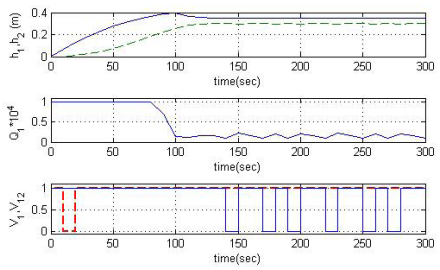

Fig. 5. Top:output of the closed loop system for both faulty and normal system:h_1(solid) h_2(dashed), Middle:continuous input Q_1, Bottom:discrete inputs: V_1(dashed line), V_{12}(solid line)

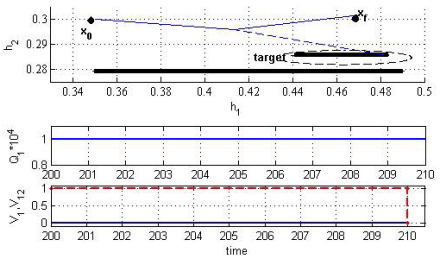

Fig. 6. Top:Output of the system(solid), expected output of the system and target set, Middle:continuous input Q_1, Bottom:discrete inputs:V_1(dashed line), V_{12}(solid line)

REFERENCES

I. Andjelkovic, K. Sweetingham, and S.L. Campbell. Active fault detection in nonlinear systems using auxiliary signals. In *American Control Conference*, pages 2142–2147, Seattle, WA, 2008.

P. Antsaklis and X. Koutsoukos. *Software-Enabled Control*, chapter Hybrid systems: Review and recent progress, pages 271–298. IEEE Press, 2003.

A. Balluchi, L. Benvenuti, M. Domenica Di Benedetto, and A. L. Sangiovanni-Vincentelli. Design of observers for hybrid systems. In *HSCC '02: Proceedings of the 5th International Workshop on Hybrid Systems: Computation and Control*, pages 76–89, London, UK, 2002. Springer-Verlag. ISBN 3-540-43321-X.

M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki. *Diagnosis and Fault-Tolerant Control*. Springer-Verlag, 2006.

S.L. Campbell and R. Nikoukhah. *Auxiliary Signal Design for Failure Detection*. Princeton University Press, 2004.

A. Girard, C. L. Guernic, and O. Maler. *Hybrid systems: Computation and Control*, volume 3927 of *Lecture Notes in Computer Science*, chapter Efficient computation of reachable sets of linear time-invariant systems with inputs, pages 257–271. Springer, 2006.

M. Hofbaur and B. Williams. *Hybrid systems: Computation and Control*, volume 3927 of *Lecture Notes in Computer Science*, chapter Mode estimation of probabilistic hybrid systems, pages 253–266. Springer, 2002.

X. Koutsoukos, J. Kurien, and F. Zhao. *Hybrid systems: Computation and Control*, volume 2623 of *Lecture Notes in Computer Science*, chapter Estimation of distributed hybrid systems using particle filtering methods, pages 298–313. Springer, 2003.

F. Lin. Diagnosability of discrete events systems and its application. *Discrete event systems*, 4:197–212, 1994.

J. Lunze. *Hybrid systems: Computation and Control*, volume 1790 of *Lecture Notes in Computer Science*, chapter Diagnosis of quantized systems by means of timed discrete-event representations, pages 258–271. Springer, New York, 2000.

R. Mohammadi, S. Hashtrudi-Zad, and K. Khorasani. A hybrid architecture for diagnosis in hybrid systems with applications to spacecraft propulsion system. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 3184–3190, Montreal, Canada, 2007.

S. Narasimhan and G. Biswas. Model-based diagnosis of hybrid systems. *IEEE transactions on man and cybernetics*, 37(3):347–361, 2007.

H.H. Niemann. A setup for active fault diagnosis. *IEEE Transactions on Automatic Control*, 51(9):1572–1578, 2006.

M. Sampath, S. Lafortune, and D. Teneketzis. Active diagnosis of discrete-event systems. *IEEE Transactions on Automatic Control*, 48(7):908–929, 1998.

F. Zhao, X. Koutsoukos, H. Haussecker, J. Reich, and P. Cheung. Monitoring and fault diagnosis of hybrid systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 6:1225–1240, 2005.