Incremental decentralized diagnosis approach for the supervision of a telecommunication network[1]

Yannick Pencolé   Marie-Odile Cordier   Laurence Rozé [2]

Abstract

This paper extends the decentralized approach proposed in [5] for diagnosing discrete-event systems. It considers the incrementality issue which is a major one in the context of on-line diagnosis and proposes two solutions.

1 Introduction

The problem we deal with is the supervision of complex and large discrete-event systems such as telecommunication networks. Our purpose is to help operators of such systems to diagnose failures in the system according to observed events (alarms). For such complex and large systems, it is mandatory to use a decentralized approach. [3] proposes an approach for diagnosing discrete-event systems using decentralized and coordinated diagnosers. But the computation of each decentralized diagnoser needs a global model which is too large to be built with our applications. [1] and [2] propose methods based on a model-simulation approach which only needs a decentralized model, but these methods are used off-line to solve a diagnosis problem *a posteriori*.

Our motivation is to propose an approach which only needs decentralized information and can provide on-line diagnosis of a large discrete-event system (telecommunication network). In [5, 6], we have already defined a decentralized diagnosis approach that respects the constraints we are faced with. Nevertheless, in order to use this approach on-line, incrementality becomes a crucial issue in order to efficiently update already computed diagnoses by taking into account new observations received by the supervision center. The paper is organized as follows. We first describe a simple example of a telecommunication network that will be used as a running example throughout the paper. We then recall the decentralized approach Two solutions are then presented

2 Running example

We introduce a very simple telecommunication system (see Figure 1) that we shall use as a running example throughout the paper. It is formed by two switches ($SW1$ and $SW2$) which send and receive data, a control station CS which is in charge of managing the switches and a supervision center SC which is in charge of monitoring the system by receiving alarms from $SW1$, $SW2$ and CS. For reasons of simplicity of the example, the connections between the components are considered as safe. A failure is defined by two events: the beginning of the failure and the end of the failure. In this example, the failures are repaired without an operator intervention. $SW1$ and $SW2$ can boot or be blocked. Those failures are defined by their start-event ($SWiblk$, $SWibt$ where $i = \{1, 2\}$) and their end-event ($SWiback$, $SWiendbt$). When the switch SWi begins to block, it emits an alarm $SWistop$. Concerning the booting, the behaviors of the switches are different: when $SW1$ begins to boot then it emits $SW1stop$; when $SW2$ begins to boot then $SW2$ emits $SW2boot$. When the switch SWi begins to work well again, it emits the alarm $SWirun$. Nevertheless, when SWi is blocked, then it does not emit any alarm when it begins to boot. As $SW1$ and $SW2$, CS has two kinds of failures (blocked and booting) defined by the events ($CSblk$, $CSback$, $CSbt$, $CSendbt$). When CS begins to block, it emits $CSstop$. When CS begins to boot, it also emits a $CSstop$ alarm and sends to SWi a message $SWibt$ for the rebooting of the SWi switches. Nevertheless, when CS begins to boot whereas CS is blocked, it only sends a message to the SWi switches (no alarm is sent). When CS begins to work well again, it emits the alarm $CSrun$.

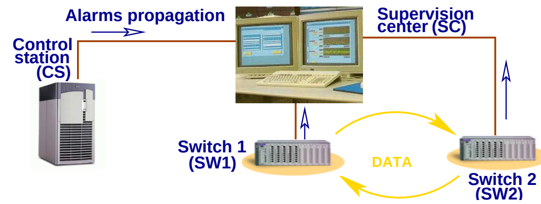

Figure 1: Running example of a simplified telecommunication network.

3 Overview of the decentralized approach

3.1 Model in a decentralized approach

This section explains how the model of the system is described in a decentralized way by means of local models, which describe the behaviors of each component of the sys-

[1] Work partially funded by RNRT (French Research Network in Telecom.) through the MAGDA project (http://magda.elibel.tm.fr).

[2] IRISA, Campus de Beaulieu, 35042 Rennes cedex, FRANCE, e-mail:{ypencole,cordier,roze}@irisa.fr

tem and the interactions between them. Each component can react to exogenous events such as failures by changing of states, emitting observable events and exchanging events with other components (named *internal events*). We make the hypothesis that no delays exist on the messages exchanged by the components. A *component* is faced to two kinds of received events: exogenous events (Σ^i_{exo}) such as failure events and internal events (Σ^i_{intrcv}). A component emits two kinds of events: observable events via its communication channel (Σ^i_{obs}) and internal events (Σ^i_{intemt}).

Definition 1 (Model of a component) *A* component *behavior is described by a communicating finite-state machine* $\Gamma_i = (\Sigma^i_{in}, 2^{(\Sigma^i_{out})}, Q_i, E_i)$ *where* Σ^i_{in} *is the set of input events (*$\Sigma^i_{in} = \Sigma^i_{exo} \cup \Sigma^i_{intrcv}$*)* Σ^i_{out} *is the set of output events (*$\Sigma^i_{out} = \Sigma^i_{obs} \cup \Sigma^i_{intemt}$*);* Q_i *is the set of states of the component;* $E_i \subseteq (Q_i \times \Sigma^i_{in} \times 2^{(\Sigma^i_{out})} \times Q_i)$ *is the set of transitions.*

The model of the system is described in a decentralized way by the models of its components.

Definition 2 (Model of a system) *The* model Γ *of a system is given by the set of models of its components* $\{\Gamma_1, \ldots, \Gamma_n\}$*, a set of exogenous events (*Σ_{exo}*), a set of observable events (*Σ_{obs}*) and a set of internal events (*Σ_{int}*) such that:* $\{\Sigma^1_{obs}, \ldots, \Sigma^n_{obs}\}$ *is a partition of* Σ_{obs}*;* $\{\Sigma^1_{exo}, \ldots, \Sigma^n_{exo}\}$ *is a partition of* Σ_{exo}*;* $\{\Sigma^1_{intrcv}, \ldots, \Sigma^n_{intrcv}\}$ *and* $\{\Sigma^1_{intemt}, \ldots, \Sigma^n_{intemt}\}$ *are partitions of* Σ_{int}*;* $\forall e \in \Sigma_{int}, \exists! \Gamma_i | e \in \Sigma^i_{intreceived} \wedge \exists! \Gamma_j | e \in \Sigma^j_{intemt} \wedge i \neq j$.

Figure 2 presents the model of our example. The observable events model the alarm emission ($CSstop$, $CSrun$, $SW1stop$, ...) (the alarms are received by local sensors, one sensor per component in this example). The exogenous events are the failure events which can occur on the system ($CSblk$, $CSback$, $SW1blk$, ...). The internal events model the propagation of the booting from the control station to the switches ($SW1bt$, $SW2bt$).

3.2 Diagnosis in a decentralized approach

The idea is to compute a diagnosis for each component (*local diagnosis*) and compute the diagnosis of the whole system (*global diagnosis*) by composing the local diagnoses.

3.2.1 Local diagnosis: Let denote $\mathcal{O}_{\Gamma_i}$ the sequence of local observations, i.e the events locally observed by a sensor plugged in the component Γ_i. We have $\mathcal{O}_{\Gamma_i} \in (\Sigma^i_{obs})^\star$. We suppose that, at the beginning of the task, the component is in one of the initial states $X^{\Gamma_i}_{init}$.

Given the model of the component Γ_i, a *local diagnosis* Δ_{Γ_i} describes the subset of trajectories from Γ_i starting from elements of $X^{\Gamma_i}_{init}$ which explains the sequence of *local* observations $\mathcal{O}_{\Gamma_i}$, i.e such that their projections on observable

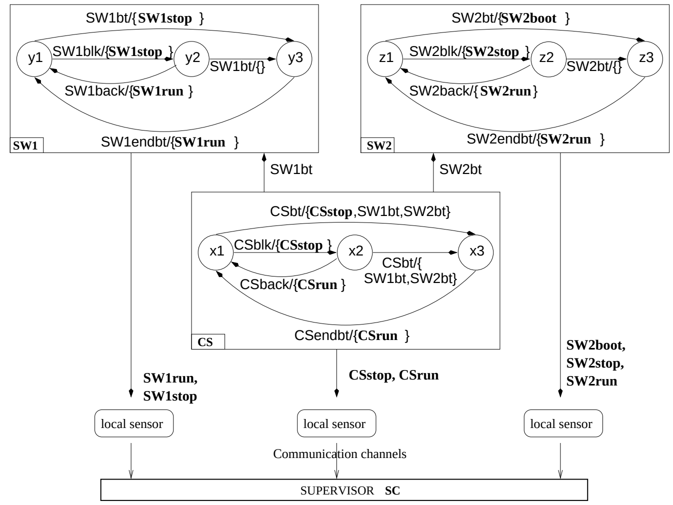

Figure 2: Model of the system.

events correspond exactly to $\mathcal{O}_{\Gamma_i}$. We propose to represent a local diagnosis as a communicating finite-state machine $\Delta_{\Gamma_i}(X^{\Gamma_i}_{init}, \mathcal{O}_{\Gamma_i})$, shortly Δ_{Γ_i}. Compared to the automaton Γ_i, the main syntactical difference is that each state Q_{Δ_i} of this automaton is associated to a pair $(s_{\Gamma_i}, \mathcal{E}_i)$ where $s_{\Gamma_i} \in Q_i$ is a state of the component and $\mathcal{E}_i$ is the prefix subsequence of $\mathcal{O}_{\Gamma_i}$ explained in this state. The initial states of Δ_{Γ_i} are those corresponding to $X^{\Gamma_i}_{init}$. The final states are those such that $\mathcal{E}_i = \mathcal{O}_{\Gamma_i}$ (states explaining the whole sequence of local observations). As in Γ_i, the transitions are labeled with exogenous or internal received events as input and with observed or internal emitted events as output.

Definition 3 (Local diagnosis) *The local diagnosis* $\Delta_{\Gamma_i}(X^{\Gamma_i}_{init}, \mathcal{O}_{\Gamma_i})$ *of* Γ_i *according to the sequence* $\mathcal{O}_{\Gamma_i}$ *is a finite-state machine:* $(\Sigma^i_{in}, 2^{(\Sigma^i_{out})}, Q_{\Delta_i}, E_{\Delta_i})$ *where* Σ^i_{in} *are the input events (*$\Sigma^i_{in} = \Sigma^i_{exo} \cup \Sigma^i_{intrcv}$*);* Σ^i_{out} *are the output events (*$\Sigma^i_{out} = \Sigma^i_{obs} \cup \Sigma^i_{intemt}$*);* $Q_{\Delta i}$ *is the set of states (*$Q_{\Delta i} \subseteq Q_i \times (\Sigma^i_{obs})^\star$*);* $E_{\Delta_i} \subseteq (Q_{\Delta_i} \times \Sigma^i_{in} \times 2^{(\Sigma^i_{out})} \times Q_{\Delta_i})$ *is the set of transitions.*

Figure 3 gives the local diagnosis of the control station when $X^{CS}_{init} = \{x1\}$ and $\mathcal{O}_{CS} = [CSstop]$. The way it is computed is not the subject of this paper. See [5] for details.

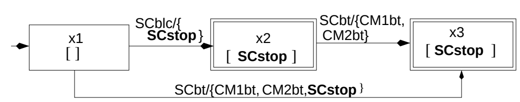

Figure 3: Local diagnosis: $\Delta_{CS}(\{x1\}, [CSstop])$.

3.2.2 Global diagnosis: The global observation $\mathcal{O}$ corresponds to what is observed by the supervisor which collects all the sequences of observable events (alarms) sent by each of the components through their own communication channels. It is described by $\{\mathcal{O}|\Sigma^1_{obs}, \dots, \mathcal{O}|\Sigma^n_{obs}\}$ where $\mathcal{O}|\Sigma^i_{obs} \in (\Sigma^i_{Obs})^*$ is the sequence of observations received from Γ_i.

Given a set of initial states X^Γ_{init}, a global diagnosis describes all the trajectories starting from elements of X^Γ_{init} which explain the global observation $\mathcal{O}$, i.e such that their projections on the observable events correspond to $\mathcal{O}$. The global diagnosis is thus represented as a communicating finite-state machine $\Delta_\Gamma(X^\Gamma_{init}, \mathcal{O})$, shortly Δ_Γ. The states Q_Δ of this automaton are pairs $(s_\Gamma, \mathcal{E})$ where $s_\Gamma \in Q_1 \times \dots \times Q_n$ is a state of the system and $\mathcal{E} = \{\mathcal{E}_1, \dots, \mathcal{E}_n\}$ where $\mathcal{E}_i$ is the prefix of the sequence $\mathcal{O}|\Sigma^i_{obs}$ explained in this state. The initial states of Δ_Γ are those corresponding to X^Γ_{init} and the final states are such that $\mathcal{E} = \mathcal{O}$, i.e $\forall i\, \mathcal{E}_i = \mathcal{O}|\Sigma^i_{obs}$ (states explaining the whole sequence of observations). The transitions are labeled with exogenous events as input (Σ_{exo}) and with observed events as output (Σ_{obs}).

Definition 4 (Global diagnosis) *The global diagnosis $\Delta_\Gamma(X^\Gamma_{init}, \mathcal{O})$ is a finite-state machine: $(\Sigma_{in}, 2^{(\Sigma_{out})}, Q_\Delta, E_\Delta)$ where Σ_{in} is the set of input events ($\Sigma_{in} = \Sigma_{exo}$); Σ_{out} is the set of output events ($\Sigma_{out} = \Sigma_{obs}$); Q_Δ is the set of states of the diagnosis; $E_\Delta \subseteq (Q_\Delta \times \Sigma_{in} \times 2^{(\Sigma_{out})} \times Q_\Delta)$ is the set of transitions.*

3.2.3 Computing global diag from local diags:

In a decentralized approach, the idea is to compute the global diagnosis from the local diagnoses. As local diagnoses are represented by automata, the global diagnosis is built by composing the local diagnoses.

Let us first define a property which expresses that the sequence of observations received by the supervisor from each component Γ_i ($\mathcal{O}|\Sigma^i_{obs}$) corresponds exactly to the sequence of local observations emitted by Γ_i ($\mathcal{O}_{\Gamma_i}$).

Definition 5 (Property 1) *The observed system is said to satisfy Property 1 iff $\mathcal{O}|\Sigma^i_{obs} = \mathcal{O}_{\Gamma_i}$.*

It can be proved that, on the condition that the observed system satisfies Property 1, the global diagnosis can be computed by using the following equation where $\odot$ is the classical composition operation between two communicating finite-state machines synchronized on the internal events exchanged between the local diagnoses:

$$\Delta_\Gamma(X^\Gamma_{init}, \mathcal{O}) = \bigodot_{i=1}^{n} \Delta_{\Gamma_i}(X^{\Gamma_i}_{init}, \mathcal{O}|\Sigma^i_{obs}) \qquad (1)$$

Figure 4 gives the global diagnosis of the model shown in Figure 2 for the initial state $X^\Gamma_{init} = \{(x1, y1, z1)\}$ and the global observation $\mathcal{O} = \{[CSstop], [SW1stop], []\}$. It was obtained by the operation: $\Delta_{CS}(\{x1\}, [CSstop]) \odot \Delta_{SW1}(\{y1\}, [SW1stop]) \odot \Delta_{SW2}(\{z1\}, [])$.

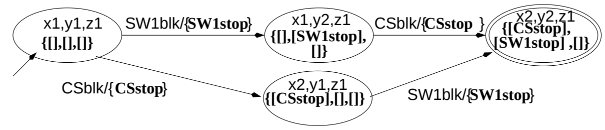

Figure 4: Global diagnosis of the system: $\Delta_\Gamma(\{(x1, y1, z1)\}, \{[CSstop], [SW1stop], []\})$.

3.2.4 Discussing Property 1:

As shown before, the computation of the global diagnosis from local diagnoses requires that the observed system satisfies Property 1. Property 1 expresses that what is received by the supervisor corresponds exactly to what is emitted by each component ($\mathcal{O}|\Sigma^i_{obs} = \mathcal{O}_{\Gamma_i}$). This property is clearly satisfied when the local sensors are directly observable, or when the messages they sent are received without delay to the supervisor. In most of the cases however, such local sensors are not directly observable, and the messages are sent to the supervisor via communication channels.

To fulfil Property 1, communication channels have to behave in such a way that i) the order in which messages are sent is the same as the one in which they are received (behaving as FIFO files) ii) all the sent messages are received. In the following, we suppose that i) is true and that there are no loss of messages.

4 Incremental diagnosis

Let us now turn to the problem of the incremental on-line computing of a diagnosis. The main difference is that the observations are considered on successive temporal windows. Having computed a global diagnosis for a given temporal window, the problem is thus to update it by taking into account the observations of the next temporal window. To keep the same computation way as before, Property 1 has clearly to be satisfied for each temporal windows and we show below that it is a central issue to preserve the correctness and the efficiency of an incremental algorithm.

Let us use the following notations : $\mathcal{O}_j$ described by $\{\mathcal{O}_j|\Sigma^1_{obs}, \dots, \mathcal{O}_j|\Sigma^n_{obs}\}$ represents all the observations that have been received from the beginning at time j. Δ_j is the diagnosis explaining $\mathcal{O}_j$. $\mathcal{W}_j$ denotes a temporal window. $\mathcal{O}_{\mathcal{W}_j}$ described by $\{\mathcal{O}_{\mathcal{W}_j}|\Sigma^1_{obs}, \dots, \mathcal{O}_{\mathcal{W}_j}|\Sigma^n_{obs}\}$ is the set of observations received during the temporal window $\mathcal{W}_j$. We have $\forall i, \mathcal{O}_j|\Sigma^i_{obs} = [\mathcal{O}_{j-1}|\Sigma^i_{obs}, \mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{obs}]$. $\Delta_{\mathcal{W}_j}$ is the diagnosis on the temporal window $\mathcal{W}_j$.

4.1 Problem of incremental diagnosis

The main problem is that, by randomly splitting the sequence of observations in temporal windows,there is no guarantee to have them satisfying Property 1.

Let us see what happens on the example of Figure 5 (subpart of the running example) with three components CS as Γ_1, $SW1$ as Γ_2 and $SW2$ as Γ_3. Each component has only two states and one transition. The initial states of the components are respectively $x1$, $y1$ and $z1$.

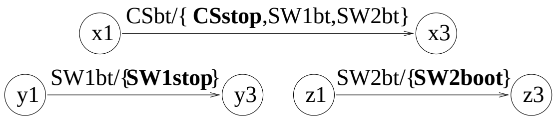

Figure 5: Simplified model of the system.

Let us first take a single temporal window $\mathcal{W}$ with $\mathcal{O}_{\mathcal{W}} = \{[CSstop], [SW1stop], [SW2boot]\}$. The global diagnosis, computed as described before, is given by Figure 6.

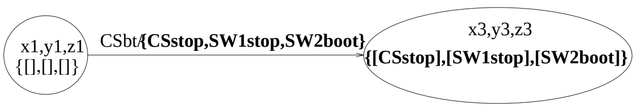

Figure 6: Global diagnosis for $\mathcal{O}_{\mathcal{W}}$ with the simplified model.

Let us now consider two successive temporal windows $\mathcal{O}_{\mathcal{W}_1} = \{[CSstop], [], []\}$ and $\mathcal{O}_{\mathcal{W}_2} = \{[], [SW1stop], [SW2boot]\}$. The local diagnosis (see Figure 7) $\Delta_{\mathcal{W}_1,\Gamma_1}$ explains $CSstop$ but requires synchronizations on $SW1bt$ and $SW2bt$ with $\Delta_{\mathcal{W}_1,\Gamma_2}$ and $\Delta_{\mathcal{W}_1,\Gamma_3}$. They are not satisfied and no global diagnosis is then found. The problem is that, during $\mathcal{W}_1$, $SW1stop$ and $SW2boot$ have been emitted by the component but are still not received by the supervision center. Both alarms will be received during $\mathcal{W}_2$. Property 1 is not satisfied on $\mathcal{W}_1$.

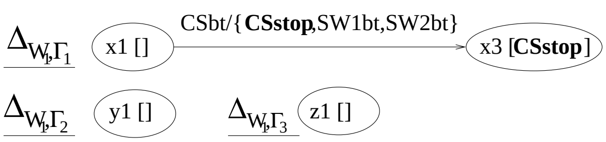

Figure 7: Local diagnoses for $\mathcal{O}_{\mathcal{W}_1}$ with the simplified model.

This example illustrates that the choice of temporal windows is very important. We firstly examine the case where, by choosing appropriate breakpoints, it is possible to ensure that each temporal window satisfies Property 1. An algorithm, based on the concatenation of automata, is proposed and allows us to compute efficiently a diagnosis on successive temporal windows. We then examine the general case and show how, by extending the definition of diagnosis, it is still possible to use the same algorithm.

4.2 Incremental algorithm for sound temporal windows

A first solution is to carefully choose the breakpoints in order to ensure that the temporal windows satisfy Property 1.

Definition 6 *A temporal window* $\mathcal{W}_j$ *is* sound *wrt a sequence of observations* $\mathcal{O}_{j-1}$ *iff* $\forall o_2 \in \mathcal{O}_{\mathcal{W}_j}, \forall o_1 \in \mathcal{O}_{j-1}$, o_2 *has been emitted after* o_1 *has been received. Two successive sound temporal windows meet at a sound breakpoint.*

When the windows are sound, the global diagnosis Δ_j results from the concatenation of the diagnosis Δ_{j-1} with the diagnosis $\Delta_{\mathcal{W}_j}$. The only condition is that the final states of Δ_{j-1}, noted X_{final}, are considered as the initial states for $\Delta_{\mathcal{W}_j}$. $\Delta_{\mathcal{W}_j}$ is computed as before (eq 1) by $\Delta_{\mathcal{W}_j} = \Delta_\Gamma(X_{\mathit{final}}, \mathcal{O}_{\mathcal{W}_j}) = \bigodot_{i=1}^{n} \Delta_{\Gamma_i}(X_{\mathit{final}}^{\Gamma_i}, \mathcal{O}_{\mathcal{W}_j}|\Sigma_{obs}^i)$. The global diagnosis Δ_j is computed by the application of a refinement operator (noted $\oplus$) defined by the incremental algorithm 1. We have $\Delta_j = \Delta_{j-1} \oplus \Delta_{\mathcal{W}_j}$.

Algorithm 1 Refinement operation: $\Delta_j = \Delta_{j-1} \oplus \Delta_{\mathcal{W}_j}$

input: Past and current window diagnoses: $\Delta_{j-1}, \Delta_{\mathcal{W}_j}$
$\Delta_{tmp} \leftarrow Append(\Delta_{j-1}, \Delta_{\mathcal{W}_j})$
{Eliminating trajectories that do not explain all the observations $\mathcal{O}_j$}
for all $x = (s_\Gamma, (\mathcal{E}_1, \ldots, \mathcal{E}_n)) \in \mathit{final\_states}(\Delta_{tmp})$ **do**
  if $\exists \mathcal{E}_i$ such that $\mathcal{E}_i \neq \mathcal{O}_j|\Sigma_{obs}^i$ **then**
    {x is not a final state in the new diagnosis.}
    $\Delta_{tmp} \leftarrow ElimTraj(\Delta_{tmp}, x)$
  end if
end for
output: $\Delta_{\mathcal{W}_j} \leftarrow \Delta_{tmp}$

$Append$ is an operation based on the classical concatenation of finite-state machines [4]. $ElimTraj$ eliminates the states x from which we cannot find a diagnosis for the observations of $\mathcal{W}_j$. It also eliminates the states that are predecessors of x and have no other successors .

Figure 8 presents the update of the diagnosis of Figure 4. Observations of the new window $\mathcal{W}_j$ are $\mathcal{O}_{\mathcal{W}_j} = \{[CSrun], [], [SW2boot]\}$. In Figure 4, there is one final state $((x2, y2, z1), \{[CSstop], [SW1stop], []\})$. The global diagnosis $\Delta_{\{CS,SW1,SW2\}}((x2, y2, z1), \mathcal{O}_{\mathcal{W}_j})$ of $\mathcal{W}_j$ is appended to the final state of Figure 4. Here, there is no elimination because each previous final state is followed by an explanation of $\mathcal{O}_{\mathcal{W}_j}$. The resulting diagnosis does not contain any trajectory explaining $CSstop$ by the occurrence of $CSbt$. In fact, by assuming the window soundness, $CSstop$ has been necessarily observed before the $SW2boot$ emission: $CSbt$ cannot explain the $CSstop$ observation.

4.3 Incremental algorithm in the general case

It is not always possible to select sound breakpoints. In the general case, the temporal windows cannot be guaranteed to be sound. To comply with Property 1, the incremental diagnostic algorithm must deal with two kind of observable events: i) the observations received by the supervisor in the current temporal window; ii) the events emitted by the components which have not yet been received by the supervisor in the current temporal window. They are still in the communication channels.

The idea is thus to complete the set of received observa-

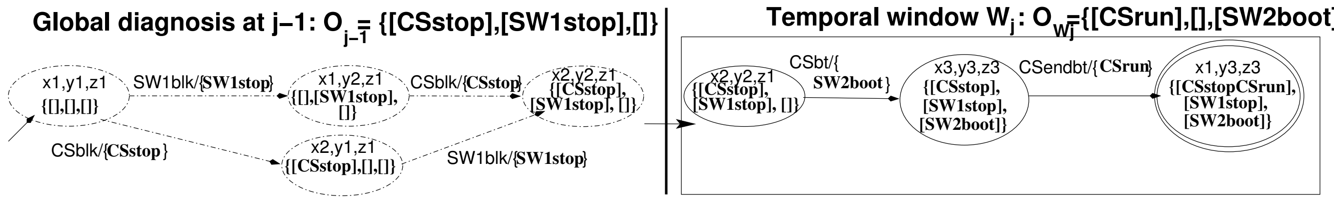

Figure 8: Update of the global diagnosis Δ_j.

tions by a set of potentially emitted (but not yet received) events. Therefore, we propose to compute, for each temporal window $\mathcal{W}_j$, an *extended* diagnosis $\Delta^{ext}_{\mathcal{W}_j}$ which summarizes trajectories that explain the events observed in $\mathcal{W}_j$ and a set of hypothetical unreceived events during $\mathcal{W}_j$. $\Delta^{ext}_{\mathcal{W}_j}$ is computed (see algorithm 2) by composing *extended* local diagnoses $\Delta^{ext}_{\Gamma_i}(\mathcal{W}_j)$ in a similar way as seen above.

4.3.1 Extended local diagnosis: The *extended* local diagnosis $\Delta^{ext}_{\Gamma_i}(\mathcal{W}_j)$ on the window $\mathcal{W}_j$ depends on the states of Γ_i described in the final states of the extended diagnosis Δ^{ext}_{j-1} (extended diagnosis of $\mathcal{O}_{j-1}$). In such a state x, $x = ((x_1, \dots, x_n), (\mathcal{E}_1, \dots, \mathcal{E}_n))$, some observations of Γ_i may have been supposed to be emitted and unreceived before $\mathcal{W}_j$, we have $\mathcal{E}_i = [\mathcal{O}_{j-1}|\Sigma^i_{obs}, SupposedObs_x]$. Looking to $\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{Obs}$, it can be checked whether this supposition is satisfied or not; if it is, we have two cases:

1. $\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{obs}$ is a prefix of $SupposedObs_x$. This means that $\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{Obs}$ was totally explained in the previous temporal window. Thus, $SupposedObs_x = [\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{Obs}, unObs_x]$ where $unObs_x$ is a sequence of observations of Γ_i potentially emitted before $\mathcal{W}_j$ and not yet observed during $\mathcal{W}_j$.

2. $SupposedObs_x$ is a prefix of $\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{obs}$. This means that $\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{Obs}$ was partially explained in the previous temporal window. Thus, $\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{Obs} = [SupposedObs_x, UnExplainObs_x]$ where $UnExplainObs_x$ is the sequence that terminates $\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{Obs}$ which is not yet explained.

In the first case, x is a state resulting from trajectories which already explain the observations $\mathcal{O}_{\mathcal{W}_j}|\Sigma^i_{Obs}$. We only need to determine local trajectories from x_i which explain hypothetical unreceived events which can follow the events of $unObs_x$. In the second case, we have to determine local trajectories from x_i which explain $UnExplainObs_x$ followed by hypothetical unreceived events. Therefore, in both cases, we have to determine hypothetical unreceived events. If we do the hypothesis that there is a bounded number k of local observations at the same time in the communication channel associated to Γ_i, then the sequence of hypothetical unreceived events of Γ_i is finite and belongs to:

Definition 7 ($UnRcvObs_i(k)$) *Let k be a positive integer. We note by $UnRcvObs_i(k)$ the sequences of observable events sq such that $sq \in (\Sigma^i_{obs})^\star, |sq| \leq k$.*

Thus from the state x, we have to compute the local trajectories of Γ_i which explain the sequences of $Comp_i(x,k)$ where $Comp_i(x,k)$ is a set of observation sequences. In the case 1, each sequence $obsSeq$ of $Comp_i(x,k)$ is such that $obsSeq \in UnRcvObs_i(k - |unObs_x|)$. In the case 2, $obsSeq = [UnExplainObs_x, UnRcvObs], UnRcvObs \in UnRcvObs_i(k)$. The extended diagnosis $\Delta^{ext}_{\Gamma_i}(\mathcal{W}_j)$ is the set of local trajectories computed from each final state x of Δ^{ext}_{j-1} that explain $Comp_i(x,k)$. Because we make hypotheses about the set of unreceived events during $\mathcal{W}_j$, each state resulting of a trajectory that explains such events is possible, so we mark it as a final state.

Algorithm 2 Extended diagnosis of $\mathcal{W}_j : \Delta^{ext}_{\mathcal{W}_j}$

input: $\mathcal{O}_{j-1}$, $\mathcal{O}_{\mathcal{W}_j}$
input: X^j_{init} {Final states of the diagnosis Δ^{ext}_{j-1} }
for all $i \in \{1, \dots, n\}$ **do**
  $\Delta_{tmp} \leftarrow \emptyset$
  for all $x \in X^j_{init}$ **do**
    $\{x = ((x_1, \dots, x_n), (\mathcal{E}_1, \dots, \mathcal{E}_n))\}$
    {Local observation sequences to explain}
    for all $obsSeq \in Comp_i(x,k)$ **do**
      $\Delta_{tmp} \leftarrow \Delta_{tmp} \cup \Delta_{\Gamma_i}(\{x_i\}, obsSeq)$
    end for
  end for
  $\Delta^{ext}_{\Gamma_i}(\mathcal{W}_j) \leftarrow \Delta_{tmp}$
end for
output: $\Delta^{ext}_{\mathcal{W}_j} \leftarrow \bigodot_{i=1}^{n} \Delta^{ext}_{\Gamma_i}(\mathcal{W}_j)$

4.3.2 Update of the global diagnosis: The global diagnosis Δ_j is computed, as before, by the application of the refinement operation (noted $\oplus$) (see algorithm 1). We have $\Delta^{ext}_j = \Delta^{ext}_{j-1} \oplus \Delta^{ext}_{\mathcal{W}_j}$. The current diagnosis Δ^{ext}_j describes trajectories that all explain $\mathcal{O}_j$ and some of them explain a set of complementary events supposed to have been emitted but not yet observed by the supervisor at $\mathcal{W}_j$. It is then clear that we have: $\Delta_\Gamma(X_{init}, \mathcal{O}_j) \subseteq \Delta^{ext}_j$. Finally, if we consider $\mathcal{O}_m$ as "complete", meaning that no more observation is expected ($\mathcal{W}_m$ is the last window), the extended diagnosis $\Delta^{ext}(\mathcal{W}_m)$ is computed with $k = 0$ (no expected

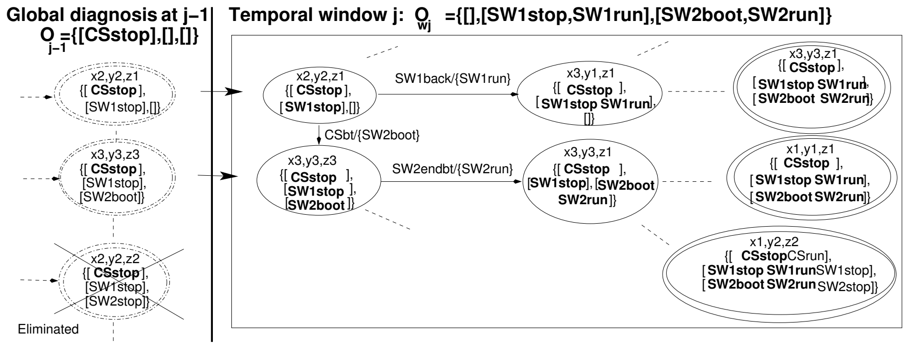

Figure 9: Update of the extended global diagnosis Δ_j^{ext}.

event). Thus, potential wrong assumptions made during the computation of the successive extended diagnoses are eliminated with help of the diagnosis of the last window. We get: $\Delta_\Gamma(X_{init}, \mathcal{O}_m) = \Delta_m^{ext}$.

4.3.3 Example:

Figure 9 presents the scheme of the Δ_j^{ext} computation. The extended diagnosis Δ_{j-1}^{ext} explains $O_{j-1} = \{[CSstop], [], []\}$. Some final states of Δ_{j-1}^{ext} not only contain the occurrence of O_{j-1} (in bold) but also other potentially unreceived alarms (here, each communication channel conveys at the most $k = 1$ observation at the same time). Given $\mathcal{W}_j$ and its observations $\mathcal{O}_{\mathcal{W}_j} = \{[], [SW1stop, SW1run], [SW2boot, SW2run]\}$, $\Delta_{\mathcal{W}_j}^{ext}$ is computed by considering the final states of Δ_{j-1}^{ext}. $\Delta_{\mathcal{W}_j}^{ext}$ summarizes the trajectories explaining $\mathcal{O}_{\mathcal{W}_j}$. We also complete the diagnosis by supposing the emission of other alarms. In $\Delta_{\mathcal{W}_j}^{ext}$, we suppose in particular the occurrence of $CSrun$, $SW1stop$ or $SW2stop$. Once $\Delta_{\mathcal{W}_j}^{ext}$ is computed, we apply the refinement algorithm. We append the initial states of $\Delta_{\mathcal{W}_j}^{ext}$ to the corresponding final states of Δ_{j-1}^{ext}. Some final states of Δ_{j-1}^{ext} are eliminated because they do not permit to find an explanation of $\mathcal{O}_{\mathcal{W}_j}$ (in Figure 9, the eliminated state considers the observation of $SW2stop$ whereas we observed $SW2boot$).

5 Conclusion

Our motivation was to extend the decentralized diagnosis approach initially presented by [5, 6] in order to be able to analyze, *on-line*, a flow of incoming observations. In an on-line context, the observations are considered on successive temporal windows. A crucial issue is to incrementally update the current diagnosis by taking into account the observations of the next temporal window.

Two solutions have been examined. The first one consists in carefully selecting the breakpoints which determine the temporal windows. We define a property (the $soundness$) which, when satisfied by the windows, allows us to use an easy and efficient refinement algorithm based on the concatenation of automata. It is not however always possible to determine such sound breakpoints. In the general case, we propose to complete the observations by guessing what is lacking and we consequently define extended local diagnoses. The refinement algorithm can still be used for the incremental computation of the global diagnosis.

In the first case, the issue is to use domain knowledge, and especially knowledge on the properties of the communication channels, in order to split the flow of observations in sound windows. For instance, when you know the maximal delay of transmission, the absence of any alarms received by the supervisor during a delay greater than this threshold determines a sound breakpoint. In the second case, an important issue to be studied is the optimal size of the window. Small windows mean small local diagnoses but frequent computations of the current global diagnosis whereas large windows mean space-consuming local diagnoses but less computations. This point is currently investigated.

References

[1] P. Baroni, G. Lamperti, P. Pogliano, and M. Zanella. Diagnosis of large active systems. *Artificial Intelligence*, 110:135–183, 1999.

[2] L. Console, C. Picardi, and M. Ribaudo. Diagnosis and diagnosibility analysis using pepa. In *Proceedings of the European Conference on Artificial Intelligence (ECAI'2000)*, pages 131–135, Berlin, 2000.

[3] R. Debouk, S. Lafortune, and D. Teneketzis. Coordinated decentralized protocols for failure diagnosis of discrete event systems. *Discrete Event Dynamic Systems*, 10(1-2):33–86, 2000.

[4] J. Hopcroft and J. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.

[5] Y. Pencolé. Decentralized diagnoser approach: application to telecommunication networks. In *Proceedings of the International Workshop on Principles of Diagnosis (DX'00)*, Morelia, Mexico, 2000.

[6] Y. Pencolé, M-O. Cordier, and L. Rozé. A decentralized model-based diagnostic tool for complex systems. In *thirteen IEEE international conference on tools with artificial intelligence (ICTAI'01)*, pages 95–102, Dallas Texas, USA, 2001.