"What Happened?" vs "What is Ab?"– A Theoretical Framework for Diagnosing Evolutive Systems

Marie-Odile Cordier and **Sylvie Thiébaux**
IRISA
Campus de Beaulieu
35042 Rennes cedex
France
e-mail: {cordier, thiebaux}@irisa.fr
tel: (33) 99 84 73 14, fax: (33) 99 84 71 71

Abstract

This paper presents a new view of model-based diagnosis for evolutive systems. The possible behaviors of the evolutive system are captured by a set of events that can affect its current state. Diagnosing amounts to determining the history of the system, in terms of a sequence of events representing what may have happened to the system between two or more successive observations. This approach is particularly useful when explaining successive observations requires taking the history of the system into account. It allows precedence constraints between faults, as well as intermittent and non-intermittent faults to be modeled. It provides a unified representation and treatment of the various types of evolutions of the system's state: evolutions due to internal faults, to exogeneous events, as well as to repair actions.

1 Motivation

Model-based diagnosis, which consists in identifying malfunctioning components in a device by confronting observations to predictions from a model of the device's behavior, has received a lot of attention in this last decade. For long, model-based diagnosis has been studied in the context of static systems, i.e., systems that are assumed to be invariant throughout the diagnosis process [19, 7]. In this context, little attention has been paid to the history (past, evolution) of the device. A first step in this direction has been the use of non-intermittency assumptions [18] to improve discrimination between diagnoses.

More recently, it has been recognized that most real systems are changing along time. Recent research work claims the need for representing time in model-based diagnosis [13, 11, 3]. In that respect, the use of qualitative models as in [14, 9, 10] seems to be well-adapted to the monitoring of dynamic processes as long as qualitative laws can easily be provided. On the other hand, [11] proposes to extend the consistency-based approach in a very general way by allowing an explicit use of time points. A state of a system accounts for different time points and a temporal diagnosis is defined as a state that is consistent with the observations. This purely logical approach is powerful in that it allows the description of the statics and dynamics of the system within the same formalism, but suffers from computational complexity. [3] proposes to decompose a time-varying system into a sequence of static systems, each of which can be handled using traditional techniques. The dynamic part is captured by a set of component-oriented transition graphs. A temporal diagnosis is then a sequence of static diagnoses obeying the transition constraints between states. This decomposition is interesting to gain efficiency and makes the reuse of the machinery developed for static domains possible. However, it is based on the assumption that the evolution of the complete system can be represented by the independent evolution of its components, which is often too restrictive.

We propose to define temporal diagnoses as sequences of events (hence the name "event-based" diagnoses) rather than as states or sequences of states as it is usually done. A diagnosis problem consists then in answering the question "What happened to the device between two (or more) successive observations?" rather than the classical question: "In which (abnormal) state is the device?". The events constituting a diagnosis correspond to components changing from a mode to another, to external events which have affected the device's behavior, or to events relative to the own dynamics of the device.

The main advantages of this new view of diagnosis are the following:

- Our approach is quite adapted to the description of dynamic systems as soon as their behavior can be represented as a discrete event system. This latter assumption can reasonably be made, and underlies the work in qualitative modeling. This ability is particularly useful for tasks that require taking the history of the system into account in order to explain successive observations, such as monitoring evolutive systems.

Our framework is suitable to static systems as well. The system is then supposed to have changed from a normal state (in which all components are correct) to the current state.

- The event-based approach provides a unified representation of the various evolution types of the system: internal changes, exogeneous events as well as repair actions.
- Temporal constraints between events, such as precedence constraints between faults, can easily be expressed via the preconditions of the events (i.e., the conditions under which the event may occur). Non-intermittency of faults can also be expressed by the absence of an event allowing a component to turn from a faulty to a normal state.
- In our framework, the diagnosis process amounts to finding a "preferred" scenario followed by the system according to observations made at successive time instants. Preferences between diagnoses can be naturally expressed as preferences between sequences of events as will be seen in Section 4.

The paper is a short version of [5]; it is organized as follows. Section 2 introduces the notion of a system in our framework, along with the example[1] that will be used throughout this paper. Section 3 defines an event-based diagnosis problem and characterizes its solutions. Section 4 and 5 deal with preferred and explanatory diagnoses, respectively. Section 6 compares our approach to related work, and discusses future work.

2 System

We begin with defining the notion of a system in our framework. We consider two types of information in order to model the behavior of a system.

Information referring to the system's statics. Relevant features of the states of a system are captured by a vector of state variables $X = (x_1, \dots x_n)$, each variable x_i ranging over a domain D_i. A state of the system is then just a value assignment to each of the state variables, i.e., an element of the state space $D = \Pi_{i=1}^{n} D_i$. Not all the states in D are physically possible; in general, there exist static constraints among the states variables, which define a subspace $PS \subseteq D$ of *possible* states.

An example of a system is that of a mission of exploration of Mars using an autonomous robot. The robot, when supplied with power, goes from its initial

[1] This example is more representative of a supervision problem than of a diagnosis problem strictly speaking; see [5] for an example of the later type in a static context.

location (1) to another location (2), sends some data concerning this latter location using its emitter, and then returns to its initial location. Mars's ground being uneven, the robot might get stuck on its way, somewhere between 1 and 2. We might represent the states of this system using four states variables: $location \in \{1, 2, \mathsf{b}\}$ (b denoting some place between 1 and 2), $emitter \in \{\mathsf{c}, \mathsf{a}\}$ (c standing for correct, and a for abnormal), $power\text{-}supply \in \{\mathsf{c}, \mathsf{a}\}$, and $sent \in \{\mathsf{t}, \mathsf{f}\}$, a boolean variable true iff the robot has sent the data. The static constraints are the following: the robot cannot be in location 2 with an abnormal power supply and have sent data; if the robot is in b, i.e., on its way between 1 and 2, then the power supply is correct. This representation yields the eighteen possible states shown in Figure 1.

Usually, the values of the state variables are not directly observable, and some other features must be observed in order to estimate the state of the system. Unless there is a bijection between observations and states, knowledge of the observations leads to several possible states that are consistent with them, and vice versa.

For our mission example, we assume that the quantities that can be observed are whether or not the data have been *received* (and they have been iff they have been sent while the emitter was correct), and the location of the robot, given by some sensor that sometimes returns inaccurate information: whenever it returns 1 (resp. 2), the robot is in 1 (resp. 2); when the robot is in b, the sensor can never determine its location and returns ?; sometimes, despite the robot is in 1 or 2, the sensor cannot determine its location, and returns ? as well. The set of consistent observations $(location\text{-}sensed, data\text{-}received)$ associated with each state is also shown in Figure 1.

Information referring to the system's dynamics. Events of several kinds can modify the current state of the system. First, these events can be part of the system's dynamics, such as the event of the robot going from a location to another. Second, events can correspond to the change of a component of the system from one behavior mode to another, e.g., from correct to abnormal. Third, external events can occur, which affect the system's state. A repair action is such an event. Last, depending on the level of abstraction chosen in representing the system, some events can be a combination of those of the previous types.

In our example, we will consider the following events:

go: the event of the robot going from location 1 to 2; its precondition, i.e., the condition under which it may occur, is that the robot must be in 1 ($location = 1$), must not have yet sent the data ($sent = \mathsf{f}$), and that the power supply must be correct ($power\text{-}supply = \mathsf{c}$); its effect, assuming everything unchanged unless specified, is that the robot is in 2 ($location = 2$).

ab-go: the event of the robot leaving 1 for 2 but getting into troubles on its way. precondition: as for go, $location = 1$, $sent = \mathsf{f}$, and $power\text{-}supply = \mathsf{c}$; it has two undeterministic effects: either the robot remains stuck between 1 and 2 ($location = \mathsf{b}$), or it eventually reaches 2 but has no power left ($location = 2$ and $power\text{-}supply = \mathsf{a}$) and thus will be unable to send the data and return to 1.

send: the event of the robot sending the data when its emitter is initially correct. precondition: $location = 2$, $sent = \mathsf{f}$, $power\text{-}supply = \mathsf{c}$, and $emitter = \mathsf{c}$; it has two undeterministic effects: either nothing is changed beyond $sent = \mathsf{t}$, or during the sending operation the emitter breaks, leading to $sent = \mathsf{t}$ and $emitter = \mathsf{a}$. Alternatively, we could have described send as a deterministic event which always achieves the sending operation. We would then have needed another deterministic event representing the emitter changing from normal to abnormal.

ab-send: the event of the robot sending the data when its emitter is initially abnormal (note that the robot has no means to know whether or not the emitter is correct and then sends the data anyway); precondition: $location = 2$, $sent = \mathsf{f}$, $power\text{-}supply = \mathsf{c}$, and $emitter = \mathsf{a}$; effect: $sent = \mathsf{t}$.

return: the event of the robot returning from 1 to 2; precondition: $location = 2$, $sent = \mathsf{t}$, and $power\text{-}supply = \mathsf{c}$; effect: $location = 1$.

ab-return: is the event that is to return what ab-go is to go; precondition: $location = 2$, $sent = \mathsf{t}$, and $power\text{-}supply = \mathsf{c}$; effects: either $location = 1$ and $power\text{-}supply = \mathsf{a}$, or $location = \mathsf{b}$.

Note that, due to the simplicity of the example, there is no external event, and that faults, such as the change of the emitter and the power supply from correct to abnormal, are assumed to be permanent. However, we can easily modify those features by extending the example to include repair actions.

We consider all these types of events on a common basis, as relations between two possible states of the system. If (s_1, s_2) belongs to such a relation (or event) e, it means that e may occur in state s_1, and that it might result in s_2. For instance, with the representation of the states of the mission given above, the go event has $\big((1,\mathsf{c},\mathsf{c},\mathsf{f}),(2,\mathsf{c},\mathsf{c},\mathsf{f})\big)$ and $\big((1,\mathsf{a},\mathsf{c},\mathsf{f}),(2,\mathsf{a},\mathsf{c},\mathsf{f})\big)$ as members, that is $(1,3)$ and $(6,8)$ using the numbering of the states given in Figure 1.

Using possible states and events as information, a system can then be viewed as a labeled directed graph whose nodes are the possible states and whose arcs are labeled with events. In the rest of the paper, we call this graph *system's graph.* Letting $\mathcal{P}(PS \times PS)$ denote the set of relations on $PS \times PS$ (that is, the powerset of $(PS \times PS)$), a system is formally defined as follows:

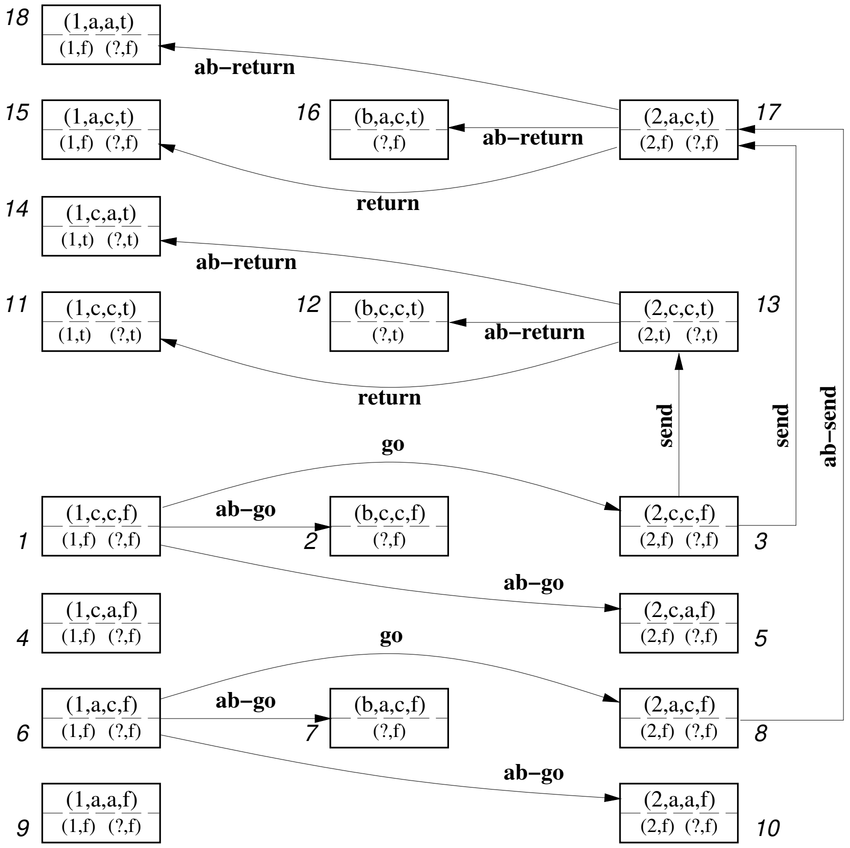

Figure 1: The system's graph for the robot mission. The upper part of each box represents a state (*location*, *emitter*, *power-supply*, *sent*), and its lower part represents the set of consistent observations (*location-sensed*, *data-received*) associated to that state.

Def. 1 (System Description) *Given a vector of state variables* $X = (x_1, \ldots, x_n)$, *with* x_i *ranging over* D_i, *a system is a pair* $(PS, \mathcal{E})$, *where*

- $PS \subseteq \Pi_{i=1}^{n} D_i$ *is the set of possible states of the system,*
- $\mathcal{E} \subseteq \mathcal{P}(PS \times PS)$ *is the set of events that can affect the system's state.*

In the following, SYS_M denotes the system corresponding to our mission example, whose graph is shown in Figure 1.

Note that the definitions given in this paper are intentionally independent of the way their various components can be represented. For example, we prefer dealing with a set of possible states of the system rather than with a set of formulae whose models represent the aforementioned states. In practice, an implicit representation of the system's graph must be adopted. In [5], we show how action formalisms as used in planning can provide a concise representation of knowledge about the system being diagnosed.

3 Event-Based Diagnosis

Given our definition of a system, we propose to define diagnoses as possible scenari (or sequences of events) that might have been followed by the system, so as to yield the observations gathered at successive time instants. Intuitively, a scenario represents a set of paths of the system's graph: this set contains those paths that are labeled with the sequence of events constituting the scenario. All this is captured in the following definition.

Def. 2 (Scenario) *Let $SYS = (PS, \mathcal{E})$ be a system. A scenario E for SYS is a (possibly empty) sequence $[e_1, e_2, \ldots, e_n]$ such that $e_i \in \mathcal{E}$ for all $i \in \{1, \ldots, n\}$. Note that an event may have multiple occurrences in a scenario. $|E|$ denotes the length of scenario E.*

A scenario $E : [e_1, \ldots, e_n]$ for SYS denotes the following relation on $PS \times PS$:

$$\begin{cases} \{\ (s_1, s_{n+1}) \ \mid\ \exists s_2 \in PS, \ldots, \exists s_n \in PS \ s.t. & \\ \qquad\qquad\qquad\quad \forall i \in \{1, \ldots, n\}(s_i, s_{i+1}) \in e_i\ \} & \text{if } |E| \geq 1 \\ I_{PS} & \text{otherwise} \end{cases}$$

where $I_{PS} = \{(s, s) \mid s \in PS\}$ denotes the identity relation on $PS \times PS$.

Let $s_1 \in PS$ and $s_2 \in PS$ be two possible states. We say that a scenario E might lead from s_1 to s_2 iff $(s_1, s_2) \in E$. That is, scenario E might lead from the initial state to the final state of each of the paths of the system's graph labeled with E. For instance, $[\mathsf{go}, \mathsf{send}]$ is a scenario for SYS_M. It might lead from state 1 to state 13 and from state 1 to state 17 through state 3. In the following, a scenario will denote, according to the context, either the corresponding sequence of events or the corresponding relation between initial states and final states.

An event-based diagnosis problem is defined as follows.

Def. 3 (Event-Based Diagnosis Problem) *An event-based diagnosis problem given $m \geq 2$ successive observations, is a pair $(SYS, [OBS_1, \ldots, OBS_m])$, where*

- *$SYS = (PS, \mathcal{E})$ is a system,*
- *$OBS_1 \subseteq PS, \ldots, OBS_m \subseteq PS$, OBS_i being the set of the possible states in which the system could be given the observations made at instant i.*

Note that in this definition, the OBS_i are set of states rather than sets of observations, and that we assume the former to be constructed from the latter. In fact, the OBS_i correspond exactly to *static* diagnoses. For our mission example, having observed *data-received* to be equal to t and *location-sensed* to be equal to ?, the possible states are those where $sent = \mathsf{t}$, $emitter = \mathsf{c}$ and

$location = \mathsf{1}, \mathsf{2}$ or b. The set of those states (an OBS_i in the above definition) is then $\{11, 12, 13, 14\}$.

We first define an event-based diagnosis for the case where two observations are available, and generalize to m observations later. Given two successive observations, a diagnosis is a scenario that might lead from some possible state verifying the first observation to some possible state verifying the second. In terms of the system's graph, a diagnosis given two observations is the label of a path leading from a state verifying the first observation to a state verifying the second observation.

Def. 4 (Event-Based Diagnosis) [given two successive observations] *An event-based diagnosis for* $(SYS, [OBS_1, OBS_2])$ *is a scenario* E *for* SYS *such that*

$$\exists s_1 \in OBS_1,\ \exists s_2 \in OBS_2 \quad s.t. \quad (s_1, s_2) \in E.$$

Recall that a scenario E defines a relation on $PS \times PS$ and may consist of more than one event. Note that the empty scenario is a diagnosis if and only if the two observations are consistent, i.e, iff $OBS_1 \cap OBS_2 \neq \emptyset$.

We now generalize the previous definition to the case where more than two observations are available. We define an event-based diagnosis given m consecutive observations as the concatenation of $m-1$ event-based diagnoses given two consecutive observations, subject to the condition that two consecutive diagnoses in the concatenation agree on intermediate states: the first diagnosis must possibly lead to a state from which the second diagnosis may start.

In the remainder of this paper, we use the following notation. Given two scenari $E_1 = [e_{1,1}, \dots, e_{1,n}]$ and $E_2 = [e_{2,1}, \dots, e_{2,m}]$ for the same system SYS, we note $E_1 \bullet E_2$ the concatenation of E_1 and E_2, i.e. $[e_{1,1}, \dots, e_{1,n}, e_{2,1}, \dots, e_{2,m}]$. Note that this concatenation is also a scenario for SYS.

Def. 5 (Event-based Diagnosis) [given m successive observations] *An event-based diagnosis for* $(SYS, [OBS_1, \dots, OBS_m])$*,* $m \geq 2$*, is a scenario* E *for* SYS *such that*

$$\begin{array}{l}\exists E_1, \dots, E_{m-1},\ \exists s_1 \in OBS_1, \dots, \exists s_m \in OBS_m \quad s.t. \\ E = E_1 \bullet \dots \bullet E_{m-1} \quad and \quad \forall i \in \{1, m-1\}\ (s_i, s_{i+1}) \in E_i.\end{array}$$

Hence, a diagnosis given m consecutive observations, corresponds to a path in the system's graph which goes through intermediate states successively verifying the observations made at successive times.

Example Consider the problem ψ_M for the robot mission, described as follows. Initially, the location of the robot is observed to be $\mathsf{1}$ and no data have been received. Hence, $OBS_1 = \{1, 4, 6, 9, 15, 18\}$. Later on, the location sensor returns $\mathsf{2}$ and no data have been received. Hence, $OBS_2 = \{3, 5, 8, 10, 17\}$. The diagnoses for these two observations are [go] leading from 1 to 3 and from

6 to 8, [go,send] leading from 1 to 17, [ab-go] leading from 1 to 5 and from 6 to 10, and [go,ab-send] leading from 6 to 17.

Third step: the sensor cannot determine the location of the robot, but data have been received. Thus, $OBS_3 = \{11, 12, 13, 14\}$. The diagnosis [go] obtained at step 2 can be extended, which yields diagnoses [go]•[send], leading from 1 to 13 through 3, [go]•[send,return], leading from 1 to 11 through 3 and 13, and [go]•[send,ab-return], leading from 1 to 12 and from 1 to 14 through 3 and 13. As can be seen from Figure 1, the three other diagnoses obtained at step 2 cannot be extended. [ab-go] cannot be extended because there is no way to get out from states 5 and 10 (the robot is stuck on its way). The diagnoses [go,send] and [go,ab-send] cannot be extended either: they both lead to state 17 in which the emitter is broken, a fault mode assumed to be permanent in our example. As long as no repair action is undertaken, no event exists that enables getting out of this mode (e.g., there is no path enabling to go from state 17, which is the state to which [go,send] leads at step 2, to a state where data can be received as it is the case at step 3. In sum, the diagnoses for ψ_M are [go,send], [go,send,return], and [go,send,ab-return].

4 Preferred Diagnoses

Unless a system is highly stable (i.e., if a large number of states are permanent), an event-based diagnosis problem often has an infinite number of solutions. Even if this number is finite, it is necessary to express preferences among diagnoses, which is the problem we address now.

4.1 Most Probable Diagnoses

When probabilistic information can be provided, it obviously makes sense to prefer diagnoses that, given the observations, are the most probable. To this end, we must be provided with[2]:

- for each state s, the (relative) probability of leaving s by transiting to state s' via event e; this probability is noted $p(e, s, s')$, and must verify $\sum_{e \in \mathcal{E}} \sum_{s' \in PS} p(e, s, s') = 1$ for all states s that can be left; Figure 2 represents $p(e, 1, s')$ for the mission example.
- the probability $p(s, OBS)$ of s being the actual state, given that this state is in OBS; this probability must verify $\sum_{s \in OBS} p(s, OBS) = 1$.

[2] As for Markov Processes, we make the assumption that the state transition probabilities depend only on the previous state of the system.

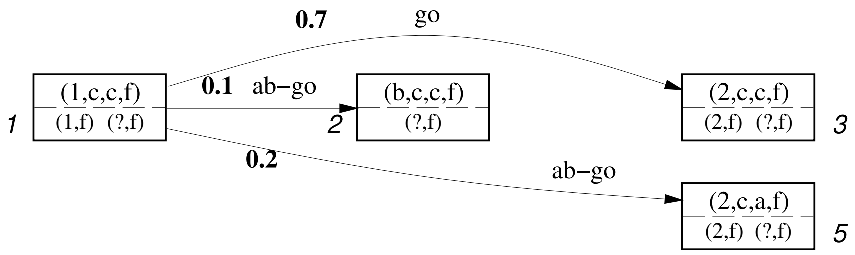

Figure 2: $p(e, 1, s')$ for the mission example

From this, we can recursively define the probability $p(E, s, OBS)$ of scenario E occurring from state s and leading into a state in OBS as follows:

$$p([], s, OBS) = \begin{cases} 1 & \text{if } s \in OBS \\ 0 & \text{otherwise} \end{cases}$$
$$p(e \bullet E, s, OBS) = \sum_{s' \in PS} p(e, s, s')\ p(E, s', OBS)$$

Now, the probability of E occurring from a state in OBS_1 and leading to a state in OBS_2 is:

$$p(E, [OBS_1, OBS_2]) = \sum_{s \in PS} (p(s, OBS_1) \times p(E, s, OBS_2)) \tag{1}$$

In order to properly compute the probability of scenario E, i.e., the probability of E occurring from OBS_1 given that OBS_2 is observed, we would have to divide this result by the probability of OBS_2 being observed. In general, this probability can only be computed with some approximation. Since we only seek the correct determination of the probability ranking of each diagnosis, the denominator, which is just a normalization factor, does not need to be computed.

Equation (1) only applies to two observations. Its generalization to multiple observations is straightforward and is given in [5].

4.2 Other Preference Criteria

Most of the time, probabilistic information cannot be provided, and more pragmatic preference criteria are used to discriminate between diagnoses. The following criteria can be naturally expressed in our framework:

shortest diagnosis: a diagnosis such that no scenario of smaller length is a diagnosis,

minimal diagnosis: a diagnosis such that no strict subsequence of it is a diagnosis,

least redundant diagnosis: a diagnosis such that no strict subsequence of it is a diagnosis defining the same relation on $PS \times PS$.

Least-redundancy is of particular interest when the system's graph is cyclic (e.g., due to repair actions), so as to discard infinite diagnoses. In [5], these criteria are examined in detail and we show how they can be seen as an approximate measure of the probability of the diagnoses.

5 Explanatory Diagnoses

Explanatory diagnoses are frequently dealt with in the literature, as an acknowledgement of commonsense's needs for explanation of observations. In the event-based diagnosis framework, explanatory diagnoses are scenari that, if they occur, *necessarily* yield the observations. The definition of diagnoses (Definition 4) only requires a diagnosis to *possibly* lead to one of the states in OBS_2 from at least one state in OBS_1. In terms of the system's graph, this amounts to saying that there exists *some* path labeled with the diagnosis which starts from some state in OBS_1 and leads to some state in OBS_2. The diagnoses are then only consistent with the observations. In order to define explanatory diagnosis, we must further require that *every* path labeled with the diagnosis and starting from a state in OBS_1 leads to a state in OBS_2. This is captured in the following definition, which furthermore generalizes to m observations.

Def. 6 (Explanatory Event-Based Diagnosis)
An explanatory event-based diagnosis for $(SYS, [OBS_1, OBS_2])$ *is an event-based diagnosis* E *for* $(SYS, [OBS_1, OBS_2])$ *such that*

$$\forall s_1 \in OBS_1, \forall s_2 \in PS, \text{ if } (s_1, s_2) \in E \text{ then } s_2 \in OBS_2.$$

An explanatory event-based diagnosis for $(SYS, [OBS_1, \ldots, OBS_m])$, $m \geq 2$, *is an event-based diagnosis* E *for* $(SYS, [OBS_1, \ldots, OBS_m])$ *such that* $E = E_1 \bullet \ldots \bullet E_{m-1}$*, where*

$$\forall i \in \{1, \ldots, m-1\} \quad E_1 \bullet \ldots \bullet E_i \text{ is an explanatory event-based diagnosis for } (SYS, [OBS_1, OBS_{i+1}]).$$

In the case of m observations, the previous definition requires, at each step, the explanation of the observation referring to this step.

Example Consider the problem ψ'_M for the robot mission. Initially, as for ψ_M, we have $OBS_1 = \{1, 4, 6, 9, 15, 18\}$. At the second step, as for ψ_M, $OBS_2 = \{3, 5, 8, 10, 17\}$. The diagnoses obtained at this second step are [go],[go,send],[ab-go], and [go,ab-send]. Among them, only [go] and [go,ab-send] are explanatory.

Third step, the robot is observed to be back at location 1, but no data have been received. We have $OBS_3 = \{1, 4, 6, 9, 15, 18\}$. We get four diagnoses: [go,send,return], [go,send,ab-return], [go,ab-send,return], and [go,ab-send,ab-return]. From figure 1, it can easily be seen that only one of them is explanatory: [go,ab-send,return]. According to this diagnosis, the emitter was abnormal since the beginning of the mission.

As noticed in [4, 1], requiring a diagnosis to explain all the observations is often too strong. It is then better to prefer diagnoses that explain the observations "as much as possible". To this end, let us examine explicative diagnoses from a probabilistic point of view. The rationale behind explanation is to choose the diagnosis E that maximizes the conditional probability of the observations given that E occurred. We can then define the explicative power of a diagnosis, i.e, the extent to which a diagnosis is explanatory, as the probability of OBS_2, given that E has occurred in OBS_1. If this probability equals 1, then E is an explanatory diagnosis as in Definition 6. The explicative power $ep(E, [OBS_1, OBS_2])$ of a diagnosis E can be computed from $p(E, [OBS_1, OBS_2])$ as follows:

$$ep(E, [OBS_1, OBS_2]) = \frac{p(E, [OBS_1, OBS_2])}{p(E, OBS_1)}$$

where $p(E, OBS_1)$ is the probability that E occurs from OBS_1:

$$\begin{array}{l} p(E, OBS_1) = \sum_{s \in PS}(p(s, OBS_1) \times p(E, s)), \quad \text{with} \\ \;p([], s) = 1 \\ \;p(e \bullet E, s) = \sum_{s' \in PS}(p(e, s, s') \times p(E, s')) \end{array}$$

Most explanatory diagnoses (diagnoses with a maximal explicative power) are different from most probable diagnoses, in that we seek to maximize the probability of the observations given the diagnosis, rather than that of the diagnosis given the observations. However, if all diagnoses E have the same probability of occurrence from OBS_1, i.e., if their $p(E, OBS_1)$ in the definition of the explicative power are equal, then selecting the most explanatory diagnoses amounts to selecting the most probable ones. This happens for instance in case diagnoses have the same length l, states in OBS_1 are equiprobable, and events have an equiprobability of occurrence in all states (i.e. $\sum_{s' \in PS} p(e, s, s') = k$ for all s and e). Indeed in that case, $p(E, OBS_1) = k^l$ for all diagnoses E.

6 Future and Related Work

In this paper, we have proposed to view the diagnosis task as the determination of the event-history of the system between successive observations. As more extensively stated in the introduction, this approach is particularly useful when explaining successive observations requires taking the history of the system into account. It allows precedence constraints between faults, as well as intermittent and non-intermittent faults to be modeled, and provides a unified representation and treatment of the various types of evolutions of the system's state: evolutions due to internal faults, to exogeneous events, as well as to repair actions.

The general framework presented in this paper has enabled us to formally define event-based diagnoses, independently of the way a system can be represented. Explicitly enumerating states in PS as well as the members of the relations capturing events, is not what we advocate. We clearly need adequate higher-level representations instantiating the general framework. In [5], we discuss the use of action formalisms based on state-change operators for this purpose. Another alternative is a representation based on the situation calculus, like that described by McIlraith in a recent paper [16]. Note that McIlraith's definition of explanatory diagnoses is a situation calculus instance of ours in the restricted case of deterministic actions and complete knowledge of the initial state of the system.

Let us compare our work to three types of related ones: work extending model-based diagnosis to evolutive systems, qualitative approaches, and situation recognition.

Within the first type of approaches, [11] advocates the extension of the consistency-based approach to the explicit use of time points. A diagnosis is a state of the system, accounting for different time points, which is consistent with the observations. This approach has the advantage of enabling the representation of both static and dynamic aspects of the system within the same logical formalism, but suffers from computational complexity. Our approach allows a more efficient way to diagnose a system by decoupling its static and dynamic behavior.

Decoupling atemporal and temporal diagnoses has already been proposed in [3]. The static part of the system is described by a set of first-order formulae (PS) and the atemporal diagnoses are obtained using the traditional techniques devoted to static systems. The dynamic part is represented by a set of Markovian transition graphs, each of which expresses the possible evolutions of a single component of the system. A temporal diagnosis is then a sequence of atemporal diagnoses obeying the transition constraints between states expressed by the transition graphs. A probabilistic reliability model is used to select the preferred diagnoses. This approach relies on the assumption that the global evolution of the system can be represented by the individual evolution of its components and that the components behaves independently, which is too restrictive in most cases. Moreover, unlike ours, the component-oriented representation is limited to internal changes and cannot account for external events or actions. The definition of temporal diagnoses in [3] is component-oriented while our definition is event-oriented. A component-oriented definition is mainly justified when dealing with component-oriented repair actions. If this is often the case for static devices such as digital circuits, it is not always sufficient when dealing with complex processes, for which actions are decided according to the identified sequence of events, rather than according to the components and their state.

[12, 20] propose the use of planning techniques to interleave repair actions and observation actions during the diagnosis process. Our proposal, viewing diagnoses as sequences of events, seems to be adequate to get a unified framework in which controlled events, such as repair or observation actions, and spontaneous events can be integrated. Thus, our approach is open towards on-line model-based repair of evolutive systems.

[2] describes an alternative view of update, where observations are incorporated into a belief set by explaining them in terms of a set of plausible events that might have caused these observations. This event-based model of update is close to our event-based view of diagnosis, but is limited to one set of observations while we consider successive sets of observations.

Let us now turn to qualitative approaches. Qualitative modeling has mainly been used for the monitoring of dynamic systems as in [9, 17, 6]. In these approaches, the possible evolutions of the system are represented by its qualitative states in a graph (envisionment graph) sharing some similarities with our system's graph. Changes of states are not expressed as explicit events but are produced by means of a qualitative simulation, which relies on the physical laws of the system and predicts the next states according to the system's dynamics. In our approach, the evolution of a system is represented in a more abstract way. Our aim is not to track the internal changes of the system by interpreting a flow of time-stamped observations, but to get a more repair-oriented view of an evolutive system.

Lastly, comparison can be made with situation recognition as proposed in [8], more dedicated to monitoring than, strictly speaking, to diagnosis. The system to be surveyed is described by the set of situation models (or scenari) of its normal and abnormal evolutions. A scenario corresponds to a set of elementary events (changes of values of observable features) with temporal constraints between them. The situation recognition system receives a stream of time-stamped observations as input, and recognizes an instance of a predefined scenario as this latter is developing. The main point to be noticed is that only predefined sequences of events can be recognized in the situation recognition approach. For combinatorial reasons, a complete set of such sequences can generally not be provided. The traditional remedy to this problem is to select scenari according to some a priori preferences. However, the drawback of this solution is that unpredicted events (surprises) cannot be taken into account. These problems are avoided with our approach, since the sequences of events explaining the observations are built and selected during the diagnosis process.

Future work will be mainly devoted to the computation of event-based diagnoses. In [5], we study the adequation of various search strategies in function of the features of the search space. In particular, the observations define layers in the search space, which makes it interesting to apply *à la* Viterbi algorithms [21]. Our preliminary investigations suggest that when observations are informative enough and the branching factor of the system's graph is small, efficient algorithms for computing a preferred event-based diagnosis can be obtained. Future work will also concentrate on exploring existing connections between event-based diagnosis and planning (see [5]), so as to adapt planning algorithms to account for the specificity of the event-based diagnosis problem. Another perspective is to explore more expressive representations (e.g., representations involving numeric time, parallel events, partially ordered sets of events, and abstraction). Finally, we want to exploit the potential of the event-based framework for the generation of repair plans, especially when interleaved with the diagnosis process.

References

[1] P. Besnard and M.-O. Cordier. Explanatory diagnoses and their computation by circumscription. In *Proc. ECAI-92*, pages 724–728, 1992.

[2] C. Boutilier. *Abduction to Plausible Causes: An Event-based Model of Belief Update.* Technical Report 94-9, University of British Columbia, March 1994.

[3] L. Console, L. Portinale, D. Theseider Dupre, and P. Torasso. Diagnostic reasoning across different time points. In *Proc. ECAI-92*, pages 369–373, 1992.

[4] L. Console and P. Torasso. Integrating models of the correct behavior into abductive diagnosis. In *Proc. ECAI-90*, pages 160–166, 1990.

[5] M.-O. Cordier and S. Thiébaux. *Event-Based Diagnosis for Evolutive Systems.* Research Report 819, IRISA, May 1994.

[6] D. DeCoste. Dynamic across-time measurement interpretation. In *J. Artificial Intelligence*, 51(1-3):273–342, 1991.

[7] J. de Kleer and B. Williams. Diagnosis with behavioral modes. In *Proc. IJCAI-89*, pages 1324–1330, 1989.

[8] C. Dousson, P. Gaborit, and M. Ghallab. Situation recognition: representation and algorithms. In *Proc. IJCAI-93*, pages 166–172, 1993.

[9] D. Dvorak and B. Kuipers. Model-based monitoring of dynamic systems. In *Proc. IJCAI-89*, pages 1238–1243, 1989.

[10] K. Forbus. Interpreting Observations of physical systems. In *IEEE Trans. on Systems, Man and Cybernetics*, pages 350–359, 1987.

[11] G. Friedrich and F. Lackinger. Diagnosing temporal misbehaviour. In *Proc. IJCAI-91*, pages 1116–1122, 1991.

[12] G. Friedrich and W. Nejdl. Choosing observations and actions in model-based diagnosis-repair systems. In *Proc. KR-92*, pages 489–498, 1992.

[13] W. Hamscher. Modeling digital circuits for troubleshooting. *J. Artificial Intelligence*, 51:223–271, 1991.

[14] B. Kuipers. Qualitative simulation. *J. Artificial Intelligence*, 29:289–338, 1986.

[15] F. Lackinger and W. Nejdl. Integrating model-based monitoring and diagnosis of complex dynamic systems. In *Proc. IJCAI-91*, pages 1123–1128, 1991.

[16] S. McIlraith. Towards a theory of diagnosing, testing and repair. In G. Provan, editor, *Fifth International Workshop on Principles of Diagnosis (DX'94)*, pages 185–192, New Paltz, NY, USA, 1994.

[17] H. T. Ng. Model-based, multiple fault diagnosis of dynamic, continuous physical devices. *IEEE Expert*, 6(6):38–43, 1991.

[18] O. Raiman, J. de Kleer, and V. Saraswat. Exploiting non-intermittency. In *Proc. 1^{st} Int. Workshop on Principles of Diagnosis*, pages 120–123, 1990.

[19] R. Reiter. A theory for diagnosis from first principles. *J. Artificial Intelligence*, 32:57–95, 1987.

[20] Y. Sun and D. Weld. A framework for model-based repair. In *Proc. AAAI-93*, pages 182–187, 1993.

[21] A.J. Viterbi. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm *IEEE Tr. on Inf. Theory* , 13(2):260–269, 1967.