24th European Conference on Artificial Intelligence - ECAI 2020
Santiago de Compostela, Spain

Diagnosis of Temporal Faults in Discrete-Event Systems

Nicola Bertoglio[1] and **Gianfranco Lamperti**[1] and **Marina Zanella**[1] and **Xiangfu Zhao**[2]

Abstract. Model-based diagnosis of discrete-event systems (DESs) generates a set of *candidates* upon the reception of a temporal observation. In the literature, a candidate is a *set* of faults produced by a trajectory of the DES that is consistent with the temporal observation. As such, a candidate does not convey any temporal relationship between faults, nor does it account for multiple occurrences of the same fault. To overcome the limitations of this set-oriented approach to diagnosis of DESs, the novel notions of *temporal fault* and *temporal diagnosis* are proposed, along with two diagnosis techniques. A temporal fault is the (possibly unbounded) sequence of faults produced by a trajectory. A temporal diagnosis is a (possibly infinite) set of temporal faults. Hence, in this new temporal-oriented approach to diagnosis of DESs, a candidate is a temporal fault. The fact that a temporal diagnosis turns out to be a regular language is key to coping with the infinity of candidates, which can be represented by a regular expression. The diagnosis task can be performed either by restricting the DES space to the trajectories that are consistent with the temporal observation, or by exploiting a *temporal diagnoser* which allows for fast online diagnosis. The claim of this paper is that the extra temporal information embedded in candidates may be essential in taking critical decisions based on the diagnosis results.

1 INTRODUCTION

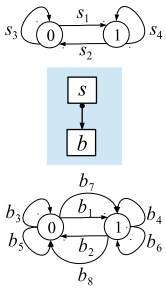
Diagnosis aims at finding the causes of the abnormal behavior of a system based on the observations relevant to its operation that are perceived from the outside. In the Artificial Intelligence (AI) community, the definition of the task [25, 10] led to the model-based paradigm [11], according to which the normal behavior of the system to be diagnosed is described by a model, and to the notion of *consistency-based* diagnosis, which was initially conceived for static systems, such as combinational circuits, and later applied also to dynamical systems [29, 23]. Consistency-based diagnosis produces as output a collection of sets of *faulty* components: each set, called a *candidate*, explains the given observation in that assuming that all the components in the candidate are not behaving normally and all the others are behaving normally is consistent with the observation.

For diagnosing a dynamical system, a discrete-event system (DES) model can be adopted [8], this being either a Petri net [2, 30, 9, 24, 21] or a net of communicating finite automata [6], an automaton for each component, like in the current paper. Being untimed, these models do not explicitly consider any time length, with the only temporal information being relevant to the reciprocal order of the transitions. Although consistency-based diagnosis is applicable to DESs by modeling their normal behavior only [22], a DES specification usually involves also its abnormal behavior, as proposed in the seminal work by Sampath et al. [27, 28, 26], where each state transition in the automaton relevant to a DES component is either normal or abnormal. The input of the diagnosis task for a DES is a sequence of observations, called hereafter a *temporal observation*, which are listed in relative temporal order. The output is a set of *candidates*, with each candidate being a set of *faults*, where a fault is associated with an abnormal state transition. A candidate is inherent to some global state of the DES (namely, a state composed of the states of all the DES components), which means that, once the temporal observation has been gathered, the DES is possibly in that global state and the faults in the relevant candidate have occurred. Given the perceived temporal observation, the whole set of global states that the DES may have reached starting from the initial state is the current *belief state*. Albeit the current actual state of the DES falls in the belief state, generally speaking, we cannot set the former apart. Diagnosing a DES becomes a form of *abductive* reasoning, inasmuch the candidates are generated based on the trajectories (sequences of state transitions) of the DES that entail the temporal observation. The approach in [27] relies on a *diagnoser*, a data structure that is derived from the global DES model in a preprocessing phase performed offline. The diagnoser is exploited online in order to generate a new set of candidates upon perceiving each observation (*monitoring-based diagnosis*) or just one set of candidates, given the whole sequence of observations (*a posteriori diagnosis*).

If the diagnoser approach is still a theoretical reference framework for the definition of diagnosis and diagnosability of DESs represented as finite automata, a computationally more viable alternative is the *active-system* approach [1, 13, 14, 16, 18], which does not require building the global DES model, an impractical task for real DESs. The rationale behind the traditional active-system approach is to perform the abduction online, a possibly costly operation that, however, being driven by the temporal observation, can only focus on the trajectories that produce this sequence. Still, in either approach, a candidate is a *set* of faults. Consequently, the diagnosis output is devoid of any temporal information while in the real world faults occur in a specific temporal order. One may argue that, in monitoring-based diagnosis, since a new set of candidates is output upon the reception of a new observation, it is possible to ascertain whether some additional faults have occurred with respect to the previous observation. Still, even in monitoring-based diagnosis one cannot ascertain whether a fault occurred previously has occurred again, in other words, no information about *intermittent* faults is provided.

In a perspective of explainable diagnosis and, more generally, of explainable AI, this paper proposes a novel technique for diagnosis of DESs based on the notion of a *temporal fault*. In the active-system approach, which this paper stems from, a candidate is the set of faults relevant to a trajectory of the DES. The diagnosis output is the set of candidates relevant to the (possibly infinite) set of trajectories of

[1] Department of Information Engineering, University of Brescia, Italy, email: {n.bertoglio001, gianfranco.lamperti, marina.zanella}@unibs.it
[2] School of Computer and Control Engineering, Yantai University, China, email: xiangfuzhao@gmail.com



Component transition	*Description*
$s_1 = \langle 0, (ko, \{op\}), 1\rangle$	The sensor detects a threatening external event *ko* and generates the open event
$s_2 = \langle 1, (ok, \{cl\}), 0\rangle$	The sensor detects a liberating external event *ok* and generates the close event
$s_3 = \langle 0, (ko, \{cl\}, 0\rangle$	The sensor detects a threatening external event *ko*, yet generates the close event
$s_4 = \langle 1, (ok, \{op\}), 1\rangle$	The sensor detects a liberating external event *ok*, yet generates the open event
$b_1 = \langle 0, (op, \emptyset), 1\rangle$	The breaker reacts to the open internal event by opening
$b_2 = \langle 1, (cl, \emptyset), 0\rangle$	The breaker reacts to the close internal event by closing
$b_3 = \langle 0, (op, \emptyset), 0\rangle$	The breaker does not react to the open internal event and remains closed
$b_4 = \langle 1, (cl, \emptyset), 1\rangle$	The breaker does not react to the close internal event and remains open
$b_5 = \langle 0, (cl, \emptyset), 0\rangle$	The breaker reacts to the close internal event by remaining closed
$b_6 = \langle 1, (op, \emptyset), 1\rangle$	The breaker reacts to the open internal event by remaining open
$b_7 = \langle 0, (op, \emptyset), 1\rangle$	The breaker reacts to the close internal event by opening
$b_8 = \langle 1, (op, \emptyset), 0\rangle$	The breaker reacts to the open internal event by closing

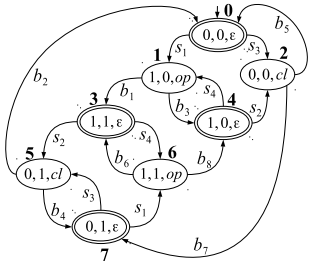


Figure 1: DES $\mathcal{P}$ (left), details of component transitions (center), and space $\mathcal{P}^*$ (right), where the the states 0, 3, 4, and 7 are final.

the DES that produce the temporal observation. Since the domain of faults is finite, both the candidates and the diagnosis output are finite and bounded.

In this paper, instead, a candidate is a temporal fault, namely the (possibly unbounded) sequence of faults relevant to a trajectory that produces the temporal observation. Consequently, the diagnosis output, called a *temporal diagnosis*, is the (possibly infinite) set of temporal faults relevant to the (possibly infinite) set of trajectories of the DES that imply the temporal observation. As such, a temporal fault differs from a classical candidate mainly in two ways: (*i*) a temporal fault includes the *multiset* of faults occurred in the trajectory, where several (possibly an unbounded number of) occurrences of the same fault are encompassed, and (*ii*) in a temporal fault, the relative temporal order of the occurrences of each fault is clearly shown. Since a temporal diagnosis (set of temporal faults) is a regular language, it can be represented by a regular expression defined on the alphabet of faults. In other words, each temporal fault of a temporal diagnosis is a string of the language of a regular expression defining the temporal diagnosis.

The temporal aspect characterizing the temporal faults can be important for ranking purposes and for helping the diagnostician in taking critical decisions based on richer information, typically in order to perform specific repair actions on the DES. The paper proposes two methods for generating the temporal diagnosis of a temporal observation. The first method, presented in Section 3, requires the online reconstruction of the DES behavior that is constrained by the temporal observation and, then, generates the regular expression of the temporal diagnosis based on this behavior. The second method, presented in Section 4 and Section 5, relies on a data structure compiled offline, called a *temporal diagnoser*, which allows for the online efficient computation of the temporal diagnosis.

t	o	f
s_1	*act*	ε
s_2	*sby*	ε
s_3	ε	$\mathbf{f_1}$
s_4	ε	$\mathbf{f_2}$
b_1	*opn*	ε
b_2	*cls*	ε
b_3	ε	$\mathbf{f_3}$
b_4	ε	$\mathbf{f_4}$
b_5	*nop*	ε
b_6	*nop*	ε
b_7	*opn*	$\mathbf{f_5}$
b_8	*cls*	$\mathbf{f_6}$

o	*Observation description*
act	The sensor is in activation
sby	The sensor is in standby
opn	The breaker opens
cls	The breaker closes
nop	The breaker performs no operation

f	*Fault description*
$\mathbf{f_1}$	The sensor sends the *cl* command instead of *op*
$\mathbf{f_2}$	The sensor sends the *op* command instead of *cl*
$\mathbf{f_3}$	The breaker remains closed on the *op* command
$\mathbf{f_4}$	The breaker remains open on the *cl* command
$\mathbf{f_5}$	The breaker opens on the *cl* command
$\mathbf{f_6}$	The breaker closes on the *op* command

Figure 2: Mapping table $\mu(\mathcal{P})$ (left) and symbol description (right).

2 SYSTEM MODELING

A DES $\mathcal{X}$ is a network of *components*, where the behavior of each component is modeled as a communicating automaton [6]. A component is endowed with input and output *pins*, where each output pin is connected with an input pin of another component by a *link*. The way a transition is triggered in a component is threefold: (*a*) spontaneously, formally denoted by the empty event ε, (*b*) by an *external* event coming from outside $\mathcal{X}$, or (*c*) by an *internal* event coming from another component. Initially, $\mathcal{X}$ is *quiescent*, that is, all links are empty. When performing a transition, a component consumes the triggering (input) event and possibly generates new events on its output pins, which are bound to trigger the transitions of other components, thereby causing a cascade process which is assumed to terminate when no new event is generated and $\mathcal{X}$ becomes quiescent anew. A transition generating an event on an output pin can occur only if this pin is not occupied by another event. Assuming that only one component transition at a time can occur, the process that moves a DES from the initial state to a final quiescent state can be represented by a sequence of component transitions, called a *trajectory* of $\mathcal{X}$. At the occurrence of a component transition, $\mathcal{X}$ changes its state, with a state x of $\mathcal{X}$ being a pair (C, L), where C is the array of current states of components and L the array of the (possibly empty) current events placed in links. Formally, the (possibly infinite) set of trajectories of $\mathcal{X}$ is specified by a deterministic finite automaton (DFA), namely the *space* $\mathcal{X}^*$ of $\mathcal{X}$,

$$\mathcal{X}^* = \left(\Sigma, X, \tau, x_0, X_q\right) \tag{1}$$

where Σ (the alphabet) is the set of component transitions, X is the set of states, τ is the deterministic transition function mapping a state and a component transition into a new state, $\tau : X \times \Sigma \mapsto X$, x_0 is the initial state, and X_q is the set of final (quiescent) states.

For diagnosis purposes, the model of $\mathcal{X}$ needs to be enhanced by specifying its *observability* and *normality*. Based on a *mapping table*, each transition is defined either as *observable* or *unobservable* and, orthogonally, either as *normal* or *faulty*. Specifically, let $\mathbf{T}$ be the set of component transitions in $\mathcal{X}$, $\mathbf{O}$ a finite set of *observations*, and $\mathbf{F}$ a finite set of *faults*. The *mapping table* μ of $\mathcal{X}$ is a function $\mu(\mathcal{X}) : \mathbf{T} \mapsto (\mathbf{O} \cup \{\varepsilon\}) \times (\mathbf{F} \cup \{\varepsilon\})$, where ε is the *empty* symbol. The table $\mu(\mathcal{X})$ can be represented as a finite set of triples (t, o, f), where $t \in \mathbf{T}$, $o \in \mathbf{O} \cup \{\varepsilon\}$, and $f \in \mathbf{F} \cup \{\varepsilon\}$. The triple (t, o, f) defines the observability and normality of t: if $o \neq \varepsilon$, then t is *observable*, else t is *unobservable*; likewise, if $f \neq \varepsilon$, then t is *faulty*, else t is *normal*. Based on $\mu(\mathcal{X})$, each trajectory T in $\mathcal{X}^*$ can be associated with a *temporal observation* and a *temporal fault*. The *temporal observation* of T is the sequence of observations involved in T,

$$Obs(T) = [\, o \mid t \in T, (t, o, f) \in \mu(\mathcal{X}), o \neq \varepsilon \,]\,. \tag{2}$$

The *temporal fault* of T is the sequence of faults involved in T,

$$Flt(T) = [\, f \mid t \in T, (t, o, f) \in \mu(\mathcal{X}),\ f \neq \varepsilon \,]. \tag{3}$$

Although T is finite, its length is in general unbounded; hence, the length of both $Obs(T)$ and $Flt(T)$ is in general unbounded.

Example 1 The small DES used as a running example throughout the paper is inspired by the domain of power transmission networks, where each line is protected at its ends. This means that, in case a sensor detects a dangerous condition on the line (typically, the impedance below a given threshold), it commands a breaker to open, so as to electrically isolate the line until a liberating condition has been perceived. Shaded in the middle of the left side of Fig. 1 is a DES called $\mathcal{P}$ (protection), which includes two components, a sensor s and a breaker b, and one link connecting the (single) output pin of s with the (single) input pin of b. The model of s (outlined over $\mathcal{P}$) involves two states (denoted by circles) and four transitions (denoted by arcs). The model of b (outlined under $\mathcal{P}$) involves two states and eight transitions. Each component transition t from a state p to a state p', triggered by an input event e, and generating a set of output events E, is denoted by the angled triple $t = \langle p, (e, E), p' \rangle$, as detailed in the table displayed in the center of Fig. 1. The space of $\mathcal{P}$, namely $\mathcal{P}^*$, is depicted on the right side of Fig. 1, where each state is identified by a triple (s_s, s_b, e), with s_s being the state of the sensor, s_b the state of the breaker, and e the internal event in the link (ε means no event). To ease referencing, the states of $\mathcal{P}$ are renamed by numbers $0 \cdots 7$. The initial state is 0; the final states (denoted by double ellipses) are 0, 3, 4, and 7. The mapping table $\mu(\mathcal{P})$ is displayed on the left side of Fig. 2, with observations and faults being described on the right side of the figure. Owing to cycles, the set of possible trajectories of $\mathcal{P}$ is infinite, one of them being $T = [s_1, b_1, s_4, b_8]$, which ends in state 4 and is such that $Obs(T) = [\, act, opn, cls \,]$ and $Flt(T) = [\, \mathbf{f_2}, \mathbf{f_6} \,]$. T gives rise to the following evolution: s detects a threatening event and commands b to open; b opens; s detects a liberating event, yet commands b to open; instead, b eventually closes.

3 TEMPORAL DIAGNOSIS

The core problem in diagnosing a DES $\mathcal{X}$ is generating the set of candidates relevant to a temporal observation $\mathcal{O}$ of $\mathcal{X}$. In this paper a candidate is a temporal fault that is produced by a trajectory of $\mathcal{X}$ that entails $\mathcal{O}$. The (possibly infinite) set of candidates is the *temporal diagnosis* of $\mathcal{O}$, as formalized in Definition 1.

Definition 1 (temporal diagnosis) *Let $\mathcal{O}$ be a temporal observation of a DES $\mathcal{X}$. The* temporal diagnosis *of $\mathcal{O}$ is the set of temporal faults*

$$\Delta(\mathcal{O}) = \left\{ Flt(T) \mid T \in \mathcal{X}^*, \mathcal{O} = Obs(T) \right\}. \tag{4}$$

The notion of a space is adopted for formal reasons only, that is, $\mathcal{X}^*$ is not actually built, mainly because its generation is impractical in real applications. Therefore, eqn. (4) is not operational in nature, as it relies on $\mathcal{X}^*$. In practice, $\Delta(\mathcal{O})$ is yielded based on the portion of $\mathcal{X}^*$ that is consistent with $\mathcal{O}$, which is called the *$\mathcal{O}$-constrained space* of $\mathcal{X}$, as specified in Definition 2.

Definition 2 ($\mathcal{O}$-constrained space) *Let $\mathcal{O}$ be a temporal observation of $\mathcal{X}$. The* $\mathcal{O}$-constrained space *of $\mathcal{X}$, $\mathcal{X}^*_{\mathcal{O}}$, is a DFA whose language equals the set of trajectories $T \in \mathcal{X}^*$ where $\mathcal{O} = Obs(T)$.*

$\mathcal{X}^*_{\mathcal{O}}$ is generated based on the component models and the links in $\mathcal{X}$. A state of $\mathcal{X}^*_{\mathcal{O}}$ is a pair (x, i), where x is a state of $\mathcal{X}$ and i is an *index* in the range $[0\,..\,n]$, where n is the number of observations in $\mathcal{O}$.

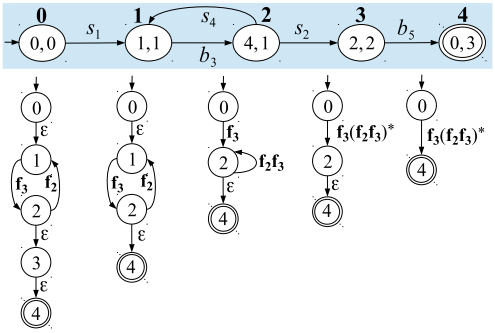


Figure 3: $\mathcal{O}$-constrained space of $\mathcal{P}$ where $\mathcal{O} = [act, sby, nop]$ (top), and generation of the temporal diagnosis $\Delta(\mathcal{O})$ (bottom).

Starting from the initial state $(x_0, 0)$, the transition function of $\mathcal{X}^*_{\mathcal{O}}$ is constructed by taking into account the component transitions that are triggerable in the state considered. When an observable transition t is triggerable in a state (x, i), with x' being the new state of $\mathcal{X}$ reached by t, a transition $\langle (x, i), t, (x', (i+1)) \rangle$ is created in $\mathcal{X}^*_{\mathcal{O}}$ iff $(t, \mathcal{O}[i+1], f) \in \mu(\mathcal{X})$, $i < n$. If, instead, t is unobservable, the index i is unchanged. When x is quiescent and $i = n$, that is, when all the observations in $\mathcal{O}$ are matched, the state (x, i) is final.

Example 2 Displayed on the top of Fig. 3 is the $\mathcal{O}$-constrained space of the DES $\mathcal{P}$ introduced in Example 1, where $\mathcal{O} = [act, sby, nop]$, namely $\mathcal{P}^*_{\mathcal{O}}$, where each state is identified by a pair (p, i), with p being a state of $\mathcal{P}$ and $i \in [0\,..\,3]$. The only final state is $4 = (0, 3)$.

Based on eqn. (4) and Definition 2, since each trajectory T in the $\mathcal{O}$-constrained space $\mathcal{X}^*_{\mathcal{O}}$ is such that $T \in \mathcal{X}^*$ and $\mathcal{O} = Obs(T)$, we have $Flt(T) \in \Delta(\mathcal{O})$. In fact, the temporal diagnosis $\Delta(\mathcal{O})$ is exactly the set of temporal faults relevant to the set of trajectories in $\mathcal{X}^*_{\mathcal{O}}$. However, this approach is impractical as it requires the consideration of a possibly infinite set of trajectories in order to generate $\Delta(\mathcal{O})$. So, what to do in order to compute $\Delta(\mathcal{O})$ based on Definition 1? Fortunately, a formal property of $\Delta(\mathcal{O})$ claimed in Proposition 1 is key to overcoming this computational obstacle.

Proposition 1 *Let $\mathcal{O}$ be a temporal observation of $\mathcal{X}$. The temporal diagnosis $\Delta(\mathcal{O})$ is a regular language on the set of faults of $\mathcal{X}$.*

Proof. Let $\mathcal{X}^*_{\mathcal{O}}$ be the $\mathcal{O}$-constrained space of $\mathcal{X}$. As such, the set of trajectories in $\mathcal{X}^*_{\mathcal{O}}$ equals the set of trajectories $T \in \mathcal{X}^*$ such that $\mathcal{O} = Obs(T)$. Let $\mathcal{N}$ be the nondeterministic finite automaton (NFA) obtained from $\mathcal{X}^*_{\mathcal{O}}$ by substituting f for each component transition t marking an arc of $\mathcal{X}^*_{\mathcal{O}}$, where $(t, o, f) \in \mu(\mathcal{X})$. The set of strings marking a path from the initial state to a final state in $\mathcal{N}$ is the temporal diagnosis $\Delta(\mathcal{O})$. Since it is accepted by an NFA, this language is a regular language. □

In practice, what makes Proposition 1 interesting is that a possibly infinite regular language can be always represented as a regular expression.[3] Hence, the set of temporal faults in $\Delta(\mathcal{O})$ can be represented by a regular expression on the alphabet of the faults of $\mathcal{X}$.

[3] A regular expression is defined inductively on the alphabet Σ as follows. The empty symbol ε is a regular expression. If $a \in \Sigma$, then a is a regular expression. If x and y are regular expressions, then the followings are regular expressions: $x \mid y$ (alternative), xy (concatenation), $x?$ (optionality), x^* (repetition zero or more times), and x^+ (repetition one or more times). When parentheses are missing, the concatenation has precedence over the alternative, while repetition has the highest precedence; for example, $ab^* \mid c$ equates to $(a(b)^*) \mid c$.

Algorithm 1 *Temporal Diagnosis*

1: **procedure** TEMPORAL DIAGNOSIS($\mathcal{O}$, $\mathcal{R}$)
2: **input** $\mathcal{O}$: a temporal observation of a DES $\mathcal{X}$
3: **output** $\mathcal{R}$: a regular expression denoting the temporal diagnosis $\Delta(\mathcal{O})$
4: **begin**
5: Generate $\mathcal{X}^*_{\mathcal{O}}$, the $\mathcal{O}$-constrained space of $\mathcal{X}$
6: **for all** transition $\langle \beta, t, \beta' \rangle$ in $\mathcal{X}^*_{\mathcal{O}}$ **do**
7: Substitute f for t, where $(t, o, f) \in \mu(\mathcal{X})$
8: **end for**
9: **if** the initial state β_0 of $\mathcal{X}^*_{\mathcal{O}}$ is entered by a transition **then**
10: Insert both a new initial state α_0 and a new ε-transition $\langle \alpha_0, \varepsilon, \beta_0 \rangle$
11: **end if**
12: **if** several final states exist or the final state is exited by a transition **then**
13: Insert the new final state α_q
14: **for all** original final state β_q of $\mathcal{X}^*_{\mathcal{O}}$ **do**
15: Insert a new ε-transition $\langle \beta_q, \varepsilon, \alpha_q \rangle$
16: **end for**
17: **end if**
18: Let $\mathcal{N}$ be the NFA so obtained, with initial state n_0 and final state n_q
19: Reduce $\mathcal{N}$ to a single transition $\langle n_0, \mathcal{R}, n_q \rangle$ by the algorithm in [7], where $\mathcal{R}$ is a regular expression denoting the language of $\mathcal{N}$ at line 18
20: **end procedure**

Example 3 Let $\mathcal{O} = [act, sby, nop]$ be the temporal observation of the DES $\mathcal{P}$ introduced in Example 1. The temporal diagnosis of $\mathcal{O}$ includes the temporal faults $Flt(T)$ where T is a trajectory in the $\mathcal{O}$-constrained space $\mathcal{P}^*_{\mathcal{O}}$ depicted in Fig. 3 (top). As the only faulty transitions involved are b_3 and s_4, with faults $\mathbf{f_3}$ (the breaker does not open) and $\mathbf{f_2}$ (the sensor commands the breaker to open rather than to close), respectively, one such temporal fault is $Flt(T) = [\mathbf{f_3}]$, where $T = [s_1, b_3, s_2, b_5]$. The temporal diagnosis $\Delta(\mathcal{O})$ is infinite, as the loop 1→2→1 in $\mathcal{P}^*_{\mathcal{O}}$ can be traversed an unbounded number of times, yielding a new temporal fault each time, namely $[\mathbf{f_3}, \mathbf{f_2}, \mathbf{f_3}]$, $[\mathbf{f_3}, \mathbf{f_2}, \mathbf{f_3}, \mathbf{f_2}, \mathbf{f_3}]$, $[\mathbf{f_3}, \mathbf{f_2}, \mathbf{f_3}, \mathbf{f_2}, \mathbf{f_3}, \mathbf{f_2}, \mathbf{f_3}]$, and so on. Still, despite the infinite set of temporal faults, $\Delta(\mathcal{O})$ turns out to be a regular language (cf. Proposition 1), which can therefore be defined as a regular expression, specifically $\Delta(\mathcal{O}) = \mathbf{f_3}(\mathbf{f_2}\mathbf{f_3})^*$. In plain words, $\Delta(\mathcal{O})$ includes the temporal faults starting with the fault $\mathbf{f_3}$ (the breaker does not open), which is followed by zero or more instances of the contiguous faults $\mathbf{f_2}\mathbf{f_3}$ (the sensor wrongly commands the breaker to open and the breaker remains closed instead of opening). If a candidate were a set of faults (instead of a temporal fault), the diagnosis output would consist of two candidates, namely $\{\mathbf{f_3}\}$ and $\{\mathbf{f_2}, \mathbf{f_3}\}$, which, differently from the temporal diagnosis, do not provide any information about the fact that, if fault $\mathbf{f_2}$ has occurred, then faults $\mathbf{f_2}$ and $\mathbf{f_3}$ are intermittent.

With reference to Example 3, the temporal diagnosis $\Delta(\mathcal{O})$ is determined by inspection of the NFA obtained from the $\mathcal{O}$-constrained space by substituting the symbols marking the transitions. What we need, however, is a general technique allowing for the automatic generation of $\Delta(\mathcal{O})$ starting from this NFA. To this end, we exploit and adapt the algorithm proposed in [7] in the context of sequential circuit state diagrams. Essentially, this algorithm takes as input an NFA and generates the regular expression of the language accepted by this NFA. This is exactly what we need to automatize the process of generating the regular language of a temporal diagnosis $\Delta(\mathcal{O})$.

The pseudocode of our (adapted) algorithm, called *Temporal Diagnosis*, is listed in Algorithm 1 (lines 1–20). It takes as input a temporal observation $\mathcal{O}$ of a DES $\mathcal{X}$ and generates as output a regular expression $\mathcal{R}$ (on the set of faults of $\mathcal{X}$) denoting the temporal diagnosis $\Delta(\mathcal{O})$. To this end, it first generates the $\mathcal{O}$-constrained space of $\mathcal{X}$ and replaces each component transition with the corresponding (possibly empty) fault (lines 5–8). Then, in lines 9–17, possibly a new initial state α_0 and a new (single) final state α_q is inserted along with an ε-transition connecting α_0 with the original initial state and one ε-transition from each original final state to α_q. This results in an NFA $\mathcal{N}$ with initial state n_0 and final state n_q (line 18). The ultimate goal of the algorithm is to transform $\mathcal{N}$ into a new NFA that is composed of just the initial state, the final state, and a single transition $\langle n_0, \mathcal{R}, n_q \rangle$, where $\mathcal{R}$ is in fact a regular expression identifying the temporal diagnosis $\Delta(\mathcal{O})$. The key idea in [7] is to simplify the NFA by progressively eliminating states and transitions while preserving the regular language accepted. This can be achieved by changing the alphabet of the NFA (being initially the set of faults) into a set of regular expressions on such faults.

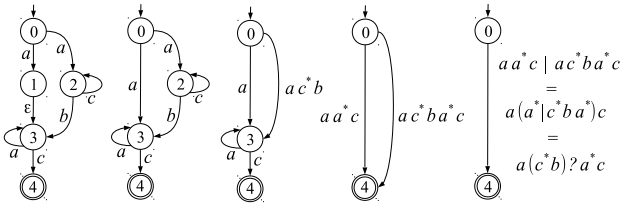


Figure 4: Generation of the temporal diagnosis based on Algorithm 1.

Example 4 With reference to Example 3, where $\mathcal{O} = [act, sby, nop]$ and the $\mathcal{O}$-constrained space is displayed on the top of Fig. 3, the NFA obtained in line 18 of Algorithm 1 is depicted on the bottom-left side of Fig. 3, where $n_0 = 0$ and $n_q = 4$. The reduction of the NFA in line 19 is performed in four steps, leading to the subsequent NFAs displayed in Fig. 3. In the first step, the cascade $[\langle 2, \varepsilon, 3 \rangle, \langle 3, \varepsilon, 4 \rangle]$ of transitions is replaced with the transition $\langle 2, \varepsilon, 4 \rangle$. In the second step, the state 1 and its entering/exiting transitions are replaced with the new transitions $\langle 0, \mathbf{f_3}, 2 \rangle$ and $\langle 2, (\mathbf{f_2}\mathbf{f_3}), 2 \rangle$. In the third step, the state 2 and its relevant transitions are replaced with the new transition $\langle 0, (\mathbf{f_3}(\mathbf{f_2}\mathbf{f_3})^*), 2 \rangle$. In the fourth step, the cascade $[\langle 0, (\mathbf{f_3}(\mathbf{f_2}\mathbf{f_3})^*), 2 \rangle, \langle 2, \varepsilon, 4 \rangle]$ of transitions is replaced with the transition $\langle 0, (\mathbf{f_3}(\mathbf{f_2}\mathbf{f_3})^*), 4 \rangle$. Since now $\mathcal{N}$ includes one transition only, no other action is carried out, thereby obtaining $\mathcal{R} = \mathbf{f_3}(\mathbf{f_2}\mathbf{f_3})^*$, which in fact equals the regular expression denoting $\Delta(\mathcal{O})$ that was determined by inspection of $\mathcal{P}^*_{\mathcal{O}}$ in Example 3.

Example 5 Another (abstract) example of the application of Algorithm 1 is outlined in Fig. 4, where the graph displayed on the left is assumed to be the NFA $\mathcal{N}$ obtained in line 18, with a, b, and c being the faults involved. The states 1, 2, and 3, along with relevant transitions, are removed one by one and substituted with other transitions. Eventually, the parallel transitions $\langle 0, (aa^*c), 4 \rangle$ and $\langle 0, (ac^*ba^*c), 4 \rangle$ are replaced with the transition $\langle 0, ((aa^*c) \mid (ac^*ba^*c)), 4 \rangle$. The regular expression denoting $\Delta(\mathcal{O})$ is thus $\mathcal{R} = aa^*c \mid ac^*ba^*c = a(a^*|c^*ba^*)c = a(c^*b)?a^*c$.

Proposition 2 *Algorithm 1 is sound and complete.*

Proof (*sketch*). Based on Definition 2, the language of the $\mathcal{O}$-constrained space generated in line 5 equals the set of trajectories $T \in \mathcal{X}^*$ such that $Obs(T) = \mathcal{O}$. After the substitutions performed in lines 6–8, the language of the NFA equals the set of temporal faults $Flt(T)$ where $Obs(T) = \mathcal{O}$, in other words, it equals the temporal diagnosis $\Delta(\mathcal{O})$ (Definition 1). After the possible insertions of the new initial state and the new final state, the language of the NFA $\mathcal{N}$ obtained in line 18 is still $\Delta(\mathcal{O})$. Since the reduction of $\mathcal{N}$ to a single transition $\langle n_0, \mathcal{R}, n_q \rangle$ in line 19 does not alter the language of $\mathcal{N}$, the regular expression $\mathcal{R}$ marking this transition equals $\Delta(\mathcal{O})$. □

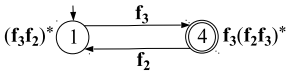


Figure 5: Fault space of state 1 of $\mathcal{P}^*$ (shown on right side of Fig. 1).

The technique presented above for generating the temporal diagnosis $\Delta(\mathcal{O})$ does not exploit any compiled knowledge of the DES. This is why Algorithm 1 requires the generation of the $\mathcal{O}$-constrained space upfront (line 5), a solution that may be less than optimal when time constraints on the diagnosis output are stringent. To alleviate this drawback, the notion of a *temporal diagnoser* is introduced in Section 4 and exploited for fast diagnosis in Section 5.

4 TEMPORAL DIAGNOSER

Roughly, the temporal diagnoser $\mathcal{X}^\Delta$ of a DES $\mathcal{X}$ is an NFA resulting from the compilation of $\mathcal{X}$. The alphabet of $\mathcal{X}^\Delta$ is a set of pairs (o, r), where o is an observation of $\mathcal{X}$ and r a regular expression on the faults of $\mathcal{X}$ defined in the mapping table $\mu(\mathcal{X})$. Intuitively, each state of $\mathcal{X}^\Delta$ (called a fault space) embodies a sort of local diagnosis information defined by regular expressions on the faults. When a temporal observation $\mathcal{O}$ occurs, $\mathcal{X}^\Delta$ allows for the efficient generation of the temporal diagnosis $\Delta(\mathcal{O})$.

Definition 3 (fault space) *Let $\mathcal{X}^* = (\Sigma, X, \tau, x_0, X_\mathrm{q})$ be the space of a DES $\mathcal{X}$,* $\mathbf{F}$ *the set of faults of $\mathcal{X}$, and x a state in X. The* fault space *of x is an NFA (extended with X'_e)*

$$x^\delta = \left(\Sigma', X', \tau', x'_0, X'_\mathrm{e}, X'_\mathrm{q}\right) \tag{5}$$

where $\Sigma' = \mathbf{F} \cup \{\varepsilon\}$ is the alphabet, $X' \subseteq X$ is the set of states, $x'_0 = x$ is the initial state, X'_e is the set of exit states*, where $x_\mathrm{e} \in X'_\mathrm{e}$ iff $\langle x_\mathrm{e}, t, x'\rangle \in \tau$ and t is observable, $X'_\mathrm{q} = X' \cap X_\mathrm{q}$ is the set of final states, and $\tau' : X' \times \Sigma' \mapsto 2^{X'}$ is the transition function, where $\langle x_1, f, x_2\rangle$ is an arc in τ' iff $\langle x_1, t, x_2\rangle \in \tau$ and $(t, o, f) \in \mu(\mathcal{X})$. Moreover, each state $x' \in X'_\mathrm{e} \cup X'_\mathrm{q}$, called a* labeled state*, is marked with the regular language of the strings of faults of the subtrajectories from x to x', denoted $\Delta(x')$. The* diagnosis language *of x^δ is a regular language defined as follows:*

$$\Delta\left(x^\delta\right) = \begin{cases} \varepsilon & \text{if } X'_\mathrm{q} = \emptyset \\ \Delta(x) & \text{if } X'_\mathrm{q} = \{x\} \\ \Delta(x_1) \mid \ldots \mid \Delta(x_n) & \text{if } X'_\mathrm{q} = \{x_1, \ldots, x_n\}. \end{cases} \tag{6}$$

Example 6 With reference to the DES $\mathcal{P}$ introduced in Example 1, shown in Fig. 5 is the fault space of state 1, namely 1^δ, where $X'_\mathrm{e} = \{1, 4\}$ and $X'_\mathrm{q} = \{4\}$ (cf. the space $\mathcal{P}^*$ displayed on the right side of Fig. 1). Both states 1 and 4 are marked with a regular expression denoting the language of the segments of temporal faults relevant to the subtrajectories of $\mathcal{P}$ starting in 1 and ending in each of these two states. Since $X'_\mathrm{q} = \{4\}$, we have $\Delta(1^\delta) = \mathbf{f_3(f_2f_3)}^*$.

In order to mark the internal states of a fault space with the regular expressions required, we cannot apply Algorithm 1 as is because several states are involved in the marking process and, in general, each of them is associated with a distinct regular expression. Still, after the substitution of the component transitions with the corresponding faults, the actions performed on the NFA $\mathcal{N}$ by Algorithm 1 remain substantially the same, with the exception of a few variations. First, a new final state n_q is always inserted, along with an ε-transition from each labeled state to n_q. Then, the replacement of a sequence of transitions, namely $[\langle n, r_1, n_1\rangle, \langle n_1, r_2, n_2\rangle, \ldots, \langle n_{k-1}, r_k, n'\rangle]$, is extended when the last transition $\langle n_{k-1}, r_k, n'\rangle$ is such that n_{k-1} is a labeled state and $n' = n_\mathrm{q}$ (hence, $r_k = \varepsilon$). If so, the transition replacing the sequence of transitions will be $\langle n, (r_1 r_2 \cdots r_{k-1})_{(n_{k-1})}, n'\rangle$, where the additional subscript (n_{k-1}) indicates that the regular expression $(r_1 r_2 \cdots r_{k-1})$ is associated with the state n_{k-1}. Also, the replacement of a set of parallel transitions with a single transition needs caution: only those transitions with the same subscript (or without any subscript) can be replaced with a single transition marked with the alternative of the corresponding regular expressions. Hence, in general, the set of parallel transitions is replaced with another set of parallel transitions, where each new transition is marked with a regular expression having a different (if any) subscript. Finally, the subscript may also come into play when a state is removed.

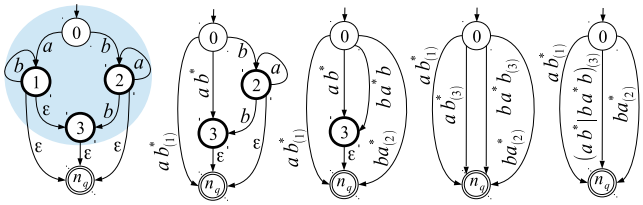


Figure 6: Generation of $\Delta(x')$, where $x' \in \{1, 2, 3\}$.

Example 7 Outlined in Fig. 6 is an abstract example showing the computation of the regular expressions marking the states 1, 2, and 3 (depicted in bold) in the NFA displayed on the left side of the figure. This NFA is assumed to be obtained from the portion of the space (shaded in the figure) rooted in 0 and encompassing all the states that are reachable by the arcs marked with unobservable component transitions only. Then, the component transition marking each arc has been substituted with the corresponding (possibly empty) fault, namely ε, a, or b. Moreover, a final state n_q has been inserted along with three ε-transitions, namely $\langle 1, \varepsilon, n_\mathrm{q}\rangle$, $\langle 2, \varepsilon, n_\mathrm{q}\rangle$, and $\langle 3, \varepsilon, n_\mathrm{q}\rangle$. This resembles the initial configuration of the NFA $\mathcal{N}$ in Algorithm 1 (line 18). Considering the NFA in the second position in Fig. 6, the original transitions $\langle 0, a, 1\rangle$, $\langle 1, b, 1\rangle$, and $\langle 1, \varepsilon, n_\mathrm{q}\rangle$ lead to the insertion of the new transition $\langle 0, ab^*_{(1)}, n_\mathrm{q}\rangle$ since the third transition connects 1 with the final state n_q. This will allow the algorithm to eventually recognize ab^* as part of the language $\Delta(1)$. A similar scenario holds for the NFA in the third position, where the removal of the transitions involving the state 2 yields the regular expression $ba^*_{(2)}$. Likewise, when the state 3 is removed along with its relevant transitions (NFA in the fourth position), the two new transitions are marked with the regular expressions $ab^*_{(3)}$ and $ba^*b_{(3)}$, respectively. Eventually, the parallel transitions relevant to the same state are merged into a single transition; in our example, the two transitions relevant to 3 are merged into a single transition marked with the alternative of the regular expressions of these transitions, namely $(ab^* \mid ba^*b)_{(3)}$. The languages of (partial) temporal faults marking the labeled states are therefore $\Delta(1) = ab^*$, $\Delta(2) = ba^*$, and $\Delta(3) = ab^* \mid ba^*b$.

Definition 4 (temporal diagnoser) *Let $\mathcal{X}^* = (\Sigma, X, \tau, x_0, X_\mathrm{q})$ be the space of $\mathcal{X}$, and let* $\mathbf{O}$ *be the set of observations of $\mathcal{X}$,* $\mathbf{F}$ *the set of faults of $\mathcal{X}$, and* $\mathbf{R}$ *the set of regular expressions on* $\mathbf{F}$*, respectively. The* temporal diagnoser *of $\mathcal{X}$ is an NFA*

$$\mathcal{X}^\Delta = \left(\Sigma', X', \tau', x'_0, X'_\mathrm{q}\right) \tag{7}$$

where $\Sigma' \subseteq \mathbf{O} \times \mathbf{R}$ is the alphabet, X' is the the set of states, where each state is a fault space, x'_0 is the fault space of x_0, $X'_\mathrm{q} \subseteq X'$ is the set of final states, where $x'_\mathrm{q} \in X'_\mathrm{q}$ iff the set of final states (in X_q)

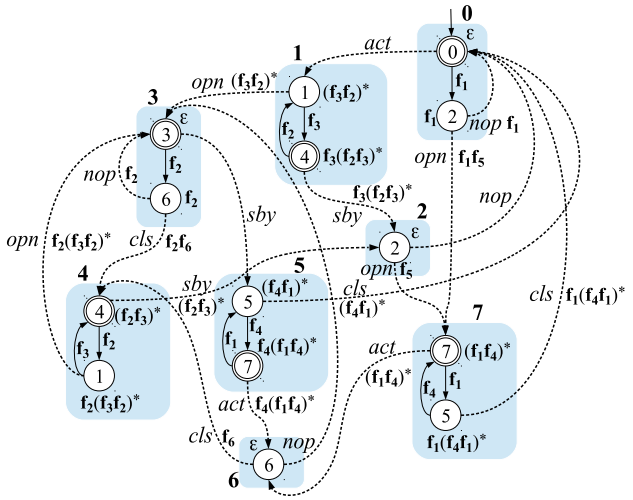


Figure 7: Temporal diagnoser $\mathcal{P}^{\Delta}$.

within the fault space x'_{q} is not empty, and $\tau' : (X' \times X) \times \Sigma' \mapsto 2^{(X' \times X)}$ is the transition function, where $\langle (x'_1, x_1), (o, r), (x'_2, x_2) \rangle$ is an arc in τ' iff x_1 is an exit state of x'_1, $\langle x_1, t, x_2 \rangle \in \tau$, $(t, o, f) \in \mu(\mathcal{X})$, $r = \Delta(x_1) f$, x'_2 is the fault space of x_2.

Example 8 With reference to the DES $\mathcal{P}$ introduced in Example 1, shown in Fig. 7 is the temporal diagnoser $\mathcal{P}^{\Delta}$, where the fault spaces are renamed $0 \cdots 7$. Unlike transitions between states of each fault space, which are represented with plain arcs, the arcs representing transitions between states of the temporal diagnoser $\mathcal{P}^{\Delta}$ are denoted with dashed arcs. For each pair (o, r) marking a transition, the parentheses are omitted, as well as the regular expression r when $r = \varepsilon$.

5 FAST DIAGNOSIS

A temporal diagnoser $\mathcal{X}^{\Delta}$ is compiled knowledge built offline that allows for the efficient online generation of a temporal diagnosis $\Delta(\mathcal{O})$ of $\mathcal{X}$ by means of an algorithm called *Fast Diagnosis*. Roughly, $\mathcal{X}^{\Delta}$ is traversed based on $\mathcal{O}$ and the regular expressions marking the transitions of $\mathcal{X}^{\Delta}$ are concatenated in the given order. When the transition relevant to the last observation in $\mathcal{O}$ is traversed and a final state x_{f} is entered, the regular expression composed so far is eventually appended with the diagnosis language $\Delta(x_{\mathrm{f}})$, which was itself precomputed offline. Since $\mathcal{X}^{\Delta}$ is an NFA, several paths can generate the same temporal observation $\mathcal{O}$; therefore, the final regular expression is in general composed by the alternative of several subexpressions. Algorithm *Fast Diagnosis* is faster than algorithm *Temporal Diagnosis* in the online computation perspective, which is the user perspective.

The pseudocode of *Fast Diagnosis* is listed in Algorithm 2 (lines 1–26). It takes as input a temporal diagnoser $\mathcal{X}^{\Delta}$ and a temporal observation $\mathcal{O}$, and generates as output a regular expression $\mathcal{R}$ whose language equals $\Delta(\mathcal{O})$, as proven in Proposition 3. To this end, the algorithm exploits a set of *contexts*, namely χ, with each context being a pair (x, r), where x is a state of $\mathcal{X}^{\Delta}$ and r a regular expression on the faults of $\mathcal{X}$. Initially, χ includes just the initial context (x_0, ε), where x_0 is the initial state of $\mathcal{X}^{\Delta}$ (line 6). Then, a loop is performed on the observations in $\mathcal{O}$ (lines 7–20). At each iteration, a new set of contexts, namely χ_{new} is generated based on the current content of χ. Specifically, for each context (x', r') in χ and for each arc of $\mathcal{X}^{\Delta}$ exiting x' and marked with the pair (o, r), where o is the current observation, a regular expression $r_2 = r'r$ is computed (line 11). In fact, r' accounts for the faults up to x', while r accounts for the faults up to the internal state x of x' plus the (possibly empty) fault associated with the component transition that is observable by means of o. The update of χ_{new} is performed in lines 12–16, depending on whether a context involving the reached state x'_2 exists in χ_{new} or not. If a context (x'_2, r'_2) exists, then its regular expression is extended with the alternative r_2, thereby yielding the updated context $(x'_2, (r'_2|r_2))$ (line 13). Otherwise, a new context (x'_2, r_2) is created (line 15). Before the end of the iteration, χ is replaced with χ_{new} (line 19). When all the observations have been considered (termination of the outer loop), the regular expression $\mathcal{R}$ is determined (lines 21–25). Two scenarios are possible for χ: it contains either one context (x, r) or $k > 1$ contexts. In the first scenario (lines 21–22), $\mathcal{R}$ is generated by appending r with the diagnosis language of x, thereby obtaining $\mathcal{R} = r\Delta(x)$. This is because r accounts for the faults up to the initial state of x, while $\Delta(x)$ accounts for the faults within x (up to any final state within x). In the second scenario (lines 23–24), since several contexts exist, the same operation is performed for each context (x_i, r_i), $i \in [1 \,..\, k]$, thereby yielding the regular expression $\mathcal{R}$ that is composed of the alternatives $r_i(\Delta(x_i))$.

Algorithm 2 *Fast Diagnosis*

```
1:  procedure FAST DIAGNOSIS(X^Δ, O, R)
2:    input X^Δ = (Σ, X, τ, x0, Xq): the temporal diagnoser of a DES X
3:          O: a temporal observation of X
4:    output R: a regular expression denoting the temporal diagnosis Δ(O)
5:  begin
6:    χ ← {(x0, ε)}
7:    for all observation o ∈ O do
8:      χnew ← ∅
9:      for all (x', r') ∈ χ do
10:       for all arc ⟨(x', x), (o, r), (x'2, x2)⟩ in τ do
11:         r2 ← r'r
12:         if (x'2, r'2) ∈ χnew then
13:           Substitute (x'2, (r'2|r2)) for (x'2, r'2) in χnew
14:         else
15:           Insert (x'2, r2) into χnew
16:         end if
17:       end for
18:     end for
19:     χ ← χnew
20:   end for
21:   if χ = {(x, r)} then
22:     R ← rΔ(x)
23:   else if χ = {(x1, r1), ..., (xk, rk)} where k > 1 then
24:     R ← (r1(Δ(x1))) | ... | (rk(Δ(xk)))
25:   end if
26: end procedure
```

Example 9 With reference to the temporal diagnoser $\mathcal{P}^{\Delta}$ displayed in Fig. 7, let $\mathcal{O} = [act, sby, nop]$ be the temporal observation of $\mathcal{P}$ considered in Example 3. Based on line 6 of Algorithm 2, we have $\chi = \{(0, \varepsilon)\}$. On the first observation, namely *act*, the only arc involved in the loop (line 10) is $\langle (0, 0), (act, \varepsilon), (1, 1) \rangle$; hence, $r_2 = \varepsilon$ (line 11) and $\chi_{\mathrm{new}} = \{(1, \varepsilon)\}$ (line 15). On the observation *sby* (second iteration of the outer loop), we have $\chi = \{(1, \varepsilon)\}$. Now, the only arc involved in line 10 is $\langle (1, 4), (sby, \mathbf{f_3(f_2f_3)^*}), (2, 2) \rangle$; hence, $r_2 = \mathbf{f_3(f_2f_3)^*}$ (line 11) and $\chi_{\mathrm{new}} = \{(2, \mathbf{f_3(f_2f_3)^*})\}$ (line 15). On the last observation, namely *nop*, we have $\chi = \{(2, \mathbf{f_3(f_2f_3)^*})\}$. The only arc involved is $\langle (2, 2), (nop, \varepsilon), (0, 0) \rangle$; hence, $r_2 = \mathbf{f_3(f_2f_3)^*}$ (line 11) and $\chi = \chi_{\mathrm{new}} = \{(0, \mathbf{f_3(f_2f_3)^*})\}$ (line 19). Eventually, since χ is a singleton, the regular expression $\mathcal{R}$ is computed in line 22, namely $\mathcal{R} = \mathbf{f_3(f_2f_3)^*}$, as $\Delta(0) = \varepsilon$. As expected, the lan-

guage of $\mathcal{R}$ equals the temporal diagnosis $\Delta(\mathcal{O})$ that was yielded both in Example 3, by inspection of the $\mathcal{O}$-constrained space of $\mathcal{P}$, and in Example 4, based on Algorithm 1.

Example 10 Considering again the temporal diagnoser $\mathcal{P}^\Delta$ displayed in Fig. 7, let $\mathcal{O} = [act, opn, cls]$ be a new temporal observation of $\mathcal{P}$. Initially, based on Algorithm 2, we have $\chi = \{(0, \varepsilon)\}$. On the first observation, namely *act*, the only arc involved in the loop is $\langle (0,0), (act, \varepsilon), (1,1)\rangle$; hence, $r_2 = \varepsilon$ and $\chi_{\text{new}} = \{(1, \varepsilon)\}$. On the second observation, namely *opn*, we have $\chi = \{(1, \varepsilon)\}$. Now, the only arc involved in line 10 is $\langle (1,1), (opn, (\mathbf{f_3 f_2})^*), (3,3)\rangle$; hence, $r_2 = (\mathbf{f_3 f_2})^*$ and $\chi_{\text{new}} = \{(3, (\mathbf{f_3 f_2})^*)\}$. On the last observation, namely *cls*, we have $\chi = \{(3, (\mathbf{f_3 f_2})^*)\}$. The only arc involved is $\langle (3,6), (cls, \mathbf{f_2 f_6}), (4,4)\rangle$; hence, $r_2 = (\mathbf{f_3 f_2})^* \mathbf{f_2 f_6}$ (line 11) and $\chi = \{(4, (\mathbf{f_3 f_2})^* \mathbf{f_2 f_6})\}$ (line 19). Eventually, since χ is a singleton, the regular expression $\mathcal{R}$ is computed in line 22, namely $\mathcal{R} = r\Delta(4) = (\mathbf{f_3 f_2})^* \mathbf{f_2 f_6} (\mathbf{f_2 f_3})^*$, where $\Delta(4) = (\mathbf{f_2 f_3})^*$. In the set-oriented approach, where a candidate is a set of faults rather than a temporal fault, the diagnosis output consists of two candidates only, namely $\{\mathbf{f_2}, \mathbf{f_6}\}$ and $\{\mathbf{f_2}, \mathbf{f_3}, \mathbf{f_6}\}$, without any temporal relationships between faults. Once these candidates have been output, the diagnostician knows that both fault $\mathbf{f_2}$ and $\mathbf{f_6}$ have certainly occurred, while the occurrence of fault $\mathbf{f_3}$ is uncertain; he or she has no hint about how many times such faults have manifested themselves and in which temporal order. In temporal-oriented diagnosis, the diagnostician knows not only that both fault $\mathbf{f_2}$ and $\mathbf{f_6}$ have certainly occurred, but also that $\mathbf{f_6}$ has occurred just once and that it has been preceded by the occurrence of $\mathbf{f_2}$ (with no other fault in between), and that, in case $\mathbf{f_3}$ had occurred, it was the first and/or the last fault in the sequence, etc. This comparison between set-oriented diagnosis and temporal diagnosis shows once more that the latter can explain better what has happened inside a DES since it provides a temporal view that is missing in the corresponding set-oriented diagnosis. When the diagnostic result is a set of faults, the diagnostician neither knows whether a fault has occurred intermittently nor whether a fault has occurred before or after another, a piece of information that is important in a causal analysis.

Proposition 3 *Algorithm 2 is sound and complete.*

Proof (*sketch*). We have to show that the language of the regular expression $\mathcal{R}$ computed by Algorithm 2 equals the temporal diagnosis $\Delta(\mathcal{O})$ defined in eqn. (4), where $\mathcal{O} = [o_1, \ldots, o_n]$. First, notice that the graph obtained from the temporal diagnoser $\mathcal{X}^\Delta$, once the fault spaces are unfolded, resembles the space $\mathcal{X}^*$ where the symbols t (component transitions) marking the transitions of $\mathcal{X}^*$ are replaced with either a (possibly empty) fault (within a state of $\mathcal{X}^\Delta$) or with a pair (o, r) (between states of $\mathcal{X}^\Delta$). In each state (fault space) of $\mathcal{X}^\Delta$, each component transition, marking the (internal) arcs before the substitution, is unobservable. Moreover, each labeled state x within a fault space x' is marked with a regular expression r denoting the set strings of faults relevant to the segments of trajectories of $\mathcal{X}$ from the initial state of x' to x. Let $\mathcal{F}$ be a temporal fault in $\Delta(\mathcal{O})$. Based on eqn. (4), there is a trajectory T in $\mathcal{X}^*$ such that $\mathcal{F} = Flt(T)$ and $\mathcal{O} = Obs(T)$. The trajectory T can be traced by a path $\wp$ in $\mathcal{X}^\Delta$ starting from the initial state of the fault space representing the initial state of $\mathcal{X}^\Delta$ and ending in a final state x_f of a (final) state x'_f of $\mathcal{X}^\Delta$. When the current component transition in T is observable via an observation o in $\mathcal{O}$, a transition marked with (o, r) is performed on $\mathcal{X}^\Delta$. For its part, the regular expression $\mathcal{R}$ generated by Algorithm 2 certainly accounts for the path $\wp$, as the generation of χ is driven by the observations in $\mathcal{O}$. In other words, among the alternatives in $\mathcal{R}$, there is a regular expression $r_\wp$ that is constructed based on $\wp$ by concatenating the regular expressions r_i associated with the observations o_i, $i \in [1 \,..\, n]$, within the pairs (o_i, r_i) marking the transitions of $\mathcal{X}^\Delta$ and appending this concatenation with the diagnosis language of the final state x'_f of $\mathcal{X}^\Delta$, namely $\Delta(x'_\text{f})$. Since each r_i accounts for the (segments of) temporal faults from one state to the next, the language of $r_\wp$ necessarily includes the temporal fault $\mathcal{F}$ (soundness). To prove the completeness, we have to show that, if $\mathcal{F}$ is a string in the language of $\mathcal{R}$, then $\mathcal{F}$ is a temporal fault in $\Delta(\mathcal{O})$. As such, $\mathcal{F}$ is generated by means of a path $\wp$ traversing the unfolded $\mathcal{X}^\Delta$ starting from the initial state of the initial state of $\mathcal{X}^\Delta$ and ending in a final state of a final state of $\mathcal{X}^\Delta$, with the constraint that the subsequence of (external) transitions of $\mathcal{X}^\Delta$ in $\wp$ generates the sequence of observations in $\mathcal{O}$. Since $\wp$ corresponds to a trajectory T in $\mathcal{X}^*$ where $Obs(T) = \mathcal{O}$ and $Flt(T) = \mathcal{F}$, based on eqn. (4), $\mathcal{F} \in \Delta(\mathcal{O})$. □

6 CONCLUSION

In this paper, the notions of temporal fault and temporal diagnosis have been introduced, which allow for a novel characterization of the diagnosis results for DESs. Instead of being a set of faults, a candidate is a temporal fault, namely a (possibly unbounded) multiset of temporally ordered faults, just as a temporal observation is a multiset of temporally ordered observations. Consequently, a temporal diagnosis is a (possibly infinite) set of temporal faults that are produced by the trajectories that generate the temporal observation. Despite possibly including an infinite number of temporal faults, a temporal diagnosis can be represented by a (finite) regular expression. This endows the diagnosis results with a temporal aspect that can help explain better what has happened inside the DES. To the best of our knowledge, this characterization has never been introduced before.

Albeit being embedded in the active-system approach [18], it is interesting to discuss the proposal in the context of the diagnoser approach also [27]. The diagnoser approach assumes that both the language of the transitions of the DES and the language of the observable events of the DES are live, whereas the active-system approach does not make any such assumption. Moreover, the diagnoser approach assumes that the faulty transitions are unobservable, while this assumption is relaxed in the active-system approach. Consequently, while according to the diagnoser approach there does not exist any unobservable behavioral cycle, and hence, there does not exist any cycle of faults, both such cycles are allowed in the active-system approach. Therefore, the regular expressions relevant to the occurrence of faults in active systems can represent an unbounded number of iterations, while this is not needed in case the temporal diagnosis characterization were adopted by the diagnoser approach (or by approaches making the same assumptions). Another feature of the active-system approach is that the task of a posteriori diagnosis considers only trajectories that end in a global final state, where all links are empty. The notion of a final state does not apply to the DES models taken into account by the diagnoser approach, as such models, supporting synchronous communication between components, do not include any link. In the temporal diagnoser, final states are identified within the fault spaces, so as to differentiate them from the other global states. No such differentiation is needed in case the temporal diagnoser were adopted in the diagnoser approach.

Fault detection in DESs was generalized in [12] to the recognition of a pattern, this being a DFA that can represent the occurrence of multiple faults, the ordered occurrence of significant events, the multiple occurrences of the same fault, etc. It is tempting to speculate that temporal diagnosis resembles diagnosis with supervision pat-

terns; after all, a pattern enables the detection of a specific language of transitions and, therefore, the detection of a specific language of faulty transitions also. In other words, given the temporal observation taken as input by the diagnosis task, the supervision pattern approach can find out whether there exists a trajectory implying such a sequence that complies with the given (pattern) language. Notice that there may exist several other trajectories that imply the temporal observation while producing sequences of faults that do not belong to the given (pattern) language: the supervision pattern approach does not produce any output about them. The difference with respect to temporal diagnosis is that the latter is not given any automaton upfront recognizing a language, instead it produces a regular expression representing the language of the faults of all the trajectories that imply the given temporal observation. Moreover, the output of the supervision pattern approach clarifies whether the pattern has occurred; however, it does not compute the number of its occurrences, nor does it show the relative order of these occurrences and those of individual faults within the trajectories implying the temporal observation. On the other hand, from the point of view of the approach presented in the current paper, if a fault is associated with a pattern, this can be part of a temporal fault as all other faults are. In other words, temporal diagnosis is orthogonal to the classification of faults, being they "simple" or somehow "complex" as in [12, 15, 20]. Recent works of the authors [3, 4, 5] have shown how to compile the knowledge relevant to a DES without generating the global behavior. Similar techniques can be exploited in order to build the temporal diagnoser based on scenarios, an interesting topic for future research. Future efforts can be devoted also to adopting the notion of temporal diagnosis in the context of complex DESs [19, 17]. Finally, an implementation of the algorithms presented in this paper is needed in order to carry out some experimental activities.

ACKNOWLEDGEMENTS

We would like to thank the referees for their constructive comments. This work was supported in part by Regione Lombardia (Smart4CPPS, Linea Accordi per Ricerca, Sviluppo e Innovazione, POR-FESR 2014-2020 Asse I) and by the National Natural Science Foundation of China (grant number 61972360).

REFERENCES

[1] P. Baroni, G. Lamperti, P. Pogliano, and M. Zanella, 'Diagnosis of large active systems', *Artificial Intelligence*, **110**(1), 135–183, (1999).
[2] F. Basile, 'Overview of fault diagnosis methods based on Petri net models', in *Proceedings of the 2014 European Control Conference, ECC 2014*, pp. 2636–2642, (2014).
[3] N. Bertoglio, G. Lamperti, and M. Zanella, 'A posteriori diagnosis of discrete-event systems with symptom dictionary and scenarios', in *Advances and Trends in Artificial Intelligence. From Theory to Practice. IEA/AIE 2019*, eds., F. Wotawa, G. Friedrich, I. Pill, R. Koitz-Hristov, and M. Ali, volume 11606 of *Lecture Notes in Computer Science*, 325–333, Springer International Publishing, Cham, (2019).
[4] N. Bertoglio, G. Lamperti, and M. Zanella, 'Temporal diagnosis of discrete-event systems with dual knowledge compilation', in *Machine Learning and Knowledge Extraction*, eds., A. Holzinger, P. Kieseberg, E. Weippl, and A Min Tjoa, volume 11713 of *Lecture Notes in Computer Science*, 333–352, Springer, Berlin, (2019).
[5] N. Bertoglio, G. Lamperti, M. Zanella, and X. Zhao, 'Twin-engined diagnosis of discrete-event systems', *Engineering Reports*, **1**, 1–20, (2019).
[6] D. Brand and P. Zafiropulo, 'On communicating finite-state machines', *Journal of the ACM*, **30**(2), 323–342, (1983).
[7] J.A. Brzozowski and E.J. McCluskey, 'Signal flow graph techniques for sequential circuit state diagrams', *IEEE Transactions on Electronic Computers*, **EC-12**(2), 67–76, (1963).
[8] C.G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, Springer, New York, second edn., 2008.
[9] 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).
[10] J. de Kleer and B.C. Williams, 'Diagnosing multiple faults', *Artificial Intelligence*, **32**(1), 97–130, (1987).
[11] *Readings in Model-Based Diagnosis*, eds., W. Hamscher, L. Console, and J. de Kleer, Morgan Kaufmann, San Mateo, CA, 1992.
[12] 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)*, pp. 262–268, Ann Arbor, MI, (2006). IEEE Computer Society.
[13] G. Lamperti and M. Zanella, 'Diagnosis of discrete-event systems from uncertain temporal observations', *Artificial Intelligence*, **137**(1–2), 91–163, (2002).
[14] G. Lamperti and M. Zanella, 'A bridged diagnostic method for the monitoring of polymorphic discrete-event systems', *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, **34**(5), 2222–2244, (2004).
[15] G. Lamperti and M. Zanella, 'Context-sensitive diagnosis of discrete-event systems', in *Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI 2011)*, ed., T. Walsh, volume 2, pp. 969–975, Barcelona, Spain, (2011). AAAI Press.
[16] 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).
[17] G. Lamperti, M. Zanella, and X. Zhao, 'Abductive diagnosis of complex active systems with compiled knowledge', in *Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference (KR2018)*, eds., M. Thielscher, F. Toni, and F. Wolter, pp. 464–473, Tempe, Arizona, (2018). AAAI Press.
[18] G. Lamperti, M. Zanella, and X. Zhao, *Introduction to Diagnosis of Active Systems*, Springer, Cham, 2018.
[19] G. Lamperti, M. Zanella, and X. Zhao, 'Knowledge compilation techniques for model-based diagnosis of complex active systems', in *Machine Learning and Knowledge Extraction*, eds., A. Holzinger, P. Kieseberg, A Min Tjoa, and E. Weippl, volume 11015 of *Lecture Notes in Computer Science*, 43–64, Springer, Cham, (2018).
[20] 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).
[21] B. Li, M. Khlif-Bouassida, and A. Toguyéni, 'Reduction rules for diagnosability analysis of complex systems modeled by labeled Petri nets', *IEEE Transactions on Automation Science and Engineering*, (2019).
[22] Y. Pencolé, G. Steinbauer, C. Mühlbacher, and L. Travé-Massuyès, 'Diagnosing discrete event systems using nominal models only', in *28th International Workshop on Principles of Diagnosis (DX 2017)*, pp. 169–183, Brescia, Italy, (2017).
[23] Ingo Pill and Thomas Quaritsch, 'Behavioral diagnosis of LTL specifications at operator level', in *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pp. 1053–1059. AAAI Press, (2013).
[24] N. Ran, H. Su, A. Giua, and C. Seatzu, 'Codiagnosability analysis of bounded Petri nets', *IEEE Transactions on Automatic Control*, **63**(4), 1192–1199, (2018).
[25] R. Reiter, 'A theory of diagnosis from first principles', *Artificial Intelligence*, **32**(1), 57–95, (1987).
[26] M. Sampath, S. Lafortune, and D.C. Teneketzis, 'Active diagnosis of discrete-event systems', *IEEE Transactions on Automatic Control*, **43**(7), 908–929, (1998).
[27] 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).
[28] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D.C. Teneketzis, 'Failure diagnosis using discrete-event models', *IEEE Transactions on Control Systems Technology*, **4**(2), 105–124, (1996).
[29] Viktor Schuppan, 'Towards a notion of unsatisfiable and unrealizable cores for ltl', *Sci. Comput. Program.*, **77**(7–8), 908–939, (jul 2012).
[30] X. Yin and S. Lafortune, 'On the decidability and complexity of diagnosability for labeled Petri nets', *IEEE Transactions on Automatic Control*, **62**(11), 5931–5938, (2017).