ECAI 2014
T. Schaub et al. (Eds.)

doi:10.3233/978-1-61499-419-0-405

Diagnosis of Hybrid Systems with SMT: Opportunities and Challenges

Alban Grastien[1]

Abstract. We propose a new approach to diagnosis of hybrid systems. In this approach, questions about the behavior of the system are asked and translated into Satisfiability Modulo Theory (SMT) problems, which are then solved by an SMT solver. We show the reduction to SMT. We also discuss the benefits and the drawbacks of this approach and conclude with a number of research directions that will make this approach applicable to large systems.

1 Introduction

Because of imperfection, misuse, or natural ageing, any system is prone to malfunction. Diagnosis is the problem of detecting these malfunctions and identifying/isolating which components and what type of faults are involved. Model-based diagnosis uses a description of the system, the *model*, to reason about the possible defects of the system.

Hybrid systems are a class of models for dynamic systems that involve both discrete dynamics and continuous dynamics. This combination of different variable types makes diagnosis of hybrid systems a hard problem, as discrete dynamics generally require to branch while continuous dynamics involve complex computations. Narasimhan and Biswas [17] used the hybrid bond graph formalism. The approach pioneered by Bayoudh et al. [4] decomposes the problem in a continuous state estimation followed to a reasoning at the discrete level.

In this paper we propose the first diagnoser that fully incorporates both aspects of hybrid systems. Our approach builds on the consistency-based theory of diagnosis developed by de Kleer, Reiter, and Williams [19, 5] that we recently revived for discrete event systems [14]. Our diagnoser generates "diagnostic questions", in practice consistency checks, that ask whether the model allows for a behaviour consistent with the observations and satisfying a specified assumption (e.g., that the behaviour is nominal). An external solver performs these consistency check and the diagnosis is logically inferred from these answers, unless the diagnoser needs to ask more questions. Fault detection (determining that a fault occurred on the system) is a special case of our approach.

More concretely each consistency check is formulated as a Satisfiability Modulo Theory (SMT) problem, similarly to Bounded Model Checking by SMT [2, 9, 15]. 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 [6]; the linear arithmetic is sufficient for this paper. The reduction to SMT consists in finding the values of the (discrete and continuous) state variables at relevant instants of the diagnostic window. The solution to the SMT problem represents a system behaviour; SMT constraints are defined that make sure that the set of solutions is exactly the set of behaviours authorized by the model, and consistent with the observation and the assumption. Using the linear arithmetic capabilities of SMT solvers allows us to handle both the discrete dynamics and the continuous dynamics.

Next section we present the diagnosis of hybrid systems with an emphasis on how we model the system. We then show how we can solve this problem with a consistency-based approach powered by SMT solvers. Section 4 illustrates the current state of this approach. A long discussion concludes the paper which presents a critical analysis of this approach.

2 Diagnosis of Hybrid Systems

2.1 Hybrid Systems

We are interested in model-based diagnosis, a general approach to diagnosis where a model of the system is provided. We are dealing here with hybrid systems, that is, systems that involve both discrete and continuous variables. A discrete variable would for instance represent the state open/close of a switch whilst a continuous variable would model the temperature of a component. Importantly we do not assume that discrete changes are observable: the value of the discrete variables at a time is a priori unknown.

There exist many different types of hybrid system formalisms. Many approaches in the continuous community use the model to simulate the system behaviour and estimate the likelihood of these simulations compared to the actual observations. In contrast the consistency-based approach to diagnosis essentially just necessitates a description of how the system *cannot* operate in certain situations. For instance in a nominal state (and at a certain level of abstraction) a closed circuit breaker cannot have different voltages at its two ends.

A hybrid system is a tuple $\langle V, C, T\rangle$ where

[1] Optimisation Research Group, NICTA, and Artificial Intelligence Group, Australian National University. NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

- V is the set of *state variables*;
- C is a set of *state constraints* over the value of the state variables;
- T is a set of *transition constraints* defined over the value of the state variables V and a copy V' of the state variables.

A state s of the system is a total assignment of the variables in their domain (Boolean or real); the set of variables includes a variable t which represents the current time $t(s)$. The set C represents constraints that the system state cannot violate, such as the circuit breaker constraint mentioned before.

Transition constraints describe what sequences of states are allowed by the model: given two states s and s', the system may evolve from state s to state s' iff all the constraints in T are satisfied by s and s' where the variables of s' are replaced by their copy in V'. This is represented by the proposition $T(s, s'[V/V'])$. If the times of s and s' are different ($t(s) \neq t(s')$) the transition is said to be continuous; otherwise it is discrete. It is implicitely assumed that every continuous transition could be split into infinitely many transitions, i.e., for all $\tau \in [t(s), t(s')]$, there exists a state s_τ such that $(t(s_\tau) = \tau) \wedge T(s, s_\tau[V/V']) \wedge T(s_\tau, s'[V/V'])$. This can generally be ensured by defining convex transition constraints.

A *system behaviour* is a sequence of states $bhv = s_0, \ldots, s_k$ that satisfies the state and transition constraints: $\forall i \in \{0, \ldots, k\}.\ C(s_i) \wedge \forall i \in \{1, \ldots, k\}.\ T(s_{i-1}, s_i[V/V'])$.

2.2 Diagnosis of Hybrid Systems

Diagnosis is the problem of determining and identifying/isolating malfunctions in a system. We assume that a subset $V_{\mathrm{f}} \subseteq V$ of the Boolean-valued state variables represents the possible faults. The faults are permanent but they can take place during the diagnostic window, i.e., some variables could evaluate to false at the beginning of the behaviour and to true at its end.

Faults must be defined explicitely; however the system behaviour may remain partially unspecified in case of faults (but also in case of non faulty behaviours). For instance, if a component binds together three variables v_1, v_2, and v_3, and if fault variable v represents the health of the component, then a state constraint may model this binding $\neg v \to f(v_1, v_2, v_3)$ where f is some constraint. If the behaviour of the component is (possibly partially) specified when faulty, then another constraint may be defined: $v \to f'(v_1, v_2, v_3)$; no such constraint is defined otherwise.

Observations The system behaviour is partially observed. For simplicity we assume that the observation is state based as opposed to event based although there is no specific difficulty associated with observed events.

An (atomic) *observation* o is a pair $\langle \tau, A \rangle$ where τ is a time and A is a partial assignment of the state variables. A behaviour $bhv = s_0, \ldots, s_k$ is consistent with the observation o iff it includes a state that is consistent with the observed values, i.e., $\exists i \in \{0, \ldots, k\}.\ t(s_i) = \tau \wedge A \subseteq s_i$.

The observations are very flexible as there is no explicit definition of observed variables, which allows to easily accomodate sensor disruptions, different frequency rates, or dynamic observability (preprocessed based methods cannot handle this flexibility). Observations can also be used to represent the initial state: the known state variables in the initial state can be treated as observations. This is very permissive as simulation-based approaches generally require to know the initial state, or at least to have a probability distribution over the (initial) set of states.

For notational simplicity we assume that the observations are precise (the exact value of the observed variable is known). This implies that the noise on the sensor has to be integrated to the model. If for instance the voltage is being monitored at some point of the system, the model will include two variables: *volt* will represent the voltage at this point and *obs_volt* will represent the observed value; a state constraint will define the possible noise, e.g., $volt - 1 < obs\_volt < volt + 1$.

Consistency-Based Diagnosis A *diagnosis* δ is a subset of faults that are consistent with the model and the observations, meaning that there exists a behaviour of the system that is consistent with the observations and such that the subset of faulty variables that evaluate to true in the final state is exactly δ. Because the number of diagnoses can be very large and many of them are unlikely, we are interested in *minimal diagnoses* which are diagnoses such that no strict proper subset is a diagnosis.

Notice that our definition of diagnosis is free of probabilities, as were the observations and their associated noise. Probabilities are useful because they allow to handle the noise on observations and the imprecisions of the model quite nicely. They also allow to rank the diagnoses and put forward the most probable ones. However realistic probabilities are very hard to obtain and their validity as well as the assumptions (for instance Gaussian or white noise) are often questionable. We can however incorporate apriori fault probabilities to rank the diagnoses [22] (the minimal-cardinality diagnoses are one such example).

Because diagnosis only asks for consistency, a significant part of the model can be left unspecified, for instance the behavior of the system in certain situations. Probabilistic methods in contrast require to be able to assign a probability distribution on the future state, even under faulty conditions.

3 Consistency-Based Diagnosis

We now present our approach to diagnosis of hybrid systems. This approach is consistency based, not only in the sense that the diagnosis is defined in terms of consistency (as opposed to probabilities), but also in the sense that the diagnosis procedure is based on operations that test the logical consistency of the model, the observations, and some assumption on the faulty state of the system.

These consistency tests are reduced to SMT problems that are solved using SMT solvers. We first present SMT and show the reduction from a consistency test to an SMT problem. Finally we show how a diagnoser can choose the tests in order to extract the diagnosis.

3.1 SAT Modulo Theory

The Satisfiability Modulo Theory (SMT) problem is a decision problem akin to the propositional satisfiability problem (SAT) with a background theory such as the Boolean, bit-vectors, arithmetic, arrays, and recursive datatypes [3]; the linear arithmetic (LA) is sufficient for this paper.

SMT problems using LA will typically involve two types of variables: the traditional SAT (Boolean) variables as well as real-valued variables. An SMT formula is defined as a collection of Boolean formulas where each literal is either a Boolean variable (or its negation) or a linear inequality over the real-valued variables. Here is an example of an SMT formula:

$$(A \lor (x > 3 * y)) \land (B \lor (x < 2 * y)) \land (\neg A \lor \neg B)$$

where A and B are the SAT variables and x and y are the real-valued variables.

The procedure for deciding SMT problems is generally two-level [10]. The background theory sentences (the inequalities in the example above) are treated as SAT variables, which leads to a SAT problem that is solved with standard SAT solving techniques. When a solution is found to the SAT problem, the consistency of the set of sentences that were assigned to *true* is tested by an external solver (depending on the strategy of the SMT solver, the consistency may actually be performed before the SAT problem has been fully solved). If this external solver finds this set inconsistent, it returns a subset of incompatible sentences that are turned into a clause (a logical constraint that forbids this set of sentences at the SAT level) that is added to the SAT problem.

3.2 Consistency Tests as SMT Problems

We now show how a (diagnosis) consistency test is reduced to an SMT problem. The reduction is similar to the one used in bounded model checking of hybrid systems with SMT.

Recall that a consistency test is defined by a model $\langle V, C, T\rangle$, a set of observations O, and an assumption over the faulty state. For simplicity we limit ourselves to the case where an assumption is a set of faults $\delta \subseteq V_{\mathrm{f}}$ assumed to have occurred (other faults have not).

The test is said to be *consistent* if the model allows for a trajectory (which we refer to as a *support*) that is consistent with the observations and the assumption. Consistency test therefore amounts to searching for such a support. We assume that this trajectory has a bounded length, i.e., that it involves at most a bounded number of states k (other approaches to diagnosis of continuous or hybrid systems often make similar assumptions, for instance that only one discrete transition is allowed between two consecutive observations). This assumption is reasonable if we assume that the set of observations is small enough, i.e., in general we assume that only the last observations will be used to diagnose the current situation.

The reduction from the test to an SMT problem is done as follows. We define a set of variables that represent the value of the state variables at every one of the k states of the support. For instance variables $v@1, v@2, \ldots$ will represent the value of the state variable v in the first, second, etc., state of the support. If v is a Boolean variable, then $v@i$ will be a Boolean variable; otherwise it will be a real-valued variable. We write $V@i$ the set of variables associated with the ith state of the support, and $\mathcal{V} = \bigcup_{i \in \{1,\ldots,k\}} V@i$ is the set of variables used in the SMT problem.

We will then define an SMT formula Φ such that the set of assignments of $\mathcal{V}$ that satisfy Φ represents exactly the set of supports to the consistency test. The test is therefore consistent iff there exists at least one support, i.e., iff the set of solutions to the SMT problem is non empty, i.e., iff the SMT problem is satisfiable. The set of assignments of $\mathcal{V}$ that represent a support are those that are consistent with i) the model, ii) the observations, and iii) the fault assumption.

We first look at the model. The following SMT formula enforces the state and transition constraints on the variables of the SMT problem:

$$\bigwedge_{i \in \{1,\ldots,k\}} C[V@i/V] \land \bigwedge_{i \in \{1,\ldots,k-1\}} T[V@i/V, V@i{+}1/V']. \tag{1}$$

For instance if the state constraints specify that a working closed circuit breaker has the same voltage at both end:

$$\mathit{closed\_cb} \land \mathit{ok\_cb} \to (\mathit{v\_in} = \mathit{v\_out}),$$

this will translate in the SMT problem as

$$\begin{gathered}(\mathit{closed\_cb}@1 \land \mathit{ok\_cb}@1 \to (\mathit{v\_in}@1 = \mathit{v\_out}@1)) \land \\ (\mathit{closed\_cb}@2 \land \mathit{ok\_cb}@2 \to (\mathit{v\_in}@2 = \mathit{v\_out}@2)) \land \\ \ldots\end{gathered}$$

Similarly if the transition constraints specify that the water level decreases in a leaking tank:

$$\mathit{leaking} \land \mathit{level} > 0 \to \mathit{level}' < \mathit{level},$$

this will translate in the SMT problem as

$$\begin{gathered}(\mathit{leaking}@1 \land \mathit{level}@1 > 0 \to \mathit{level}@2 < \mathit{level}@1) \land \\ (\mathit{leaking}@2 \land \mathit{level}@2 > 0 \to \mathit{level}@3 < \mathit{level}@2) \land \\ \ldots\end{gathered}$$

Consider now the observations O. We assume that the number i of the state s_i when each observation o_i is made is known. (The assumption can be made non-restrictive by increasing the value of k. It can also be lifted, or one could assume more complex observations such as untimed partially-ordered observations as was done in SAT-based diagnosis of discrete event systems [12].) Let i_o be this state number associated with observation o. The following SMT formula enforces the observations on the variables of the SMT problem:

$$\bigwedge_{o=\langle\tau,A\rangle\in O} (t@i_o = \tau) \wedge A[V@i_o/V]. \tag{2}$$

For instance the observation that the voltage was 24.1 at time 10.0:

$$o = \langle 10.0, (\text{volt} = 24.1)\rangle$$

this will translate in the SMT problem as

$$(t@i_o = 10.0) \wedge (\text{volt}@i_o = 24.1)\,.$$

Finally consider the fault assumption $\delta \subseteq V_\mathrm{f}$. We want the support to involve exactly all the faults in the specified set δ. Because the faults are permanent, it suffices to specify their occurrence at the end of the support. The following SMT formula enforces the fault assumption on the variables of the SMT problem:

$$\bigwedge_{f\in\delta} (f@k) \wedge \bigwedge_{f\in V_\mathrm{f}\setminus\delta} (\neg f@k)\,. \tag{3}$$

The SMT problem that we reduce the consistency test to is the conjunction of these three constraints (1), (2) and (3).

In general, and apart for the time steps associated with observations, the time of the state $s@i$ is not prespecified. Practically, $t@i$s are real-valued variables. If, for instance, the observations imply a discrete transition at time τ (which will imply $t@i = \tau = t@(i+1)$ for some i), the SMT solver will automatically deduce the value of $t@i$, Therefore this approach does not require to search explicitely for the time of the transition.

3.3 Diagnosis of Hybrid Systems as Consistency Tests

Diagnosis can be performed by asking the right consistency tests as has been acknowledged by de Kleer, Reiter, and Williams [19, 5]. Because we are interested in minimal diagnoses, we start by checking the consistency of the nominal assumption, i.e., the assumption $\delta_0 = \emptyset$. If the test is successful (the consistency holds), the system is diagnosed as non-faulty. Otherwise a fault was detected and more tests need to be performed. To this end, the original theory used *conflicts*. A diagnosis conflict is a subset of fault variables which, when assumed to be nominal, allow the test solver to infer the inconsistency of the formula. It is well-known that all minimal diagnoses are supersets of a minimal hitting set of any collection of conflicts. Therefore the classical diagnosis strategy consists in testing the consistency of such minimal hitting sets, which will either allow to prove that they are diagnoses or produce more conflicts.

SMT solvers are able to produce conflicts, and we therefore use this approach to compute diagnosis. The standard approach consists in labeling every conjunct in Equation (3), and passing these conjuncts as assumptions to the SMT solver.

Different strategies can be used to solve the diagnosis problem. We have shown [13] that it is possible to ask a completely different set of tests. The Preferred-Last strategy for instance searches for any support (hence producing a diagnosis) and then tries to improve this diagnosis by asking for a support strictly "better" than the previous one. In general, this strategy implies more consistent tests and fewer inconsistent tests than the original strategy. In our experiments, the SMT solver needs significantly more time to solve consistent tests than inconsistent ones, so we did not explore this strategy much further.

4 Experiments

We show some experimental validation of the approach we proposed. We first introduce the AdaptLite system from the 2009 DX competition [16]. We then illustrate our approach on different problem instances. Finally we demonstrate the power of this approach when the number of observations reduces.

4.1 The Adapt System

The Adapt System was introduced in the first DX competition in 2009. It features the Electrical Power System testbed in the ADAPT lab at NASA Ames Research Center. The AdaptLite variant consists of roughly 10 components (depending on how one counts them) monitored by 20 sensors (only 16 were kept in our experiments as some of them are irrelevant).

We modeled AdaptLite the way we would model the (larger) Adapt system. The latter system allows to reconfigure the system during diagnosis, meaning that the flow of electricity may change. Our model for instance contains variables that are useful only if the power flows in a direction that is impossible in AdaptLite. We end up with 129 real-valued state variables and 154 Boolean state variables.

4.2 Experiment 1

In the first set of experiments, we use a setting similar to the original competition. We assume that every sensor communicates its current reading at a frequency of 2Hz, i.e., twice every second. We diagnose windows of 10 consecutive observations. We limit ourselves to minimal cardinality diagnoses, as the number of minimal diagnoses can get absurdly high.

The experiments were performed on an Intel i5-2520M 2.5GHz with 3.75GiB and running GNU/Linux Mint 16 "petra". The diagnoser was implemented on Java 7 using SMT solver Z3 version 4.3.1. (Experiments with cvc3 gave similar results, bearing in mind that the runtime can be very volatile. The new version

Prob. instance	Time (s)	Card	# δ
1	3.428	0	1
2	5.314	1	2
3	5.298	1	1
4	3.476	1	1
5	6.477	2	4

Table 1. Example runtime: computation time, cardinality of the diagnoses, number of minimal-cardinality diagnoses.

cvc4 of the cvc family does not implement conflict generation yet.)

The results are summarized on Table 1. Most of the runtime is used by the SMT solver. As it turns out, SMT problems that are consistent are the most time-consuming. We conjecture that this is due to the expansive procedure of verifying consistency of a set of linear inequalities and that any reduction in the number of real-valued variables for instance would speed up the process.

Traditional methods can treat this particular system as a continuous one and still remain accurate. Their runtime is much better than our method; this is mainly because the system is very observable and nearly every state variable in the system is observed (although through noise). The next set of experiments is meant to show that our approach does not suffer for a reduction in observability.

4.3 Experiment 2: Reducing the Observations

For this set of experiments we reused the problem instances from the previous subsection but deleted some values observed at random. We run the diagnoser on these truncated observations. The runtime is reported on Figure 1; the x-axis shows the percentage of observations that were deleted. The line stops when the returned diagnoses differ from the original diagnosis: in a problem with no observation for instance the single minimal diagnosis is $\delta = \emptyset$, and comparing such problems is not relevant.

Traditional approaches do not cope well with this type of problems: we discuss this point in more details in the next section.

As we can see, the runtime of the diagnosis approach is very uncertain but is not clearly correlated to the number of observations. Most existing benchmarks on continuous systems assume a very large number of sensors as well as a synchronized flow of observations; under these assumptions, our approach is at a disadvantage, but outside them, it becomes much more competitive.

5 Discussion

We presented a consistency-based approach to diagnosis of hybrid systems. We now conclude by presenting the benefits of our approach as well as possible future works to address its current shortcomings.

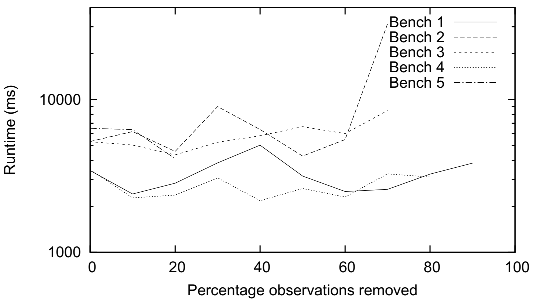

Figure 1. Evolution of runtime when observations get removed.

5.1 Benefits of Our Approach

The approach presented in this paper is the first one that handles both the discrete and continuous aspects of hybrid systems. Previous approaches either have to concentrate on one aspect (previous work on diagnosis of hybrid system with SMT [8] only considered snapshots) or look at these aspects separately [4].

Continuous or switching systems are often diagnosed by identifying patterns of faults, such as possible conflicts and indicators [18]. This approach requires preprocessing and is very rigid with respect to observability. Consider for instance a system where the voltage and intensity are linked through a constraint when the system is not faulty; imagine that three successive observations are available, a voltage reading, an intensity reading, and a voltage reading similar to the first one; if the readings contradict the constraint, approaches based on preprocessed patterns will most likely not be able to detect the fault.

Another common approach to diagnose continuous systems is through simulation, generally coupled with probabilistic reasoning. This type of approach requires a predictive model, i.e., a model that can either predict how the system state will evolve or assign a probability distribution over this evolution. Such a model may not be available, and may require to ignore entirely parts of the system whose behavior is only partly known. Furthermore simulating a system requires being able to maintain the belief state (the set of states that the system is believed to be in) or the probability distribution of this belief state. This is very hard or even impossible, especially because of the interaction between the continuous and discrete variables, and approximate approaches—when applicable—are quite involved and are subject to overapproximation.

Finally we would like to emphasize a last benefit of our approach. Pattern-based approaches (such as indicators or possible conflicts) are good at explaining why an assumption is not valid: they essentially prove that such assumptions lead to a contradiction. Simulation-based approaches are good at explaining why an assumption seems valid: the simulation actually produces supports for these assumptions. Our approach exhibits both characteristics, being both able to justify why a hypothesis should be rejected and able to provide supports for its diagnoses.

5.2 Disadvantages and Future Works

At this stage the main issue of SMT-based diagnosis of hybrid systems is the computational cost. We believe however that the scope for improvement here is quite large. Experience in different domains but using similar test-based approaches shows that simple changes can better the runtime dramatically: changing the search strategy for diagnosis and using an incremental SMT solver reduced the runtime by up to one order of magnitude compared to our first works [11]. Other improvements include: pruning irrelevant variables [3], using dedicated SMT solvers [21, 9], improving the reduction to SMT [1, 20].

Whilst the current approach requires no preprocessing, this does not mean that preprocessing cannot be used to speed up the diagnosis. Possible conflicts can be used in the first phase of diagnosis to already narrow down the search. SMT solvers are then used mostly as validation. Amongst other possible preprocessing techniques, diagnosability or similar techniques can be used to determine how the problem can be decomposed; for instance diagnosability might allow to determine that reasoning about a given subset of the network is sufficient to accurately diagnose a fault.

An issue of the approach presented here is that it is only applicable on “short” windows. In an online context, this means that only the last observations can be taken into account, which may harm the precision of diagnoser. Simulation-based approaches are not concerned by this issue because the current belief state carries all the relevant information about past observations. One way to address this issue would be to discover facts about the current window that can be carried over to the next window. Such facts should be compact (otherwise, it would be equivalent to computing the belief state), for instance: *the circuit breaker is known to be open*, or *either one of the two bulbs in the system is broken*. This information would help both the precision of diagnosis and the computation time.

Finally of interest would be to consider nonlinear systems as proposed by Eggers et al. [7]. The current method handles such systems by overapproximating the state and behavior space (in the worst case, just ignoring the nonlinear constraints) at the cost of precision. SMT solvers are not limited to linear arithmetic but to what extend they can handle more complex constraints is uncertain.

REFERENCES

[1] A. Anbulagan and A. Grastien, ‘Importance of variables semantic in CNF encoding of cardinality constraints’, in *Eighth Symposium on Abstraction, Reformulation and Approximation (SARA-09)*, (2009).

[2] G. Audemard, M. Bozzano, A. Cimatti, and R. Sebastiani, ‘Verifying industrial hybrid systems with MathSAT’, in *Second International Workshop on Bounded Model Checking (BMC-04)*, pp. 17–32, (2004).

[3] F. Balarin and A. Sangiovanni-Vincentelli, ‘An iterative approach to language containment’, in *Fifth International Conference on Computer-Aided Verification (CAV-93)*, (1993).

[4] M. Bayoudh, L. Travé-Massuyès, and X. Olive, ‘Coupling continuous and discrete event system techniques for hybrid system diagnosability analysis’, in *18th European Conference on Artificial Intelligence (ECAI-08)*, (2008).

[5] J. de Kleer and B. Williams, ‘Diagnosing multiple faults’, *Artificial Intelligence (AIJ)*, **32**, 97–130, (1987).

[6] L. de Moura, B. Dutertre, and N. Shankar, ‘A tutorial on satisfiability modulo theories’, in *19th International Conference on Computer-Aided Verification (CAV-07)*, pp. 20–36, (2007).

[7] A. Eggers, N. Ramdani, N. Nedialkov, and M. Fränzle, ‘Set-membership estimation of hybrid systems via SAT modulo ODE’, in *16th IFAC Symposium on System Identification (SYSID-12)*, pp. 440–445, (2012).

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

[9] M. Fränzle and C. Herde, ‘HySAT: an efficient proof engine for bounded model checking of hybrid systems’, *Formal Methods in System Design (FMSD)*, **30**(3), 179–198, (2007).

[10] H. Ganzinger, G. Hagen, R. Nieuwenhuis, Al. Oliveras, and C. Tinelli, ‘DPLL(T): fast decision procedures’, in *16th International Conference on Computer-Aided Verification (CAV-04)*, pp. 175–188, (2004).

[11] A. Grastien, ‘Diagnosis of hybrid systems by consistency testing’, in *24th International Workshop on Principles of Diagnosis (DX-13)*, pp. 9–14, (2013).

[12] A. Grastien and A. Anbulagan, ‘Diagnosis of discrete event systems using satisfiability algorithms: a theoretical and empirical study’, *IEEE Transactions on Automatic Control (TAC)*, **58**(12), 3070–3083, (2013).

[13] 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)*, pp. 60–67, (2011).

[14] A. Grastien, P. Haslum, and S. Thiébaux, ‘Conflict-based diagnosis of discrete event systems: theory and practice’, in *13th International Conference on the Principles of Knowledge Representation and Reasoning (KR-12)*, (2012).

[15] T. King and C. Barrett, ‘Exploring and categorizing error spaces using BMC and SMT’, in *Ninth International Workshop on Satisfiability Modulo Theories (SMT-11)*, (2011).

[16] 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)*, pp. 383–396, (2009).

[17] S. Narasimhan and G. Biswas, ‘Model-based diagnosis of hybrid systems’, *IEEE Transactions on Systems, Man, and Cybernetics (TSMC)*, **37**(3), 348–361, (2007).

[18] B. Pulido and C. Alonso González, ‘Possible conflicts: a compilation technique for consistency-based diagnosis’, *IEEE Transactions on Systems, Man, and Cybernetics (TSMC)*, **34**(5), 2192–2206, (2004).

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

[20] J. Rintanen, ‘Compact representation of sets of binary constraints’, in *17th European Conference on Artificial Intelligence (ECAI-06)*, pp. 143–147, (2006).

[21] J. Rintanen, ‘Planning with specialized SAT solvers’, in *25th Conference on Artificial Intelligence (AAAI-11)*, (2011).

[22] B. Williams and R. Ragno, ‘Conflict-directed A* and its role in model-based embedded systems’, *Discrete Applied Mathematics (DAM)*, **155**, 1562–1595, (2007).