Minimalist Diagnosis of Discrete-Event Systems

Gianfranco Lamperti
Department of Information Engineering, University of Brescia, Italy

Marina Zanella
Department of Information Engineering, University of Brescia, Italy

Abstract

Model-based diagnosis of discrete-event systems (DESs) is afflicted by two major difficulties, the former being the huge size of the search space, which has a heavy impact on the processing time, the latter being a possibly large number of diagnoses explaining the perceived sequence of observations, which may cause a cognitive overload in human diagnosticians or even delays in post-processing. These difficulties add up and they are exacerbated in critical scenarios where an action must be taken in real-time. To make DES diagnosis viable in these contexts, a Minimalist Diagnosis Engine is presented, which is based on a parsimony principle: instead of computing the set of all diagnoses inherent to the given sequence of observations, only minimal diagnoses are elicited as candidates. Since in this paper, as in most contributions on model-based diagnosis of DESs in the literature, a diagnosis is defined as a set of faults, minimal diagnoses are *subset* minimal. The proposal is justified since minimal diagnoses are suitable for DESs, and since the new diagnosis engine is able to prune the search space, thus reducing the computation effort with respect to a sound and complete method. Moreover, in order to further decrease the execution time, whenever the method is dealing with a new observation, it performs online a (partial) knowledge-compilation so as the portions of the DES space that have already been processed and transformed into chunks of compiled knowledge can speed up the next abductive reasoning steps, relevant to the upcoming observations.

2012 ACM Subject Classification Computing methodologies → Causal reasoning and diagnostics

Keywords and phrases model-based reasoning, diagnosis during monitoring, discrete-event systems, active systems, subset-minimal diagnosis, dynamical knowledge-compilation, minimalism, laziness

Digital Object Identifier 10.4230/OASIcs.DX.2024.12

Funding We acknowledge the support of the PNRR project FAIR - Future AI Research (PE00000013), Spoke 9 - Green-aware AI, under the NRRP MUR program funded by the NextGenerationEU, specifically by the project *Argumentation for Informed Decisions with Applications to Energy Consumption in Computing* – AIDECC (CUP D53C24000530001).

1 Introduction

Automated diagnosis, which is aimed at finding out what is wrong in a given (natural or synthetic) system, is still a challenge to AI. Model-based diagnosis [9, 25] is a well-founded, principled approach to accomplish this task. Its rationale is to process a model that is specific to the considered system by means of a domain-independent reasoning engine. In 1987, a fundamental work [24] (*a*) defined the concept of *consistency-based* diagnosis, by applying it to static composite systems, and (*b*) adopted *weak* models, i.e. models that describe only the normal behavior of each component.

A consistency-based diagnosis problem instance amounts to an observation that is inconsistent with the given (modeled) system if all the components are assumed to behave normally. Intuitively, a *diagnosis* (result) is a conjecture that some components are behaving abnormally (they are *faulty*) and the rest (i.e. the components that do not belong to the diagnosis) are behaving normally, where such conjecture is consistent with the system description and the observation. Notice that several alternative diagnoses can be found and that, if the models are weak, given a set of components that is a consistency-based diagnosis, any

35th International Conference on Principles of Diagnosis and Resilient Systems (DX 2024).
Editors: Ingo Pill, Avraham Natan, and Franz Wotawa; Article No. 12; pp. 12:1–12:20
OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

superset of it is a consistency-based diagnosis itself. Hence, in consistency-based diagnosis, when weak models are adopted, the set of all the diagnoses can be characterized by the only diagnoses that are *subset-minimal*, where a diagnosis is subset-minimal if no subset of it is a diagnosis. Computing only the minimal diagnoses adheres to the so-called *parsimony principle* [24]. Unfortunately, as explained in [4], this characterization of consistency-based diagnoses does not hold if the adopted models are *strong*, that is, they describe not only the normal behavior of components but also (some) faulty behavioral modes. Analogously, the above characterization does not hold for *abduction-based* diagnosis [23], which necessarily adopts strong models. An abduction-based diagnosis (result) is an assignment of specific modes to the system components such that the conjunction of this assignment with the system description *entails* the (given) observation. The above theoretical distinction between consistency-based and abduction-based diagnosis applies also to model-based diagnosis of dynamical systems, where models have to describe the state changes over time.

This paper deals with model-based diagnosis of dynamical systems represented as discrete-event systems (DESs). The models adopted in this paper are compositional, strong and *complete*, that is, they encompass *all* the normal and abnormal behaviors of the considered components. The complete behavior of each component is represented as a nondeterministic communicating finite automaton endowed with an *initial state*, where each state transition is either normal or affected by a specific *fault* and, orthogonally, either observable, through a specific event, or unobservable. The state evolves over qualitative time, that is, time tags are ignored. The system operation corresponds to a *trajectory*, which is a chronologically ordered sequence of component state-transitions that starts from the initial state of the overall DES (this being the composition of the initial states of its components) and generates a sequence of observable events; each individual event that has been observed is called an *observation* and the whole sequence of the observed events is called a *temporal observation*. The considered DES diagnosis approach is abduction-based as a *diagnosis* is the set of faults on a trajectory that entails the temporal observation perceived so far. Since several trajectories (possibly an unbounded number, if the model allows for unobservable cycles of state transitions) can generate the same given temporal observation, the number of distinct diagnoses, each being a set of faults, may be very large, and a diagnosis may be a superset of another one. Unfortunately, as already remarked, in this context subset-minimal diagnoses do not characterize all the diagnoses (as in fact some trajectories whose relevant set of faults is a superset of a diagnosis may not entail the given sequence of observable events). Hence, most existing approaches to model-based diagnosis of DESs usually produce *all* the diagnoses relevant to the given temporal observation. However, in this paper, where we focus on the task of DES *diagnosis during monitoring*, which issues a new set of diagnoses upon the reception of each new observation, we propose an algorithm that computes minimal diagnoses only. This choice cannot be grounded just on the (legitimate) need to reduce the number of outputted results, each called a *candidate*, since too many candidates may be overwhelming for the human diagnostician who has to make a decision, possibly under stringent time constraints, or even for an artificial real-time agent. In fact, if the intent of our proposal were just reducing the number of candidates, it would be enough to implement a (purposefully efficient) post-processor that could draw the minimal diagnoses from the collection of all diagnoses computed by a sound and complete existing method. Proposing a new diagnostic method is justified only if two conditions hold: minimal diagnoses are suitable for diagnosis of DESs, and the diagnosis method can reduce the computation time with respect to a sound and complete method. The first condition will be discussed in the next section, while Sections 3 and 4 will provide some background and introduce the new diagnosis method, respectively.

The search space of the new method is smaller than that of a sound and complete method, hence also the second condition above is fulfilled. Section 5 hints to previous works in the literature that have some links with this one. A few summarizing remarks and intentions for future research conclude the paper in Section 6.

2 Motivation

We have already distinguished the notion of a *diagnosis*, which is theoretical and defines a domain, from that of a *candidate*, which is an output actually produced by the diagnosis engine at hand. In this paper, as in most contributions on model-based diagnosis of DESs in the literature, a diagnosis is a *set* of faults relevant to a trajectory that entails the given temporal observation. If the method is *sound*, each candidate is a diagnosis; if it is *complete*, each diagnosis is a candidate. Hence, if the method is sound and complete, the set of candidates includes all and only the diagnoses relevant to the considered problem instance. In this section, if a proposition or a definition mentions *all diagnoses*, it implicitly refers to the output produced by a sound and complete DES diagnosis method. Analogously to all contributions on model-based diagnosis of DESs by other authors, this paper assumes that the observations are not affected by any *uncertainty* [12], that is, they are received, without any alteration in their content and number, in their emission order. If the observations are not affected by any uncertainty, then the set of all diagnoses outputted by a sound and complete method includes the *actual diagnosis*, that is, the only diagnosis that reflects what has really happened inside the system. In Section 4 we propose a method that is sound while it is not complete, as it computes subset-minimal diagnoses only. This implies that the actual diagnosis may not be one of the (computed) candidates. However, when several candidates are outputted, rather that analyzing them individually, an interesting piece of information is their intersection. In fact, if (there is no uncertainty in the observations and) all diagnoses are provided (by a sound and complete method), their intersection is a set of faults that have occurred with certainty, as all such faults are included in the actual diagnosis. Luckily, if only the minimal diagnoses are provided, there is no loss in such information, as stated by the following proposition (in this section, the proofs of all propositions are omitted for the sake of space).

▶ **Proposition 1.** *The intersection of all diagnoses equals the intersection of all the subset-minimal diagnoses. If there is no uncertainty in the DES observations, all the faults in such intersection have occurred with certainty.*

Hence, if we compute minimal diagnoses only, we know the same set of faults that have occurred with certainty as if we had computed all diagnoses. This property provides a motivation for computing minimal diagnoses only and plays an important role in diagnosis during monitoring of DESs, as it will be explained in Subsections 2.2 and 2.3.

2.1 Probability

Assuming that faults in a DES are reciprocally independent, following [5], the probability of a candidate is the product of the individual probabilities of each fault in the candidate to occur and each remaining fault not to occur. If the (possibly unknown) probability value is (sensibly assumed to be) less than 0.5 for each fault, be the probabilities of faults equal to each other or different from each other, each minimal diagnosis δ is more probable than any of its supersets $\delta' \supset \delta$. Hence, the (possibly not unique) most probable diagnosis is a minimal diagnosis (although, unfortunately, the remaining minimal diagnoses are not necessarily very probable). Therefore, the set of minimal diagnoses includes (among others) the most probable diagnoses, which is a good point for minimal diagnoses.

2.2 Monotonicity

When the task of diagnosis during monitoring is performed, the diagnostic engine processes each observable event o_i as soon as it has been perceived and produces a new output Δ_i, that is, a new collection of candidates that are inherent to the whole temporal observation $[o_1, \ldots, o_i]$ received so far. If the method is sound and complete, at each monitoring step all the diagnoses relevant to the temporal observation received so far are computed. A property relevant to this task is *monotonicity* [15], which is recalled here below.

▶ **Definition 2** (Monotonicity [15]). *The results $\langle \Delta_1, \ldots, \Delta_n \rangle$ inherent to a DES (diagnosis during) monitoring problem instance are* monotonic *iff $\forall i \in [1..(n-1)]$, we have $\forall \delta_{i+1} \in \Delta_{i+1}, (\exists \delta_i \in \Delta_i, \delta_i \subseteq \delta_{i+1})$.*

Monotonicity holds if not all the candidates in Δ_i are refuted once a new observation o_{i+1} is processed, instead, some of them, possibly extended through the addition of further faults, will become the new candidates in Δ_{i+1}. This property ensures that the faults in the intersection of the candidates produced in a processing step will never be refuted in the subsequent processing steps, as stated in Proposition 3.

▶ **Proposition 3.** *If the results $\langle \Delta_1, \ldots, \Delta_n \rangle$ inherent to a DES (diagnosis during) monitoring problem instance are monotonic, then, $\forall i \in [1..(n-1)]$, the intersection of the candidates in Δ_i is a subset in all the candidates in any Δ_j, $j > i$.*

Notice that Proposition 3 is independent of the method for DES diagnosis during monitoring, provided that it produces monotonic results. It can easily be proven that, if the observations are not affected by any uncertainty, the results generated by a sound method that is either complete or computes minimal diagnoses only are monotonic. This brings to the following proposition.

▶ **Proposition 4.** *Let $\langle \Delta_1, \ldots, \Delta_n \rangle$ be the results inherent to a DES (diagnosis during) monitoring problem where each Δ_i consists of either all diagnoses (produced by a sound and complete method) or minimal diagnoses only. If there is no uncertainty in the observations, the results are monotonic, which in turn implies that the intersection of the candidates in Δ_i, this intersection being a set of faults that have occurred with certainty, is a subset of faults in all the candidates in any Δ_j, $j > i$.*

Hence, when considering minimal diagnoses only, we can identify the same set of faults that have certainly occurred as when we compute all diagnoses, we do not lose any fault identification ability. This is a really good point for minimal diagnoses.

2.3 Diagnosability

Which are the faults that we can identify? This depends on the *diagnosability* of DESs. In order to recall the definition of diagnosability [26], let us introduce some formalism. Let L be the language representing all the trajectories (sequences of state transitions) inherent to a given DES D. Let $L_f \subseteq L$ be the language of all the trajectories of D that include a transition affected by fault f, and $\bar{L_f} \subseteq L_f$ be the language of all the trajectories in L_f where such transition is the last one. Let *Obs* be the function that provides the (chronologically ordered) sequence of observable events that have occurred on a given (chronologically ordered) sequence of state transitions. Let . be the concatenation operator for sequences of state transitions.

▶ **Definition 5** (Diagnosability [26]). *Given a DES D whose set of faults is Σ_f, a fault $f \in \Sigma_f$ is* diagnosable *if* $\forall \tau_1 \in \bar{L}_f, \exists k \in \boldsymbol{N}, \forall \tau_2 : \tau_1 . \tau_2 \in L, |Obs(\tau_2)| \geq k \Rightarrow (\forall \tau \in L), (Obs(\tau) = Obs(\tau_1 . \tau_2) \Rightarrow (\tau \in L_f))$*. System D is* diagnosable *if all its faults are diagnosable.*

In other words, a fault f is diagnosable if, for whichever sequence of transitions τ_1 that has preceded the occurrence of f, and for whichever sequence of transitions τ_2 (generating a finite number k of observable events) that has followed it, all the trajectories that produce the same temporal observation as $\tau_1 . \tau_2$ include the fault. The value k associated (according to Definition 5) with a diagnosable fault is here called the *(diagnosability) delay* of that fault.

▶ **Proposition 6.** *Given a DES D whose set of faults is Σ_f, let $f \in \Sigma_f$ be a* diagnosable *fault with delay $k \in \boldsymbol{N}$. Let $\tau_1 . \tau_2$ be the actual trajectory followed by D, where $\tau_1 \in \bar{L}_f$ and $|Obs(\tau_2)| = k$. Let $Obs(\tau_1 . \tau_2)$ be the temporal observation (with no uncertainty) provided altogether as input either to a sound and complete method or to a method that computes minimal diagnoses only. Then, all the diagnoses, and minimal diagnoses as well, include fault f, which is therefore included in the intersection of all the diagnoses, as well as in the intersection of all minimal diagnoses.*

What is the impact of diagnosability when the task of diagnosis during monitoring is performed? Proposition 7 provides an answer.

▶ **Proposition 7.** *Given a DES D whose set of faults is Σ_f, let $f \in \Sigma_f$ be a* diagnosable *fault with delay $k \in \boldsymbol{N}$. Let $\langle \Delta_1, \ldots, \Delta_n \rangle$ be the results inherent to a DES (diagnosis during) monitoring problem instance, where every Δ_i consists of either all diagnoses or the minimal diagnoses only. Let us assume that fault f occurs after the reception of the i-th observation and before or upon the reception of the $(i+1)$-th observation. If there is no uncertainty in the observations, fault f belongs to the intersection of the candidates in any Δ_j, $j \geq i + k$, that is, for each step $j \geq i + k$, it belongs to the set of faults that have occurred with certainty.*

Thus, computing minimal diagnoses only (instead of all diagnoses) does not involve any drawback in detecting and isolating diagnosable faults. Unfortunately, there is a drawback if a fault is not diagnosable, that is, if there exists an (ambiguous) trajectory wherein the fault occurs that is indefinitely observationally identical to a trajectory wherein the fault never occurs. If the perceived temporal observation is consistent with an ambiguous trajectory, there will be a pair of diagnoses, one that includes the fault and the other that does not. Hence, the fault will not fall in the intersection of all diagnoses (nor in the intersection of minimal diagnoses), since it is not certain. However, while the set of all diagnoses includes both diagnoses, the set of minimal diagnoses does not necessarily include both of them (and it may include neither of them). Anyway, if we are interested in the set of faults that have certainly occurred rather than in knowing each and every diagnosis, computing minimal diagnoses only does not cause any loss in the identified certain faults nor any longer delay in their identification with respect to computing all diagnoses. This is why we stick to the computation of minimal diagnoses only.

3 Preliminaries

A distributed, asynchronous DES is a network of components that are modeled as finite communicating automata and are connected to other components via *links*. When a DES starts its behavior, each component is in its initial state and links are assumed to be empty. The occurrence of an external event (outside the DES) may trigger a state transition in a component that is sensitive to that event, which may generate internal events towards other components, resulting in a cascade of state transitions, called a *trajectory* of the DES, which moves the DES from its initial system state to a new system state. The trajectories of a DES $\mathcal{X}$ are confined to a *space*, which is itself a finite automaton, namely $\mathit{Space}(\mathcal{X}) = (\mathbf{T}, X, \tau, x_0)$, where the alphabet $\mathbf{T}$ is the set of component transitions, X is the set of (system) states, where a state is a pair of a tuple of states of components and a tuple of the events within links, τ is the transition function mapping a state and a component transition into a new state, and x_0 is the initial state. Formally, each string (sequence) of component transitions in the language of $\mathit{Space}(\mathcal{X})$ is a trajectory of $\mathcal{X}$.

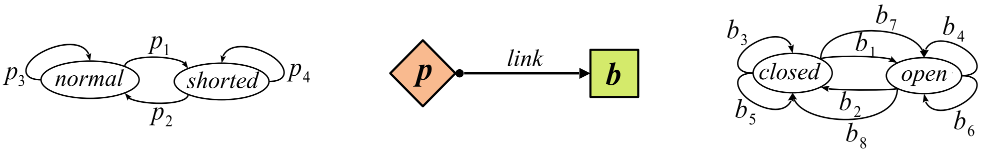

Figure 1 DES watcher $\mathcal{X}_w$ (center), and models of protection p (left) and breaker b (right).

▶ **Example 8.** Outlined in the center of Figure 1 is a DES, called $\mathcal{X}_w$ (*watcher*), which is designed to protect a power transmission line from short circuits. A watcher includes two components, a protection p and a breaker b, and a link from p to b, which serves as a communication channel for the commands sent by the protection to the breaker. When a short circuit strikes a protected line, the protection is expected to command the breaker to open in order to get the short circuit extinguished. If the short circuit is eventually extinguished, the protection commands the breaker to close in order to restore the electric power. The communicating automata (models) of both p (left) and b (right) involve two states, namely *normal* and *shorted* for the protection, and *closed* and *open* for the breaker. Transitions are represented as arrows between states, which are labeled with their identifiers. Component transitions are described in Table 1 (first and second column). Each component transition from a state s to a state s' that is triggered by an input event e and generates a set of output events E is denoted by a triple $\langle s, (e, E), s' \rangle$. If event e is ε, it means that the transition is triggered by an event outside the DES, i.e. an event that is not sent by another component via a link. For instance, transition $p_1 = \langle \mathit{normal}, (\varepsilon, \{\mathit{op}\}), \mathit{shorted} \rangle$ of the protection is triggered by an external event (drop in line voltage, which indicates the occurrence of a short circuit on the line), generates the single event *op* (to open the breaker), and moves the protection from state *normal* to state *shorted*. All transitions of the breaker

Table 1 Description of transition actions for protection p and breaker b in DES $\mathcal{X}_w$ (columns 1 and 2), along with observations and faults in $\mathit{Map}(\mathcal{X}_w)$ (columns 3 and 4).

Transition	*Action*	*Obs*	*Fault*
$p_1 = \langle \mathit{normal}, (\varepsilon, \{\mathit{op}\}), \mathit{shorted} \rangle$	p reacts to a short circuit by generating the open event	$\mathbf{p}$	ε
$p_2 = \langle \mathit{shorted}, (\varepsilon, \{\mathit{cl}\}), \mathit{normal} \rangle$	p reacts to a short circuit extinction by generating the close event	$\mathbf{p}$	ε
$p_3 = \langle \mathit{normal}, (\varepsilon, \{\mathit{cl}\}), \mathit{normal} \rangle$	p reacts to a short circuit by generating the close event	ε	*pfo*
$p_4 = \langle \mathit{shorted}, (\varepsilon, \{\mathit{op}\}), \mathit{shorted} \rangle$	p reacts to a short circuit extinction by generating the open event	ε	*pfc*
$b_1 = \langle \mathit{closed}, (\mathit{op}, \emptyset), \mathit{open} \rangle$	b reacts to the open event by opening	$\mathbf{b}$	ε
$b_2 = \langle \mathit{open}, (\mathit{cl}, \emptyset), \mathit{closed} \rangle$	b reacts to the close event by closing	$\mathbf{b}$	ε
$b_3 = \langle \mathit{closed}, (\mathit{op}, \emptyset), \mathit{closed} \rangle$	b reacts to the open event by remaining closed	ε	*bfo*
$b_4 = \langle \mathit{open}, (\mathit{cl}, \emptyset), \mathit{open} \rangle$	b reacts to the close event by remaining open	ε	*bfc*
$b_5 = \langle \mathit{closed}, (\mathit{cl}, \emptyset), \mathit{closed} \rangle$	b reacts to the close event by remaining closed	$\mathbf{b}$	ε
$b_6 = \langle \mathit{open}, (\mathit{op}, \emptyset), \mathit{open} \rangle$	b reacts to the open event by remaining open	$\mathbf{b}$	ε
$b_7 = \langle \mathit{closed}, (\mathit{cl}, \emptyset), \mathit{open} \rangle$	b reacts to the close event by opening	$\mathbf{b}$	*bop*
$b_8 = \langle \mathit{open}, (\mathit{op}, \emptyset), \mathit{closed} \rangle$	b reacts to the open event by closing	$\mathbf{b}$	*bcl*

are triggered by an event generated by the protection (either op or cl), and do not generate output events ($E = \emptyset$). The space of $\mathcal{X}_w$ is depicted in Figure 2, where states are labeled $0 \ldots 7$, with 0 being the initial state. Each state is identified by a pair of states of components p and b, along with the (possibly empty, namely ε) event within the link. For instance, a trajectory of $\mathcal{X}_w$ is $T = [p_3, b_5, p_1, b_3, p_4, b_3, p_2, b_5]$. Due to cycles in the space, $\mathcal{X}_w$ may follow (at least in theory) an infinite number of different trajectories.

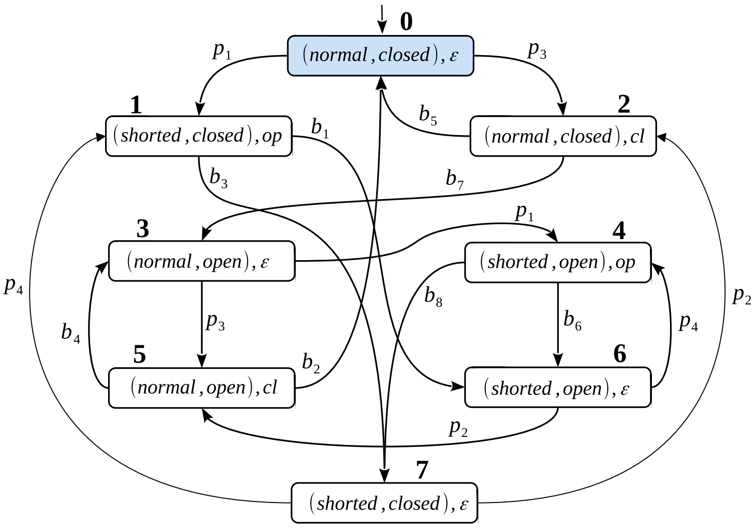

Figure 2 $\mathit{Space}(\mathcal{X}_w)$, the space of the *watcher*, where **0** is the initial state.

To support the diagnosis of a DES $\mathcal{X}$, we need to define both the *observability* and the *abnormality* of $\mathcal{X}$. This is specified by a table, namely $\mathit{Map}(\mathcal{X})$, which is a set of triples (t, o, f), where t is a component transition, o is a (possibly empty, namely ε) observation, and f is a (possibly empty, namely ε) fault. Specifically, if $o \neq \varepsilon$, then t is *observable*, otherwise t is *unobservable*; similarly, if $f \neq \varepsilon$, then t is *faulty*, otherwise t is *normal*. Based on $\mathit{Map}(\mathcal{X})$, each trajectory T of $\mathcal{X}$ is associated with a *temporal observation*, which is the sequence of the observations associated with the observable component transitions in T, namely $\mathit{Obs}(T) = [\, o \mid t \in T, (t, o, f) \in \mathit{Map}(\mathcal{X}), o \neq \varepsilon \,]$. A trajectory T is said to *conform* with a temporal observation $\mathcal{O}$ iff $\mathit{Obs}(T) = \mathcal{O}$. $\mathit{Map}(\mathcal{X})$ also associates T with a *diagnosis*, which is the set of faults associated with the component transitions in T, namely $\mathit{Dgn}(T) = \{\, f \mid t \in T, (t, o, f) \in \mathit{Map}(\mathcal{X}), f \neq \varepsilon \,\}$. The *diagnosis set* $\mathcal{D}$ of $\mathcal{O}$ is the set of diagnoses of the trajectories of $\mathcal{X}$ conforming with $\mathcal{O}$ (i.e. it is the set of *all diagnoses*, according to the terminology in Section 2), namely: $\mathcal{D}(\mathcal{O}) = \{\mathit{Dgn}(T) \mid T \in \mathit{Space}(\mathcal{X}), \mathcal{O} = \mathit{Obs}(T)\}$. Let $\mathcal{O}_i$ denote a nonempty prefix $[o_1, \ldots, o_i]$ of $\mathcal{O}$, $i \geq 1$. The *temporal diagnosis set* $\mathbf{D}$ of $\mathcal{O}$ is the sequence of diagnosis sets $\mathcal{D}(\mathcal{O}_i)$, $i \geq 1$, namely: $\mathbf{D}(\mathcal{O}) = [\mathcal{D}(\mathcal{O}_1), \mathcal{D}(\mathcal{O}_2), \ldots]$.

▶ **Example 9.** For DES $\mathcal{X}_w$, we assume two possible observations, namely $\mathbf{p}$: the protection performs a normal action, and $\mathbf{b}$: the breaker performs a (possibly faulty) action. We also assume six possible faults, namely *pfo*: the protection fails to send the open command, *pfc*: the protection fails to send the close command, *bfo*: the breaker fails to open, *bfc*: the breaker fails to close, *bop*: the breaker opens instead of closing, and *bcl*: the breaker closes instead of opening. Based on these sets of observations and faults, the observability and abnormality of $\mathcal{X}_w$ is defined in $\mathit{Map}(\mathcal{X}_w)$, which is embedded in Table 1 (third and fourth columns), where both an observation label and a fault label are associated with each component transition.

Accordingly, transition p_1 is observable and normal, p_3 is unobservable and faulty, while b_7 is both observable and faulty. Since the same observation is associated with several transitions, the component transition that actually occurred cannot be univocally identified based on the given observation only. With trajectory $T = [p_3, b_5, p_1, b_3, p_4, b_3, p_2, b_5]$, we have $\mathit{Obs}(T) = [\mathbf{b}, \mathbf{p}, \mathbf{p}, \mathbf{b}]$ and $\mathit{Dgn}(T) = \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\}$. Based on $\mathit{Space}(\mathcal{X}_w)$, we can find that the diagnosis set of $\mathit{Obs}(T)$ includes six diagnoses: $\{\mathit{pfo}, \mathit{bfo}\}$, $\{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\}$, $\{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\}$, $\{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}, \mathit{bop}\}$, $\{\mathit{pfo}, \mathit{bfo}, \mathit{bfc}, \mathit{bop}\}$, and $\{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}, \mathit{bfc}, \mathit{bop}\}$. Even if the diagnosis set involves the actual diagnosis $\{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\}$ relevant to trajectory T, due to ambiguity, five additional diagnoses are embodied in that set. The temporal diagnosis set of $\mathcal{O}$ is $[\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3, \mathcal{D}_4]$, where $\mathcal{D}_1 = \{\{\mathit{pfo}\},\{\mathit{pfo}, \mathit{bop}\},\{\mathit{pfo}, \mathit{bfc}, \mathit{bop}\}\}$, $\mathcal{D}_2 = \{\{\mathit{pfo}\}, \{\mathit{pfo}, \mathit{bfo}\}, \{\mathit{pfo}, \mathit{bop}\}, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\}, \{\mathit{pfo}, \mathit{bfc}, \mathit{bop}\}\}$, $\mathcal{D}_3 = \{\{\mathit{pfo}, \mathit{bfo}\}, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\}, \{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\}, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}, \mathit{bfc}\}, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}, \mathit{bop}\}, \{\mathit{pfo}, \mathit{bfo}, \mathit{bfc}, \mathit{bop}\}, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}, \mathit{bfc}, \mathit{bop}\}\}$, and $\mathcal{D}_4 = \{\{\mathit{pfo}, \mathit{bfo}\}, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\}, \{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\}, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}, \mathit{bop}\}, \{\mathit{pfo}, \mathit{bfo}, \mathit{bfc}, \mathit{bop}\}, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}, \mathit{bfc}, \mathit{bop}\}\}$. We can easily appreciate that the results (in the temporal diagnosis set) are monotonic (cf. Definition 2), as expected based on Proposition 4. The sequence of the intersections of all candidates in each $\mathcal{D}_i$, $i \in [1 \mathinner{..} 4]$, is $[\{\mathit{pfo}\}, \{\mathit{pfo}\}, \{\mathit{pfo}, \mathit{bfo}\}, \{\mathit{pfo}, \mathit{bfo}\}]$, where each intersection provides a set of faults that have occurred with certainty. Thus, after processing the first observation, we know that fault pfo has certainly occurred, the same as after the second observation. Notice that each intersection for $i \in [1 \mathinner{..} 2]$ equals the actual diagnosis (relevant to $\mathcal{O}_i$). After processing the third observation, we know that also fault bfo has certainly occurred. The fourth observation does not add any new certain fault. The intersections for $i \in [3 \mathinner{..} 4]$ do not involve fault pfc, although it has actually occurred in T. Given $\mathit{Map}(\mathcal{X}_w)$, fault pfc is not diagnosable. A proof of its non-diagnosability is provided, for instance, by two trajectories, $[p_3, b_5, p_1, b_3, p_4]$ and $[p_3, b_5, p_1]$, which are observationally identical and end in the same state (1). The former trajectory is affected by fault pfc (associated with p_4) while the latter is not. Both trajectories can go on indefinitely, producing the same observations, and, if they will not follow p_4 any more, the occurrence of pfc will be uncertain for ever. For instance, if the next transitions of both trajectories are $[b_3, p_2, b_5]$, the former is indeed T, which is why fault pfc does not belong to the intersections for $i \in [3 \mathinner{..} 4]$.

4 Minimalist diagnosis

We now present a Minimalist Diagnosis Engine for DESs, which generates the set of minimal diagnoses at the reception of each newly-occurred observation with the support of a minimalist diagnoser generated lazily. The algorithm makes use of the definitions given below.

▶ **Definition 10.** *Let $\mathcal{D}$ be the diagnosis set of a temporal observation $\mathcal{O}$. A diagnosis $\delta \in \mathcal{D}$ is* minimal *iff there is no other diagnosis $\delta' \in \mathcal{D}$ such that $\delta' \subset \delta$. The* candidate set *of $\mathcal{O}$, $\Delta(\mathcal{O})$, is the set of minimal diagnoses in $\mathcal{D}$.*

▶ **Example 11.** With reference to Example 9, the candidate set of the temporal observation $\mathcal{O} = [\mathbf{b}, \mathbf{p}, \mathbf{p}, \mathbf{b}]$ is the singleton $\Delta(\mathcal{O}) = \{\{\mathit{pfo}, \mathit{bfo}\}\}$, as every other diagnosis in the diagnosis set of $\mathcal{O}$ is a superset of $\{\mathit{pfo}, \mathit{bfo}\}$.

▶ **Definition 12.** *Let $\mathcal{O} = [o_1, o_2, \ldots]$ be a temporal observation of $\mathcal{X}$. The* temporal candidate set *of $\mathcal{O}$ is the sequence of candidate sets $\Delta(\mathcal{O}_i)$, $i \geq 1$, namely, $\boldsymbol{\Delta}(\mathcal{O}) = [\Delta(\mathcal{O}_1), \Delta(\mathcal{O}_2), \ldots]$, where $\mathcal{O}_i$ is the prefix of $\mathcal{O}$ up to the i-th observation.*

▶ **Example 13.** With reference to the temporal diagnosis set of $\mathcal{O} = [\mathbf{b}, \mathbf{p}, \mathbf{p}, \mathbf{b}]$ in Example 9, the temporal candidate set of $\mathcal{O}$ is $\mathbf{\Delta}(\mathcal{O}) = [\{\{pfo\}\}, \{\{pfo\}\}, \{\{pfo, bfo\}\}, \{\{pfo, bfo\}\}]$, where, incidentally and in contrast with the temporal *diagnosis* set, each candidate set is a singleton. We can easily appreciate that also these results, relevant to minimal diagnoses only, are monotonic, as stated by Proposition 4. The sequence of the intersections of all candidates in each Δ_i, $i \in [1 \mathinner{..} 4]$, is $[\{pfo\}, \{pfo\}, \{pfo, bfo\}, \{pfo, bfo\}]$, where each intersection consists of faults that have occurred with certainty. As expected, this sequence is the same as when all candidates are computed (cf. Example 9).

▶ **Definition 14.** *An* abduction item *of a DES $\mathcal{X}$ is a pair $\Im = (x, \delta)$, where x is a state in $Space(\mathcal{X})$ and δ is a diagnosis of a trajectory of $\mathcal{X}$ ending in x. The* frontier *of $\Im$, $Front(\Im)$, is a set of triples (x', f, o) where $\langle x, t, x' \rangle$ is a transition in $Space(\mathcal{X})$ and $(t, o, f) \in Map(\mathcal{X})$. A* candidate item *of $\mathcal{X}$ is a triple (x, δ, o), where (x, δ) is an abduction item of $\mathcal{X}$ and $o \neq \varepsilon$ is an observation such that $(t, o, f) \in Map(\mathcal{X})$ and t is the last transition of a trajectory of $\mathcal{X}$ ending in x.*

▶ **Example 15.** With reference to $Space(\mathcal{X}_w)$ in Figure 2 and $Map(\mathcal{X}_w)$ in Example 9, an abduction item of $\mathcal{X}_w$ is $\Im = (4, \{pfo, bop\})$, where $4 = ((shorted, open), op)$ is a state of $\mathcal{X}_w$ and $\{pfo, bop\}$ is the diagnosis of trajectory $[p_3, b_7, p_1]$ of $\mathcal{X}_w$ ending in state 4. The frontier of $\Im$ is the set $Front(\Im) = \{(6, \mathbf{b}, \varepsilon), (7, \mathbf{b}, bcl)\}$. A candidate item of $\mathcal{X}_w$ is $(7, \{pfo, bop, bcl\}), \mathbf{b})$, where $7 = ((shorted, closed), \varepsilon)$ is a state of $\mathcal{X}_w$ and $\{pfo, bop, bcl\}$ is the diagnosis of trajectory $[p_3, b_7, p_1, b_8]$ of $\mathcal{X}_w$ ending in 7, with $(b_8, \mathbf{b}, bcl) \in Map(\mathcal{X}_w)$.

▶ **Definition 16.** *Let $\mathbf{O}$ be the domain of observations of a DES $\mathcal{X}$ with initial state x_0. The* minimalist diagnoser *of $\mathcal{X}$ is a finite automaton*

$$Mind(\mathcal{X}) = (\mathbf{O}, D, \tau, d_0) \tag{1}$$

where D is the set of states $(\mathbb{I}, \mathbb{D})$, with $\mathbb{I}$ being a set of abduction items of $\mathcal{X}$ and $\mathbb{D}$ a set of diagnoses, $d_0 = (\{(x_0, \emptyset)\}, \{\emptyset\})$ is the initial state, and $\tau : D \times \mathbf{O} \mapsto D$ is the transition function, where $\tau((\mathbb{I}, \mathbb{D}), o) = (\mathbb{I}', \mathbb{D}')$ iff, for each $(x, \delta) \in \mathbb{I}$, there is in $Space(\mathcal{X})$ a sequence $[t_1, \ldots, t_k]$ of contiguous transitions, $k \geq 1$, where t_1 exits state x, all transitions $t_1, \ldots, t_{k-1}$ are unobservable while t_k is observable via observation o and enters a state x', δ' is the extension of δ by the faults involved in transitions $t_1, \ldots, t_k$, $(x', \delta') \in \mathbb{I}'$ provided that there is no other abduction item $(x', \bar{\delta}) \in \mathbb{I}'$ such that $\bar{\delta} \subset \delta'$, and $\mathbb{D}'$ is the minimal set of diagnoses in $\{\delta' \mid (x', \delta') \in \mathbb{I}'\}$.

▶ **Example 17.** A portion of the minimalist diagnoser of DES $\mathcal{X}_w$ is shown in Figure 5. The upper and lower parts of each of the eight depicted states $d_0 \ldots d_7$ display the relevant set $\mathbb{I}$ of abduction items and the set $\mathbb{D}$ of minimal diagnoses, respectively.

▶ **Proposition 18.** *Let $\mathcal{O} = [o_1, \ldots, o_n]$, $n \geq 0$, be a temporal observation of a DES $\mathcal{X}$. Let $d = (\mathbb{I}, \mathbb{D})$ be a state of $Mind(\mathcal{X}) = (\mathbf{O}, D, \tau, d_0)$ such that, if $n = 0$, then $d = d_0$, else $d_1 = \tau(d_0, o_1)$, $d_2 = \tau(d_1, o_2)$, …, $d_n = \tau(d_{n-1}, o_n)$. $\mathbb{D}$ equals the candidate set $\Delta(\mathcal{O})$.*

Proof. By induction on $\mathcal{O}$.

(*Basis*). *The property holds for $\mathcal{O}_0 = [\,]$.* In fact, in this case, $d = d_0 = (\{(x_0, \emptyset)\}, \{\emptyset\})$, where $\mathbb{D} = \{\emptyset\}$. Based on Definition 10, the candidate set $\Delta(\mathcal{O}_0)$ is the set of minimal diagnoses in the diagnosis set $\mathcal{D}(\mathcal{O}_0) = \{Dgn(T) \mid T \in Space(\mathcal{X}), \mathcal{O}_0 = Obs(T)\}$. Since $\mathcal{O}_0$ is empty, the diagnosis relevant to the empty trajectory belongs to $\mathcal{D}(\mathcal{O}_0)$. Since the diagnosis of the empty trajectory is $\emptyset$, the empty diagnosis cannot be a superset of any other diagnosis in $\mathcal{D}(\mathcal{O}_0)$, and every other diagnosis in $\mathcal{D}(\mathcal{O}_0)$ is a superset of it. Hence, $\Delta(\mathcal{O}_0) = \mathbb{D} = \{\emptyset\}$.

(*Induction*). *If the property holds for* $\mathcal{O}_i = [o_1, \ldots, o_i]$, $0 \leq i < n$, *then it also holds for* $\mathcal{O}_{i+1} = [o_1, \ldots, o_{i+1}]$. Considering state $d_i = (\mathbb{I}_i, \mathbb{D}_i)$, based on Definition 16, for each trajectory T of $\mathcal{X}$ conforming with $\mathcal{O}_i$, $\mathbb{I}_i$ includes all abduction items (x, δ) such that x is the state reached by T, and δ is a minimal diagnosis in $\{Dgn(T') \mid T' \in Space(\mathcal{X}), T' = Obs(\mathcal{O}), T'$ ending in $x\}$. This will not jeopardize the completeness of candidate sets because, given two abduction items (x, δ) and (x, δ') where $\delta' \supset \delta$, all minimal diagnoses generated by trajectories exiting x and generating δ' in x can also be generated by the trajectories exiting x and generating δ in x, since the conformity of a trajectory with the rest of the temporal observation depends only on state x, not on the associated diagnosis in the abduction item. On the other hand, in order to preserve completeness, it is necessary keeping both abduction items (x, δ) and (x', δ') in $\mathbb{I}_i$ where $x \neq x'$ and $\delta' \supset \delta$, since we do not know whether the trajectories exiting x will still conform with the rest of the temporal observation. Thus, the set $\mathbb{D}'_i = \{\delta \mid (x, \delta) \in \mathbb{I}_i\}$ will include all the diagnoses in candidate set $\Delta(\mathcal{O}_i)$. This is also (trivially) true for $\mathbb{D}'_0$. Moreover, for the same reasons, based on Definition 16, $\mathbb{I}_{i+1}$ is such that $\mathbb{D}'_{i+1} = \{\delta \mid (x, \delta) \in \mathbb{I}_{i+1}\}$ includes all the diagnoses in $\Delta(\mathcal{O}_{i+1})$. Hence, $\mathbb{D}_{i+1} = \Delta(\mathcal{O}_{i+1})$. ◀

▶ **Corollary 19.** *Let* $\mathcal{O} = [o_1, \ldots, o_n]$, $n \geq 0$, *be a temporal observation of a DES* $\mathcal{X}$. *Let* $[d_1, \ldots, d_n]$ *be the sequence of states in minimalist diagnoser* $Mind(\mathcal{X}) = (\mathbf{O}, D, \tau, d_0)$, *where, for each state* $d_i = (\mathbb{I}_i, \mathbb{D}_i)$, $i \in [1 \mathinner{..} n]$, $d_1 = \tau(d_0, o_1), d_2 = \tau(d_1, o_2), \ldots, d_i = \tau(d_{i-1}, o_i)$. *The sequence* $[\mathbb{D}_1, \ldots, \mathbb{D}_n]$ *equals the temporal candidate set* $\boldsymbol{\Delta}(\mathcal{O})$.

4.1 Minimalist diagnosis engine

The Minimalist Diagnosis Engine algorithm (lines 1–38) takes as input a temporal observation $\mathcal{O} = [o_1, o_2, \ldots]$ of a DES $\mathcal{X}$ (with initial state x_0), and generates as output a sequence $\boldsymbol{\Delta}$, which is in fact the temporal candidate set of $\mathcal{O}$. The algorithm exploits four main internal data structures: a set $\mathcal{G}$ of abduction items *generated* already, a stack $\mathcal{U}$ of abduction items *under* processing, that is, relevant to the current observation, a set $\mathcal{C}$ of *candidate* items (x, δ, o') relevant to any possible next observation o', and the minimalist diagnoser of $\mathcal{X}$, namely $\mathcal{M} = Mind(\mathcal{X})$. Note that $\mathcal{M}$ is not constructed upfront (offline), but only generated lazily based on the sequence of observations in $\mathcal{O}$. In other words, at any point of the processing, $\mathcal{M}$ represents the part of $Mind(\mathcal{X})$ materialized so far (online) by the algorithm. In line 1, $\mathcal{G}$ and $\mathcal{U}$ are initialized with the abduction item $\Im_0 = (x_0, \emptyset)$. In line 2, the initial state d_0 of $\mathcal{M}$ is generated (as unmarked) and assigned to d, which represents the state of $\mathcal{M}$ relevant to the matching of the current prefix of $\mathcal{O}$ (initially, this is the empty sequence). The aim is to generate the transition function of d (if not materialized already, that is, when d is unmarked), so that, at the next occurring observation o', the relevant candidate set is already computed and stored in the target state d' of $\mathcal{M}$, based on transition $\langle d, o', d' \rangle$. After the initialization of the data structures, the rest of the algorithm consists of two nested loops. The main loop (lines 3–38) is repeated until the DES stops being operated (no further observation can be generated). The nested loop (lines 5–22) is repeated until stack $\mathcal{U}$ becomes empty, in which case no further new abduction item can be generated for the current observation. The idea is to keep generating the frontier of the abduction items in stack $\mathcal{U}$ up to any possible next observation, based on the corresponding trajectories of $\mathcal{X}$, thereby possibly updating δ in each successive abduction item. In the first statement of the main loop (line 4), if the current state d of $\mathcal{M}$ is not marked (that is, if the transition function of d has not been materialized already), the nested loop is executed. At each iteration, an abduction item $\Im = (x, \delta)$ is popped from $\mathcal{U}$ (line 6) and, if it is unmarked in $\mathcal{G}$, that is,

Algorithm 1 Minimalist Diagnosis Engine.

```
input  : $\mathcal{O} = [o_1, o_2, \ldots]$, a temporal observation of a DES $\mathcal{X}$ having initial state $x_0$
output : $\mathbf{\Delta}$, the temporal candidate set of $\mathcal{O}$
$\Im_0 \leftarrow (x_0, \emptyset)$,  $\mathcal{G} \leftarrow \{\Im_0\}$,  $\mathcal{U} \leftarrow \lfloor \Im_0 \rfloor$,  $\mathcal{C} \leftarrow \emptyset$,  $\mathbf{\Delta} \leftarrow [\,]$
Create the initial state $d_0 = (\{\Im_0\}, \{\emptyset\})$ of the minimalist diagnoser $\mathcal{M}$ of $\mathcal{X}$, $d \leftarrow d_0$
repeat
    if the current state $d$ of $\mathcal{M}$ is not marked then
        repeat
            Pop an abduction item $\Im = (x, \delta)$ from $\mathcal{U}$
            if $\Im$ is unmarked in $\mathcal{G}$ then
                foreach $(x', f, o) \in Front(\Im)$ do
                    $\delta' \leftarrow \delta \uplus f$,  $\Im' \leftarrow (x', \delta')$
                    if $o = \varepsilon$ then
                        if $\Im' \notin \mathcal{G}$ then
                            if $(x', \bar{\delta}) \in \mathcal{G}$ where $\bar{\delta} \subseteq \delta'$ then
                                Mark $\Im'$
                            else if $x' \neq x$ then
                                Mark in $\mathcal{G}$ every $(x', \bar{\delta})$ where $\bar{\delta} \supset \delta'$
                                Push the abduction item $\Im'$ onto $\mathcal{U}$
                            Insert (the possibly marked) abduction item $\Im'$ into $\mathcal{G}$
                    else if there is no $(x', \bar{\delta}, o) \in \mathcal{C}$ where $\delta' \supseteq \bar{\delta}$ then
                        Insert $(x', \delta', o)$ into $\mathcal{C}$
                        Remove from $\mathcal{C}$ every $(x', \delta'', o)$ where $\delta'' \supset \delta'$
            Mark $\Im$ in $\mathcal{G}$
        until $\mathcal{U}$ is empty
        Let $\mathbf{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$ be a partition of $\mathcal{C}$, where each $\mathcal{C}_j$, $j \in [1 \mathinner{..} k]$, includes the
         candidate items $(x, \delta, o'_j)$ relevant to observation $o'_j$
        foreach $\mathcal{C}_j \in \mathbf{C}$, $\mathbb{I}_j = \{(x, \delta) \mid (x, \delta, o'_j) \in \mathcal{C}_j\}$ do
            if $d_j = (\mathbb{I}_j, \_)$ is not a state already generated in $\mathcal{M}$ then
                $\mathbb{D}_j \leftarrow \emptyset$
                foreach abduction item $(x', \delta') \in \mathbb{I}_j$ do
                    if there is no diagnosis $\bar{\delta} \in \mathbb{D}_j$ where $\bar{\delta} \subseteq \delta'$ then
                        Insert the diagnosis $\delta'$ into $\mathbb{D}_j$
                        Remove from $\mathbb{D}_j$ every diagnosis $\delta''$ where $\delta'' \supset \delta'$
                Generate a state $d_j = (\mathbb{I}_j, \mathbb{D}_j)$ in $\mathcal{M}$
            Generate a transition $\langle d, o'_j, d_j \rangle$ in $\mathcal{M}$
        Mark $d$
    $\mathcal{G} \leftarrow \emptyset$,  $\mathcal{C} \leftarrow \emptyset$
    Let $o'$ be the next observation in $\mathcal{O}$, $\langle d, o', d' \rangle$ a transition in $\mathcal{M}$, where $d' = (\mathbb{I}', \mathbb{D}')$
    Append $\mathbb{D}'$ to $\mathbf{\Delta}$
    $d \leftarrow d'$,  $\mathcal{G} \leftarrow \mathbb{I}'$,  $\mathcal{U} \twoheadleftarrow \mathbb{I}'$
until $\mathcal{X}$ stops being operated.
```

neither processed nor pruned, each triple (x', f, o) in its frontier is considered (lines 8–20). First, an abduction item $\Im' = (x', \delta')$ is generated (line 9), where $\delta' = \delta \uplus f$ is the extension of δ by f, which has no effect when $f = \varepsilon$. Then, two scenarios are considered: either when $o = \varepsilon$ (lines 10–17) or $o \neq \varepsilon$ (lines 18–20). When $o = \varepsilon$, abduction item $\Im'$ is still relevant to the current observation. Thus, if not already processed (line 11), in order to avoid processing abduction items derived from $\Im'$ that cannot lead to new minimal diagnoses, $\Im'$ is marked in case δ' is a superset of a diagnosis $\bar{\delta}$ relevant to an abduction item sharing the same state x' in $\Im'$ (lines 12–13); otherwise (lines 14–16), if $x' \neq x$, every abduction item $(x', \bar{\delta})$, where $\bar{\delta}$ is a (strict) superset of δ', is marked to avoid computing non-minimal diagnoses: in this case, $\Im'$ is pushed onto $\mathcal{U}$ (line 16). In either case, $\Im'$ (which may have been marked in line 13) is eventually inserted into $\mathcal{G}$ (line 17). When, instead, o is observable (lines 18–20), a new candidate item (x', δ', o) is inserted into $\mathcal{C}$ provided that there is no other candidate item $(x', \bar{\delta}, o)$ in $\mathcal{C}$ (that is, relevant to same state x' and same observation o) where δ' is a superset of $\bar{\delta}$ (otherwise, candidate item (x', δ', o) is bound to lead to a non-minimal diagnosis when processing the next observation o). Furthermore, once inserted the new candidate item, all the other candidate items (x', δ'', o) in $\mathcal{C}$, where δ'' is a (strict) superset of δ', are removed for the same reason (non-minimality). At the end of the iteration (line 21), once all triples in the frontier of $\Im$ have been processed (and the corresponding new abduction items $\Im'$ have been generated), abduction item $\Im$ is marked in $\mathcal{G}$. When $\mathcal{U}$ becomes empty, the nested loop terminates (line 22). Now, the candidate set $\mathcal{C}$ contains the initial abduction items for every possible next observation $o'_1, \ldots, o'_k$, which makes it possible to materialize the transition function of the current state d of $\mathcal{M}$. To this end, a partition $\mathbf{C}$ of $\mathcal{C}$ is considered based on the possible next observations $o'_1, \ldots, o'_k$, so that the projection of each $\mathcal{C}_j \in \mathbf{C}$, $j \in [1\,..\,k]$, on the first two elements of the candidate items, namely x and δ, is in fact the set of abduction items identifying state d_j, where $\langle d, o'_j, d_j \rangle$ is a transition in $\mathcal{M}$ to be materialized. This is why, in the loop in lines 24–32, the set $\mathbb{I}_j$ of abduction items (relevant to observation o'_j) is determined in order to possibly generate (if not already generated) the target state d_j in $\mathcal{M}$. In lines 26–31, the corresponding candidate set $\mathbb{D}_j$ is computed. Specifically, in lines 27–30, for each abduction item (x', δ') in $\mathbb{I}_j$, δ' is inserted into $\mathbb{D}_j$ provided that, in order to preserve minimality, $\mathbb{D}_j$ does not include any diagnosis $\bar{\delta}$ that is a subset of δ'. Moreover, if δ' is inserted into $\mathbb{D}_j$, every diagnosis in $\mathbb{D}_j$ that is a superset of δ' is removed (line 30). Once $\mathbb{D}_j$ is computed, a new state $d_j = (\mathbb{I}_j, \mathbb{D}_j)$ is generated in $\mathcal{M}$ (line 31). Eventually, in line 32, a new transition $\langle d, o'_j, d_j \rangle$ is created in $\mathcal{M}$, which allows for the immediate generation of the candidate set $\mathbb{D}_j$ in case the next observation will be o'_j. Once all transitions exiting d have been materialized, the current state d of $\mathcal{M}$ is marked (line 33), meaning that the materialization of the transition function of d has been completed. At this point, in order to process the next occurring observation, the sets $\mathcal{G}$ and $\mathcal{C}$ are emptied (line 34). When a new observation o' occurs (line 35), $\mathcal{M}$ surely includes the relevant transition $\langle d, o', d' \rangle$, where $d' = (\mathbb{I}', \mathbb{D}')$, with $\mathbb{D}'$ being the next diagnosis (candidate) set, which is then appended to $\mathbf{\Delta}$ without any additional processing. Before ending the iteration of the main loop, the current state d of $\mathcal{M}$ is changed to d' (the state reached by d with observation o'), while set $\mathcal{G}$ is set to $\mathbb{I}'$, the new initial abduction items, which are also pushed into stack $\mathcal{U}$ (symbol "$\twoheadleftarrow$" denotes this collective push operation): this way, the nested loop will start its computation with the initial abduction items relevant to the newly-occurred observation o'.

▶ **Example 20.** Consider the temporal observation for the DES given in Example 9, which has been extended by a further observation $\mathbf{b}$, namely $\mathcal{O} = [\mathbf{b}, \mathbf{p}, \mathbf{p}, \mathbf{b}, \mathbf{b}]$. To trace the run of Minimalist Diagnosis Engine on $\mathcal{O}$, shown in Figure 3 is the space of $\mathcal{X}_w$ (cf. Figure 2) where each component transition identifier t has been replaced by the observation and fault

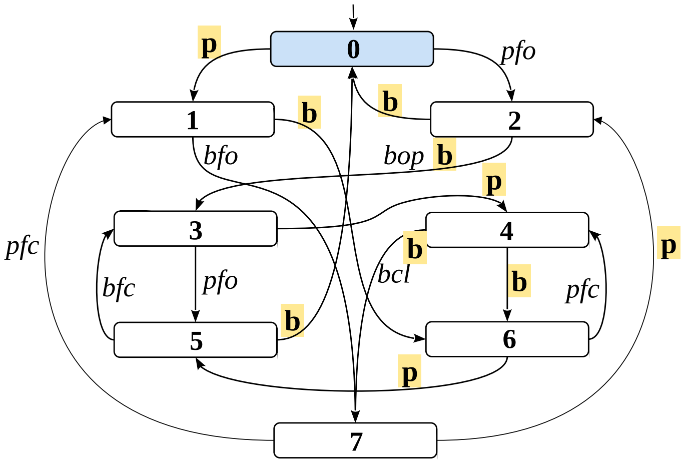

Figure 3 $\mathit{Space}(\mathcal{X}_w)$, with each component transition being replaced by the corresponding observation and fault defined in $\mathit{Map}(\mathcal{X}_w)$ (Table 1), while space states are identified by numbers.

associated with t in $\mathit{Map}(\mathcal{X}_w)$. For instance, transition b_7 marking the arc from state 2 to state 3 in $\mathit{Space}(\mathcal{X}_w)$ is substituted by labels $\mathbf{b}$ and bop, as $(b_7, \mathbf{b}, \mathit{bop}) \in \mathit{Map}(\mathcal{X}_w)$. It should be plain, however, that the labeling of transitions in $\mathit{Space}(\mathcal{X}_w)$ with observations and faults is only instrumental for the clarity of the example: it is not necessary to the algorithm. Or better still, $\mathit{Space}(\mathcal{X}_w)$ is neither available nor necessary to the engine. The generation of the abduction and candidate items is outlined in Figure 4, where numbers $1 \mathinner{..} 5$ on the left indicate the iterations of the main loop, while the labels on the right are the observations involved in $\mathcal{O}$. The graphs displayed in the center represent the genesis of the abduction items (ellipses), namely (x, δ), and the candidate items in $\mathcal{C}$ (rectangles), namely (x, δ, o). Specifically, an arc from a node $\Im$ (an abduction item) to a node α (either an abduction item $\Im'$ or a candidate item), which is possibly marked with an observation o and/or a fault f, indicates that α is somewhat derived from a triple $(x', f, o) \in \mathit{Front}(\alpha)$ (line 8). The figures of each iteration are listed in Table 2, specifically, the initial instance of stack $\mathcal{U}$, the instance of sets $\mathcal{G}$ and $\mathcal{C}$ at the end of the nested loop (line 22), the newly-occurred observation o' (line 35), and the instance of the set $\mathbb{D}'$ of state d' (line 35), which is in fact the new candidate set appended to $\boldsymbol{\Delta}$ in line 36. Furthermore, depicted in Figure 5 is the part of minimalist diagnoser $\mathit{Mind}(\mathcal{X}_w)$ materialized in $\mathcal{M}$ by the algorithm based on $\mathcal{O}$. Details of each iteration are given below.

Iteration 1

Initially, the only item in $\mathcal{U}$ is $\Im_0 = (0, \emptyset)$, which is in fact popped from $\mathcal{U}$ in line 6 at the first iteration of the main loop. According to Figure 3, the frontier of $\Im_0$ includes two triples: $(1, \varepsilon, \mathbf{p})$ and $(2, \mathit{pfo}, \varepsilon)$. Since the former involves observation $\mathbf{p}$, there is an arc from abduction item $(0, \emptyset)$ to candidate item $(1, \emptyset, \mathbf{p})$, which is inserted into $\mathcal{C}$ (line 19). The latter, instead, leads to a new abduction item $\Im' = (2, \{\mathit{pfo}\})$, which is pushed onto $\mathcal{U}$ (line 16). Note that each arc is marked with the same labels marking the corresponding arc in Figure 3. The processing of the new abduction item $(2, \{\mathit{pfo}\})$ leads to the creation of candidate items $(0, \{\mathit{pfo}\}, \mathbf{b})$ and $(3, \{\mathit{pfo}, \mathit{bop}\}, \mathbf{b})$, which are inserted into $\mathcal{C}$. The relevant instances of $\mathcal{U}$, $\mathcal{G}$, and $\mathcal{C}$ pertaining to the first iteration are listed in the first line of Table 2. At the end of the nested loop (line 22), the transition function of state d_0 is generated based on the partition $\mathbf{C}$ of $\mathcal{C}$, which leads to the creation of the new states d_1 and d_2 in $\mathcal{M}$, along with

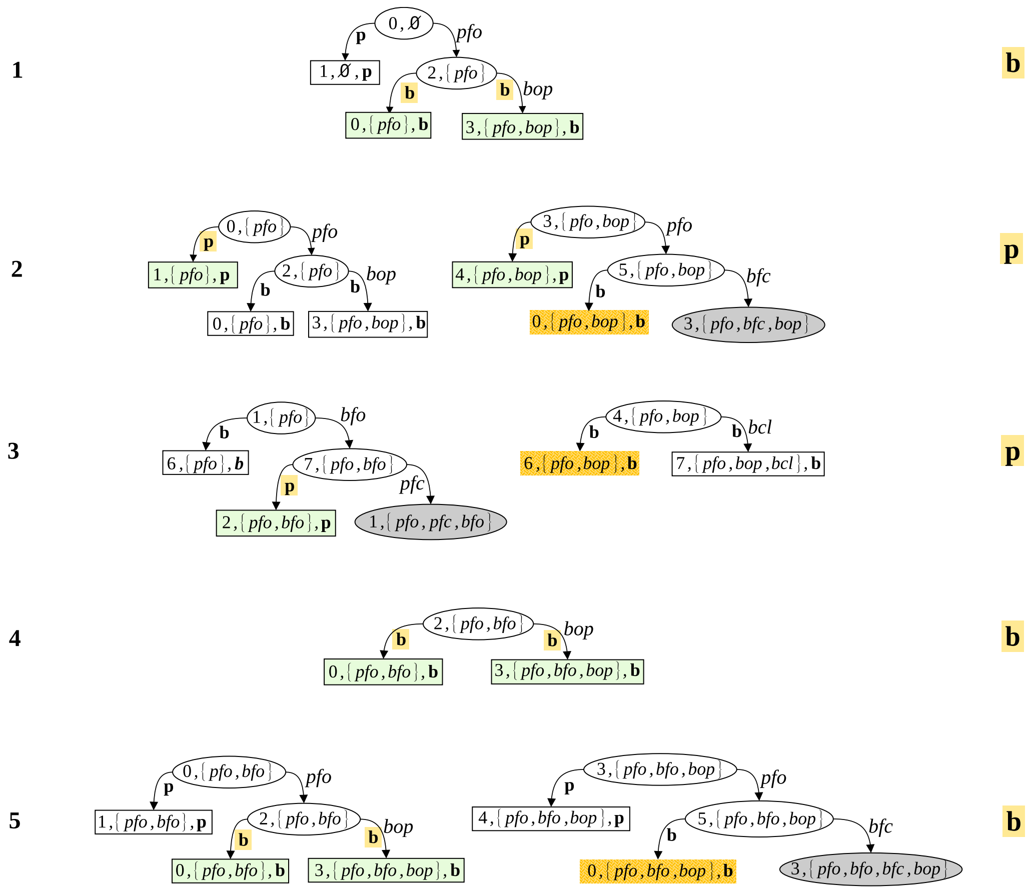

Figure 4 Trace of MINIMALIST DIAGNOSIS ENGINE applied to $\mathcal{O} = [\mathbf{b}, \mathbf{p}, \mathbf{p}, \mathbf{b}, \mathbf{b}]$ of DES $\mathcal{X}_w$.

their entering transitions $\langle d_0, \mathbf{p}, d_1 \rangle$ and $\langle d_0, \mathbf{b}, d_2 \rangle$. Note how field $\mathbb{D}$ in state d_2 includes only the minimal diagnosis $\{pfo\}$, which is a subset of diagnosis $\{pfo, bop\}$ within the other candidate item associated with observation $\mathbf{p}$. Then, at the occurrence of the first observation $\mathbf{b}$ (line 35), the candidate set $\mathbb{D}' = \{\{pfo\}\}$ is extracted from state d_2 and appended to $\mathbf{\Delta}$ (which is initially empty). At the end of the iteration (line 37), $\mathcal{G}$ and $\mathcal{U}$ are initialized with $\mathbb{I}' = \{(0, \{pfo\}), (3, \{pfo, bop\})\}$, as displayed in the first column of the second iteration in Table 2, while the current state d of $\mathcal{M}$ is set to d_2.

Iteration 2

At the second iteration, the two initial items included in $\mathcal{U}$ are the roots of the two graphs in position 2 in Figure 4, namely $(0, \{pfo\})$ and $(3, \{pfo, bop\})$. The processing of $(0, \{pfo\})$ leads to the generation of a candidate item $(1, \{pfo\}, \mathbf{p})$ in $\mathcal{C}$, and a new abduction item $(2, \{pfo\})$. The subsequent processing of $(2, \{pfo\})$ generates two candidate items, namely $(0, \{pfo\}, \mathbf{b})$ and $(3, \{pfo, bop\}, \mathbf{b})$. On its part, item $(3, \{pfo, bop\})$ generates the candidate item $(4, \{pfo, bop\}, \mathbf{p})$ and a new abduction item $(5, \{pfo, bop\})$, whose processing leads to the creation of the candidate item $(0, \{pfo, bop\}, \mathbf{b})$ and a new abduction item $(3, \{pfo, bfc, bop\})$. On the one hand, since $\mathcal{C}$ includes candidate item $(0, \{pfo\}, \mathbf{b})$, where $\{pfo\} \subset \{pfo, bop\}$, condition in line 18 is not fulfilled, thus $(0, \{pfo, bop\}, \mathbf{b})$ is ignored. On the other, since $\mathcal{G}$ includes abduction item $(3, \{pfo, bop\})$, which is the root of the second graph, where

Table 2 Data structures of Minimalist Diagnosis Engine with $\mathcal{O} = [\mathbf{b}, \mathbf{p}, \mathbf{p}, \mathbf{b}, \mathbf{b}]$ of $\mathcal{X}_w$.

i	$\mathbb{I}$	$\mathcal{G}$	$\mathcal{C}$	o'	$\mathbb{D}'$
1	$(0, \emptyset)$	$(0, \emptyset)$ $(2, \{\mathit{pfo}\})$	$(1, \emptyset, \mathbf{p})$ $(0, \{\mathit{pfo}\}, \mathbf{b})$ $(3, \{\mathit{pfo}, \mathit{bop}\}, \mathbf{b}))$	$\mathbf{b}$	$\{\{\mathit{pfo}\}\}$
2	$(0, \{\mathit{pfo}\})$ $(3, \{\mathit{pfo}, \mathit{bop}\})$	$(0, \{\mathit{pfo}\})$ $(2, \{\mathit{pfo}\})$ $(3, \{\mathit{pfo}, \mathit{bop}\})$ $(5, \{\mathit{pfo}, \mathit{bop}\})$ $(3, \{\mathit{pfo}, \mathit{bfc}, \mathit{bop}\})$	$(1, \{\mathit{pfo}\}, \mathbf{p})$ $(0, \{\mathit{pfo}\}, \mathbf{b})$ $(3, \{\mathit{pfo}, \mathit{bop}\}, \mathbf{b})$ $(4, \{\mathit{pfo}, \mathit{bop}\}, \mathbf{p})$	$\mathbf{p}$	$\{\{\mathit{pfo}\}\}$
3	$(1, \{\mathit{pfo}\})$ $(4, \{\mathit{pfo}, \mathit{bop}\})$	$(1, \{\mathit{pfo}\})$ $(4, \{\mathit{pfo}, \mathit{bop}\})$ $(7, \{\mathit{pfo}, \mathit{bfo}\})$ $(1, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\})$	$(6, \{\mathit{pfo}\}, \mathbf{b})$ $(2, \{\mathit{pfo}, \mathit{bfo}\}, \mathbf{p})$ $(7, \{\mathit{pfo}, \mathit{bop}, \mathit{bcl}\}, \mathbf{b})$	$\mathbf{p}$	$\{\{\mathit{pfo}, \mathit{bfo}\}\}$
4	$(2, \{\mathit{pfo}, \mathit{bfo}\})$	$\{(2, \{\mathit{pfo}, \mathit{bfo}\})\}$	$(0, \{\mathit{pfo}, \mathit{bfo}\}, \mathbf{b})$ $(3, \{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\}, \mathbf{b})$	$\mathbf{b}$	$\{\{\mathit{pfo}, \mathit{bfo}\}\}$
5	$(0, \{\mathit{pfo}, \mathit{bfo}\})$ $(3, \{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\})$	$(0, \{\mathit{pfo}, \mathit{bfo}\})$ $(3, \{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\})$ $(2, \{\mathit{pfo}, \mathit{bfo}\})$ $(5, \{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\})$ $(3, \{\mathit{pfo}, \mathit{bfo}, \mathit{bfc}, \mathit{bop}\})$	$(1, \{\mathit{pfo}, \mathit{bfo}\}, \mathbf{p})$ $(0, \{\mathit{pfo}, \mathit{bfo}\}, \mathbf{b})$ $(3, \{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\}, \mathbf{b})$ $(4, \{\mathit{pfo}, \mathit{bfo}, \mathit{bop}\}, \mathbf{p})$	$\mathbf{b}$	$\{\{\mathit{pfo}, \mathit{bfo}\}\}$

$\{\mathit{pfo}, \mathit{bop}) \subset \{\mathit{pfo}, \mathit{bfc}, \mathit{bop}\}$, abduction item $(3, \{\mathit{pfo}, \mathit{bfc}, \mathit{bop}\})$ is marked in $\mathcal{G}$ (line 13), so that it will be discarded without being processed when considered subsequently. Since now $\mathcal{U}$ is empty (line 22), the nested loop terminates. The candidate abduction items relevant to observation $\mathbf{p}$ lead to the creation of the new state d_3 in $\mathcal{M}$, along with its entering transition. Those relevant to $\mathbf{b}$, instead, lead to the set $\mathbb{I}_j = \{(0, \{\mathit{pfo}\}), (3, \{\mathit{pfo}, \mathit{bop}\})\}$, which identifies state d_2. Hence, the target state of the transition exiting d_2 and marked by $\mathbf{b}$ is already in $\mathcal{M}$, so that a transition $\langle d_2, \mathbf{b}, d_2 \rangle$ can be created immediately in line 32. Then, at the occurrence of the second observation $\mathbf{p}$ (line 35), the candidate set $\mathbb{D}' = \{\{\mathit{pfo}\}\}$ is extracted from state d_3 and appended to $\mathbf{\Delta}$. At the end of the iteration (line 37), $\mathcal{G}$ and $\mathcal{U}$ are initialized with $\mathbb{I}' = \{(1, \{\mathit{pfo}\}), (4, \{\mathit{pfo}, \mathit{bop}\})\}$, as displayed in the first column of the third iteration in Table 2, while the current state d of $\mathcal{M}$ is set to d_3.

Iteration 3

At the third iteration, the graph on the left (Figure 4, position 3) generates two candidate items in $\mathcal{C}$: $(6, \{\mathit{pfo}\}, \mathbf{b})$ and $(2, \{\mathit{pfo}, \mathit{bfo}\}, \mathbf{p})$. Note that the abduction item $(1, \{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\})$ is pruned on the grounds that the relevant diagnosis is a superset of the diagnosis in the root (sharing the same state 1), namely $\{\mathit{pfo}, \mathit{pfc}, \mathit{bfo}\} \supset \{\mathit{pfo}\}$. The second graph (on the right) generates a candidate item $(6, \{\mathit{pfo}, \mathit{bop}\}, \mathbf{b})$, which, however, contrasts with $(6, \{\mathit{pfo}\}, \mathbf{b})$, which is already in $\mathcal{C}$ (cf. the first graph on the left); thus, this candidate item is not inserted into $\mathcal{C}$, as it was for the candidate item $(0, \{\mathit{pfo}, \mathit{bop}\}, \mathbf{b})$ in the second iteration. The second candidate item generated in the same graph, instead, namely $(7, \{\mathit{pfo}, \mathit{bop}, \mathit{bcl}\}, \mathbf{b})$, is inserted into $\mathcal{C}$. Eventually, $\mathcal{C}$ includes three candidate items (cf. third row in Table 2). When $\mathcal{U}$ becomes empty (line 22), the candidate abduction items relevant to observation $\mathbf{b}$ lead to the creation of the new state d_4 in $\mathcal{M}$, while those relevant to $\mathbf{p}$ lead to the creation of state d_5, along with corresponding entering transitions. Then, at the occurrence of the third observation $\mathbf{p}$ (line 35), the candidate set $\mathbb{D}' = \{\{\mathit{pfo}, \mathit{bfo}\}\}$ is extracted from state

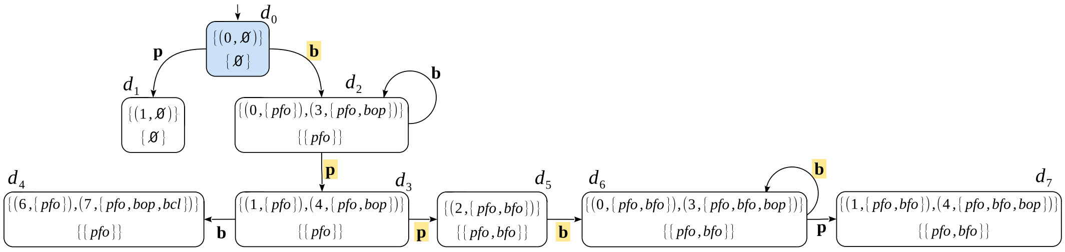

Figure 5 Lazy generation of minimalist diagnoser $\mathit{Mind}(\mathcal{X}_w)$ based on $\mathcal{O} = [\mathbf{b},\mathbf{p},\mathbf{p},\mathbf{b},\mathbf{b}]$ of $\mathcal{X}_w$.

d_5 and appended to $\mathbf{\Delta}$. At the end of the iteration (line 37), $\mathcal{G}$ and $\mathcal{U}$ are initialized with $\mathbb{I}' = \{(2,\{\mathit{pfo},\mathit{bfo}\})\}$, as displayed in the first column of the fourth iteration in Table 2, while the current state d of $\mathcal{M}$ is set to d_5.

Iteration 4

At the fourth iteration, the only initial item in $\mathcal{U}$ is $(2,\{\mathit{pfo},\mathit{bfo}\})$, which is also the root of the graph in Figure 4 (position 4). Now, since both transitions exiting state 2 in $\mathit{Space}(\mathcal{X}_w)$ are observable, two candidate items are generated and inserted into $\mathcal{C}$: $(0,\{\mathit{pfo},\mathit{bfo}\},\mathbf{b})$ and $(3,\{\mathit{pfo},\mathit{bfo},\mathit{bop}\},\mathbf{b})$. Since $\mathcal{U}$ is now empty, the partition $\mathbf{C}$ in line 23 is in fact a singleton, which includes $\mathcal{C}$ (both candidate items in $\mathcal{C}$ are associated with observation $\mathbf{b}$). Thus, a new state d_6 is generated in $\mathcal{M}$, along with transition $\langle d_5,\mathbf{b},d_6\rangle$. At the occurrence of the fourth observation $\mathbf{b}$ (line 35), the candidate set $\mathbb{D}' = \{\{\mathit{pfo},\mathit{bfo}\}\}$ is extracted from state d_6 and appended to $\mathbf{\Delta}$. At the end of the iteration (line 37), $\mathcal{G}$ and $\mathcal{U}$ are initialized with $\mathbb{I}' = \{(0,\{\mathit{pfo},\mathit{bfo}\}),(3,\{\mathit{pfo},\mathit{bfo},\mathit{bop}\})\}$, as displayed in the first column of the fifth iteration in Table 2, while the current state d of $\mathcal{M}$ is set to d_6.

Iteration 5

In the fifth iteration, the two graphs rooted in the abduction items $(0,\{\mathit{pfo},\mathit{bfo}\})$ and $(3,\{\mathit{pfo},\mathit{bfo},\mathit{bop}\})$, as outlined in the last row in Figure 4 and Table 2, involve the generation of abduction items $(2,\{\mathit{pfo},\mathit{bfo}\})$, $(5,\{\mathit{bfo},\mathit{pfo},\mathit{bop}\})$, and $(3,\{\mathit{pfo},\mathit{bfo},\mathit{bfc},\mathit{bop}\})$, the last of which is marked in $\mathcal{G}$ because the relevant diagnosis is a superset of the diagnosis inherent to the root, namely $\{\mathit{pfo},\mathit{bfo},\mathit{bop}\}$. Furthermore, of the five candidate items generated, $(0,\{\mathit{pfo},\mathit{bfo},\mathit{bop}\},\mathbf{b})$ is not inserted into $\mathcal{C}$ because the latter includes candidate item $(0,\{\mathit{pfo},\mathit{bfo}\},\mathbf{b})$, where $\{\mathit{pfo},\mathit{bfo}\} \subset \{\mathit{pfo},\mathit{bfo},\mathit{bop}\}$. Then, partition $\mathbf{C}$ in line 23 leads to the generation of a new state d_7 in $\mathcal{M}$, along with a new transition $\langle d_6,\mathbf{p},d_7\rangle$. Since, however, the part associated with observation $\mathbf{b}$ leads to $\mathbb{I}'_j = \{(0,\{\mathit{pfo},\mathit{bfo}\}),(3,\{\mathit{pfo},\mathit{bfo},\mathit{bop}\})$, which equals the field $\mathbb{I}$ of the same state d_6, a new auto-transition $\langle d_6,b,d_6\rangle$ is generated without the need to materialize the (existing) target state d_6. At the occurrence of the fifth observation $\mathbf{b}$ (line 35), the candidate set $\mathbb{D}' = \{\{\mathit{pfo},\mathit{bfo}\}\}$ is extracted from state d_6 and appended to $\mathbf{\Delta}$. At the end of the iteration (line 37), the current state d of $\mathcal{M}$ is still d_6.

Since $\mathcal{O}$ includes five observations, the sequence of diagnosis sets generated by Minimalist Diagnosis Engine is $\mathbf{\Delta} = [\{\{\mathit{pfo}\}\},\{\{\mathit{pfo}\}\},\{\{\mathit{pfo},\mathit{bfo}\}\},\{\{\mathit{pfo},\mathit{bfo}\}\},\{\{\mathit{pfo},\mathit{bfo}\}\}]$. Note how, at the fourth observation, $\mathbf{\Delta}$ equals the temporal candidate set anticipated in Example 13. Remarkably, as claimed in Proposition 21, this is no coincidence.

4.2 Correctness of the algorithm

▶ **Proposition 21.** *Algorithm* Minimalist Diagnosis Engine *is correct.*

Proof. According to Corollary 19, it suffices to show that the part of $\mathit{Mind}(\mathcal{X}) = (\mathbf{O}, D, \tau, d_0)$ materialized in $\mathcal{M}$ is correct, which can be proven by induction on the transition function τ.

(*Basis*) *The initial state d_0 generated in line 2 is correct.* In fact, $\Im_0 = (x_0, \emptyset)$, $d_0 = (\{\Im_0\}, \{\emptyset\}) = (\{(x_0, \emptyset)\}, \{\emptyset\})$, which equals the initial state of $\mathit{Mind}(\mathcal{X})$ in Definition 16.

(*Induction*) *If $d = (\mathbb{I}, \mathbb{D})$ is a state materialized in $\mathcal{M}$, then all transitions $\langle d, o'_j, d_j \rangle$ generated in line 32 are correct.* Based on the abduction items in $\mathbb{D}$, which have been pushed onto $\mathcal{U}$ in line 37, the nested loop in line 5–22 is bound to generate in $\mathcal{C}$ all the candidate items for every possible next observation $o'_1, \ldots, o'_k$. To this end, an unmarked abduction item (that is, an abduction item neither pruned not yet processed) at a time is popped from $\mathcal{U}$, and its successive abduction items $\Im' = (x', \delta')$, which are relevant to its frontier, are computed (line 9). If the involved component transition from x to x' is unobservable ($o = \varepsilon$), then, provided that $\Im'$ has not been generated already (line 11), some abduction items are possibly marked in lines 12–16 in order to prune the trajectories that will not contribute to the instances of $\mathbb{D}_j$ in the newly-materialized states d_j of $\mathcal{M}$. This pruning is dictated by efficiency reasons only: if not performed, the processing in lines 18–20 still allows for the eventual generation of the correct candidate set $\mathbb{D}_j$, but at the expense of useless computation of additional abduction items in $\mathbb{I}_j$. In fact, if condition in line 12 is true, then any abduction item derived from (x', δ') cannot have a diagnosis that is a subset of an abduction item derived from $(x', \bar{\delta})$, as all the additional faults involved in an extension of δ' are also involved in an extension of $\bar{\delta}$, precisely because the suffix of a trajectory starting in x' depends only on x', not on the diagnosis associated in the abduction item. If, instead, condition in line 12 is false, then we need to mark every abduction item $(x', \bar{\delta})$ where $\bar{\delta} \supset \delta'$, for the same reason of keeping only abduction items involving the same state of $\mathcal{X}$ that have the minimal diagnosis. In contrast with the processing in lines 14–16, however, we do not impose in this case the condition $x' \neq x$. In fact, in conformity with Definition 16, once a candidate item (x', δ', o) is inserted into $\mathcal{C}$ (line 19), no other successive item needs to be generated (starting from its frontier). Hence, the removal in line 20 is necessary even if $x' = x$. When $\mathcal{U}$ becomes empty in line 22 (end of the nested loop), it means that there is no other abduction or candidate item that can be generated (we have reached the completeness). In line 23, the partition $\mathbf{C} = \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$ allows us to associate with each possible next observation o'_j the relevant candidate items (x, δ, o'_j). In other words, each part $\mathcal{C}_j \in \mathbf{C}$ contains the information for generating the field $\mathbb{I}_j$ of state d_j in transition $\langle d, o'_j, d_j \rangle$, which is performed in lines 24–32. Specifically, $\mathbb{I}_j$ is distilled from $\mathcal{C}_j$ and, if there is no state d_j in $\mathcal{M}$ identified by $\mathbb{I}_j$, then it is generated by first computing the set $\mathbb{D}_j$ based on Definition 16, that is, as the set of minimal diagnoses in $\mathbb{I}_j$ (lines 26–31). Eventually, a transition $\langle d, o'_j, d_j \rangle$ is generated in $\mathcal{M}$. Since, based on the argumentation expressed above, the set $\mathbb{I}_j$ of abduction items computed by the algorithm equals the first field of the target state d'_j of the transition exiting d and marked by observation o'_j, and the set $\mathbb{D}_j$ equals the second field of d'_j, the transition function of d materialized in $\mathcal{M}$ is correct. ◀

5 Related work

The work presented in this paper stems from the active system approach [13, 16], which represents DESs by means of complete explicit (i.e. operational) strong component models and performs abduction-based diagnosis. More specifically, in this paper a DES is a network of communicating finite automata. Other explicit models for DESs are Petri nets [1, 3] and labeled time Petri nets [19]. Strong models can be adopted to represent dynamical systems also when consistency-based reasoning is performed. For instance, the diagnostic approach in [7], which is applied to synchronous sequential circuits, is consistency-based. Both the normal and abnormal behavior (fault modes) of system components are described by means of the Finite Trace Next Logic, and minimum-cardinality diagnoses are computed. Weak models and consistency-based diagnosis are applied to DESs in [18]. There are contributions about diagnostic reasoning that ignore any explicit DES models, instead, they consider some specifications. The specification of a dynamical system (i.e. the properties the system has to exhibit over time) can be given as a formula in a temporal logic [6], such as Linear Temporal Logic (LTL) [22]. An LTL specification is the implicit representation of an automaton, where LTL operators describe the state transitions. Typical diagnostic tasks are aimed at finding out whether a given behavioral evolution, called a *trace*, satisfies the specification formula and/or uncovering the causes for a trace violates the specification formula, where such causes can be searched for either in the trace [2] or in the specification [20, 21], that is, the specification may be wrong.

To the best of our knowledge, the first work that adopted subset-minimal diagnoses for DESs is [27], where, however, their appropriateness is not investigated: minimal diagnoses are reckoned good as they are “more probable” and they reduce the cognitive load of a human operator. The method in [27], differently from the proposal presented here, is based on a total offline knowledge-compilation, aimed at generating a so-called *minimal diagnoser*. In [27], computing minimal candidates only does not translate into any pruning of the search space; on the contrary, the construction of the minimal diagnoser requires the previous generation the whole DES space, which is unfeasible for real systems since the number of states is exponential in the number of components. The process in this paper, instead, performs online a (partial dynamical) knowledge-compilation, if and when needed (laziness), and every observation-driven behavior reconstruction assumes the unavailability of the DES space, which is only instrumental to the formalization, but never materialized.

6 Conclusion

From the computational viewpoint, diagnosis of real DESs is a complex task because of the huge size of the DES space in which the search for the trajectories complying with a temporal observation is carried out. This is why the total compilation of a DES into a diagnoser is impractical, even when performed offline. Also the online reconstruction of the DES trajectories based on the given temporal observation is no panacea. From the user (or post-processing) side, analyzing all the candidates is possibly time consuming. Hence, in this paper, instead of generating all the diagnoses, each being a set of faults, we generate (subset) minimal diagnoses only. First, we have discussed the meaningfulness of minimal diagnoses for task of DES diagnosis (and, more specifically, for diagnosis during monitoring). Then, we have proposed a method that, at each newly perceived observation, updates the trajectories built up to the reception of the previous observation, while pruning the trajectories that either are not consistent with the new observation or are bound to bring to non-minimal candidates, thus reducing the size of the DES space to be explored. In addition, every time a new observation is perceived, the method updates (if needed) an (initially empty) compiled knowledge structure, called a *minimalist diagnoser*, so as to speed up the next abductive reasoning steps.

The method presented in this paper comes with expectations of gains in spatial and temporal performances with respect to the computation of all diagnoses; such expectations are being confirmed by an ongoing experimental activity.

A future version of the method could generate candidates both incrementally and in a sorted way based on a preference criterion, such as cardinality or likelihood, that is significant in the considered domain so as every candidate that is preferred to another is outputted before it. Further research could adapt the proposed algorithm to notions of DES diagnosis different from that adopted in this paper. For instance, a recent work [11] defines a DES diagnosis as the *sequence* of faults associated with a trajectory, namely a *temporal fault*. The adaptation would require defining the concept of *minimal temporal fault*, representing concisely the minimal temporal fault candidates, pruning the portions of the search space that lead to non-minimal temporal faults, and progressively updating the relevant sequence-oriented minimalist diagnoser.

In the literature the notion of a fault in a DES has been generalized to the violation of a predefined behavioral property, as in [8], or to a pattern of transitions, called *supervision pattern* [10]. A fault can also be defined as an event that arises when a sub-trajectory of the DES matches a given pattern, like in [14, 17]. Minimal candidates could in principle be computed also for generalized faults; their significance, however, needs some investigation.

References

1 F. Basile. Overview of fault diagnosis methods based on Petri net models. In *Proceedings of the 2014 European Control Conference, ECC 2014*, pages 2636–2642, 2014. doi:10.1109/ECC.2014.6862631.

2 I. Beer, S. Ben-David, H. Chockler, A. Orni, and R. Trefler. Explaining counterexamples using causality. In A. Bouajjani and O. Maler, editors, *21st International Conference on Computer-Aided Verification (CAV 2009)*, pages 94–108. Springer-Verlag, Berlin, Heidelberg, 2009. doi:10.1007/978-3-642-02658-4_11.

3 X. Cong, M.P. Fanti, A.M. Mangini, and Z. Li. Decentralized diagnosis by Petri nets and integer linear programming. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(10):1689–1700, 2018. doi:10.1109/TSMC.2017.2726108.

4 J. de Kleer, A. K. Mackworth, and R. Reiter. Characterizing diagnoses. In G. Gottlob and W. Nejdl, editors, *Expert Systems in Engineering, Principles and Applications*, pages 1–15. Springer, 1990. doi:10.1007/3-540-53104-1_27.

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

6 E.A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics*, pages 995–1072. MIT Press, Cambridge, MA, USA, 1990. doi:10.1016/B978-0-444-88074-1.50021-4.

7 Alexander Feldman, Ingo Pill, Franza Wotawa, Ion Matei, and Johan de Kleer. Efficient model-based diagnosis of sequential circuits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:2814–2821, 2020. doi:10.1609/aaai.v34i03.5670.

8 G. Göessler, T. Mari, Y. Pencolé, and L. Travé-Massuyès. Towards causal explanations of property violations in discrete event systems. In *30th International Workshop on Principles of Diagnosis (DX-19)*, Klagenfurt, Austria, 2019. URL: https://api.semanticscholar.org/CorpusID:226220644.

9 W. Hamscher, L. Console, and J. de Kleer, editors. *Readings in Model-Based Diagnosis*. Morgan Kaufmann, San Mateo, CA, 1992.

10 T. Jéron, H. Marchand, S. Pinchinat, and M.O. Cordier. Supervision patterns in discrete event systems diagnosis. In *Workshop on Discrete Event Systems (WODES 2006)*, pages 262–268, Ann Arbor, MI, 2006. IEEE Computer Society. URL: https://api.semanticscholar.org/CorpusID:526355.

11 G. Lamperti, S. Trerotola, M. Zanella, and X. Zhao. Sequence-oriented diagnosis of discrete-event systems. *Journal of Artificial Intelligence Research*, 78:69–141, 2023. doi:10.1613/jair.1.14630.

12 G. Lamperti and M. Zanella. Diagnosis of discrete-event systems from uncertain temporal observations. *Artificial Intelligence*, 137(1–2):91–163, 2002. doi:10.1016/S0004-3702(02)00123-6.

13 G. Lamperti and M. Zanella. *Diagnosis of Active Systems: Principles and Techniques*, volume 741 of *Springer International Series in Engineering and Computer Science*. Springer, Dordrecht, Netherlands, 2003. doi:10.1007/978-94-017-0257-7.

14 G. Lamperti and M. Zanella. Context-sensitive diagnosis of discrete-event systems. In T. Walsh, editor, *Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI 2011)*, volume 2, pages 969–975, Barcelona, Spain, 2011. AAAI Press. doi:10.5591/978-1-57735-516-8/IJCAI11-167.

15 G. Lamperti and M. Zanella. Monitoring of active systems with stratified uncertain observations. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 41(2):356–369, 2011. doi:10.1109/TSMCA.2010.2069096.

16 G. Lamperti, M. Zanella, and X. Zhao. *Introduction to Diagnosis of Active Systems*. Springer, Cham, 2018. doi:10.1007/978-3-319-92733-6.

17 G. Lamperti and X. Zhao. Diagnosis of active systems by semantic patterns. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(8):1028–1043, 2014. doi:10.1109/TSMC.2013.2296277.

18 Y. Pencolé, G. Steinbauer, C. Mühlbacher, and L. Travé-Massuyès. Diagnosing discrete event systems using nominal models only. In M. Zanella, I. Pill, and A. Cimatti, editors, *28th International Workshop on Principles of Diagnosis (DX'17)*, volume 4, pages 169–183. Kalpa Publications in Computing, 2018. doi:10.29007/1d2x.

19 Y. Pencolé, A. Subias, and C. Coquand. A model checking method to solve the event pattern diagnosis problem in safe labeled time Petri nets. In *32nd International Workshop on Principles of Diagnosis (DX-21)*, Hamburg, Germany, 2021. URL: https://laas.hal.science/hal-03348338v1.

20 Ingo Pill and Thomas Quaritsch. Behavioral diagnosis of LTL specifications at operator level. In *Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI 2013)*, pages 1053–1059. AAAI Press, 2013. URL: http://dl.acm.org/citation.cfm?id=2540128.2540280.

21 Ingo Pill and Franz Wotawa. Automated generation of (F)LTL oracles for testing and debugging. *Journal of Systems and Software*, 139(C):124–141, 2018. doi:10.1016/J.JSS.2018.02.002.

22 Amir Pnueli. The temporal logic of programs. In *8th Annual Symposium on Foundations of Computer Science*, SFCS '77, pages 46–57, Washington, DC, USA, 1977. IEEE Computer Society. doi:10.1109/SFCS.1977.32.

23 D. Poole. Normality and faults in logic-based diagnosis. In *Eleventh International Joint Conference on Artificial Intelligence (IJCAI 1989)*, pages 1304–1310, Detroit, MI, 1989. URL: http://ijcai.org/Proceedings/89-2/Papers/073.pdf.

24 R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95, 1987. doi:10.1016/0004-3702(87)90062-2.

25 Patrick Rodler. *How Should I Compute My Candidates? A Taxonomy and Classification of Diagnosis Computation Algorithms*, pages 1986–1993. IOS Press, 2023. doi:10.3233/faia230490.

26 M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D.C. Teneketzis. Diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control*, 40(9):1555–1575, 1995. doi:10.1109/9.412626.

27 X. Zhao, D. Ouyang, G. Lamperti, and X. Tong. Minimal diagnosis and diagnosability of discrete-event systems modeled by automata. *Complexity*, 2020:1–17, 2020. doi:10.1155/2020/4306261.