TOWARDS LOW-COST FAULT DIAGNOSIS IN LARGE COMPONENT-BASED SYSTEMS [1]

Yannick Pencolé * **Dmitry Kamenetsky** **
Anika Schumann ***

* *Centre National de la Recherche Scientifique*
** *The Australian National University, NICTA*
*** *The Australian National University, NICTA*

Abstract: We address the problem of fault diagnosis in discrete-event systems. Our contribution is the development of a set of specialised diagnosers whose computation is much more realistic than that of the classical diagnoser. A specialised diagnoser is devoted to the diagnosis of one particular type of fault and is based on the observation of only a subpart of the system.

Keywords: Model-Based Diagnosis, Discrete Event Systems, Diagnoser Approach

1. INTRODUCTION

Monitoring large event-driven systems like communication networks, web services and business processes is a complex activity that requires automated methods. When a system operates, some critical events or *faults* may occur and the system supervisor has to detect them in order to make decisions to keep the system working. Most of these systems are component-based, i.e. each component communicates with other components by exchanging messages. The problem, known as fault diagnosis in discrete-event systems, is to determine a method for monitoring large systems and efficiently performing diagnosis given the flow of observations.

The classical model-based approach for monitoring discrete-event system is the *diagnoser approach* proposed by Sampath et al. (1995). A *diagnoser* is a finite state machine which is able to provide a diagnosis of the system for any sequence of observations produced by the system. The main advantage of this machine is that it is computed from a behavioural model of the system and performs fault diagnosis efficiently. Its main drawback is that its computation is based on the *global model* of the system, so it is exponential to the number of components of the system. This space complexity makes this diagnoser infeasible for large component-based systems that are more and more common in real-world applications.

In this paper, we propose a new diagnoser approach for component-based systems based on a set of *specialised diagnosers* whose computation is less complex. As opposed to the classical diagnoser, our diagnoser is devoted to the diagnosis of one type of fault only (one diagnoser per fault). Secondly, instead of taking into account the system as a whole, we propose to analyse the system in order to detect a *subsystem* that is *sufficient* for diagnosing this particular type of fault. In practice, the purpose of this approach is to drastically decrease the computation cost of any monitoring agent for component-based systems.

The paper is organised as follows. First, we present the background, i.e. component-based model and classical diagnoser. The second section then informally characterises specialised diagnosers and sufficient subsystems. Next, we present an algorithm which detects, for

[1] This research was supported by National ICT Australia (NICTA) in the framework of the SuperCom project (Model-Based Supervision of Composite Systems). NICTA is funded through the Australian Government's *Backing Australia's Ability* initiative, in part through the Australian National Research Council.

the diagnosis of a given fault, whether the observation of a given subsystem is sufficient and if so computes a specialised diagnoser for it.

2. BACKGROUND

2.1 Component-based Model

We study component-based and event-driven systems. Their model is based on classical automata (see Figure 1): one automaton represents the behaviour (also called the *local model*) of one component (Sampath et al. (1995)). This formalism is aimed at modelling any discrete event system with multiple and permanent faults. A fault occurs in one component and its consequences may propagate to other components.

Definition 1. (Local model). The *local model* G_i is an automaton $G_i = (Q_i, \Sigma_i, E_i, q_{0i})$ where:

- Q_i is a finite set of states; q_{0i} is the initial state;
- Σ_i is the set of events and $E_i \subseteq Q_i \times \Sigma_i \times Q_i$ is the set of transitions.

The set of events is divided into four disjoint subsets ($\Sigma_i = \Sigma_i^{obs} \oplus \Sigma_i^{com} \oplus \Sigma_i^{norm} \oplus \Sigma_i^{flt}$): Σ_i^{obs} the set of *observable events*, Σ_i^{com} the set of *communication events*, Σ_i^{norm} the set of *normal events* and Σ_i^{flt} the set of *fault events*.

A *subsystem* g is a non-empty set $\{G_{i_1} \dots G_{i_m}\}$ of components of the system. The *global model* of g is the automaton which results from the classical synchronised composition of the automata contained in g (see Sampath et al. (1995)). A state q of the subsystem g is a m-tuple $(q_{i1}, \dots, q_{im})$ of m local states. The global model of the system is the global model of its biggest subsystem, a system state is a n-tuple $(q_1, \dots, q_n)$ with q_i a state of G_i and n the number of components in the system.

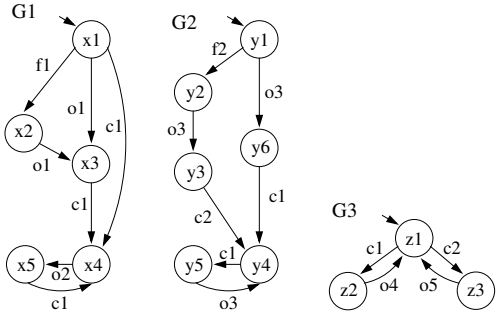

Fig. 1. Component-based system.

2.2 Extended component-based model

For the sake of clarity throughout this paper, we introduce an *extended representation* of the previous model based on an *extended automaton*. The aim of this extended representation is to provide further definitions in a unified way based only on a composition operation and a projection operation. The only difference between an extended automaton and a classical one is that it is composed of extended states. An extended state $x \in X$ is a couple $(Bs, Label) \in Bs(X) \times Label(X)$ where Bs represents a *belief state* (a set of model states denoted $Bs(x)$) and $Label$ represents a property about the belief state (denoted $Label(x)$). The extended local model Γ_i corresponding to the local model G_i is defined as follows:

Definition 2. (Extended local model). The *extended local model* Γ_i is an automaton

$$\Gamma_i = (X_i, \Sigma_i, T_i, x_{0i})$$

where:

- X_i is a finite set of extended states x such that $Bs(x) \in Q_i$ and $Label(x) \in 2^{\Sigma_i^{flt}}$;
- $x_{0i} = (q_{0i}, \emptyset)$ is the initial state;
- Σ_i is the set of events and
- $T_i \subseteq X_i \times \Sigma_i \times X_i$ is the set of transitions.

The extended version of the model is equivalent to the previous one (see Figure 2): an extended state $x = (q, \mathcal{F})$ of Γ_i simply means that the component G_i is in state $Bs(x) = q$ and there exists a transition path from q_{0i} to q in G_i in which the set of faults that occur is exactly $Label(x) = \mathcal{F}$.

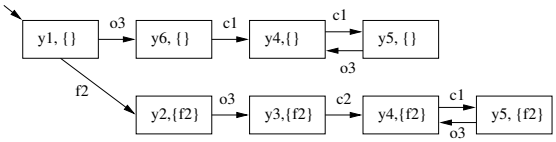

Fig. 2. Extented model Γ_2 of component $G2$.

We also define the extended model of any subsystem $g = \{G_{i1}, \dots, G_{im}\}$ (denoted $\gamma = \{\Gamma_{i1}, \dots, \Gamma_{im}\}$).

Definition 3. (Extended global model). The *extended global model* $\|\gamma\|$ is the extended automaton defined by:

$$\|\gamma\| = \Gamma_{i_1} \| \dots \| \Gamma_{i_m}.$$

The operator $\|$ is the composition operation synchronised on the communicating events $\mathcal{E} = \bigcup_{i=1}^{n} \Sigma_i^{com}$ (see Appendix A). By definition, an extended state x of γ is such that

$$x = (x_{i_1}, \dots, x_{i_m}) \equiv$$

$$(Bs(x_{i_1}), \dots, Bs(x_{i_m}), Label(x_{i_1}), \dots, Label(x_{i_m})).$$

So, by extension, we denote

$$Bs(x) = (Bs(x_{i_1}), \dots, Bs(x_{i_m})) \in \prod_{j=1}^{m} Q_{i_j}$$

and

$$Label(x) = (Label(x_{i_1}), .., Label(x_{i_m})) \in \prod_{j=1}^{m} 2^{\Sigma_{i_j}^{flt}}.$$

In the rest of this paper, we will only use the notations of the extended representation to denote any part of the system (component, subsystem...).

2.3 Classical Diagnoser

In this paper, we address the problem of detecting the occurrence of fault events in a monitoring context: given a flow of observable events emitted by the system, the problem is to provide diagnosis updates after each observation of the flow. For that purpose, Sampath et al. (1995) defines a determinitic finite-state machine, called *diagnoser*, that diagnoses the set of faults $\Sigma^{flt} = \bigcup_{i=1}^{n} \Sigma_i^{flt}$ given an observation flow from the system. Generally, the diagnoser is defined relying on the global model of the system. In the following, we define the same machine based on the extended global model. Let us consider the extended global model $\|\Gamma\| = (X, \Sigma, T, x_0)$ such that $X \subseteq Q \times 2^{\Sigma^{flt}}$ where Q is the set of system states. Before defining the diagnoser based on $\|\Gamma\|$, we define the diagnoser function f_Δ which gathers the diagnosis information from the extended states of $\|\Gamma\|$:

$$f_\Delta : 2^X \rightarrow 2^{Q \times 2^{\Sigma^{flt}}}$$
$$f_\Delta(x_1, \ldots, x_m) = \bigcup_{i=1}^{m} (Bs(x_i), Label(x_i))$$

The classical diagnoser is then defined by projecting the extended global model on the observable events $\Sigma^{obs} = \bigcup_{i=1}^{n} \Sigma_i^{obs}$ (see Appendix B).

Definition 4. The classical diagnoser $\Delta(\Gamma)$ of the system Γ is the extended automaton:

$$\Delta(\Gamma) = P_{\Sigma^{obs}, f_\Delta}(\|\Gamma\|).$$

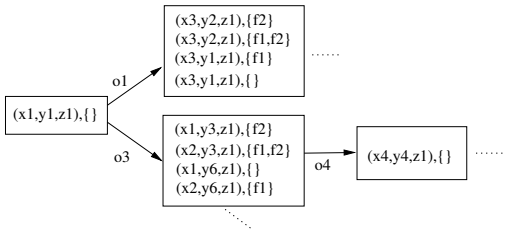

Fig. 3. Part of the classical diagnoser of the model from Fig.1.

The diagnoser is a deterministic extended automaton whose transitions are labelled with observable events only and that is able to efficiently provide a diagnosis after each observation (see Figure 3). The provided diagnoses are contained in the diagnoser states. The diagnosis of a diagnoser state z is contained in $Label(z)$. This diagnosis is composed of a set of belief states (i.e. $Label(z) = \bigcup_i (Bs(x_i), Label(x_i))$), each belief state (i.e. $Bs(x_i) \in Q$) being associated with a set of possible faults that could have occurred before reaching this belief state (i.e. $Label(x_i) \in 2^{\Sigma^{flt}}$).

If we consider the monitoring of a component-based system, the main problem is about the algorithmic cost of the diagnoser computation. If n is the number of components in the system, then the number of states in the classical diagnoser is in the worst case in $2^{2^n} \times 2^{|\Sigma^{flt}|}$. Clearly, the computation of such a machine is unrealistic because of limited computing resources.

3. FAULT DIAGNOSIS SPECIALISATION

Diagnosing a particular fault given a flow of observations can be an independent process where the diagnosis of the other faults is not involved. Instead of having one machine that diagnoses every type of fault, we can set up a set of $|\Sigma^{flt}|$ *specialised* machines where each machine is in charge of diagnosing one type of fault only. Adopting this point of view has two advantages. Firstly, if the diagnosis task is to only detect the occurrence of faults, then the set of specialised diagnosers provides the same diagnosis information as the classical diagnoser.[2] Secondly, the size of a diagnoser does not depend on the number of possible faults but only on the number of components in the system.

3.1 Definition

Definition 5. Let γ be a subsystem and F a fault that could occur in γ, an F-diagnoser for γ is a finite-state machine that, given any flow of observations from γ, can decide at any time if γ is:

- *safe* (F has not occurred),
- *faulty* (F has occurred),
- *ambiguous* (F may have occurred).

An F-diagnoser is devoted to the diagnosis of one particular fault F. Moreover, it must be able to provide a diagnosis at any time, which means that this machine must be able to *follow* the observation flow from γ and to efficiently *provide* a diagnosis after each new observation. The F-diagnoser is not unique, F-diagnosers are a family of machines which is denoted $\mathcal{D}_\gamma(F)$.

[2] Only information about fault correlations is lost by the specialised diagnosers.

3.2 Classical diagnoser as an F-diagnoser

By definition, $\Delta(\Gamma)$ is an F-diagnoser of Γ and is actually an F-diagnoser for every type of fault F that could occur in the system Γ.

Proposition 6.

$$\forall F \in \Sigma^{flt}, \Delta(\Gamma) \in \mathcal{D}_\Gamma(F)$$

Indeed, a diagnoser state z contains the diagnosis $Label(z) = \{(q_1, \mathcal{F}_1), \ldots, (q_p, \mathcal{F}_p)\}$ where $\mathcal{F}_i$ is a set of possible faults that could have occurred before the system reaches the state q_i. Therefore, from that state z, we can easily decide if the system is faulty ($\forall i, F \in \mathcal{F}_i$), safe ($\forall i, F \notin \mathcal{F}_i$) or ambiguous ($\exists i, j, F \in \mathcal{F}_i \wedge F \notin \mathcal{F}_j$).

3.3 Towards a small F-diagnoser

A fault F that occurs in a component produces some consequences in this component and also in its neighbourhood, and among these consequences, some of them are observable. We can argue that it is sufficient to look at a given subsystem (a neighbourhood) in order to observe these consequences and then diagnose this fault. In other words, a sufficient F-diagnoser does not need to take into account all the observations from the system, but only part of it to diagnose this fault. The challenge is to find a subsystem that is sufficient to observe in order to perform fault diagnosis with *accuracy* and efficiency. Diagnosis *accuracy* is defined as follows:

Definition 7. An F-diagnoser is *accurate* iff for every observation sequence σ of the system ending with an event observed by the F-diagnoser, the diagnosis of the F-diagnoser is the diagnosis of the classical diagnoser with respect to F.

In other words, an accurate F-diagnoser only observes a part of the observation flow but is able to provide the same diagnosis as the classical diagnoser each time the last observation of the flow is seen by both machines.

4. COMPUTATION OF AN ACCURATE F-DIAGNOSER

This section presents an algorithm that computes an accurate F-diagnoser. Before explaining how to compute it, we present a way to compute an F-diagnoser Δ_γ for the subsystem γ.

4.1 F-diagnoser computation

The computation of the *F-diagnoser* Δ_γ is the same as the computation of a classical diagnoser except that:

(1) it is defined on any subsystem γ where F occurs;
(2) it uses a *specialised diagnoser* identification function f_F^γ instead of the classical diagnoser function f_Δ.

Given $\|\gamma\| = (X, \Sigma, T, x_0)$ the extended global model of γ, the identification function $f_F^\gamma : 2^X \rightarrow \{safe, faulty, ambiguous\}$ is defined as follows:

$$f_F^\gamma(\{x_1, .., x_m\}) = \begin{cases} safe & \text{iff } \forall i, F \notin Label(x_i) \\ faulty & \text{iff } \forall i, F \in Label(x_i) \\ ambiguous & \text{otherwise.} \end{cases}$$

Like the classical diagnoser, Δ_γ is then defined as a projection of the extended global model $\|\gamma\|$. The projection is performed on the observable events Σ_{obs}^γ of γ (see Figure 4 on the right).

Definition 8. The *F-diagnoser* Δ_γ of γ is the extended automaton:

$$\Delta_\gamma = P_{\Sigma_{obs}^\gamma, f_F^\gamma}(\|\gamma\|).$$

A state x of Δ_γ has the form

$$x = (Bs(x), Label(x))$$
$$= (\{Bs(x_1), \ldots, Bs(x_m)\}, f_F^\gamma(x_1, \ldots, x_m))$$

where $x_i = (Bs(x_i), Label(x_i))$ is a state of $\|\gamma\|$. Informally, a state x is the association of a belief state ($Bs(x) = \bigcup_{i=1}^m Bs(x_i)$) and a diagnosis property about the occurrence of the fault.

By construction of Δ_γ, the following property holds: let σ be an observation sequence of the system ending with an observation from γ, σ_γ be the subpart of σ observed from γ, and $d(\sigma)$ (resp. $d(\sigma_\gamma)$) be the diagnosis of $\Delta(\Gamma)$ (resp. Δ_γ) after the observation of σ (resp. σ_γ),

Proposition 9. The following assertions hold:

(1) $d(\sigma) = safe \Rightarrow d(\sigma_\gamma) \in \{safe, ambiguous\}$
(2) $d(\sigma) = faulty \Rightarrow d(\sigma_\gamma) \in \{faulty, ambiguous\}$
(3) $d(\sigma) = ambiguous \Rightarrow d(\sigma_\gamma) = ambiguous$

In other words, after the observation of σ, if the last observation of σ is from γ, Δ_γ provides a diagnosis that is never incorrect but is generally more ambiguous since the diagnosis is based on fewer observations. If the diagnoser is accurate then the provided diagnosis is exactly the same which means that an accurate diagnoser observes a flow of events that is sufficient to provide the same diagnosis.

4.2 Detection of an accurate diagnoser

Checking whether monitoring γ is sufficient for diagnosing the fault F is performed by checking a property

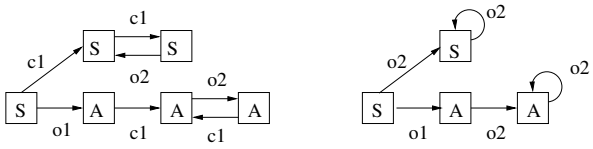

Fig. 4. Δ_γ^{int} and Δ_γ on the subsystem $\gamma = \{\Gamma_1\}$. For any state x, only $Label(x)$ is depicted.

on an extention of Δ_γ called the *interactive diagnoser* of γ and denoted Δ_γ^{int}. The only difference between Δ_γ^{int} and Δ_γ is that the communication events from γ are supposed to be observable in Δ_γ^{int}.

Definition 10. The *interactive F-diagnoser* Δ_γ^{int} of γ is the extended automaton:

$$\Delta_\gamma^{int} = P_{\Sigma_\gamma^{obs} \cup \Sigma_\gamma^{com}, f_F^\gamma}(\|\gamma\|).$$

The detection of an accurate diagnoser is based on Proposition 11. Let σ_γ be an observable sequence from γ and $\mathcal{P}(\sigma_\gamma)$ be the set of paths from the initial state in Δ_γ^{int} whose observable part is exactly σ_γ and the last event of the path is an event of σ_γ.

Proposition 11. Δ_γ is accurate if the following criterium holds: $\forall \sigma_\gamma, \forall x, x' \in \Delta_\gamma^{int}$ target states of paths from $\mathcal{P}(\sigma_\gamma)$, $Label(x) = Label(x')$.

PROOF. Let σ be an observation sequence ending with an observable event from γ and σ_γ be the subpart of σ emitted by γ. Let $d(\sigma)$ be the diagnosis provided by the classical diagnoser after the observation of σ. Every transition path p from $\|\Gamma\|$ that emits σ has a representative $p_\gamma \in \mathcal{P}(\sigma_\gamma)$ (by removing from p any event that does not belong to $\Sigma_\gamma^{com} \cup \Sigma_\gamma^{obs}$). If the criterium holds in Δ_γ^{int} then the target states of the paths in $\mathcal{P}(\sigma_\gamma)$ provide only one kind of diagnosis denoted $d(\sigma_\gamma)$. By construction, $d(\sigma_\gamma)$ is also the diagnosis provided by Δ_γ after the observation of σ_γ. If $d(\sigma_\gamma) = safe$ (resp. $faulty$), every path p_γ represents a set of paths from $\|\gamma\|$ that do not contain (resp. contains) F so $d(\sigma) = safe$ (resp. $d(\sigma) = faulty$). If $d(\sigma_\gamma) = ambiguous$ then every path p_γ represents at least two paths from $\|\gamma\|$, the first one contains F but not the second one, so $d(\sigma) = ambiguous$.

To summarise, the detection and the computation of an accurate F-diagnoser is depicted below:

1: *Input:* $F \in \Sigma^{flt}, \Gamma = \{\Gamma_1, \ldots, \Gamma_n\}$
2: $\gamma \leftarrow \{\Gamma_1\}$; Compute $\Delta_{\Gamma_1}^{int}$
3: **while** $\neg criterium(\Delta_\gamma^{int})$ **do**
4: Select Γ_j a neighbour of γ; $\gamma \leftarrow \gamma \cup \{\Gamma_j\}$
5: Compute Δ_γ^{int}
6: **end while**
7: *Output:* $\Delta_\gamma = P_{\Sigma_\gamma^{obs}, f_F^\gamma}(\Delta_\gamma^{int})$

The basic idea is to select the smallest subsystem where the fault F occurs (in the algorithm, we suppose F occurs in Γ_1). If the criterium for accuracy in the current subsystem holds, we compute the corresponding F-diagnoser. If not, we select a component Γ_j that communicates with the current subsystem γ (the neighbour selection requires a merging strategy that is similar to the strategy presented in Pencolé and Cordier (2005)) and we do the checking again. The algorithm terminates in the worst case with an F-diagnoser on the whole system but still smaller than the diagnoser. This case only occurs if the observable consequences of a given fault depend on all the components which is unlikely in large component-based systems.

4.3 Example

In the running example of Figure 1, the classical diagnoser $\Delta(\Gamma)$ (see Figure 3) contains 35 states and 68 transitions. Figure 4 presents an accurate f_1-diagnoser (right side). In the interactive diagnoser $\Delta_{\Gamma_1}^{int}$ (left side), every path emitting the sequences $o2^\star$ is $safe$ and every path emitting $o1o2^\star$ is ambiguous. There is no way to disambiguise this diagnosis relying on the observations of other components. As far as the fault f_2 is concerned (see Figure 1), the observation of the subsystem $\{G2, G3\}$ is sufficient to diagnose f_2 with the same accuracy than the diagnoser. The corresponding f_2-diagnoser contains 10 states and 16 transitions.

5. RELATED WORK

Fault diagnosis on discrete event systems have been studying for several years in both AI and Control communities (see Sampath et al. (1995), Lamperti and Zanella (2003), Fabre et al. (2005), Pencolé and Cordier (2005) for instance). The approach we propose is for fault identification and mainly follows the framework of Sampath et al. (1995). This approach is original in the sense that the method is centralised (the diagnoser is self-dependent, no communication is required with other diagnosers to perform the diagnosis) but does not require in practice the computation of the global model like the classical technique. This approach is related to the notion of clustering (see Lamperti and Zanella (2003)) which consists in detecting offline clusters of components (or subsystems) whose properties make the monitoring task (online diagnosis) easier. This approach can be easily extended to perform a decentralised diagnosis: if the subsystem contains different observation sites, the specialised diagnoser can be split into a set of local diagnosers (one per site) that communicate by using a communication protocol (Debouk et al. (2002)). Finally, the notion of accuracy is closely related to the notion of local diagnosability (see Sengupta (1998) and Pencolé (2004)): if F is locally diagnosable on a set of subsystems $\gamma_1, \ldots, \gamma_l$, then any accurate F-diagnoser Δ_γ is such that $\exists i \in \{1, \ldots, l\}, \gamma_i \subseteq \gamma$.

6. CONCLUSION

We presented a new type of generic machines for diagnosing faults in a component-based discrete event system. The originality of these machines is that they are devoted to the identification of a type of fault only, which makes their computation more tractable than the computation of the classical diagnoser and allows to diagnose larger component-based systems. Each specialised diagnoser is based on a subsystem that guarantees the diagnoser is accurate and provides a correct diagnosis. The detection of an accurate diagnoser is based on a sufficient condition which characterises the fact that a subsystem contains enough observable information to diagnose the given fault.

The next challenge is to find a necessary and sufficient condition so that we are able to characterise optimal accurate diagnosers. The main idea consists then in finding an optimal merging strategy which guarantees the computation of the smaller accurate diagnoser. Another interesting topic is the study of the relationship between the notions of diagnosability and accuracy in order to take into account the fact that a system is diagnosable when computing an accurate diagnoser.

REFERENCES

R. Debouk, S. Lafortune, and D. Teneketzis. Coordinated decentralized protocols for failure diagnosis of discrete event systems. *JDEDS: Theory and Application*, 10(1–2):33–86, 2002.

E. Fabre, A. Benveniste, S. Haar, and C. Jard. Distributed monitoring of concurrent and asynchronous systems. *Journal of Discrete Event Dynamic Systems*, 15:33–83, March 2005.

G. Lamperti and M. Zanella. *Diagnosis of active systems*. Kluwer Academic Publishers, 2003.

Y. Pencolé. Diagnosability analysis of distributed discrete event systems. In *Proc. ECAI'04*, pages 43–47, 2004.

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*, 164: 121–170, May 2005.

M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis. Diagnosability of discrete event system. *IEEE Trans. on Automatic Control*, 40(9):1555–1575, 1995.

R. Sengupta. Diagnosis and communication in distributed systems. In *Proc. WODES'98*, pages 144–151, Cagliari, Italy, 1998.

Appendix A. COMPOSITION OPERATION

Let $\{\mathcal{A}_i\}_{i\in\{1,\ldots,m\}}$ be m automata, let $\mathcal{E}_i$ be the set of events of $\mathcal{A}_i$ and let $\mathcal{E}$ be one subset of $\bigcup_{i=1}^{m}\mathcal{E}_i$. The composition $\|$ is such that the automaton $\mathcal{A} = \mathcal{A}_1\|\ldots\|\mathcal{A}_m$ is defined as an automaton $(X, \bigcup_{i=1}^{m}\mathcal{E}_i, T, q_0)$ where $X \subseteq \prod_{i=1}^{m} X_i$ such that for all $q \in X, q = (q_1, \ldots, q_m)$. The set of transitions is the subpart of the cartesian product of the $\mathcal{A}_i$'s containing the synchronised transitions according to a set of events $\mathcal{E}$. A transition $t = (q_1, \ldots, q_m) \xrightarrow{e} (q'_1, \ldots, q'_m)$ is synchronised according to $\mathcal{E}$ iff

$$
\begin{array}{c}
(e \notin \mathcal{E} \Rightarrow (\exists j \in \{1,\ldots,m\}, q_j \xrightarrow{e} q'_j \in \mathcal{A}_j \\
\wedge \forall i \in \{1,\ldots,m\} \setminus \{j\}, q_i = q'_i)) \\
\wedge (e \in \mathcal{E} \Rightarrow (\forall i \in \{1,\ldots,m\}, \\
e \in \mathcal{E}_i \Rightarrow q_i \xrightarrow{e} q'_i \in \mathcal{A}_i \wedge e \notin \mathcal{E}_i \Rightarrow q_i = q'_i))
\end{array}
$$

$\mathcal{A}$ represents the behaviour of $\{\mathcal{A}_i\}_{i\in\{1,\ldots,m\}}$ where only the events of $\mathcal{E}$ are synchronised.

Appendix B. PROJECTION OPERATION

Let $\mathcal{A} = (X, \Sigma, T, x_0)$ be an extended automaton based on the set of events Σ and Σ' be another set of events and let $f : 2^X \rightarrow \mathit{Label}$ be a state-property function on $\mathcal{A}$, the projection $P_{\Sigma',f}(\mathcal{A})$ of $\mathcal{A}$ on Σ' and f is the deterministic extended automaton (X', Σ', T', x'_0) such that:

- $X' \subseteq 2^{Bs(X)} \times \mathit{Label}$ is the set of states
- $T' \subseteq X' \times \Sigma' \times X'$ is the set of transitions
- x'_0 is the initial state

This machine is built as follows:

(1) $x'_0 = (\{Bs(x_0)\}, f(\{x_0\}))$

(2) for a given $x' = (\{Bs(x_{i_1}), \ldots, Bs(x_{i_m})\}, f(\{x_{i_1}, \ldots, x_{i_m}\})) \in X'$, we consider all the transition paths p from $\mathcal{A}$ such that $p = x_{i_j} \xrightarrow{\sigma_1} \ldots \xrightarrow{\sigma_m} x'_j \xrightarrow{\sigma'} x''_j$ where $\sigma_l \notin \Sigma', \forall l$ and $\sigma' \in \Sigma'$. We denote by $x''(\sigma')$ the set of target states x''_j for a given event σ', then we have $x'' = (Bs(x''(\sigma')), f(x''(\sigma')) \in X'$ and $x' \xrightarrow{\sigma'} x'' \in T'$.

The result of the projection is an extended automaton whose events are in Σ' only. Any state is defined as a set of belief states of $\mathcal{A}$ (i.e. a new belief state) and a label resulting from the application of the function f on that belief state.