Incremental Diagnosis of Discrete-Event Systems

Alban Grastien, Marie-Odile Cordier, Christine Largouët

Incremental Diagnosis of Discrete-Event Systems

Alban Grastien
Irisa – Université Rennes 1
Rennes – France
agrastie@irisa.fr

Marie-Odile Cordier
Irisa – Université Rennes 1
Rennes – France
cordier@irisa.fr

Christine Largouët
University of New Caledonia
Noumea – New Caledonia
largouet@univ-nc.nc

Abstract

When dealing with real systems, it is unrealistic to suppose that observations can be totally ordered according to their emission dates. The partially ordered observations and the system are thus both represented as finite-state machines (or automata) and the diagnosis formally defined as the synchronized composition of the model with the observations. The problem we deal with in this paper is that, taking into account partially ordered observations rather than sequential ones, it becomes difficult to consider the observations one after the other and to incrementally compute the global diagnosis.

In this paper, we rely on a *slicing* of the observation automaton and propose to compute diagnosis slices (for each observation slice) before combining them to get the global diagnosis. In order to reach this objective, we introduce the concept of *automata chain* and define the computation of the diagnosis using this chain, first in a modular way and then, more efficiently, in an incremental way. These results are then extended to the case where observations are sliced according to temporal windows. This study is done in an off-line context. It is a first and necessary step before considering the on-line context which is discussed in the conclusion.

1 Introduction

It is established that diagnosing dynamical systems, represented as discrete-event systems [Cassandras and Lafortune, 1999] amounts to finding what happened to the system from existing observations [Baroni *et al.*, 1999; Cordier and Thiébaux, 1994; Barral *et al.*, 2000; Console *et al.*, 2000; Lunze, 1999]. In this context, the diagnostic task consists in determining the trajectories (a sequence of states and events) compatible with the observations. When dealing with real systems, it is unrealistic to suppose that observations can be totally ordered according to their emission dates. The partially ordered observations and the system are thus both represented as finite-state machines (or automata) and the diagnosis formally defined as the synchronized composition of the model with the observations.

A problem that can be encountered is the size of the observation automaton, due to the temporal uncertainties on the observations or/and the duration of the observation recording. For instance, we may want to compute an a posteriori diagnosis from log files of observations during a few days period, as in the domain of telecommunication networks. In this article, we propose a way to avoid this global computation by slicing the observations automaton and building the diagnosis incrementally on successive observation slices. The problem of building the sliced observation automata is not considered in this paper where we consider it as given.

It should also be clear that this proposal is complementary of a decentralized approach. In the decentralized case, as for instance [Pencolé and Cordier, 2005], instead of globally considering the system model, diagnoses are computed locally for each component before being merged to get the global diagnosis. In our paper, instead of globally considering the observations, diagnoses are computed for each observation window, before being incrementally integrated to get the current global diagnosis.

After a brief reminder of the definitions about automata (section 2), we introduce, in section 3, the concept of *automata chain*, to represent an automaton by a sequence of automata slices. We provide the properties such an automata chain has to satisfy to be a *correct slicing* and define a *reconstruction* operation to get the global automaton back. Then, we demonstrate, provided the observations are correctly sliced, that the diagnosis can be correctly (section 4) and incrementally (section 5) computed from the observation slices. In section 6, these results are extended to the case where observations are sliced according to time, i.e according to *temporal windows*. We here focus on the off-line diagnosis context; the extension to the on-line diagnosis context is discussed in the conclusion.

2 Preliminaries: automata and trajectories

In this paper, we are more particularly interested in diagnosing reactive systems. Reactive systems are event-driven since their behaviour evolves with the occurrence of events and can cause by propagation a succession of state changes [Baroni *et al.*, 1999]. In this approach, the behavioural model of the system is represented by finite state machines. This section thus recalls some basic notions about automata and trajectories.

Definition 1 (Automaton). *An automaton A is defined by the t-uplet (Q, E, T, I, F) where:*

- *Q is the finite set of states;*
- *E is the finite set of events;*
- *$T \subseteq (Q \times 2^E \times Q)$ is the finite set of transitions. A transition t is a 3-uplet (q, l, q') such that t connects q to q' on the label l, with $l \subseteq E$ a subset of events. When $q \neq q'$ then l must be not empty. However, we consider that, $\forall q \in Q$, the implicit transition $(q, \emptyset, q)$ exists and belongs to T.*
- *I is the finite set of initial states ($I \subseteq Q$);*
- *F is the finite set of final states ($F \subseteq Q$).*

Definition 2 (Path). *A* path between the states q and q' *of an automaton $A = (Q, E, T, I, F)$ is the couple $((q_0, \ldots, q_m),(l_1, \ldots, l_m))$, where $(q_0, \ldots, q_m)$ is the finite sequence of states and $(l_1, \ldots, l_m)$ the finite sequence of labels, such that:*

- *$\forall i \in \{0, \ldots, m\},\ q_i \in Q$,*
- *$\forall i \in \{1, \ldots, m\},\ t_i = (q_{i-1}, l_i, q_i) \in T$,*
- *$q_0 = q$ and $q_m = q'$.*

Definition 3 (Trajectory). *A* trajectory, *denoted $traj$, in an automaton A is a path $((q_0, \ldots, q_m),(l_1, \ldots, l_m))$, where $q_0 \in I$ and $q_m \in F$.*

Two automata A and A' are equal ($A = A'$) if their trajectory sets are equal. We call *simplified automaton* of A, the automaton $A' = A$ from which all the states and transitions not reachable from an initial state or not leading to a final state have been removed. In the following, when computing new automata, only simplified ones are considered.

Let us consider the synchronization of two automata A_1 and A_2. If their transition labels share events, these common events, $E_1 \cap E_2$, are called synchronization events. To be synchronizable, two labels l_1 and l_2 must include exactly the same synchronization events. The synchronization label, when it exists, is $\Theta(l_1, l_2) = l_1 \cup l_2$. The synchronization consists in trigerring simultaneously the only transitions the labels of which are synchronizable.

Definition 4 (Synchronization of automata). *Given $A_1 = (Q_1, E_1, T_1, I_1, F_1)$ and $A_2 = (Q_2, E_2, T_2, I_2, F_2)$ two automata. The* synchronized automaton *of A_1 and A_2, denoted $A_1 \otimes A_2$, is the automaton $A = (Q, E, T, I, F)$ such that:*

- $Q = Q_1 \times Q_2$,
- $E = E_1 \cup E_2$,
- $T = \{((q_1, q_2), l, (q_1', q_2')) \mid \exists l_1, l_2, (q_1, l_1, q_1') \in T_1 \wedge (q_2, l_2, q_2') \in T_2 \wedge l = \Theta(l_1, l_2)\}$,
- $I = I_1 \times I_2$,
- $F = F_1 \times F_2$.

3 Automata chain

In this section we introduce the concept of *automata chain* and show that this concept can be used to represent an automaton as a sequence of automata slices, providing that a correct slicing property is satisfied. A synchronization operation on automata chains is then defined and properties are given which show that automata synchronization can be computed on automata chains. This point is used in section 4 to define diagnosis by slices.

An automata chain enables to slice an automaton into pieces. An automata chain is a sequence of automata whose main property (second bullet) is that a state is not allowed to appear in two distinct automata of the chain, except if it is a frontier state between two successive automata, *i.e* it is a final state of the former and an initial state of the later. Consequently, if a state belongs to the ith automaton and also to the jth automaton, with $j > i$, it appears in all the automata between the ith and the jth as a frontier state.

Definition 5 (Automata chain). *A sequence of automata $(A^1, \ldots, A^n)$ with $A^i = (Q^i, E^i, T^i, I^i, F^i)$ is called* automata chain, *and denoted $\mathcal{E}_A$, if:*

- *$\forall i, j,\ E^i = E^j$,*
- *$\forall i, j,\ j > i, \forall q,\ q \in Q^i \cap Q^j \Rightarrow q \in F^i \wedge q \in I^{i+1}$,*
- *$\forall i, j, \forall q, q',$ if $\{q, q'\} \subseteq Q^i \cap Q^j$ then $\forall p$, path of A^i between q and q', p is also a path of A^j.*

An automata chain is given in Figure 1. To simplify, the labels over the transitions are not represented.

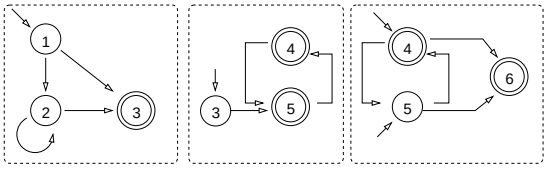

Figure 1: Chain of three automata

Let $\mathcal{E}_A$ be an automata chain $(A^1, \ldots, A^n)$. A trajectory of $\mathcal{E}_A$ is defined as being the ordered (from 1 to n) concatenation of n trajectories, one for each automaton. For instance, the path going from state 1 to state 6 through the states 3 and 5 is a trajectory of the automata chain of Fig. 1. Conversely, the path going from state 3 to 6 through 5 is not a trajectory.

Definition 6 (Correct slicing). *Let A be an automaton and $\mathcal{E}_A = (A^1, \ldots, A^n)$ an automata chain. $\mathcal{E}_A$ is a* correct slicing *of A iff the set of trajectories of $\mathcal{E}_A$ is equal to the set of trajectories of A. We denote $Sli(A)$ a correct slicing of A into an automata chain $\mathcal{E}_A$ such that $\mathcal{E}_A = Sli(A)$.*

The chain in Figure 1 is a correct slicing of the automaton of Figure 2.

From an automata chain, it is also possible to get back an automaton by the reconstruction operation defined below.

Definition 7 (Automaton reconstruction). *Let $\mathcal{E}_A = (A^1, \ldots, A^n)$ be an automata chain with $A^i = (Q^i, E^i, T^i, I^i, F^i)$. We call* reconstruction *of the chain $\mathcal{E}_A$, the simplified automaton obtained from $A_R = (Q_R, E_R, T_R, I_R, F_R)$ defined as follows:*

- $Q_R = Q^1 \cup \ldots \cup Q^n$,
- $E_R = E^1 = \ldots = E^n$,

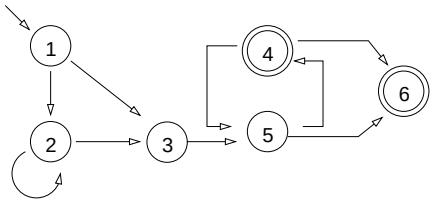

Figure 2: The automata chain of Fig. 1 is one of the correct slicings of this automaton. This automaton can be obtained by reconstruction of the automata chain given in Fig. 1.

- $T_R = T^1 \cup \ldots \cup T^n$,
- $I_R = I^1$,
- $F_R = F^n$.

The automaton of Figure 2 is obtained by reconstruction of the automata chain of Figure 1. It can be shown that, an automata chain, provided that it is a correct slicing of an automaton, gives back this automaton by reconstruction.

Theorem 1. *Let A be an automaton and $\mathcal{E}_A$ an automata chain. Let us denote the reconstruction operation by Sli^{-1}. If $\mathcal{E}_A$ is a correct slicing of A, then A is obtained by reconstruction of $\mathcal{E}_A$, i.e $A = Sli^{-1}(\mathcal{E}_A)$.*

Proof: It is given in Appendix.

It is important to remark that the automata chain of Figure 3 is also a correct slicing of the automaton of Figure 2. It contains states (as state 7, 8, 9) which are unnecessary and discarded by the reconstruction operation during the simplification step (they do not belong to any trajectory). An automata chain can be pruned from these unnecessary states without loss of information. This operation is called refinement. It can concern initial states which do not appear as final states in the preceding automaton (I-refinement) or final states which do not appear as initial states in the following automaton (F-refinement). In the following definition, the state q is removed from the set I^i of the automaton A^i.

Definition 8 (I-Refinement). *Let $\mathcal{E}_A = (A^1, \ldots, A^n)$ with $A^i = (Q^i, E^i, T^i, I^i, F^i)$. We call I-refinement of $\mathcal{E}_A$ a sequence $\mathcal{E}_{A'} = (A'^1, \ldots, A'^n)$ such that $\exists q, \exists i > 1$, $q \in I^i \wedge q \notin F^{i-1}$ with:*

- *$\forall j \neq i$, $A'^j = A^j$,*
- *A'^i is the simplified automaton obtained from $(Q^i, E^i, T^i, I^i \setminus \{q\}, F^i)$.*

F-refinement can be defined in an analog way as I-refinement.

Property 2. *Let $\mathcal{E}_A$ be an automata chain. The sequence of automata $\mathcal{E}_{A'}$ obtained by refinement of $\mathcal{E}_A$ is a chain. Moreover, the refinement operation on automata chain preserves the equality of the reconstructed automata.*

The proof is not given here.

Refinement operation enables us to get a smaller equivalent automata chain. After successive refinements, (two I-refinements removing states 8 and 9, and one F-refinement removing state 7), the automata chain of Figure 3 is refined in the chain of Figure 1.

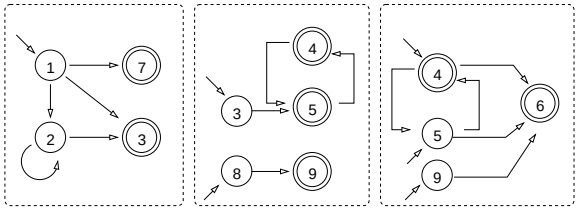

Figure 3: This automata chain is one of the correct slicings of the automaton of Fig. 2. When refined, this automata chain gives the one of Fig. 1.

Let us now turn to automata chain synchronization which is a key issue for defining diagnosis by slices (next section).

Definition 9 (Prefix- and suffix-closed automaton). *Let $A = (Q, E, T, I, F)$ be an automaton.*

We call prefix-closed automaton of A, *denoted A^+, the automaton A whose all states are final: $F^+ = Q$.*

We call suffix-closed automaton of A, *denoted A^-, the automaton A whose all states are initial: $I^- = Q$.*

We denote $A^\#$, the automaton which is both prefix-closed and suffix-closed ($A^\# = A^{+^-} = A^{-^+}$).

When synchronizing an automata chain with an automaton M, each automaton of the chain is synchronized with M. The only subtlety is that, except when synchronizing the first automaton of the chain, the initial states of M are not considered. In the same way, except when synchronizing the last automaton of the chain, the final states of M are not considered.

Definition 10 (Automata chain synchronization). *We call* synchronization of an automata chain $\mathcal{E}_A = (A^1, \ldots, A^n)$ with an automaton M *the sequence denoted $\mathcal{E}_A \otimes M$ defined by: $\mathcal{E}_A \otimes M = (A^1 \otimes M^+, A^2 \otimes M^\#, \ldots, A^{n-1} \otimes M^\#, A^n \otimes M^-)$.*

Theorem 3. *Let $\mathcal{E}_A$ be an automata chain and $M = (Q_M, E_M, T_M, I_M, F_M)$ an automaton, then $\mathcal{E}_A \otimes M$ is an automata chain. Moreover $\mathcal{E}_A \otimes M$ is a correct slicing of $Sli^{-1}(\mathcal{E}_A) \otimes M$.*

The proof is not given here. This result means that the synchronization operation on an automaton can be performed on its sliced representation without loss of information and that the result can be recovered by reconstruction.

4 Diagnosis by slices

This section proposes to use the formalism of automata chains to represent the observations and to compute, given Theorem 3, the system diagnosis. The section 5 then presents how to compute the diagnosis incrementally.

Let us first recall the definitions used in the domain of discrete-event systems diagnosis where the model of the system is traditionnally represented by an automaton.

Definition 11 (Model). *The* model of the system, *denoted* Mod, *is an automaton* $(Q^{Mod}, E^{Mod}, T^{Mod}, I^{Mod}, F^{Mod})$. I^{Mod} *is the set of possible states at* t_0. *All the states of the system may be final, then* $F^{Mod} = Q^{Mod}$. *The set of observable events of the system is denoted* $E^{Mod}_{Obs} \subseteq E^{Mod}$.

The model of the system describes its behaviour and the trajectories of Mod represent the evolutions of the system. Let us remark that we do not have any information on the final states of Mod, and so $Mod^+ = Mod$ and $Mod^\# = Mod^-$.

Let us turn to observations and diagnosis definitions. The observable events are observed by sensors and sent via communication channels to a unique supervisor. Therefore, the observations are subject to uncertainties: the clocks of the sensors are not synchronized, the transfer policy and duration are variable or partially unknown, some observations may even be lost, etc. Generally, we do not know the total order on the observations emitted by the system. Consequently, the observations are represented by an automaton, each trajectory of which represents a possible order of emission of the observations.

Definition 12 (Observations). *The* observations, *denoted* Obs_n, *is an automaton describing the observations emitted by the system during the period* $[t_0, t_n]$.

Definition 13 (Diagnosis). *The* diagnosis, *denoted* Δ_n, *is an automaton describing the possible trajectories on the model of the system compatible with the observations sent by the system during the period* $[t_0, t_n]$.

The diagnosis can be formally defined as resulting from the synchonization of the automaton representing the model (Mod), and the automaton representing the observations Obs_n on the period $[t_0, t_n]$ (see [Sampath *et al.*, 1996]). We have :

$$\Delta_n = Mod \otimes Obs_n \quad (1)$$

Due to Theorem 3, the diagnosis can be computed by computing diagnosis slices, corresponding to each observation slices, giving a diagnosis automata chain. The global diagnosis is then reconstructed from this diagnosis automata chain.

Definition 14 (Diagnosis by slices - Diagnosis slice). *Let* Mod *be the system model and* Obs_n *the observation emitted during the period* $[t_0, t_n]$. *Let* $\mathcal{E}_{Obs_n} = (Obs^1, \ldots, Obs^n)$, *be a correct slicing of* Obs_n. *The synchronization (see definition 10) of* $\mathcal{E}_{Obs_n}$ *with* Mod, *i.e* $\mathcal{E}_{Obs_n} \otimes Mod = (Obs^1 \otimes Mod, Obs^2 \otimes Mod^\#, \ldots, Obs^n \otimes Mod^-)$ *is the* diagnosis by slices *of the system.*

It can be denoted by the diagnosis automata chain $(\Delta^1, \ldots, \Delta^n)$, *where* Δ^i *is called the* i*th* diagnosis slice *of the system.*

Using Theorem 3, it can be proved that the diagnosis by slices of a system, here $\mathcal{E}_{Obs_n} \otimes Mod$, correctly represents the diagnosis computed on the global observations since the reconstruction of $\mathcal{E}_{Obs_n} \otimes Mod$ equals the global diagnosis:

Result 1. $\Delta_n = Mod \otimes Obs_n = Sli^{-1}(\mathcal{E}_{Obs_n} \otimes Mod)$

This result is illustrated by Figure 4.

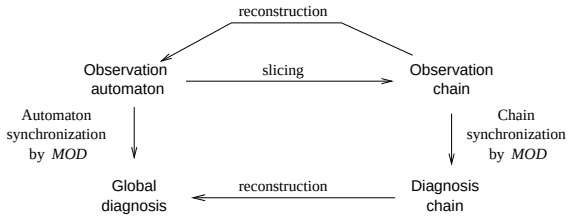

Figure 4: Illustration of Result 1

5 Incremental diagnosis

In the diagnosis by slices as presented above, the ith diagnosis slice, Δ^i, is computed independently from the others, by synchronizing the ith observation slice from the chain $\mathcal{E}_{Obs_n}$, Obs^i, with the system model $Mod^\#$. One of the interests of the observation slicing is to make the parallelized computation of each diagnosis slice possible. In this section, we focus on another approach, which elaborates an incremental diagnosis, using Δ^{i-1} to restrict the set of initial states of Mod when computing Δ^i [1]. In this section we first present a new definition of the synchronization for the incremental case and tackle the specific problem of incremental diagnosis.

Definition 15 (Restriction). *Let* $A = (Q, E, T, I, F)$ *be an automaton. The* automata restriction *of* A *by the states of* I', *denoted* $A[I']$, *is the automaton* $A' = (Q, E, T, I \cap I', F)$.

In the incremental synchronization the set of initial states of an automaton of the chain is restricted by the set of final states of its predecessor.

Definition 16 (Incremental synchronization). *The* incremental synchronization *of the automata chain* $\mathcal{E}_A = (A^1, \ldots, A^n)$ *with the automaton* M, *denoted* $\mathcal{E}_A \odot M$ *is defined as* $(A'^1, \ldots, A'^n)$ *with* $A'^i = (Q'^i, E', T'^i, I'^i, F'^i)$ *and:*

- $A'^1 = A^1 \otimes M^+$,
- $\forall i \in \{2, \ldots, n-1\},\ A'^i = (A^i \otimes M^\#)[F'^{i-1}]$ *and*
- $A'^n = (A^n \otimes M^-)[F'^{n-1}]$.

Property 4. *Let* $\mathcal{E}_A$ *be an automata chain and* M *an automaton. Then* $\mathcal{E}_A \odot M$ *is the automata chain obtained by successive I-refinements of* $\mathcal{E}_A \otimes M$.

The proof is not given.

Theorem 5. *Let* $\mathcal{E}_A$ *be an automata chain and* $M = (Q_M, E_M, T_M, I_M, F_M)$ *an automaton. We have* $Sli^{-1}(\mathcal{E}_A \odot M) = Sli^{-1}(\mathcal{E}_A \otimes M)$.

This theorem can be proved using Prop 2 and Prop 4.

Given this new definition of synchronization, a formalization of incremental diagnosis can be proposed. Provided that $\mathcal{E}_{Obs_n} = (Obs^1, \ldots, Obs^n)$ is a correct slicing of Obs_n we have: $\Delta_n = Mod \otimes Obs_n = Sli^{-1}(\mathcal{E}_{Obs_n} \odot Mod)$.

[1] We could conversely use Δ^i to restrict the set of final states of Mod when computing the diagnosis Δ^{i-1}.

We note $\forall i,\ \mathcal{E}_{Obs_i} = (Obs^1, \ldots, Obs^i)$, the automata chain of the first i observations automata. Let $i < n$, and $\mathcal{E}_{\Delta_i} = (\Delta^1, \ldots, \Delta^i)$ the automata chain resulting from the incremental synchronization of $\mathcal{E}_{Obs_i}$ with the system model Mod. We can incrementally compute $\mathcal{E}_{\Delta_{i+1}} = \mathcal{E}_{Obs_{i+1}} \odot Mod$ as follows:

Result 2. *$\mathcal{E}_{\Delta_{i+1}} = (\Delta^1, \ldots, \Delta^i, \Delta^{i+1})$ with $\Delta^{i+1} = (Obs^{i+1} \otimes Mod^{\#})[F^i_\Delta]$ where F^i_Δ is the set of final states of Δ^i.*

This result comes from the fact that $Mod^- = Mod^{\#}$ (all the states in Mod are final states). Thus it is possible to compute the automata chain that represents the diagnosis in an incremental way by synchronizing the one after the other each of the automata of the observation chain.

The automaton provided by the reconstruction operation on $\mathcal{E}_{Obs_i}$ is denoted Obs_i. Then:

Result 3. *Let $\Delta_i = Sli^{-1}(\mathcal{E}_{\Delta_i})$. Then, $\Delta_i = Obs_i \otimes Mod$.*

6 Temporal windows diagnosis

It has been proved above that, at the condition to have a correct slicing of the observation automaton, it is possible to incrementally compute the global system diagnosis by considering in sequence the slices of observations and computing for each of them its diagnosis slice. In this section, we show that this result can be instantiated to the case where the observation automaton is sliced according to time, according to temporal windows. Firstly, we extend the definition of *correct slicing* to *temporally correct slicing* by requiring temporal properties. Then, the incremental computation is demonstrated as valid on temporal windows which correctly slice the observation automaton.

Definition 17 (Correct sequence of temporal windows). *Let t_i be time instants and $[t_0, t_n]$ be the global diagnosis temporal window. A sequence of temporal windows is correct w.r.t $[t_0, t_n]$ iff it is a sequence $\mathcal{W} = (\mathcal{W}_1, \ldots, \mathcal{W}_i, \ldots, \mathcal{W}_n)$ such that $\mathcal{W}_1 = [t_0, t_1]$, $\mathcal{W}_n = [t_{n-1}, t_n]$, and $\mathcal{W}_i = [t_{i-1}, t_i]$.*

Definition 18 (Temporally correct slicing). *Let Obs_n be the observation automaton on $[t_0, t_n]$. The automata chain $\mathcal{E}_{Obs_n} = (Obs^{\mathcal{W}_1}, \ldots, Obs^{\mathcal{W}_n})$ is a temporally correct slicing of Obs_n according to $\mathcal{W} = (\mathcal{W}_1, \ldots, \mathcal{W}_i, \ldots, \mathcal{W}_n)$ iff*

- *the slicing is correct;*
- *$\mathcal{W}$ is a correct sequence of temporal windows w.r.t $[t_0, t_n]$;*
- *for each trajectory in $Obs^{\mathcal{W}_i}$, the transitions have occured during $[t_{i-1}, t_i]$ (i.e the observations labelling the transitions have been emitted by the system in $\mathcal{W}_i$).*

It can be noted that, for any $i \in \{1, \ldots, n\}$, the initial states of $Obs^{\mathcal{W}_i}$ are possible states at t_{i-1} and that the final states of $Obs^{\mathcal{W}_i}$ are possible states at t_i. Note also that, if a final state of a temporal window can be reached by two trajectories, it is required that both trajectories have occured during the temporal window, i.e the final state is a possible state in t_i whatever the trajectory used to get it.

The results of section 4 can be used in the case of temporally correct slicing. Let us denote $\forall i,\ \mathcal{E}_{Obs_{\mathcal{W}_i}} = (Obs^{\mathcal{W}_1}, \ldots, Obs^{\mathcal{W}_i})$. Let $i < n$, and $\mathcal{E}_{\Delta_{\mathcal{W}_i}} = \mathcal{E}_{Obs_{\mathcal{W}_i}} \odot Mod = (\Delta^{\mathcal{W}_1}, \ldots, \Delta^{\mathcal{W}_i})$. Then, $\mathcal{E}_{\Delta_{\mathcal{W}_{i+1}}} = \mathcal{E}_{Obs_{\mathcal{W}_{i+1}}} \odot Mod$ can be computed as follows:

Result 4. *$\mathcal{E}_{\Delta_{\mathcal{W}_{i+1}}} = (\Delta^{\mathcal{W}_1}, \ldots, \Delta^{\mathcal{W}_i}, \Delta^{\mathcal{W}_{i+1}})$ with $\Delta^{\mathcal{W}_{i+1}} = (Obs^{i+1} \otimes Mod^{\#})[F^{\mathcal{W}_i}_\Delta]$ where $F^{\mathcal{W}_i}_\Delta$ is the set of final states of $\Delta^{\mathcal{W}_i}$.*

Let $Obs_{\mathcal{W}_i}$, the automaton provided by the reconstruction operation on $\mathcal{E}_{Obs_{\mathcal{W}_i}}$. $Obs_{\mathcal{W}_i}$ represents the observations emitted on the period $[t_0, t_i]$.

Result 5. *Let $\Delta_{\mathcal{W}_i} = Sli^{-1}(\mathcal{E}_{\Delta_{\mathcal{W}_i}})$. Then, $\Delta_{\mathcal{W}_i} = Obs_{\mathcal{W}_i} \otimes Mod$ is the diagnosis of the period $[t_0, t_i]$.*

The incremental computation of diagnosis from temporal windows seems promising firstly because the diagnosis gives then the possible states of the system at time t_i w.r.t the (possibly uncertain) observations gathered at time t_i. Another good reason appears when turning into an on-line diagnosis context. The observation automata chain has now to be built on-line, i.e without knowing by advance the whole set of observations gathered on the global diagnosis window. This point is not examined in this paper but it can be shown that taking profit of temporal information, it is easier, on-line, to build temporally correct slicing than only correct slicing. Observations, which should be considered as possible in the general case, can be discarded as not satisfying the temporal constraints collected on the system behaviour (as delays between observations emission and reception; communication channels politics...).

7 Conclusion

In this paper, we formalized the incremental computation of diagnosis for discrete-event systems. We introduced and defined the concept of automata chain that enables us to handle slices of observations and slices of diagnosis rather than global observations and global diagnosis. We proved that the diagnosis can be computed first in a modular way and then, more efficiently, in an incremental way, both methods using the automata chain. We then presented how the results can be extended to the case where observations are sliced according to temporal windows.

In the diagnostic literature the notion of incremental diagnosis is relatively new. It can be explained by the fact that, in most cases, observations are supposed to be totally ordered, received without delays, and without any loss. In these cases, the problem of slicing the observations does not exist. In [Baroni *et al.*, 1999] however, the authors examine the case where observations are uncertain and represented by partially ordered graphs. In the case of decentralized systems, Pencolé *et al.* [2001] consider the incremental diagnosis computation applied to the on-line diagnosis for telecommunication networks. The property of *safe window* is defined and algorithms are given in the case where the temporal windows satisfy this property. Extensions to more complicated cases are proposed. Compared to this work, our proposal is more general and give a formal view of the problem which allows to

better situate the algorithmic approach proposed in [Pencolé *et al.*, 2001]. In [Cordier and Largouët, 2001] an incremental approach of diagnosis is considered from a model-checking point of view.

Our study exhibits the (non trivial) correctness properties that the observation slicing, in an automata chain, has to satisfy in order to guarantee the completeness of the diagnosis computation. This first step is then essential before considering the incrementality of on-line diagnosis computation.

The next step will consider the building of the observations automata chain in the context of off-line and then on-line diagnosis. The case of on-line diagnosis is particularly interesting since the goal is to dynamically build an automata chain without having all the observations. As seen at the end of section 6, this task can take profit of temporal information known on the system, even if, for complexity reasons, these temporal constraints are not encoded in the system model. Another interesting point is to use the concept of automata chains for the diagnosis of reconfigurable systems.

Appendix

Proof of Theorem 1. Let $A = (Q, E, T, I, F)$ be an automaton and $\mathcal{E}_A = (A^1, \ldots, A^n)$ an automata chain with $A^i = (Q^i, E^i, T^i, I^i, F^i)$ so that $\mathcal{E}_A$ is a correct slicing of A. Let $A_R = (Q_R, E_R, T_R, I_R, F_R)$ be the reconstruction of $\mathcal{E}_A$. We have to prove that the set of trajectories of A (which is the same as the set of trajectories of $\mathcal{E}_A$) equals the set of trajectories of A_R.

Let $\mathcal{E}_{A_{1,2}} = (A^1, A^2)$. $\mathcal{E}_{A_{1,2}}$ is an automata chain. Let $A_{1,2}$ be the reconstruction of $\mathcal{E}_{A_{1,2}}$. Let us consider a transition (q, l, q') of $A_{1,2}$.

Remark 1: $\{q, q'\} \subseteq Q^1$ or $\{q, q'\} \subseteq Q^2$ (because $(q, l, q') \in T_{1,2} = T^1 \cup T^2$). Consequently, if a state does not belong to Q^2 (resp. Q^1), it belongs to Q^1 (resp. Q^2) and its predecessor too. Moreover, if a path on $A_{1,2}$ goes from a state from Q^1 to a state from Q^2, there exists at least one state on the path belonging to $Q^1 \cap Q^2$.

Remark 2: $\forall j \in \{1, 2\},\ \{q, q'\} \subseteq Q^j \Rightarrow (q, l, q') \in T^j$.

i) $\forall traj = ((q_0, \ldots, q_m), (l_1, \ldots, l_m))$, trajectory of $\mathcal{E}_{A_{1,2}}$, then $traj$ is also a trajectory of $A_{1,2}$ since (by definition) any transition of A^1 or A^2 is a transition of $A_{1,2}$, $q_0 \in I^1$ and $q_m \in F^2$.

ii) $\forall traj = ((q_0, \ldots, q_m), (l_1, \ldots, l_m))$, trajectory of $A_{1,2}$. Let k be the smallest value in $\{0, \ldots, m\}$ so that $q_k \in Q^1 \cap Q^2$ (k exists due to Remark 1).

$\forall i \leq k,\ q_i \in Q^1$, so $traj^1 = ((q_0, \ldots, q_k), (l_1, \ldots, l_k))$ is a trajectory of A^1 (cf. Remark 2).

Let us now prove that $\forall i > k,\ q_i \in Q^2$. Let us suppose it exists j, the smallest value so that $j > k$ and $q_j \notin Q^2$. $q_j \in Q^1$ and, due to Remark 1, $q_{j-1} \in Q^1 \cap Q^2$. For the same reason as for k, $\exists l$ the smallest value so that $l > j$ and $q_l \in Q^1 \cap Q^2$. The path $p = ((q_{j-1}, \ldots, q_l), (l_j, \ldots, l_l))$ is a path of A^1. But, since q_{j-1} and q_l are both belonging to $Q^1 \cap Q^2$, p is also a path of A^2. It implies that q_j is a state of Q^2, which is in contradiction with the existence of j. So, $\forall i > k,\ q_i \in Q^2$. And $traj^2 = ((q_k, \ldots, q_m), (l_{k+1}, \ldots, l_m))$ is a trajectory of A^2. $traj$ is built by reconstruction of $traj^1$ and $traj^2$. It is then a trajectory of $\mathcal{E}_{A_{1,2}}$.

Since the trajectories of $A_{1,2}$ and (A^1, A^2) are equal, $\mathcal{E}_A$ and $(A_{1,2}, A^3, \ldots, A^n)$ have the same trajectories. We define recursively $\forall i > 2,\ A_{1,i}$ the reconstruction of $(A_{1,i-1}, A^i)$. Then, we prove recursively that $\mathcal{E}_A$ has the same trajectories as $(A_{1,i}, A^{i+1}, \ldots, A^n)$ in particular $(A_{1,n}) = (A_R)$. So, $\mathcal{E}_A$ and A_R have the same trajectories. As $\mathcal{E}_A$ is a correct slicing of A, $A = A_R$. □

References

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

[Barral *et al.*, 2000] C. Barral, S. McIlraith, and T.C. Son. Formulating diagnostic problem solving using an action language with narratives and sensing. In *International Conference on Knowledge Representation and Reasoning (KR'2000)*, pages 311–322, 2000.

[Cassandras and Lafortune, 1999] C. G. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems*. Kluwer Academic Publishers, 1999.

[Console *et al.*, 2000] L. Console, C. Picardi, and M. Ribaudo. Diagnosis and diagnosability analysis using PEPA. In *14th European Conference on Artificial Intelligence (ECAI-2000)*, pages 131–135, Berlin, Allemagne, 2000.

[Cordier and Largouët, 2001] M.-O. Cordier and C. Largouët. Using model-checking techniques for diagnosing discrete-event systems. In *12th International Workshop on Principles of Diagnosis (DX'01)*, pages 39–46, 2001.

[Cordier and Thiébaux, 1994] M.-O. Cordier and S. Thiébaux. Event-based diagnosis for evolutive systems. In *5th International Workshop on Principles of Diagnosis (DX-94)*, pages 64–69, 1994.

[Lunze, 1999] J. Lunze. Discrete-event modelling and diagnosis of quantized dynamical systems. In *10th International Workshop on Principles of Diagnosis (DX-99)*, pages 147–154, Loch Awe, Écosse, Royaume Uni, 1999.

[Pencolé and Cordier, 2005] Y. Pencolé and M.-O. Cordier. A formal framework for the decentralised diagnosis of large scale discrete event systems and its application to telecommunication networks. *Artificial Intelligence Journal*, 164(1-2):121–170, 2005.

[Pencolé *et al.*, 2001] Y. Pencolé, M.-O. Cordier, and L. Rozé. Incremental decentralized diagnosis approach for the supervision of a telecommunication network. In *12th International Workshop on Principles of Diagnosis (DX'01)*, pages 151–158, 2001.

[Sampath *et al.*, 1996] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D.C. Teneketzis. Failure diagnosis using discrete-event models. In *IEEE Transactions on Control Systems Technology (CST-96)*, pages 105–124, 1996.