Exploiting P-invariant analysis for distributed systems diagnosis based on interacting behavioral Petri nets

Hammadi Bennoui
Computer Science Depart.
University of Biskra
07 000, Algeria
bennoui@gmail.com

Allaoua Chaoui
Computer Science Depart.
University of Constantine
25 000, Algeria
a_chaoui2001@yahoo.com

Kamel Barkaoui
CNAM, 292 Rue St-Martin
Paris, 75 141
Cedex 03, France
kamel.barkaoui@cnam.fr

This paper deals with the problem of distributed causal model-based diagnosis on interacting Behavioral Petri Nets (BPNs). The system to be diagnosed comprises different interacting subsystems (each modeled as a BPN) and the diagnostic system is defined as a multi-agent system where each agent is designed to diagnose a particular subsystem on the basis of its local model, the local received observation and the information exchanged with the neighboring agents. The interactions between subsystems are captured by tokens that may pass from one net model to another via bordered places. The diagnostic reasoning scheme is accomplished locally within each agent by analyzing the P-invariants of the corresponding BPN model. Once local diagnoses are obtained, agents begin to communicate to ensure that such diagnoses are consistent and recover completely the results obtained by a centralized agent having a global knowledge about the whole system.

Keywords: Model-based diagnosis, causal models, Petri nets, reachability analysis, P-invariants.

1. INTRODUCTION

This paper addresses the problem of causal model-based diagnosis of distributed systems. We consider the system to be diagnosed as a collection of interacting subsystems in which when a fault occurs in one subsystem, it may generate some fault indications (i.e. symptoms) and may propagate to the neighboring. The diagnostic system itself is defined as a set of diagnostic agents each of which is associated to a specific subsystem. In particular, each agent has a local model of the assigned subsystem and may receive observations generated only by elements of this subsystem. The local model describes the causal behavior of the subsystem as well as its interactions within adjacent ones. When agents observe an aberrant behavior, each one is charged to explain the received local observation on the basis of its local model. This leads to calculate for each diagnostic agent a set of local diagnoses. In causal models [6], the diagnoses are to be given in terms of initial states that explain the set of observed symptoms using the cause-effect relationships described in the model. Such initial states represent the initial perturbations leading the system to behave abnormally.

Since each agent has a limited knowledge about the whole system to be diagnosed, it may be possible that local diagnoses of different agents are inconsistent when they are considered altogether. In order to ensure the required consistency and to guarantee that such local diagnoses recover completely global ones that are obtained by a centralized agent having a global view of the whole system, agents should communicate among them to reject inconsistent diagnoses.

In this paper we extend our previous work presented in [2] in which a particular class of Petri nets called "Behavioral Petri Nets" (BPNs) is used as a modelling and reasoning tool. In such a work, the causal behavior of each subsystem is described by a local BPN model and interactions among subsystems are captured through tokens that may pass via bordered places between BPNs. The adopted diagnostic reasoning scheme has been implemented as a reachability analysis based on the associated reachability graphs. In particular, during local computation agents exploit a backward reachability analysis (BW-analysis) of such graphs to explain locally the received observations. The BW-analysis leads to obtain by each agent a set of local initial markings from which diagnoses have to be given. Then, to achieve the consistency with the local diagnoses of all other agents, each one requests from its neighboring the required marking of its bordered places for each computed diagnosis. At this step, agents receiving such a request will construct their reachability graphs in a forward fashion to check if

the requested marking of bordered places is reachable from at least one of their computed initial markings. If so, the local diagnosis from which the exchanged message has been generated is considered globally consistent; otherwise, it is not supported by diagnoses of the neighboring and consequently it must be discarded.

The above described method suffers from the so-called state space explosion problem even for small net models. This is due to the utilization of reachability graphs as a diagnostic reasoning scheme especially in the consistency checking phase where several graphs may be constructed by each agent. In order to face such a problem, we propose to exploit algebraic analysis techniques, known also as invariant analysis, which are shown useful in [4, 14, 15] for improving complexity in centralized diagnostic reasoning based on Petri nets versus reachability graph analysis. In particular, we concentrate in this paper on the distributed analysis of P-invariants of the net models which are generated in an off-line manner. More specifically, we require that each agent utilizes the set of minimal supports of its P-invariants to implement local preliminary computation as well as to check the required consistency of its local diagnoses with those of the other agents. Thus, the set of minimal supports of P-invariants may be considered as a pre-compiled structure of the system model on which diagnosis is implemented. The idea of using a compiled structure of the system model to the on-line diagnosis is borrowed to Sampath *et al.* [18] in their work on discrete event systems (DES) diagnosis. They propose to generate from a finite state automaton describing the system model another automaton termed the Diagnoser which encompasses more information about the system state (i.e. information about the presence or absence of faults). The diagnoser is used to both test the diagnosability properties of the system and perform on-line monitoring of the system for the purpose of diagnosis which necessitates a synchronization between the diagnoser and the system model. Thus, the present work is similar to the Diagnoser approach regarding the off-line pre-compilation of the system model to face the complexity problem during on-line diagnosis. However, they differ according to several aspects, namely we use causal models in which the observations to be explained are modelled as partial states of the system to be diagnosed and not as observable events of the Diagnoser approach. Similarly, the faults in terms of which diagnoses will be given are considered as initial states which have no causes in the causal model and not as unobservable transitions adopted in the context of DES diagnosis. Another difference is that we do not require that such a compiled structure will be synchronized with the system model which is one of the key features of the approach of [18] and all its extensions [8, 9, 10]. It is to be noted however that in this paper we do not treat the question of diagnosability analysis and we concentrate only on how relating the P-invariants of the BPN models to diagnostic solutions.

The remainder of this paper is organized as follows: section 2 recalls briefly some basic notions related to Petri nets. In section 3, local calculations performed by each diagnostic agent to explain the received set of symptoms are presented. We begin the section by introducing the system description as a set of interacting BPNs; then we show how to characterize the set of diagnoses for a given diagnostic problem where the model corresponds to a net model by the set of minimal supports of the net's P-invariants. The protocol used between the different diagnostic agents to refine the obtained local diagnoses is detailed in section 4. Related works are discussed in section 5. Section 6 finally summarizes the paper and outlines future works.

2. PETRI NETS: OUTLINE

This section outlines briefly some basic definitions on which we will rely throughout the paper. An interested reader can find more details in [12].

Definition 2.1. A Petri net is a triple $N = \langle P, T, F \rangle$ where

- $P \cap T = \emptyset$
- $P \cup T \neq \emptyset$

- $F \subseteq (P \times T) \cup (T \times P)$
- $dom(F) \cup cod(F) \subseteq P \cup T$.

P is the set of places, T is the set of transitions and F is the flow relation represented by means of directed arcs. If the transitive closure F^+ of the arcs is irreflexive, the net is said to be *acyclic*.

In a Petri net, an arc multiplicity function is usually defined as $W: (P \cup T) \times (T \cup P) \rightarrow \leq$; if W is

such that $W(f) = 1$ if $f \in F$ and $W(f) = 0$ if $f \notin F$, the net is said to be an *ordinary Petri net*. For

each $x \in P \cup T$ we will use the classical notations $\bullet x = \{y \mid y F x\}$ and $x^\bullet = \{y \mid x F y\}$. If $\bullet x = 2$, x is

said to be a *source*; while if $x^\bullet = 2$, x is said to be a *sink*. A marking is a function $\mu: P \rightarrow \leq$ from places to nonnegative integers represented by means of *tokens* into places. A marked Petri net is a pair $\langle N, \mu \rangle$ where $N = \langle P, T, F \rangle$ is a Petri net and μ is a marking.

The dynamics of the net is described by moving tokens from places to places according to the following definition of *enabling* (i.e. *concession*) and *firing rules*.

Definition 2.2. Let $\langle P, T, F, \mu \rangle$ be a marked ordinary Petri net; a transition $t \in T$ is enabled at μ if and only if $\forall p \in \bullet t: \mu(p) \geq 1$; if t is enabled at μ , then t may occur (fire) yielding a new marking μ' (we write $\mu \xrightarrow{t} \mu'$) such that for every place $p \in P$ we have: $\mu'(p) = \mu(p) - W(p, t) + W(t, p)$.

The reachability set from a marking μ_0 , indicated as $[\mu_0]$, is the smallest set of markings such that: 1) $\mu_0 \in [\mu_0]$; 2) if $\mu_1 \in [\mu_0]$ and $\mu_1 \xrightarrow{t} \mu_2$ for some $t \in T$, then $\mu_2 \in [\mu_0]$. If a place of a marked net cannot be marked with more than one token, the place is said to be *safe*; if the property holds for every place, the net itself and every marking are said to be *safe*. Moreover, let $Q \subseteq P$, a marking μ of N covers Q if and only if $\forall p \in Q \rightarrow \mu(p) = 1$; while it zero-covers Q if and only if $\forall p \in Q \rightarrow \mu(p) = 0$.

Given a Petri net $N = \langle P, T, F \rangle$ with $n = |T|$ and $m = |P|$, the *incidence matrix* of N is the $n \times m$ matrix of integers $A = [a_{ij}]$ such that $a_{ij} = W(i, j) - W(j, i)$ ($i \in T, j \in P$). An m -vector of integers Y such that $A \cdot Y = 0$ is said to be a *P-invariant* of the net represented by A , the entry $Y(j)$ corresponds to place j . The support σ_Y of a P-invariant Y is the subset of places corresponding to nonzero entries of Y . In a dual way, if A^T is the transpose matrix of A , an n -vector of integers X such that $A^T \cdot X = 0$ is said to be a *T-invariant* (entries corresponding to transitions). It is well known that any invariant can be obtained as a linear combination of invariants having minimal (with respect to set inclusion) supports.

3. LOCAL PRELIMINARY DIAGNOSIS

After introducing how BPNs are used to model the causal behavior of a distributed system, we present in this section how relating P-invariants to diagnostic solutions where the system model is described as a set of place-bordered BPNs.

3.1. The system model

BPNs are introduced by [1] to deal with centralized diagnosis based on causal models. In particular, the causal behavior of the system to be diagnosed is described by a safe BPN model; and diagnostic reasoning schemes are reformulated in terms of reachability problems based on such a net model. Before showing how BPNs have to be used for modelling the causal behavior of a distributed system, let us firstly recall the following definition.

Definition 3.1. (from [1]) A Behavioral Petri Net (BPN) is a 4-tuple $N = (P, T_N, T_{OR}, F)$ such that

$(P, T_N \cup T_{OR}, F)$ is an acyclic ordinary Petri net that satisfies the following axioms:

1. $\forall p \in P (|\bullet p| \leq 1 \wedge |p\bullet| \leq 1)$
2. $\forall p_1, p_2 \in P (\bullet p_1 = \bullet p_2) \wedge (p_1\bullet = p_2\bullet) \rightarrow p_1 = p_2)$
3. $\forall t \in T_N (|\bullet t| = 1 \wedge |t\bullet| > 0) \vee (|\bullet t| > 0 \wedge |t\bullet| = 1)$
4. $\forall t \in T_{OR} (|\bullet t| \geq 2 \wedge |t\bullet| = 1)$

In this definition, the set of transitions is partitioned into two sets T_N and T_{OR} . Transitions in T_N (and-transitions) are intended in the usual way; while those in T_{OR} (or-transitions) are intended to represent the logical connective *OR*. Informally, a transition $t \in T_{OR}$ (graphically represented as an empty thick bar) has concession in a marking if and only if at least one of its input places is marked.

In causal models, the behavior of a system is characterized by a set of states (in fact, *partial* states that partially describe a situation in which the modelled system can be at a given time) and relations among these states (i.e. cause-effect transformations among the states). For diagnostic purposes, the set of states are classified into three categories [6]: *initial states* which correspond to entities that have no causes in the model and represent in the case of an abnormal behavioral model the initial perturbations leading the system to a given malfunction, *internal states* corresponding to the unobservable consequences of initial states, and *findings* which are considered as observable manifestations of internal states. In this view, a diagnostic process exploits the relationships among states in the model to explain a set of manifestations in terms of initial states. Such a process has been captured in the framework of BPNs by [16] as a reachability problem based on the corresponding net model in which each state of the causal model is represented by a place in the net, and cause-effect relationships among states are represented by transitions between the corresponding places.

In a case where the system to be diagnosed is physically distributed and large, it is often infeasible to maintain a global model of the whole system. Instead, several spatially distributed local models have to be used, each of which corresponds to a particular subsystem. The diagnostic system is viewed as a multi-agent system that reflects the network structure of the system to be diagnosed where each diagnostic agent A_i is associated to the corresponding subsystem S_i . As we have noted, we consider the system model as a distributed set of interacting BPNs. Each net models the causal behavior of the corresponding subsystem and interactions among subsystems are captured through tokens that may pass (either observably or unobservably) from one BPN to another via common bordered places. In other words, the whole diagnostic problem DP is partitioned into a set of local dependent ones, each of which corresponds to a particular area of the overall system. More formally, we consider the problem

description as follows: $DP = \cup_{i=1}^n DP_i$ with n subsystems. Each $DP_i = (N_i, P_i^{In}, P_i^{Out}, \langle P_i^+, P_i^- \rangle)$

where:

- $N_i = \langle P_i, T_i, F_i \rangle$ is a safe BPN model representing the causal behavior of a subsystem S_i ;
- P_i^{In}, P_i^{Out} are sets of places from P_i denoting bordered places of S_i within other subsystems. They correspond respectively to places modelling inputs to S_i from its neighboring subsystems and its outputs to adjacent ones S_j ;
- and $\langle P_i^+, P_i^- \rangle$ are places that represent observable manifestations of the subsystem S_i . P_i^+ corresponds to manifestations that need to be entailed by a diagnosis, whereas P_i^- is the set of manifestations that are known to be absent in the case under examination and thus they are used for local consistency checking. See [7] for more details about such a classification.

Thus, the set of behavioral Petri nets $\{N_i \mid i = 1 \dots n\}$ can be viewed as a distribution of a global

net model $N = \langle P, T, F \rangle = \cup_{i=1}^n N_i$ such that:

$$1. \quad P = \cup_{i=1}^n P_i, \text{ and } \forall i \rightarrow \exists j \text{ s.t. } P_i \cap P_j = P_{ij} \neq \emptyset, P_{ij} \subseteq P_i^{In} \cap P_j^{Out};$$

$$2. \quad T = \cup_{i=1}^n T_i, \text{ and } \forall i \neq j \Rightarrow T_i \cap T_j = \emptyset;$$

$$3. \quad P_i^{In} = \{p \mid (p^\bullet \in T_i) \wedge (p^\bullet \notin T_j)\};$$

$$4. \quad P_i^{Out} = \{p \mid (p^\bullet \in T_i) \wedge (p^\bullet \notin T_j)\};$$

$$5. \quad P^+ = \cup_{i=1}^n P_i^+, P^- = \cup_{i=1}^n P_i^-.$$

It is to be noted that for each BPN model, there exists in addition to P_i^{In} a set of source places modelling initial states of the corresponding causal model in terms of which diagnoses are to be given.

As an example, let us consider a system S composed of two interacting subsystems S_1 and S_2 . The model of each subsystem is described by a BPN representing its causal model. Figure 1 gives the graphical representation of the corresponding models. Dotted circles, labelled x , y , and z , represent the common places that are used to model interactions between the two subsystems. x and z model the fact that tokens can pass from BPN_2 to BPN_1 ; while y models the inverse direction. The models are adapted from an example given in [14] which is used to represent a partial fault centralized model of a car engine. BPN_1 is characterized by the entities *pist_ring_state(worn)*, *pist_state(worn)* and *oil_sump_state(worn)* as local initial states of the described causal model, and *ex_smoke(black)*, *oil_light(red)* and *accel_resp(del)* as local manifestations. Similarly, for BPN_2 , three local initial states are considered, they are modelled by places *road_cond(poor)*, *ground_clear(low)* and *spark_plug_mealeage(high)*, and two local manifestations *hole_oil_sump(yes)* and *temp_ind(red)*. Transitions of each net model the cause-effect relationships among the corresponding entities; for example in BPN_1 , transition t_1 models the fact that an "increased oil consumption" (modelled by place *oil_cons(incr)*) is caused by either a "worn state of piston rings" (modelled by place *pist_ring_state(worn)*) or a "worn state of pistons" (modelled by place *pist_state(worn)*). In our discussion, the meaning of the different modelled entities is irrelevant, since our aim is to show how to implement diagnostic inference reasoning by analyzing such net models.

FIGURE 1: Example of interacting BPNs.

3.2. Local diagnosis by analyzing P-invariants

Since the observable findings are considered as measurable partial states in causal models and they are modelled by a set of sink places in BPNs, a solution to a given diagnostic problem consists to explain a set of findings in terms of source states that represent initial perturbations leading the system to such a misbehavior. In Petri nets notations, the problem consists to identify an initial marking μ^{ini} in which only source places are marked that entails the marking μ^{Obs} corresponding to the observed findings. More formally, each agent A_i should calculate an initial marking μ_i^{ini} from the observation marking μ_i^{Obs} :

$$\mu_i^{ini} = \{\mu \mid \mu(p) \neq 0 \Rightarrow \bullet p = 2 \wedge \mu_i^{Obs} \in [\mu] \wedge \mu_i^{Obs} \text{ covers } P_i^+ \text{ and zero-covers } P_i^-\} \quad (1)$$

Local diagnoses are obtained by projecting the calculated initial markings on the source places modelling local initial states of the corresponding causal model as well as bordered places used as inputs to such a model from the neighboring ones. In fact, the marking of bordered places will be used later for refining the sets of local diagnoses.

$$\Delta_i = \{\Pi_{P_{is}}(\mu) \mid \mu \in \mu_i^{ini}\} \quad (2)$$

where $\Pi_{P_{is}}$ denotes the projection on $P_{is} = \{p \mid \bullet p = 2\}$.

In order to identify such initial marking, we need to go back from the places modelling the observed manifestations to source places by firing transitions in a backward fashion. This has been the preserve of an approach (BW-analysis) presented in [3] that exploits the backward reachability graphs as a diagnostic reasoning mechanism. Thus, the major shortcoming of the mentioned approach is the state space explosion problem especially when BPNs models become large. Another alternative to obtain the set μ_i^{ini} is to exploit structural properties of net models. As we have outlined, the aim of this paper is to concentrate on invariant analysis to realize diagnostic inference procedures rather than reachability graphs. In particular, we will concentrate on how to generate initial markings satisfying the conditions of Eq.(1) from a set of P-invariant supports. By definition, P-invariants of a net $N = \langle P, T, F \rangle$ correspond to T-invariants of its dual net $N_D = \langle T, P, F \rangle$. The following lemma has been proved in [11, 13]:

Lemma 1. Let $N = \langle P, T, F \rangle$ be a Petri net such that $\forall t \in T \mid t^* \leq 1$ and $t \in T$ be a sink transition; there exists a T-invariant X of N such that $X(t) \neq 0$ iff t is fireable from the empty marking.

This means that in N there are source transitions firing from the empty marking, eventually leading to the firing of t . Consider now the dual net of N , $\forall t \in T \mid t^* \leq 1$ becomes $\forall p \in P \mid p^* \leq 1$, which is guaranteed by axiom 1 of definition 3.1, and the previous lemma can be translated as follows:

Proposition 1. Let $N = \langle P, T, F \rangle$ be a Petri net such that $\forall p \in P \mid p^* \leq 1$, and $p \in P$ be a sink place; there exists a P-invariant Y of N such that $Y(p) \neq 0$ iff p can be marked by firing a sequence of transitions from an initial marking μ in which $\mu(p) \neq 0 \Rightarrow \bullet p = 2$.

This proposition means that the supports of the P-invariants of a net N modelling a causal behavior of a system characterize the diagnostic solutions that explain a set of manifestations. In fact, [15] proposes an algorithm based on analyzing such supports for the centralized causal model-based diagnosis on BPNs. Before applying the algorithm, [15] requires to transform, via an $\wedge$ -fusion operation, the BPN model to another equivalent net model in which places that are "And-ed" in the original BPN are collapsed into a single place representing their conjunction. More formally:

$\forall t \in T_N$: if $\bullet t = \{p_1, \dots, p_k\} (k > 1)$ then substitute in P the set $\{p_1, \dots, p_k\}$ with the place $p_{1,k}$ such that

$\bullet p_{1,k} = \cup_{i=1}^k \bullet p_i$ and $p_{1,k}^\bullet = \{\emptyset\}$. It is to be noted that such a transformation is needed only for

getting a right interpretation of P-invariants; and that even if the resulting net is no longer a BPN, it encodes the same kind of knowledge of the original BPN.

The algorithm can now be sketched as follows: after having calculating the minimal supports of P-invariants of the $\wedge$ -fusion transform of the net model, those leading to mark places in P (i.e. $\{\sigma \mid p \in P \wedge p \in \sigma\}$) are eliminated by taking into account the fact that if τ, τ' are two sets of source places such that $\tau \subseteq \tau'$, if the marking of τ leads to mark $p \in P$ then the marking of τ' leads also to mark p . Then, the algorithm considers the coverability of P^+ ; for each $p \in P^+$, it builds from remaining supports the list of source places (i.e. places denoting initial states of the causal model) supporting p (i.e. contained in a P-invariant support containing p). Final diagnoses are obtained by combining such lists.

In order to show how these can be done, let us return to our example depicted in Figure 1. Suppose that A_1 receives the observation *oil_light(green)* and *accel_resp(del)* (i.e. “oil light is green” and there is a “delay in the acceleration response”) from S_1 and A_2 observes *hole_oil_sump(no)* and *temp_ind(red)* (i.e. there is “no hole in the oil sump” and the “indicator of temperature is red”) from S_2 . Let us also suppose that all abnormal observations have to be covered, then:

$$P_1^+ = \{accel_resp(del)\}, P_1^- = \{oil_light(red)\};$$

$$P_2^+ = \{temp_ind(red)\}, P_2^- = \{hole_oil_sump(yes)\}.$$

It is to be noted that in the observation of S_1 , the value of *ex_smoke(black)* is not specified, which imply that we have an incomplete knowledge about the behavior of S_1 ; and that the $\wedge$ -fusion transformation results in replacing places *oil_sump_state(worn)* and p_5 by place $p_{oil_sump_state(worn),p5}$ in BPN_1 .

The set of minimal supports of P-invariants of S_1 that are computed by A_1 are the following:

$$\sigma_1 = \{pist_ring_state(worn), oil_cons(incr), p_1, ex_smoke(black)\};$$

$$\sigma_2 = \{pist_state(worn), oil_cons(incr), p_1, ex_smoke(black)\};$$

$$\sigma_3 = \{pist_ring_state(worn), oil_cons(incr), p_2, lack_of_oil(sev), p_4, y\};$$

$$\sigma_4 = \{pist_state(worn), oil_cons(incr), p_2, lack_of_oil(sev), p_4, y\};$$

$$\sigma_5 = \{pist_ring_state(worn), oil_cons(incr), p_2, lack_of_oil(sev), p_3, p_6, accel_resp(del)\};$$

$$\sigma_6 = \{pist_state(worn), oil_cons(incr), p_2, lack_of_oil(sev), p_3, p_6, accel_resp(del)\};$$

$$\sigma_7 = \{pist_ring_state(worn), oil_cons(incr), p_2, lack_of_oil(sev), p_3, p_7, p_{oil_sump_state(worn),p5}, oil_light(red)\};$$

$$\sigma_8 = \{pist_state(worn), oil_cons(incr), p_2, lack_of_oil(sev), p_3, p_{oil_sump_state(worn),p5}, p_7, oil_light(red)\};$$

$$\sigma_9 = \{x, lack_of_oil(sev), p_4, y\};$$

$$\sigma_{10} = \{x, lack_of_oil(sev), p_3, p_6, accel_resp(del)\};$$

$$\sigma_{11} = \{x, lack_of_oil(sev), p_3, p_{oil_sump_state(worn),p5}, p_7, oil_light(red)\};$$

$$\sigma_{12} = \{z, accel_resp(del)\}.$$

Since place $oil_light(red) \in P_1^-$, any support predicting the marking of such a place will be eliminated; hence, σ_7 , σ_8 , σ_{10} and σ_{11} will be discarded because they contain $oil_light(red)$. Moreover, any support that contain one of the places $pist_ring_state(worn)$, $pist_state(worn)$ and x will be eliminated, since the marking of one of these places conducts to mark $oil_light(red)$ according to the discarded supports. Thus, the only support that survive is σ_{12} which explain locally the marking of $accel_resp(del)$ by the arrival of a token in place z . In other words, the observed local findings are explained by an outside failure propagated to S_1 through z .

Similarly, for A_2 seven minimal supports of P-invariants of S_2 are generated:

$$\partial_1 = \{road_cond(poor), p_8, x\};$$

$$\partial_2 = \{road_cond(poor), p_8, p_9, hole_oil_sump(yes)\};$$

$$\partial_3 = \{groud_clear(low), p_{10}, hole_oil_sump(yes)\};$$

$$\partial_4 = \{spark_plug_mileage(high), spark_plugs(used_up), p_{11}, spark_ign(irr), temp_ind(red)\};$$

$$\partial_5 = \{spark_plug_mileage(high), spark_plugs(used_up), p_{11}, z\};$$

$$\partial_6 = \{y, p_{11}, spark_ign(irr), temp_ind(red)\};$$

$$\partial_7 = \{y, p_{11}, z\}.$$

Supports ∂_2 , ∂_3 are discarded because they contain $hole_oil_sump(yes)$ which belongs to P_2^- . Since ∂_1 contains $road_cond(poor)$ which is contained with $hole_oil_sump(yes)$ in the same support, it will be also eliminated. The remaining supports will be used by A_2 to generate diagnoses that explain locally the marking of $temp_ind(red)$.

As a result, the observed symptom is explained by one of the following two local diagnoses:

$$\Delta_2^1 = \{spark_plug_mileage(high)\};$$

$$\Delta_2^2 = \{y\}.$$

The first one means that $temp_ind(red)$ is caused by a local failure; while the latter signifies that there is a failure in the neighboring affecting the behavior of S_2 through y .

Thus, the general diagnostic algorithm based on the P-invariant analysis used by each agent A_i to compute local solutions can be sketched informally as follows:

Algorithm 1: A_i 's Local Computation

Input: a local diagnostic problem in terms of Petri nets $DP_i = (N_i, P_i^{In}, P_i^{Out}, P_i^+, P_i^-)$;

Output: a set of local diagnoses in terms of minimal supports;

begin

 compute the minimal supports of the P-invariants for N_i ;

 let L be the list of such minimal supports;

for each $p \in P_i^-$ **do**

for each support $\sigma \mid p \in \sigma$ **do**

$\tau \leftarrow \{p' \in \sigma \mid \bullet p' = 2\}$;

 delete from L all supports where τ occurs;

end for

end for

end.

Up to now, each agent works independently without taking into account knowledge of its neighboring. To refine the obtained local diagnoses and to ensure that they are globally consistent, agents should exchange information about their common places used to explain locally the received observations.

It is to be noted that in the above algorithm, N_i is considered as the $\wedge$ -fusion transform of the original BPN model, and that the local diagnoses can be obtained by combining sources places belonging to the remaining supports. We choose to not combine such supports during this first step of pruning because they are needed by agents during the consistency checking step.

3. PROTOCOL FOR DISTRIBUTED DIAGNOSIS

Once local diagnoses are obtained, agents begin to communicate among them for guaranteeing that such diagnoses are consistent and recover global ones computed by a centralized agent that knows the system's global model and receives all manifestation signalizations. In particular, each agent asks their neighboring ones for the required set of its input places that necessitate to be marked (i.e. that necessitate to receive tokens from neighboring net models) for explaining locally the received observation. According to Eq.(1), such places can be obtained by choosing the marked places from the result of projecting the set μ_i^{ini} on P_i^{In} (i.e. $\{p \mid \mu(p) \neq 0 \wedge \mu \in \Pi_{P_i^{In}}(\mu_i^{ini})\}$) which is equivalent to choose the marked input places from Δ_i . Thus, agent A_i will send to each of its neighboring a message indicating what input places are used to explain the local observation for each of its obtained local diagnoses Δ_i .

(3)

$$Msg_{i \rightarrow j} = \{ p \mid p \in P_i^{In} \cap P_{ij} \wedge \mu(p) \neq 0 \wedge \mu \in \Delta_i \}$$

After receiving such a message, agent A_j will respond to A_i by either a positive reply or negative one; depending on the fact that at least one its local diagnoses is consistent with the received message. In order to check such a consistency, [3] proposes to construct from each of A_j 's local diagnoses the forward reachability graph to see if tokens requested by A_i are produced by the net model of A_j through its corresponding output places when supposing such a diagnosis. Thus, besides the state space explosion during local computations, the communication protocol suffers of the same problem in refining the set of local diagnoses of different agents because it exploits several reachability graphs as a basis for consistency checking for each exchanged message.

Since the diagnosis method described in the previous section is based on the set of minimal supports of P-invariants and not directly on the net models, we wish to exploit such supports in order to checking the required consistency among local diagnoses of different agents; and hence avoiding the problem of state space explosion characterizing reachability graphs. In this spirit, when agent A_j receives a message $Msg_{i \rightarrow j}$, it will examine its remaining set of supports to test if places contained in the received message belong to at least one of such supports. If so, it will respond by a positive reply indicating that diagnoses of A_j are consistent with the diagnosis of A_i from which $Msg_{i \rightarrow j}$ has been generated. Otherwise, A_j 's local diagnoses does not support the diagnosis Δ_i of A_i ; and hence, a negative reply should be sent. Consequently, when agent A_i receives a negative reply to a message $Msg_{i \rightarrow j}$ from an adjacent agent A_j , it will discard its local diagnosis Δ_i from which $Msg_{i \rightarrow j}$ has been generated, since it does not conform to diagnoses of the neighboring even if it explains locally the observed misbehavior. Moreover, it may be that the discarded diagnosis has been used to validate consistency between diagnoses of A_i and those of another adjacent agent A_k (i.e. $k \neq j$). As a result, some of A_k 's diagnoses should be eliminated since they become inconsistent with diagnoses of A_i ; and thus the communication between agents will be initiated again. Accordingly, the consistency checking will terminate after some communication rounds when a stability condition in terms of local diagnoses of all agents is achieved.

We can now extend our previous algorithm of local computation to account for message exchange between the different agents. For simplicity purposes, the algorithm will be presented in three parts: the first (Algorithm 2) extends Algorithm 1 to generate messages that will be sent to the neighboring and to eliminate local solutions for which it receives a negative reply. The second (Algorithm 3) treats the case of a message reception by a neighboring agent A_j . Finally, Algorithm 4 generates local diagnoses that are globally consistent and may be viewed as the

last task of each diagnostic agent to be executed after the completion of the communication protocol.

Algorithm 2: *A_i's Local computation with communication*

Input: the list L of minimal supports pruned by Algorithm 1;

Output: a list of minimal supports that are consistent with those of the neighboring;

begin

$mess_list \leftarrow 2$;

for each $\sigma \mid p \in \sigma \cap P_i^{In}$ **do**

if $(Msg_{i \rightarrow j}, \sigma) \notin mess_list \mid Msg_{i \rightarrow j} = p$ **then**

$Msg_{i \rightarrow j} \leftarrow \{p\}$ where j is such that $p \in P_j^{Out}$;

Send $Msg_{i \rightarrow j}$ to A_j ;

$mess_list \leftarrow mess_list \cup \{(Msg_{i \rightarrow j}, \sigma)\}$;

end if

end for

while $mess_list \neq 2$ **do**

receive(rep);

let rep be the reply corresponding to $(Msg_{i \rightarrow j}, \sigma)$;

if $rep = negative$ **then**

delete from L all supports where $Msg_{i \rightarrow j}$ occurs;

$mess_list \leftarrow mess_list \setminus \{Msg_{i \rightarrow j}, \sigma\}$;

end if

end while

end.

Algorithm 3: *Treatment of a received message*

Input: a received message $Msg_{i \rightarrow j}$;

Output: a positive or a negative response;

begin

if $\exists \sigma \in L \mid Msg_{i \rightarrow j} \in \sigma$ **then**

reply to $Msg_{i \rightarrow j}$ by a positive response;

else

reply to $Msg_{i \rightarrow j}$ by a negative response;

end if

end.

Algorithm 4: *Diagnoses generation*

Input: the list L of minimal supports pruned by Algorithm 1;

Output: local diagnoses that are globally consistent;

begin

$X \leftarrow 2$;

for each $p \in P^*$ **do**

if $\exists \sigma \in L \mid p \in \sigma$ **then** no solution;

$H' \leftarrow 2$;

for each $\sigma \in L \mid p \in \sigma$ **do**

$H' \leftarrow H' \cup \{p \mid p \in \sigma \wedge \#p' = 2\}$;

end for

$X \leftarrow X \cup \{H'\}$;

end for

combine elements in X to produce Δ ;

end.

In order to show how agents update their diagnoses, let us return again to our example. After local computations, each agent sends to the other a message that contains the required places used as its inputs for each of its local diagnoses. Thus, agent A_1 sends to A_2 the message $Msg_{1 \rightarrow 2} = \{z\}$ (Notice that z has been identified as the only local diagnosis for the observation of A_1). Similarly, A_2 sends $Msg_{2 \rightarrow 1} = \{y\}$ to agent A_1 . Then, each agent tries to test if the received common place belongs to its remaining set of supports. Thus, A_1 concludes that there is an inconsistency between its diagnosis and knowledge of A_2 ; because its list of supports contains only σ_{12} and $y \notin \sigma_{12}$. Consequently, it responds by a negative reply; and accordingly, A_2 will discard ∂_6 and ∂_7 . Independently, when A_2 receives $Msg_{1 \rightarrow 2} = \{z\}$, it concludes immediately that z belongs to its supports ∂_5 and ∂_7 (i.e. z belongs to the remaining supports even after eliminating ∂_7), and it responds by a positive reply. Since this first round of communication has resulted in a modification of A_2 's diagnoses; agents will restart the communication again. During the second communication round, local diagnoses remain unchanged; and the process terminates with σ_{12} as the only support for A_1 and $\{\partial_4, \partial_5\}$ for A_2 .

4. RELATED WORKS

Over the last few years, big efforts have been spent to distributed model-based diagnosis in the sense that the network structure of the system to be diagnosed is reflected by a similar structure of local agents, where an agent is a piece of software that has the properties of autonomy and interaction. The agents may interact by exchanging information through the communication channels. Each agent may observe uniquely the associated subsystem. In [17], the authors propose a multi-agent diagnostic approach where the attention was focused on logical definitions and on declarative characterizations by extending the centralized logical definition proposed in [7]. For diagnosis based on causal models, we mention the work presented by [5] in which the authors concentrate on how to partition a global model of the system to be diagnosed to a set of local models (regions in the terminology of [5]), and not on how to calculate diagnoses. They propose a framework ensuring that the obtained local regions are maximally independent and their common elements are minimal. The maximal independency level allows diagnosis to be performed locally, without the need of communicating with the other models, as

long as the border with them is healthy; and the minimum number of borders ensures that the communication among diagnostic agents is minimal.

In the context of Petri nets, there has been other recent works for the diagnosis of discrete event systems. We mention in this regard the works presented in [8, 9] in which the authors extend the Diagnoser approach based on automata models [10, 18] to that of distributed Petri net models. In such works, transitions of the net models are labelled by symptoms ("fault events" in the terminology of these works) in terms of which the diagnoses have to be given. The work presented in this paper differ from these and others in that it models fault events as well as system conditions (i.e., conditions denoting states of the modelled system) by places of the net model. Moreover, transitions of the model express causality links among such states. It is to be noted that such modelization has been adopted by several works [1, 2, 4, 11, 14, 15] that deal with the problem of centralized diagnosis of Petri nets. The diagnosis mechanisms used in these works exploit an analysis of reachable markings through reachability graph or invariant techniques. The work presented in [11] adds probabilities to Petri nets for ranking among diagnoses.

5. CONCLUSION

The approach presented in this paper uses Petri nets as a modelling and reasoning tool. It is a distributed multi-agents approach where interactions among subsystems are captured by tokens that may pass from one net model to another through common places. The reasoning scheme exploits a structural analysis of the net models to avoid the combinatorial explosion of the state space characterizing reachability graphs analysis. After the completion of local calculations, each agent exchanges with its neighboring a limited information about the status of common places to reject the computed local diagnoses that are not consistent with those of the neighboring. Many issues remain to be investigated. Among those we mention: the possibility of introducing precedence relationships among symptom signalizations as in [2, 4] since the defined approach considers that all observations are given at a single time point which is restrictive in many domains; and the treatment of symptoms masking which is a rule rather than an exception in some applications such as fault management in communication networks.

REFERENCES

- [1] C. Anglano and L. Portinale, B-W Analysis: A Backward Reachability Analysis for Diagnostic Problem Solving Suitable to Parallel Implementation, In *Application and Theory of Petri Nets, LNCS(815)*, pp. 39-58, 1994.
- [2] H. Bennoui and A. Chaoui, Reachability Analysis based on BPNs to Model-based Diagnosis with Relationships among Symptoms, *International Review on Computers and Software (I.RE.CO.S)*, vol. 4(2), pp. 350-356, 2007.
- [3] H. Bennoui, A. Chaoui and K. Barkaoui, Distributed Causal Model-based Diagnosis based on Interacting Behavioral Petri Nets, *Proceedings of the 8th International Symposium on Parallel and Distributed Computing*, pp. 99-106, 2009.
- [4] H. Bennoui, M. Maouche and M. Bettaz, Une approche pour le diagnostic des pannes dans les réseaux, *Proceedings of the Colloque Francophone sur l'Ingénierie des Protocoles (CFIP'99)*, Hermès Edition, pp. 433-445, Apr. 1999.
- [5] C.D. Bocaniala and J. Sa da Costa, Novel framework for using causal models in distributed fault diagnosis, *Proceedings of the Workshop on Advances in Control and Diagnosis*, Karlsruhe, Germany, pp. 142-147, 2004.
- [6] L. Console and P. Torasso, Hypothetical reasoning in causal models, *International Journal of Intelligent Systems*, vol. 5, pp. 83-124, 1990.
- [7] L. Console and P. Torasso, A Spectrum of Logical Definitions of Model-Based Diagnosis, *Computational Intelligence*, vol. 7(3), pp. 133-141, 1991.
- [8] S. Genç and S. Lafortune, Distributed Diagnosis of Place-Bordered Petri Nets, *IEEE TASE*, vol. 4(2), pp. 206-219, 2007.
- [9] G. Jiroveanu and R. K. Boel, Petri Net model-based distributed diagnosis for large interacting systems, *Proceedings of the DX'05*, pp. 25-30, 2005.
- [10] S. Lafortune, D. Teneketzis, M. Sampath, R. Sengupta and K. Sinnamohideen, Failure Diagnosis of Dynamic Systems: An Approach based on Discrete Event Systems, *Proceedings of the American Control Conf.*, pp. 2058-2071, 2001.
- [11] K. Lautenbach, S. Philippi and A. Pinl, Bayesian Networks and Petri Nets, In E. Schnieder (Hrsg), *Entwurf komplexer Automatisierungssysteme*, EKA 2006, 9. Fachtagung, Braunschweig, 2006.
- [12] T. Murata, Petri Nets: Properties, Analysis and Applications, *Proceedings of the IEEE* 77(4), pp. 541-580, 1989.

- [13] G. Peterka and T. Murata, Proof Procedure and Answer Extraction in Petri Net Model of Logic Programs, *IEEE TSE* 15(2), pp. 209-217, 1989.
- [14] L. Portinale, Exploiting T-invariant Analysis in Diagnostic Reasoning on a Petri Net Model, *In Application and Theory of Petri Nets, LNCS* 691(1), pp. 339-356, 1993.
- [15] L. Portinale, Improving Model-Based Diagnosis through Algebraic Analysis: the Petri Net Challenge, *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI'96)*, pp. 952-958, 1996.
- [16] L. Portinale, Behavioral Petri Nets: A Model for Diagnostic Knowledge Representation and Reasoning, *IEEE TSMC--Part B: Cybernetics*, vol. 27(2), pp. 184-195, 1997.
- [17] N. Roos, A. T. Teije, A. Bos and C. Witteveen, Multi-Agent Diagnosis with spatially distributed knowledge, *Proceedings of the BNAIC'02*, pp. 275-282, 2002.
- [18] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen and D. Teneketzis, Diagnosability of Discrete Event Systems, *IEEE TAC*, vol. 40(9), pp. 1555-1575, 1995.