Auton Agent Multi-Agent Syst (2009) 19:30–52
DOI 10.1007/s10458-007-9017-6

Models and methods for plan diagnosis

Nico Roos · Cees Witteveen

Published online: 10 October 2007

Abstract We consider a model-based diagnosis approach to the diagnosis of plans. Here, a plan performed by some agent(s) is considered as a system to be diagnosed. We introduce a simple formal model of plans and plan execution where it is assumed that the execution of a plan can be monitored by making partial observations of plan states. These observed states are used to compare them with states predicted based on (normal) plan execution. Deviations between observed and predicted states can be explained by qualifying some plan steps in the plan as behaving abnormally. A diagnosis is a subset of plan steps qualified as abnormal that can be used to restore the compatibility between the predicted and the observed partial state. Besides minimum and subset minimal diagnoses, we argue that in plan-based diagnosis maximum informative diagnoses should be considered as preferred diagnoses, too. The latter ones are diagnoses that make the strongest predictions with respect to partial states to be observed in the future. We show that in contrast to minimum diagnoses, finding a (subset minimal) maximum informative diagnosis can be achieved in polynomial time. Finally, we show how these diagnoses can be found efficiently if the plan is distributed over a number of agents.

Keywords Planning · Diagnosis · Complexity · Multi-agents

N. Roos
Department of Computer Science, Universiteit Maastricht, P.O. Box 616, 6200 MD Maastricht, The Netherlands
e-mail: roos@cs.unimaas.nl

C. Witteveen (✉)
Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Netherlands
e-mail: C.Witteveen@tudelft.nl

1 Introduction

With a growing complexity of plans, the possibility that something goes wrong during their execution increases correspondingly. No wonder then that attention has to be paid to the development of *robust* plans. One way to enhance robustness is to perform plan diagnosis in order to identify the causes of failures, to predict future failures and, if possible, to prevent failures from occurring. Since there are a huge number of potential factors that might prevent correct plan execution, it is not surprising that current approaches to plan diagnosis are rather diverse. For example, a changing *environment* might be such an important disturbing factor, preventing some parts of the plan to be executed by changing the preconditions of some instances of actions occurring in the plan. Another important source of plan failures could be attributed to the *agent(s)* controlling the actions prescribed in the plan by being unable to perform some of the actions required or accidently changing some of the preconditions of actions. In a broader, multi-agent perspective, one could even concentrate on *incompatibilities* between different agents involved in the execution of a joint plan as a major factor that could prevent parts of a joint plan from being executed correctly.

In this paper we want to specify a general framework for plan diagnosis where, in principle, the above mentioned aspects of plan diagnosis could be dealt with and the computational properties of finding suitable plan diagnoses can be investigated. In particular, we hope to find suitable diagnostic tasks that can be executed efficiently.

In developing such a framework it seems unavoidable to concentrate on some aspects of plan diagnosis and to (temporarily) neglect others. In this paper, we concentrate on *internal* failure sources and leave external failure sources such as the environment, failures of executing agents as in [2], or incompatibilities between agents as in [14,15] for future research. In particular, we confine ourselves to the identification of failing *actions* as the only source of plan failure. Our main motivation for this restriction is that if the plan is correctly specified, errors in the plan execution process become manifest in the incorrect behavior of one or more instances of actions.[1] Whether or not we should be satisfied with the mere identification of one or more of such failing actions, a diagnostic process that identifies a set of actions that can be shown to be responsible for the abnormalities observed seems to be a useful analysis on its own. In a multi-agent planning systems, for example, identification of such failing actions can be used to identify incompatibilities between plans, to identify failing agents responsible for executing plans or to identify incompatibilities between agents involved in the plans. In the conclusion section we elaborate on the potential extensions of the framework to deal with these questions.

Although the specification of a general framework for plan diagnosis is our first and main objective, we also want to identify suitable (preference) criteria for choosing diagnoses in plan based diagnosis, to characterize the type of most preferred diagnosis in terms of these criteria and to identify the computational complexity of finding these diagnoses. In Model-Based Diagnosis (MBD), for example, size- and subset-minimal diagnoses are preferred. The underlying preference criterion is based on an intuitive acceptable assumption concerning the default behavior of the system: we assume that for every component of the system it is more likely that it behaves normally than abnormally. If we would prefer the most likely diagnoses, it is not difficult to show that minimum diagnoses will be considered as the most preferred ones. It is well-known that in general finding such minimum diagnoses is hard.

[1] Of course, some of these actions might not be specified in the plan.

In our framework, we investigate another intuitive acceptable assumption concerning the default behavior of a system and a preference relation based on it that can be used to characterize another set of preferred diagnoses. This latter assumption states that it is more likely that a faulty action will produce (directly or indirectly) incorrect results than correct results. Based on this principle, we prefer diagnoses that are more precise (more *informative*) in their predictions. This gives rise to *maximum informative diagnoses* as preferred diagnoses. Hoping to find efficiently computable diagnostic tasks, we want to investigate the properties of this diagnostic concept and the computational complexity of finding such diagnoses. This last computational goal is also the reason for temporarily limiting our approach to deterministic planning. This implies that we do not address approaches focussing on *non-deterministic* planning like [1,8,12]. One justification is that in nondeterministic contexts even the simplest diagnostic tasks are intractable, such as finding a discrepancy or recognizing a diagnosis for such discrepancies.

As our third objective, we would like to investigate the merits of this framework for *multi-agent* plan diagnosis. What we have in mind are situations where the plan is distributed over a set of agents who are communicating with each other. Together the agents are capable of predicting what will happen if the plan would be executed given some initial situation. If something is wrong, their predictions will not correspond to observations. We want to investigate the suitability of the above mentioned preferred diagnostic concepts in such a distributed environment.

1.1 Results

The results obtained in this paper are threefold. First of all, we present a formal framework for plan diagnosis that enables us to define exactly how observations of a plan in execution can be used to derive an arbitrary plan diagnosis. We show that establishing a plan diagnosis comes down to finding a subset of actions in a plan such that if these actions are qualified as abnormal, the observed plan states are compatible with predicted plan states.

Second, with respect to preferred diagnoses, we point out that, like in MBD, in our framework minimum diagnoses are also hard to find. But quite surprisingly, maximum informative diagnoses can be found very efficiently. Moreover, every subset-maximal informative diagnosis also turns out to be a size maximal, i.e., a maximum, informative diagnosis. Finally, combining the two preferred types of diagnoses, we show that maximal informative diagnoses that are also subset-minimal diagnoses can be found in polynomial time, while on the other hand the intersection between the set of minimum diagnoses and the set of maximal informative diagnoses is sometimes empty.

Third, we extend the plan diagnosis framework to the multi-agent case where the plan is distributed over several agents. We show that in such a distributed setting subset minimal maximum diagnoses still can be computed very efficiently by a distributed label setting and label propagating algorithm making them a suitable candidate for fast diagnosis in a multi-agent setting.

1.2 Organization

In Sect. 2, we first discuss some related approaches to plan diagnosis. In Sect. 3 we introduce a simple formal framework for representing states, actions and plans. Then, in Sect. 4, we introduce the main concepts of plan-based diagnosis by introducing qualifications as the basis for characterizing diagnoses and we define the derivability relation between two plan states. In Sect. 5, we formally define the notion of plan diagnosis and we introduce the

concept of a maximum informative diagnosis as a preferred diagnosis. In Sect. 6 we discuss some complexity results for preferred diagnoses and we present an efficient algorithm to find minimal maximum informative diagnoses. In Sect. 7, we extend our framework to diagnosis in a distributed environment and we discuss an efficient distributed algorithm that can be used to find minimal maximum informative diagnoses. Section 8 concludes this paper with a brief outlook to future research. Some proofs are given in the appendix.

2 Related research

In this section we briefly discuss some other approaches to plan diagnosis we already mentioned above.

Birnbaum et al. [2] apply MBD to *planning agents* relating health states of agents to *outcomes* of their planning activities. They do not take into account abnormalities that can be attributed to actions in a plan as a separate source of errors. In contrast to their approach, in this paper we do not take into account abnormalities of the executing agents, but exclusively focus upon the detection of abnormal actions in the plan. As we already remarked above, we feel that such an approach focusing upon actions as the immediate factors underlying abnormal plan behavior should precede more elaborate failure analyses.

Another approach that directly applies model-based diagnosis to plan execution has been proposed in De Jonge et al. [13]. There, the authors focus on agents each having an individual plan, and where conflicts between these plans may arise (e.g., if they require the same resource). Diagnosis is applied to determine those factors that are accountable for *future* conflicts.

Kalech and Kaminka [14,15] apply *social diagnosis* in order to find the cause of an anomalous plan execution. They consider hierarchical plans consisting of so-called *behaviors*. Such plans do not prescribe a (partial) execution order on a set of actions. Instead, based on its observations and beliefs, each agent chooses the appropriate behavior to be executed. Each behavior in turn may consist of primitive actions to be executed, or of a set of other behaviors to choose from. Social diagnosis then addresses the issue of determining what went wrong in the joint execution of such a plan by identifying the disagreeing agents and the causes for their selection of incompatible behaviors (e.g., belief disagreement, communication errors). This approach might complement our approach when conflicts not only arise as the consequence of faulty actions, but also as the consequence of different selections of sub-plans in a joint plan.

Lesser et al. [3,11] also apply diagnosis to (multi-agent) plans. Their research concentrates on the use of a *causal model* that can help an agent to refine its initial diagnosis of a failing *component* (called a *task*) of a plan. As a consequence of using such a causal model, the agent would be able to generate a new, situation-specific, plan that is better suited to pursue its goal. While their approach in its ultimate intentions (establishing anomalies in order to find a suitable plan repair) comes close to our approach, their approach to diagnosis concentrates on specifying the exact causes of the failure of one single *component* (task) of a plan. Diagnosis is based on observations of a component without taking into account the consequences of failures of such a component with respect to the remaining plan. In our approach, instead, we are interested in applying MBD-inspired methods to *detect* plan failures. Such failures are based on observations during plan execution and may concern individual components of the plan. Furthermore, we do not only concentrate on identifying failing components themselves, but also on the consequences of these failures for the future execution of plan elements.

3 Preliminaries

We consider plan-based diagnosis as a simple extension of the model-based diagnosis (MBD) approach [4,5,17], where the model is not a description of an underlying physical system but a *plan* of one or more agents. To keep this model simple and general, we will keep the plan representation details minimal.[2] The main features, however, of planning formalisms like STRIPS [9] are covered in this framework.

3.1 States

By executing plans we change a part of the world. Therefore, before we discuss plans, we need to introduce a simple state-based view on the world. We assume that for the planning problem at hand, the world can be described by a set $Var = \{v_1, v_2, \ldots, v_n\}$ of variables and their respective *value domains* D_i. A *complete state of the world* σ then is a value assignment $\sigma : Var \rightarrow \bigcup_{i=1}^{n} D_i$ to the variables. Slightly abusing terminology, we simply denote a complete state by an n-tuple $\sigma = (\sigma(v_1), \ldots, \sigma(v_n)) \in D_1 \times D_2 \times \cdots \times D_n$. A *partial state* is an element $\pi \in D_{i_1} \times D_{i_2} \times \cdots \times D_{i_k}$, where $1 \leq k \leq n$ and $1 \leq i_1 < \cdots < i_k \leq n$. We use $Var(\pi)$ to denote the set of variables $\{v_{i_1}, v_{i_2}, \ldots, v_{i_k}\} \subseteq Var$ specified in π. The value $\sigma(v_j)$ of variable $v_j \in Var(\pi)$ will be denoted by $\pi(v_j)$. The value of a variable $v_j \in Var$ not occurring in a partial state π is said to be *undefined* (or *unpredictable*) in π, denoted by $\perp$. Including $\perp$ in every value domain D_i allows us to consider every partial state π as an element of $D_1 \times D_2 \times \cdots \times D_n$.

Partial states π can be ordered with respect to their information content: Given values d and d', we say that d' is at least as informative as d, abbreviated as $d \leq d'$, iff $d = \perp$ or $d = d'$.[3] The containment relation $\sqsubseteq$ between partial states is the point-wise extension of $\leq$: π is said to be contained in π', denoted by $\pi \sqsubseteq \pi'$, iff $\forall v \in Var[\pi(v) \leq \pi'(v)]$.

An important notion in plan diagnosis is the notion of *compatibility* between partial states. Intuitively, two states π and π' are said to be compatible if they could characterize the same state of the world, that is, if there exists a complete state σ such that $\pi \sqsubseteq \sigma$ and $\pi' \sqsubseteq \sigma$. Equivalently, this compatibility relation can also be expressed without making reference to such a state σ, since the existence of such a state σ clearly implies that for every $v \in Var$, either $\pi(v) = \pi'(v)$ or at least one of the values $\pi(v)$ and $\pi'(v)$ is undefined:

Definition 1 (compatibility relation) Two partial states π and π' are said to be compatible, denoted by $\pi \approx \pi'$, if $\forall v \in Var[\, \pi(v) \leq \pi'(v)$ *or* $\pi'(v) \leq \pi(v)\,]$.

Finally, we need the concept of *information fusion*. If two partial states π_1 and π_2 are compatible, their information content can be *fused* to obtain a new partial state $\pi = \pi_1 \sqcup \pi_2$ that contains them both: $\pi = \pi_1 \sqcup \pi_2$ holds iff $\forall v \in Var[\pi(v) = \max_{\leq}\{\pi_1(v), \pi_2(v)\}]$. Note that this new fused state π combines the information about a possible common state σ the two partial states π_1 and π_2 are characterizing.

3.2 Actions, plan operators and plan steps

In the preceding sections we used to term "actions" in a rather informal way. From now on, we make a distinction between *actions*, *plan operators* and *plan steps*. First of all, an *action* refers to an activity that results in some change of the (current) state of the world, such as

[2] For an earlier version of this model, see [18].

[3] In domain theory [10], the domain of variables is known as a flat domain.

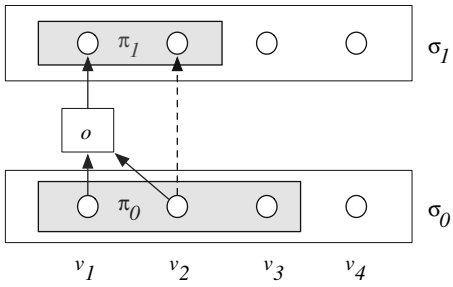

Fig. 1 Plan operators, states and partial states

loading a vehicle or assembling components. A *plan operator* refers to a description of such an action in a plan. More exactly, a plan operator o is a function mapping partial states to partial states by replacing the values of a subset $V_o \subseteq Var$ by other values (dependent upon the values of another set V_o' of variables. Hence, every plan operator o can be modeled as a (partial) function $f_o : D_{i_1} \times \cdots \times D_{i_k} \to D_{j_1} \times \cdots \times D_{j_l}$, where $1 \leq i_1 < \cdots < i_k \leq n$ and $1 \leq j_1 < \cdots < j_l \leq n$. The set of variables whose value domains occur in $dom(f_o)$ will be denoted by $dom_{Var}(o) = \{v_{i_1}, \ldots, v_{i_k}\}$. Likewise, the set of variables whose value domains occur in $ran(f_o)$ will be denoted by $ran_{Var}(o) = \{v_{j_1}, \ldots, v_{j_l}\}$. Often it will be convenient to extend the domain of such a function f_o to apply f_o on a partial state π such that $Var(\pi) \supseteq dom_{Var}(o)$. The result $f_o(\pi)$ then is defined as follows: Let π' be the restriction[4] of π to $dom_{Var}(o)$. Then $f_o(\pi)(v) = f_o(\pi')(v)$ if $v \in ran_{Var}(o)$ and $f_o(\pi)(v) = \pi(v)$, otherwise.

This functional specification f_o constitutes the *normal* behavior of the plan operator o, also denoted by f_o^{nor}.

Example 1 Suppose there is a courier who has to pickup a package at some person's home. It is known that this person also has a car and a dog. The car is at the same address, but the current location of the dog is unknown. Here, the world can be described by four variables: v_1 (location of package), v_2 (location of person's home), v_3 (location of the car), v_4 (location of the dog). Figure 1 depicts two states σ_0 and σ_1 (the white boxes) of this world each characterized by the values of the four variables v_1, v_2, v_3 and v_4. The partial states π_0 and π_1 (the gray boxes) characterize a subset of variables in a (complete) state. The pickup of the package can be modeled as a state change by a plan operator o modeling the pickup. The domain of the plan operator o is the subset $\{v_1, v_2\}$, i.e., it requires the values of the location of the package and the person's home address, denoted by the arrows pointing to o. The result of the pickup is a change in the location of the package. Therefore, the range of o is the subset $\{v_1\}$, which is denoted by the arrow pointing from o. Finally, the dashed arrow denotes that the value of variable v_2 is not changed by operator causing the state change. □

A plan operator o may be used at several places in a plan. A specific occurrence of o is called a *plan step* mapping a specific partial state into another specific partial state. A plan step s as an occurrence of o then describes a specific *function application* of the function f_o specified by the operator o at a specific place in the plan. Therefore, given a set $\mathcal{O}$ of plan operators, we consider a set $S = inst(\mathcal{O})$ of *instances* of plan operators in $\mathcal{O}$, called the set of plan steps. A plan step will be denoted by a small roman letter s_i. The plan operator o

[4] More precisely, π' is the unique partial state such that $\pi' \sqsubseteq \pi$ and $Var(\pi') = dom_{Var}(o)$.

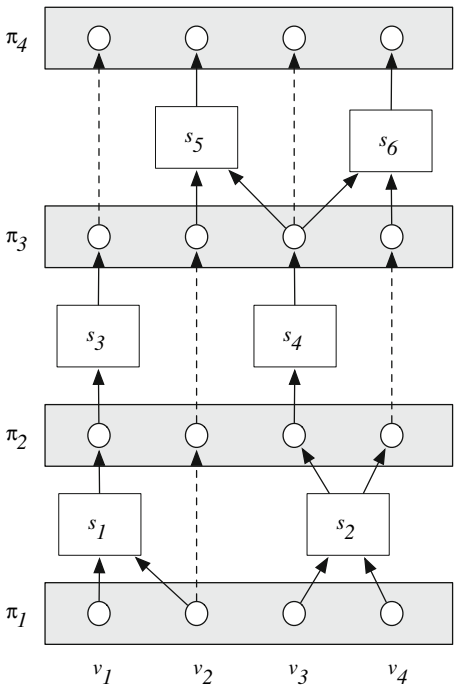

Fig. 2 Plans and plan steps. Each state characterizes the values of four variables v_1, v_2, v_3 and v_4. States are changed by application of plan steps s_i for $i = 1, 2, \ldots, 6$

the instance s_i was instantiated from is denoted by $o(s_i)$. If $o(s_i) = o$, the instance s_i is said to be of *type* o. In that case, the function associated with s_i is $f_{s_i} = f_o$. Likewise, $ran_{Var}(s) = ran_{Var}(o)$ and $dom_{Var}(s) = dom_{Var}(o)$.

3.3 Plans and plan execution

A plan is a tuple $P = \langle \mathcal{O}, S, < \rangle$ where $S \subseteq inst(\mathcal{O})$ is a set of plan steps occurring in $\mathcal{O}$ and $(S, <)$ is a partial order (cf. [6]). The partial order relation $<$ specifies an *execution relation* between plan steps: for each $s \in S$ it holds that s is executed immediately after all plan steps s' such that $s' < s$ have been finished. We will denote the *transitive reduction* of $<$ by $\ll$, i.e., $\ll$ is the smallest subrelation of $<$ such that the transitive closure $\ll^+$ of $\ll$ equals $<$.

Example 2 Figure 2 gives an illustration of a plan. Arrows relate the objects a plan step uses as inputs and the objects it produces as its outputs to the plan step itself. In this plan, the direct execution relation is specified as $s_1 \ll s_3$, $s_2 \ll s_4$, $s_4 \ll s_5$ and $s_4 \ll s_6$. □

Without loss of generality,[5] we assume that every plan step $s \in S$ takes a unit of time to execute and the execution of the first plan step starts at time $t = 0$. Using this assumption and the definition of the execution relation $<$, the time t at which a plan step s will be executed is uniquely determined: Let $depth_P(s)$ be the depth of plan step s in plan F. Here, as usual, $depth_P(s) = 0$ if $\{s' \in S \mid s' \ll s\} = \varnothing$ and $depth_P(s) = 1 + max\{depth_P(s') \mid s' \ll s\}$, else. If the context is clear, we omit the subscript P. Then the time t_s at which the plan step

[5] That is, if we still assume that the duration of executing each plan step is known. Otherwise we would have to deal with non-deterministic aspects.

s is executed is $t_s = depth_P(s)$ and s will be completed at time $t_s + 1$. Let P_t denote the set of plan steps s with $depth_P(s) = t$.

Example 3 Consider again Fig. 2. In this plan, the depth of s_1 and s_2 is 0, the depth of s_3 and s_4 is 1, and the depth of s_5 and s_6 is 2. Therefore, $P_0 = \{s_1, s_2\}$, $P_1 = \{s_3, s_4\}$ and $P_2 = \{s_5, s_6\}$. □

Given the plan state σ at some time t and the set P_t of plan steps to be executed at time t we want to be sure that the next state σ' at time $t + 1$ is uniquely defined. There are two ways in which the predictability of the next state might be affected by interacting plan steps: First of all, the output of plan step $s \in P_t$ might affect the input of another plan step $s' \in P_t$. This might happen whenever there are plan steps $s, s' \in P_t$ such that $ran_{Var}(s) \cap dom_{Var}(s') \neq \varnothing$. As a consequence, the result of executing s' is not uniquely defined. Secondly, if P_t contains two plan steps s and s' with overlapping ranges, i.e., if $ran_{Var}(s) \cap ran_{Var}(s') \neq \varnothing$, the final result of a variable v occurring in this intersection is not uniquely defined in σ'. To guarantee uniquely defined outcomes of plan execution, we therefore assume that such plan steps do not occur together in an execution set P_t:

Determinism Let $P = \langle \mathcal{O}, S, < \rangle$ be a plan. Then for every $s \neq s' \in S$, $(ran_{Var}(s) \cap ran_{Var}(s') \neq \varnothing$ or $ran_{Var}(s) \cap dom_{Var}(s') \neq \varnothing)$ implies $depth_P(s) \neq depth_P(s')$.

It is not difficult to see (and can be easily proven using the derivability relations to be discussed) that Determinism guarantees that a future plan state can be defined uniquely given a plan and a currently uniquely defined plan state.

4 Qualifications, predictions and derivability relations

4.1 Plan qualifications

The correct execution of a plan step may fail either because of an inherent malfunctioning or because of a malfunctioning of an agent responsible for executing the action, or because of unknown external circumstances. In all these cases we would like to model the effects of executing such failing plan steps. Therefore, we introduce a set of *health modes* H_s for each plan step s. This set H_s contains at least the normal mode nor, the mode ab indicating the most general abnormal behavior, and possibly several other specific fault modes. Let $o = o(s)$ be the plan operator of which s is an instance. The most general abnormal behavior of plan step s is specified by the function f_s^{ab}, where $f_s^{ab}(d_{i_1}, d_{i_2}, \ldots, d_{i_k}) = (\bot, \bot, \ldots, \bot)$ for every partial state $(d_{i_1}, d_{i_2}, \ldots, d_{i_k}) \in dom(f_o)$. To simplify the discussion, in the sequel we distinguish only the health modes nor and ab.

Let us assume, for the moment, that each plan step can be viewed as an independent component of a plan. To each plan step s then a health mode $h_s \in \{nor, ab\}$ can be assigned and the result is called a *qualified* plan. In establishing which part of the plan fails, we are only interested in those plan steps qualified as abnormal. Therefore, we define a qualified version P_Q of a plan $P = \langle \mathcal{O}, S, < \rangle$ as a tuple $P_Q = \langle \mathcal{O}, S, <, Q \rangle$, where $Q \subseteq S$ is the subset of plan steps qualified as abnormal (and therefore, $S - Q$ is the subset of plan steps qualified as normal).

Since a qualification Q corresponds to assigning the health mode ab to every plan step in Q and since $f_s^{ab}(d_{i_1}, d_{i_2}, \ldots, d_{i_k}) = (\bot, \bot, \ldots, \bot)$ the results of anomalously behaving plan steps are unpredictable. Note that a "normal" plan P corresponds to the qualified plan $P_\varnothing$ and that in our context "undefined" is considered to be equivalent to "unpredictable".

4.2 Predicting results of executing plan steps

In order to find out whether or not some plan steps might be failing, we have to be able to predict the effect of executing plan steps on some given partial state π_0. In general, executing a (qualified) plan P on a given initial state π_0 will induce a sequence $(\pi_0, \pi_1, \ldots, , \pi_k)$ of states, where π_{t+1} is derived from π_t by applying the set of plan steps P_t to σ_t. We call a partial state π at a given time t a *timed state*, denoted by (π, t). To predict the final result π_k starting from π_0, we will first define a direct derivability relation enabling us to predict a next timed state $(\pi', t+1)$ using the current timed state (π, t). This direct derivability relation takes into account the information available in the partial state π, the set P_t of plan steps s that have to be executed at time t and the qualification of each of these plan steps. The intuitive idea behind the definition of this direct derivability relation is very simple: we are only capable of making an exact prediction about the results of executing a plan step $s \in P_t$ if (i) we know for each variable in the domain $dom_{Var}(s)$ its value[6] and (ii) s is not qualified as abnormal.

This idea leads to the following description of the relation between the predicted timed state $(\pi', t+1)$ and the current state (π, t). For every variable $v \in Var$, we distinguish the following cases:

1. v does not occur in the range of a plan step $s \in P_t$. We predict its value $\pi'(v)$ to be the same as its value in π, i.e., $\pi(v) = \pi'(v)$. This is a simple consequence of our assumption to consider plan step executions as the only source of changes to be considered.
2. v occurs in the range of some plan step $s \in P_t$.
   Thanks to Determinism, this plan step s is uniquely defined. We now distinguish the following subcases:
   (a) s is qualified as abnormal.
   Then $\pi'(v) = \perp$, since every variable in the range of f_s^{ab} is undefined.
   (b) s is qualified as normal and $dom_{Var}(s) \subseteq Var(\pi)$;
   In this case, $\pi'(v) = f_s^{nor}(\pi)(v)$. This means that if the plan step s does not fail and we completely know all values of variables occurring in its domain, we are able to predict the effect of the plan step by considering the effect of the function application f_s^{nor} on π. In this case we say that s is *fully enabled* in P_t;
   (c) s is qualified as normal, but for some $v' \in dom_{Var}(s)$, $\pi(v') = \perp$;
   Since we don't have exact information about at least one value of a variable in the domain of s, we are not able to predict the consequences of executing s in π. Therefore, we consider $\pi'(v') = \perp$ for all $v' \in ran_{Var}(s)$. Hence, since $v \in ran_{Var}(s)$, $\pi'(v) = \perp$.

This direct derivability relation can be defined in a more formal and concise way as follows:

Definition 2 A timed state $(\pi', t+1)$ is (directly) generated by execution of the Q-qualified plan P_Q from (π, t), abbreviated by $(\pi, t) \rightarrow_{Q;P} (\pi', t+1)$, iff for every $v \in Var$ the following conditions hold:

1. if $v \notin \bigcup_{s \in P_t} ran_{Var}(s)$ then $\pi'(v) = \pi(v)$;
2. if $v \in \bigcup_{s \in P_t} ran_{Var}(s)$, let s be the unique plan step such that $v \in ran_{Var}(s)$.
   (a) If s is fully enabled in P_t and $s \notin Q$ then $\pi'(v) = f_s^{nor}(\pi)(v)$,

[6] Considering e.g., f_s as a blackbox and a huge size of the domains of the variables involved, it is not feasible to predict the effect of f_s on all possible values of a variable v whose value is unknown.

(b) else $\pi'(v) = \perp$.

It is easy to see that, thanks to Determinism, this direct derivability relation $\rightarrow_{Q;P}$ is well-defined and deterministic:

Proposition 1 *Let P_Q be a qualified plan and let (π, t) a timed state. Then $(\pi, t) \rightarrow_{Q;P} (\pi', t+1)$ and $(\pi, t) \rightarrow_{Q;P} (\pi'', t+1)$ implies $\pi'' = \pi'$.*

Proof Suppose that the conditions stated in the proposition do hold and that, on the contrary, there exists some $v \in Var$ such that $\pi''(v) \neq \pi'(v)$. According to Definition 2, this can only occur if there exist at least two plan steps $s, s' \in P_{t'}$ such that $v \in ran_{Var}(s) \cap ran_{Var}(s')$ or $v \in ran_{Var}(s) \cap dom_{Var}(s')$. Hence, by Determinism, we have $depth_P(s) \neq depth_P(s')$. But that immediately implies that s and s' cannot both occur in $P_{t'}$; contradiction. Therefore, $\pi'' = \pi'$. □

We extend this direct derivability relation to a general derivability relation in a straightforward way:

Definition 3 For arbitrary values of $t \leq t'$ we say that (π', t') is (directly or indirectly) generated by execution of P_Q from (π, t), denoted by $(\pi, t) \rightarrow^*_{Q;P} (\pi', t')$, iff the following conditions hold:

1. if $t = t'$ then $\pi' = \pi$;
2. if $t' = t + 1$ then $(\pi, t) \rightarrow_{Q;P} (\pi', t')$;
3. if $t' > t + 1$ then there must exists some state $(\pi'', t' - 1)$ such that $(\pi, t) \rightarrow^*_{Q;P} (\pi'', t' - 1)$ and $(\pi'', t' - 1) \rightarrow_{Q;P} (\pi', t')$.

Note that $(\pi, t) \rightarrow^*_{\varnothing;P} (\pi', t')$ denotes the normal execution of a normal plan $P_\varnothing$. Such a normal plan execution will also be denoted by $(\pi, t) \rightarrow^*_P (\pi', t')$.

Using these definitions, it is not difficult to show that for every $0 \leq t \leq k$, the timed state (π', t), where $(\pi, 0) \rightarrow^*_{Q;P} (\pi', t)$, is uniquely defined if $<$ satisfies the Determinism requirement.

Example 4 Figure 3 gives an illustration of an execution of a plan with abnormal plan steps.

Suppose plan step s_3 is abnormal and generates a result that is unpredictable ($\perp$). Given the qualification $Q = \{s_3\}$ and the partially observed state π_0 at time point $t = 0$, we predict the partial states π_i as indicated in Fig. 3, where $(\pi_0, t_0) \rightarrow^*_{Q;P} (\pi_i, t_i)$ for $i = 1, 2, 3$. Note that since the value of v_1 and of v_5 cannot be predicted at time $t = 2$, the result of plan step s_6 and of plan step s_8 cannot be predicted and π_3 contains only the value of v_3. □

5 Observations and diagnoses

To establish plan diagnosis, we need to make *observations*. Our framework provides a natural candidate for representing such observations: an observation $obs(t)$ at time t is a timed state (π, t) where π is a partial state. This implies that we do not require observations to specify a complete state. Suppose we have an observation $obs(t) = (\pi, t)$ and an observation $obs(t') = (\pi', t')$ at some later time $t' > t \geq 0$ during the execution of a plan P. To indicate that these observations pertain to P, we will use the triple $\langle P, obs(t), obs(t')\rangle$.

We would like to use this triple $\langle P, obs(t), obs(t')\rangle$ to infer the health modes of the plan steps occurring in P. First, assuming a normal execution of P, we can easily predict the partial state of the world at a time point t' given the observation $obs(t)$: if all plan steps behave

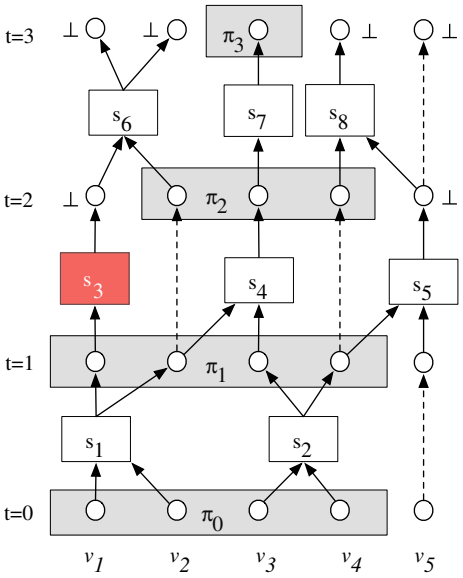

Fig. 3 Plan execution with an abnormal plan step (s_3)

normally, we predict the timed state $(\pi'_{\varnothing}, t')$ such that $obs(t) \rightarrow^*_P (\pi'_{\varnothing}, t')$. Such a prediction has to be compared with the actual observation $obs(t') = (\pi', t')$ made at time t'. It is easy to see when the predicted state and the observed state match: in that case we should be able to find a state σ such that both the observed state π' and the predicted state $\pi'_{\varnothing}$ are contained in σ, that is, $\pi' \sqsubseteq \sigma$ and $\pi'_{\varnothing} \sqsubseteq \sigma$. By definition of compatibility (see Definition 1), such a σ can only exist if $\pi'_{\varnothing}$ and π' are *compatible* states, i.e., if $\pi' \approx \pi'_{\varnothing}$ holds.

If this is not the case, the execution of some plan steps must have gone wrong and we have to determine a qualification Q such that the predicted state π'_Q derived using Q is compatible with π'. Hence, we have the following straight-forward extension of the diagnosis concept in MBD to plan diagnosis (cf. [5]):

Definition 4 Let $P = \langle \mathcal{O}, S, < \rangle$ be a plan with observations $obs(t) = (\pi, t)$ and $obs(t') = (\pi', t')$, where $t < t' \leq depth(P)$, and let $obs(t) \rightarrow^*_{Q;P} (\pi'_Q, t')$ be a derivation using P_Q. Then Q is said to be a *plan diagnosis* of $\langle P, obs(t), obs(t') \rangle$ iff $\pi' \approx \pi'_Q$.

Example 5 Consider again Fig. 3 and suppose that we did not know that plan step s_3 was abnormal and that we observed $obs(0) = ((d_1, d_2, d_3, d_4), 0)$ and $obs(3) = ((d'_1, d'_3, d'_5), 3)$. Using the normal plan derivation relation starting with $obs(0)$ we will predict a state $\pi'_{\varnothing}$ at time $t = 3$ where $\pi'_{\varnothing} = (d''_1, d''_2, d''_3)$. If everything is ok, the values of the variables predicted as well as observed at time $t = 3$ should correspond, i.e., we should have $d'_j = d''_j$ for $j = 1, 3$. If, for example, only d'_1 would differ from d''_1, then we could qualify s_6 as abnormal, since then the predicted state at time $t = 3$ using $Q = \{s_6\}$ would be $\pi'_Q = (d''_3)$ and this partial state π'_Q is compatible with the predicted state $\pi'_{\varnothing}$. □

Remark 1 Note that for all variables in $Var(\pi') \cap Var(\pi'_Q)$, the qualification Q provides an *explanation* for the observation π' made at time point t'. Hence, for these variables the qualification provides an *abductive diagnosis* [4]. For all observed variables in $Var(\pi') - Var(\pi'_Q)$, no value can be predicted given the qualification Q. Hence, by declaring them to be unpredictable, possible conflicts with respect to these variables if a normal execution of all plan steps

is assumed, are resolved. This corresponds with the idea of a *consistency-based diagnosis* [17]. □

5.1 Preferred diagnoses

In almost every application, one assumes that normal behavior is the rule and not an exception. In plan diagnosis it also seems reasonable to assume that the likelihood that a plan step fails is smaller than the likelihood that it behaves normally. So, assuming that plan steps fail independently, it is not difficult to conclude that in order to explain a deviation from normal behavior we should prefer, like in MBD, qualifications that do not contain more actions qualified as abnormal than necessary. Hence, like in MBD, we prefer *subset-minimal* diagnoses and especially *minimum* diagnoses among the set of minimal diagnoses. These subset- and cardinality minimal notions can be easily defined in our framework as follows: Given a plan with observations $\langle P, (\pi, t), (\pi', t')\rangle$, a qualification Q is said to be

1. a (*subset*) *minimal plan diagnosis* if for every plan diagnosis Q' such that $Q' \subseteq Q$, it holds that $Q = Q'$.
2. a *minimum plan diagnosis* if for every plan diagnosis Q', it holds that $|Q| \leq |Q'|$.

Example 6 Consider the plan depicted in Fig. 4. Suppose $obs(0) = (\pi_0, 0)$ and $obs(3) = (\pi'_3, 3)$ and π'_3 equals π_3 except that there is a deviation in the value of v_2 at time $t = 3$ (as indicated by the black dot). Note that there are three possible minimal diagnoses that are also minimum diagnoses : $Q_1 = \{s_1\}$, $Q_2 = \{s_3\}$ and $Q_3 = \{s_6\}$. Let π'_{Q_i} denote the state derived at time $t = 3$ by using Q_i as a qualification. Then $Var(\pi'_{Q_1}) = \varnothing$, $Var(\pi'_{Q_2}) = \{v_4, v_5\}$ and $Var(\pi'_{Q_3}) = \{v_3, v_4, v_5\}$, so these partial states predicted differ in their information content. □

Example 6 shows that, in general, minimum or minimal diagnoses might considerably differ in their predictive power or *preciseness*. For example, if we take Q_1 as a diagnosis, the values of all variables predicted at time $t = 3$ will be undefined, while taking Q_3 as a diagnosis, only v_1 and v_2 are undefined. Intuitively then we would prefer more precise diagnoses over less precise ones.

Like the underlying *likelihood of failure* principle that gives rise to a preference for size or subset minimal diagnoses, we would like to point out a principle that underlies a preference for most precise diagnoses. This principle is called the *likelihood of failure propagation* and simply assumes that the likelihood that a failed step produces incorrect results (either directly or indirectly) is larger than the likelihood that it produces correct results.[7]

The ramifications of this principle in our framework can be explained as follows: Let $P = \langle \mathcal{O}, S, <\rangle$ be a plan with observations $obs(t) = (\pi, t)$ and $obs(t') = (\pi', t')$, where $t < t'$, let $Q \subset S$ be a qualification of P and let $obs(t) \rightarrow^*_{Q;P} (\pi'_Q, t')$ be a derivation using P_Q.

The variables *correctly predicted* by P_Q at time t' are those variables $v \in Var(\pi'_Q)$ whose values are compatible with the value of v in the observation π'. Together they constitute the set $V_Q = \{v : v \in Var(\pi'_Q) \wedge (\pi'(v) \sqsubseteq \pi'_Q(v))\}$. Clearly, this set V_Q measures the *predictive power* of the qualification Q.

Now consider the empty qualification $V_\varnothing$. This set specifies the predictive power of the plan assuming that all plan steps are normal. From the definition of the derivability relation

[7] In particular, this principle implies that we should consider the phenomenon of *error masking* to be an unlikely event.

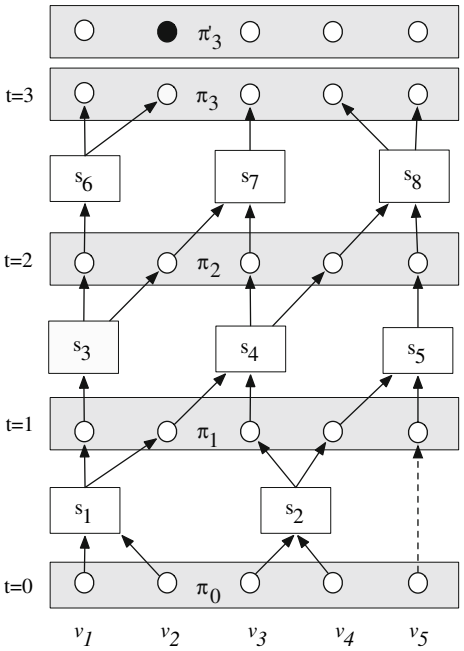

Fig. 4 Plan execution with an observation deviating from the expected observation, as indicated by the black dot

we immediately derive that for every qualification Q, $V_Q \subseteq V_\varnothing$. Clearly, the *likelihood of failure propagation* principle now implies that a qualification Q_1 that induces less variables in $V_\varnothing$ to be undefined than another qualification Q_2 does should be preferred above Q_2. For, any plan step s in a qualification Q that causes some variable $v \in V_\varnothing$ to become undefined lowers the likelihood of Q. Hence, according to the likelihood of failure propagation principle, we should prefer a diagnosis Q_1 above diagnosis Q_2 if $|V_{Q_1}| > |V_{Q_2}|$ and therefore, in general, we would prefer those diagnoses Q that *maximize* their predictive content V_Q.

If Q is a diagnosis, it is easy to see that $V_Q = Var(\pi'_Q)$: According to the definition of V_Q it suffices to show that $Var(\pi'_Q) \subseteq V_Q$. Assume that for some $v \in Var(\pi'_Q)$, $v \notin V_Q$. Then we must have $\bot \neq \pi'(v) \neq \pi'_Q(v) \neq \bot$, contradicting the fact that Q is a diagnosis. Hence $Var(\pi'_Q) \subseteq V_Q$.

Hence, we conclude that for every diagnosis Q, its predictive content $Var(\pi'_Q)$ is contained in the set $V_\varnothing$ and therefore, we would aim at finding diagnoses that maximize $|Var(\pi'_Q)|$, i.e., minimize the distance between $Var(\pi'_Q)$ and $V_\varnothing$. We call such diagnoses *maximum informative diagnoses*. Likewise, a diagnosis Q is said to be a *subset maximal* informative diagnosis if there exists no diagnosis Q' such that $Var(\pi'_Q) \subset Var(\pi'_{Q'})$.

A surprising property of maximum informative diagnoses is that they all have the same predictive content.[8] That is, whenever two diagnoses Q_1 and Q_2 are both maximum informative diagnoses then $Var(\pi_{Q_1}) = Var(\pi_{Q_2})$. This result can be explained by introducing the scope $scope_P(s)$ of a plan step s. Intuitively, the scope of a plan step s contains all plan steps s' such that all variables $v \in ran_{Var}(s')$ will become undefined whenever s is qualified as abnormal. This scope $scope(s)$ (we will omit P if no confusion is possible) is inductively defined as follows:

1. $s \in scope(s)$ and

[8] Given a plan P and observations $obs(t)$ and $obs(t')$.

2. if there exist plan steps s' and s'' such that (i) $depth_P(s') < depth_P(s'')$ and (ii) $ran_{Var}(s') \cap dom_{Var}(s'') \neq \varnothing$ then $s' \in scope(s)$ implies $s'' \in scope(s)$.

So if s is qualified as abnormal, all $s' \in scope(s)$ will set every variable in their range to $\perp$. Hence, extending the domain of a scope to sets of plan steps, we derive that $Q \subseteq scope(Q')$ implies that $V_{Q'} \subseteq V_Q$.

Now consider the set $V^{dif} = Var(\pi'_{\varnothing}) - V_{\varnothing}$. This is the set of all variables whose predicted values do not correspond to observed values at time t'. For every $v \in V^{dif}$ we can find a unique last plan step s such that the value of v at time t' is set by s. Let Q^{dif} be the set of these plan steps s. It is easy to see that Q^{dif} is a diagnosis, because all variables in V^{dif} will be set to $\perp$ using Q^{dif}. But Q^{dif} has another important property: For every diagnosis Q and every plan step $s \in Q^{dif}$ we should have that $s \in scope(s')$ for some $s' \in Q$. For otherwise, Q would fail to set some variable $v \in V^{dif}$ to undefined and therefore would not be a diagnosis. Hence, for every diagnosis Q, $Q^{dif} \subseteq scope(Q)$ and therefore V_Q is contained in $V_{Q^{dif}}$. But that immediately implies that Q^{dif} is a maximum informative diagnosis and $V_{Q^{dif}}$ is the (unique) maximum predictive content of such a diagnosis.

This result, of course, immediately implies that there is no difference between subset maximal and maximum informative diagnoses. Hence it suffices to deal with the former ones:

Definition 5 (maxi-diagnosis) Given plan observations $\langle P, (\pi, t), (\pi', t')\rangle$, a diagnosis Q is said to be a maximally informative plan diagnosis, abbreviated maxi-diagnosis, if there exists no diagnosis Q' such that $Var(\pi_Q) \subset Var(\pi_{Q'})$.

Such maxi-diagnoses, however, are not always *subset minimal* diagnoses. By combining the two criteria, however, we obtain a qualification that is able to achieve compatibility with the observations, being as exact in its predictions as possible, without considering too many actions as behaving abnormally. We therefore define a *minimal maximally-informative* diagnosis as follows:

Definition 6 (mini–maxi diagnosis) Given plan observations $\langle P, (\pi, t), (\pi', t')\rangle$, a diagnosis Q is said to be a minimal maximally informative plan diagnosis, abbreviated as mini-maxi diagnosis, if *(i)* Q is a maxi-diagnosis and *(ii)* there exists no maxi-diagnosis Q' such that $Q' \subset Q$.

We would like to point out that although the intersection of the set of minimal diagnoses and the set of maximal informative diagnoses is nonempty, this result does not hold with respect to the set of minimum diagnoses in the sense that there exist cases where no minimum diagnosis is maximally informative and vice-versa. For an example, consider Fig. 5. Here, the black dots in π_3 denote values that are predicted incorrectly. It is easy to see that while $Q = \{s_3, s_4\}$ is a maxi diagnosis, it is not a minimum diagnosis: $Q' = \{s_1\}$ is a unique minimum diagnosis containing fewer plan steps. The diagnosis Q' however, is not maximal informative, because it causes v_1 to become undefined in π_3, while Q does not.

Now the question of course is, how difficult would it be to compute these mini–maxi diagnoses?

6 Complexity results

In the preceding section we distinguished two types of preferred diagnoses: maximal informative diagnoses (maxi-diagnoses) and minimum diagnoses. Advantages of such maxi-diagnoses as e.g., their predictive power, might easily be lost if it turns out that they are extremely

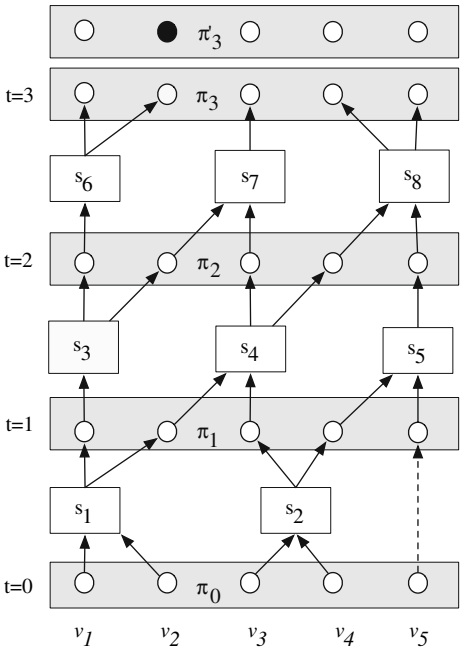

Fig. 5 An example to show that no minimum diagnosis is maximal informative

difficult to compute. For example, although in MBD minimum diagnoses are preferred, actual computations of diagnoses often aim at finding an approximation of them, since it is well-known that computing minimum diagnoses, even in very simple cases, is computationally hard.

As the first result, we have that, computationally minimum diagnoses in our plan-based framework do not differ from minimum diagnoses in MBD:

Proposition 2 *Finding minimum diagnoses in our plan-based framework is NP-hard.*

Proof There is a polynomial reduction from the well-known NP-complete Minimum Cover problem to the minimum plan diagnosis problem. See the appendix for the details. □

This at least shows that our framework does not trivialize such computations. As a corollary from the proof of this result, we note that finding a minimum plan diagnosis is already hard for plans with constant depth, more exactly from depth 2 on. Moreover, it follows that finding a minimum diagnosis is also hard for plans with plan steps having at most 3 elements in their range and domain.

Surprisingly, however, finding maxi-diagnoses and even finding subset minimal maxi-diagnoses is tractable. We will first give an intuitive description of an efficient procedure to find a (mini-) maxi diagnosis and then give a polynomial algorithm for finding a mini-maxi diagnosis.

Suppose we have plan observations $\langle P, (\pi, t), (\pi', t')\rangle$. To determine a maxi-diagnosis, we first determine the *disagreement set* V^{dif} of all those variables whose values are defined in both the observed state π' and the predicted state $\pi'_{\varnothing}$ at time t', but differ:

$$V^{dif} = \{v \mid v \in Var(\pi'_{\varnothing}) \cap Var(\pi') \wedge (\pi'_{\varnothing}(v) \neq \pi'(v))\}.$$

Next, we collect all plan steps s at time $t'-1$ such that there exists a variable $v \in ran_{Var}(s) \cap V^{dif}$. By the determinism requirement, two different plan steps s and s' occurring in some set P_t cannot have a variable in common in their range, hence for every $v \in V^{dif}$ there is at most one plan step $s_v \in P_{t'-1}$ such that $v \in ran_{Var}(s)$. Then we remove all variables v that occur in the range of the plan steps just selected from the disagreement set V^{dif}. For $i = 2, 3, \ldots$, we iteratively select new plan steps at times $t' - i$ having a variable in their range that also occurs in the disagreement set and we remove these plan steps until the disagreement set is empty. It can be shown that this procedure generates the set $Q_{max} = \{s_v \mid v \in V^{dif}\}$ where s_v is the *latest* plan step in the plan causing the value v to occur in the disagreement set. It can be easily proven that this qualification Q_{max} is a maxi-diagnosis.

In order to obtain a mini–maxi diagnosis instead of just a maxi diagnosis, we have to refine this procedure slightly using the earlier defined notion of the *scope* of a plan step.

Note that in the above procedure to generate a maxi-diagnosis, if we simply add a set S_i of new plan steps belonging to $P_{t'-i}$ to the already selected set of plan steps S, some of the plan steps $s' \in S$ might occur in $scope(s)$ for some $s \in S_i$. Hence, adding such a plan step s makes these previously added plan steps s' superfluous. Therefore, at each iteration step, we must remove such redundant plan steps s' to obtain a mini-maxi diagnosis.

The following algorithm specifies an iterative procedure to obtain a mini–maxi diagnosis[9] Q_{max}:

Algorithm 1 Algorithm to compute mini–maxi diagnoses

Require: plan observations $\langle P, (\pi, t), (\pi', t')\rangle$
Ensure: a mini–maxi informative diagnosis Q_{max}
  Let $V = V^{dif}$ and let $Q_{max} = \varnothing$;
  $i := 0$
  while $D \neq \varnothing$ **do**
    $i := i + 1$;
    $S_i := \{s \in P_{t'-i} \mid \exists v \in V[v \in ran_{Var}(s)]\}$;
    $Q_i := \{s' \in Q_{max} \mid \exists s \in S_i[s' \in scope(s)]\}$;
    $Q_{max} := (Q_{max} - Q_i) \cup S_i$;
    $V := V - \bigcup_{s \in Q_{max}} ran_{Var}(s)$
  end while
  return Q_{max}

Example 7 Consider again the plan execution depicted in Fig. 4. Given $obs(0)$ and $obs(3)$ and a deviation in the value of v_2 at time $t = 3$, we determine the disagreement set: $V^{dif} = \{v_2\}$. After selecting s_6 as a plan step to be included in the diagnosis, the disagreement set is empty. Hence, $Q = \{s_6\}$ is a maxi-diagnosis. □

Although mini–maxi diagnoses can be found efficiently, finding *minimum cardinality* maxi-diagnoses instead of mini–maxi-diagnoses constitutes an NP-hard problem. To show this, it suffices to reuse the reduction we applied to prove the hardness of finding minimum diagnoses. It is not difficult to see that this reduction from the NP-complete min-cover problem creates plan diagnosis instances where the disagreement set contains *all* variables. Hence, every diagnosis is a maxi-diagnosis by definition and therefore finding a maxi-diagnosis of minimum size comes down to finding a minimum diagnosis.

[9] Note that this algorithm finds *one* mini–maxi diagnosis.

7 Distributed plans and mini–maxi diagnoses

In real life, most plans are performed by more than one agent. That is, although there is a virtual global plan, often the plan is distributed over many actors, each knowing only a subset of the plan steps to be performed and the connections (dependencies) of their part of the plan with others. We would like to investigate the consequences of such distributed plans for diagnosing plan failures, especially with respect to finding mini-maxi diagnoses. In this section we will show that in a distributed setting mini-maxi diagnoses can be also computed in an efficient way, although the algorithm used significantly differs from the centralized version presented above.

7.1 Agents and plan distributions

Suppose that we have a plan $P = \langle \mathcal{O}, S, < \rangle$ and instead of having one agent knowing all plan steps to be performed there is a set $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$ of agents each knowing only a subset of plans steps occurring in P. Let S_i denote the subset of plan steps to be known by agent A_i and let us further assume that $\{S_1, \ldots, S_n\}$ is a partitioning of S. We furthermore assume that each agent A_i for each input v of a plan step s belonging to its set S_i knows exactly whether v occurs in the range of another plan step belonging to S_i and if not, whether s is an initial plan step or which other agent A_j should provide an input value for v. Conversely, every agent A_i is also aware of the variables v (s)he has to provide to another agent A_j. We also assume that if there exist plan steps $s \ll s'$ such that $s \in S_i$ and $s' \in S_j$ for some $i \neq j$ then agent A_i will inform agent A_j when s has been completed.

With this information and some minor modifications in the plan derivability relation (for each agent) as provided by Definition 2, it should be clear that given some initial observation at time t, the agents together are able to predict the value of each variable $v \in Var$ at time t', given a partial state obtained at time t.

7.2 Distributed diagnosis

We consider a plan $P = \langle \mathcal{O}, S, < \rangle$ with observations $obs(t)$ and $obs(t')$. The main idea of establishing a mini–maxi diagnosis in a distributed environment is somewhat more involved than in a centralized environment. Comparing the process with its centralized variant, in a distributed setting the main difficulty is that we cannot enforce any ordering on the computations of the set of agents. Therefore, instead of a backward addition of plan steps to the qualification, i.e., starting with adding plan steps in $P_{t'-1}$ then $P_{t'-2}$ and so on as done in the centralized algorithm, we have to use both a backward and a forward "qualifying" process. The "backward process" is mimicked by a local *label setting* process, where plan steps are given a preliminary label. In a subsequent "forward" *label propagating* process, plan steps acquire their definite label. Among the possible label values, there is one particular value that signifies a fault mode of the plan step s. These plan steps are announced by the agents and we will argue that the set of these plan steps constitutes a mini–maxi diagnosis.

We will use the following five values of the labels of a plan step: maybe faulty (mf), maybe healthy (mh), faulty (f), healthy (h) and no information propagating (no). Only the last three values are definite and the first two are preliminary labels. At the beginning of the procedure, no plan step $s \in S_i$ has a label value. At the end of the procedure every plan step is qualified as either h, no or f. An agent A_i stops processing as soon as every plan step $s \in S_i$ has acquired a definite label and it announces all those plan steps it has

qualified as faulty (f). We will now discuss the label setting and the label propagating procedures for an agent A_i into more detail.

7.2.1 Label setting phase

As long as there is a plan step s having no label value, agent A_i finds a label for s based on local information about s and the values of its input and output variables. First, we check whether there is some $v \in ran_{Var}(s) \cap V^{dif}$ that is not changed by any other plan step s' before time t'. If so, then s must be fully enabled. We distinguish two cases: If s occurs in P_t, all its input values are observed at time t and therefore it must be qualified. Hence, we set $l(s) = f$. Else, s is dependent upon the output of other plan steps and s may need to be qualified. Hence, we set $l(s) = mf$. This label mf signifies that the definite label may change to f if no other plan step s' preceding s is found that has the status f, or it may change to $l(s) = no$ if such a plan step is found.

If s does not directly produce a value of a variable in V^{dif}, then its label will be set to h, mh or no. It will be set to h if s is fully enabled in P_t, it will be set to mh if s is fully enabled in $P_{t''}$ for some time t'' with $t < t'' < t'$, and it will be set to no otherwise.

After the label setting phase, it is not difficult to see that the set of plan steps having a label with value mf or f constitutes a maximal informative diagnosis. We use the label propagation phase to construct a mini–maxi diagnosis.

7.2.2 Label propagating phase

For each plan step $s \in S_i$ having a preliminary label $l(s) = mh$ or $l(s) = mf$ we know that such a plan step must have predecessors s' in the plan P. Those predecessors might belong to S_i or to the set of plan steps of another agent. Such plan steps s will receive their definite label by inspecting the labels of their predecessors. Then the definite label value is set according to the following intuitive label propagation rules:

Here, $\forall x$ ($\exists x$) means that all (some) plan steps in the set of predecessors of s have the label value x. We simply assume that agents will able to retrieve the labels of these predecessors of plan step s and update the labels of s accordingly.

For each agent this label propagating procedure will stop after a finite number of steps: First of all, at the end of the label setting procedure, all plan steps s in P_t will have their definite label values. By the propagating rules, next all plan steps in P_{t+1} will get their definite labels and finally, all plan steps in $P_{t'}$ will obtain their definite label values. Moreover, during the label setting procedure, every plan step that directly contributes to the value of a variable in the disagreement set is labeled f or mf. The label mf is only changed to f if no direct or indirect predecessors of s have a label value mf or f. This clearly ensures that the final set of plan steps assigned to f is a minimal diagnosis. Hence, the resulting set is a mini–maxi diagnosis.

Example 8 Consider the plan as given in Fig. 6 and assume that for every plan step s there is a separate agent A responsible for it. In the label setting phase each agent determines the label of its plan step (See Fig. 6a). In the label propagation phase, the values of the labels of all plan steps that occur as predecessors of a plan step are propagated and the resulting final labels are given in Fig. 6b. Then each agent having a plan step qualified as f announces the identity of the plan step. Here the result is a mini-maxi diagnosis consisting of two plan steps marked as f.

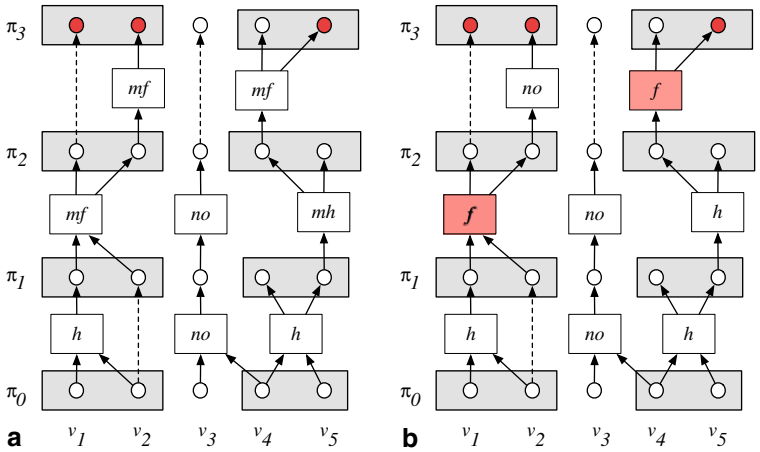

Fig. 6 An example of the results of the label setting (**a**) and the label propagating (**b**) procedures in the distributed diagnosis algorithm

Table 1 Label propagation rules

Labels of predecessors of s	*Current preliminary label s*	*Definite label s*
$\exists f$	*mf*	*no*
$\exists f$	*mh*	*no*
$\exists no$	*mf*	*no*
$\exists no$	*mh*	*no*
$\exists mf$	*mf*	*no*
$\exists mf$	*mh*	*no*
$\forall h$	*mh*	*h*
$\forall h$	*mf*	*f*

8 Conclusion

We have presented a simple formal framework to specify an executable plan and we have defined the notion of a diagnosis using partial observations of a plan in execution. We based our analysis of plans and observations upon a model-based diagnosis approach and considered a plan as a description of a system that can be observed and can be used to make predictions about its (future) behavior.

Using this framework, we derived a definition for a plan diagnosis as a set of abnormally qualified plan steps that are able to derive a partial plan state *compatible* with an observed partial plan state. In contrast to model-based diagnosis, where minimal and minimum diagnoses are aimed for, we have shown that minimality alone not always leads to the results we prefer. In order to make powerful predictions about future plan states, we argued that *maximal informative* diagnoses offer a valuable alternative.

We showed that in contrast to minimum diagnosis, a minimal maximum informative diagnosis can be found efficiently, although maximum informative diagnoses of minimum size are difficult to compute.

Finally, we extended our approach to diagnosis in a distributed setting, showing that mini-maxi diagnoses can be computed very efficiently in such environments, too.

Current work can be extended in several ways. We mention three possible extensions: First of all, we could improve our current notion of diagnosis by taking into account the difference between plan operators and plan steps. In some cases it could be useful to make a distinction between establishing diagnoses at the plan step level and diagnoses at the plan operator level. For example, if instances of a driving action (i.e., plan steps) pertain to a plan operator that refers to the use of one single vehicle and all these instances are qualified as being abnormal, there is sufficient reason to believe that the vehicle itself (the plan operator) is faulty. Such a distinction requires the inclusion of *causal rules* linking different plan steps to each other. By means of such causal rules the number of plan steps qualified as abnormal often can be significantly reduced. Secondly, going beyond plan operators, we could improve the diagnostic model to include a model of the executing agent(s) that is involved in executing one or more plan steps. In particular we need to consider cases where the agent might evolve through several abnormal states. We suspect the resulting model to be related to diagnosis in Discrete Event Systems [7,16]. Thirdly, we might investigate several relaxations of our current framework and the computational properties of (mini–maxi) diagnoses in these relaxations. For example, one interesting variant could be allowing for a more careful propagation of unknown values of variables in the domain of plan operators. Here, some variables in the range of the plan operator still might be defined, although some variables in the domain are undefined. Another variant could be a careful extension to non-deterministic aspects of plan step execution by introducing sets of values of variables in the range of plan operators, whenever some values of variables in their domain are undefined. It would be certainly interesting to investigate the properties of mini-maxi diagnoses in such variants, but we suspect their efficient computability property to be lost.

Acknowledgements We would like to thank the anonymous reviewers for pointing out some mistakes in the original version, for their very useful comments on the subjects discussed in this paper, and for their suggestions for possible extensions. This research is supported by the Technology Foundation STW, applied science division of the Dutch Science Foundation (NWO) and the technology programme of the Ministry of Economic Affairs (the Netherlands). Project DIT5780: Distributed Model Based Diagnosis and Repair.

A Appendix: Some proofs of propositions

A.1 Proof of Proposition 2

Proof We show that the decision-variant of the minimum plan diagnosis problem: Given plan observations $\langle P, (\pi, t), (\pi', t')\rangle$, and a positive integer K, does there exist a diagnosis of size K? is an NP-complete problem by reducing the following NP-complete Minimum Cover problem to it:

> Given a set E, some set $C \subseteq 2^E$ and an integer K, does there exist a subset $C' \subseteq C$ (a cover) with $|C'| \leq K$ such that every element of E belongs to at least one member of C'?

Note that without loss of generality, we may assume that $K < |E|$. The reduction is a fairly standard one. Consider an instance (E, C, K) of the Minimum Cover problem, where $E = \{e_1, e_2, \ldots, e_n\}$. We construct the following instance $(\langle P, (\pi, t), (\pi', t')\rangle, K')$, of the minimum diagnosis problem (decision variant) with $P = \langle \mathcal{O}, S, <\rangle$ as follows: Let $V =$

$\{v_{i,j} \mid c_j \in C, e_i \in c_j\}$ be the set of variables. For each $c_j \in C$, we define a plan step $s_{c_j} \in S$ with $Dom_{Var}(s_{c_j}) = ran_{Var}(s_{c_j}) = \{v_{i,j} \mid e_i \in c_j\}$. Furthermore, for every $e_i \in E$, we create a plan step s_{e_i} with $Dom_{Var}(e_{v_i}) = ran_{Var}(e_{v_i}) = \{v_{i,j} \mid e_i \in c_j\}$. The action performed by each plan step is simply to copy the value of the variables occurring in its domain to the corresponding variable in its range. Note that every plan step is range-restricted as required. Due to the concurrency requirement, we have to be careful in ordering the actions, since the value of a variable might not be affected at the same time by more than one action. Therefore we order the plan steps as follows: every plan step s_{c_j} has to precede every plan step s_{e_i}. This defines the plan P (See Fig. 7 for an example of this construction). Finally, let $K' = K$ and consider the following observations:

$$obs(0) = \{(1, 1, \ldots, 1), 0)\}$$

$$obs(2) = \{(2, 2, \ldots, 2), 2)\}$$

Let us consider the intuition behind this reduction. In the initial timed state $(\pi, 0)$, every variable $v_{i,j}$ has the value 1, while at time $t = 2$ the value (2) of every variable $v_{i,j}$ disagrees with the prediction that it will be 1, if it is assumed that no action will fail. We ask for the existence of a diagnosis Q containing K or fewer plan steps. Clearly, such a diagnosis has to predict that every value of a variable $v_{i,j}$ will be undefined at time $t = 2$. This means that we have to find K or fewer plan steps that can be qualified as abnormal and thereby induce the values of all variables to be undefined at time $t = 2$. It is not difficult to see that such a solution exists iff there are K or fewer plan steps chosen from the set $\{s_{c_j} \mid c_j \in C\}$ that are qualified as abnormal. This can be seen as follows: suppose we have a qualification Q' containing a plan step s_{e_i}. In that case there exists an action s_{c_j} such that $dom_{Var}(s_{e_i}) \cap ran_{Var}(s_{c_j}) \neq \varnothing$. But that implies that the set $(Q' - s_{e_i}) \cup s_{c_j}$ is also a diagnosis. Hence, we can exchange

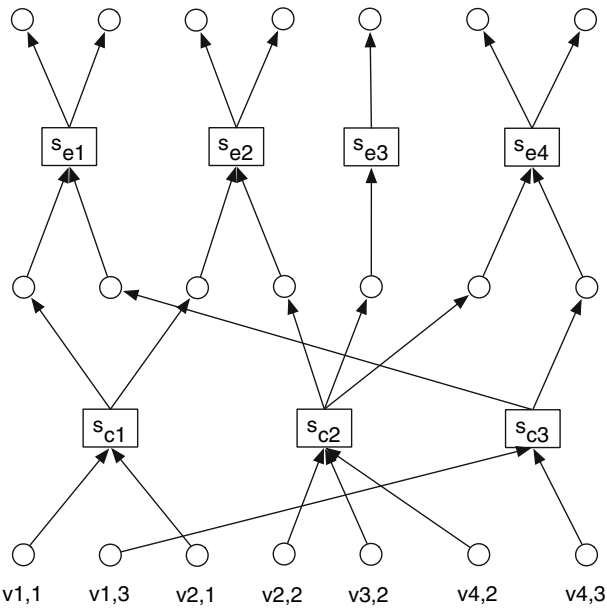

Fig. 7 A reduction from a min-cover instance (S, C, K) with $S = \{e_1, e_2, e_3, e_4\}$ and $C = \{c_1, c_2, c_3\}$, where $c_1 = \{e_1, e_2\}$, $c_2 = \{e_2, e_3, e_4\}$ and $c_3 = \{e_1, e_4\}$ to the minimum plan diagnosis problem

all plan steps s_{e_i} by plan steps s_{c_j} to obtain a diagnosis as subset of the set $\{s_{c_j} \mid c_j \in C\}$ without increasing the cardinality of the diagnosis.

Consider now a diagnosis $Q \subseteq \{s_{c_j} \mid c_j \in C\}$. Then, since every $v_{i,j}$ occurs in the range of the plan step s_{e_i}, at least one of the inputs $v_{i,j'}$ of s_{e_i} should be undefined. By the construction of the plan, this implies that for every s_{e_i} there exists at least one s_{c_j} with $e_i \in c_j$ such that $s_{c_j} \in Q$. But then it follows immediately that Q is a diagnosis of $\langle P, obs(0), obs(2)\rangle$ exactly if the set $C' = \{c_j \mid s_{c_j} \in Q\}$ is a cover of C. □

A.2 Correctness of the algorithm to compute mini-maxi diagnoses

We will prove that the algorithm indeed computes a maxi-diagnosis.

Proposition 3 *Given a plan observation $\langle P, (\pi, t), (\pi', t')\rangle$, the problem to find a mini-maxi diagnosis can be solved in polynomial time using the algorithm for computing mini-maxi diagnoses.*

Proof It is not difficult to see that for every $v \in V^{dif}$ there exists a plan step $s \in P$ such that v occurs in $ran_{Var}(s)$. Hence, the algorithm always terminates. Obviously, if Step 3.3 is omitted and Step 3.4 is changed into $Q_{max} := Q_{max} \cup S_i$, we are simply computing a maxi-diagnosis Q according to the definition. Since in Step 3.4, only plan steps s' are removed that belong to the scope $scope(s)$ of plan steps s just added to the set Q_{max}, it is not difficult to see that the final qualification O_{max} computed by the algorithm is a diagnosis. Since Q_{max} is a subset of the maxi-diagnosis Q, it easily follows from the definition of the derivation relation that $\pi'_Q \sqsubseteq \pi'_{Q_{max}}$, implying that $Var(\pi'_Q) \subseteq Var(\pi'_{Q_{max}})$. Since Q is a maxi-diagnosis, this immediately implies that $Var(\pi'_Q) = Var(\pi'_{Q_{max}})$ and, therefore, Q_{max} is a maxi diagnosis, too.

To show that Q_{max} is a subset-minimal diagnosis, let $obs(t' - i)$ denote the observation predicted using the empty qualification $\varnothing$ at time $t' - i$ for $i = 0, \ldots, t' - t$. It is easy to prove by induction over i that the following invariant holds for $i = 0, \ldots, t' - t$:

> For every i, Q_{max} is a minimal diagnosis for the plan $P^i = \bigcup_{j=o}^{i} P_{t'-j}$ with observations $obs(t' - i)$ and $obs(t')$.

Since the algorithm halts when P^i contains all plan steps between time t and time t', the minimality of Q_{max} follows.

Finally, the algorithm is clearly polynomial. Suppose there are m variables and n plan steps. Each of the sets S_i, Q_i, Q_{max} and D can be determined in $O(m \times n^2)$-time. Since $t' - t \leq n$, it follows that the algorithm halts in $O(m \times n^3)$-time. □

References

1. Bertoli, P., Cimatti, A., Pistore, M., & Traverso, P. (2002). Plan validation for extended goals under partial observability (preliminary report). In *Proceedings of the AIPS 2002 Workshop on Planning via Model Checking*, pp. 14–22, Toulouse, France.
2. Birnbaum, L., Collins, G., Freed, M., & Krulwich, B. (1990). Model-based diagnosis of planning failures. In *AAAI 90*, pp. 318–323.
3. Carver, N., & Lesser, V. R. (2003). Domain monotonicity and the performance of local solutions strategies for cdps-based distributed sensor interpretation and distributed diagnosis. *Autonomous Agents and Multi-Agent Systems, 6*(1), 35–76.
4. Console, L., & Torasso, P. (1990). Hypothetical reasoning in causal models. *International Journal of Intelligence Systems, 5*, 83–124.

5. Console, L., & Torasso, P. (1991). A spectrum of logical definitions of model-based diagnosis. *Computational Intelligence, 7*, 133–141.
6. Cox, J. S., Durfee, E. H., & Bartold, Th. (2005). A distributed framework for solving the multiagent plan coordination problem. In *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pp. 821–827, New York, NY, USA, 2005. ACM Press.
7. Debouk, R., Lafortune, S., & Teneketzis, D. (2000). Coordinated decentralized protocols for failure diagnosis of discrete-event systems. *Journal of Discrete Event Dynamical Systems: Theory and Application, 10*, 33–86.
8. Eiter, T., Erdem, E., & Faber, W. (2004). Diagnosing plan execution discrepancies in a logic-based action framework. Technical report INFSYS RR-1843-04-03, TU-Wien.
9. Fikes, R. E., & Nilsson, N. (1971). Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence, 5*, 189–208.
10. Gierz, G., Hofmann, K. H., Keimel, K., Lawson, J. D., Mislove, M., & Scott, D. S. (2003). Continuous Lattices and Domains. In *Encyclopedia of Mathematics and its Applications,* Vol. 93, Cambridge University Press.
11. Horling, B., Benyo, B., & Lesser, V. (2001). Using self-diagnosis to adapt organizational structures. In *Proceedings of the 5th International Conference on Autonomous Agents*, pp. 529–536. ACM Press.
12. Jensen, R. M., & Veloso, M. M. (1999). OBDD-based universal planning: Specifying and solving planning problems for synchronized agents in non-deterministic domains. *Lecture Notes in Computer Science, 1600*, 213–228.
13. de Jonge, F., & Roos, N. (2004). Plan-execution health repair in a multi-agent system. In *Proceedings of the 23rd Workshop of the UK Planning and Scheduling Special Interest Group (PLANSIG 2004)*, pp. 33–44, Cork, Ireland.
14. Kalech, M., & Kaminka, G. A. (2003). On the design of social diagnosis algorithms for multi-agent teams. In *IJCAI-03*, pp. 370–375.
15. Kalech, M., & Kaminka, G. A. (2005). Diagnosing a team of agents: Scaling-up. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-05)*, pp. 249–255, ACM Press.
16. Pencolé, Y., Cordier, M., & Rozé, L. (2001). Incremental decentralized diagnosis approach for the supervision of a telecommunication network. In *Twelfth International Workshop on Principles of Diagnosis Ð DXÕ01*. San Sicario, Italy, pp. 151–158.
17. Reiter, R. (1987). A theory of diagnosis from first principles. *Artificial Intelligence, 32*, 57–95.
18. Witteveen, C., Roos, N., van der Krogt, R. P. J., & de Weerdt, M. M. (2005). Diagnosis of single and multi-agent plans. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-05)*, pp. 805–812, ACM Press.