Diagnosticabilité de Réseaux de Petri Labellisés basée sur les explications minimales et les T-semiflots

Ben Li[1], Manel Khlif-Bouassida[1] and Armand Toguyéni[1]

Univ. Lille Nord de France, F-59000, Lille, France
École Centrale de Lille, CRIStAL, UMR 9189
59650 Villeneuve d'Ascq, France
{ben.li, manel.khlif-bouassida, armand.toguyeni}@ec-lille.fr

Abstract

RESUME: Dans cet article, une approche est développée pour l'analyse de la diagnosticabilité des systèmes à événements discrets (SEDs) modélisés par des Réseaux de Petri Labellisés (RdP-L). L'objectif de ce travail est de combattre l'explosion combinatoire dans l'analyse de la diagnosticabilité à partir d'un modèle RdP-L. Notre approche étend l'analyse à la volée proposée dans [13]. Elle propose des améliorations basées sur les concepts d'explications minimales et de T-semiflots. Basée sur la recherche en profondeur d'abord, cette approche améliore l'efficacité de l'analyse de diagnosticabilité en construisant l'espace d'état de manière compacte en utilisant le concept d'explications minimales. De plus, cette approche définit des priorités dans le parcours des branches du graphe d'accessibilité à l'aide de T-semiflots. Notre approche a pour objectif de trouver plus rapidement l'existence de cycles indéterminés et de construire l'espace d'état de manière compacte, afin de réduire le coût de mémoire de l'analyse de diagnosticabilité.

ABSTRACT: In this paper, an approach is developed for diagnosability analysis of discrete event systems (DESs) modeled by labeled Petri nets (LPNs). The objective of this work is to solve the combinatorial explosion problem of the diagnosability analysis of PN model. This study extends from the on-the-fly diagnosis approach proposed in [13]. It proposes some improvements based on the notions of T-invariants and minimal explanations. Based on depth-first search, this approach improves the efficiency of the diagnosability analysis by constructing the state space in a compact manner using minimal explanations. Moreover, this approach defines priorities in the investigation of reachability graph branches with the help of T-invariants. Our approach intends to find more quickly the existence of indeterminate cycles and construct the state space in a compact manner, so as to reduce the memory cost for diagnosability analysis.

MOT-CLES: Diagnostic des fautes, Systèmes à Événements Discrets, Réseaux de Petri labellisés, Analyse de la diagnosticabilité, Analyse à la volée, T-semiflots, Explications minimales

KEYWORDS: Fault diagnosis, Discrete event systems, Labeled Petri nets, Diagnosability analysis, On-the-fly diagnosis, T-invariants, Minimal explanations

Contents

1 Introduction

La diagnosticabilité est la capacité à détecter et à identifier toute faute dans un délai fini après son occurrence. Un des verrous de l'analyse de la diagnosticabilité est le problème de l'explosion combinatoire, c'est-à-dire, lorsqu'il s'agit d'un système automatisé complexe, un énorme espace d'états doit être construit.

Dans [13], l'analyse à la volée a été proposée comme solution à l'explosion combinatoire. Cette approche ne construit généralement pas *a priori* la totalité de l'espace d'états, surtout lorsque le système est non diagnosticable. Le FM-graph (Fault Marking Graph, en français graphe de marquages labellisés par les fautes) correspond à l'espace d'états et le FM-set tree (Fault Marking set tree, en français arbre des ensembles de marquages labellisés par les fautes) est le modèle à partir duquel est réalisé le diagnostic. Ces deux modèles sont construits à la volée et en parallèle. La construction est arrêtée dès qu'une condition d'arrêt est trouvée. Le but de cet article est de proposer des voies permettant de minimiser l'espace d'états construit, pour décider de la diagnosticabilité d'un système. Nous y proposons deux axes d'amélioration : premièrement, les explications minimales sont intégrées en construisant l'espace d'états comme dans [3, 9] afin de réduire davantage le coût de mémoire ; deuxièmement, la notion de T-semiflots est utilisée pour définir des priorités pour le parcours des branches afin de trouver le plus vite possible un cycle indéterminé existant.

2 État de l'art

Pour les systèmes à événements discrets (SEDs), la notion de diagnosticabilité fut introduite par Sampath [14] dans l'approche du "diagnostiqueur". Le diagnostiqueur de Sampath est un automate à états finis construit *a priori* et permettant de faire l'analyse hors ligne de la diagnosticabilité. Le diagnostiqueur est également exécuté en ligne pour diagnostiquer les fautes du système. Afin de réduire la complexité de l'analyse de la diagnosticabilité pour l'approche diagnostiqueur, les approches twin-plant et vérificateur [8, 16] ont été proposées par la suite. La complexité est polynomiale avec ces nouvelles structures. Par contre, elles ne résolvent pas le problème d'explosion combinatoire. En particulier, en pratique, beaucoup de temps et de ressources mémoire sont consommés pour analyser la diagnosticabilité de systèmes complexes .

Ensuite, des approches basées sur les réseaux de Petri (RdP) ont été développées. Les RdPs modélisent plus simplement les comportements parallèles et les synchronisations dans un système. Dans [15], une condition suffisante de la diagnosticabilité a été proposée en vérifiant les T-semiflots (*T-invariants* en anglais) pour analyser la diagnosticabilité d'un RdP *vivant* et *sauf*. Dans [3], l'analyse de MBRG (Modified Basis Reachability Graph, en français graphe d'accessibilité de base)/BRD (Basis Reachability Diagnoser, en français diagnostiqueur d'accessibilité de base) a été élaborée. MBRG/BRD fournissent une manière compacte pour la construction de l'espace d'états afin de mieux combattre l'explosion combinatoire. Dans [1], une approche de vérification de la k-diagnosticabilité a été proposée, utilisant la programmation linéaire. Dans [2], une structure appelée le VN (Verifier Net, en français Vérificateur de RdP) a

été développée pour analyser la diagnosticabilité pour les RdPs bornés et non bornés, et aussi la k-diagnosticabilité des RdPs non bornés. Cette approche réduit la complexité de l'analyse de la diagnosticabilité des RdPs bornés. Dans [13], l'analyse à la volée a été proposée, pour l'analyse de la diagnosticabilité et de la k-diagnosticabilité de RdPs vivants et bornés ; seulement une partie de l'espace d'états est construite à la volée (au lieu de développer la totalité de l'espace d'états *a priori*), afin de combattre l'explosion combinatoire.

3 Préliminaires

3.1 Réseaux de Petri labellisés

Un Réseau de Petri (RdP) est une notation mathématique et graphique pour la modélisation des SEDs. Un RdP est un 4-uplet $N = (P, T, Pre, Post)$. P est un ensemble fini de places ; T est un ensemble fini de transitions ; Pre est la matrice d'incidence avant ; $Post$ est la matrice d'incidence arrière. Un marquage est un vecteur $\mathrm{M} \in \mathbb{N}^{|P|}$ qui associe un entier non négatif à chaque place. Le marquage d'un RdP représente l'état du système. $M(p)$ désigne le nombre de jetons dans la place p et M_0 est le marquage initial de N. (N, M_0) est un RdP marqué avec le marquage initial M_0. La matrice d'incidence est $C = Post - Pre$.

Une transition t est franchissable à partir de M ssi $M \geq Pre(\cdot, t)$, noté par $M\ [\ t >$. $M\ [\ \sigma >$ indique que la séquence de transitions σ est franchissable à partir de M. Pour une séquence donnée, $\sigma \in T^*,\ \pi : T^* \to \mathbb{N}^{|T|}$ est la fonction qui associe un vecteur $\overrightarrow{\Omega} \in \mathbb{N}^{|T|}$ à σ. $\overrightarrow{\Omega} = \pi(\sigma)$ est le vecteur de tir de σ, et $\overrightarrow{\Omega}(t) = k$ indique que la transition t est contenue k fois dans σ. Le marquage atteint M' est calculé par $M' = M + C \cdot \overrightarrow{\Omega}$. Un marquage M est accessible dans (N, M_0) ssi il existe une séquence σ tel que $M_0\ [\ \sigma > M$. L'ensemble de tous les marquages accessibles à partir de M_0 est noté par $R(N, M_0)$ qui est l'ensemble des états accessibles de (N, M_0). La $i^{ème}$ transition dans σ est notée σ^i.

Un RdP est *vivant* si, à partir de tous les marquage dans $R(N, M_0)$, il est toujours possible de trouver une séquence permettant de franchir n'importe quelle transition du RdP. Un RdP est *borné* s'il existe un nombre positif m tel que $\forall M \in R(N, M_0),\ M(p) \leq m$.

Un Réseau de Petri labellisé (RdP-L) est un 4-uplet $N_L = (N, M_0, \Sigma, \mathcal{L})$. (N, M_0) est un RdP marqué ; Σ est un ensemble fini des événements. $\mathcal{L}\colon T \to \Sigma$ est la fonction de labellisation des transitions qui affecte un label à chaque transition. L'ensemble des événements est $\Sigma = \Sigma_o \uplus \{\varepsilon\}$. Σ_o est l'ensemble des événements observables qui sont associés à des transitions observables. Toutes les transitions non observables ont un label ε. Le même label peut être partagée par des transitions différentes. La fonction de labellisation des transitions peut être étendue à $\mathcal{L}\colon T^* \to \Sigma^*$ et la fonction inverse est $\mathcal{L}^{-1}(s) = \{\sigma \in T^* \mid \mathcal{L}(\sigma) = s\}$, où $s = s_1 s_2 \cdots s_n$ est la concaténation de $s_1,\ s_2, \cdots,\ s_n$ tels que $s_1,\ s_2, \cdots,\ s_n \in \Sigma^*$.

Les T-semiflots (*T-invariants* en anglais) sont des solutions entières strictement positives de l'équation homogène: $C \cdot \overrightarrow{\Omega} = 0$. $\overrightarrow{\Omega}$ est le vecteur de tir comme cela a été défini avant. Un T-semiflot $\overrightarrow{\Omega}_{min}$ est un T-semiflot minimal, si $\forall \overrightarrow{\Omega} :\ C \cdot \overrightarrow{\Omega} = 0 \Rightarrow (sub(\overrightarrow{\Omega}) \not\supseteq sub(\overrightarrow{\Omega}_{min}))$, tel que $sub(\overrightarrow{\Omega}) = \{t_i \in T |\ \overrightarrow{\Omega}(t_i) > 0\}$.

3.2 Diagnosticabilité

Dans le diagnostic des SEDs, l'ensemble des transitions est divisé en deux ensembles disjoints : $T = T_o \uplus T_u$. T_o est l'ensemble des transitions observables, et T_u est l'ensemble des transitions non observables. Les transitions fautives sont non observables. L'ensemble des transitions non observables peut être divisé en deux ensembles disjoints : $T_u = T_f \uplus T_{reg}$ où T_f est l'ensemble de toutes les transitions de faute et $T_{reg} = T_u \backslash T_f$ est l'ensemble de transitions non observables et non-fautives, également appelées transitions régulières. T_f peut encore être partitionné en k

sous-ensembles différents T_f^i tel que $i = 1, \cdots, k$ représente les différentes classes des transitions fautives. Soit C_u (resp. C_o) la restriction de la matrice d'incidence C relative à T_u (resp. T_o). De plus, $P_u : T^* \rightarrow T_u^*$ est la projection qui supprime les transitions observables dans une séquence $\sigma \in T^*$ et $P_o : T^* \rightarrow T_o^*$ est la projection qui supprime les transitions non observables de $\sigma \in T^*$. $\Sigma_T(\overrightarrow{\Omega}) = \{l \in \Sigma_o |\ t_i \in sub(\overrightarrow{\Omega}),\ \mathcal{L}(t_i) = l\}$ est l'ensemble des labels observable de $\overrightarrow{\Omega}$. Il faut remarquer qu'il peut contenir plusieurs fois le même label. Par exemple, si $\exists t_1,\ t_2$, tel que $\overrightarrow{\Omega}(t_1) = 1,\ \overrightarrow{\Omega}(t_2) = 1$ et $\mathcal{L}(t_1) = \mathcal{L}(t_2) = a$, il existera deux labels a dans $\Sigma_T(\overrightarrow{\Omega})$.

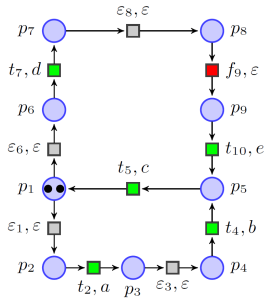

Figure 1: Un exemple de RdP-L

Exemple 1. *Un exemple de RdP-L est illustré par la Figure 1.* $T_o = \{t_2, t_4, t_5, t_7, t_{10}\}$, $T_{reg} = \{\varepsilon_1,\ \varepsilon_3,\ \varepsilon_6,\ \varepsilon_8\}$ *et* $T_f = \{f_9\}$. $a,\ b,\ c,\ d,$ *et* e *sont des événements observables tels que* $\mathcal{L}(t_2) = a,\ \mathcal{L}(t_4) = b,\ \mathcal{L}(t_5) = c,\ \mathcal{L}(t_7) = d,$ *et* $\mathcal{L}(t_{10}) = e$. ε *est le même label pour toutes les transition non observables.*

La définition de la diagnosticabilité d'un système modélisé par un RdP-L est la suivante:

Définition 1. *[3] Pour un RdP-L vivant* $N_L = (N, M_0, \Sigma, \mathcal{L})$, N_L *est diagnosticable par rapport aux classes de fautes* T_f^i, *s'il n'y a pas deux séquences* σ_1 *et* σ_2 *qui satisfont les conditions suivantes :*

1. $\forall t_f \in T_f^i,\ t_f \notin \sigma_1$;
2. $\exists t_f \in T_f^i$ *tel que* $t_f \in \sigma_2$ *et* σ_2 *est arbitrairement longue après l'occurrence de* t_f ;
3. $\mathcal{L}(\sigma_1) = \mathcal{L}(\sigma_2)$.

En d'autres termes, un RdP-L vivant et borné est diagnosticable, s'il n'existe pas de cycle indéterminé (défini formellement dans [14]), qui indique deux séquences de transitions avec la même observation de sorte que l'une contient une transition de faute et peut être arbitrairement longue après l'occurrence de la faute ; l'autre n'en contient pas. En outre, le RdP-L n'est jamais bloqué après l'occurrence des transitions fautives. D'un point de vue pratique, les séquences infinies correspondent nécessairement à des cycles dans le graphe d'accessibilité. Dans les sections suivantes, cette définition conduit à la recherche des cycles indéterminés dans le FM-set tree [13] et le BFST (Basis Fault Marking Set Tree) de notre approche.

Avant de présenter l'approche proposée par cet article nous définissons les hypothèses de base correspondant au contexte de notre étude :

1. Le RdP-L est *borné* et *vivant* (cela signifie qu'on ne s'intéresse qu'aux systèmes à fonctionnement cyclique) ;
2. Il n'existe aucun cycle de transitions non observables ;
3. Les fautes sont permanentes, c'est-à-dire que lorsqu'une faute a lieu, le système reste fautif infiniment ;
4. Le même label peut être associée à différentes transitions ;
5. La structure du RdP-L et le marquage initial sont bien connus.

4 Concepts de Base

4.1 L'analyse à la volée

L'analyse à la volée dans [13] vise à résoudre les problèmes de la diagnosticabilité, de la K-diagnosticabilité, du calcul du K minimum pour assurer la diagnosticabilité et du diagnostic en ligne. L'analyse à la volée a pour objectif d'éviter la construction entière de l'espace d'états pour analyser la diagnosticabilité afin de réduire le coût de la mémoire.

Le FM (Fault Marking, en français marquage étendu) est composé d'un marquage et d'un tag binaire par classe de fautes indiquant l'occurrence d'une faute de cette classe.

Définition 2. *Un FM relatif à une séquence $\sigma \in T^*$ et une classe de fautes T_f^i, est un vecteur $FM^i \in \mathbb{N}^{|P|+1}$:*

$$FM^i = \begin{bmatrix} marquage(FM^i) \\ faute(FM^i) \end{bmatrix}$$

où $M_0 \,[\, \sigma > marquage(FM^i)$. $faute(FM^i) = 1$ si $\exists t_f \in T_f^i, t_f \in \sigma$, sinon, $faute(FM^i) = 0$.

Soient FM et FM' deux marquage de faute, $FM \,[\, \sigma > FM'$ ssi $marquage(FM) \,[\, \sigma > marquage(FM')$; et $faute(FM') = faute(FM)$ si $\forall j, \sigma^j \notin T_f^i$, autrement, $faute(FM') = 1$.

Un FM^i-graph (défini formellement dans [13]) est considéré comme un graphe non-déterministe par rapport à la classe de fautes T_f^i. Chaque noeud correspond à un FM donné et chaque arc est associé à un événement observable. Un FM^i-graph est assimilable à un automate observateur résultant d'une ϵ-réduction du graphe des FMs du modèle [5], c'est-à-dire que les FMs obtenus juste après un événement observable sont construits dans le FM-graph. Un FM-set tree (défini formellement dans [13]) est une structure arborescente. Chaque nœud dans le FM-set tree est un FM-set (en français ensemble de marquages étendus). Le nœud-racine est le FM-set initial $x_0 = \{FM_0\}$. Les nœuds suivants sont accessible à partir du nœud précédent en utilisant la technique de l'estimation d'état. Un FM-set est comparable à un état de diagnostiqueur dans [14]. On peut lui associer un tag pour indiquer la possibilité d'occurrence d'une faute.

Définition 3. *La fonction tag: $\mathcal{X} \to \{N, F, U\}$ est définie comme suit :*

$$tag(x) = \begin{cases} N & \text{if } \forall\ FM \in x,\ faute(FM) = 0 \\ F & \text{if } \forall\ FM \in x,\ faute(FM) = 1 \\ U & \text{sinon} \end{cases}$$

Un FM-set x est normal (resp. F-certain, F-incertain) si $tag(x) = N$ (resp. F, U). Pour le FM-set x' accessible à partir de x, si $tag(x) \in \{N, U\}$, il est possible que $tag(x') \in \{N, F, U\}$; tandis que si $tag(x) = F$, alors $tag(x') = F$, car les fautes sont supposées permanentes. Ainsi, le tag F-certain est propagé à tous les FM-sets suivants.

La différence principale entre l'approche diagnostiqueur et l'analyse à la volée est que le graphe d'accessibilité et le diagnostiqueur doivent être construits *a priori*, et tous les états sont entièrement énumérés ; par contre, le FM-graph et le FM-set tree sont construits à la volée et en parallèle. Certaines conditions sont données pour arrêter l'investigation d'une branche de FM-set tree comme suit :

1. Un FM-set F-certain est généré ;
2. Un nouveau FM-set normal égal à un FM-set existant est construit ;
3. Un nouveau FM-set F-incertain égal à un FM-set existant est construit (la vérification de l'existence d'un cycle indéterminé est alors nécessaire).

Par l'analyse à la volée, il est donc possible de construire seulement une partie de l'espace d'états pour analyser la diagnosticabilité d'un RdP-L. Cette approche est efficace et a un coût mémoire moindre, en particulier pour un système non diagnosticable. Nous utilisons l'analyse à la volée pour analyser la diagnosticabilité du RdP-L de la Figure 1. La construction à la volée du FM-graph et du FM-set tree est arrêtée quand un cycle indéterminé est trouvé. Il y a 26 FMs dans le FM-graph et 14 FM-sets dans le FM-set tree jusqu'à l'arrêt de la construction.

Il faut noter qu'il existe des points faibles dans cette approche :

1. Plusieurs FMs sont générés avant le franchissement d'une transition observable à cause des transitions non observables et non fautives ;
2. Il n'y a pas de stratégie de construction des états des deux modèles afin de déterminer rapidement un cycle indéterminé lorsque le système n'est pas diagnosticable. La stratégie par défaut appliquée est de traiter en priorité les transitions observables dans l'ordre de l'alphabet.

Dans la section suivante, l'analyse à la volée est améliorée à partir de ces deux constats.

4.2 Explications minimales et marquages de base

Dans [3, 9], la notion d'explications minimales est utilisée pour l'analyse de la diagnosticabilité. Cette notion réduit l'impact des transitions régulières afin de compacter l'espace d'états. Rappelons dans ce qui suit, quelques définitions fondamentales utilisées dans [3, 4].

Définition 4. *L'ensemble des explications minimales d'une transition observable t à un marquage M est défini par*

$$\Sigma_{\min}(M,t) = \{\sigma \in T_u^* |\, [M[\sigma\rangle M', M' \geq Pre(\cdot,t)] \wedge [\nexists \sigma' \in T_u^* : (M[\sigma'\rangle M', M' \geq Pre(\cdot,t)) \wedge (\pi(\sigma') < \pi(\sigma))]\}$$

L'ensemble correspondant des e-vecteurs minimums de t à M est

$$Y_{\min}(M,t) = \{\pi(\sigma) | \sigma \in \Sigma_{min}(M,t)\}$$

Définition 5. *Soit T_l l'ensemble des transitions qui sont labellisées par l'événement l. L'ensemble des explications minimales de l au M est défini par*

$$\hat{\Sigma}_{\min}(M,l) = \cup_{t \in T_l} \cup_{\sigma \in \Sigma_{\min}(M,t)} \{(t,\sigma)\}$$

L'ensemble correspondant des e-vecteurs *(vecteurs d'explication) minimums de l à M est*

$$\hat{Y}_{\min}(M,t) = \cup_{t \in T_l} \cup_{\overrightarrow{e} \in Y_{\min}(M,t)} \{(t, \overrightarrow{e})\}$$

Définition 6. *Étant donné un RdP-L (N, M_0) avec la fonction de labellisation $\mathcal{L}: T \to \Sigma$. Soit $\omega \in \Sigma_o^*$ une observation donnée. L'ensemble de* justification *de ω est défini comme suit :*

$$\hat{\mathcal{J}}(\omega) = \{(\sigma_o, \sigma_u) |\ \sigma_o \in T_o^*,\ \mathcal{L}(\sigma_o) = \omega,\ \sigma_u \in T_u^*, [\exists \sigma \in \mathcal{L}^{-1}(\omega) :\ \sigma_o = P_o(\sigma),\ \sigma_u = P_u(\sigma)] \wedge [\nexists \sigma' \in \mathcal{L}^{-1}(\omega) :\ \sigma_o = P_o(\sigma'),\ \sigma'_u = P_u(\sigma') \wedge \pi(\sigma'_u) < \pi(\sigma_u)]\}$$

En outre, l'ensemble des j-vecteurs *(vecteurs de justification) correspondant à $\hat{\mathcal{J}}(\omega)$ est défini comme suit :*

$$\hat{Y}_{\min}(M_0,\omega) = \{(\sigma_o, \overrightarrow{y_u}) |\ \sigma_o \in T_o^*,\ \mathcal{L}(\sigma_o) = \omega, \overrightarrow{y_u} \in \mathbb{N}^{|T_u|},\ \exists (\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(\omega) :\ \pi(\sigma_u) = \overrightarrow{y_u}\}$$

Définition 7. *Pour un RdP-L N_L, soit $\omega \in \Sigma_o^*$ une observation donnée et $(\sigma_o, \sigma_u) \in \hat{\mathcal{J}}(\omega)$. Le marquage est calculé par $M_b = M_0 + C_u \cdot \overrightarrow{y_u} + C_o \cdot \overrightarrow{y_o}$ avec $\overrightarrow{y_u} = \pi(\sigma_u)$, $\overrightarrow{y_o} = \pi(\sigma_o)$, c'est-à-dire, le marquage atteint en franchissant σ_o entrelacée par σ_u est un* marquage de base *avec son j-vecteur $\overrightarrow{y}$.*

L'idée des explications minimales et des marquages de base est que, s'il n'existe pas de sous-réseau non observable, il sera possible de caractériser l'ensemble des marquages avec une observation en termes de séquences de longueur minimale. Les marquages atteints par ces séquences sont des marquages de base et tous les autres marquages peuvent être obtenus à partir de la connaissance des marquages de base [6].

Exemple 2. *Considérons le RdP-L dans la Figure 1. Soit* $\omega = ab$*, l'ensemble des justifications est* $\hat{\mathcal{J}}(\omega) = \{(t_2t_4, \varepsilon_1\varepsilon_3)\}$ *et l'ensemble j-vecteurs est* $\hat{Y}_{\min}(M_0, \omega) = \{(t_2t_4, \overrightarrow{e_1'}\}$ *avec l'e-vecteur* $\overrightarrow{e_1'} = [1\ 1\ 0\ 0\ 0]^\tau$*). Ici, pour l'e-vecteur, seulement les éléments des transitions non observables sont gardés. Par exemple,* $\overrightarrow{e_1'} = [1\ 1\ 0\ 0\ 0]^\tau$ *représente que* $\overrightarrow{e_1'}(\varepsilon_1) = 1$*,* $\overrightarrow{e_1'}(\varepsilon_3) = 1$*,* $\overrightarrow{e_1'}(\varepsilon_6) = 0$*,* $\overrightarrow{e_1'}(\varepsilon_8) = 0$ *et* $\overrightarrow{e_1'}(f_9) = 0$*. De plus, ce j-vecteur conduit à un marquage de base* $M_4 = [1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0]^\tau$*.*

Dans [3], le MBRG et le BRD sont proposés pour l'analyse de la diagnosticabilité. Le MBRG est assimilable au graphe d'accessibilité. Le BRD est un graphe déterministe qui fonctionne comme un diagnostiqueur. Cette approche a un faible coût mémoire quand le RdP-L contient plusieurs transitions régulières.

5 L'analyse à la volée utilisant les explications minimales et les T-semiflots

Dans cette section, l'analyse à la volée utilisant les explications minimales et les T-semiflots sera proposée pour améliorer l'analyse à la volée. La notion d'explications minimales est utilisée afin de construire l'espace d'états d'une manière compacte. De plus, la notion de T-semiflots est utilisée pour définir les priorités dans le parcours des branches.

Dans cette approche, deux structures seront développées en utilisant les explications minimales : le BFG (Basis Fault Marking Graph, en français graphe des marquages étendus basiques) et le BFST (Basis Fault Marking Set Tree, en français arbre d'ensembles de marquages étendus basiques), qui sont les versions améliorées respectivement du FM-graph et du FM-set tree.

L'objectif étant de trouver rapidement s'il existe des cycles d'états indéterminés, la stratégie utilisée consiste d'abord à franchir les transitions permettant de faire apparaître dans le BFST des états incertains. Pour cela, on commence par traiter les transitions menant au franchissement de transitions fautives. C'est-à-dire que les explications minimales, ainsi que les justifications, sont limitées à des transitions régulières selon une approche analogue à celle proposée dans [3].

Définissons $T = T_o^i \cup T_u^i$ où $T_o^i = T_o \cup T_f^i$ et $T_u^i = T_u \backslash T_f^i$. Ici, comme indiqué précédent, toutes les transitions fautives dans T_f^i sont considérées comme observables. Ainsi, l'ensemble des transitions "observables" T_o^i contient T_f^i. C_u^i (resp. C_o^i) est la restriction de la matrice d'incidence C, qui se réfère à T_u^i (resp. T_o^i). De plus, $P_u^i \colon T^* \to T_u^{i*}$ est la projection qui supprime les transitions observables dans T_o^i d'une séquence $\sigma \in T^*$. $P_o^i \colon T^* \to T_o^{i*}$ est la projection qui supprime les transitions non observables dans T_u^i de $\sigma \in T^*$. La fonction de labellisation des transitions par rapport à la classe de fautes T_f^i est $\mathcal{L}^i \colon T \to \Sigma$, où $\Sigma = \Sigma_o^i \uplus \{\varepsilon\}$. Puis $\forall u \in T_u^i,\ \mathcal{L}^i(u) = \{\varepsilon\}$ et $\forall t_f^i \in T_f^i,\ \mathcal{L}^i(t_f^i) = \{f^i\},\ f^i \in \Sigma_o^i$. La fonction de labellisation peut être étendue à $\mathcal{L}^i \colon T^* \to \Sigma^*$. Il faut noter que les explications minimales sont limitées à toutes les transitions dans T_u^i, qui représente l'ensemble des transitions régulières.

Définition 8. *L'ensemble de BFMs (Basic Fault Marking, en français basic marquage de faute) par rapport à la classe de fautes* T_f^i *est* $\mathcal{Q}^{b,i}$ *qui est défini par* $\mathcal{Q}^{b,i} = \{BFM^i \mid \exists \sigma \in T^*,\ \omega \in$

$\Sigma_o^{i*},\ BFM_0^i\ [\ \sigma > BFM^i, \mathcal{L}^i(\sigma) = \omega,\ \sigma_u^i = P_u^i(\sigma),\ \sigma_o^i = P_o^i(\sigma),\ s.t.\ (\sigma_o^i, \sigma_u^i) \in \hat{\mathcal{J}}\,(\omega)\}$. *Les BFMs sont notés par* BFM^i.

Le BFG est construit en lieu et place du FM-graph.

Définition 9. *Le BFG par rapport à la classe de fautes* T_f^i *est noté* BFG^i *qui est un 4-uplet* $(\mathcal{N}^{b,i}, \Sigma_o^i, \gamma, BFM_0^i)$*, où :*

- $\mathcal{N}^{b,i} \subseteq \mathcal{Q}^{b,i}$ *est l'ensemble de BFMs (parce que le BFG est construit à la volée avec les conditions d'arrêt,* $\mathcal{N}^{b,i}$ *est un sous-ensemble de* $\mathcal{Q}^{b,i}$*) ;*
- Σ_o^i *est l'ensemble fini des événements observables et* f^i *;*
- $BFM_0^i = [M_0^\tau, 0]^\tau$ *est le nœud initial ;*
- $\gamma\colon \mathcal{N}^{b,i} \times \Sigma_o^i \to 2^{\mathcal{N}^{b,i}}$ *est la fonction de transition de BFM : soient* $BFM_1^i \in \mathcal{Q}^{b,i}$ *et* $\omega \in \Sigma_o^{i*}$*,* $\gamma(BFM_1^i, \omega) = \{BFM_2^i \mid \exists\ \sigma \in T^*\ s.t.\ \mathcal{L}^i(\sigma) = \omega,\ BFM_1^i\ [\ \sigma > BFM_2^i,\ \sigma_u^i = P_u^i(\sigma),\ \sigma_o^i = P_o^i(\sigma),\ (\sigma_o^i, \sigma_u^i) \in \hat{\mathcal{J}}\,(\omega)\}$.

Tous les BFMs dans le BFG peuvent être atteints en franchissant une séquence de transition σv où $v \in T_o^i$.

Sans perte de généralité, dans cet article, le problème d'analyse de la diagnosticabilité est présenté pour une seule classe de fautes. Pour la simplicité de la représentation, l'exposant i sera omis par rapport à T_f^i.

Le BFST est construit en parallèle mais en se basant sur la partie de BFG déjà construite. Nous appelons BFS (BFM-set, en français ensemble de marquages étendus basique) un ensemble de BFM du BFG. L'ensemble des parties des BFM du BFG est $\mathcal{X}^b$. Le BFS initial est $x_0^b = \{BFM_0\}$.

Définition 10. *Le mapping entre les BFSs et les transitions* $\psi\colon \mathcal{X}^b \times \Sigma_o \to \mathcal{X}^b$ *est défini comme suit : pour un BFS* $x^b \in \mathcal{X}^b$ *et un événement* $e \in \Sigma_o$*,* $\psi(x^b, e) = \cup_{BFM \in x^b}\{BFM' \mid \exists \sigma \in T^*\ ,\ \exists t \in \Sigma_o,\ s.t.\ \mathcal{L}(\sigma t) = e,\ BFM\ [\ \sigma t > BFM',\ \sigma_u = P_u(\sigma t),\ \sigma_o = P_o(\sigma t),\ (\sigma_o, \sigma_u) \in \hat{\mathcal{J}}\,(e)\}$.

Tous les BFSs dans le BFST sont atteints en franchissant une séquence de transition σt où $t \in T_o$. La fonction de tag de la Définition 3 et la propagation de faute fonctionnent également sur le BFST.

Le BFG et le BFST sont construits à la volée. Cependant, il y a des différences entre la construction de BFG/BFST et la construction de FM-graph/FM-set tree. À partir d'un nœud BFM, toutes les branches labellisées par une transition fautive $t_f^i \in T_f^i$ avec son explication minimale sont investiguée exhaustivement *a priori*. Tous les nœuds obtenus seront construits dans le BFG. Ensuite, à partir de ces nœuds, le nœud prochain est calculé en franchissant toutes les transitions possibles labellisées par un événement observable sélectionné avec son explication minimale. L'ensemble des BFMs sera construit dans le BFG et un BFS qui contient les BFMs sera construit dans le BFST. En utilisant une technique incrémentale, le BFG et le BFST sont construits pas à pas à la volée. Les conditions d'arrêt de l'analyse à la volée sont utilisées pour terminer la construction.

L'objectif de la construction de BFG et BFST est de réduire davantage la taille correspondante du FM-graph et du FM-set tree, afin d'améliorer l'efficacité de l'analyse à la volée.

L'amélioration principale est la fonction γ de l'algorithme 1 qui remplace la fonction δ dans [13]. Cet algorithme utilise les explications minimales afin d'éviter de calculer les nœuds produits en franchissant les transition non observables régulières. Les sorties de γ sont deux ensembles de BFMs. Les BFMs dans $\mathcal{G}$ sont générés après l'occurrence des transitions fautives, mais les BFMs dans $\mathcal{F}$ sont générés après l'occurrence des transitions observables. Alors, seul les BFMs de $\mathcal{F}$ sont pris en compte pour la construction du BFS. La fonction NEXTFMSET de [13] est remplacée par la fonction NextBFS dans l'Algorithme 2.

Proposition 1. *[10] Étant donné un RdP-L, si un cycle indéterminé existe dans le FM-set tree pour l'observation ω par rapport à la classe de fautes T_f^i, un cycle indéterminé équivalant existera également dans le BFST pour l'observation ω.*

Algorithm 1 Algorithme : γ function

Entrées : un BFM et un événement observable e ;
Sorties : $[\mathcal{F},\ \mathcal{G}] = \gamma(BFM, e)$;
function $\gamma(BFM, e)$
  $\mathcal{G}_{con} \leftarrow \{BFM\}$; ▷ $\mathcal{G}_{con}$ est l'ensemble de BFMs au cours d'étude.
  $\mathcal{G} \leftarrow \emptyset$; ▷ $\mathcal{G}$ est l'ensemble de BFMs atteints à partir de BFM juste après l'occurrence d'une transition fautive dans le BFG
  $\mathcal{F} \leftarrow \emptyset$; ▷ $\mathcal{F}$ est l'ensemble de BFMs atteints à partir de BFM juste après l'occurrence de l'événement e.
  $\mathcal{A} \leftarrow \emptyset$; ▷ $\mathcal{A}$ est l'ensemble d'arcs dans le BFG
  pour tout $y \in \mathcal{G}_{con}$ **faire**
    pour tout $t_f^i \in T_f^i$ **faire**
      si $Y_{min}(marquage(y), t_f^i) \neq \emptyset$ **alors**
        pour tout $\overrightarrow{\varepsilon} \in Y_{min}(marquage(y), t_f^i)$ **faire**
          $marquage(z) \leftarrow marquage(y) + C_u^i \cdot \overrightarrow{\varepsilon} + C(\cdot, t_f^i)$;
          $faute(z) = 1$;
        $\mathcal{G}_{con} \leftarrow \mathcal{G}_{con} \cup \{z\}$;
        $\mathcal{G} \leftarrow \mathcal{G} \cup \{z\}$;
        $\mathcal{A} \leftarrow \mathcal{A} \cup \{(y, (t_f^i, \overrightarrow{\varepsilon}), z)\}$;
  pour tout $y \in \mathcal{G}_{con}$ **faire**
    pour tout $t \in T_e$ **faire** ▷ T_e est l'ensemble de transitions labellisées par l'événement e.
      si $Y_{min}(marquage(y), t) \neq \emptyset$ **alors**
        pour tout $\overrightarrow{\varepsilon} \in Y_{min}(marquage(y), t)$ **faire**
          $marquage(w) \leftarrow marquage(y) + C_u^i \cdot \overrightarrow{\varepsilon} + C(\cdot, t)$;
          $faute(w) = faute(y)$;
        $\mathcal{F} \leftarrow \mathcal{F} \cup \{w\}$;
        $\mathcal{A} \leftarrow \mathcal{A} \cup \{(y, (e, \overrightarrow{\varepsilon}), w\}$; ▷ $\mathcal{A}$ est l'ensemble d'arcs dans le BFG
  retourner $[\mathcal{F},\ \mathcal{G}]$;

Algorithm 2 Algorithme : NextBFS()

Entrées : un BFS x et un événement observable e ;
Sorties : BFS x' qui est atteint à partir de x juste après e ;
function NextBFS(x, e)
  $x' \leftarrow \emptyset$, $\mathcal{N}$, $\mathcal{N}_v$ et $\mathcal{A}_v$ sont des variables globales ;
  pour tout $y \in x$ **faire**
    $[\mathcal{F}, \mathcal{G}] \leftarrow \gamma(y, e)$;
    $x' \leftarrow x' \cup \mathcal{F}$
    $\mathcal{N} \leftarrow \mathcal{N} \cup \mathcal{G} \cup \mathcal{F}$ ▷ $\mathcal{N}$ est l'ensemble des nœuds du BFG.
  $\mathcal{N}_v \leftarrow \mathcal{N}_v \cup x'$ ▷ $\mathcal{N}_v$ est l'ensemble des nœuds du BFST.
  $\mathcal{A}_v \leftarrow \mathcal{A}_v \cup \{(x, e, x')\}$ ▷ $\mathcal{A}_v$ est l'ensemble d'arcs du BFST.
  retourner x' ;

Cette proposition prouve que l'information nécessaire est toujours conservée par rapport à la classe de fautes T_f^i pour l'analyse de la diagnosticabilité.

Afin de réduire l'espace d'états, les explications minimales nous permettent de garder le moins de nœuds possible pour analyser la diagnosticabilité.

Ensuite, la notion de T-semiflots sera introduite pour définir les priorités dans le parcours de branches afin de trouver le plus rapidement un cycle indéterminé existant.

La notion de T-semiflot est une notion importante pour l'analyse de propriété comme la vivacité d'un RdP. Dans cet article, seulement les T-semiflots minimaux sont pris en compte, parce que l'ensemble de tous les T-semiflots minimaux est une base pour tous les T-semiflots. Chaque T-semiflot peut être représenté par une combinaison linéaire de T-semiflots minimaux. En outre, les cycles dans le graphe d'accessibilité correspondants aux T-semiflots minimaux sont des cycles élémentaires. Par conséquent, seulement les T-semiflots minimaux sont étudiés pour la même raison que dans [15]. Un algorithme pour calculer les T-semiflots minimaux est proposé dans [12]. La complexité du calcul des T-semiflots minimaux est polynomiale [7]. S'il existe un cycle dans le graphe d'accessibilité du RdP, il existera un T-semiflot correspondant dans le RdP. Les cycles du graphe d'accessibilité sont vraiment importants pour l'analyse de la diagnosticabilité du système. Par conséquent, les T-semiflots du RdP peuvent être utilisés pour donner des priorités dans le parcours de branches afin d'améliorer l'analyse à la volée et de trouver plus rapidement un cycle indéterminé existant.

Introduisons quelques notations supplémentaires pour l'approche proposée. $\mathcal{I}_F$ est définie comme l'ensemble des T-semiflots minimaux qui contiennent une transition fautive et $\mathcal{I}_N$ est l'ensemble des T-semiflots minimaux qui ne contiennent pas de transition fautive. $\mathcal{S}(\overrightarrow{\Omega})$ est défini comme la trace induite par le T-semiflot $\overrightarrow{\Omega}$ [15], qui est l'ensemble de toutes les séquences de tir possibles construit par les labels dans Σ_T. Par exemple, $\overrightarrow{\Omega} = [1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0]^\tau$ est un T-semiflot du RdP-L de la Figure 1, $\Sigma_T(\overrightarrow{\Omega}) = \{1a,\ 1b,\ 1c\}$, alors $\mathcal{S}(\overrightarrow{\Omega}) = \{\emptyset,\ abc,\ acb,\ bac,\ bca,\ cab,\ cba\}$. Pour $s \in \mathcal{S}(\overrightarrow{\Omega})$, s^n désigne n fois la concaténation de s où n est un nombre entier positif assez grand.

Théorème 1. *[11] Soit* (N, M_0) *un RdP-L* borné *et* vivant. $\forall t_i \in T$, *il existe au moins un T-semiflot minimal* $\overrightarrow{\Omega}$ *tel que* $\overrightarrow{\Omega}(t_i) > 0$.

Ce théorème prouve que chaque transition d'un RdP-L *borné* et *vivant* appartient à au moins un T-semiflot minimal. Le problème fondamental de vérification de la diagnosabilité d'un RdP-L *bornée* et *vivant* est de trouver les cycles indéterminés. S'il existe un cycle indéterminé, le système n'est pas diagnosticable. En utilisant l'analyse à la volée, le but principal est de trouver le cycle indéterminé existant le plus vite possible. Tous les BFS qui construisent le cycle indéterminé sont F-incertains et la séquence des événements du cycle indéterminé est s^n, où $s \in \mathcal{S}(\overrightarrow{\Omega})$ et $\overrightarrow{\Omega} \in \mathcal{I}_N$ (il est possible qu'il existe $\overrightarrow{\Omega}_1 \in \mathcal{I}_N$ et $\overrightarrow{\Omega}_2 \in \mathcal{I}_F$, tel que $s \in \mathcal{S}(\overrightarrow{\Omega}_1) \wedge s \in \mathcal{S}(\overrightarrow{\Omega}_2)$ et la séquence d'événements dans le cycle indéterminé est s^n).

Alors l'idée principale est d'utiliser des T-semiflots pour établir les priorités de parcours de branches et de trouver premièrement un BFS F-incertain et de vérifier ensuite s'il existe une séquence franchissable s^n où $s \in \mathcal{S}(\overrightarrow{\Omega}_N)$ et $\overrightarrow{\Omega}_N \in \mathcal{I}_N$.

L'amélioration principale est la fonction α (Algorithme 3), qui est utilisée pour donner la séquence de tir en construisant à la volée le BFG et le BFST, pour définir des priorités dans le parcours des branches à l'aide des T-semiflots.

À partir d'un BFS, vérifions d'abord s'il est possible de franchir une transition fautive (lignes 6 de l'algorithme 3). Ensuite,

1. Si $Y_{min}(marquage(y), t_f^i) \neq \emptyset$, il sera possible que le prochain BFS obtenu soit F-incertain. La séquence $W = s^n$ sera franchie où $s \in \mathcal{S}(\overrightarrow{\Omega}_N)$ et $\overrightarrow{\Omega}_N \in \mathcal{I}_N$ afin de trouver le cycle indéterminé possible (les lignes 7 à 12 de l'algorithme 3) ;
2. Si $Y_{min}(marquage(y), t_f^i) = \emptyset$, le prochain BFS obtenu sera encore normal. La sequence $W = s'$ sera franchie où $s' \in \mathcal{S}(\overrightarrow{\Omega}_F)$ et $\overrightarrow{\Omega}_F \in \mathcal{I}_F$ afin d'obtenir un BFS F-incertain (les

lignes 18 à 25 de l'algorithme 3). Une fois qu'un BFS F-incertain est obtenu, nous allons à l'étape (1) pour trouver le cycle indéterminé possible.

Algorithm 3 Algorithme : α fonction

1: Entrées : un BFS X, $\mathcal{I}_F$ l'ensemble des T-semiflots minimaux qui contiennent une transition fautive et $\mathcal{I}_N$ l'ensemble des T-semiflots minimaux qui ne contiennent pas de transition fautive.
2: Sortie : une séquence de tir W ;
3: **function** $\alpha(X, \mathcal{I}_N, \mathcal{I}_F)$
4: $\quad\mathcal{T} \leftarrow \emptyset$;
5: $\quad$**pour tout** $y \in X$ **faire**
6: $\quad\quad$**si** $Y_{min}(marquage(y), t_f^i) \neq \emptyset$ **alors**
7: $\quad\quad\quad$**pour tout** $\overrightarrow{\Omega}_N \in \mathcal{I}_N$ **faire**
8: $\quad\quad\quad\quad$**pour tout** $s \in \mathcal{S}(\overrightarrow{\Omega}_N)$ **faire**
9: $\quad\quad\quad\quad\quad$**si** il n'existe pas une séquence σ franchissable tel que $\mathcal{L}(\sigma) = s$ **alors**
10: $\quad\quad\quad\quad\quad\quad$continuer ;
11: $\quad\quad\quad\quad\quad$**sinon**
12: $\quad\quad\quad\quad\quad\quad$retouner $W = s^n$;
13: $\quad\quad$**sinon**
14: $\quad\quad\quad$**pour tout** $y \in X$ **faire**
15: $\quad\quad\quad\quad$**pour tout** $t \in T_o$ **faire**
16: $\quad\quad\quad\quad\quad$**si** $Y_{min}(marquage(y), t) \neq \emptyset$ **alors**
17: $\quad\quad\quad\quad\quad\quad\mathcal{T} \leftarrow \mathcal{T} \cup \{t\}$;
18: $\quad\quad\quad$**pour tout** $t \in \mathcal{T}$ **faire**
19: $\quad\quad\quad\quad$**si** il existe $(t \in \mathcal{T}) \wedge (\exists \overrightarrow{\Omega}_F \in \mathcal{I}_F,\ \overrightarrow{\Omega}_F(t) > 0)$ **alors**
20: $\quad\quad\quad\quad\quad$**pour tout** t et le $\overrightarrow{\Omega}_F$ correspondant **faire**
21: $\quad\quad\quad\quad\quad\quad$**pour tout** $(s \in \mathcal{S}(\overrightarrow{\Omega}))\wedge$(le premier label de s est $\mathcal{L}(t)$) **faire**
22: $\quad\quad\quad\quad\quad\quad\quad$**si** il n'existe pas une séquence σ franchissable $\mathcal{L}(\sigma) = s$ **alors**
23: $\quad\quad\quad\quad\quad\quad\quad\quad$continuer ;
24: $\quad\quad\quad\quad\quad\quad\quad$**sinon**
25: $\quad\quad\quad\quad\quad\quad\quad\quad$retourner $W = s'(s' = \mathcal{L}(\sigma')$ où σ' est la séquence de transitions observables avant la première transition fautive) ;
26: $\quad\quad\quad\quad$**sinon**
27: $\quad\quad\quad\quad\quad$**pour tout** $t \in \mathcal{T}$ **faire**
28: $\quad\quad\quad\quad\quad\quad$retourner $W = \mathcal{L}(t)$;

Exemple 3. *Considérons le RdP-L de la Figure 1. L'analyse à la volée utilisant les explications minimales et les T-semiflots est appliquée pour vérifier la diagnosabilité. Le BFG (Figure 2) et le BFST (Figure 3) sont construits à la volée et en parallèle. Les BFMs et les explications minimales sont donnés par le Tableau 1. Chaque arc est associé à une transition avec son label et son e-vecteur. Les priorités dans le parcours des branches sont définies en utilisant les T-semiflots. Ici,* $\mathcal{I}_N = \{\overrightarrow{\Omega}_{N1} = [1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0]^\tau\}$ *et* $\mathcal{I}_F = \{\overrightarrow{\Omega}_{F1} = [0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1]^\tau\}$. *un BFS F-incertain ne peut pas être obtenu parce que* $Y_{min}(marquage(BFM_0), f_9) = \emptyset$. *Comme la transition observable* t_7 *est franchissable en* BFM_0 *avec* $Y_{min}(marquage(BFM_0), f_9) \neq \emptyset$ *et* $\overrightarrow{\Omega}_{F1}(t_7) > 0$, *le prochain événement observable est d (ligne 19 de l'algorithme 3). Un BFS F-incertain peut être obtenu en franchissant la transition fautive* f_9. *Ensuite, il est nécessaire de chercher s'il existe une séquence franchissable* $W = s^n$, *où* $s \in \mathcal{S}(\overrightarrow{\Omega_{N1}})$ *(lignes 7 à 12 de l'algorithme 3). Ici, la seule possibilité est* $s = abc$, *donc l'ordre des prochains événements est* $a \rightarrow b \rightarrow c \rightarrow a \cdots$. *Un cycle F-incertain est trouvé dans le BFST de la Figure 3. À l'aide du BFG de la Figure 2, il est identifié que c'est un cycle indéterminé. Alors il n'est pas nécessaire de continuer la construction du BFG et du BFST, et il peut être conclu que le système n'est pas diagnosticable.*

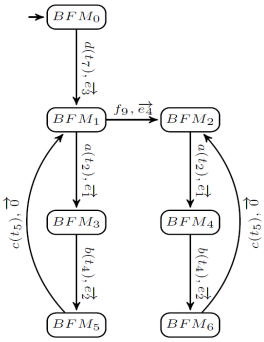

Figure 2: BFG

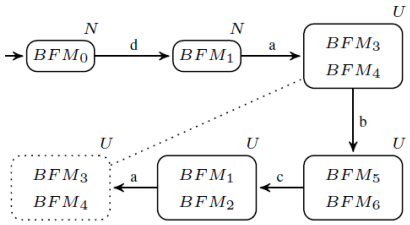

Figure 3: BFST

Table 1: BFMs de la Figure 2 et la Figure 3

j	BFM_j	j	$\overrightarrow{e_j}$
0	$[2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \mid 0]^\tau$	1	$[1\ 0\ 0\ 0]$
1	$[1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0 \mid 0]^\tau$	2	$[0\ 1\ 0\ 0]$
2	$[1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1 \mid 1]^\tau$	3	$[0\ 0\ 1\ 0]$
3	$[0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0 \mid 0]^\tau$	4	$[0\ 0\ 0\ 1]$
4	$[0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1 \mid 1]^\tau$		
5	$[0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0 \mid 0]^\tau$		
6	$[0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1 \mid 1]^\tau$		

Dans cet exemple, l'analyse à la volée utilisant les explications minimales et les T-semiflots permet d'économiser beaucoup de mémoire comparée à l'analyse à la volée originale proposée dans [13]. L'espace d'états est construit d'une manière compacte en utilisant les explications minimales afin d'éliminer l'impact des transitions régulières avec la même idée que dans [3]. De plus, cette approche définit des priorités dans le parcours des branches en produisant la séquence de tir suivante à l'aide des T-semiflots. Moins de branches sont investiguées pour trouver plus rapidement le cycle indéterminé existant. En résumé, l'analyse à la volée utilisant les explications minimales et les T-semiflots est plus efficace avec un plus faible coût mémoire pour l'analyse de la diagnosticabilité, en particulier pour un RdP-L non diagnosticable.

6 Analyse de l'efficacité et de la complexité

Le Tableau 2 offre une comparaison de l'espace d'états (et donc indirectement la mémoire nécessaire) à construire pour analyser la diagnosticabilité du modèle RdP-L de la Figure 1 par 4 approches évoquées dans cet article.

Pour l'approche "diagnostiqueur", l'espace d'états entier du RdP-L de la Figure 1 est construit. Le graphe d'accessibilité contient 44 états et l'automate du diagnostiqueur contient 33 macro-états construits par $\epsilon-$réduction. En raison de nombreuses transitions régulières, l'analyse MBRG/BRD compacte l'espace d'états à l'aide d'explications minimales. Pour l'analyse à la volée, seulement une partie de l'espace d'états est construite et la construction est arrêtée dès qu'un cycle indéterminé est trouvé. Cependant, l'efficacité de l'approche à la volée dépend extrêmement des modèles de RdP-L, parce qu'elle n'utilise pas une véritable stratégie de recherche des cycles indéterminés. Pour le RdP-L de la Figure 1, l'analyse à la volée consomme beaucoup de mémoire avec 26 FMs dans le FM-graph et 14 FM-sets dans le FM-set tree. À l'aide des explications minimales, l'espaces d'états est construit de manière compacte. À l'aide des T-semiflots, les priorités dans le parcours des branches sont définies et

Table 2: Nombre d'états dans les différentes approches

Nom des approches	Espace d'états (Graph d'accessibilité)	Modèle pour le diagnostic
Approche "Diagnostiqueur" (Graph d'accessibilité/Diagnostiqueur)	44	33
MBRG/BRD	15	14
Analyse à la volée classic (FM-graph/FM-set tree)	26	14
Analyse à la volée utilisant les T-semiflots et les explications minimales (BFG/BFST)	7	6

le cycle indéterminé existant est trouvé de manière plus directe. L'analyse à la volée utilisant les explications minimales et les T-semiflots a juste 7 BFMs dans le BFG et 6 BFS dans le BFST. En général, notre approche réduit le coût de calcul en économisant du temps et de la mémoire, en particulier pour un RdP-L non diagnosticable.

Analysons à présent la complexité des différentes approches. Pour une classe de fautes, p est le nombre des marquages du RdP-L considéré. Notons que, si le graphe d'accessibilité est considéré comme un automate, p est aussi le nombre d'états de l'automate.

Pour l'approche "diagnostiqueur", le graphe d'accessibilité contient p nœuds et le nombre des nœuds dans le diagnostiqueur est $\leq 2^{2p}$ par rapport à la classe de fautes. Pour l'analyse de MBRG/BRD, le nombre des nœuds dans le MBRG est $m+f$ $(m+f \leq p)$, où m est le nombre des marquages de base et f est le nombre des marquages qui ne sont pas des marquages de base mais qui sont produits en traitant les transitions fautives comme des transitions observables. En théorie, le nombre des nœuds dans le BRD est $2^{2m} \cdot \alpha$ avec $\alpha \leq 4$. Le coefficient α est dû à l'utilisation dans l'approche MBRG/BRD de la fonction Δ permettant d'indiquer l'occurrence ou pas d'une faute [3, 4]. Dans le pire des cas, il n'existe pas de transition non observable régulière, alors $m = p$ et le nombre des nœuds dans le BRD est $2^{2p} \cdot \alpha$. Pour l'analyse à la volée, le nombre des nœuds dans le FM-graph est $\leq 2p$ et le nombre des nœuds dans le FM-set tree est $\leq 2^{2p}$. Dans le pire des cas, tous les FM-sets sont générés, c'est-à-dire les états de FM-set tree sont entièrement énumérés. La complexité en termes de mémoire est alors égale à celle de l'approche "diagnostiqueur" [14]. Pour l'analyse à la volée utilisant les explications minimales et les T-semiflots, le nombre des nœuds dans le BFG est $\leq 2m' + f'$ $(m' \leq m)$, où m' est le nombre de marquages de base générés et f' est le nombre de marquage qui ne sont pas de marquages de base, mais qui sont produits en considérant les transitions fautives comme des transitions observables. Le nombre des nœuds dans le BFST est $\leq 2^{2m'+f'}$. Dans le pire des cas, il n'existe pas de transition non observable et non fautive et les états de BFST sont entièrement énumérés. La complexité en termes de mémoire est alors du même ordre que celle de l'approche "diagnostiqueur" [14].

Il faut noter que, s'il existe plusieurs classes de faute, l'approche diagnostiqueur construit un seul graphe d'accessibilité et un diagnostiqueur pour toutes les classes de faute. De même, l'analyse de MBRG/BRD construit un seul MBRG et un BRD. Les autres approches construisent un FM-graph (BFG) et un FM-set tree (BFST) pour chaque classe de fautes.

7 Conclusion

Dans cet article, la contribution principale est l'analyse à la volée utilisant les explications minimales et les T-semiflots, qui améliore la méthode originale [13]. Grâce à cette approche, l'espace d'états peut être construit à la volée d'une manière compacte et un cycle indéterminé

peut être trouvé plus rapidement. Notre approche réduit le coût de mémoire de l'analyse de la diagnosticabilité, particulièrement pour un RdP non diagnosticable. Lorsque le système est diagnosticable, l'espace d'états peut être important si les conditions d'arrêt ne sont pas rapidement atteintes. Toutefois, l'espace d'états dans cette approche est toujours inférieur ou égal (dans le pire des cas) à celui de l'approche du "diagnostiqueur", qui peut être construit à l'aide du graphe d'accessibilité en appliquant la technique de Sampath définie dans [14].

Les perspectives de recherche sont d'abord de relâcher l'hypothèse forte d'exigence de vivacité du RdP-L considéré. Cette hypothèse est forte car elle est difficile à garantir en pratique après l'occurrence d'une faute dans un système. Une manière de relâcher cette hypothèse serait de considérer que l'occurrence d'une faute ne doit pas rendre bloquant le système. En outre, afin de réduire la complexité théorique, l'analyse à la volée sera associée au VN (Verifier Net, en français réseau de vérificateur) proposé dans [2] et qui a une complexité plus faible.

References

[1] F. Basile, P. Chiacchio, and G. De Tommasi. On K-diagnosability of Petri Nets via Integer Linear Programming. *Automatica*, 48(9):2047–2058, 2012.

[2] M.P. Cabasino, A. Giua, S. Lafortune, and C. Seatzu. A New Approach for Diagnosability Analysis of Petri Nets Using Verifier Nets. *IEEE Transactions Automatic Control*, 57(12):3104–3117, 2012.

[3] M.P. Cabasino, A. Giua, and C. Seatzu. Diagnosability of Bounded Petri Nets. In *Proc. of the 48th IEEE Conf. on decision and control. Shanghai, China. December*, pages 1254–1260, 2009.

[4] M.P. Cabasino, A. Giua, and C. Seatzu. Fault Detection for Discrete Event Systems Using Petri Nets with Unobservable Transitions. *Automatica*, 46(9):1531–1539, 2010.

[5] C.G. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems*. Spring, 2007.

[6] D. Corona, A. Giua, and C. Seatzu. Marking Estimation of Petri Nets With Silent Transitions. In *43rd Conference on Decision and Control*, pages 966–971, 2004.

[7] R. David and H. Alla. Discrete, Continuous,and Hybrid Petri Nets. Springer, 2005.

[8] S. Jiang, Z. Huang, V. Chandra, and R. Kumar. A Polynomial Algorithm for Testing Diagnosability of Discrete Event Systems. *IEEE Transactions Automatic Control*, 46(8):1318–1321, 2001.

[9] G Jiroveanu and R Boel. The Diagnosability of Petri Net Models Using Minimal Explanations. *IEEE Transactions on Automatic Control*, 55(7):1663–1668, 2010.

[10] B. Li, B. Liu, and A. Toguyéni. On-the-fly Diagnosability Analysis of Labeled Petri Nets Using Minimal Explanations. In *9th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes - SAFEPROCESS'2015*, 2015.

[11] B. Li, A. Toguyéni, and M. Khlif-bouassida. On-the-fly Diagnosability Analysis of Labeled Petri Nets Using T-invariants. In *5th International Workshop on Dependable Control of Discrete Systems - DCDS'2015*, 2015.

[12] C. Lin, S. T.Chanson, and T. Murata. Petri Net Models and Efficient T-invariant Analysis for Logical Inference of Clauses. In *Procs. of the 1996 IEEE International Conference on Systems, Man, and Cybernetics*, pages 3174–3179, 1996.

[13] B. Liu, M. Ghazel, and A. Toguyéni. Toward an Efficient Approach for Diagnosability Analysis of DES Modeled by Labeled Petri Nets. In *13th European Control Conference - ECC'2014*, 2014.

[14] M. Sampath, R. Sengupta, and S. Lafortune. Diagnosability of Discrete-Event Systems. *IEEE Transactions Automatic Control*, 40(9):1555–1575, 1995.

[15] Y. Wen and M. Jeng. Diagnosability Analysis Based on T-invariants of Petri Nets. In *Networking, Sensing and Control*, pages 371–376, 2005.

[16] T. Yoo and S. Lafortune. Polynomial-Time Verification of Diagnosability of Partially Observed Discrete-Event Systems. *IEEE Transactions Automatic Control*, 47(9):1491–1495, 2002.

A Méthodes alternatives d'analyse de la diagnosticabilité de l'exemple illustratif

A.1 Méthode de l'analyse à la volée

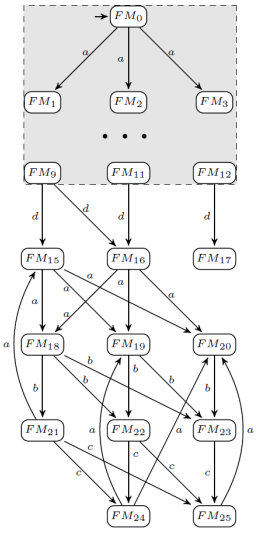

Figure 4: FM-graph

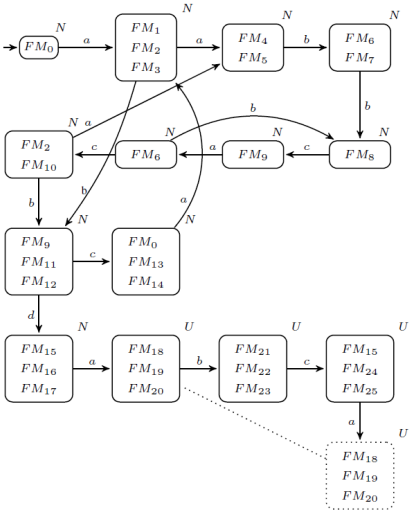

Figure 5: FM-set tree

Table 3: Fault markings de Fig. 4 et Fig. 5

j	FM_j	j	FM_j
0	$[2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \mid 0]^\tau$	13	$[1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0 \mid 0]^\tau$
1	$[0\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0 \mid 0]^\tau$	14	$[1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0 \mid 0]^\tau$
2	$[1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0 \mid 0]^\tau$	15	$[1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0 \mid 0]^\tau$
3	$[0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0 \mid 0]^\tau$	16	$[0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0 \mid 0]^\tau$
4	$[0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0 \mid 0]^\tau$	17	$[0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0 \mid 0]^\tau$
5	$[0\ 0\ 2\ 0\ 0\ 0\ 0\ 0\ 0 \mid 0]^\tau$	18	$[0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0 \mid 0]^\tau$
6	$[0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0 \mid 0]^\tau$	19	$[0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0 \mid 0]^\tau$
7	$[0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0 \mid 0]^\tau$	20	$[0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1 \mid 1]^\tau$
8	$[0\ 0\ 0\ 0\ 2\ 0\ 0\ 0\ 0 \mid 0]^\tau$	21	$[0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0 \mid 0]^\tau$
9	$[1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0 \mid 0]^\tau$	22	$[0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0 \mid 0]^\tau$
10	$[1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0 \mid 0]^\tau$	23	$[0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1 \mid 1]^\tau$
11	$[0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0 \mid 0]^\tau$	24	$[1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0 \mid 0]^\tau$
12	$[0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0 \mid 0]^\tau$	25	$[1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1 \mid 1]^\tau$

Exemple 4. *Pour le RdP-L de la Figure 1, l'analyse à la volée est utilisée pour l'analyse de la diagnosticabilité. Quant aux priorités dans le parcours des branches, la transition labellisée par a est choisie* a priori *avent la transition labellisée par b, et ensuite c, d and e. Le FM-graph et*

le FM-set tree sont construits à la volée en parallèle. Dans la Figure 5, le tag de chaque FM-set est indiqué à côté. Les FMs sont montrés dans le Tableau 3. Il faut noter que seulement les FMs observables sont présentés dans les deux structures. Les FMs atteints en franchissant des transitions non observables ne sont pas intégrés mais ils sont exploités pour calculer le prochain FM-set qui sera atteint après l'occurrence d'un événement observable. La construction est arrêtée à cause d'un cycle détecté : un nouveau FM-set F-incertain est égal à un FM-set existant (le FM-set qui contient FM_{18}, FM_{19} et FM_{20}). À l'aide de FM-graph, il est indiqué que le cycle détecté est indéterminé. Donc il n'est pas nécessaire de continuer la construction du FM-graph et du FM-set tree, et il peut être conclu que le système n'est pas diagnosticable. La numérotation des FMs dans la Figure 4 et la Figure 5 correspond à l'ordre de la construction d'état par l'algorithme d'analyse en profondeur d'abord. Cette analyse en profondeur d'abord est basée sur la construction du FM-set tree. Par exemple, au FM initial $FM-set_0$ qui contient FM_0, le prochain événement franchissable peut être a ou d. L'événement a est franchi parce que a doit être choisi avant d. Ensuite, les nœuds FM_1, FM_2 et FM_3 sont ajoutés au FM-graph. En parallèle, $FM-set_1$ est construit et contient FM_1, FM_2 et FM_3. Ensuite, commençons à $FM-set_1$ pour continuer la construction comme cette façon jusqu'à ce que le cycle indéterminé soit trouvé.

A.2 L'analyse de MBRG/BRD

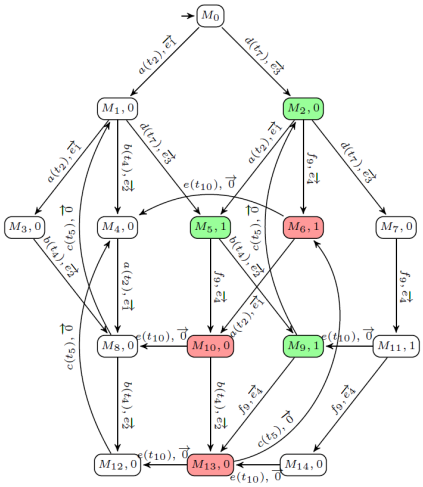

Figure 6: MBRG

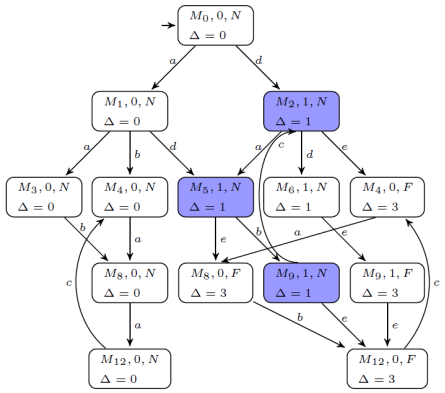

Figure 7: BRD

Table 4: Markings et e-vectors de Fig. 6 et Fig. 7

j	M_j	j	M_j
0	$[2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^\tau$	11	$[0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1]^\tau$
1	$[1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0]^\tau$	12	$[0\ 0\ 0\ 0\ 2\ 0\ 0\ 0\ 0]^\tau$
2	$[1\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0]^\tau$	13	$[0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1]^\tau$
3	$[0\ 0\ 2\ 0\ 0\ 0\ 0\ 0\ 0]^\tau$	14	$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 2]^\tau$
4	$[1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0]^\tau$		
5	$[0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0]^\tau$		
6	$[0\ 0\ 0\ 0\ 0\ 0\ 2\ 0\ 0]^\tau$	j	$\overrightarrow{e_j}$
7	$[1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1]^\tau$	1	$[1\ 0\ 0\ 0]$
8	$[0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0]^\tau$	2	$[0\ 1\ 0\ 0]$
9	$[0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0]^\tau$	3	$[0\ 0\ 1\ 0]$
10	$[0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1]^\tau$	4	$[0\ 0\ 0\ 1]$

Exemple 5. *Le MBRG construit correspond au RdP-L de la Figure 1 est montré dans la Figure 6. Les marquages et les e-vecteurs sont donnés par le Tableau 4. Ici, pour l'e-vecteur, seulement les éléments des transitions non observables et non fautives sont gardés. Par exemple,* $\overrightarrow{e_1} = [1\ 0\ 0\ 0]^T$ *représente que* $\overrightarrow{e_1}(\varepsilon_1) = 1$, $\overrightarrow{e_1}(\varepsilon_3) = 0$, $\overrightarrow{e_1}(\varepsilon_6) = 0$ *et* $\overrightarrow{e_1}(\varepsilon_8) = 0$, *au lieu de* $\overrightarrow{e_1} = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$. *Le BRD correspondant au MBRG est montré dans la Figure 7. On en conclu que le system est non diagnosable, car il existe un cycle indéterminé et les états de ce cycle sont marqués dans les zones d'ombre.*