Diagnosis of Hybrid Systems by Consistency Testing

Alban Grastien[1,2]
[1]Optimisation Research Group, NICTA, Australia
[2]Artificial Intelligence Group, Australian National University, Australia
e-mail: first.last@nicta.com.au

Abstract

We propose a new approach of diagnosis of hybrid systems using SMT (SAT Modulo Theory) technology. Diagnosis of hybrid systems is reduced to a series of diagnosis questions that are solved using SMT solvers. We show that this approach allows to deal with a new class of systems and paves the way to a new approach to diagnosis of hybrid systems.

1 Introduction

We consider the problem of diagnosis of hybrid systems, i.e., dynamic systems that involve both continuous (e.g., temperature) and discrete – but possibly on continuous domains – (e.g., voltage, switch position) evolutions.

1.1 Related Work

Hybrid systems have been extensively studied in the literature. They are sometimes designated by the expression "multiple mode systems," where the set of modes is the domain of the discrete variables and each mode defines a different set of rules for the continuous evolution.

There are essentially two classes of diagnostic approaches for hybrid systems. The first class builds on the redundancies in the system model: under certain assumptions on the system mode (for instance, no fault in a specific part of the system), the model equations allow to derive a constraint (or *indicator*) on the observed variables, such that the violation of this constraint is indicative of a violation of one of the assumptions [Staroswiecki and Comtet-Varga, 2001]. The main issue addressed by the research community for this type of approach is that of the number of indicators, in particular because the set of system modes is the Cartesian product of its constituent modes and is consequently of exponential size. The popular strategy is to instantiate the indicators on the fly [Heintz *et al.*, 2008]. Other issues include flexibility: because the indicators are built for a given observability, they cannot accomodate optimally a change of observability (for instance, different sampling rates, temporary masking, etc.) and the automatic generation of the indicators when the model involves non reversible functions.

The second class of algorithms is based on the simulation of the hybrid system either tentatively exhaustively (multiple Kalman filters [Blom and Bar-Shalom, 1988]) or through Monte Carlo sampling (particle filters [Arulampalam *et al.*, 2002]). Given a probabilistic belief state, the possible evolutions of the state are computed and compared to the actual observation to update their respective probabilities. Again a major problem of this type of approach is that the number of evolutions is too large to enumerate and a substantial body of work is dedicated to reducing the representation of the belief state through approximation or pruning [Blom and Bar-Shalom, 1988; Benazera and Travé-Massuyès, 2009]. Such operations are however very hazardous because the correct assumption of unlikely events such as faults must often go through a steady-increasing probability period where the probability is still so low that it may be reset by the approximation process. Furthermore, this approach necessitates a model from which the state evolution can be predicted. In case of unknown behavioural mode, the best that can be done is to ignore the variables whose value cannot be predicted [Hofbaur and Williams, 2004].

Diagnosis of hybrid systems with SMT solvers was first suggested by Ernit and Dearden [2011]. The main differences with the work presented here is that i) they limit themselves to real-valued variables and do not discuss the dynamics of continuous variables, and ii) they only consider a conflict-based approach, while other diagnostic strategies can be used [Grastien *et al.*, 2011]. Finally the object of this article is also to argue that a consistency-based approach, as opposed to simulation-based approaches presented above, allows for less precise models, i.e., models that may not predict the future state but merely give constraints on the state variables.

1.2 Our Approach

In this work we propose an approach for the diagnosis of hybrid systems based on consistency. We use the framework of Grastien *et al.* [2012] which formulates the diagnosis problem as a series of diagnosis questions, each of which tests the consistency of some diagnostic assumptions with the diagnosis problem at hand. We assume that the number of discrete steps can be bounded and we reduce each question to the problem of finding an assignment to the state variables (both discrete and continuous) at these time steps that is consistent with (i.e., does not contradict) the model, the observation, and the assumptions. More specifically we reduce the diagnosis question to a constraint satisfaction problem written in the SMT (SAT Modulo Theory) framework. As opposed to the classical approach where the state of the system at a given time is defined as a function of the state at previous time (time being either discrete or continuous), we define the model as a number of constraints between the

values of the continuous and discrete variables at the current and the previous (discrete) times. For simplicity we restrict ourselves to piece-wise linear constraints; our approach can deal with more general models, but the number of existing solvers for such SMT problems drops quickly with the complexity of the constraints.

Compared to the simulation-based approaches, our approach does not require to maintain a probabilistic belief state which, to be precise enough, is prohibitive to maintain (in the case of the Adapt Lite system used for our experiments, the discrete space is already in the order of 10^{12} without considering the real-valued state variables). Furthermore we can deal with unknown behavioural modes: state variables whose evolution in certain modes is unspecified; in such situations, the CSP simply leaves these state variables assignments free. Compared to the redundancy-based approaches, our approach can accomodate any observability.

The here-proposed approach bears some similarities with bounded model-checking for hybrid systems [Audemard *et al.*, 2004]. There are however significant differences in the type of SMT problems generated. BMC for instance is incomplete and, consequently, is generally used to its limits; with diagnosis, we want to return solutions that are as precise as the model allows. On the positive side, diagnosis is based on a partial-observed observed run of the system, which means that the state space that needs to be explored is more restricted than in a model-checking problem.

We first present the hybrid system model and the diagnostic problem. We next present the reduction to SMT of a diagnostic test, and illustrate the implementation on the Adapt-Lite from the DX Competition.

2 Hybrid Systems

Hybrid systems are a class of models that include both variables that evolve continuously and variables that evolve discretely. The evolution of the continuous variables is usually defined by a set of equations that changes according to the value of the discrete variables.

Ideally the evolution in the continuous space is described by differential equations: $\dot{\mathbf{x}} = f_Q(\mathbf{x}, \mathbf{i}, \mathbf{w})$ where $\mathbf{x}$ is the vector of (continuous) state variables, $\dot{\mathbf{x}}$ is the differential state vector, $\mathbf{i}$ is the input vector, $\mathbf{w}$ is a (usually white Gaussian) noise, and Q is the system mode (discrete state). For computational reasons complex differential equations are often dropped in favour of difference equations: $\mathbf{x}@\tau' = f_Q(\mathbf{x}@\tau, \mathbf{i}, \tau' - \tau, \mathbf{w})$ where $\tau < \tau'$ are two times such that the input and the mode of the system is unchanged during the time period $[\tau, \tau']$. The problem is often restricted to situations where f_Q is linear, which is an acceptable assumption when piece-wise linear approximations are precise enough (each piece then corresponds to a different system mode).

The models presented above are used either to simulate the system or to deduce useful indicators. In a consistency-based approach, we do not need such a precise model, although more precision in the model will mean more precision in the diagnosis. We will still consider a discretisation of time, i.e., that the state of the system will be specified at important instants called timesteps although the time of these timesteps will be unspecified in general, and their number unbounded. The model will simply be a collection of *constraints* between the values of system variables at consecutive timesteps. Notice that there is a reversal with discrete event systems where the timesteps are generally associated with transitions, and states are represented for time intervals.

To simplify notations, we assume that each state variables is defined over the set of reals $\mathbf{R}$; one such variable is `time` which models the time. We limit ourselves to linear constraints because the number of solvers and their performance diminish quickly with more general constraints. A linear expression is an inequality involving over linear terms, i.e., a linear combination of variables with rational coefficients. A linear constraint is a collections of linear expressions connected with the logical operators such as $\neg$, $\wedge$, and $\vee$.

Our definition of hybrid systems is very similar to the one proposed by de Moura et al. [2008] except that i) I does not refer to the initial state (which is included in the observations) but to the state invariants, ii) the state invariants are extracted from the transition relation, and iii) to simplify modeling, we distinguish *soft* invariants from *hard* invariants. Hard invariants are constraints that apply in any state; soft invariants can temporarily become true as an effect of an instantaneous transition but must lead to another instantaneous transition. For instance, a nominal circuit breaker cannot remain closed in a state where the current is over a given limit; however that limit can be temporarily exceeded as a consequence of a short circuit.

Definition 1 *A* hybrid system *is a tuple* $\langle V, I_{\mathrm{H}}, I_{\mathrm{S}}, T\rangle$ *where* V *is a set of* variables*,* I_{H} *and* I_{S} *are two sets of hard and soft* invariants *both defined as sets of constraints over* V*, and* T *is a set of* transition constraints *defined as a set of cosntraints over* $V \cup V'$ *where* V' *is a copy of the variables in* V*.*

The set of transition constraints is here interpreted as a conjunction of constraints; we could see them as a disjunction of constraints i.e., an enumeration of how the system can evolve, or better as a conjunction of disjunctions, i.e., a set of synchronised local models.

A *state* of the hybrid system $\langle V, I_{\mathrm{H}}, I_{\mathrm{S}}, T\rangle$ is an assignment of the state variables: $V \to \mathbf{R}$. A *run* is a sequence of states $s_0, \ldots, s_k$ such that:

- all states s_i satisfy the hard constraints where v refers to $s_i(v)$;
- a state s_i may not satisfy a soft constraint iff a discrete transition occurred before and after s_i, i.e., $s_{i-1}(\texttt{time}) = s_i(\texttt{time}) = s_{i+1}(\texttt{time})$;
- all pairs of consecutive states s_i, s_{i+1} satisfy all transition constraints where a variable v refers to $s_i(v)$ and a variable v' refers to $s_{i+1}(v)$.

A continuous transition is a pair of consecutive states s_i, s_{i+1} such that $s_i(\texttt{time}) < s_{i+1}(\texttt{time})$. This transition is represented as follows $s_i \xrightarrow{\delta} s_{i+1}$ where $\delta = s_{i+1}(\texttt{time}) - s_i(\texttt{time})$. It is assumed that if such a transition exists, then for all $\delta' \in]0, \delta[$, there exists a state s' such that $s_i \xrightarrow{\delta'} s'$ and $s' \xrightarrow{\delta-\delta'} s_{i+1}$ exist too. This can be ensured by considering only convex constraints (the only Boolean operator allowed is $\wedge$).

3 Diagnostic Problem

A model-based diagnosis problem is defined by a model, some observations, and some faulty behaviours. We now

describe the missing elements of this definition.

3.1 Observations

The framework proposed in this paper is very flexible with respect to the observations: essentially, an observation is just an information about the system run. We illustrate state-based observations and event-based observations, but more complex observations would be allowed. In particular, it is assumed here that the observations are perfect; any uncertainty on the observations is modeled in the hybrid system itself.

Definition 2 (State-based observations) *The* state-based observations *are a collection of triples $\langle \tau, v, \nu \rangle$ where τ is the time of observation, v is the observed variable, and ν is the observed value.*

Notice that state-based observations do not assume that all observed variables are observed at the same time or at the same rate. The initial state can be modeled as (possibly partial) observations of the initial state.

A run is consistent with the state-based observation $\langle \tau, v, \nu \rangle$ if it contains a state s such that $s(\texttt{time}) = \tau$ and $s(v) = \nu$.

To keep the model simple, we did not explicitly represent events. Instead, we assume that a variable v models the occurrence of events such that certain discrete transitions set v to 1 and all other discrete transitions and all continuous transitions set v to 0.

Definition 3 (Event-based observations) *The* event-based observations *are a tuple $\langle \Sigma_{\mathrm{o}}, O \rangle$ where Σ_{o} is a subset of* observable variables *modeling the occurrence of observable events and O is a set of pairs $\langle v, \tau \rangle$ where $v \in \Sigma_{\mathrm{o}}$ is the observed event and τ is the time when the event was observed.*

As opposed to state-based observations, event-based observations provide information not only when the events are observed but also when they are not. A run $\rho = s_0, \ldots, s_k$ is consistent with the event-based observations $\langle \Sigma_{\mathrm{o}}, O \rangle$ iff $obs_{\Sigma_{\mathrm{o}}}(\rho) = O$ where $obs_{\Sigma_{\mathrm{o}}} = \{\langle v, \tau \rangle \in \Sigma_{\mathrm{o}} \times [s_0(\texttt{time}), s_k(\texttt{time})] \mid \exists i \in \{0, \ldots, k\}.\ s_i(v) = 1 \wedge s_i(\texttt{time}) = \tau\}$.

3.2 Faults

Diagnosis is about determining what is faulty in a system. A fault may be defined as a pattern of events [Jéron *et al.*, 2006]. More often though is a fault in dynamic systems defined as the occurrence of a special type of event (a "faulty" event). Equivalently a fault can be seen as the specific assignment of a state variable; this is the definition used in this document.

Definition 4 (Faults) *The faults are a set of variables $V_{\mathrm{f}} \subseteq V$.*

The faulty state of a run $s_0, \ldots, s_k$ is the set of faulty variables that are assigned to 1 at the end of the run: $\{v \in V_{\mathrm{f}} \mid s_k(v) = 1\}$.

3.3 Diagnosis

A diagnostic problem is defined by a tuple $\langle M, o, V_{\mathrm{f}} \rangle$ where M is a hybrid system, o are observations, and V_{f} is the set of faults.

A diagnostic candidate is a subset of faults $F \subseteq V_{\mathrm{f}}$ such that there exists a run allowed by the model, that can generate the observations, and whose faulty state is precisely F. The diagnosis is the set $\Delta \subseteq 2^{V_{\mathrm{f}}}$ of diagnostic candidates. Because the diagnosis is often uncertain (i.e., the size $|\Delta|$ is large) the focus is often on the minimal candidates, i.e., the sets F such that $F' \subset F \Rightarrow F' \notin \Delta$. The minimal diagnosis is the subset of minimal diagnostic candidates.

The consistency-based approach to diagnosis has been formalized by Reiter [1987] and has been recently extended for dynamic systems [Grastien *et al.*, 2011]. Consistency-based algorithms work by iteratively testing the intersection of Δ with sets of diagnostic hypotheses. Such tests are implemented by checking the consistency of i) the model and the observations (implicitly representing the diagnosis) with ii) some assumption on the faulty state (representing the diagnostic hypotheses).

The present work follows this approach and the following section is therefore dedicated to answering such diagnostic questions.

4 Answering a Diagnostic Question

We consider the problem of deciding whether a set of assumptions on the faulty state is consistent with a hybrid system and its observations. In this paper, we reduce this problem to a SAT Modulo Theory (SMT) problem and to solve it using state of the art technology in SMT.

SMT is an extension of the problem of propositional satisfiability (SAT) to contain operations from various theories such as the Boolean, bit-vectors, arithmetic, arrays, and recursive datatypes [de Moura *et al.*, 2007]; the linear arithmetic ($\mathcal{LA}$) is sufficient for this paper.

A $\mathcal{LA}$ problem is a tuple $\langle \mathcal{V}_{\mathrm{B}}, \mathcal{V}_{\mathrm{L}}, Cs \rangle$ where $\mathcal{V}_{\mathrm{B}}$ is a set of Boolean-valued variables, $\mathcal{V}_{\mathrm{L}}$ is a set of real-valued variables, and $Cs \subset \mathrm{Constraints}(\mathrm{LI}(\mathcal{V}_{\mathrm{L}}) \cup \mathcal{V}_{\mathrm{B}})$ is a set of constraints defined by the Boolean-valued variables and linear inequalities over the set of real-valued variables. A *solution* to a $\mathcal{LA}$ problem is an assignment of the variables $\mathcal{V}_{\mathrm{B}} \cup \mathcal{V}_{\mathrm{L}}$ that makes all the constraints in Cs logically true. The problem is said *satisfiable* if there exists a solution, *unsatisfiable* otherwise.

4.1 Defining the Set of Variables

The reduction of a diagnostic question to a $\mathcal{LA}$ problem bears many similarities with the reduction from classical AI planning to SAT [Kautz and Selman, 1996] and even more with Bounded Model Checking for hybrid systems [de Moura *et al.*, 2008]. The SMT solver will be asked to find a system run that generates the observations while satisfying the faulty assumptions. As usual, we assume that the number n of transitions in the system run is bounded. Since any continuous evolution can be splitted in any number of transitions, it can be assumed that the number of transitions is exactly n. We are therefore looking for the run $s_0, \ldots, s_k$ which can be modeled by defining for each variable $v \in V$ and each timestep $i \in \{0, \ldots, k\}$ an SMT variable $v@t$ which will be assigned the value ν iff $s_t(v) = \nu$.

4.2 Translating the Model

We define a set of SMT constraints whose set of solutions maps exactly the system runs with $k + 1$ states.

For every hard constraint C, for every timestep $t \in \{0, \ldots, k\}$, we include the constraint $C@t$ that corresponds to the constraint C where every variable v is replaced by $v@t$.

Similarly a soft constraint must be satisfied in a state unless the state is surrounded by discrete transitions: for every soft constraint C, for every timestep $t \in \{0, \ldots, k\}$, we include $C@t \vee (\texttt{time}@(t-1) = \texttt{time}@t \wedge \texttt{time}@t = \texttt{time}@(t+1))$ (whether the soft constraints should be satisfied in the initial/final states is debatable).

Because the set of transition constraints is interpreted as a conjunction, each such constraint must be satisfied. We write $C@(t, t')$ the rewritting of the constraint C where each variable v is replaced by $v@t$ and each variable v' is replaced by $v@t'$. Each transition constraint C is therefore enforced by the SMT constraints $C@(t-1, t)$ where $t \in [1, \ldots, k]$.

4.3 Translating the Observation

Given observations, we define a set of SMT constraints whose set of solutions maps the runs that generate these observations.

Regardless of the type of observations, we assume that a timestep t_τ is associated with each time τ mentionned in the observations.

Given the state-based observation $\langle \tau, v, \nu \rangle$, we simply enforce the SMT constraint $v@t_\tau = \nu$.

Given the event-based observations $\langle \Sigma_o, \{\langle \tau_i, v \rangle\}_{i \in [1,\ldots,n]} \rangle$, for every observable event $v \in \Sigma_o$ and every timestep $t \in \{0, \ldots, k\}$, if there exists $i \in [1, \ldots, n]$ such that $v_i = v$ and $t = t_{\tau_i}$, then we define the SMT constraint: $v@t = 1$; otherwise (if no such i exists), we enforce $v@t = 0$.

4.4 Translating the Assumptions

In the experiments next section, each test will propose to find a candidate F' that is strictly better ($F' \subset F$) than another candidate F already found. To do so, we need i) to disallow the faults in $V_f \setminus F$ and ii) to forbid all faults from F to be active:

$$\bigwedge_{v \in V_f \setminus F} v@k = 0 \ \wedge \ \bigvee_{v \in F} v@k = 0.$$

5 The Adapt System

The validation of this approach is based on the industrial track of the international diagnostic competition [Kurtoglu *et al.*, 2009], i.e., the Adapt-Lite EPS. The system is composed of roughly 35 components including 20 sensors (which may be subject to fault).

5.1 Modeling the Components

The object of this subsection is to illustrate how the components in Adapt can be modeled.

Real-Valued Sensors

Adapt includes a number of sensors. The real-valued sensors return the (real) value of a state variable (say `m`).

The sensor can be in nominal, offset, stuck, or drifting state. This is modeled by three faulty state variable `fo`, `fs`, and `fd`. The value actually observed is the value of the free variable `o`. In nominal state, the observation will be the actual value plus the noise (here limited to N), which is represented by the following hard constraint:

$$\texttt{fo} = 0 \wedge \texttt{fs} = 0 \wedge \texttt{fd} = 0 \Rightarrow \texttt{m} - N \leq \texttt{o} \leq \texttt{m} + N$$

Discrete variable `t` represents the offset:

$$\texttt{fo} = 1 \Rightarrow \texttt{m} - N \leq \texttt{o} - \texttt{t} \leq \texttt{m} + N.$$

Discrete variable `k` represents the stuck-at value.

$$\texttt{fs} = 1 \Rightarrow \texttt{o} = \texttt{k}.$$

Continuous variable `co` represents the current offset of a drifting, and is used similarly to the offset value. Contrary to `offset` though, its value varies continuously. The drift could be defined as a linear variation, i.e., $\texttt{co}' - \texttt{co} = \texttt{slope} \times (\texttt{time}' - \texttt{time})$, but this function is not linear. Therefore, either $(\texttt{time}' - \texttt{time})$ is assumed known for every timestep (which is a possibility), but in general the drift should be modeled in a discrete fashion as follows: $\texttt{fd} = x \Rightarrow m_1 \times (\texttt{time}' - \texttt{time}) \leq \texttt{co}' - \texttt{co} \leq m_2 \times (\texttt{time}' - \texttt{time})$ where m_1 and m_2 are the lower and upper bound of the x-th discretisation of the drifting.

Fan Output

The fan output is interesting because it varies very slowly (as opposed to the voltage for instance). Temperature (for light bulbs) acts similarly in the large Adapt system. The flow increases roughly linearly from 0 to a maximum M and decreases similarly depending. We model the flow with variable `v` and hard constraints are defined to forbid the variable to leave the interval $[0, M]$. Furthermore four continuous constraints define the continuous evolution such as:

$$(\texttt{running} \wedge \texttt{v} \leq M) \Rightarrow \texttt{v}' - \texttt{v} = sl \times (\texttt{time}' - \texttt{time})$$

where sl is the constant slope of the linear increasing function.

5.2 Experiments

We modeled the Adapt-Lite system proposed by the DX-Competition [Kurtoglu *et al.*, 2009]. We ran a series of experiments over diagnostic windows of 5 seconds each at a rate of 2 hertz. The initial state is assumed nominal.

The classical consistency-based approach to diagnosis [Reiter, 1987] tests whether the nominal state is consistent with the diagnostic problem and uses the conflict returned by the solver to generate more diagnostic tests until a solution is found. The `cvc4` solver used in our experiments does not implement the conflict generation as yet and we therefore turned to the so-called PLS strategy proposed by Grastien et al. [2011]. This strategy searches for any diagnosis (i.e., checks the consistency of trivial assumption with the model and the observation). The SMT solver returns a possible behaviour of the system. The diagnostic state is extracted from this behaviour and another consistency check is performed, this time with the assumption that the faulty state should be preferred.

We provide some experimental results on a few selected instances on Table 1. The experiments were performed on an Intel i5-2520M 2.5GHz with 3.75GiB and running GNU/Linux 2.6.43.8-1.fc15.x86_64. The solver used is cvc4-1.0-x86_64. Each diagnostic test is performed independently, i.e., not incrementally.

The runtime are still very weak, with generally 2 min to solve any problem. Notice however that we did not try to apply any optimization or heuristic. It is well known that the SMT encoding affects greatly the performance of the solver. Furthermore, the search strategy presented here might impair the performance: for instance, testing the consistency

instance	time	nb faults	nb tests
1	2mn20	0	11
2	2mn00	1	10
3	3mn00	1	9
4	0mn56	1	4
5	2mn15	2	11

Table 1: Experimental results: time: the time necessary to solve the diagnostic problem; nb faults: size of the minimal cardinality diagnosis returned by the diagnoser; nb tests: number of calls to the SMT solver.

of the nominal faulty state with the observations in the first instance takes less than 10 seconds, i.e., the conflict-based approach would solve the first instance in less than 10 seconds despite the current encoding's shortcomings.

An advantage of the PLS strategy is however that finding multiple cardinality minimal diagnoses is not harder than finding single cardinality diagnoses since the diagnoser can actually stop earlier when the minimal cardinality is larger. Considering the issue of multiple cardinality again, a simulation-based approach would struggle with a scenario involving several faults at the same time as the probability of any consistent scenario would be extremely low. Our approach, on the other hand, is able to deal with any number of faults.

6 Conclusion and Future Works

We presented a novel approach to diagnosis of hybrid systems that is based on consistency tests. We discussed its advantages compared to existing approaches. In short, it does not require the astronomical space necessary for simulation-based approaches while being highly flexible with respect to its input (model, observation, and fault).

The experimental results provided in this article show that the approach is feasible although it is not quite applicable yet. We are optimistic that careful reduction to SMT and dedicated implementation of the SMT solver can gain orders of magnitude in runtime. In particular, we believe that there is much more potential here than, say, in SMT-based solving of model-checking problems [Ábrahám *et al.*, 2005]. We believe, for instance, that observations (which are inexistant in model-checking) make the SMT problem much easier to solve; also, there is much more potential for decomposition of a diagnostic problem into problems of (potentially overlapping) subsystems.

Similarly to the redundancy-based approach, the consistency-based approach as presented here is short-sighted, in the sense that it does not take into account all observations since the beginning, but only the last bits of observation. This is one strength of the simulation-based approach which keeps in the belief state the important information about old observations. Typically for the Adapt System, one wants to remember the configuration of the network (which switches were open/closed). Incrementality, and in particular how much information should/needs to be remembered about past observations will allow to track faulty behaviours that manifest themselves in long periods.

Acknowledgments

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References

[Ábrahám *et al.*, 2005] E. Ábrahám, B. Becker, F. Klaedtke, and M. Steffen. Optimizing bounded model checking for linear hybrid systems. In *Sixth International Conference on Verification, Model Checking and Abstract Interpretation (VMCAI-05)*, pages 396–412, 2005.

[Arulampalam *et al.*, 2002] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing (TSP)*, 50(2):174–188, 2002.

[Audemard *et al.*, 2004] Gi. Audemard, M. Bozzano, A. Cimatti, and R. Sebastiani. Verifying industrial hybrid systems with MathSAT. In *Second International Workshop on Bounded Model Checking (BMC-04)*, pages 17–32, 2004.

[Benazera and Travé-Massuyès, 2009] E. Benazera and L. Travé-Massuyès. Set-theoretic estimation of hybrid system configurations. *IEEE Transactions on Systems, Man, and Cybernetics (TSMC)*, 39(5):1277–1290, 2009.

[Blom and Bar-Shalom, 1988] H. Blom and Y. Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control (TAC)*, 33(8):780–783, 1988.

[de Moura *et al.*, 2007] L. de Moura, B. Dutertre, and N. Shankar. A tutorial on satisfiability modulo theories. In *Nineteenth International Conference on Computer-Aided Verification (CAV-07)*, pages 20–36, 2007.

[de Moura *et al.*, 2008] L. de Moura, H. Rueß, and M. Sorea. Bounded model checking and induction: from refutation to verification. In *20th International Conference on Computer-Aided Verification (CAV-08)*, pages 14–26, 2008.

[Ernits and Dearden, 2011] J. Ernits and R. Dearden. Towards diagnosis modulo theories. In *22nd International Workshop on Principles of Diagnosis (DX-11)*, pages 249–256, 2011.

[Grastien *et al.*, 2011] A. Grastien, P. Haslum, and S. Thiébaux. Exhaustive diagnosis of discrete event systems through exploration of the hypothesis space. In *22nd International Workshop on Principles of Diagnosis (DX-11)*, pages 60–67, 2011.

[Grastien *et al.*, 2012] A. Grastien, P. Haslum, and S. Thiébaux. Conflict-based diagnosis of discrete event systems: theory and practice. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning (KR-12)*, 2012.

[Heintz *et al.*, 2008] F. Heintz, M. Krysander, J. Roll, and E. Frisk. FlexDx: a reconfigurable diagnosis framework. In *Nineteenth International Workshop on Principles of Diagnosis (DX-08)*, pages 79–86, 2008.

[Hofbaur and Williams, 2004] M. Hofbaur and B. Williams. Hybrid estimation of complex systems. *IEEE Transactions on Systems, Man, and Cybernetics (TSMC)*, 34(5):2178–2191, 2004.

[Jéron *et al.*, 2006] T. Jéron, H. Marchand, S. Pinchinat, and M.-O. Cordier. Supervision patterns in discrete-event systems diagnosis. In *Seventeenth International Workshop on Principles of Diagnosis (DX-06)*, pages 117–124, 2006.

[Kautz and Selman, 1996] H. Kautz and B. Selman. Pushing the envelope : planning, propositional logic, and stochastic search. In *Thirteenth Conference on Artificial Intelligence (AAAI-96)*, pages 1194–1201, 1996.

[Kurtoglu *et al.*, 2009] T. Kurtoglu, S. Narasimhan, S. Poll, D. Garcia, L. Kuhn, J. de Kleer, A. van Gemund, and A. Feldman. First international diagnosis competition – DXC'09. In *20th International Workshop on Principles of Diagnosis (DX-09)*, pages 383–396, 2009.

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

[Staroswiecki and Comtet-Varga, 2001] M. Staroswiecki and G. Comtet-Varga. Analytical redundancy relations for fault detection and isolation in algebraic dynamic systems. *Automatica*, 37:687–699, 2001.