Automatic sensor assignment of a supermarket refrigeration system

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

Published in:
I E E E Conference on Control Applications. Proceedings

DOI (link to publication from Publisher):
10.1109/CCA.2009.5281004

Publication date:
2009

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

18th IEEE International Conference on Control Applications
Part of 2009 IEEE Multi-conference on Systems and Control
Saint Petersburg, Russia, July 8-10, 2009

Automatic Sensor Assignment of a Supermarket Refrigeration System

Seyedmojtaba Tabatabaeipour, Roozbeh Izadi-zamabadi, Thomas Bak, Anders P. Ravn

Abstract— **Wrong sensor assignment is a major source of faults in industrial systems during the commissioning phase. In this paper a method for automatic sensor assignment based on active diagnosis is proposed. The active diagnosis method is developed for diagnosis of linear hybrid systems. It generates the appropriate test signal which can be used for sanity check at the commissioning phase. It could also be used for faster detection of faults during the normal phase of operation or for detection of faults which are impossible to detect by passive methods because of regulatory actions of the controller. The method is tested on a supermarket refrigeration system.**

I. INTRODUCTION

In a large system there are many sensors, actuators and other components. Every measurement from a sensor or output to an actuator should be assigned correctly to its corresponding variable in the control algorithm. Yest, it happens that a technician connects components of a system wrongly. Wrong sensor or actuator assignment potentially results in malfunction of the overall system. Therefore, it is desirable to design a controller which provides a sanity check in the commissioning phase for verifying sensor and actuator assignment by generating an appropriate test signal.

A way to tackle this problem is to consider wrong connections as faults and use fault diagnosis methods. Diagnosis methods can be divided into two main categories: active and passive. In passive diagnosis the diagnoser observes the system and based on the observation decides about the occurance of faults. In active fault diagnosis the diagnoser generates a test signal which excites the system to decide whether the observed system dynamics exhibits the normal behaviour or the faulty behaviour and if feasible decide which faulty behaviour occurs.

Industrial systems typically include both discrete and continuous components and a hybrid system formulation is therefore natural to adopt. 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 see [7], [8], [12]. Most of the available methods are in the area of passive diagnosis. [3] propose a method for active diagnosis of linear systems using an auxiliary signal for fault detection. The results of [3] are extended to nonlinear systems in [1] using linearization and also a direct optimization approach. A setup for active diagnosis of linear system for parametric faults is proposed by [9]. In [6] and [10] the problem for discrete event systems is investigated. In our previous work [11], we proposed an active fault diagnosis method for diagnosis of linear hybrid systems in discrete time. The method is based on prediction of the behaviour of the system in the future by means of reach set computations based on a faulty and a normal model of the system. If we apply the method directly to the sensor assignment problem, in other words, if we consider all possible assignments as a fault, we need a model for each possible assignments which yields a high computational effort. In this paper we extend the active diagnosis algorithm to the sensor assignment problem such that only one model of the system is necessary. To illustrate the method a supermarket refrigeration system is considered.

S. Tabatabaeipour and T. Bak are with Department of Electronic Systems, Aalborg University, DK-9220, Denmark. {smt,tba}@es.aau.dk
R. Izadi-Zamanabadi is with Central R&D-RA-DP, Danfoss A/S, DK-6470, Denmark. roozbeh@danfoss.com
A. P. Ravn is with Department of Computer Science, Aalborg University, DK-9220, Denmark. apr@cs.aau.dk

II. PRELIMINARIES AND PROBLEM FORMULATION

To make our ideas precise we first define the problem and give some preliminary definitions.

A. Problem Formulation

Consider a system with n sensors *i.e.* $[S_1, \cdots, S_n]$. The sensor assignment problem is to find among all permutations the one that conforms to the dynamic behaviour of the undelying system. The problem is defined as follows.

Problem *1 (Sensor assignment problem):* Given a set of measurements $\mathbf{y} = [y_1, \cdots, y_n]$ representing measurements from $[S_1, \cdots, S_n]$ and a model of the system as $\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)),\ [\hat{y}_1(k), \cdots, \hat{y}_n(k)]' = h(\mathbf{x}(k), \mathbf{u}(k))$. Find a permutation of $\mathbf{y}$ namely $\mathbf{V}$ such that for a large N, for all i

$$\Sigma_{k=1}^{N} |V_i(k) - \hat{y}_i(k)| < \Sigma_{k=1}^{N} |V_j(k) - \hat{y}_i(k)| \qquad (1)$$

for all $j \in 1, \cdots, n, j \neq i.\square$

B. Preliminaries

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,

- $G : E \rightarrow 2^{X \times U}$ assigns to each $e = (q, q') \in E$ a guard $g(e) \subset X \times U$,
- $J : E \times X \times U \rightarrow 2^X$ 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 discrete time 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 the output map is of the form $y_{i+1} = C_{q_i} x_i + D_{q_i} u_i$. We refer to $(u, y) \in U \times Y$ as an observation of $\mathcal{H}$.

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 $\sigma_{i+1} = (q_{i+1}, x_{i+1}, u_{i+1}, y_{i+1})$ which satisfies the discrete and continuous evolution constraints imposed by hybrid automata and σ_0 satisfies the initial condition [11].

Both discrete faults and continuous faults are modeled as a mode in hybrid automata as in [7]. It is supposed that events that describe transitions from a normal mode to a faulty mode are unobservable. 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 $K = \{N, F_1, \ldots, F_p\}, p \geq 1$ is a set of conditions that constitutes 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 $Init_\kappa \subset \kappa \times X$.

III. THE PROPOSED ALGORITHM

In this section active fault diagnosis is described firstly and then it is explained that how the sensor assignment problem can be solved by extending the proposed algorithm.

A. Active Diagnosis

We are going to solve Problem 1 by means of an active diagnoser which generates a test signal in the commissioning phase for finding the true sensor assignment. A 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 as input and generates an estimate of the current condition $\hat{\kappa}(k)$ as output. In active diagnosis an input sequence $\langle u(k+1), \ldots, u(k+m) \rangle$ is generated by the diagnoser and applied to the system. The resulting output sequence $\langle y(k+1), \ldots, y(k+m) \rangle$ is observed by the diagnoser to determine the system condition. The active diagnosis problem is defined as follows:

Problem *2 (Active diagnosis problem):* Given a hybrid automaton $\mathcal{H}$, Find a sequence of inputs $\langle u(0), \ldots, u(m) \rangle$ such that the condition $\kappa(0)$ is determined by observing the sequence $\langle y(0), \ldots, y(m) \rangle$.

If the input sequence exists, *i.e.* if the system is diagnosable, we can look for the optimal solution, where optimality can be interpreted in different senses. The proposed algorithm looks for the shortest sequence of inputs that can diagnose the system.

A model-based passive diagnoser usually checks the consistency of the I/O pair with the expected behaviour of system based on a given model. If the consistency is verified, the system is in the normal mode otherwise it is faulty. Now consider 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 the observed I/O pair is uniquely in the set $\mathbf{B}_0$ or $\mathbf{B}_1$, such as point A or B, the diagnoser can detect whether the system is faulty or not. But for a point such as C which belongs to the intersection of $\mathbf{B}_0$ and $\mathbf{B}_1$ it is impossible to detect the mode of the system. The idea here 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 $\mathbf{B}_1$.

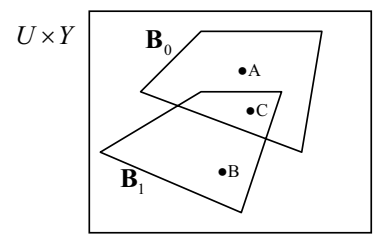

Fig. 1. System behaviour

Given a model of the normal and the faulty system, from the current state we predict all possible behaviour that each model of the system can present in the next step considering all possible inputs. This task is repeated as long as the predicted behaviour of the fautly and the normal model are the same. As soon as they become different, we find the set holding these different behaviours. We choose one of them, e.g. belonging to the 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.

It is supposed that the initial state of the system is given by an observer-based passive diagnoser as proposed by [2]. The diagnoser consists of two parts: mode observer and countinuous observer. 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 with the I/O sequence. A mode is consistent with the I/O sequence when the corrresponding element in the residual vector $\rho = \{r_1, \ldots r_m\}$ generated by the mode observer is zero. Consistency of two modes with the I/O sequence means that they have indistinguishable executions. Two executions are called indistinguishable in a time interval if their corresponding continuous output in that time interval are identical.

B. The proposed algorithm

In this subsection the proposed algorithm for one faulty mode is described. In [11] it is explained how to expand the algorithm to more than one faulty mode.

TABLE I

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 algorithm looks for two distinguishable executions χ_1 and χ_2 respectively from the system in normal condition, Σ_N, and the faulty system, Σ_F. In order to accomplish this task, all possible outputs that both systems could reach in the future time steps considering all admissible inputs and starting from the given initial state is computed which is equal to reach set computation.

Definition *2 (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 soon as the corresponding outputs of the reach sets of the system based on the normal and the faulty model of the system becomes different the algorithm terminates.

In Algorithm 1, reach sets of the normal system and the faulty system at time k are respectively denoted by $\mathcal{R}_{N_k}$ and $\mathcal{R}_{F_k}$ and the area of tolerable perfomance is denoted by the set $\mathcal{T}$. The area of tolerable performance is defined by the minimum level of control objectives and system constraints, which are required to maintain safe operation. At each time step $\mathcal{R}_{N_k}$ and $\mathcal{R}_{F_k}$ are computed. To ensure that the solution found by the algorithm does not include any intolerable performance, the area of intolerable 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 these two sets are not exactly the same or in other words 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*. As soon as the discriminating set becomes nonempty the algorithm proceeds to the next step which is determining the system condition.

To determine the system condition we need to make a hypothesis about it at time 0. If we assume that the system at time 0 is in the Normal condition, as it is assumed in algorithm 1, to test this hypothesis, the algorithm chooses a point which uniquely belongs to the future behaviour of the normal system i.e $\tilde{y}(K_{max}) \in Y(\mathcal{R}_{N_{K_{max}}}) \backslash Y(\mathcal{R}_{F_{K_{max}}})$ where $k = K_{max}$ shows the first time that the discriminating set becomes nonempty. 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 the hypothesis is verified and the system is in the normal condition otherwise it is in the faulty condition. Fig. 2 illustrates the algorithm.

Since the termination of the algorithm is not guaranteed, 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.

The above results are valid only if the reach set at time k is computed from the initial state without any uncertainty. But suppose that the initial state is given in the set $\mathcal{X}(0)$, then two different cases should be considered. In the first case, if all the states in the obtained reach sets $\mathcal{R}_{N_{K_{max}}}$ and $\mathcal{R}_{F_{K_{max}}}$ are reachable from the initial set $\mathcal{X}(0)$ within K_{max} sampling time then the previous result is hold, in other words, $\Delta_k = (Y(\mathcal{R}_{N_k}) \cup Y(\mathcal{R}_{F_k})) \backslash (Y(\mathcal{R}_{N_k}) \cap Y(\mathcal{R}_{F_k}))$. Checking the reachability condition for hybrid systems is not simple. In the second case, if the reachability condition does not hold then the conservative approach is to check when the two reach sets $Y(\mathcal{R}_{N_K})$ and $Y(\mathcal{R}_{F_K})$ are totally distinct from each other *i.e.* $Y(\mathcal{R}_{N_K}) \cap Y(\mathcal{R}_{F_K}) = \emptyset$. When this condition is satisfied the algorithm must terminate and $\Delta_k = Y(\mathcal{R}_{N_k}) \cup Y(\mathcal{R}_{F_k})$.

To find the optimal input, the following cost function is used:

$$J(\mathbf{x}_k, \mathbf{u}_k, \mathbf{y}_k) =$$

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

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

Two groups of constraints are applied in the optimization. The first one is that the state variables should evolve based on the dynamic of the system which is dependent on our hypothesis. The second group ensures that the system remains in the area of tolerable performance for the situation that our hypothesis was wrong and the system is actually faulty. Suppose that the area of tolerable performance is given by the

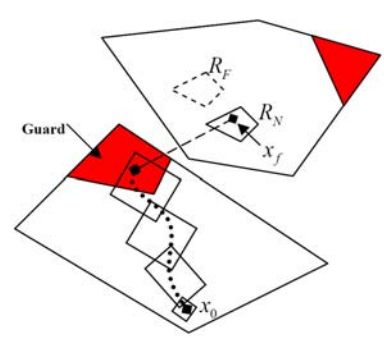

Fig. 2. Active diagnosis method

polytope $\mathcal{T} = \{x \in \mathbb{R}^n | \mathcal{P}x \leq \mathcal{M}\}$. To ensure that system states will remain in $\mathcal{T}$, the following constraints should be added to the optimization problem: $\{\mathcal{P}x(i) \leq \mathcal{M}\}_{i=k+1}^{k+K_{max}}$.

For a linear systems the reach set can be computed as:

$$Reach(\Sigma, \mathcal{X}(0), \mathcal{U}) = A\mathcal{X}(0) \oplus B\mathcal{U}, \tag{2}$$

where $\mathcal{X}(0), \mathcal{U}$ denote the convex polyhedra of the initial state and the input respictvely and $\oplus$ is the geometric or Minkowski sum. For computational effciency 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 [4]. In the case of linear hybrid systems enabled transitions and the corresponding jump functions should be considered. The reach set computation is descibed with more details in [11].

C. Sensor Assignment

Wrong assignment of sensors can be considered as a fault, and it can be modelled as a permutation of the output vector. The problem is to find among all permutations the one that is consistent with the dynamic behaviour of system. Consider a system with n sensors. There are $n!$ candidate assignments or in another words $n! - 1$ fault hypothesis. If we use algorithm 1 directly, it will be computationally very expensive. The method proposed here only needs one model of the system. It is supposed that the initial state of the system is given such that the outputs are indistinguishable, i.e. $y_i = y_j, i, j \in 1, \cdots, n, i \neq j$. In order to simplify the explanation, the idea is described for a system with two sensors. We assume that as long as $|y_1 - y_2| < \epsilon$ outputs can not be distinguished. If we excite the system such that as its result $y_1 > y_2 + \epsilon$ or $y_2 > y_1 + \epsilon$ then they can be distinguished. Therefore as before we compute future reach sets of the system. As soon as the corresponding output set goes outside the region $|\hat{y}_1 - \hat{y}_2| < \epsilon$ the algorithm terminates. A state correspondent to a point in $Y(\mathcal{R}_{K_{max}}) \cap (|\hat{y}_1 - \hat{y}_2| > \epsilon)$ is chosen. Any point in this set exibits an order between its elements *i.e.* $\hat{y}_1 > \hat{y}_2$ or $\hat{y}_2 > \hat{y}_1$. A point in this set is chosen. We find the input for leading the system to the chosen point and apply it to the system. By comparing the order in the elements of the output vectors and the predicted orders between elements of $[\hat{y}_1, \hat{y}_2]$ we can find the correct assignment. For example, if a point in $\hat{y}_2 > \hat{y}_1$ is chosen and the observed output presents the following order $y_1(K_{max}) > y_2(K_{max})$ then S_1 should be assigned to the variable $\hat{y}_2$ and S_2 to $\hat{y}_1$. If there are more than two sensors the strategy is the same. The algorithm looks for an area where the outputs present an order which is $Y(\mathcal{R}_{K_{max}}) \cap (|y_i - y_j| > \epsilon, 0 \leq i, j \leq n, i \neq j)$. The system is then driven to that area. By comparing the predicted order and the obseved order the assignment is accomplished.

IV. System Description

In a supermarket, for customer's convenience, goods are usually placed in an open display case in a refrigerator. Fig. 3 shows a supermarket refrigeration system with two display cases. The system consists of five main parts, namely liquid manifold, display cases, suction manifold, compressor and condenser. The refrigerant in the liquid manifold is in the liquid phase. It is led into the evaporators inside the display cases through inlet valves. The compressor keeps the evaporator temperature at a certain level by keeping the pressure in the suction manifold at a constant pressure. The refrigerant removes heat from goods while evaporating in the evaporators and transforming into low pressure gas. The low pressure refrigerant is compressed in the compressor rack. The refrigerator circuit is closed by feeding back the liquid refrigerant from the condenser to the liquid manifold.

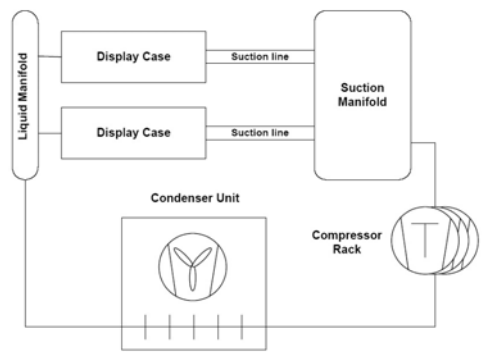

Fig. 3. A Simplified Supermarket Refrigeration System

Fig. 4 shows an schematic illustration of the measurements and control instrumentation in a typical display case used in a supermarket refrigeration system. An air flow is circulating through the evaporator. The refrigerant is led into the evaporator through an on/off inlet valve and evaporates while absorbing the heat from the surrounding. The circulating air flow creates a cold air curtain at the front of the display case. Since the air curtain is colder than the goods and the surroundings, it absorbs the heat from the goods $(Q_{goods-air})$ and the surroundings $(Q_{airload})$. The absorbed heat is transferred through the evaporator wall to the evaporator $(Q_{air-wall})$.

V. The Hybrid Model of the System

The hybrid model we use is based on the model proposed in [5].

A. Evaporator

The dynamic of the evaporator is obtained by writing energy balance equations:

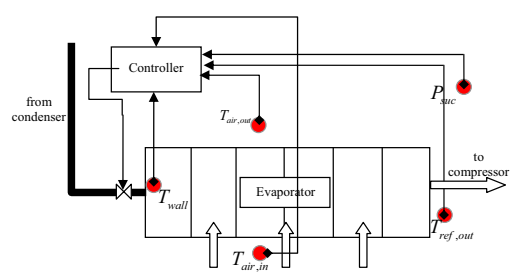

Fig. 4. An evaporator and its instrumentation

$$\frac{dT_{air,in}}{dt} = \frac{\dot{Q}_{goods-air} + \dot{Q}_{airload} - \dot{Q}_{air-wall}}{M_{air}Cp_{air}} \quad (3)$$

$$\frac{dT_{wall}}{dt} = \frac{\dot{Q}_{air-wall} - \dot{Q}_e}{M_{wall}Cp_{wall}} \quad (4)$$

$$\frac{dT_{goods}}{dt} = \frac{-\dot{Q}_{goods-air}}{M_{goods}Cp_{goods}} \quad (5)$$

Moreover,

$$\dot{Q}_{air-wall} = UA_{air-wall}(T_{air} - T_{wall}) \quad (6)$$

$$\dot{Q}_e = UA_{wall-ref}(M_{ref})(T_{wall} - T_e) \quad (7)$$

$$\dot{Q}_{goods-air} = UA_{goods-air}(T_{goods} - T_{air}) \quad (8)$$

$$UA_{wall-ref} = UA_{wall-ref,max}\frac{M_{ref}}{M_{ref,max}} \quad (9)$$

$$T_{air,in} - T_{air,out} = \frac{\dot{Q}_{air-wall}}{\dot{m}_{air}Cp_{air}}, \quad (10)$$

where M denotes the mass, Cp the heat capacity and UA the overall heat transfer coefficient with the subcript denoting the media between which the heat is transferred. T_e shows the evaporation temperature which is a refrigerant dependant function of the evaporation pressure P_e. Here it is assumed that there is no pressure drop in the suction line and therefore the suction pressure P_{suc} is equal to the evaporation pressure.

It is assumed that the evaporator will be filled or emptied abruptly as the valve is opened or closed respectively. Consequently, the value of the mass of refrigerant ,M_{ref}, switches between 0 and $M_{ref,max}$.

B. The Suction Manifold

The dynamic of the suction pressure is described by

$$\frac{dP_{suc}}{dt} = \frac{\dot{m}_{in-suc} + \dot{m}_{ref-const} - \dot{V}_{comp}.\rho_{suc}}{V_{suc}\frac{d\rho_{suc}}{dP_{suc}}} \quad (11)$$

where V_{suc} is the volume of the suction manifold, $\dot{V}_{comp}$ is the volume flow from the suction manifold to the compressor and $\dot{m}_{in-suc}$ is the total mass flow from the evaporator to the suction manifold which is given by

$$\dot{m}_{in-suc} = \Sigma_{i=1}^{n}\frac{\dot{Q}_e, i}{\Delta h_{lg}}, \quad (12)$$

where n is the number of the display cases. $\dot{m}_{ref-const}$ is a constant disturbance representing mass flow from other unmodelled refrigerator entities. ρ_{suc} represents the density of the vapor in the suction manifold which is a nonlinear refrigerant-dependent function of P_{suc}.

C. The Compressor

A number of compressors working in parallel that can be switched on or off separately constitute the entire compressor capacity. The entire volume flow out of the suction manifold is described by $\dot{V}_{comp} = \Sigma_{i=1}^{q}\dot{V}_{comp,i}$, where $\dot{V}_{comp,i}$ is the volume flow created by one compressor which is given by

$$\dot{V}_{comp,i} = \frac{comp_i.\eta_{vol}.V_{sl}}{100} \qquad i = 1, \cdots, q, \quad (13)$$

where $comp_i$ denotes the capacity of the i'th compressor, q is the number of compressors, η_{vol} is the constant volumetric efficiency and V_{sl} is the total displacement volume.

D. The Overall Model

Putting together the above subsystems we get the overall dynamical model of the system. Each display case has three states, namely $T_{air,in}, T_{goods}, T_{wall}$ and the suction manifold has one state which is P_{suc}. Measured variables are $T_{air,in}, T_{wall}, T_{air,out}, T_e$. Inputs of the system are the evaporator inlet valves and comppressors valves. These valves are considered as on/off valves and therefore the overall model of the systems represents a hybrid dynamic.

In order to apply our method to this system we need a linear hybrid dynamical model of it. Therefore nonlinearities such as the dependency of T_e and ρ_{suc} on P_{suc} in equations 7, 11 are substituted by linear approximations of them.

VI. Simulation Results

The sensor assignment algorithm is tested on the refrigeration system for assignment of the wall and input air temperature sensors to $[T_{air,in}, T_{wall}]$. Because we are considering the commissioning phase there is no goods inside the display case and therefore $\dot{Q}_{goods-air} = 0$ and T_{goods} is not a state. It is assumed that the initial states are in the polytope $\{14 \leq T_{air,in} \leq 16, 14 \leq T_{wall} \leq 16, 1 \leq P_{suc} \leq 3\}$. Also $\dot{Q}_{airload}$ is considered as a disturbance and is assumed to vary between 1500 and 4500. We have used $T_s = 2$ as sampling time for discretization and $\epsilon = 1$.

To consider the effect of all possible binary inputs, for every corresponding discrete mode the reach set is computed via algorithm 1 and $\mathcal{R}_k$ is obtained by calculating the union of the results. Because of uncertainties due to the initial states given as a polytope and $\dot{Q}_{airload}$ considered as disturbance, as explained in section III, we should either check the reachability condition or consider the conservative solution. Here the conservative solution is considered. Consider the reach set at time k. Because of the switching effect of the binary inputs it is a union of p polytopes $\mathcal{R}_k = \cup_{i=1}^{p}\mathcal{P}_i$. We can not say that every state in $\mathcal{R}_k$ is reachable but we know that a state in $\mathcal{P}_i$ is reachable by choosing the corresponding sequence of binary inputs. Therefore, for termination of the algorithm it is enough to check whether there exist a $\mathcal{P}_i$ in $\mathcal{R}_k$ such that its intersection with $|y_1 - y_2| < \epsilon$ is empty. It happens at $k = 4$ and the reach set is depicted in Fig. 5.

Fig. 6 shows the initial states, the target polytope and the observed and predicted output. Comparing the expected order and the observed order the assignment is $(y_2, T_{air,in}), (y_1, T_{wall})$. The obtained input sequence for both valves is $[1, 1, 1, 1]$ which means that both valves should be opened, that is we should cool down the system as soon as possible. It is shown in [11] that when there is both continuous and discrete inputs, the main complexity of the algorithm is due to discrete inputs which cause switching and therefore nonconvexity in the reach set. If the computational complexity is too high, it is possible to fix some of discrete inputs and diagnose the system at the cost of losing the

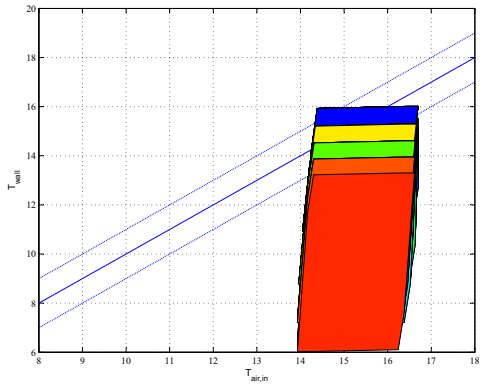

Fig. 5. Reach set of the system at $k = 4$, $T_{wall} = T_{air,in} + 1$, $T_{wall} = T_{air,in} - 1$ and $T_{wall} = T_{air,in}$

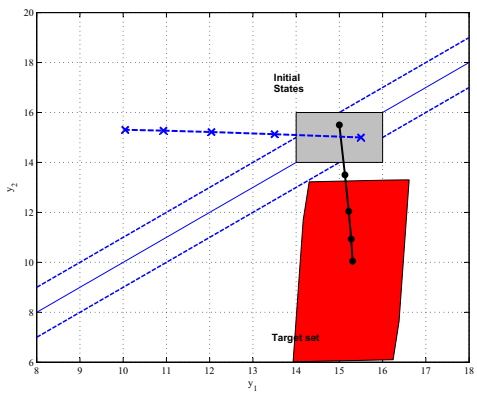

Fig. 6. Initial states, the target polytope, the observed output (dashed) and the predicted output (solid).

optimal input. However, sensor assignment computations can be done offline.

A frequent fault in the refrigeration system is that the value of the pressure sensor is fixed at its value when the fault happens. Fig.7 shows a simulation of the refrigeration system controlled by a hysteresis controller for $T_{air,in}, P_{suc}$ where the upper and the lower values for $T_{air,in}$ are $0, 4$ and those of P_{suc} are $1, 1.5$. If this fault happens, for example at $t = 300$, no passive diagnosis method will be able to detect it until $t = 1162$. This is because the normal system and the faulty system in this period exhibit the same behaviour. Using the active diagnosis method helps us to diagnose the fault faster. We have applied the algorithm and the input sequence is to open V_{evap} for 3 sampling times. The reason for this can be easily seen if one looks at the behaviour of the system at $t = 1162$ when the controller opens V_{evap} and as its result P_{suc} increases.

VII. Conclusion

In this paper an approach to the problem of sensor assignment based on active fault diagnosis is proposed and tested on a supermarket refrigeration system. The active diagnosis approach could also be used for sanity check at

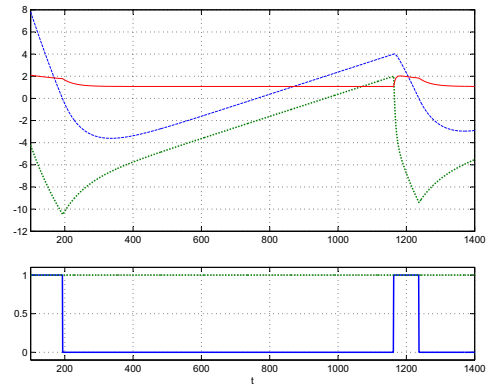

Fig. 7. Top: $T_{air,in}$(dashed), T_{wall}(dotted), P_{suc}(solid), Bottom: V_{comp}(dotted)V_{evap}(solid).

the commissioning phase or for faster detection of faults during the normal operation of the system.We extended the previous result on active diagnois for sensor assignment such that we do not need reach set computation for every possible assignment, but reach set computation is itself a computationally burdensome task. An algorithm that does not need reach set computation would be desirable. In our future work it will be investigated using a reformulation as an optimization problem.

References

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

[2] A. Balluchi, L. Benvenuti, M. D. D. 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.

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

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

[5] L. Larsen, R. Izadi-Zamanabadi, R. Wisniewski, and C. Sonntag. Supermarket refrigeration systems-a benchmark for the optimal control of hybrid systems. Technical report, Technical report for the HYCON NoE, 2007. http://www. bci. tudortmund. de/ast/hycon4b/index. php.

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

[7] 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, 2007.

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

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

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

[11] S. Tabatabaeipour, A. P. Ravn, R. Izadi-zamanabadi, and T. Bak. Active fault diagnosis of linear hybrid systems. Accepted in SAFEPROCESS, 2009.

[12] 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.