Hybrid Causal Model Based Diagnosis. Application to Automotive Embedded Functions.

R. Pons A. Subias L. Travé-Massuyès

CNRS; LAAS; 7, avenue du Colonel Roche, F-31077 Toulouse, France
Université de Toulouse; UPS, INSA, INP, ISAE; LAAS; F-31077 Toulouse, France (e-mail: rpons,subias,louise@laas.fr)

Abstract: The behavior of embedded systems is commonly characterized by hybrid phenomena in which each operational mode is triggered by commands sent by *electronic control units* (ECU), involving hardware and software components. While hardware components are inherently continuous, the ECUs introduce discrete switching between the behavioral modes of the apparatus. For continuous systems, the theory of logical diagnosis casts the diagnosis problem within a consistency-based reasoning scheme that requires normal behavior models only. On the other hand, hybrid model based diagnosis methods rely on the availability of fault models and implement abductive reasoning similarly to what is done in discrete event model based diagnosis approaches.
In this paper we propose a hybrid model consistency based method in which ideas are borrowed from discrete event system diagnosis to build a partial diagnoser, and from continuous systems to check consistency and track causal dependencies underlying discrepancies between expected and observed behaviours.

Keywords: hybrid systems, diagnosis, consistency-based reasoning, causal graphs, automotive.

1. INTRODUCTION

Today embedded systems are found everywhere and form an inward part in the design of contemporary artefacts, in interaction with hardware components spanning multi-domain technologies. The intimate coupling of software and hardware capacities exhibits complex patterns of behavior and numerous nominal modes of operations. In the automative field this hardware-software coupling is found in the different systems used to implement electronic functions such as fuel injection, ABS etc. These electronic systems are composed of voltage supplies, sensors and actuators linked to Electronic Control Units (ECU) by a wire harness. Diagnosing such systems must not only account for the structural interconnection of components but also for the different configurations underlying behavioral modes.

ECUs are equipped with an auto-diagnosis function that reliably detects the failing electronic circuit which is connected to the ECU. However, they are unable to localize precisely the faulty components. In practice, such electronic circuits are diagnosed from diagnosis trees built beforehand, often manually. These trees allow the car mechanic to find the faulty component(s) by performing a guided sequence of measurements. The main problem is the determination of a proper sequence of tests and measures at available control points, which would lead to greedily localize the fault quickly and at the lowest cost. This problem is known as the *Test Sequencing Problem*. In the target automotive domain, troubleshooting starts with a set of preliminary symptoms gathered by the car mechanic: fault codes from ECUs, client symptoms and other preliminary car mechanic observations. Then, the fault isolation problem is defined as the determination of the required additional information (obtained by tests) which allows the best discrimination among the diagnostic hypotheses generated with the preliminary symptoms.

When facing the factors of complexity related to embedded systems, one must admit that the traditional methods requiring to anticipate the set of faults that may occur show limitations. The challenge is then to develop diagnosis approaches applicable to the general case in which the available models (normal or faulty) cannot be considered as exhaustive. These diagnosis approaches must fully account for the hybrid nature of the embedded systems. Previous works have proposed solutions to diagnose electric circuits Esser and Struss (2007); Faure (2001); Olive (2003); Price et al. (1995); Sachenbacher and Struss (2001), among which only few of them tackle the hybrid aspect Ressencourt (2008).

Like for discrete event model based diagnosis approaches, all the hybrid model based diagnosis methods are based on the assumption that fault models are available and work by implementing abductive reasoning. Hybrid models, like hybrid automata, are used to represent interlinked continuous and discrete dynamics. An behavioral mode corresponds to a discrete state of the hybrid automaton and mode changes are modeled by discrete transitions labelled by appropriate discrete events that may or may not be observable. The model accounts for a set of anticipated faulty situations, represented by as many behavioral

modes. The on-line diagnosis problem is then formulated as a state estimation problem and the troubleshooting problem is equivalent to a test problem.

In this paper, we are breaking with the assumption that fault models are available and exhaustive. We propose to cast the hybrid diagnosis problem within the framework used for continuous systems and known as consistency-based reasoning. The consistency-based reasoning scheme is commonly used by both the FDI community Gertler (1998) and the DX community through the logical theory of diagnosis Weld and De Kleer (1989). This approach allows one to achieve diagnosis when only normal behavior models are known. The work proposed in this paper can can be used to complement the fault dictionary based former methods with a consistency based method designed for hybrid systems.

The method borrows ideas from discrete event systems (DES) diagnosis to build a *partial diagnoser* and from continuous systems consistency based diagnosis through *causal graphs* to track the causal dependencies underlying discrepancies and ultimately isolate the diagnosis component candidates. Our hybrid diagnosis reasoning balances abductive and consistency based reasoning.

The paper is organized as follows. Section 2 gives an overview of the proposed diagnosis method. Section 3 summarizes the different modeling steps to be performed off-line to support the diagnosis approach. It includes the presentation of the hybrid model and of the qualitative mode signatures that capture the expected values of the continuous variables in a given behavioral mode. This section introduces also the generation of the partial diagnoser. Section 4 then presents the main results of causal model based diagnosis and the extensions we propose for hybrid systems. Section 5 presents the hybrid causal model based diagnosis approach that we propose. Finally, the last section describes the application of the method to the troubleshooting of the rear windscreen wiper embedded function.

2. GLOBAL VIEW OF THE DIAGNOSIS METHOD

Figure 1 gives an overview of the proposed hybrid causal diagnosis method. The method includes the following two main stages described below.

Off-line stage

This stage concerns the modeling step and the generation of a Partial Diagnoser. The modeling step starts with the hybrid model of the system formalized in the hybrid automaton formalism (1) and aims at abstracting the hybrid automaton into a pure discrete event model. From the hybrid model, we derive three types of mathematical objects that represent three different aspects of the system. The first one is the underlying Discrete Event System (2). The second one is the mode signatures (3) that capture the expected values of the observable continuous variables within each behavioral mode ; they are generated using a dedicated simulation tool for hybrid systems (Modelica). The third one is a set of causal graphs that describe the causal relationships among the variables of the system (4). Then abstraction functions are used to abstract the

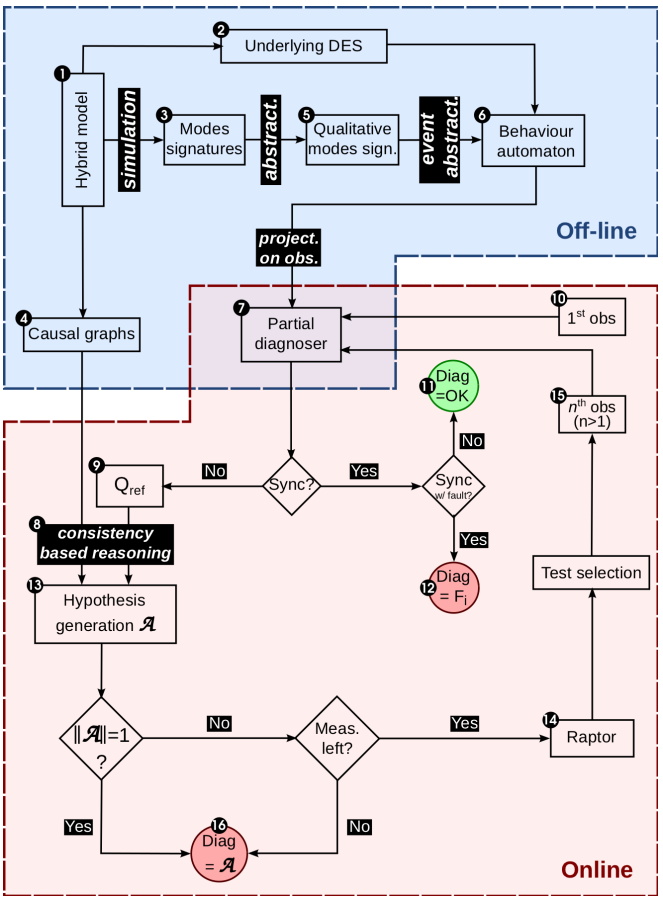

Fig. 1. Modeling, detection and diagnosis algorithm.

domain value of continuous variables in terms of a finite set of qualitative values. A qualitative mode signature (5) can then be associated to each mode. The next step abstracts the continuous dynamics captured by qualitative mode signature changes in terms of events. From the event based abstraction of the continuous dynamics and from the underlying DES model the behavior automaton (6) is genarated. This automaton provides a pure discrete event view of the hybrid system. The modeling step is completed by the generation of the causal graphs assigned to each mode of the system. The construction of a Diagnoser (7) including normal trajectories and some fault trajectories (so named Partial Diagnoser) is then performed like in Sampath et al. (1995) by projecting the behaviour automaton onto the observable space.

Online stage

The online stage [1] includes a diagnosis hypothesis generation step followed by a test selection step. Hypothesis generation is performed by applying consistency based reasoning (8) based on the causal graphs and on the knowledge of a set of possible reference behavioral modes Q_{ref} for checking the consistency. The set Q_{ref} (9) is obtained when the sequence of observations issued by the car mechanic (10) cannot be synchronized with any trajectory of the Partial Diagnoser. In other cases, if the synchronized trajectory is faultless no fault is detected (11) else the

[1] The term *online* means during the troubleshooting stage performed in the garage.

fault associated to the trajectory is recognized (12). At the end of the hypothesis generation step an ambiguity set $\mathcal{A}$ is generated (13). If $\mathcal{A}$ is not a singleton and if there are some measurements left, the test selection step takes place. The method chosen for test selection is RAPTOR (14), proposed by Gonzalez-Sanchez et al. (2011). With the new selected test (15), i.e. variable to be measured, the Partial Diagnoser is updated and the process is reiterated. If $\mathcal{A}$ is a singleton or if there are no measurement left the diagnosis result (16) is given by $\mathcal{A}$.

The next sections provide a detailed presentation of the different steps mentioned above.

3. HYBRID MODELING FOR DIAGNOSIS PURPOSES

3.1 The hybrid model

The hybrid systems at hand are modeled by a hybrid automaton whose discrete states represent the modes of operation for which the continuous dynamics are specified. The discrete event part (DES) constraints the possible transitions among modes.

Formally, a hybrid system is defined as a tuple $\Gamma = (\mathcal{X}, \mathcal{D}, Inv, T, \Sigma, CSD, Init)$ Henzinger (1996); Lehmann and Lunze (2009), where:

- $\mathcal{X}$ is a set of continuous variables denoted by the symbol x corresponding to state and input/output variables, which are functions of time t. The set of directly observable, i.e. measured, continuous variables is denoted by $\mathcal{X}_{OBS}$ [2]
- $\mathcal{D}$ is a set of discrete variables. $\mathcal{D} = Q \cup \mathcal{K} \cup \overline{\mathcal{X}}$, where Q is the set of locations, i.e. states of the DES, $\mathcal{K}$ is a set of auxiliary discrete variables used to represent the system configuration, and $\overline{\mathcal{X}}$ is a set of discrete variables denoted by the symbol $\overline{x}$, obtained from abstracting continuous variables as explained in 3.2.1. Each location $q_i \in Q$ represents a behavioral mode of the system, including nominal and anticipated fault modes. Each $K_j^i \in \mathcal{K}$ represents a behavioral mode of the component j composing the system. Discrete variables are not observable [3]
- Inv is an invariant which defines a domain for each location. There are two types of invariants:
  - $Inv_{\mathcal{K}}(q_i) : Q \rightarrow \otimes_i D(K_i)$, where $D(K_i)$ is the domain of K_i, is a *configuration invariant* [4];
  - $Inv_{\mathcal{X}}(q_i) : Q \rightarrow \otimes_i D(\overline{x}_i)$, where $D(\overline{x}_i)$ is the domain of $\overline{x}_i$, is a *signature invariant*
- $T : Q \times \Sigma \rightarrow Q$ is the transition function. The transition from mode q_i to mode q_j with associated event σ is noted (q_i, σ, q_j) or $q_i \xrightarrow{\sigma} q_j$
- Σ_{hyb} is a finite set of events associated to the transitions and generated from the invariants of each location. The guard conditions are boolean conditions that may depend on continuous variables [5]. Without loss of generality, we assume that the model is deterministic i.e. whenever $q_i \xrightarrow{\sigma} q_j$ and $q_i \xrightarrow{\sigma} q_k$ then $q_j = q_k$ for each $(q_i, q_j, q_k) \in Q^3$ and each $\sigma \in \Sigma$. Σ_{hyb_o} is the set of observable events (i.e events associated to the activation of observable transition guards) and $\Sigma_{hyb_{uo}}$ is the set of unobservable events
- $CSD \supseteq \bigcup_i CSD_i$ is the *Causal System Description*, or causal model, used to represent the constraints underlying the continuous dynamics of the hybrid system. Every CSD_i is given by a graph $(\mathcal{X} \cup \mathcal{D}, A)$. There is an arc $a(v_i, v_j) \in A$ from $v_i \in \mathcal{X} \cup \mathcal{D}$ to $v_j \in \mathcal{X} \cup \mathcal{D}$ if variable v_i influences variable v_j
- $Init \in \mathcal{X} \times \mathcal{D}$ is the initial condition.

3.2 Invariants and event generation

For analysis purposes, it is often useful to abstract a system in a way that preserves the properties being analyzed while hiding the details that are of no interest Alur et al. (2000). This section presents how the domain value of continuous variables is abstracted in terms of a finite set of qualitative values so that qualitative signatures can be defined for every system mode, providing *signature invariants*. A second type of invariant, namely *configuration invariants*, are then introduced to map the system mode to the modes of the components forming the system. We then present two kinds of event generators:

- signature invariants are used to generate *induced events* that inform about the changes of continuous dynamics;
- configuration invariants are used to generate *configuration events* that inform about the changes of configuration.

Qualitative abstraction of continuous variables The idea is to partition the domain value of continuous variables into a finite number of labels such that the label remains invariant when the system is operating within a given mode. A function $f_{D(x_i) \rightarrow \overline{D}(x_i)}$ that maps the continuous domain $D(x_i) \subseteq \mathbb{R}$ of a continuous variable $x_i \in \mathcal{X}$ into a finite discrete domain $\overline{D}(x_i) \subseteq \mathcal{P}(\mathbb{R})$ is defined. $\overline{D}(x_i)$ is generally built from a partition of $D(x_i)$. The associated qualitative variable is noted $\overline{x}_i$. We have $\overline{D}(x_i) = D(\overline{x}_i)$. The qualitative abstraction function is defined as:

$$f_{D \rightarrow \overline{D}} : D(x_i) \longrightarrow \overline{D}(x_i) = D(\overline{x}_i). \quad (1)$$

After the variable abstraction step, transition guards that originally depend on continuous variables of $\mathcal{X}$ must be rewritten in terms of the corresponding abstract variables of $\overline{\mathcal{X}}$. In choosing the abstraction functions, one must pay attention to the fact that the transition guards must remain expressible. For instance, if a guard is given by the condition $x_i > k$, k being a constant, than k must be among the landmarks of the partition of $D(x_i)$ leading to $\overline{D}(x_i)$.

Qualitative mode signature and signature invariant This concept captures the expected qualitative values of the observable continuous variables within a given mode. It

[2] We assume that the set of system observable variables is the same in all system modes. This assumption is generally verified when the set of system's sensors is permanent.

[3] The observability of discrete variables is indirectly achieved through the observability of a subset of events.

[4] $\otimes$ is the Cartesian product.

[5] Guard conditions are evaluated through specific monitors that also take as input a set of boolean variables Φ.

characterizes the expected behavior of the system in this mode w.r.t all the other modes and provides a mode invariant.

Definition 1. (Qualitative Mode Signature). The qualitative signature of a mode q_i noted $Sig(q_i)$ is the vector of discrete values $[\overline{x_j}]_{OBS}$ taken by the variables $\overline{x_j} \in \mathcal{X}_{OBS}$ in this mode:

$$Sig(q_i) = [\overline{x}_i]_{OBS/q_i} \tag{2}$$

Definition 2. (Qualitative mode partial signature). Any sub vector of $Sig(q_i)$ is defined as a *partial signature* of q_i.

The qualitative signature of a mode q_i is the *signature invariant* of this mode, i.e $Inv_{\mathcal{X}}(q_i) = Sig(q_i)$.

Induced events generated by signature invariants Following the idea of Bayoudh et al. (2008b), signature invariants are used to generate *induced events* that inform us about the changes of continuous dynamics. A set of events Σ^{Sig} is defined through an event generator $f_{Sig\to\sigma}$ that maps signature invariants to discrete events. An event of Σ^{Sig} is associated to every transition of Γ and depends on the signature invariant of the source mode versus the destination mode. The event generator $f_{Sig\to\sigma}$ and the set of events Σ^{Sig} are defined as follows:

$$f_{Sig\to\sigma} : Q \times T \longrightarrow \Sigma^{Sig}$$
$$(q_i, q_j) \longmapsto \begin{cases} r^{o}_{i,j} \in \Sigma^{Sig}_{o} \text{ if } Sig(q_i) \neq Sig(q_j) \\ r^{uo}_{i,j} \in \Sigma^{Sig}_{uo} \text{ if } Sig(q_i) = Sig(q_j) \end{cases} \tag{3}$$

where Σ^{Sig}_{o} (resp. Σ^{Sig}_{uo}) is a set of observable (resp. unobservable) events generated when the mode signature of the source mode is different (resp. equal) from the mode signature of the destination mode. $\Sigma^{Sig} = \Sigma^{Sig}_{o} \cup \Sigma^{Sig}_{uo}$.

Remark 3. (Incomplete set of observed variables). The diagnosis iterates starting from one or a reduced set of the observed variables $\mathcal{X}^{\text{current}}_{\text{OBS}} \subseteq \mathcal{X}_{\text{OBS}}$ and proposing additional variables within the observable set $\mathcal{X}_{\text{OBS}}$ to be measured. It is important to notice that while $\mathcal{X}^{\text{current}}_{\text{OBS}}$ is a strict subset of $\mathcal{X}_{\text{OBS}}$, any of the induced events are unobservable.

Configuration invariant The system is assumed to be composed of a set of components denoted by COMP. The system mode hence results from the mode of every component $C_i \in$ COMP, or the configuration, which is represented by the variables $K_i \in \mathcal{K}$. A second type of invariant, namely the *configuration invariant*, is now introduced to map the system mode to the configuration.

Definition 4. (Configuration invariant). The *configuration invariant* $Inv_{\mathcal{K}}(q_i)$ of a mode q_i is defined as the vector of discrete variables $[K_k]$, $K_k \in \mathcal{K}$ valuated according to the modes of the underlying components, i.e. the configuration $Conf(q_i)$ corresponding to this mode.

Configuration events generated by configuration invariants Configuration invariants are used to generate *configuration events* that inform us about the changes of configuration. The event generator $f_{Conf\to\sigma}$ and the set of events Σ are defined as follows:

$$f_{Conf\to\sigma} : Q \times T \longrightarrow \Sigma$$
$$(q_i, q_j) \longmapsto \begin{cases} \sigma^{o}_{i,j} \in \Sigma_o \text{ if } Conf(q_i) \neq Conf(q_j) \\ \sigma^{uo}_{i,j} \in \Sigma_{uo} \text{ if } Conf(q_i) = Conf(q_j) \end{cases} \tag{4}$$

The behavior automaton The generation of the events Σ^{Sig} and Σ allows us to abstract the hybrid automaton Γ by a discrete event model that we define as the *behavior automaton*, denoted $B_A(\Gamma) = (Q_{beh}, \Sigma_{hyb}, T_{beh}, q_0)$, where $q_0 \in Q$ is the initial condition. The behavior automaton is obtained by defining a set of *transient* modes Q_t that model the continuous dynamics reaction to the occurrence of a mode change, and hence lead to the generation of an *induced discrete event* of Σ^{Sig}. The set of transient modes is obtained through a bijective function that associates a transient mode $q_{i,j}$ to each transition $t(q_i, \sigma_{i,j}, q_j) \in T$ of the original hybrid system Γ.The set of modes of the behavior automaton is given by $Q_{beh} = Q \cup Q_t$ and the transition function is $T_{beh} \subseteq (Q_{beh} \times \Sigma_{hyb} \longrightarrow Q_{beh})$, with $\Sigma_{hyb} = \Sigma_{hyb_o} \cup \Sigma_{hyb_{uo}}$ with $\Sigma_{hyb_o} = \Sigma_o \cup \Sigma^{Sig}_{o}$ and $\Sigma_{hyb_{uo}} = \Sigma_{uo} \cup \Sigma^{Sig}_{uo}$. The behavior automaton $B_A(\Gamma) = (Q_{beh}, \Sigma_{hyb}, T_{beh}, q_0)$ is obtained by considering that on the occurrence of an event $\sigma_{i,j} \in \Sigma$ that triggers a transition from mode q_i to mode q_j, the system passes by a transient mode $q_{i,j}$ in which the transition is not yet effective Bayoudh et al. (2008a). The transition to mode q_j is confirmed by the occurrence of the corresponding induced event $r^{o/uo}_{i,j}$, providing evidence that the continuous dynamics match the targeted mode.

3.3 The partial diagnoser

We apply the diagnoser approach of Sampath et al. (1995), hence providing an extension to hybrid systems and to normal behavior of he system. The diagnoser we propose can be viewed as a partial diagnoser as it doesn't integrate all the fault trajectories but only some fault trajectories corresponding to anticipated faults and the trajectory corresponding to the normal behavior of the system.

The diagnoser of the hybrid system is a deterministic finite state machine built from the behavior automaton, $PDiag(B_A(\Gamma)) = (Q_{PD}, \Sigma_{PD}, T_{PD}, q_{PD_0})$ where:

- $q_{PD_0} = \{(q_0, \emptyset)\}$ is the initial state of the partial diagnoser (assuming Γ is normal to start with);
- $\Sigma_{PD} = \Sigma_{hyb_o}$ is the set of all observable events of the system;
- $Q_{PD} \subseteq 2^{Q_{beh} \times \mathcal{L}}$ is the set of states of the partial diagnoser where $\mathcal{L} = 2^{\Sigma_{hyb_{uo}}}$ The states of the partial diagnoser provide a set of couples whose first element refers to the state of the behavior automaton and the second is a label providing the unobservable events on the path leading to this state. In other words, an element $q_{PD} \in Q_{PD}$ is a set $q_{PD} = \{(q_1, l_1), (q_2, l_2), \ldots, (q_n, l_n)\}$, where $q_i \in Q_{beh}$ and $l_i \in \mathcal{L}$.
- $T_{PD} \subseteq Q_{PD} \times \Sigma_{hyb_o} \to Q_{PD}$ is the partial transition function of the diagnoser defined as follows:

$$T_{PD}(q_{PD}, \sigma) = \bigcup_{\substack{(q,l)\in q_{PD} \\ s\in L_\sigma(\Gamma,q)}} \{(\mathcal{T}_{beh}(q, s), LP(q, l, s))\} \tag{5}$$

$\mathcal{T}_{beh}(q, s)$ is the recursive application of T_{beh} along the string $s = s_1.s_2.\ldots.s_n.\sigma$ of events defined as $T_{beh}(q, s) = T_{beh}(\ldots T_{beh}(T_{beh}(q, s_1), s_2), \ldots, s_n), \sigma)$.

Remark 5. (Incomplete set of observed variables). When the set of observed variables at hand $V_{\text{OBS}}^{\text{current}}$ is a strict subset of V_{OBS}, the corresponding partial diagnoser $PDiag^*(B_A(\Gamma))$ is obtained from $PDiag(B_A(\Gamma))$ by changing the status of all observable induced events $r_{i,j}^o$ into unobservable events $r_{i,j}^{uo}$.

4. CAUSAL MODEL BASED DIAGNOSIS

4.1 Modeling a system with a causal graph

Causal models have been shown to be suitable for diagnosis in several pieces of work Biswas et al. (2006); de Kleer and Williams (1987); Gentil et al. (2004); Travé-Massuyès et al. (2001). Causal models are supported by an oriented *causal graph* in which an oriented edge from vertex v_i to vertex v_j exists if the cause variable v_i has an influence on the effect variable v_j, i.e. if a value change of variable v_i affects the value of variable v_j. v_i and v_j are called the *cause* and the *effect* variables of the influence, respectively. Influences represent the causal structure of the equational model but they may also capture behavioral information when adequately labelled Leyval et al. (1994); Travé-Massuyès and Calderon-Espinoza (2007). Based on the work by Iwasaki and Simon (1986), Travé-Massuyès and Pons (1997) automatically derive the causal graph of a multimode system in an incremental way.

In this work, the variables at hand belong to the set $\mathcal{V} = \overline{\mathcal{X} \cup \mathcal{K}}$. Let us define $\mathcal{V}_{OBS} \subseteq \mathcal{V}$ the subset of observed variables. Every influence is labelled by a symbol $C_j \subseteq$ COMP standing for its supporting physical component and by a boolean *activation condition* that depends on (discretized) continuous and/or discrete variables allowing us to model the underlying multimode system. An observed variable is qualified as *normal* (resp. *misbehaving*) at some time point when there is a match (resp. discrepancy) between the measured value and the predicted value; it is noted $\text{OK}(v_i)$ (resp. $\neg\text{OK}(v_i)$). The corresponding vertices are labelled accordingly. The causal graph is explored backwards to determine the cause(s) of discrepancies using the results of the logical theory of model-based diagnosis (MBD) summarized in the following sections Reiter (1987).

4.2 The diagnosis problem

Given CSD the Causal System Description, COMP the set of components and OBS the set of observations at some time point, the diagnosis problem can be defined as below.

Definition 6. (Diagnosis problem). It is a triple (CSD, COMP, OBS) where (CSD, COMP) is the causal system model and OBS a set of observations i.e. a tuple that qualifies every observed variable v_i as $\neg\text{OK}(v_i)$ or $OK(v_i)$.

The set of observations OBS defines a partial labelling of the vertices of CSD. When some vertices are labelled ¬OK, the diagnosis system must derive all sets of faulty components of COMP that are consistent with the observations OBS. Given a vertex of CSD with non zero in-degree, the set of in-going influences have all the same supporting component $C_i \in$ COMP and the spanned subgraph $\mathcal{G}_{C_i}$ is the causal representation of the behaviour of C_i; the set of input vertices is noted $V_{C_i}^{\text{in}}$, the output vertex $v_{C_i}^{\text{out}}$ and the bunch of influences $\mathcal{I}_{C_i}$ (Fig. 2).

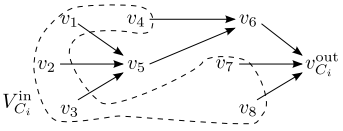

Fig. 2. Causal graph of component C_i.

The component C_i is AB (abnormal) if and only if at least one of the influences in $\mathcal{I}_{C_i}$ is AB; it is qualified ¬ AB otherwise. In the same way, $V_{C_i}^{\text{in}}$ is qualified as ¬OK if and only if at least one of its vertex elements is ¬OK, OK otherwise. Two diagnosis assumptions can be adopted:

- *single fault:* if C_i is AB then $V_{C_i}^{\text{in}}$ is OK (i.e. one component is faulty at a time);
- *exoneration:* if $v_{C_i}^{\text{out}}$ is OK then C_i is ¬ AB.

The labels of $V_{C_i}^{\text{in}}$ and $v_{C_i}^{\text{out}}$ are constrained by a relational model shown on table 1 which is generic to all components. From the partial labelling of CSD defined by OBS, it derives the consistent component health status assignments by propagating the labels through the causal graph.

Definition 7. (Diagnoses and minimal diagnoses). A diagnosis for (CSD, COMP, OBS) is a set of components $\Delta \subseteq$ COMP such that the assignment $\text{AB}(Ci)$ for $Ci \in \Delta$ and $\neg\text{AB}(Ci)$ for $Ci \in \text{COMP} - \Delta$ is consistent with CSD and OBS.

4.3 Components, conflicts and diagnoses

The notion of *conflict* (in the sense of Reiter) plays an important role because of its relation with diagnoses.

Definition 8. (Reiter conflict and minimal conflict). The conflict in the sense of Reiter (1987), or R-conflict, for (CSD, COMP, OBS) is a set of components $S = \{C_1, \ldots, C_k\} \subseteq$ COMP such that the assignment of ¬AB to any $C_i \in S$ is inconsistent. A minimal conflict is a conflict which does not strictly include any conflict.

Table 1. Component model.

C_i	$V_{C_i}^{\text{in}}$	$v_{C_i}^{\text{out}}$
¬AB	OK	OK
¬AB	¬OK	OK
¬AB	¬OK	¬OK
AB	OK	¬OK

Interpreting the notion of *R-conflict* requires to define the notion of *Observed Macro-Component (OMC).*

Definition 9. (Observed Macro-Component). An OMC $\mathcal{C}_i$ is defined by a in-degree output vertex $v_{\mathcal{C}_i}^{\text{out}} \in \mathcal{V}_{\text{OBS}}$ and a set of input vertices $v_i \in V_{\mathcal{C}_i}^{\text{in}}$ defined as the first observed predecessors of $v_{\mathcal{C}_i}^{\text{out}}$. The behavior of $\mathcal{C}_i$ is represented by the subgraph of CSD $\mathcal{G}_{\mathcal{C}_i}$ given by the in-tree in which only $v_{\mathcal{C}_i}^{\text{out}}$ is reachable from every other vertex.

Definition 10. (Test and covered components). The labels of an OMC $\mathcal{C}_i$ is defined as a *test* T_i based on $v_{\mathcal{C}_i}^{\text{out}}$: if $v_{\mathcal{C}_i}^{\text{out}}$ is labelled ¬OK, than the test is said to *fail* and if it is labelled OK, the test is said to *pass*. The components $C_{j_1}, \ldots, C_{j_{K_i}}$ are called the *covered components* of T_i.

The labelling model for a component given in table 1 extends to OMCs when considering an exoneration assumption corresponding to the *RRA-exoneration* assumption Cordier et al. (2004).

Conflict sets are sets of components which cannot behave normally altogether according to the observations. Conflicts can be identified according to the following result:

Proposition 11. (Conflict identification). The set of components $\{C_{j_1}, \ldots, C_{j_{K_i}}\}$ covered by a test T_i that fails and whose input label is OK, i.e. $V_{\mathcal{C}_i}^{\text{in}}$ is OK, defines a conflict.

Proposition 12. (Diagnosis). $\Delta \subseteq$ COMP is a (minimal) diagnosis for (CSD, COMP, OBS) if and only if Δ is a (minimal) hitting set for the collection of (minimal) conflict sets of (CSD, COMP, OBS) Reiter (1987).

Proposition 13. (Exonerated components). If T_i passes, the covered components $C_{j_1}, \ldots, C_{j_{K_i}}$ can be exonerated.

Under the single fault assumption, diagnoses are obtained by computing the intersection of conflict sets. If, in addition, the RRA-exoneration assumption is used, the exonerated components can be removed. The result provides the *ambiguity set* $\mathcal{A}$ in which every element is a diagnosis.

5. CONSISTENCY BASED HYBRID DIAGNOSIS

A diagnosis consistency based-approach relies on the use of a reference model, in our case the partial diagnoser $PDiag(B_A(\Gamma))$ and the causal System Description CSD, and on the observation of the real behaviour of the monitored system. This observation takes the form of an observed sequence of events $s_{obs} \in L_{obs}(\Gamma, q) = \{s \in L(\Gamma, q) \mid s = u\sigma, u \in \Sigma^*_{hyb_o}, \sigma \in \Sigma_{hyb_o}\}$, i.e. s_{obs} is a word of $L_{obs}(\Gamma, q)$. The events may be natural discrete events or induced events, coming from the continuous dynamics.

5.1 Interfacing event-based and variable-based diagnosis reasoning

Diagnosis interlinks event-based and variable-based reasoning interfaced by the addition, in CSD, of a set of vertices corresponding to discrete variables $\mathcal{K}$. They represent the modes of the different components of the system. For instance, in the case of a switch, the corresponding discrete variable materializes the two modes, open or closed. The influences outgoing these vertices are different since they influence the occurrence of events that act on the causal graph structure.

They are represented by dotted lines as illustrated in Fig. 7 and have an event and a standard influence as destination node. They are labelled by the corresponding underlying multi-mode component.

The event nodes are labelled $OK/\neg OK$ depending on whether they are actually observed or not compared to what is expected from the model. These labels are then used in the same way as the other labels in the causal diagnosis procedure explained in section 5.2.

5.2 Diagnosis steps

The consistency based hybrid diagnosis algorithm is structured along the following iterated steps:

Step 1: Fault detection and reference mode hypothesis generation It is achieved by synchronizing $PDiag^*(B_A(\Gamma))$ [6] with s_{obs}. If none synchronized trajectory corresponds to a complete trajectory then a fault is detected. The last state of each synchronized trajectory indicates a possible reference mode for checking the consistency, and possibly explaining the inconsistency. These modes are put in set Q_{ref} and the ambiguity set is initialized to $\mathcal{A} =$ COMP.

Step 2: Diagnosis hypothesis generation Every hypothesized reference mode $q_i \in Q_{ref}$ provides evidence about the faulty situation. For every $q_i \in Q_{ref}$, we consider the corresponding CSD_i and apply the consistency based causal diagnosis approach to obtain an ambiguity set $\mathcal{A}_i$. Global *ambiguity set* is updated as $\mathcal{A} = \text{COMP} \cap \bigcap_i \{\mathcal{A}_i\}$.

Step 3: Test selection This step determines the best next variable $x_i \in \mathcal{X}_{\text{OBS}} - \mathcal{X}_{\text{OBS}}^{\text{current}}$ to be tested to maximize ambiguity reduction. It is detailed below in section 5.3.

Step 4: Hypothesis discrimination The current ambiguity set is reduced by going to step 2 or to step 1 when the set of observed variables is complete, i.e. $\mathcal{X}_{\text{OBS}}^{\text{current}} = \mathcal{X}_{\text{OBS}}$, and $PDiag^*(B_A(\Gamma))$ must be replaced by $PDiag(B_A(\Gamma))$.

5.3 Test selection

Given an ambiguity set $\mathcal{A}$, the goal of the test selection step is to determine the best next test T_i based on a variable $x_i \in \mathcal{X}_{\text{OBS}} - \mathcal{X}_{\text{OBS}}^{\text{current}}$. It should maximize diagnostic information while minimizing the overall testing cost C_T [7] . A standard heuristics would be the *Information Gain* relying on entropy: it is based on a theoretical measurement of the quality of the current diagnosis, the probability of the test passing or failing de Kleer and Williams (1987). It requires costly on-line calculations based on the results of previously executed tests.

The method for our test selection procedure has been proposed by Gonzalez-Sanchez et al. (2011) as the *gReedy diAgnostic Prioritization by ambiguiTy Reduction* (RAPTOR) method. It is based on maximizing diagnosis ambiguity reduction. Performance is expressed in terms of a cost metric C_d that measures the excess effort incurred in finding the faulty component. Tests are characterized by their *coverage*, i.e. set of covered components, whereas components are characterized by their *signature*, i.e. set of tests that cover/don't cover the component. This information is summarized in the *signature matrix* $\mathcal{S}$ for which lines correspond to component signatures and columns correspond to tests and their coverage (table 4). Two components having the same signature cannot be discriminated. Ambiguity groups are defined as sets of such components. Consider $AG = \{g_1, g_2, \ldots, g_L\}$ be the ambiguity groups generated by the submatrix $\mathcal{S}^k$ of $\mathcal{S}$ corresponding to the k previously executed tests. The expected diagnostic effort if components were picked randomly in g_i is:

$$E[C_{d_i}] = \frac{|g_i| - 1}{2} \tag{6}$$

RAPTOR considers that faults are distributed uniformly through the system, hence $Pr(g_i) = |g_i|/|\text{COMP}|$. Averaging the effort in each group g_i by this probability :

$$G(AG) = \sum_{i=1}^{L} Pr(g_i) E\left[C_{d_i}\right] = \sum_{i=1}^{L} \frac{|g_i|}{|\text{COMP}|} \frac{|g_i| - 1}{2} \tag{7}$$

[6] or $PDiag(B_A(\Gamma))$ if $\mathcal{X}_{\text{OBS}}^{\text{current}} = \mathcal{X}_{\text{OBS}}$

[7] The cost is not taken into account yet in this study

$G(AG)$ estimates the residual diagnostic effort C_d and can be seen as an estimation of diagnosis quality.

Each executed test breaks each ambiguity group into two smaller ambiguity groups, one corresponding to the components covered by the test, and one corresponding to the components that are not covered. The ambiguity reduction heuristic is defined as the difference in ambiguity caused by appending one more test T_i to the test matrix :

$$AR(\mathcal{S}, T_i) = G(AG(\mathcal{S})) - G(AG(\mathcal{S}\|T_i)) \quad (8)$$

In our hybrid framework, one has to deal with multiple signature tables, each corresponding to one of the reference modes in Q_{ref}. The ambiguity sets resulting from these multiple references are intersected to obtain the resulting ambiguity set; the test selection strategy must be applied to each signature table and the overall best test is chosen.

6. APPLICATION TO AN AUTOMOTIVE EMBEDDED FUNCTION

In the automotive field, the use of electronic systems to control functions (fuel injection, ABS) has considerably increased. In these electronic systems, Electronic Control Units (ECU) impose discrete switching between the several behavioral modes. The coupling of software and hardware capacities exhibits the hybrid nature of the system through complex patterns of behavior and numerous nominal modes.

6.1 System description

We consider the rear windscreen wiper of a car whose simplified synoptic is depicted on figure 3.

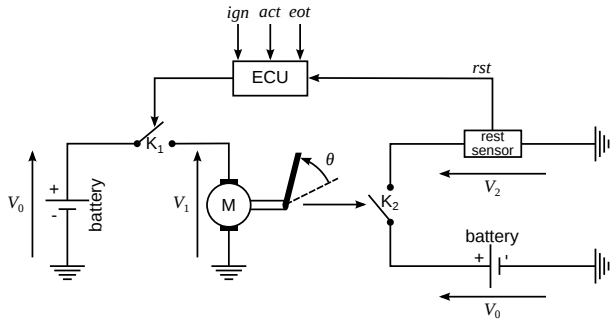

Fig. 3. Rear windscreen wiper synoptic.

When the driver acts on the actuator *act*, the electronic control unit ECU closes the switch K_1 and then supplies electrical power from the battery *bat* to the wiper motor M. The rotational move of the motor flange is transformed into an alternative straight move via the wiper linkage that allows the wiper to wipe the screen. After the wiper moved forward and backward on the screen, it closes the switch K_2, supplying electrical power to the wiper rest position sensor *rst*. The sensor sends a signal back to the ECU to indicate that the wiper is in its rest position. The ECU then opens the switch K_1 for a given timeout: the wiper motor is no longer supplied with power and the wiper stays in its rest position for a given timeout after which the ECU closes again the switch K_1. The wiper moves forward and backward on the screen, stops during timeout, etc. until the driver turns the actuator off.

The discrete variables are *ign* (ignition status), *act* (actuator position), *rst* (wiper rest position) and *eot* (end of timeout). They are boolean variables and they generate the σ events of table 3 when their values switch from true to false and conversely. The continuous variables are the battery voltage V_0, the wiper motor input voltage V_1, the rest position sensor output voltage V_2, the wiper angular velocity Ω and its angular position θ.

6.2 Building the models for diagnosis

The underlying DES is directly issued from the specification data of the function and corresponds to a simplified version of the control embedded in the ECU. The event based abstraction of the continuous dynamics is given through the mode signatures definition and induced events. Qualitative signatures are obtained from hybrid system simulation techniques (using Modelica) underlying the fault dictionary method. The case study contains two switches: K_1 lies in the ECU and allows electrical power delivery to the wiper motor and K_2 gives information to the ECU about the wiper rest position. The system has thus four different behavioral modes. The continuous variables and their abstracted values providing the qualitative mode signatures are shown in table 2 and the modes switching events are listed in table 3.

Table 2. Continuous variables and qualitative mode signatures.

	q_0 Off	q_1 On	q_2 Wiping	q_3 Timeout
V_0	1	1	1	1
V_1	0	0	1	0
V_2	1	1	1	0
Ω	0	0	1	0

Table 3. System modes and events.

σ_{ign}	σ_{act}	σ_{rst}	σ_{eot}
$q_0 \longrightarrow q_1$	$q_1 \longrightarrow q_2$	$q_2 \longrightarrow q_3$	$q_3 \longrightarrow q_2$

As an example of abstraction, consider the voltage V_1. The abstract values "0" and "1" represent the real values "0V $\pm\epsilon$" (ground voltage) and "12V $\pm\epsilon$" (battery voltage). The figure 4 shows the behavior automaton of the system with the *diagnosis aware* events associated to the signature changes and the *transient* modes that model the continuous dynamics reaction due to a mode change.

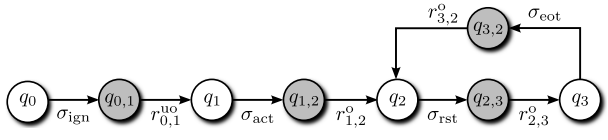

Fig. 4. Behavior automaton.

6.3 Diagnosis reasoning

The fault scenario is the following: the wiper motor coil is broken (opened circuit), the command to move the wiper is sent by the ECU (σ_{act} is issued) and the motor is powered but the wiper obviously cannot move. The wiper hence never gets to the rest position and the ECU never receives

the rest position event σ_{rst}. We assume for the sake of simplicity of the case study that the fault is not on the battery neither its connections.

The discrete events are always observable as they are linked to the state of the ECUs and can be obtained by a reading of the ECU parameters, i.e. $\Sigma_o = \{\sigma_{\text{ign}}, \sigma_{\text{act}}, \sigma_{\text{rst}}, \sigma_{\text{eot}}\}$. The car mechanic starts the diagnosis session by measuring the input voltage of the wiper motor, i.e. $\mathcal{V}_{\text{obs}}^{\text{current}} = \{V_1\}$. The observed measurement is plotted on figure 5. The system is able to issue the induced

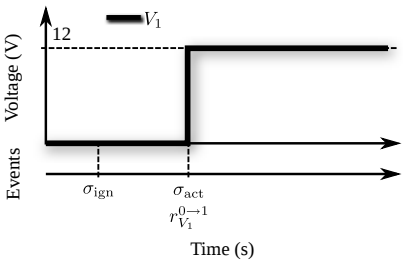

Fig. 5. V_1 observed behaviour.

events if any, and to deliver the observed event sequence $s_{\text{obs}} = \{\sigma_{\text{ign}}, \sigma_{\text{act}}\}$. Let us notice that the occurrence of the event $r_{V_1}^{0\to1}$ (V_1 value transitioning from 0 to 1) is not a sufficient condition for $r_{1,2}^{o}$ to be issued.

Step 1: Fault detection and reference mode hypothesis generation The synchronization of the observed event sequence with $PDiag^*(B_A(\Gamma))$ (see figure 6, in which the labels are not shown for sake of simplicity) indicates that the system can be synchronized along the sub-trajectory $[\{q_0\}, \{q_{0,1}, q_1\}, \{q_{1,2}, q_2\}]$. This is not a complete trajectory, hence a fault is detected and $Q_{ref} = \{q_1, q_2\}$, as $q_{1,2}$ is semantically equivalent to q_1 or q_2. The diagnosis is within the ambiguity set $\mathcal{A} =$ {motor, wiper, rest sensor, K_1, K_2, ECU }.

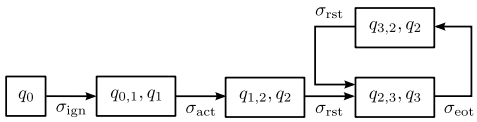

Fig. 6. Partial diagnoser $PDiag^*(B_A(\Gamma))$.

Step 2: Diagnosis hypothesis generation The graphs of the figure 7 represent the causal models of the modes q_1 and q_2. The dotted influences indicate actions by K_1 and K_2, a thin line represent an influence that is not active in the current mode.

Interlinking temporally the observed events σ_{ign}, σ_{act} and $r_{V_1}^{0\to1}$ as shown in figure 5 provides the observed (partial) signatures for every synchronized state of $PDiag^*(B_A(\Gamma))$ (Fig. 8). s_{obs} includes event σ_{act}, so K_1 is labelled OK in q_1. Oppositely, s_{obs} does not include σ_{rst} and K_2 is labelled ¬OK in q_2. These must be compared to the theoretical partial signatures of every mode as given in table 2, from which one obtains the labelling of the observed vertices of the two causal models. In both modes q_1 and q_2, V_1 is labelled OK so the corresponding test T_{V_1} passes.

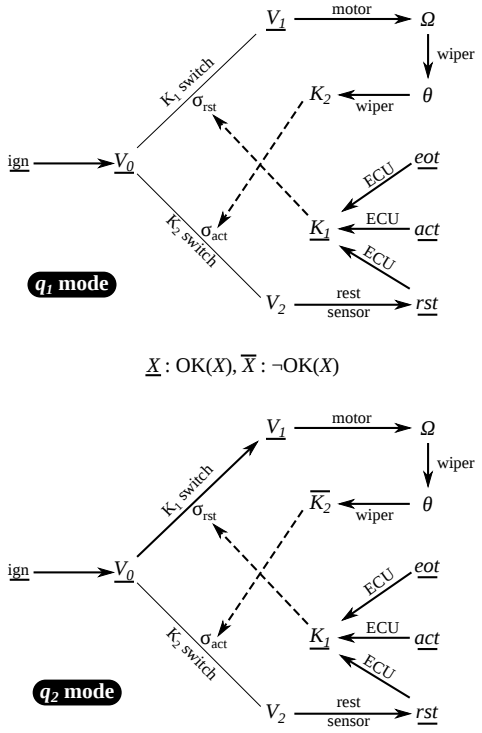

Fig. 7. Causal models of modes q_1 and q_2.

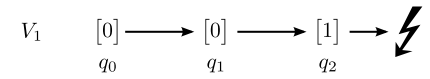

Fig. 8. Observed partial signatures based on V_1.

In this case study, it is not appropriate to use the exoneration assumption. The conflict indicated by the label of K_2 consists in $\mathcal{A} = \{K_2$, wiper, motor$\}$

Step 3: Test selection Having considered the events and T_{V_1}, we have the signature table 4.

Table 4. Test selection table for modes q_1 & q_2.

q_1 / q_2	T_{V_1}	T_{V_2}	T_θ	T_Ω
motor	0/0	0/0	1/1	1/1
wiper	0/0	0/0	1/1	0/0
rest sensor	0/0	0/0	0/0	0/0
K_2	0/0	1/0	0/0	0/0
K_1	1/0	0/0	0/0	0/0

The ambiguity group is $g_1 = \{K_2$, wiper, motor$\}$, and $G(AG) = \frac{3}{5}\frac{3-1}{2} = \frac{3}{5} = 0.6$. T_θ and T_Ω break the ambiguity group in two groups. The new ambiguity groups are more balanced for T_θ. We get $G(AG\|T_\theta) = \frac{2}{5}\frac{2-1}{2} + \frac{3}{5}\frac{3-1}{2} = \frac{2}{5} + \frac{3}{5} = 1$ versus $G(AG\|T_\Omega) = \frac{1}{5}\frac{1-1}{2} + \frac{4}{5}\frac{4-1}{2} = \frac{12}{10} = 1.2$. The test T_θ is hence proposed.

The signal for θ remains flat at $\theta = 0$, which indicates a conflict in mode q_2 and the ambiguity set is reduced to $\mathcal{A} = \{\text{motor}, \text{wiper}\}$. The same reasoning indicates to test Ω, providing a conflict in mode q_2, leading to $\mathcal{A} = \{\text{motor}\}$ and the final single component diagnosis $\Delta = \{\text{motor}\}$.

7. CONCLUSION

This paper addresses the diagnosis problem of hybrid systems by proposing a theoretical framework merging ideas from discrete event diagnosis and from continuous systems.

The hybrid behavior automaton is abstracted into a pure discrete event model using qualitative fault signature. A partial diagnoser is built and used for detection. A set of possible modes for checking the consistency and possibly explaining the inconsistency is pointed out. In each incriminated mode a consistency-based causal diagnosis is applied. The ambiguity set is reduced in an iterative way using a test selection procedure to determine the additional information which allows the best discrimination among the diagnostic hypothesis.

This method does not require the availability of fault models and then can be be viewed as an extension aiming at complementing an available fault dictionary based method with a consistency based method designed for hybrid systems. Indeed, a fault dictionary based method is very powerful to diagnose extreme faults (i.e. faulty parameter values null or infinite) which are easily anticipated. However, when the actual fault is out of the anticipated set, for instance the fault is a parameter deviation, the fault dictionary method is no more suitable. Future work will consolidate this approach with a series of test cases. How to apply the exoneration assumption in the case of circuits which may have non powered branches needs further investigation. The method may be extended to a distributed framework, which would allow to avoid the combinatorial problem related to the number of operation modes.

ACKNOWLEDGEMENTS

This work has been developed in collaboration with ACTIA Automative company in the context of the AMIC-TCP project. The authors thank Jérôme Thomas and Hervé Poulard from ACTIA for their help.

REFERENCES

Alur, R., Henzinger, T., Lafferriere, G., and Pappas, G.J. (2000). Discrete abstractions of hybrid systems. In *Proceedings of the IEEE*, 971–984.

Bayoudh, M., Travé-Massuyès, L., and Olive, X. (2008a). Coupling continuous and discrete event system techniques for hybrid system diagnosability analysis. In *Proceeding of the 2008 conference on ECAI 2008: 18th European Conference on Artificial Intelligence*, 219–223. IOS Press, Amsterdam, The Netherlands, The Netherlands.

Bayoudh, M., Travé-Massuyès, L., and Olive, X. (2008b). Hybrid systems diagnosis by coupling continuous and discrete event techniques. *Proceedings of the IFAC World Congress Seoul Korea*, 7265–7270.

Biswas, S., Sarkar, D., Mukhopadhyay, S., and Patra, A. (2006). Diagnosability analysis of real time hybrid systems. In *Proceedings of the IEEE International Conference on Industrial Technology, (ICIT)*, 104–109. Mumbai, India.

Cordier, M.O., Dague, P., Lévy, F., Montmain, J., Staroswiecki, M., and Travé-Massuyès, L. (2004). Conflicts versus analytical redundancy relations : A comparative analysis of the model-based diagnostic approach from the artificial intelligence and automatic control perspectives. *IEEE Transactions on Systems, Man and Cybernetics - Part B.*, 34(5), 2163–2177.

de Kleer, J. and Williams, B.C. (1987). Diagnosing multiple faults. *Artificial Intelligence*, 32(1), 97–130. doi:http://dx.doi.org/10.1016/0004-3702(87)90063-4.

Esser, M. and Struss, P. (2007). Fault-model-based test generation for embedded software. In *20th International Joint Conference on Artificial Intelligence*, 342347. Hyderabad, India.

Faure, P.P. (2001). *Une approche à base de modèles fondés sur les intervalles pourla génération automatique d'arbres de diagnostic optimaux. Applicationau domaine de l'automobile.* Ph.D. thesis, Université Paris 13, Paris, France.

Gentil, S., Montmain, J., and Combastel, C. (2004). Combining fdi and ai approaches within causal model-based diagnosis. *IEEE Transactions on Systems, Man and Cybernetics*, 34(5), 2207–2201.

Gertler, J. (1998). *Fault Detection and Diagnosis in Engineering Systems.* Marcel Deker.

Gonzalez-Sanchez, A., Abreu, R., Gross, H., and Gemund, A.V. (2011). Raptor: Greedy diagnostic prioritization by ambiguity group reduction. In *Proceedings of the 22nd International Workshop on Principles of Diagnosis, DX2011*, 84–91.

Henzinger, T. (1996). The theory of hybrid automata. In *Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science, LICS'96*, 278–292. IEEE Computer Society Press, New Brunswick, New Jersey, USA.

Iwasaki, Y. and Simon, H. (1986). Causality in device behaviour. *Artificial intelligence*, 29(1–3), 63–67.

Lehmann, D. and Lunze, J. (2009). *Handbook of Hybrid Systems Control.* Cambridge University Press.

Leyval, L., Gentil, S., and Feray-Beaumont, S. (1994). Model-based causal reasoning for process supervision. *Automatica*, 30(8), 1295–1306.

Olive, X. (2003). *Approche intégrée à base de modèles pour le diagnostic hors ligne et la conception. Application au domaine de l'automobile.* Ph.D. thesis, Université Paul Sabatier, Toulouse, France.

Price, C., Pugh, D.R., Wilson, M.S., and Snooke, N. (1995). The flame system: Automating electrical failure mode effects analysis (fmea). In *Annual Reliability and Maintainability Symposium.* Washington D.C., USA.

Reiter, R. (1987). A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1), 57–95.

Ressencourt, H. (2008). *Diagnostic hors-ligne à base de modèles : approche multi-modèle pour la génération automatique de séquences de tests. Application au domaine de l'automobile.* Ph.D. thesis, Université Paul Sabatier, Toulouse, France.

Sachenbacher, M. and Struss, P. (2001). Aqua : A framework for automated qualitative abstraction. In *In Proceedings of 15th International Workshop on Qualitative Reasoning*, 5–12.

Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K., and Teneketzis, D. (1995). Disgnosability of discrete-

event systems. *IEEE Transactions on Automatic Control*, 40(9), 1555–1575.

Travé-Massuyès, L. and Calderon-Espinoza, G. (2007). Timed fault diagnosis. In *European Control Conference (ECC-07)*.

Travé-Massuyès, L., Escobet, T., Pons, R., and Tornil, S. (2001). The ca˜en diagnosis system and its automatic modelling method. *Computacion i Sistemas Journal*, 5(2), 128–143.

Travé-Massuyès, L. and Pons, R. (1997). Causal ordering for multiple mode systems. In *11th International Workshop on Qualitative Reasoning*. Cortona, Italy.

Weld, D. and De Kleer, J. (1989). *Readings in qualitative reasoning about physical systems*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA.