Diagnosis of Intermittent Faults with Conditional Preferences

Cédric Pralet and **Xavier Pucel** and **Stéphanie Roussel**
Systems Control and Flight Dynamics Department,
Onera – Centre de Toulouse, FR-31000 Toulouse, France
firstname.lastname@onera.fr

Abstract

Diagnosis of intermittent faults is significantly different from diagnosis of permanent faults, especially when selecting one or several preferred diagnoses. In this paper, we describe a new modeling approach for intermittent faults based on Past Time Linear Temporal Logic (PTLTL), and we suggest a conditional diagnosis selection approach based on Conditional Preference theories. We describe how intermittent faults can create diagnosis instability and show how our approach can be used to mitigate it. We then describe an incremental way to compute diagnoses at each time step, and a reduction of the incremental diagnosis computation to a MaxSAT query. Finally we discuss some limitations of our diagnoser and how they can be overcome.

1 Introduction

Autonomous systems are faced with the task of automatically handling unexpected situations, in particular faults. A common way to do so is, upon detection of a disruption or anomaly, to switch to a "safe mode", and wait for a human operator to take over and perform the system's diagnosis. Another approach is to use on-line fault diagnosis and reconfiguration, which can significantly improve the reliability and the availability of autonomous systems. However, such an approach poses technical challenges for discrete-event systems with discrete dynamics, especially when faults with intermittent or unknown dynamics must be accounted for. Moreover, we aim at coupling our online fault diagnoser with a reconfiguration and decision tool supported by a deterministic planner [Ghallab *et al.*, 2004], that requires a unique assessment of the system's health status as input instead of the set of all possible diagnoses.

The main contribution of this paper is the description of an approach for modeling discrete event systems with intermittent faults, and for computing a unique preferred diagnosis. We show how diagnosis selection is considerably more difficult for intermittent faults than it is for permanent faults. In particular, a phenomenon known as diagnosis instability can arise, which is problematic when diagnosis is used to trigger reconfiguration actions.

Our diagnosis approach is based on a behavioral model of the autonomous system and on a description of diagnosis conditional preferences. We provide a formal definition of an incremental diagnosis computation problem, and describe an original compilation technique for reducing this problem to a MaxSAT query. Finally, we illustrate how our diagnoser can fail to compute the preferred diagnosis, and discuss possible ways to mitigate this problem.

2 Diagnosis Model

We first introduce a few notations.

- Let $\mathtt{X}$ be a set of propositional variables, then an assignment a to $\mathtt{X}$ is a function from $\mathtt{X}$ to $\mathbb{B} = \{true, false\}$.
- If $\mathtt{x} \in \mathtt{X}$, then x (resp. $\overline{x}$) denotes the specific assignment to $\{\mathtt{x}\}$ that assigns value *true* (resp. *false*) to $\mathtt{x}$.
- If a and b are two assignments to two disjoint sets $\mathtt{A}$ and $\mathtt{B}$ then ab is the assignment to $\mathtt{A} \cup \mathtt{B}$ such that $\forall \mathtt{x} \in \mathtt{A}$, $ab(\mathtt{x}) = a(\mathtt{x})$ and $\forall \mathtt{x} \in B$, $ab(\mathtt{x}) = b(\mathtt{x})$.
- If a is an assignment to $\mathtt{X}$ and $\mathtt{A} \subseteq \mathtt{X}$, then $a_{|\mathtt{A}}$ is the restriction of a to $\mathtt{A}$.
- The satisfaction relation $\vDash$ between assignments and propositional formulas is defined by:
  $a \vDash \top$, $a \nvDash \bot$, $a \vDash \mathtt{x}$ *iff* $a(\mathtt{x}) = true$, $a \vDash \neg \mathtt{F}$ *iff* $a \nvDash \mathtt{F}$,
  $a \vDash \mathtt{F}_1 \, op \, \mathtt{F}_2$ *iff* $(a \vDash \mathtt{F}_1)$ *op* $(a \vDash \mathtt{F}_2)$ *with* $op \in \{\wedge, \vee, \ldots\}$.
- For a sequence of assignments $a = (a_0, \ldots, a_k)$, $a[i] = (a_0, \ldots, a_i), i \in [0\,..\,k]$ is a's prefix until index i.

The diagnosis model is a tuple (s_0, Δ, Γ), where s_0 is the system's initial state, Δ is the behavioral model of the system, and Γ is the diagnosis preference model. Δ describes the possible behaviors of the system as a finite state machine, and provides *evidence* for the diagnosis process. The purpose of Γ is to describe what is likely, or preferable, and provides *hints* for the diagnosis reasoning. Both parts share a common set of boolean variables $\mathtt{V} = \{\mathtt{v}_1, \ldots, \mathtt{v}_n\}$ associated with a partial function $\mathtt{pre} : \mathtt{V} \to \mathtt{V}$. The variables represent the system's state, the events (e.g. fault events) impacting this state, and the inputs and outputs of the diagnoser, at the current instant and to some extent at past instants. Events are modeled by a dedicated boolean variable in $\mathtt{V}$, true at the instant the event occurs and false otherwise. For any two variables $\mathtt{v}$ and $\mathtt{v}'$, $\mathtt{v}' = \mathtt{pre}(\mathtt{v})$ means that $\mathtt{v}'$ holds $\mathtt{v}$'s value at the

previous time step. When possible, we refer to $\mathtt{v}'$ simply as $\mathtt{pre}(\mathtt{v})$. Naturally, $\mathtt{pre}$ must form no cycle, i.e. $\mathtt{pre}^n(\mathtt{v}) \neq \mathtt{v}$ for all $\mathtt{v}$ and $n > 0$. In the following, $\mathtt{V}_{\mathtt{hasPre}}$ denotes the domain of $\mathtt{pre}$ and $\mathtt{V}_{\mathtt{pre}}$ its image, i.e. the set of variables associated with past instants, and $\mathtt{V}_{\mathtt{now}} = \mathtt{V} - \mathtt{V}_{\mathtt{pre}}$. We assume a synchronous model where time steps all last the same predefined duration. We denote by $\mathtt{O} \subseteq \mathtt{V}_{\mathtt{now}}$ the set of observable variables, and $\mathtt{D} \subseteq \mathtt{V}_{\mathtt{now}}$ the set of diagnosis variables. Without loss of generality we assume that $\mathtt{D} \cap \mathtt{O} = \emptyset$. A *system state* is an assignment to $\mathtt{V}$, and we denote by S the set of all states. An observation is an assignment to the variables of $\mathtt{O}$.

The first part of the model, Δ, consists in a set of propositional formulas over $\mathtt{V}$ that specifies the system's behavior as a finite state machine. The transition relation $\xrightarrow{\Delta} \subseteq S \times S$ is encoded by Δ as follows: a state s' is a consistent successor to s if it satisfies all the formulas of Δ, and the values of $\mathtt{V}_{\mathtt{pre}}$ variables in s' hold the same value as their respective current time variables in s. Formally:

Definition 1 (Reachable state)**.** *A state s' can be reached from a state s, denoted by $s \xrightarrow{\Delta} s'$ if and only if $\forall \delta \in \Delta, s' \vDash \delta$ and $\forall \mathtt{v} \in \mathtt{V}_{\mathtt{hasPre}}, s(\mathtt{v}) = s'(\mathtt{pre}(\mathtt{v}))$.*

The system's initial state s_0 satisfies the formulas of Δ. We now define how a sequence of observations leads to a set of diagnoses.

Definition 2 (Explanation)**.** *Let $obs = (o_0, \ldots, o_k)$ be a sequence of observations. An explanation for obs is a state sequence $(s_0, \ldots, s_k)$ such that $\forall i \in [0\,..\,k-1]$ $s_i \xrightarrow{\Delta} s_{i+1}$ and $\forall i \in [0\,..\,k]$ $s_{i|\mathtt{O}} = o_i$.*

Definition 3 (Diagnosis)**.** *Let $obs = (o_0, \ldots, o_k)$ be a sequence of observations. An assignment d to $\mathtt{D}$ is a* diagnosis *for obs if and only if there exists an explanation $(s_0, \ldots, s_k)$ for obs such that $s_{k|\mathtt{D}} = d$.*

Example 1. *We consider a switch mechanism between two redundant actuators $A1$ and $A2$, powered by a common power supply, as represented in Figure 1. Our system is modeled by the following variables:*

$\mathtt{h_{pow}}$ *: power supply health status (true when nominal)*
$\mathtt{h_{sw}}$ *: switch health status (true when nominal)*
$\mathtt{h_1}$ *: actuator $A1$ health status (true when nominal)*
$\mathtt{h_2}$ *: actuator $A2$ health status (true when nominal)*
$\mathtt{f_{sw}}$ *: switch permanent fault event*
$\mathtt{f_2}$ *: actuator $A2$ temporary fault event*
$\mathtt{sw_{in}}$ *: switch command (true to activate actuator $A1$)*
$\mathtt{sw_{pos}}$ *: switch position (true when actuator $A1$ is active)*
$\mathtt{a_1}$ *: actuator $A1$ active (true when operating)*
$\mathtt{a_2}$ *: actuator $A2$ active (true when operating)*

The above variables correspond to set $\mathtt{V}_{\mathtt{now}}$. The set of diagnosis variables is $\mathtt{D} = \{\mathtt{h_{pow}}, \mathtt{h_{sw}}, \mathtt{h_1}, \mathtt{h_2}\}$, and the set of observable variables is $\mathtt{O} = \{\mathtt{sw_{in}}, \mathtt{a_1}, \mathtt{a_2}\}$.

The transition relation is defined as[1]*:*

$$\Delta_1 = \left\{ \begin{array}{c} \mathtt{a_1} \leftrightarrow (\mathtt{h_{pow}} \wedge \mathtt{sw_{pos}} \wedge \mathtt{h_1}), \\ \mathtt{a_2} \leftrightarrow (\mathtt{h_{pow}} \wedge \neg\mathtt{sw_{pos}} \wedge \mathtt{h_2}) \\ \mathtt{h_{sw}} \leftrightarrow (\mathtt{pre}(\mathtt{h_{sw}}) \wedge \neg\mathtt{f_{sw}}), \\ \mathtt{h_2} \leftrightarrow (\neg\mathtt{f_2} \wedge \neg\mathtt{pre}(\mathtt{f_2}) \wedge \neg\mathtt{pre}(\mathtt{pre}(\mathtt{f_2}))), \\ \mathtt{sw_{pos}} \leftrightarrow \mathtt{ite}(\mathtt{h_{sw}}, \mathtt{sw_{in}}, \mathtt{pre}(\mathtt{sw_{pos}})), \end{array} \right\} \begin{array}{l} (1) \\ (2) \\ (3) \\ (4) \\ (5) \end{array}$$

[1] $\mathtt{ite}$ is the boolean if-then-else operator defined by $\mathtt{ite}(\mathtt{a}, \mathtt{b}, \mathtt{c}) \leftrightarrow \big((\mathtt{a} \wedge \mathtt{b}) \vee (\neg\mathtt{a} \wedge \mathtt{c})\big)$.

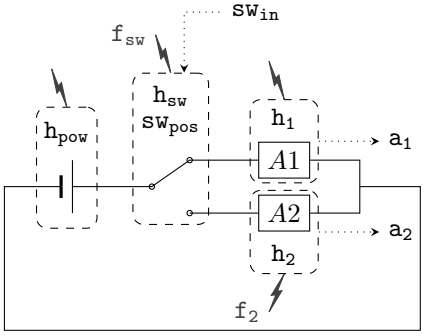

Figure 1: A system with two redundant actuators. The switch is subject to a permanent fault. The first actuator and the power supply are subject to random intermittent faults. The second actuator will temporarily be off during 2 time steps following a fault.

Each actuator is active if and only if the power supply is healthy, the switch is in the appropriate position, and it is healthy (1)(2)*. The switch is healthy if and only if it was healthy at previous time step and if there is no permanent fault event* (3)*. The second actuator is healthy is there has not been any fault in the last three time steps* (4)*. Finally, if the switch is healthy, then its position is the one from the command, else its position is the one it was stuck in* (5)*.*

We know the switch is subject to a permanent fault, so there is a relation between $\mathtt{h_{sw}}$, its past and $\mathtt{f_{sw}}$. Actuator 2 goes offline during 3 timesteps after a fault occurrence, which means $\mathtt{h_2}$ is influenced by $\mathtt{f_2}$'s recent values. However we have no model for the faults in the power supply and actuator $A1$, so $\mathtt{h_1}$ and $\mathtt{h_{pow}}$ are independent from past variables, so as to represent random intermittent failures. We could introduce fault events $\mathtt{f_1}$, $\mathtt{f_{pow}}$, and even "repair events" $\mathtt{r_1}$, $\mathtt{r_{pow}}$, but this would not help modeling the system's behaviour. Moreover, in this example we are interested in the current health statuses of the components more than their history, so such variables have no use in the model.

With such a behavioral model of the system, $\mathtt{V}_{\mathtt{pre}}$ is the set $\{\mathtt{pre}(\mathtt{h_{sw}}), \mathtt{pre}(\mathtt{f_2}), \mathtt{pre}(\mathtt{pre}(\mathtt{f_2})), \mathtt{pre}(\mathtt{sw_{pos}})\}$. $\mathtt{V} = \mathtt{V}_{\mathtt{now}} \cup \mathtt{V}_{\mathtt{pre}}$ contains 14 variables. Finally, we assume that the initial state is:

$$s_0 = \left[\begin{array}{l} h_{pow}\, h_{sw}\, h_1\, h_2\, \overline{f_{sw}}\, \overline{f_2} \\ sw_{in}\, sw_{pos}\, a_1\, \overline{a_2} \\ pre(h_{sw})\, \overline{pre(f_2)}\, \overline{pre(pre(f_2))}\, pre(sw_{pos}). \end{array} \right.$$

Let us consider the observation sequence $obs_1 = (sw_{in}\, a_1\, \overline{a_2}, sw_{in}\, \overline{a_1\, a_2})$. The first observation is simply $s_{0|O}$. At time step 1 the possible diagnoses are $\overline{h_{pow}} h_1$, $h_{pow}\, \overline{h_1}$ and $\overline{h_{pow}\, h_1}$, with any value for $\mathtt{h_{sw}}$ and $\mathtt{h_2}$.

The second part of the model, Γ, consists in a Conditional Preference theory acting on the variables of $\mathtt{V}$. It defines a partial ordering on S, that, by lexicographical extension, lets one compare state sequences. It supports the selection of one explanation $(s_0, \ldots, s_k)$ for any observation $(o_0, \ldots, o_k)$.

We describe thereafter the specification languages for both parts of the model. The specification language for Δ, described in Section 2.1, is able to model intermittent faults, while the language for Γ, described in

Section 2.2 is suitable for defining conditional diagnosis preferences. In Section 2.3 we describe the problem of diagnosis stability that naturally arises when monitoring intermittent faults, and demonstrate how our approach can address this problem.

2.1 Intermittent faults

Diagnosis of intermittent faults has been addressed in [De Kleer, 2009] by using multiple system executions to produce a single, constant diagnosis. For embedded systems, diagnosis should be produced online and vary across time. In [Contant *et al.*, 2004], intermittent faults are modelled by fault events and associated reset events. The fault is considered present after its fault event and before its reset event, and absent otherwise. As illustrated in Example 1, as we are not concerned by the occurrence of fault events, but by the current health status of the components.

An important aspect of intermittent fault modeling is the distinction between the occurrence of a fault, the presence of its effects, and the likeliness of it happening again. Which of these properties should be monitored by diagnosis is application dependent, this is why we only speak of a set $\mathtt{D}$ of diagnosed variables, and we do not make any assumption on how they are related to the system's behavior.

Defining faults as behavioral patterns has been developed in [Jéron *et al.*, 2006] and more particularly in [Jiang and Kumar, 2004] using LTL specifications. These approaches are expressive enough for our needs. We adopt a very similar approach using Past Time Linear Temporal Logic (PTLTL). The main difference between LTL and PTLTL is that LTL formulas are evaluated from the initial state in the system's history, while PTLTL formulas are evaluated from the current (last) state, and can be evaluated incrementally in a more efficient way than LTL [Emerson, 1990].

PTLTL formulas and their semantics extend those of propositional calculus as recalled below, where $t = (s_0, \ldots, s_k)$ denotes a state sequence, $\mathtt{F}$ a PTLTL formula, and $\mathtt{v_i}$ a variable of $\mathtt{V}$. The Y, O, $\mathcal{S}$ and O_n operators stand respectively for "yesterday", "once", "since" and "once in last n".

$t \vDash \top$
$t \vDash \neg\mathtt{F}$ *iff* $t \nvDash \mathtt{F}$
$t \vDash v_i$ *iff* $s_k(\mathtt{v_i}) = true$, *i.e. iff* $\mathtt{v_i}$ *is true in* t*'s last state*
$t \vDash \mathtt{F}_1\, op\, \mathtt{F}_2$ *iff* $(t \vDash \mathtt{F}_1)\ op\ (t \vDash \mathtt{F}_2)$ *with* $op \in \{\wedge, \vee, \rightarrow, \ldots\}$
$t \vDash Y(\mathtt{F})$ *iff* $k = 0$ *or* $t[k-1] \vDash \mathtt{F}$
$t \vDash O(\mathtt{F})$ *iff* $\exists i \in [0\,..\,k], t[i] \vDash \mathtt{F}$
$t \vDash O_n(\mathtt{F})$ *iff* $\exists i \in [k-n+1\,..\,k], t[i] \vDash \mathtt{F}$
$t \vDash \mathtt{F}_1\, \mathcal{S}\, \mathtt{F}_2$ *iff* $\exists i \in [0\,..\,k], t[i] \vDash \mathtt{F}_2 \wedge \forall j \in [i{+}1\,..\,k], t[j] \vDash \mathtt{F}_1$

[Havelund and Rosu, 2002] proves that any PTLTL formula defined over set of variables $\mathtt{V_{now}}$ can be equivalently compiled into a set of formulas over $\mathtt{V_{now}} \cup \mathtt{V_{pre}}$. This allows us to use PTLTL as a specification language for modeling both the system's dynamics and the properties that need to be monitored.

Example 2. *In our system, instead of Equation (3), the effect of permanent fault event* $\mathtt{f_{sw}}$ *can be modeled directly as* $\mathtt{h_{sw}} \leftrightarrow \neg O(\mathtt{f_{sw}})$ *(the switch is healthy if and only if no fault occurred at the present time or in the past). Moreover, instead of Equation (4), the temporary effect of fault event* $\mathtt{f_2}$ *can be modeled by* $\mathtt{h_2} \leftrightarrow \neg O_3(\mathtt{f_2})$ *(actuator A2 is healthy if and only if no fault event occurred during the last three time steps). As a result, in PTLTL, the transition relation of our system can be written:*

$$\Delta_2 = \left\{ \begin{array}{c} \mathtt{a_1} \leftrightarrow (\mathtt{h_{pow}} \wedge \mathtt{sw_{pos}} \wedge \mathtt{h_1}), \\ \mathtt{a_2} \leftrightarrow (\mathtt{h_{pow}} \wedge \neg\mathtt{sw_{pos}} \wedge \mathtt{h_2}), \\ \mathtt{h_{sw}} \leftrightarrow \neg O(\mathtt{f_{sw}}), \\ \mathtt{h_2} \leftrightarrow \neg O_3(\mathtt{f_2}), \\ \mathtt{sw_{pos}} \leftrightarrow \mathtt{ite}(\mathtt{h_{sw}}, \mathtt{sw_{in}}, Y(\mathtt{sw_{pos}})) \end{array} \right\}$$

In comparison to Δ_1*, all variables of* $\mathtt{V_{pre}}$ *have been replaced by instances of the PTLTL operators* Y*,* O *and* O_3 *applied to variables of* $\mathtt{V_{now}}$*. This PTLTL sentence can be compiled into a propositional formula roughly equivalent to that of Example 1. Note that in* Δ_2*, random intermittent faults such as* $\mathtt{h_{pow}}$ *and* $\mathtt{h_1}$ *are still not associated with any fault event, and are still not constrained by their past values.*

PTLTL can also model reset events: in a system where a fault event $\mathtt{f}$ over a component were associated with a reset event $\mathtt{r}$, the health status $\mathtt{h}$ of the component can be modeled by $\neg\mathtt{h} \leftrightarrow \neg\mathtt{r}\,\mathcal{S}\mathtt{f}$ (the system is not healthy if there has been no reset since the last fault event).

2.2 Conditional diagnosis preferences

Selecting preferred diagnoses is usually addressed by specifying an ordering on the possible diagnoses. For example diagnoses can be partially ordered by cardinality [Reiter, 1987]. Diagnoses of the same cardinality can be distinguished by totally ordering faults, and defining a lexicographical order on diagnoses based on the fault ordering [Felfernig and Schubert, 2010]: if two faults $\mathtt{f_1}$ and $\mathtt{f_2}$ are ordered $\mathtt{f_1} < \mathtt{f_2}$, then the lexicographical diagnosis ordering is $(\overline{f_1}\overline{f_2}) < (\overline{f_1}f_2) < (f_1\overline{f_2}) < (f_1 f_2)$. Assuming that diagnoses are ordered, [Grastien *et al.*, 2009] extends the notion of preferred diagnosis to that of preferred sequence of states in a natural way.

Another way to select diagnoses is to use exoneration assumptions to eliminate diagnoses that involve false negative tests [Cordier *et al.*, 2004]. Yet another way is to explicitly account for diagnosis probability [Ricks and Mengshoel, 2010; Abreu and Cardoso, 2013; Zabi *et al.*, 2013].

Work based on an unconditional diagnosis ordering is not satisfactory from our point of view because the diagnosis ordering does not take the current observations into account. The nominal mode is usually unconditionally preferred, and can only be eliminated by *evidence* that the system is faulty: even if we have strong suspicions about a fault, as long as the nominal mode is possible, it is selected as the preferred diagnosis. Our ambition is to use diagnosis preferences to express *hints*, i.e. to make the diagnosis ordering situational. There are many reasons for one to prefer a different diagnosis in some particular situation, including diagnosis likelihood, but also external factors such as the impact of modeling errors in the diagnosis model, the cost and safety of associated reconfigurations, the robustness of control algorithms with respect to some faults, or the presence of other diagnosers that detect some faults better than this one. All these aspects are not part of a behavioral model, and none of the aforementioned

approaches can handle them. The purpose of the second part of our model Γ is to provide the tools for the designers to specify their own conditional diagnosis ordering.

Example 3. *In our example, we do not know the exact dynamics behind* $\mathtt{h_{pow}}$ *and* $\mathtt{h_1}$*. However, we know that it is very unlikely that both actuators fail during a single autonomous run. Thus, when one actuator fails, we suspect a fault in this actuator, but once both actuators have shown failure symptoms, we suspect the power source. And as usual, we consider a component to be nominal when we do not use it. This means that for the observation sequence of Example 1* $obs_1 = (sw_{in}\, a_1\, \overline{a_2}, sw_{in}\, \overline{a_1\, a_2})$*, the preferred diagnosis is* $d_1 = h_{pow} h_{sw} \overline{h_1} h_2$*. However, if it was preceded by an sequence where actuator A2 has failed such as* $obs_2 = (sw_{in}\, a_1\, \overline{a_2}, \overline{sw_{in}\, a_1}\, a_2, \overline{sw_{in}\, a_1\, a_2}, sw_{in}\, a_1\, \overline{a_2}, sw_{in}\, \overline{a_1\, a_2})$ *then at time step 4 the preferred diagnosis would be* $d_2 = \overline{h_{pow}} h_{sw} h_1 h_2$*.*

An cardinality-based diagnosis ordering cannot implement such a diagnosis strategy, as the diagnoses d_1 *and* d_2 *have the same cardinality. Moreover, a lexicographic ordering is unsuitable as well: if we use a variable ordering such that* $\mathtt{h_1} < \mathtt{h_{pow}}$*, then* d_2 *can never be preferred. Conversely if we have* $\mathtt{h_{pow}} < \mathtt{h_1}$*, then* d_1 *will be discarded regardless of the previous observations.*

Our approach to conditional diagnosis is to *express preferences* not just about the diagnoses, but *about the explanations behind the diagnoses.* We do so by defining a partial order on the set of all states S, and extend it lexicographically to the set of all explanations S^*.

Specifying an ordering on S and maintaining this specification can be complex and error prone, but specific languages known as Conditional Preference theories (CP-theories [Wilson, 2011]), that generalize CP-nets [Boutilier *et al.*, 2004], provide a compact and intuitive representation of conditional preferences over variable assignments. We define a *diagnosis CP-theory*[2] as a set Γ of *diagnosis conditional preferences.*

Definition 4. *A* conditional preference *on a variable* $\mathtt{v_i} \in \mathtt{V}$ *is a statement of the form* $\Phi : v_i \prec \overline{v_i}$ *where* Φ *is a propositional formula over a set of variables* $\mathtt{U_{v_i}} \subseteq \mathtt{V}$*.*

$\Phi : v_i \prec \overline{v_i}$ is interpreted as follows: let t be an assignment to $\mathtt{V} - (\mathtt{U_{v_i}} \cup \{\mathtt{v_i}\})$ and u an assignment to $\mathtt{U_{v_i}}$, then assignment tuv_i is preferred to assignment $tu\overline{v_i}$ if and only if $u \vDash \Phi$. Informally, it means that v_i is preferred to $\overline{v_i}$ if and only if Φ holds, other variables (those in t) being fixed.

The graph associated with the set Γ of diagnosis preferences uses the variables of $\mathtt{V}$ as nodes, and for each preference $\Phi : v_i \prec \overline{v_i}$ in Γ, there is a directed arc from each variable in the scope of Φ to $\mathtt{v_i}$. Additionally, when a variable $\mathtt{v}$ is parent of $\mathtt{v'}$ in the graph, it means that the preference over $\mathtt{v}$ preempts the preference over $\mathtt{v'}$.

Definition 5. *A* diagnosis CP-theory *is a set of conditional preferences whose associated graph is acyclic and with at most one preference per variable in* $\mathtt{V}$*. For a diagnosis CP-theory* Γ*,* $\mathtt{P_\Gamma}$ *denotes the set of variables subject to conditional preferences, also called the set of variables targeted by* Γ*.*

A state s satisfies a preference $\Phi : v_i \prec \overline{v_i}$ if and only if $s \vDash \Phi \leftrightarrow \mathtt{v_i}$. Note that $\Phi : v_i \prec \overline{v_i}$ is strictly equivalent to $\neg\Phi : \overline{v_i} \prec v_i$. Following [Wilson, 2011], any diagnosis CP-theory Γ is consistent, i.e. it defines a partial order $\prec_\Gamma$ on S. This order is defined as follows: $s \prec_\Gamma s'$ if and only if there is a preference γ that is satisfied by s and violated by s', and for all preferences γ' that preempt γ, s and s' both satisfy or violate γ'. This partial order matches the topological order induced by the associated graph.

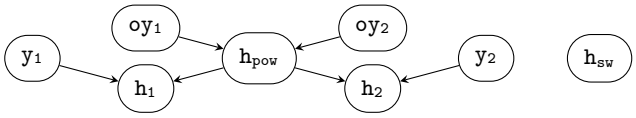

Figure 2: Graph induced by the preference set Γ_1 of Example 4. $\mathtt{oy_1}$ and $\mathtt{oy_2}$ are variables generated by the PTLTL compilation to represent respectively the PTLTL formulas $O(y_1)$ and $O(y_2)$.

Example 4. *A possible way to implement the diagnosis strategy detailed in Example 3 is to introduce two new variables* $\mathtt{y_1}$ *and* $\mathtt{y_2}$ *(*$\mathtt{y}$ *stands for symptom) that are true when an actuator is expected to work and is not working, to enrich* Δ_2 *in consequence, and to use the set of diagnosis preferences* Γ_1 *described below.*

$$\Delta_3 = \Delta_2 \cup \left\{ \begin{array}{l} \mathtt{y_1} \leftrightarrow (\mathtt{sw_{pos}} \wedge \neg \mathtt{a_1}), \\ \mathtt{y_2} \leftrightarrow (\neg \mathtt{sw_{pos}} \wedge \neg \mathtt{a_2}) \end{array} \right\}$$

$$\Gamma_1 = \left\{ \begin{array}{rl} O(\mathtt{y_1}) \wedge O(\mathtt{y_2}) : \overline{h_{pow}} \prec h_{pow}, & (1) \\ \mathtt{y_1} \wedge \mathtt{h_{pow}} : \overline{h_1} \prec h_1, & (2) \\ \mathtt{y_2} \wedge \mathtt{h_{pow}} : \overline{h_2} \prec h_2, & (3) \\ \top : h_{sw} \prec \overline{h_{sw}} & (4) \end{array} \right\}$$

Informally, preference (1) *means that once symptom* $\mathtt{y_1}$ *and* $\mathtt{y_2}$ *have each been present, we prefer to blame the power supply. Preferences* (2) *and* (3) *mean that when symptom* $\mathtt{y_i}$ *is present and the power supply is nominal, we blame actuator Ai. Preference* (4) *means that we always expect the switch to be nominal. The variable graph induced by* Γ_1 *is represented in Figure 2, and indicates that preference* (1) *preempts preferences* (2)-(3) *and must be evaluated prior to them. Preference* (4) *can be evaluated any time.*

For observation sequence obs_1 *(Example 1), we have by definition* $s_0 \nvDash O(\mathtt{y_1})$ *and* $s_0 \nvDash O(\mathtt{y_2})$*. At time step 1, since* $\mathtt{pre}(\mathtt{sw_{pos}})$ *takes value true initially, the observation* $sw_{in}\overline{a_1\, a_2}$ *entails* $y_1 \overline{y_2}$*. We first consider preference* (1)*: its condition is false, thus we select* h_{pow}*. Then, we consider preference* (2)*: its condition is true, and we select* $\overline{h_1}$*. Preference* (3) *leads to* h_2*, and preference* (4) *to* h_{sw}*. The preferred diagnosis is* $h_{pow} \overline{h_1} h_2 h_{sw}$*. We leave it up to the reader to check that* obs_2 *(Example 3) leads to the preferred diagnosis* $\overline{h_{pow}} h_1 h_2 h_{sw}$*.*

Note that once $O(\mathtt{y_1}) \wedge O(\mathtt{y_2})$ is true, we always prefer $\overline{h_{pow}}$ to h_{pow}, even when the symptoms $\mathtt{y_1}$ and $\mathtt{y_2}$ are absent at the current time step. This does not mean

[2] We adapt [Wilson, 2011]'s definition to use boolean variables, reuse the model's variable sets, and force empty sets of irrespective variables.

that $\overline{h_{pow}}$ must hold once the symptoms have been observed. In fact, Δ_3 excludes the assignments containing $\overline{h_{pow}}\,\overline{y_1}\,\overline{y_2}$, so $\overline{h_{pow}}$ can only be part of a diagnosis when a symptom is present. This illustrates that Γ is only used to order the diagnoses accepted by Δ, but only Δ defines what is and what is not a diagnosis. In Example 4, the condition for preference (1) could be written $O(\mathtt{y_1}) \wedge O(\mathtt{y_2}) \wedge (\mathtt{y_1} \vee \mathtt{y_2})$ with equivalent results. Δ provides diagnosis with evidence, while Γ provides diagnosis with clues. Together, they implement the diagnosis reasoning described in Example 3.

We now define how $\prec_\Gamma$ can be used as a lexicographical base to order all the explanations in S^*, and define the preferred diagnosis for an observation sequence.

Definition 6. *Let $t_{0..k} = (t_0, \ldots, t_k)$ and $u_{0..k} = (u_0, \ldots, u_k)$ be two state sequences such that $t_0 = u_0 = s_0$, and let Γ be a diagnosis CP-theory entailing a partial order $\prec_\Gamma$ on S. We define the partial ordering $\prec_{S^*}$ on S^* as the lexicographical order induced by $\prec_\Gamma$, i.e. $t_{0..k} \prec_{S^*} u_{0..k}$ if and only if there exists $i \in [0\,..\,k]$ such that $t_j = u_j$ for $j \in [0\,..\,i[$, and either $i = k$ or $t_i \prec_\Gamma u_i$.*

Definition 7. *Let $obs = (o_0, \ldots, o_k)$ be a sequence of observations, and let $E \subseteq S^k$ be the set of all explanations for obs. An explanation $e \in E$ is a* preferred explanation *if and only if there is no explanation $e' \in E$ such that $e' \prec_{S^*} e$. A diagnosis d_k is a* preferred diagnosis *for obs at time step k if and only if there exists a preferred explanation $e = (s_0 \ldots s_k)$ such that $s_{k|\mathtt{D}} = d_k$.*

Let us emphasize that in the previous definition, a preferred explanation must satisfy the constraints described in Δ. In terms of CP-theories, this means that we manipulate kinds of constrained Conditional Preference networks [Boutilier *et al.*, 2004; Prestwich *et al.*, 2004]. Note also that if $\prec_\Gamma$ is a total order, it makes $\prec_{S^*}$ a total order as well. This guarantees that the preferred explanation and diagnosis are unique at each time step and for any observation.

Note that the notion of preferred explanation based on CP-theories generalizes the ordering by cardinality strategy of [Reiter, 1987]. In fact, for a set of faults $\{f_1, \ldots f_n\}$, the preferences $\top : \overline{f_i} \prec f_i$ guarantee that the set of diagnosed faults is minimal for set inclusion. The total ordering of faults in [Felfernig and Schubert, 2010] can be reproduced by adding the same preferences $\top : \overline{f_i} \prec f_i$ and explicitly extending the partial order $\prec_\Gamma$ to a total order $<_\Gamma$ that represents the fault priority. Rejecting false negatives ([Cordier *et al.*, 2004]) can be implemented with one preference per test.

Preferences can also express optimism or pessimism about the presence of faults in the diagnosis. More precisely, for a fault $\mathtt{f_i}$, the preference $\top : \overline{f_i} \prec f_i$ encodes that without any proof for f_i, $\overline{f_i}$ will be preferred, which can represent some form of optimism with respect to $\mathtt{f_i}$. This kind of behavior can be used for non-critical faults. On the opposite, the preference $\top : f_i \prec \overline{f_i}$ expresses that without proof for $\overline{f_i}$, f_i will be diagnosed. This indicates some pessimism with respect to $\mathtt{f_i}$ and can be used for more critical faults.

2.3 Diagnosis stability

Diagnosis of intermittent faults raises the problem of diagnosis stability that, to our knowledge, has not been treated as such in the literature. When diagnosis is used to trigger reconfiguration actions, it is important to avoid spurious diagnoses. In the case of permanent faults, the strategy that consists in selecting minimal diagnoses is guaranteed to be stable, as faults are only added when we have evidence of their occurrence, and the diagnosis can only grow over time. However, in the case of intermittent faults, a naive diagnosis selection approach, such as the one presented in Example 4, can lead to scenarios in which the preferred diagnosis alternates between two or more values, while the system's health status is actually constant, as illustrated in Figure 3.

	s_0	s_1	s_2	s_3	s_4	s_5
$\mathtt{h_{pow}}$	1	1	1	1	1	1
$\mathtt{h_{sw}}$	1	1	1	1	1	1
$\mathtt{h_1}$	1	0	0	0	0	0
$\mathtt{h_2}$	1	1	1	1	1	1
$\mathtt{sw_{in}}$	1	0	1	0	1	0
$\mathtt{sw_{pos}}$	1	0	1	0	1	0
$\mathtt{a_1}$	1	0	0	0	0	0
$\mathtt{a_2}$	0	1	0	1	0	1
pref. diag.	$h_{pow}h_1$ $h_{sw}h_2$	$h_{pow}h_1$ $h_{sw}h_2$	$h_{pow}\overline{h_1}$ $h_{sw}h_2$	$h_{pow}h_1$ $h_{sw}h_2$	$h_{pow}\overline{h_1}$ $h_{sw}h_2$	$h_{pow}h_1$ $h_{sw}h_2$

Figure 3: An execution scenario based on Example 4's model, in which actuator $A1$ becomes faulty at time step 1 and stays so. It displays *unstable diagnosis*: even though the system's actual health status is constant after time step 1, the diagnosis perpetually alternates between two values.

A formal definition of diagnosis instability is beyond the scope of this paper. Still, it is possible to use CP-theory to induce stability mechanisms even with intermittent faults, by using the diagnosed variables' past values. For example, let $\mathtt{v_i}$ model an intermittent fault, the preference $Y(\mathtt{v_i}) : v_i \prec \overline{v_i}$ states that we always prefer the previous value for $\mathtt{v_i}$. Then, its diagnosis value will not change unless we have evidence of it. Many other stability strategies are possible according to the system's behavior, fault criticality, and available reconfiguration actions.

Example 5. *The diagnosis preference set Γ_1 of Example 4 can be stabilized as follows to avoid the situation described in Figure 3:*

$$\Gamma_2 = \left\{ \begin{array}{rl} O(\mathtt{y_1}) \wedge O(\mathtt{y_2}) : & \overline{h_{pow}} \prec h_{pow}, \\ \neg Y(\mathtt{h_1}) \vee (\mathtt{y_1} \wedge \mathtt{h_{pow}}) : & \overline{h_1} \prec h_1, \\ \neg Y(\mathtt{h_2}) \vee (\mathtt{y_2} \wedge \mathtt{h_{pow}}) : & \overline{h_2} \prec h_2, \\ \top : & h_{sw} \prec \overline{h_{sw}} \end{array} \right\}$$

The diagnosis value for $\mathtt{h_1}$, once set to $\overline{h_1}$, stays so as long as it is consistent with h_{pow} and Δ, as we prefer to keep its past value. The same goes for $\mathtt{h_2}$. Such a mechanism is unnecessary for $\mathtt{h_{sw}}$ and $\mathtt{h_{pow}}$, because $\mathtt{f_{sw}}$ is a permanent fault, and because $\mathtt{h_{pow}}$'s effects cannot be masked, so its diagnosis cannot be unstable. Note that $\neg Y(\mathtt{h_1})$ is false at the first time step, while $Y(\neg \mathtt{h_1})$ would be true.

3 Diagnoser synthesis

We now describe a way to compute the preferred diagnosis incrementally at each time step. On-line in-

cremental diagnosis has been addressed under three different forms: classical diagnoser approaches, knowledge representation approaches, and sliding windows. Classical diagnoser approaches [Sampath *et al.*, 1995; Contant *et al.*, 2004] consist in precompiling the diagnoser as an explicit finite state automaton. These approaches suffer from well known scalability problems as the diagnoser states grow exponentially with the system faults. Approaches based on knowledge representation techniques such as OBDDs [Torta and Torasso, 2007; Darwiche and Marquis, 2002] offer complete freedom in their definition of faults, and support the encoding of conditional preferences as hard constraints, but also suffer from well known scalability problems. SAT-based sliding windows approaches [Grastien *et al.*, 2009] scale well, and we adopt a similar MaxSAT approach.

3.1 Incremental evaluation

Our diagnoser is invoked at each time step, and produces a diagnosis based on the current observation and on some information computed at the previous time step, named a *transmitted assignment*. As we require a unique diagnosis, the first step in the diagnoser synthesis is to arbitrarily extend the partial order $\prec_\Gamma$ to a total order $<_\Gamma$, so that there exists only one preferred explanation at any time.

We consider a subset of variables $\mathtt{T} \subseteq \mathtt{V_{hasPre}}$ that represent variables that will be transmitted at each time step. The precise content of $\mathtt{T}$ is discussed in Section 4, as it significantly impacts the diagnoser's behaviour. For now we assume that at time step k, the diagnoser receives as input an observation o_k, an assignment t_{k-1} of $\mathtt{T}$, and computes a selected state s_k as follows:

Definition 8. *Let $obs = (o_0, \ldots, o_k)$ be a sequence of observations and let $\mathtt{T} \subseteq \mathtt{V_{hasPre}}$. The diagnoser's procedure computes s_k, d_k and t_k as follows:*

- *at time step 0, s_0 is the selected state for $obs[0]$, $d_0 = s_{0|\mathtt{D}}$ is the selected diagnosis, $t_0 = s_{0|\mathtt{T}}$ is the transmitted assignment;*
- *at time step $j \in [1..k]$, $s_j \in S$ is the selected state for obs if and only if (1) $s_{j|\mathtt{O}} = o_j$, (2) there exists α such that $\alpha_{|\mathtt{T}} = t_{j-1}$ and $\alpha \xrightarrow{\Delta} s_j$, and (3) for all $s' \in S$ satisfying conditions 1 and 2 we have $s_j <_\Gamma s'$. $d_j = s_{j|\mathtt{D}}$ is the selected diagnosis and $t_j = s_{j|\mathtt{T}}$ is the transmitted assignment.*

Note that based on the previous definition, state sequence $(s_0, \ldots, s_k)$ is not necessarily an explanation for $(o_0, \ldots, o_k)$ (in the sense of Definition 2), because not all variables are transmitted from one step to another. This is discussed in Section 4.1.

3.2 MaxSAT encoding

[Prestwich *et al.*, 2005] describes an algorithm for computing all optimal outcomes in a constrained CP-net. This algorithm handles theories with non-boolean variables and cyclic preferences but suffers from computational issues. We adopt here a MaxSAT based approach particularly suited to our acyclic boolean preferences.

Our first step is to sort the preferences of Γ with a total order $\lhd_\Gamma$ such that if preference γ_1 targets variable $\mathtt{v_1}$ and γ_2 targets $\mathtt{v_2}$, and $(\mathtt{v_1}, \mathtt{v_2})$ is an edge in Γ's graph, (i.e. γ_1 preempts γ_2), then $\gamma_1 \lhd_\Gamma \gamma_2$. We then reduce the preferred state selection to an optimization problem under constraints in which a state s is a solution if it satisfies constraints induced by observations, transmitted variables and transition formula Δ, and is optimal if it maximizes the satisfaction of preferences in the order defined by $\lhd_\Gamma$. More precisely, conditions (1) and (2) of definition 8 are constraints that a selected state s has to satisfy. Condition (3) is encoded into a objective function that takes the form of a lexicographic order.

There are several ways to address lexicographic boolean optimization [Marques-Silva *et al.*, 2011] with boolean satisfaction techniques. We present here a Weighted Partial MaxSAT approach that consists in finding an assignment that minimizes the total weights of unsatisfied *soft clauses* while satisfying *hard clauses*. The main idea is to encode Δ as hard clauses, and Γ as soft clauses associated with weights that implement a lexicographical order. More formally:

- for each variable $\mathtt{o_i}$ of $\mathtt{O}$, if $o_k(\mathtt{o_i}) = true$ then $\mathtt{o_i}$ is a hard clause, otherwise $\neg\mathtt{o_i}$ is a hard clause;
- for each variable $\mathtt{t_i}$ of $\mathtt{T}$, if $t_{k-1}(\mathtt{t_i}) = true$ then $\mathtt{pre(t_i)}$ is a hard clause, otherwise $\neg\mathtt{pre(t_i)}$ is a hard clause;
- formula Δ is transformed into hard clauses using a Tseitin's based transformation [Tseitin, 1968];
- if γ_i is the ith preference in Γ ordered by $\lhd\Gamma$, and is written $\Phi_i : v_{\gamma i} \prec \overline{v_{\gamma i}}$, we create a boolean variable ρ_i representing the satisfaction of γ_i. For each clause c obtained in the transformation of $\Phi \leftrightarrow \mathtt{v_i}$ into clauses, $c \vee \rho_i$ is added as a hard clause and $\neg\rho_i$ is a soft clause with the associated weight $w_i = 2^{\|\Gamma\|-i}$.

The model returned by a MaxSAT solver on the set of hard and soft clauses above encodes the selected state at the current time step.

4 Deadlocks

Our approach suffers from limitations similar to that of [Grastien *et al.*, 2009], namely that due to its incremental and non exhaustive implementation, the diagnoser can choose a wrong explanation in the execution path, and return non preferred diagnoses, or no diagnosis at all (deadlock).

Definition 9 (Deadlock)**.** *Let $obs = (o_0, \ldots o_k)$ be a sequence of observations. The procedure* deadlocks *at step k if and only if it can select a state for $obs[k-1]$ (or $k = 0$) but there exists no selected state for $obs[k]$.*

Example 6. *We consider the following sets of variables: $\mathtt{V} = \{\mathtt{a}, \mathtt{x}, \mathtt{pre(x)}\}$, $\mathtt{O} = \{\mathtt{a}\}$, $\mathtt{D} = \{\mathtt{x}\}$ and $\mathtt{T} = \mathtt{V_{hasPre}} = \{\mathtt{x}\}$. $\Delta = \{\neg\mathtt{a} \rightarrow \neg\mathtt{pre(x)}\}$ means that $\overline{a}$ can only be observed when $\overline{x}$ held at the precedent time step, while the observation a can occur in any state. $\Gamma = \{\top : x \prec \overline{x}\}$ represents the preference for diagnosis x over $\overline{x}$. We suppose that $s_0 = axpre(x)$.*

Let's consider the sequence of observations $obs = \{a, a, \overline{a}\}$. The procedure selects $s_1 = axpre(x)$ for $obs[1]$ and thus cannot select a state for $obs[2]$. More precisely, s_1 is the selected state at time step 1, the transmitted assignment is $t_1 = x$. Thus, a selected state s_2 must verify (1) $s_2 \models \neg\mathtt{a}$, (2) there exists s'_1 such that

$s'_1 \models \mathtt{a}$ *(observations),* $s'_1 \models \mathtt{x}$ *(transmitted variables),* $s'_1(\mathtt{x}) = s_2(\mathtt{pre(x)})$ *(consistency for variables of* $\mathtt{V_{hasPre}}$*) and* $s_2 \models \neg\mathtt{a} \rightarrow \neg\mathtt{pre(x)}$ *(consistency with* Δ*). Such a state does not exist.*

A deadlock can occur for two reasons: the observation at time step k is inconsistent with the model, or at some point in the past, the diagnoser has selected a state that is not the true system state, and that cannot explain the ulterior observations, as illustrated in Example 6. Possible means of mitigation are backtracking, increasing the diagnoser's memory, and resetting the diagnoser's state. A comparison of these techniques is beyond the scope of this paper, we discuss here how the content of the T set impacts the diagnoser's deadlock behaviour.

4.1 Handling deadlocks

First, let us assume that $\mathtt{T} = \mathtt{V_{hasPre}}$. In this case, the diagnoser keeps all the information from the previous state, and is certain to return a preferred diagnosis as of Definition 7. The drawback is that the diagnoser cannot correct a wrong branching choice made at a previous time step, and it is likely to deadlock.

Proposition 1. *Assuming* $\mathtt{T} = \mathtt{V_{hasPre}}$*, let obs* $= (o_0, \ldots, o_k)$ *be a sequence of observations. If there exists a sequence of states* $(s_0, \ldots, s_k)$ *such that for all* $i \in [0\,..\,k]$*,* s_i *is a selected state for obs*$[i]$*, then this sequence is a preferred explanation for obs.*

Sketch of proof (by induction). The proposition is obvious for $k = 0$. If we suppose that this property is true for $k-1$, then transmitted variables at time step k come from the state s_{k-1} of a preferred explanation $(s_0, \ldots, s_{k-1})$. From Definition 8, the selected state s_k, if it exists, is maximal for $<_\Gamma$ and is consistent with the selected state s_{k-1}. That makes $(s_0, \ldots, s_{k-1}, s_k)$ a preferred explanation. ■

In Example 6, $\mathtt{T} = \mathtt{V_{hasPre}}$ ensures stability and consistency of diagnoses but is very deadlock prone.

Second, we consider the other extreme case where $\mathtt{T} = \emptyset$. In this case, the diagnoser is allowed to modify the assignments to the variables made at previous time steps and can correct a wrong branching choice. If the procedure deadlocks with $\mathtt{T} = \emptyset$, this means that there is some inconsistency in the model of the system: there exists no diagnosis consistent with the observations and the model.

Proposition 2. *Assuming* $T = \emptyset$*, let obs be a sequence of observations. If the procedure deadlocks for obs then there exists no diagnosis for obs.*

Sketch of proof The existence of a diagnosis means that there exists an explanation, thus a preferred explanation $(\alpha_0, \ldots, \alpha_k)$. This explanation is such that $\alpha_{k|\mathtt{O}} = o_k$, $\alpha_{k-1|\mathtt{O}} = o_{k-1}$, $\alpha_{k-1} \xrightarrow{\Delta} \alpha_k$ and α_k is maximal for $\prec_\Gamma$. This means that α_k should be selected by the procedure. ■

However, assuming $\mathtt{T} = \emptyset$ can also let the diagnoser introduce spurious explanations. This is illustrated in the following example.

Example 7. *We reuse the system model from example 6, and assume now that* $\mathtt{T} = \emptyset$*.* $s_1 = ax\overline{pre(x)}$ *is selected for obs*$[1]$ *and* $s_2 = a\overline{x}\overline{pre(x)}$ *is selected for obs*$[2]$*. Note that* (s_0, s_1, s_2) *is not an explanation as* s_1 *and* s_2 *are not consistent successors by* Δ *but* (s_0, s'_1, s_2)*,where* $s'_1 = a\overline{x}pre(x)$*, is an explanation.*

We now suppose that $\overline{a}$ *is observed at time step* 1*. This observation is not consistent with the model of the system and the initial state* s_0*. However, the procedure does not deadlock and selects state* $s_1 = ax\overline{pre(x)}$*. A spurious diagnosis has been produced.*

A possible compromise is to use $\mathtt{T} = \mathtt{V_{hasPre}}$ until we have a deadlock, and temporarily set $\mathtt{T} = \emptyset$ only upon deadlock, which produces an approach similar to the state reset described in [Grastien *et al.*, 2009]. Another option is to introduce "model correctness" variables in Δ that trigger the transition relation constraints, in a modeling style similar to [Reiter, 1987]. This makes it possible to express conditional preferences on these additional variables.

4.2 Predicting deadlocks

Rather than handling deadlocks online, another approach is to anticipate their existence, and prevent them by modifying the diagnosis model. A wide range of checks is relevant to the deadlock problem: check that the diagnoser accepts all the observation sequences generated by Δ, the finite trackability check from [Grastien *et al.*, 2009], correctness checks that ensure the diagnoser is wrong during a bounded window. A non trivial task is to differentiate deadlocks due to the diagnoser choosing a wrong branch, from those due to an observation sequence inconsistent with Δ. All these checks are closely related to the diagnosis stability check and are beyond the scope of this paper, and left for future work. Approaches such as the twin plant [Zaytoon and Lafortune, 2013] may apply.

5 Conclusion

Modeling intermittent faults is considerably more complicated than permanent faults, and requires the use of customized observers. The diagnosis task is also made difficult due to phenomena such as instability that are not documented in the literature. We presented a modeling approach that introduces conditional diagnosis preferences for selecting a unique diagnosis among a set of candidates, as opposed to pure evidence based reasoning.

This work requires some effort before reaching maturity: formal definitions for stability, conditions for bounded diagnosability, comparison of compilation techniques and incremental evaluation approaches are currently being studied. However, we release fundamental assumptions about the diagnosis process that we have found to be crippling in previous studies, and we are confident that this approach will open a new range of application domains.

This work can be extended in several directions: the preference specification language can be enriched, other encodings for lexicographic optimization criteria could be used [Marques-Silva *et al.*, 2011], and group MaxSAT encodings [Heras *et al.*, 2012] are also applicable to our approach.

References

[Abreu and Cardoso, 2013] Rui Abreu and Nuno Cardoso. A kernel density estimate-based approach to component goodness modeling. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*, pages 152–158. AAAI, 2013.

[Boutilier *et al.*, 2004] C. Boutilier, R. I. Brafman, C. Domshlak, H. H. Hoos, and D. Poole. Preference-based constrained optimization with cp-nets. *Computational Intelligence*, 20:137–157, 2004.

[Contant *et al.*, 2004] Olivier Contant, Stéphane Lafortune, and Demosthenis Teneketzis. Diagnosis of intermittent faults. *Discrete Event Dynamic Systems*, 14(2):171–202, 2004.

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

[Darwiche and Marquis, 2002] Adnan Darwiche and Pierre Marquis. A knowledge compilation map. *Journal of Artificial Intelligence Research*, 17(1):229–264, 2002.

[De Kleer, 2009] Johan De Kleer. Diagnosing multiple persistent and intermittent faults. In *IJCAI*, pages 733–738, 2009.

[Emerson, 1990] E. Emerson. Temporal and Modal Logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B:Formal Models and Semantics*, pages 995–1072. Elsevier, 1990.

[Felfernig and Schubert, 2010] Alexander Felfernig and Monika Schubert. Fastdiag: A diagnosis algorithm for inconsistent constraint sets. In *Proceedings of the 21st International Workshop on the Principles of Diagnosis (DX 2010), Portland, OR, USA*, pages 31–38, 2010.

[Ghallab *et al.*, 2004] M. Ghallab, D. Nau, and P. Traverso. *Automated Planning: Theory and Practice*. Morgan Kaufmann, 2004.

[Grastien *et al.*, 2009] Alban Grastien, Anbu Anbulagan, et al. Incremental diagnosis of DES with a non-exhaustive diagnosis engine. In *Proceedings of the 20th International Workshop on Principles of Diagnosis*. Linkoping University Institute of Technology, 2009.

[Havelund and Rosu, 2002] K. Havelund and G. Rosu. Synthesizing monitors for safety properties. In *Proc. of the 8th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS-02)*, pages 342–356, 2002.

[Heras *et al.*, 2012] Federico Heras, Antonio Morgado, and Joao Marques-Silva. *Proceedings of the 25th Canadian Conference on Artificial Intelligence, Canadian AI 2012, Toronto, ON, Canada*, chapter An Empirical Study of Encodings for Group MaxSAT, pages 85–96. 2012.

[Jéron *et al.*, 2006] Thierry Jéron, Hervé Marchand, Sophie Pinchinat, and Marie-Odile Cordier. Supervision patterns in discrete event systems diagnosis. In *Discrete Event Systems, 2006 8th International Workshop on*, pages 262–268. IEEE, 2006.

[Jiang and Kumar, 2004] Shengbing Jiang and Ratnesh Kumar. Failure diagnosis of discrete-event systems with linear-time temporal logic specifications. *Automatic Control, IEEE Transactions on*, 49(6):934–945, 2004.

[Marques-Silva *et al.*, 2011] J. Marques-Silva, J. Argelich, A. Graça, and I. Lynce. Boolean lexicographic optimization: algorithms & applications. *Ann. Math. Artif. Intell.*, 62(3-4):317–343, 2011.

[Prestwich *et al.*, 2004] S. Prestwich, F. Rossi, K. Venable, and T. Walsh. Constrained cp-nets. In *Proceedings of the Joint Annual Workshop of ERCIM/CoLogNet on Constraint Solving and Constraint Logic Programming (CSCLP'04)*, 2004.

[Prestwich *et al.*, 2005] Steve Prestwich, Francesca Rossi, Kristen Brent Venable, and Toby Walsh. Constraint-based preferential optimization. In *AAAI*, volume 5, pages 461–466, 2005.

[Reiter, 1987] Raymond Reiter. A theory of diagnosis from first principles. *Artificial intelligence*, 32(1):57–95, 1987.

[Ricks and Mengshoel, 2010] BW Ricks and OJ Mengshoel. Diagnosing intermittent and persistent faults using static bayesian networks. In *Proc. of the 21st International Workshop on Principles of Diagnosis (DX-10)*, 2010.

[Sampath *et al.*, 1995] Meera Sampath, Raja Sengupta, StCphane Lafortune, Kasim Sinnamohideen, and Demosthenis Teneketzis. Diagnosability of discrete-event systems. *Automatic Control, IEEE Transactions on*, 40(9):1555–1575, 1995.

[Torta and Torasso, 2007] Gianluca Torta and Pietro Torasso. An on-line approach to the computation and presentation of preferred diagnoses for dynamic systems. *AI Communications*, 20(2):93–116, 2007.

[Tseitin, 1968] Grigori S Tseitin. On the complexity of derivation in propositional calculus. *Studies in constructive mathematics and mathematical logic*, 2(115-125):10–13, 1968.

[Wilson, 2011] Nic Wilson. Computational techniques for a simple theory of conditional preferences. *Artificial Intelligence*, 175(7):1053–1091, 2011.

[Zabi *et al.*, 2013] S. Zabi, P. Ribot, and E. Chanthery. Health monitoring and prognosis of hybrid systems. In *Annual Conference of the Prognostics and Health Management Society (PHM)*, 2013.

[Zaytoon and Lafortune, 2013] Janan Zaytoon and Stéphane Lafortune. Overview of fault diagnosis methods for discrete event systems. *Annual Reviews in Control*, 37(2):308–320, 2013.