Multiresolution aspects of the diagnosis problem
Internship report

Alexandre Blanché
Parcours Recherche et Innovation, L3 Rennes 1 University and ENS Rennes
Under the supervision of Éric Fabre (Inria)
Inria Rennes - Team SUMO

***Abstract*—The problems of state estimation and fault diagnosis are essential in the monitoring of discrete-event systems. These two problems are studied here in the simple setting of finite automata. We first explain the problems, the relation between them and the concept of diagnosability, before defining the "ambiguity depth" and proving a simple result which links these two concepts. Then we study some multiresolution aspects of the diagnosis problem. Here, the goal is to reduce the size of the system on which the resolution of the diagnosis will be achieved. After having solved the diagnosis problem on a smaller system, we try to efficiently find the solution of the diagnosis on the actual system. In the last section, we try a multiresolution approach of the diagnosis by merging states of the automaton. We prove that a certain type of diagnosability is conserved by the merger of all the faulty states of the system.**

Introduction

In the monitoring of discrete-event systems, some central problems we may encounter are that of *state estimation* and *diagnosis* [1]. Given a finite labeled transition system, here represented as a finite automaton whose labels are called *events*, and are either *observable* or *unobservable* [2, 4], we partition the set of states into *safe* and *faulty* states. The problem of state estimation is to decide, given the observable part of an input word, what states have been reached by the system, while the problem of diagnosis is to decide if a faulty state has been reached or not. Since the automaton is nondeterministic, this decision can be ambiguous, as there may exist a run of the automaton that reaches a faulty state, and another run that reaches a safe state with the same observable events. Therefore, we define the property of *diagnosability*, which expresses that there exists a uniform bound on the events we have to observe in order to diagnose a fault after its occurrence. The first contribution of this report is the introduction of a new quantity, the *ambiguity depth*, which prevents arbitrarily long ambiguous paths to be diagnosed. We then examine a multiresolution problematic : under which conditions can we merge some states of the system without losing the diagnosability? We redefine diagnosability for systems with merged states, and we prove that this specific diagnosability is conserved by the merger of the whole faulty component.

I. The problems of state estimation and diagnosis

Automaton: This section will formalize the problems of state estimation and diagnosis, in the simple framework of finite automata. We follow the setting of [3]. All the results of this section, until the reachability of a faulty state, are taken from [1]. A *system* is a nondeterministic automaton $\mathcal{A} = (S, \Sigma, I, \delta)$, where S is a finite set of states, Σ is a finite alphabet, partitioned between the set of *observable* labels Σ_o and the *unobservable* labels Σ_u. $I \subseteq S$ is the set of possible initial states, and $\delta : S \times \Sigma \to 2^S$, the transition function of the automaton, which extends naturally into $\delta : S \times \Sigma^* \to 2^S$ by iteration on the second variable. In a first framework, the state set S is partitioned in two components: the *safe* states S_s and the *faulty* states S_f: $S = S_s \uplus S_f$. The faults are also assumed to be permanent: $\forall s \in S_f, \delta(s, \Sigma) \subseteq S_f$.

Since the unobservable events are unusable, the system that will be used in the paper, as in [1], is the ε-reduction $\mathcal{A}'$ of the current system $\mathcal{A}$, with all unobservable letters considered as ε. As the ε-reduction is chosen "to the left" of the observable events, such an automaton is defined by $\mathcal{A}' = (S, \Sigma_o, I, \delta')$, where $\delta' : S \times \Sigma_o \to 2^S$ is defined by $\delta'(s, \alpha) = \delta(s, \Sigma_u^* \alpha) = \bigcup_{\omega \in \Sigma_u^* \alpha} \delta(s, \omega)$. The study of the ε-reduced system is equivalent to the study of the actual system, and the ε-reduction can easily be achieved in linear time in the number of transitions of the automaton.

If $T = \{(s, \alpha, s') \in S \times \Sigma \times S | s' \in \delta(s, \alpha)\}$ is the set of the transitions, and if we adopt $s^-((s, \alpha, s')) = s, s^+((s, \alpha, s')) = s'$ and $\sigma((s, \alpha, s')) = \alpha$ as notations, we define a *path* in $\mathcal{A}$ as a sequence of transitions $\pi = t_1 t_2 \dots t_N \in T^N, N \in \mathbb{N}^*$, such that $\forall i \in [\![1, N-1]\!], s^+(t_i) = s^-(t_{i+1})$ and that the last transition

is observable. By definition of the ε-reduction to the left, the paths of $\mathcal{A}$ end in the same states that the paths in $\mathcal{A}'$. For this reason we only consider paths of $\mathcal{A}'$, we assume that $\Sigma = \Sigma_o$ and we only work with $\mathcal{A}'$ until the end of the report. Some notations we adopt are $|t_1 t_2 \dots t_N| = N$ for the length of a path and $\sigma(t_1 t_2 \dots t_N) = \sigma(t_1)\sigma(t_2)\dots\sigma(t_N) \in \Sigma_o^*$ for its signature. A *run* π of $\mathcal{A}'$ is a path which starts from I: $s^-(\pi) \in I$.

The system $\mathcal{A}$ must be Σ_o-*live*, which means that the system can meet an observable transition from every state. In other words, $\mathcal{A}$ is Σ_o-live if and only if $\mathcal{A}'$ is live, which means $\forall s \in S, \delta'(s, \Sigma_o) \neq \emptyset$.

State estimation: Given a system $\mathcal{A}$ and an input sequence of observations $\omega \in \Sigma_o^*$, the problem of *state estimation* is to compute the possible current states of $\mathcal{A}$ after having observed ω. In other words, we have to build a function $obs : \Sigma_o^* \to 2^S$, such that $\forall \omega \in \Sigma_o^*$, $obs(\omega) = \{s^+(\pi) \mid \pi \text{ is a run of } \mathcal{A}' \text{ and } \sigma(\pi) = \omega\}$.
A natural way to solve this problem for a word $\omega \in \Sigma_o^*$, is to compute the successive sets of possible states that have encountered each prefix of ω: if $\omega = \alpha_1\alpha_2\dots\alpha_n$, we compute the sets

$$X_k(\omega) = s^+(\sigma^{-1}(\{\alpha_1 \dots \alpha_k\})) \tag{1}$$

for $k \in [\![1, n]\!]$. Hence, the last set $X_n(\omega)$ answers the question, by definition of obs, $obs(\omega) = X_n(\omega)$. Such a sequence of sets is interesting because it can easily be computed recursively:

$$\begin{aligned} &X_{k+1}(\omega) \\ &= \delta'(X_k(\omega), \alpha_{k+1}) \\ &= \{s^+(t) | t \in T_o, \sigma(t) = \alpha_{k+1} \wedge s^-(t) \in X_k(\omega)\}. \end{aligned}$$

Another way to solve the problem of state estimation is to build an *observer* $Obs(\mathcal{A})$ of $\mathcal{A}$, defined by $Obs(\mathcal{A}) = (Q, \Sigma_o, q_0, \overline{\delta})$, a deterministic automaton, still over the alphabet Σ_o, where $Q \subseteq 2^S$, $q_0 = I$ is the unique initial state of $Obs(\mathcal{A})$, and $\overline{\delta} : Q \times \Sigma_o \to Q$ its transition function. We define it in the canonical way as $Obs(\mathcal{A}) = Det(\mathcal{A}') = Det(Red(\mathcal{A}))$, with Det the determinization process, using the classical subset construction. Such an observer answers the question, by definition of the subset construction: for all $\omega \in \Sigma_o^*$, $obs(\omega)$ is the unique state of $Obs(\mathcal{A})$ reached with ω for input, since this unique state represents the set of all possible states of $\mathcal{A}'$ reached after having read ω as input. However, the main issue with such a construction is the exponential number of states of the observer.

Diagnosis: Another problem, closely linked to the previous one, is the *diagnosis* problem: at the end of a given run, we have to decide if a fault has occurred or not. As explained before, such a decision can be ambiguous, because there may exist two runs π_1 and π_2, with π_1 reaching a safe state and π_2 reaching a fault before its end, both runs having the same observable signature ($\sigma(\pi_1) = \sigma(\pi_2)$) and hence are indistinguishable. More precisely, we want to compute a function $diag : \Sigma_o^* \to \{s, f, a\}$, such that:

$$\forall \omega \in \Sigma_o^*, \quad diag(\omega) = \begin{cases} s & \text{if } obs(\omega) \subseteq S_s \\ f & \text{if } obs(\omega) \subseteq S_f \\ a & \text{otherwise} \end{cases}$$

This can be done either by a recursive evaluation as previously established with the state estimation problem, or by simply reusing the observer: associating it with a label function $\phi : Q \to \{s, f, a\}$ which "tests" the inclusion of $q \in Q$ in S_s or S_f suffices to answer the question. Such a pair $(Obs(\mathcal{A}), \phi)$ is called a *diagnoser* of $\mathcal{A}$. It will often be called "the" diagnoser, because only the canonical construction is considered here. Obviously, like the observer, the diagnoser has a number of states that is exponential in the number of states of $\mathcal{A}$.

The diagnosis function introduced in this paragraph may enable some arbitrarily long sequences of a, even if a fault has occurred, which is not desirable to solve our problem. We wish to be able to detect a fault every time it occurs, after a finite number of steps. This property, called *diagnosability*, is formalized as follows:

Definition 1 (Diagnosability). *A system $\mathcal{A}$ is diagnosable iff $\exists N \in \mathbb{N}$, such that $\forall \pi$ run : $s^+(\pi) \in S_f, \forall \pi'$ run : $s^-(\pi') = s^+(\pi), [|\pi'| > N \Rightarrow diag(\sigma(\pi\pi')) = f]$*

Or in other words, there exists a uniform bound N such that at most N observations after the occurrence of a fault, this fault is surely diagnosed.

We will now characterize the diagnosability with a natural object, the so-called *twin-machine* [5] $\mathcal{T}_\mathcal{A}$, defined by $\mathcal{T}_\mathcal{A} := \mathcal{A}' \times \mathcal{A}'_s$, where $\mathcal{A}'_s$ is the restriction of $\mathcal{A}'$ to its safe states, and $\times$ is the classical synchronous product of automata. We define an *ambiguous path* of $\mathcal{T}_\mathcal{A}$ as a couple $(\pi, \pi') = (t_1 \dots t_n, t'_1 \dots t'_n)$ of paths of $\mathcal{A}'$ such that $\forall i \in [\![1, n]\!], s^+(t_i) \in S_f$, and $\sigma(\pi) = \sigma(\pi')$. The *ambiguous cycles* of $\mathcal{T}_\mathcal{A}$ are also defined in the same way. We can easily prove the following result:

Proposition 1. *$\mathcal{A}$ is diagnosable if and only if there is no reachable ambiguous cycle in $\mathcal{T}_\mathcal{A}$.*

This property can be considered as a characterization of the diagnosability, and it can be checked in polynomial time.

Reachability of a faulty state: Here comes the beginning of my personal contribution, which will last until the end of this report. Given a system $\mathcal{A}$, even if it is diagnosable, it is still possible to have an arbitrarily long sequence of diagnosis a for a given non-faulty path. Figure 1 below gives an example: the system is diagnosable, but an arbitrarily long sequence $0 \xrightarrow{a} 1 \xrightarrow{b} 0 \xrightarrow{a} 1 \xrightarrow{b} 0 \dots$ leads to an always ambiguous diagnosis, even if no fault occurs.

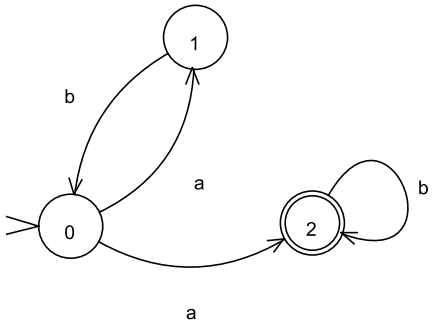

Figure 1. The system $\mathcal{A}'$: 0 and 1 are safe, 2 is faulty

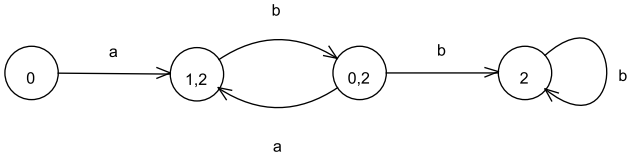

Figure 2. The diagnoser of $\mathcal{A}$, with its ambiguous cycle between 1,2 and 0,2. The state 0 is safe, 2 is faulty and both 1,2 and 0,2 are ambiguous.

We now give a result that may not seem obvious, given the previous counter-example:

Proposition 2. *If a system $\mathcal{A}$ is diagnosable and at least one faulty state of $\mathcal{A}$ is reachable, then at least one faulty state of the diagnoser of $\mathcal{A}$ is reachable.*

This is not immediately clear, considering the example of infinite ambiguous but non-faulty path of the previous counter-example. However, it can be proved easily with the definition of diagnosability:

Proof. Let $\mathcal{A}$ be a diagnosable system with at least one reachable faulty state, and $\mathcal{D}$ its diagnoser. We will prove by contradiction that at least one faulty state of $\mathcal{D}$ is reachable: $\exists \omega \in \Sigma_o^*$, such that $obs(\omega) \subseteq S_f$. So let us assume that no faulty state of $\mathcal{D}$ is reachable.

$\mathcal{A}$ is diagnosable, so there exists $N \in \mathbb{N}$ such that for all runs π of $\mathcal{A}'$, $s^+(\pi) \in S_f, \forall \pi' : s^-(\pi') = s^+(\pi), [|\pi'| > N \Rightarrow diag(\sigma(\pi\pi')) = f]$. The consequent $diag(\sigma(\pi\pi')) = f$ is impossible, because for any $\omega \in \Sigma_o^*$, $diag(\omega) = f$ if and only if $obs(\omega) \subseteq S_f$, which is contradictory with the hypothesis that no faulty state of $\mathcal{D}$ is reachable.

Hence, if π is a run that ends in S_f, then for all runs π': $[s^+(\pi) = s^-(\pi') \Rightarrow |\pi'| \leq N]$. We now use the property that $\mathcal{A}'$ is Σ_o-live: from any state we can reach a (visible) transition. Hence we can build an arbitrarily long path π'' which starts from the state $s^+(\pi)$ and then takes random transitions. Such a path, at least longer that $N+1$, contradicts the property $s^+(\pi) = s^-(\pi') \Rightarrow |\pi'| \leq N$. The result follows by contradiction. □

Notice that the diagnosability is a property of the system and not of the language. There exist two systems that recognize the same safe and faulty languages, but only one of these systems is diagnosable. For instance, let us consider again the system in figure 1, and call it $\mathcal{A}_0$. The system $\mathcal{A}_1$ in figure 3 recognize the same safe language, $\mathcal{L}_S = (ab)^* + (ab)^*a$ and the same faulty language, $\mathcal{L}_F = (ab)^*ab^*$, but $\mathcal{A}_0$ is diagnosable and not $\mathcal{A}_1$.

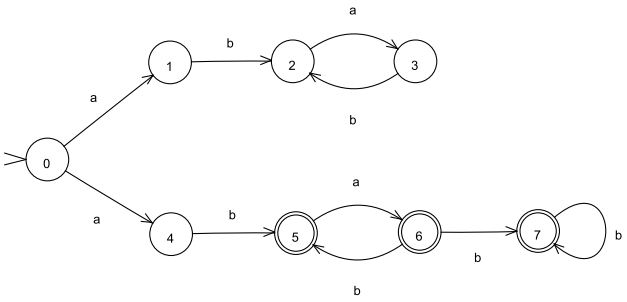

Figure 3. The system $\mathcal{A}_1$, not diagnosable because of runs with signature $ab(ab)^*$ that can be faulty and match a safe run as long as we want.

II. The ambiguity depth

Ambiguous runs: This section introduces a new useful quantity, the *ambiguity depth*, in order to give a simple method to bound the length of the ambiguous chains in the diagnoser. First, we define a binary relation between faulty and safe runs.

If π is a faulty run of $\mathcal{A}'$, i.e. if it hits a faulty state somewhere before its end, and if π' is a safe run of $\mathcal{A}'$, i.e. if all of its transitions reach only safe states, then $\pi \sim \pi'$ if and only if $\sigma(\pi) = \sigma(\pi')$.

We are now able to define a function $l : \{\pi \mid \pi \text{ run of } \mathcal{A}'\} \to \mathbb{N}$ which counts the number of ambiguous steps in a given run. More precisely, if the run is faulty and equivalent (according to the relation $\sim$) to a run π', l gives the number of faulty transitions of π, that will lead to ambiguous states of the diagnoser.

First, we define the function $\widetilde{l}$. Let $\pi = t_1 t_2 \dots t_n$ be a faulty run of $\mathcal{A}'$:

- If there exists π' safe, $\pi \sim \pi'$, then $\widetilde{l}(\pi) := Card\{i \mid s^+(t_i) \in S_f\}$.
- Else, $\widetilde{l}(\pi) = 0$.

We now define the function l:

$$\forall \pi \text{ faulty run of } \mathcal{A}',\ l(\pi) = \max_{\overline{\pi} \text{ prefix of } \pi} \widetilde{l}(\overline{\pi})$$

The function l is closely linked with the twin-machine: if $\pi \sim \pi'$, then (π, π') is a run of $\mathcal{T}_\mathcal{A}$, and $\widetilde{l}(\pi)$ is the number of transitions of this run that reach a state belonging to $S_f \times S_s$. In other words, $l(\pi)$ is the length of the ambiguous part of (π, π') in $\mathcal{T}_\mathcal{A}$. The link with the

diagnosability is now obvious, with the characterization as the non-existence of ambiguous cycle in $\mathcal{T}_\mathcal{A}$: $\mathcal{A}$ is diagnosable if and only if its function l is upper-bounded.

Definition 2 (Ambiguity depth)**.** *The ambiguity depth of a system $\mathcal{A}$ is defined by*

$$\mathcal{D}_\mathcal{A} := \max \{l(\pi) \mid \pi \text{ run of } \mathcal{A}'\}$$

According to the fact that $\mathcal{D}_\mathcal{A}$ is the length of the longest ambiguous path of $\mathcal{T}_\mathcal{A}$, we can give a new characterization of the diagnosability with the ambiguity depth:

Proposition 3. *A system $\mathcal{A}$ is diagnosable if and only if $\mathcal{D}_\mathcal{A} < +\infty$.*

Indeed, $\mathcal{D}_\mathcal{A} < +\infty$ means that l is upper-bounded, or that there is a bound on the length of the ambiguous paths of $\mathcal{T}_\mathcal{A}$. Hence there there is no ambiguous cycle in $\mathcal{T}_\mathcal{A}$ if and only if $\mathcal{D}_\mathcal{A} < +\infty$.

Let us now examine a simple example. The figure below shows the twin-machine of the automaton in figure 1: the longest ambiguous path has two transitions that lead to states in $S_f \times S_s$, $(0,0) \xrightarrow{a} (2,1) \xrightarrow{b} (2,0)$. Hence we deduce that $\mathcal{D}_\mathcal{A} = 2$.

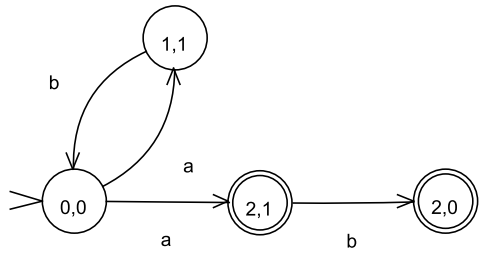

Figure 4. Twin-machine of the system given in figure 1.

State estimation: We have introduced the notations of the state estimation in equation 1. First we can remark that this set of states $X_k(\omega)$ is precisely the state of the diagnoser reached by the unique run which has $\alpha_1 \dots \alpha_k$ as signature. Before stating the main theorem of this section, we introduce new notations. For all $\omega = \alpha_1\alpha_2 \dots \alpha_n \in \Sigma_o^*$ and for all $k \leq n$:
$\widehat{X}_{k|n}(\omega) := \{s^+(t_k) \mid \pi = t_1 t_2 \dots t_n \in T_o^n, \sigma(\pi) = \omega\}$

It is different from X_k, but we have these two properties:

$$\widehat{X}_{n|n}(\omega) = X_n(\omega) \tag{2}$$

$$\widehat{X}_{k|n}(\omega) \subseteq X_k(\omega) \tag{3}$$

- Proposition 2 is straightforward:
  $\widehat{X}_{n|n}(\omega) = \{s^+(t_n) = s^+(\pi) \mid \pi = t_1 \dots t_n, \sigma(\pi) = \omega\} = s^+(\sigma^{-1}(\omega)) = X_n(\omega)$.
- For proposition 3, we consider $k \leq n$ and $s \in \widehat{X}_{k|n}(\omega)$. By definition, there exists a run $\pi = t_1 \dots t_n$ of $\mathcal{A}'$ which reaches s at step k and has ω as signature. The prefix $t_1 \dots t_k$ of π has $\alpha_1 \dots \alpha_k$ for signature and reaches s at step k as well. Hence, $s \in X_k(\omega)$, and $\widehat{X}_{k|n}(\omega) \subseteq X_k(\omega)$.
- The other inclusion does not hold in general. In the example in figure 1, $X_1(abb) = \{1,2\}$ (the state of the diagnoser reached by a), but the only run which has abb as signature is the run $0 \xrightarrow{a} 2 \xrightarrow{b} 2 \xrightarrow{b} 2$. Hence $\widehat{X}_{1|3}(abb) = \{2\} \neq X_1(abb)$.

We can now state our main theorem, which will help disambiguate the paths of the diagnoser by bounding the length of the ambiguity:

Theorem 1. *For $\mathcal{A}$ such that $\mathcal{D}_\mathcal{A} < +\infty$ (i.e. $\mathcal{A}$ is diagnosable) and for $\omega \in \Sigma_o^*$ with $|\omega| = n$,*

$$diag(\omega) = a \text{ implies } \widehat{X}_{n-\mathcal{D}_\mathcal{A}|n}(\omega) \subseteq S_s.$$

Or in other words, if a word is diagnosed as ambiguous at step n, then it was safe $\mathcal{D}_\mathcal{A}$ steps earlier, given the whole sequence of observations ω. So if a word sees its diagnosis becoming ambiguous at step n, we just have to examine the diagnosis $\mathcal{D}_\mathcal{A}$ steps further: if it is safe or still ambiguous, we can solve the ambiguity and declaring the word as safe at the current step n.

Proof. We will prove this result by contradiction. Let ω be a word of length n in Σ_o^*. Let us assume by contradiction that $diag(\omega) = a$ and $\widehat{X}_{n-\mathcal{D}_\mathcal{A}|n}(\omega) \not\subseteq S_s$, i.e. $\exists \pi = t_1 \dots t_n$ run of $\mathcal{A}'$ such that $\sigma(\pi) = \omega$ and $s^+(t_{n-\mathcal{D}_\mathcal{A}}) \in S_f$. Since $diag(\omega) \neq f$, there exists at least one safe run π' of $\mathcal{A}'$ that matches the word ω. Hence $\pi \sim \pi'$, and (π, π') is a run of the twin-machine $\mathcal{T}_\mathcal{A}$. Since faults are permanent, we have $l(\pi) \geq n - (n - \mathcal{D}_\mathcal{A} - 1) = \mathcal{D}_\mathcal{A} + 1$, by definition of l. $l(\pi) > \mathcal{D}_\mathcal{A} = \max_\pi l(\pi)$: by contradiction, there does not exist such a run π. We deduce that $\widehat{X}_{n-\mathcal{D}_\mathcal{A}|n}(\omega) \subseteq S_s$. □

This result is useful to bound the ambiguity by $\mathcal{D}_\mathcal{A}$. Given a word $\omega = \alpha_1\alpha_2 \dots$, and a step n of the diagnosis of ω, such that $|\omega| \geq n + \mathcal{D}_\mathcal{A}$, we assume that $diag(\alpha_1 \dots \alpha_n) = a$:

- if $diag(\alpha_1 \dots \alpha_{n+\mathcal{D}_\mathcal{A}}) = s$: then for any run π of $\mathcal{A}'$ with $\sigma(\pi) = \alpha_1 \dots \alpha_{n+\mathcal{D}_\mathcal{A}}$, we have $s^+(\pi) \in S_s$, and hence for any π' with $\sigma(\pi') = \alpha_1 \dots \alpha_n$, since $s^+(\pi') \in S_s$,
- else if $diag(\alpha_1 \dots \alpha_{n+\mathcal{D}_\mathcal{A}}) = f$: then for any run π with $\sigma(\pi) = \alpha_1 \dots \alpha_{n+\mathcal{D}_\mathcal{A}}$, a fault occurs before step $n + \mathcal{D}_\mathcal{A}$,
- else, $diag(\alpha_1 \dots \alpha_{n+\mathcal{D}_\mathcal{A}}) = a$, and by the previous theorem, we can conclude that for any π with $\sigma(\pi) = \alpha_1 \dots \alpha_n$, we have $s^+(\pi) \in S_s$.

Application: We consider again the example in figure 1 above. We saw in figure 4 that the system $\mathcal{A}$ was diagnosable, with $\mathcal{D}_\mathcal{A} = 2$. However, we saw on the diagnoser of $\mathcal{A}$, in figure 2, that there exist arbitrarily long sequences with an ambiguous diagnosis, for words

in $(ab)^*$. We consider here the word $\omega = ababbb$. The first six observations are diagnosed ambiguous, but the seventh is faulty:

$$\begin{array}{ccccccccccccc} 0 & \xrightarrow{a} & \binom{1}{2} & \xrightarrow{b} & \binom{0}{2} & \xrightarrow{a} & \binom{1}{2} & \xrightarrow{b} & \binom{0}{2} & \xrightarrow{a} & \binom{1}{2} & \xrightarrow{b} & \binom{0}{2} & \xrightarrow{b} & 2 \\ \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ S & \rightarrow & A & \rightarrow & A & \rightarrow & A & \rightarrow & A & \rightarrow & A & \rightarrow & A & \rightarrow & F \end{array}$$

We have got six ambiguous observations, but with the help of the theorem, we can bound it at two by synchronizing it with the diagnosis two steps further (because $\mathcal{D}_{\mathcal{A}} = 2$) :

$$\begin{array}{ccccccccccccc} \cdots & \rightarrow & A & \rightarrow & A & \rightarrow & A & \rightarrow & A & \rightarrow & F & & & & \\ & & \Downarrow & & \Downarrow & & \Downarrow & & \Downarrow & & & & & & \\ 0 & \xrightarrow{a} & \binom{1}{2} & \xrightarrow{b} & \binom{0}{2} & \xrightarrow{a} & \binom{1}{2} & \xrightarrow{b} & \binom{0}{2} & \xrightarrow{a} & \binom{1}{2} & \xrightarrow{b} & \binom{0}{2} & \xrightarrow{b} & 2 \\ \downarrow & & \Downarrow & & \Downarrow & & \Downarrow & & \Downarrow & & \downarrow & & \downarrow & & \downarrow \\ S & \rightarrow & S & \rightarrow & S & \rightarrow & S & \rightarrow & S & \rightarrow & A & \rightarrow & A & \rightarrow & F \end{array}$$

III. Merging states

Framework: Here comes the central topic of this report, the multiresolution aspects of the diagnosis problem. This problem can be explained like this: given an instance of a diagnosis problem, we assume we can solve this problem for an abstraction of this instance. Is it possible to deduce from it, in an efficient way, the diagnosis of the actual instance? The type of abstraction considered here is a reduction of the number of states of the automaton. Given a system $\mathcal{A}$, we merge two of its states. The goal is to reduce the number of states of the automaton without changing the diagnosis of the words received as input by $\mathcal{A}$. Such a family of words can be formalized: we denote $\mathcal{L}(\mathcal{A})$ the *language of* $\mathcal{A}$ as $\mathcal{L}(\mathcal{A}) := \{\omega \in \Sigma^* \mid \exists \pi \text{ run of } \mathcal{A}, \sigma(\pi) = \omega\}$. We once again consider the *safe language*, as the words that are matched by at least one safe run, the *faulty language* defined in the same way for faulty runs, and the *ambiguous language* as the intersection of the safe language and the faulty language. The goal of the study is to merge two states of the system: the language will change, but not necessarily the diagnosis of the words from the language of $\mathcal{A}$. More precisely, we will define the *merged system* $\overline{\mathcal{A}}$ such that $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\overline{\mathcal{A}})$.

Given a system $\mathcal{A} = (S, \Sigma, I, \delta)$, nondeterministic and such that all letters of Σ are assumed to be observable. Let s_1 and s_2 be two different states of S. The merged system is defined as:

Definition 3. *If we merge s_1 and s_2 in $\mathcal{A}$, the merged system is defined by $\overline{\mathcal{A}} = (\overline{S}, \Sigma, \overline{I}, \overline{\delta})$, with:*

- $\overline{S} = S \setminus \{s_1, s_2\} \uplus \{s_{12}\}$, *where s_{12} is a new state of $\overline{\mathcal{A}}$,*
- $\overline{I} = \begin{cases} I \text{ if } \{s_1, s_2\} \cap I = \emptyset \\ I \setminus \{s_1, s_2\} \uplus \{s_{12}\} \text{ otherwise} \end{cases}$
- $\overline{\delta} : \overline{S} \times \Sigma \to 2^{\overline{S}}$ *such that* $\forall \alpha \in \Sigma$, $\overline{\delta}(s_{12}, \alpha) = \delta(s_1, \alpha) \cup \delta(s_2, \alpha)$,
  and $\forall s \in S \setminus \{s_1, s_2\}$,

$$\overline{\delta}(s, \alpha) = \begin{cases} \delta(s, \alpha) \text{ if } \{s_1, s_2\} \cap \delta(s, \alpha) = \emptyset \\ \delta(s, \alpha) \setminus \{s_1, s_2\} \uplus \{s_{12}\} \text{ otherwise} \end{cases}$$

Let us consider an example. The automaton on the left of figure 5 is the system we are studying, the one on the right is the system where the states 2 and 3 have been merged.

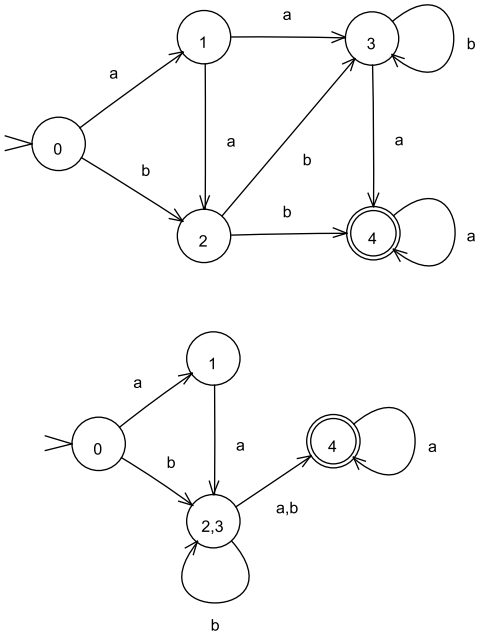

Figure 5. The system and its merged version.

By this definition, the language of the merged system $\overline{\mathcal{A}}$ contains the language of the first system $\mathcal{A}$, since the paths of $\mathcal{A}$ still exist in $\overline{\mathcal{A}}$. New words can obviously be matched, for instance by a run that reaches s_1 and then uses a transition starting from s_2. For example, in figure 5, the word *ba* is not recognized by $\mathcal{A}$, but is recognized by $\overline{\mathcal{A}}$.

However, if the two merged states are of the same type (safe or faulty), which is the case in our framework, then we can notice that the diagnosis of a word can only change to a more ambiguous one: if it was faulty or ambiguous, then it cannot become safe, and if it is ambiguous, then it cannot become safe or faulty. Indeed, if a run matching a given word leads to a faulty state (respectively safe), then it will still be the case after having merged two states of the same type. The state estimation of a word for $\overline{\mathcal{A}}$ will contain the state estimation of this word for $\mathcal{A}$, and then the diagnosis cannot become more accurate.

Merging faulty states: First, we examine the properties of a merger of two faulty states. A first result we will prove is that the diagnosability of a system can be lost when two faulty states are merged. As a counter-example, the first system $\mathcal{A}$ in figure 6 is diagnosable (its ambiguity depth $\mathcal{D}_{\mathcal{A}} = 1$), but the merged one is not,

since a run of $(ab)^*$ can be faulty and eternally diagnosed ambiguous.

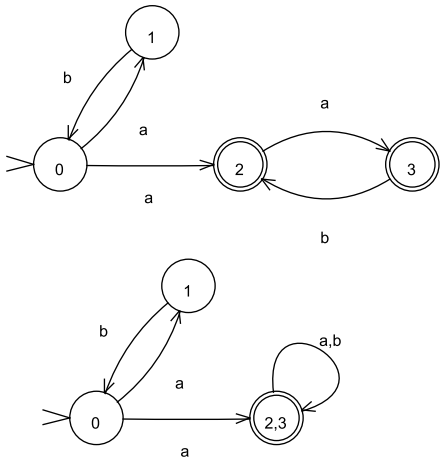

Figure 6. The diagnosable system $\mathcal{A}$ and its merged non-diagnosable version $\overline{\mathcal{A}}$.

The goal is now to "drive" the merged system $\overline{\mathcal{A}}$ with the former language $\mathcal{L}(\mathcal{A})$, and examine if the diagnosability is preserved for such a language. Obviously, we have to redefine the type of diagnosability we want for the merged system.

Definition 4 ((s_1, s_2)-Diagnosability)**.** *A **diagnosable** system $\mathcal{A}$ is (s_1, s_2)-diagnosable if and only if every **faulty** word of $\mathcal{A}$ is diagnosed faulty in the merged system $\overline{\mathcal{A}}$ in bounded time:* $\exists N \in \mathbb{N}, \forall \omega \in \mathcal{L}(\mathcal{A}) : \omega = \omega_1\omega_2$,
$\left[diag(\omega_1) = f \wedge |\omega_2| > N \Rightarrow \overline{diag}(\omega) = f\right]$

where $\overline{diag}$ is the diagnosis function of the merged system $\overline{\mathcal{A}}$, according to the merger of s_1 and s_2.

We do not care about the safe words of $\mathcal{L}(\mathcal{A})$, they can become infinitely ambiguous. For instance, in the example in figure 6, the word $ababab\ldots ab$ is safe in $\mathcal{A}$, but infinitely ambiguous in $\overline{\mathcal{A}}$. Hence the system $\mathcal{A}$ is $(2,3)$-diagnosable, but $\overline{\mathcal{A}}$ is not diagnosable according to the first definition.

We will prove the following theorem, which will finally allow us to merge all the faulty states, and restrict the framework to systems with only one faulty state:

Theorem 2. *If a system $\mathcal{A}$ is diagnosable and s_1 and s_2 are two faulty states, $\mathcal{A}$ is (s_1, s_2)-diagnosable.*

We first consider the automaton $\mathcal{A} \times \mathcal{T}_{\overline{\mathcal{A}}}$, and proves this lemma:

Lemma 1. *$\mathcal{A}$ is **not** (s_1, s_2)-diagnosable if and only if there is a cycle of type $S_f \times (\overline{S}_f \times \overline{S}_s)$ in $\mathcal{A} \times \mathcal{T}_{\overline{\mathcal{A}}}$.*

Proof. (of the lemma)

- Let us assume that there is such a cycle in $\mathcal{A} \times \mathcal{T}_{\overline{\mathcal{A}}}$. Hence for any $N \in \mathbb{N}$, we can find a word $\omega \in \Sigma^*$ of size n, which is diagnosed faulty in $\mathcal{A}$ at step $n - N - 1$ (because $\mathcal{A}$ is assumed to be diagnosable), and is still ambiguous $N + 1$ steps later in $\overline{\mathcal{A}}$. It is the logical complement of the definition of (s_1, s_2)-diagnosability:
$\forall N \in \mathbb{N}, \exists \omega \in \mathcal{L}(\mathcal{A}) : \omega = \omega_1\omega_2, diag(\omega_1) = f \wedge |\omega_2| > N \wedge \overline{diag}(\omega) \neq f$
- For the other implication, the negation of the property of (s_1, s_2)-diagnosability implies the existence of a cycle of type $S_f \times (\overline{S}_f \times \overline{S}_s)$ in $\mathcal{A} \times \mathcal{T}_{\overline{\mathcal{A}}}$: $\mathcal{A}$ is finite, so in order to find a word ω as long as we want, there exists a cycle in $\mathcal{A} \times \mathcal{T}_{\overline{\mathcal{A}}}$. Any word matching this cycle is faulty in $\mathcal{A}$ if it is long enough, but still ambiguous in $\overline{\mathcal{A}}$, since a faulty run can only become more ambiguous after the merger (i.e. $\overline{diag}(\omega) \neq f \Rightarrow \overline{diag}(\omega) = a$). Hence this cycle needs to be of type $S_f \times (\overline{S}_f \times \overline{S}_s)$.

□

Proof. (of the theorem 2)

According to the previous lemma, the system $\mathcal{A}$ is (s_1, s_2)-diagnosable if and only if there is no cycle of type $S_f \times (\overline{S}_f \times \overline{S}_s)$ in $\mathcal{A} \times \mathcal{T}_{\overline{\mathcal{A}}}$. Let us consider such a cycle. It can be described by a triplet of sequences of states $((f_k)_k, (f'_k)_k, (s_k)_k)$. If the sequences $(f_k)_k$ and $(f'_k)_k$ are different, then there exists a cycle described by $((f_k), (f_k), (s_k))$, by definition of the product. Since we only want the existence of such a cycle, we only consider cycles with $(f_k) = (f'_k)$. Hence, it suffices to examine the restricted automaton $\mathcal{A} \times \overline{\mathcal{A}}_{|S}$ instead of $\mathcal{A} \times \mathcal{T}_{\overline{\mathcal{A}}} = \mathcal{A} \times (\overline{\mathcal{A}} \times \overline{\mathcal{A}}_{|S})$, and search for a cycle of type $S_f \times \overline{S}_s$.

The merged states s_1 and s_2 are faulty, so the safe component of $\overline{\mathcal{A}}$ has not changed: $\overline{\mathcal{A}}_{|S} = \mathcal{A}_{|S}$ and $\overline{S}_s = S_s$. We immediately deduce that $\mathcal{A} \times \overline{\mathcal{A}}_{|S} = \mathcal{A} \times \mathcal{A}_{|S} = \mathcal{T}_{\mathcal{A}}$. The system $\mathcal{A}$ has been assumed to be diagnosable, so $\mathcal{T}_{\mathcal{A}}$ does not contain a cycle of $S_f \times S_s$. Hence, the automaton $\mathcal{A} \times \mathcal{T}_{\overline{\mathcal{A}}}$ does not contain a cycle of type $S_f \times (\overline{S}_f \times \overline{S}_s)$, and the result follows. □

We now want to use this theorem to prove that we can restrict the framework to systems with only one faulty state without loss of generality, i.e. we can merge the whole faulty component into one unique faulty state. We first need to prove the following result:

Proposition 4. *The order of the successive mergers does not change the final merged system.*

Proof. We only have to prove that the operation of merging two states is associative: given three states s_1, s_2, s_3, we want to prove $(s_1 \leftrightarrow s_2) \leftrightarrow s_3 = s_1 \leftrightarrow (s_2 \leftrightarrow s_3)$, where $(s_1 \leftrightarrow s_2)$ denotes the merger of s_1 and s_2. Then we will conclude by the generalized associative law that it is also true for any number of states merged.

Let $\overline{\mathcal{A}}_{12,3} = (\overline{S}_{12,3}, \Sigma, \overline{I}_{12,3}, \overline{\delta}_{12,3})$ be the system resulting from the merger $(s_1 \leftrightarrow s_2) \leftrightarrow s_3$, and $\overline{\mathcal{A}}_{23,1} = (\overline{S}_{23,1}, \Sigma, \overline{I}_{23,1}, \overline{\delta}_{23,1})$ the one that results from the merger $s_1 \leftrightarrow (s_2 \leftrightarrow s_3)$.

- $\overline{S}_{12,3} = (S \setminus \{s_1, s_2\} \uplus \{s_{12}\}) \setminus \{s_{12}, s_3\} \uplus \{s_{123}\} = (S \setminus \{s_1, s_2, s_3\}) \uplus \{s_{123}\}$
  $= (S \setminus \{s_2, s_3\} \uplus \{s_{23}\}) \setminus \{s_{23}, s_1\} \uplus \{s_{123}\}$
  Hence $\overline{S}_{12,3} = \overline{S}_{23,1}$.
- In the same way: if $\{s_1, s_2, s_3\} \neq \emptyset$, then $\overline{I}_{12,3} = \overline{I}_{23,1} = I \setminus \{s_1, s_2, s_3\} \uplus \{s_{123}\}$, and $\overline{I}_{12,3} = \overline{I}_{23,1} = I$ otherwise.
- Idem, $\forall \alpha \in \Sigma, \forall s \in S \setminus \{s_1, s_2, s_3\}$, if $\{s_1, s_2, s_3\} \cap \delta(s, \alpha) = \emptyset$, then $\overline{\delta}_{12,3}(s, \alpha) = \overline{\delta}_{23,1}(s, \alpha) = \delta(s, \alpha)$.
  Else, $\overline{\delta}_{12,3}(s, \alpha) = \overline{\delta}_{23,1}(s, \alpha) = \delta(s, \alpha) \setminus \{s_1, s_2, s_3\} \uplus \{s_{123}\}$.
  And for all $\alpha \in \Sigma$, $\overline{\delta}_{12,3}(s_{123}, \alpha) = \overline{\delta}_{23,1}(s_{123}, \alpha) = \delta(s_1, \alpha) \cup \delta(s_2, \alpha) \cup \delta(s_3, \alpha)$, by associativity of the union.

Hence $\overline{\mathcal{A}}_{12,3} = \overline{\mathcal{A}}_{23,1}$, and so $(s_1 \leftrightarrow s_2) \leftrightarrow s_3 = s_1 \leftrightarrow (s_2 \leftrightarrow s_3)$. By the generalized associative law, the merger of an arbitrarily number of states does not depend on the order of the successive mergers. □

Using this proposition, we can extend the definition of diagnosability for a merged system, since the definition of (s_1, s_2)-diagnosability does not depend on s_1 and s_2.

Definition 5. *(E-diagnosability) Given a diagnosable system $\mathcal{A} = (S, \Sigma, I, \delta)$, and $E \subseteq S_s$ or $E \subseteq S_f, E \neq \emptyset$. $\overline{\mathcal{A}}$ is the system resulting from the merger of all the states of E (in any order, according to proposition 4).*
$\mathcal{A}$ is E-diagnosable if and only if:
$\exists N \in \mathbb{N}, \forall \omega \in \mathcal{L}(\mathcal{A}) : \omega = \omega_1\omega_2,$
$\left[diag(\omega_1) = f \wedge |\omega_2| > N \Rightarrow \overline{diag}(\omega) = f\right]$
where $\overline{diag}$ is the diagnosis function of the merged system $\overline{\mathcal{A}}$.

This generalized definition of the diagnosability for a merged system now enables us to deduce a new property of the merger, which will lead us to the desired restriction of the framework.

Theorem 3. *Given a diagnosable system $\mathcal{A} = (S, \Sigma, I, \delta)$, and $E \subseteq S_s$ or $S_f, E \neq \emptyset$:*
$\left[\forall (a, b) \in E^2, \mathcal{A} \text{ is } (a, b)\text{-diagnosable}\right]$
$\Rightarrow \mathcal{A}$ *is E-diagnosable.*

Proof. We only have to prove that, after having merged a family $(x_1, \ldots, x_{n-1})$ of states (of the same type), if a new state x_n can be merged with all other x_i, it can be merged with the state $\overline{x}$ resulting from the merger of the x_i. An immediate induction on the cardinal of the family will then suffice to prove, using proposition 4, the theorem 3. So we have to prove the following proposition. Given $E = \{x_1, \ldots, x_{n-1}\} \subseteq S_f$ or S_s, with $\mathcal{A}$ being E-diagnosable, $\overline{\mathcal{A}}$ the merged system according to the merger of E, $\overline{x}$ the resulting state, and $x_n \in S \setminus E$ (in the same component as the elements of E): if $\mathcal{A}$ is (x_i, x_n)-diagnosable for any $i \in [\![1, n-1]\!]$, then $\mathcal{A}$ is $(E \uplus \{x_n\})$-diagnosable.

We start by assuming these hypotheses. We denote $\overline{\mathcal{A}}_E$ the resulting system from the merger of E, $\overline{diag}_E$ its associated diagnosis function, and for $x \in E$, $\overline{\mathcal{A}}_x$ the resulting system from the merger of x with x_n, and $\overline{diag}_x$ its diagnosis function. The diagnosis function of the system $\overline{\mathcal{A}}$ resulting from the merger of $E \uplus \{x_n\}$ is denoted $\overline{diag}$. To prove that $\mathcal{A}$ is $(E \uplus \{x_n\})$-diagnosable, we exhibit one N that verifies the definition 5. We have assumed that $\mathcal{A}$ is E-diagnosable: we denote N_E the "N" of the definition 5. We have also assumed that $\mathcal{A}$ is (x, x_n)-diagnosable for any $x \in E$: we denote N_x the N in the definition 4. The N we will consider is:

$$\tilde{N} = \max(N_E, \max_{x \in E} N_x)$$

Let ω be a word of $\mathcal{L}(\mathcal{A})$, such that $\omega = \omega_1\omega_2$, with $|\omega_2| > \tilde{N}$ and $diag(\omega_1) = f$ (with $diag$ the diagnosis function of the original system $\mathcal{A}$). By this way, we have: $\overline{diag}_E(\omega) = f$, and $\forall x \in E, \overline{diag}_x(\omega) = f$. We have to prove that $\overline{diag}(\omega) = f$, i.e. that $\forall \pi$ run of $\overline{\mathcal{A}}$ with $\sigma(\pi) = \omega, s^+(\pi) \in \overline{S}_f$. Let π be a run of $\overline{\mathcal{A}}$ with $\sigma(\pi) = \omega$.

- *Case 1:* If π does not hit $\overline{x}$ before its end, then the path of π is the same as in $\overline{\mathcal{A}}_E$, and since $\overline{diag}_E(\omega) = f$, then $s^+(\pi) \in \overline{S}_f$,
- *Case 2:* Else, if $s^+(\pi) \in \overline{S}_s$, then it means that $\exists x \in E, \exists \pi_0, \pi_1$ two **paths** of $\mathcal{A}$ such that $s^-(\pi_0) \in I, \sigma(\pi_0\pi_1) = \omega$, and either $s^+(\pi_0) = x \wedge s^-(\pi_1) = x_n$ or $s^+(\pi_0) = x_n \wedge s^-(\pi_1) = x$. But this implies that the run equivalent of π in $\overline{\mathcal{A}}_x$ also ends in a safe state, which implies $\overline{diag}_x(\omega) \neq f$ which is absurd. Hence, $s^+(\pi) \in \overline{S}_f$.

Hence for all π run of $\overline{\mathcal{A}}$ matching ω, π ends in a faulty state, which means $\forall \omega = \omega_1\omega_2$ with $|\omega_2| > \tilde{N}$ and $diag(\omega_1) = f$, $\overline{diag}(\omega) = f$. Our $\tilde{N}$ verifies the definition 5: $\mathcal{A}$ is $(E \uplus x_n)$-diagnosable.

As said in the beginning of the proof, an immediate induction on $n = |E| - 1$ suffices to prove that for any $E \subseteq S_s$ or S_f, if $\forall (a, b) \in E^2, \mathcal{A}$ is (a, b)-diagnosable, then $\mathcal{A}$ is E-diagnosable. □

We finally deduce the desired result:

Theorem 4. *A diagnosable system $\mathcal{A}$ is S_f-diagnosable.*

Proof. We use the theorem 3 with $E = S_f$. The hypothesis "$\forall (a, b) \in E^2, \mathcal{A}$ is (a, b)-diagnosable" is given by theorem 2. □

In other words, all the faulty states of a system can be merged, and the diagnosability is preserved according to our definition. Hence we can restrict the framework

to systems with only one faulty state, and only focus on the safe ones.

S_s-diagnosability: We finally consider a simple example that proves that there exists a system $\mathcal{A}$ which is not S_s-diagnosable. In figure 7 below, we can see the system $\mathcal{A}$ and its merged system $\overline{\mathcal{A}}$ according to the merger of its safe component S_s:

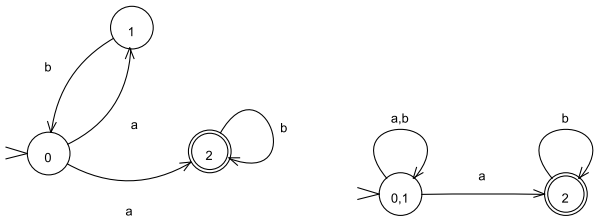

Figure 7. The system $\mathcal{A}$ and its merged system $\overline{\mathcal{A}}$.

$\mathcal{A}$ is diagnosable according to the classical definition, but is not S_s-diagnosable: we consider the words of $\mathcal{L}(\mathcal{A})$ of type $abb(b^*)$. Such words are as long as we want, diagnosed faulty in $\mathcal{A}$ from the third letter (a safe word cannot have two successive b). However, in $\overline{\mathcal{A}}$, such a word is infinitely ambiguous. Hence we have the property:
$\forall N \in \mathbb{N}, \exists \omega \in \mathcal{L}(\mathcal{A}), \omega = abb\omega'$,
$|\omega'| > N \wedge diag(abb) = f \wedge \overline{diag}(\omega) = a \neq f$.

This proves that $\mathcal{A}$ is not S_s-diagnosable.

Conclusion

In this report we introduced the framework of state estimation and diagnosis problems on finite automata. We first focused on the aspects of diagnosability, proving a first simple result about reachability of a faulty state in the diagnoser of a diagnosable system. Then we introduced the ambiguity depth, and used it to bound the lengths of the ambiguous chains in the diagnoser. We then focused on multiresolution aspects, by introducing a framework of merging several states. We defined a notion of diagnosability for merged systems, and finally proved that all faulty states of a diagnosable system could be merged without losing the diagnosability, but that it was not the case for the safe states.

The main line of research that should now be achieved is a study of the necessary and sufficient conditions that enable us to merge safe states without losing the diagnosability as we did for the faulty states. It must be more difficult, as the whole safe component cannot be merged according to our last counterexample. It can however lead to a problem of minimization of the number of states of the system, which would be really interesting.

Acknowledgments

I would like to thank my internship supervisor Éric Fabre, Romain Amand, Laura Chassard, Alexandre Debant, Olivier Havot, Sophie Pinchinat for their help, and the whole team SUMO from Inria Rennes for the welcome I have received. I would also like to thank Raphaël Berthon, David Pichardie and Owen Rouillé for their review of the current report.

References

[1] Eric Fabre. *Distributed Control of Large Plants; Chapter 5, Observers and automata*. Ed. by Carla Seatzu, Manuel Silva, and Jan H. van Schuppen. Springer, 2012.

[2] Peter J Ramadge and W Murray Wonham. "Supervisory control of a class of discrete event processes". In: *SIAM journal on control and optimization* 25.1 (1987), pp. 206–230.

[3] Meera Sampath et al. "Diagnosability of discrete-event systems". In: *Automatic Control, IEEE Transactions on* 40.9 (1995), pp. 1555–1575.

[4] JohnN. Tsitsiklis. "On the control of discrete-event dynamical systems". English. In: *Mathematics of Control, Signals and Systems* 2.2 (1989), pp. 95–107. ISSN: 0932-4194. DOI: 10.1007/BF02551817. URL: http://dx.doi.org/10.1007/BF02551817.

[5] Tae-Sic Yoo and Stephane Lafortune. *Polynomial-Time Verification of Diagnosability of Partially-Observed Discrete-Event Systems*. 2002.