

Pronostic en ligne des fautes des Systèmes Industriels modélisés par Réseaux de Petri Etiquetés

Braian Igreja de Freitas^{1,*}, A.K. Armand Toguyeni^{2,†}, and João Carlos Basilio^{3,‡}

¹ Université Fédérale de Rio de Janeiro, Rio de Janeiro, Brésil
bfreitas.mesc@gmail.com

² Centrale Lille Institut/CRISTAL, Lille, France.
armand.toguyeni@centralelille.fr

³ Université Fédérale de Rio de Janeiro, Rio de Janeiro, Brésil.
basilio@dee.ufrj.br

Abstract

Cet article traite le problème du pronostic de fautes des systèmes à événements discrets modélisés par des Réseaux de Petri Etiquetés (LPN). Le pronostic de fautes est un processus en ligne, également connu sous le nom de prédiction de fautes. Il a pour objectif principal de détecter qu'une faute non observable, se produira inéluctablement dans le futur, compte tenu de l'observation actuelle d'événements. En utilisant les concepts de marquage de base et d'arbres d'accessibilité, nous montrons que le pronostic peut être effectué en ligne en exploitant une copie du LPN d'origine, appelée Réseau de Petri piloté par les fautes. Ce travail propose donc une solution au problème du pronostic de fautes pour tout LPN borné, sans cycles de transitions non observables.

1 Introduction

Dans le contexte des systèmes à événements discrets (SEDs), le pronostic de faute, également connu sous le nom de prédiction de faute, consiste à détecter qu'une faute non observable se produira inévitablement dans le futur. Cette prédiction peut être effectuée après un nombre fini d'événements survenus pendant le fonctionnement du système. Ce problème a été abordé pour la première fois par [9], dans lequel le problème de la pronosticabilité des SEDs modélisés par des automates est abordé.

Après [9], plusieurs travaux ont été consacrés au pronostic et à la pronosticabilité (ou prédictabilité) des SEDs modélisés par automates [16]. La notion de pronostic pour les automates a d'abord été étendue à la co-pronosticabilité par [11], et récemment revisitée par [3]. En outre, les notions de pronostic et de pronosticabilité ont également été étendues aux automates ayant des propriétés stochastiques [5, 6]. Cependant, beaucoup moins d'articles ont été proposés pour les SEDs modélisés par des Réseaux de Petri (RdP). L'utilisation de RdP comme formalisme pour traiter le problème offre plusieurs avantages par rapport aux automates, tels qu'une modélisation plus compacte du parallélisme, de la synchronisation et du partage de ressources entre processus. Néanmoins, certains travaux ont fait progresser de manière significative l'étude du pronostic des fautes, avec des modèles RdP. Certaines approches se concentrent sur la vérification de la pronosticabilité des Réseaux de Petri

*Les travaux de Braian I. Freitas ont été financés par la Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Finance Code 001 au Brésil et, la Région Hauts-de-France.

†Ce travail d'Armand Toguyeni a été financé par la Région Hauts-de-France (France)

‡Les travaux de João C. Basilio ont été soutenus en partie par le Conseil brésilien de la recherche (CNPq), grâce à la subvention 316881/2021-0, et en partie par la Fondation Carlos Chagas, Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ), grâce à la subvention E-26/201.144-2022.

Etiquetés (également appelés Labeled Petri Nets - LPNs) [19, 18, 17] plutôt que sur le pronostic en lui même. Certaines méthodes de pronostic concernant l'apparition future d'une faute sont en fait une estimation, qui peut se révéler correcte ou non au fil du temps [1, 10]. D'autres méthodes se sont concentrées sur le pronostic exact de fautes dans différentes classes de RdP [2, 12, 13]. Les travaux de [2] portent sur une approche incrémentale pour le pronostic de fautes des RdP stochastiques bornés et partiellement observables, où les transitions et les places du RdP peuvent être observables ou non. De même, [12] et [7] traitent du k pronostic des RdP partiellement observables, en analysant toutes les séquences de transitions possibles qui peuvent se déclencher à partir des marquages courants ou, en résolvant des problèmes de programmation linéaire en nombres entiers. L'approche de [13] se concentre sur les LPNs bornés. Son objectif est de détecter que le LPN atteindra inévitablement certains marquages déclencheurs d'alarme.

Dans [8], une nouvelle approche pour le pronostic des fautes des LPNs bornés et non bornés qui n'ont pas de cycles de transitions non observables a été proposée. Cette approche consiste en l'analyse d'une copie du LPN du système, appelée Réseau de Petri piloté par les fautes (également appelée Fault Driven Petri Nets – FDPN). Le FDPN se comporte de la même manière que le LPN du système ; cependant, il existe une relation entre les marquages accessibles du FDPN et l'occurrence de la faute. En utilisant cette relation, des arbres de couvertures sont construits pour analyser les marquages de base du FDPN et effectuer en ligne le pronostic de fautes. Bien que [8] soit une approche générale, son principal inconvénient est le calcul en ligne des arbres de couvertures, qui peut devenir lourd pour certains LPNs. Par conséquent, le temps de calcul en ligne peut devenir inapproprié pour les applications temps réel.

Afin de remédier à cet inconvénient, nous proposons dans cet article une approche similaire à celle de [8], mais qui se concentre sur le pronostic de fautes des LPNs bornés. Au lieu de calculer les arbres de couvertures en ligne, nous calculons hors ligne les arbres d'accessibilité issus des marquages de base du graphe d'accessibilité de base (i.e. Basic Reachability Graph ou BRG) du LPN. Ce graphe ne peut être calculé que pour des LPNs bornés [4]. Cette construction a priori, nous permet, en ligne, de décider de l'état du pronostiqueur directement à partir des nœuds du BRG cohérents avec l'observation de l'événement en cours.

Le calcul en ligne de notre approche est non seulement plus efficace que celui de [8] pour les LPNs bornés, mais il est également plus efficace que ceux de [12] et [7]. En effet, nous pouvons effectuer le pronostic en ligne sans avoir besoin d'effectuer l'analyse en ligne de toutes les séquences de transition possibles qui peuvent être déclenchées à partir des marquages cohérents avec l'observation de l'événement ou de résoudre des problèmes de programmation linéaire en nombres entiers pour chaque observation d'événement.

Cet article est organisé comme suit. La Section 2 présente quelques concepts de base relatifs aux LPNs, aux arbres d'accessibilité, aux marquages et justifications de base et aux BRGs. La Section 3 formule le problème du pronostic de fautes des LPNs. La Section 4 présente notre approche pour le pronostic de fautes en ligne des LPNs bornés. Enfin, la Section 5 conclut ce document et présente les perspectives de ces travaux.

2 Contexte théorique

2.1 Réseaux de Petri étiquetés ou Labeled Petri nets

Les Réseaux de Petri Etiquetés (LPNs) sont des graphes biparties pondérés définis par le septuple $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$, où $P = \{p_1, \dots, p_{n_P}\}$ (resp. $T = \{t_1, \dots, t_{n_T}\}$), est un ensemble de places (resp. de transitions) Pre (resp. $Post$) est une matrice $n_P \times n_T$ dont les

éléments $Pre(p, t)$ (resp. $Post(p, t)$) représente le poids de l'arc reliant la place $p \in P$ à la transition $t \in T$ (resp. transition $t \in T$ à la place $p \in P$), \vec{m}_0 est le vecteur de marquage initial, Σ est un ensemble d'événements, et $\ell : T \rightarrow \Sigma$ est une fonction d'étiquetage qui attribue à chaque transition $t \in T$ un événement $\sigma \in \Sigma$. La matrice d'incidence de \mathcal{N}_l est notée $W = Post - Pre$.

Les états d'un LPN \mathcal{N}_l sont désignés par les vecteurs marquage $\vec{m} = [m(p_1) \ m(p_2) \ \dots \ m(p_{n_P})]^T$, où $m(p) \in \mathbb{Z}_+$ pour $p \in P$, et \mathbb{Z}_+ désigne l'ensemble des nombres entiers non négatifs. Le marquage \vec{m} peut également être représenté par $\vec{m} = m(p_1)p_1 + m(p_2)p_2 + \dots + m(p_{n_P})p_{n_P}$, ce qui est simplifié en supprimant la composante $m(p_k)p_k$, $k \in \{1, 2, \dots, n_P\}$, chaque fois que $m(p_k)$ est égal à zéro.

Une transition $t \in T$ est validée par un marquage \vec{m} lorsque $m(p) \geq Pre(p, t)$ pour tout $p \in P$, ce qui est désigné par $\vec{m}[t]$. Lorsqu'un événement $\sigma \in \Sigma$ se produit alors que le marquage actuel est \vec{m} , l'une des transitions $t \in T$ activée par \vec{m} et étiquetée par σ se déclenche, transformant \vec{m} en un nouveau marquage \vec{m}' , où, pour tout $p \in P$, $m'(p) = m(p) + W(p, t)$. Ce franchissement est désigné par $\vec{m}[t]\vec{m}'$.

L'ensemble de toutes les séquences de transition possibles formées avec T est noté T^* . Le vecteur de tirs d'une séquence de transitions $r \in T^*$ est désigné par $\vec{\pi}_r = [\pi_r(t_1) \ \pi_r(t_2) \ \dots \ \pi_r(t_{n_T})]^T$, où $\pi_r(t) \in \mathbb{Z}_+$ est le nombre de fois que la transition t apparaît dans r . Une séquence $r \in T^*$ est validée par un marquage \vec{m} si les transitions de r peuvent être déclenchées séquentiellement à partir de \vec{m} , ce qui est dénoté par $\vec{m}[r]$. Le déclenchement séquentiel des transitions de r à partir d'un marquage \vec{m} , noté $\vec{m}[r]\vec{m}'$, produit un nouveau marquage $\vec{m}' \in \mathbb{Z}_+^{n_P}$, où $\vec{m}' = \vec{m} + W\vec{\pi}_r$. L'ensemble Σ^* désigne la fermeture de Kleen de Σ , à savoir l'ensemble de toutes les séquences d'événements formées avec Σ , y compris la séquence vide ε . La fonction ℓ peut être étendue aux séquences de transition $r \in T^*$, produisant des séquences d'événements $s \in \Sigma^*$ telles que $s = \ell(r)$.

Partitionnons l'ensemble des événements en deux sous-ensembles, soit $\Sigma = \Sigma_o \dot{\cup} \Sigma_{uo}$, où Σ_o et Σ_{uo} sont respectivement les ensembles d'événements observables et inobservables. De même, nous partitionnons $T = T_o \dot{\cup} T_{uo}$, où T_o (resp. T_{uo}) est l'ensemble des transitions observables (resp. inobservables) étiquetées par les événements de Σ_o (resp. Σ_{uo}). La projection observable des événements est l'application $P_o : \Sigma^* \rightarrow \Sigma_o^*$, définie par la récursivité suivante : (i) $P_o(\varepsilon) = \varepsilon$, (ii) $P_o(\sigma) = \sigma$, si $\sigma \in \Sigma_o$, ou $P_o(\sigma) = \varepsilon$, si $\sigma \in \Sigma_{uo}$, et (iii) $P_o(s\sigma) = P_o(s)P_o(\sigma)$, pour $s \in \Sigma^*$ et $\sigma \in \Sigma$. La projection inverse est l'application $P_o^{-1} : \Sigma_o^* \rightarrow 2^{\Sigma^*}$, où $P_o^{-1}(s_o) = \{s \in \Sigma^* : s_o = P_o(s)\}$.

Le sous-réseau induit par T' de \mathcal{N}_l est obtenu en supprimant les transitions de T qui ne sont pas dans T' . Un LPN est acyclique s'il n'a pas de cycles formés avec les places et les transitions du LPN. L'ensemble de toutes les séquences de transition activées de \mathcal{N}_l est défini par $LT(\mathcal{N}_l) = \{r \in T^* : \vec{m}_0[r]\}$. L'ensemble des marquages accessibles de \mathcal{N}_l est défini comme suit : $\mathcal{RC}(\mathcal{N}_l) = \{\vec{m} \in \mathbb{Z}_+^{n_P} : (\exists r \in LT(\mathcal{N}_l))[\vec{m}_0[r]\vec{m}]\}$. Le LPN est borné si $\mathcal{RC}(\mathcal{N}_l)$ est fini ; sinon, \mathcal{N}_l est non borné. Le langage de \mathcal{N}_l est défini comme $L(\mathcal{N}_l) = \{s \in \Sigma^* : (\exists r \in LT(\mathcal{N}_l))[\ell(r) = s]\}$. Le langage observable de \mathcal{N}_l est défini comme $L_o(\mathcal{N}_l) = \{s_o \in \Sigma_o^* : (\exists s \in L(\mathcal{N}_l))[s_o = P_o(s)]\}$. Étant donné une séquence d'événements $s \in L(\mathcal{N}_l)$, l'ensemble des marquages s -cohérents est défini par $C(\mathcal{N}_l, s) = \{\vec{m} \in \mathcal{RC}(\mathcal{N}_l) : (\exists r \in LT(\mathcal{N}_l))[(\ell(r) = s) \wedge (\vec{m}_0[r]\vec{m})]\}$. L'ensemble des marquages s_o -cohérents de toute séquence d'événements observables $s_o \in L_o(\mathcal{N}_l)$ est défini par $C_o(\mathcal{N}_l, s_o) = \{\vec{m} \in \mathcal{RC}(\mathcal{N}_l) : (\exists s \in L(\mathcal{N}_l))[(P_o(s) = s_o) \wedge (\vec{m} \in C(\mathcal{N}_l, s))]\}$.

Dans le reste du papier, nous faisons l'hypothèse suivante :

- **A1.** Le sous-réseau induit par T_{uo} du LPN est acyclique.

2.2 Arbres d'accessibilité

Étant donné un LPN borné, un arbre d'accessibilité ou Reachable Tree (RT) est un triplé $\Psi = (Q, A, \mu)$, où $Q = \{q_1, q_2, \dots, q_{n_Q}\}$ est l'ensemble des nœuds, $A \in 2^{Q \times T \times Q}$ est l'ensemble des arcs qui relient les nœuds de Q et sont associés aux transitions de T , et $\mu : Q \rightarrow \mathcal{RC}(\mathcal{N}_l)$ est la fonction d'étiquetage des nœuds qui associe chaque nœud à un marquage accessible. Un algorithme de calcul de Ψ est donné dans [14]. Nous désignons par $\Psi_{\vec{m}}$ le RT calculé avec le marquage $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$ étant considéré comme marquage initial.

L'ensemble des nœuds terminaux de Ψ est défini comme $Q_T(\Psi) = \{q \in Q : (\forall q' \in Q)(\forall t \in T)((q, t, q') \notin A)\}$, c'est-à-dire l'ensemble des nœuds qui n'ont pas de successeur. Les nœuds de $Q_T(\Psi)$ correspondent aux deux cas suivants : (i) des marquages accessibles qui correspondent à des blocages ou (ii) des nœuds étiquetés par les mêmes marquages accessibles que des nœuds précédents (nœuds répétitifs). Nous partitionnons donc $Q_T(\Psi)$ en $Q_T(\Psi) = Q_D(\Psi) \dot{\cup} Q_R(\Psi)$, où $Q_D(\Psi)$ et $Q_R(\Psi)$ sont des ensembles formés respectivement de nœuds de blocages et de nœuds répétitifs. Pour un nœud répétitif $q \in Q_R(\Psi)$, nous désignons par q^R son prédécesseur ayant le même marquage accessible.

2.3 Marquages de base et justifications

Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN. Étant donné un marquage accessible $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$ et une transition observable $t_o \in T_o$, l'ensemble correspondant d'explications minimales est défini par $\mathcal{E}_{min}(\vec{m}, t_o) = \{r_{uo} \in T_{uo}^* : (\vec{m}[r_{uo}t_o]) \wedge ((\forall r'_{uo} \in T_{uo}^* \setminus \{r_{uo}\})(\neg(\vec{m}[r'_{uo}t_o])) \vee (\neg(\vec{\pi}_{r'_{uo}} < \vec{\pi}_{r_{uo}}))))\}$, où $\vec{\pi}_{r'_{uo}} < \vec{\pi}_{r_{uo}}$ si, et seulement si $\pi_{r'_{uo}}(t) \leq \pi_{r_{uo}}(t)$ pour tout $t \in T$ et s'il existe $t \in T$ tel que $\pi_{r'_{uo}}(t) < \pi_{r_{uo}}(t)$. En d'autres termes, $\mathcal{E}_{min}(\vec{m}, t_o)$ possède les séquences minimales de transitions inobservables qui permettent à t_o d'être validée à partir de \vec{m} . Un algorithme pour calculer $\mathcal{E}_{min}(\vec{m}, t_o)$ est donné par [4].

L'ensemble des marquages de base de \vec{m} et t_o est défini comme $\mathcal{M}(\vec{m}, t_o) = \{\vec{m}_b \in \mathcal{RC}(\mathcal{N}_l) : (\exists r_{uo} \in \mathcal{E}_{min}(\vec{m}, t_o))(\vec{m}[r_{uo}t_o]\vec{m}_b)\}$. Remarquons que $\mathcal{M}(\vec{m}, t_o)$ est l'ensemble des marquages que \mathcal{N}_l peut atteindre après avoir franchi une explication minimale puis la transition t_o à partir de \vec{m} . Nous étendons également la définition des marquages de base aux séquences d'événements observables $s_o \in \Sigma_o^*$ en utilisant la notation $\mathcal{M}_{seq}(\vec{m}, s_o)$, qui est définie à l'aide de la récursivité suivante : (i) $\mathcal{M}_{seq}(\vec{m}, \varepsilon) = \{\vec{m}\}$; (ii) $\mathcal{M}_{seq}(\vec{m}, \sigma_o) = \bigcup_{t_o \in T_o : \ell(t_o) = \sigma_o} \mathcal{M}(\vec{m}, t_o)$, pour $\sigma_o \in \Sigma_o$; (iii) $\mathcal{M}_{seq}(\vec{m}, s_o\sigma_o) = \bigcup_{\vec{m}_b \in \mathcal{M}_{seq}(\vec{m}, s_o)} \mathcal{M}_{seq}(\vec{m}_b, \sigma_o)$, pour $s_o \in \Sigma_o^*$ et $\sigma_o \in \Sigma_o$.

Sous l'hypothèse que le sous-réseau de \mathcal{N}_l induit par T_{uo} est acyclique, il est possible d'établir une relation entre l'ensemble des marquages de base $\mathcal{M}_{seq}(\vec{m}_0, s_o)$ et l'ensemble des marquages cohérents $C_o(\mathcal{N}_l, s_o)$, de la manière suivante.

Théorème 1 (Représentation du marquage de base [4]). *Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN. Alors, pour chaque séquence d'événements observables $s_o \in L_o(\mathcal{N}_l)$, il existe un marquage s_o -consistant $\vec{m}' \in C_o(\mathcal{N}_l, s_o)$ si et seulement s'il existe un marquage de base $\vec{m}_b \in \mathcal{M}_{seq}(\vec{m}, s_o)$ et une séquence de transitions non observables $r_{uo} \in T_{uo}^*$ tel que $\vec{m}_b[r_{uo}]\vec{m}'$. \square*

2.4 Graphe d'accessibilité de base

Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN borné. Afin d'analyser les marquages de base du LPN de manière plus efficace, nous pouvons construire son BRG, comme suit [4].

Définition 1 (Graphe d'accessibilité de base ou BRG). *Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN et Σ_o l'ensemble des événements observables. Alors, le BRG correspondant est désigné par $\mathcal{BG} = (M_G, A_G)$, où*

- $M_G = \{\vec{m} \in \mathcal{RC}(\mathcal{N}_l) : (\exists s_o \in L_o(\mathcal{N}_l))[\vec{m} \in \mathcal{M}_{seq}(\vec{m}_0, s_o)]\}$; et
- $A_G = \{(\vec{m}, \sigma_o, \vec{m}') \in M_G \times \Sigma_o \times M_G : \vec{m}' \in \mathcal{M}_{seq}(\vec{m}, \sigma_o)\}$. □

Notons que les nœuds du BRG sont les marquages accessibles du LPN qui sont également des marquages de base cohérents avec les séquences d'événements observables qui peuvent être observés dans le LPN. En outre, chaque arc du BRG est étiqueté par un événement observable dont l'observation conduit du marquage de base du nœud d'origine au marquage de base du nœud de destination.

3 Formulation du problème

Partitionnons l'ensemble Σ_{uo} en $\Sigma_{uo} = \Sigma_f \dot{\cup} \Sigma_n$, où Σ_f est l'ensemble des événements de fautes et Σ_n est l'ensemble des événements ordinaires non observables. Dans notre approche, nous supposons, sans perte de généralité, qu'il n'existe qu'un seul événement de faute, c'est-à-dire $\Sigma_f = \{\sigma_f\}$. S'il y a deux événements de fautes ou plus, l'approche présentée dans cet article peut être appliquée à chaque événement de fautes séparément et en considérant les événements de fautes restants comme des événements ordinaires non observables [15]. En outre, nous désignons le sous-ensemble des séquences d'événements de $L(\mathcal{N}_l)$ qui se terminent soit par un événement observable, soit par une séquence vide, par $L_{eo}(\mathcal{N}_l)$, qui est défini comme suit : $L_{eo}(\mathcal{N}_l) = \{s \in L(\mathcal{N}_l) : (s = \varepsilon) \vee ((\exists s' \in L(\mathcal{N}_l))(\exists \sigma_o \in \Sigma_o)[s = s'\sigma_o])\}$.

Notre objectif est de développer un pronostiqueur en ligne qui, étant donné une séquence d'événements observables $s_o \in L_o(\mathcal{N}_l)$, prend l'une des décisions suivantes : (i) l'événement de faute se produira inévitablement dans le futur ; (ii) l'événement de faute ne se produira pas ; ou (iii) l'événement de faute peut ou ne peut pas se produire. Formellement, ces trois scénarios sont définis comme des états du pronostiqueur, comme suit.

Définition 2 (États du pronostiqueur). *Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN borné. Un pronostiqueur est une application $\mathcal{P} : L_o(\mathcal{N}_l) \rightarrow \{F, N, I\}$, où, pour une séquence d'événements observables $s_o \in L_o(\mathcal{N}_l)$, les trois états possibles du pronostiqueur sont définis comme suit :*

- $\mathcal{P}(s_o) = F$ si $(\forall s \in L_{eo}(\mathcal{N}_l) \cap P_o^{-1}(s_o))(\exists k_s \in \mathbb{Z}_+^*)(\forall s' \in \Sigma^*)((ss' \in L(\mathcal{N}_l)) \wedge (|s'| \geq k_s) \vee (\{ss'\} \cap L(\mathcal{N}_l) = \emptyset)) \implies (\sigma_f \in s'))$;
- $\mathcal{P}(s_o) = N$ si $(\forall s \in L_{eo}(\mathcal{N}_l) \cap P_o^{-1}(s_o))(\forall s' \in \Sigma^*)((ss' \in L(\mathcal{N}_l)) \implies (\sigma_f \notin s'))$;
- $\mathcal{P}(s_o) = I$ si $\mathcal{P}(s_o) \neq F$ et $\mathcal{P}(s_o) \neq N$. □

Selon la Définition 2, pour chaque séquence d'événements $s \in L_{eo}(\mathcal{N}_l) \cap P_o^{-1}(s_o)$, nous analysons toutes les séquences d'événements s' dans la suite de s au sein de $L(\mathcal{N}_l)$, c'est-à-dire $ss' \in L(\mathcal{N}_l)$ qui soit conduisent à des marquages ne permettant aucune transition, soit sont suffisamment longues. Si tous les s' ont l'événement de faute, alors l'état du pronostiqueur est F , ce qui signifie que l'événement de faute se produira inévitablement. Inversement, si aucun s' a l'événement de faute, alors l'état du pronostiqueur est N , ce qui implique que l'événement de faute ne se produira pas. Enfin, si l'état du pronostiqueur n'est ni F ni N , alors l'état du pronostiqueur est I pour déduire que l'événement de faute pourra potentiellement se produire ou pas.

Afin d'aborder correctement le problème du pronostic en ligne des fautes des LPN, nous formulons les hypothèses supplémentaire suivantes :

- **A2.** Le LPN est borné.
- **A3.** Il existe un entier non négatif $k \in \mathbb{Z}_+$ tel qu'aucune séquence d'événements $s \in L(\mathcal{N}_l)$ ne comporte plus de k instances de l'événement de faute σ_f .

Les hypothèses **A1**, **A2** et **A3** sont nécessaires car elles garantissent que la copie du LPN du système que nous appelons FDPN, n'a pas non plus de cycles de transitions inobservables et qu'elle est bornée. Cela nous permet d'utiliser les notions de marquage de base et de BRG sur le FDPN afin d'effectuer le pronostic de fautes [4].

4 Pronostic en ligne des fautes des LPNs

Dans cette section, nous présentons notre approche pour le pronostic en ligne des fautes des LPNs bornés. Nous utilisons d'abord le LPN du système pour construire un Réseau de Petri piloté par les fautes (FDPN), dont les marquages accessibles sont liés à l'occurrence des fautes. Ensuite, nous utilisons le FDPN pour pronostiquer l'occurrence d'une faute à partir d'un marquage. Enfin, nous montrons comment le pronostic de fautes est implémenté en ligne.

4.1 Réseau de Petri piloté par les fautes

La première étape du pronostic de fautes est la construction du Réseau de Petri piloté par les fautes (FDPNs). Cette construction s'effectue comme suit.

Définition 3 (Réseau de Petri piloté par les fautes). *Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN. Le Réseau de Petri piloté par les fautes (FDPNs) de \mathcal{N}_l est un LPN $\mathcal{N}_f = (P_d, T, Pre_d, Post_d, \vec{m}_{d,0}, \Sigma, \ell)$, où*

- $P_d = P \cup \{p_d\}$;
- $m_{d,0}(p_d) = 0$ et, pour chaque $p \in P$, $m_{d,0}(p) = m_0(p)$;
- pour chaque transition $t \in T$:
 - pour chaque place $p \in P$, $Pre_d(p, t) = Pre(p, t)$ et $Post_d(p, t) = Post(p, t)$;
 - $Pre_d(p_d, t) = 0$;
 - $Post_d(p_d, t) = 1$ si $\ell(t) = \sigma_f$, ou 0 sinon. □

Étant donné un LPN \mathcal{N}_l , le FDPN \mathcal{N}_f est une copie de \mathcal{N}_l qui possède une place supplémentaire p_d , que nous appelons la place du détecteur de fautes. Selon la Définition 3, \mathcal{N}_f possède les mêmes transitions que \mathcal{N}_l , dont les tirs modifient les jetons des places de P de la même manière que dans \mathcal{N}_l ; cependant, pour chaque transition $t_f \in T$ étiquetée par l'événement de faute σ_f , le déclenchement de t_f dans \mathcal{N}_f ajoute un jeton à p_d afin d'indiquer l'occurrence de la faute. Puisque \mathcal{N}_f n'a aucune transition qui consomme des jetons de p_d , il n'est pas difficile de vérifier que (i) $LT(\mathcal{N}_f) = LT(\mathcal{N}_l)$, (ii) \mathcal{N}_f satisfait les hypothèses **A1** et **A2**, et (iii) toute séquence de transition donnée $r \in T^*$ qui peut se déclencher dans \mathcal{N}_l a une transition de faute si et seulement si le franchissement de r dans \mathcal{N}_f ajoute des jetons à p_d . Par exemple, le FDPN du LPN de la Figure 1 (a) est représenté dans la Figure 1 (b).

Notons que l'hypothèse **A3** sert à garantir que nous construisons des FDPN bornés. En effet, sans cette hypothèse, si une transition de faute pouvait se déclencher sans limite, elle pourrait ajouter un nombre illimité de jetons dans la place p_d , rendant donc le FDPN non borné.

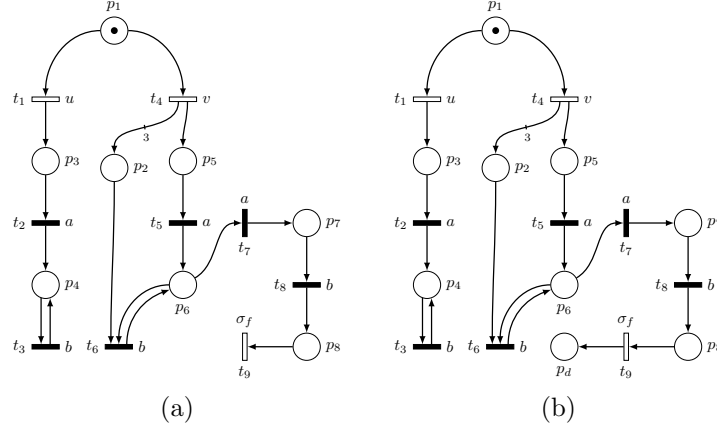


Figure 1: Un LPN (a) et son FDPN (b) correspondant, où $\Sigma_o = \{a, b\}$, $\Sigma_{uo} = \{u, v, \sigma_f\}$, et σ_f est l'évènement de faute.

4.2 Pronostic d'occurrence d'une faute à partir d'un marquage

Dans cette section, nous présentons deux conditions nécessaires et suffisantes qui nous permettent de pronostiquer les futures occurrences de fautes à partir de n'importe quel marquage donné du LPN. Cependant, avant de présenter ces conditions, nous définissons les marquages dits équivalents aux fautes, comme suit.

Définition 4 (Marquages fautifs équivalents). Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN, $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$ un marquage accessible \mathcal{N}_l , Σ_{uo} l'ensemble des événements non observables, $\sigma_f \in \Sigma_{uo}$ l'évènement de faute, et $\mathcal{N}_f = (P_d, T, Pre_d, Post_d, \vec{m}_{d,0}, \Sigma, \ell)$ le FDPN correspondant. Alors, l'ensemble des marquages fautifs équivalents de \vec{m} est défini par $M_{d,\vec{m}} = \{\vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f) : (\forall p \in P)(m_d(p) = m(p))\}$. \square

En d'autres termes, étant donné un marquage accessible $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$, l'ensemble correspondant aux marquages équivalents fautifs $M_{d,\vec{m}}$ contient tous les marquages accessibles de \mathcal{N}_f qui ont le même nombre de jetons que \vec{m} dans les composants associés aux places de P .

Dans la suite, nous établissons une relation entre le déclenchement des transitions de faute dans les séquences de transition $r \in T^*$ à partir des marquages atteignables $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$ et l'augmentation du nombre de jetons dans le composant p_d après le déclenchement de r à partir de marquages équivalents aux fautes $\vec{m}'_d \in M_{d,\vec{m}}$, comme suit.

Lemme 1. Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN, $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$ un marquage accessible de \mathcal{N}_l , $r \in T^*$ une séquence de transition telle que $\vec{m}[r]$, Σ_{uo} l'ensemble des événements non observables, $\sigma_f \in \Sigma_{uo}$ l'évènement de faute, $\mathcal{N}_f = (P_d, T, Pre_d, Post_d, \vec{m}_{d,0}, \Sigma, \ell)$ le FDPN correspondant, et $\vec{m}_d, \vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f)$ deux marquages accessibles de \mathcal{N}_f tels que $\vec{m}_d \in M_{d,\vec{m}}$ et $\vec{m}_d[r] \vec{m}'_d$. Alors $A \iff B$, où :

- $A : (\exists t_f \in T)[(\ell(t_f) = \sigma_f) \wedge (t_f \in r)]$; et
- $B : m_d(p_d) < m'_d(p_d)$.

Proof. ($A \implies B$). Supposons que la séquence de transitions $r \in T^*$ possède une transition de faute $t_f \in T$ telle que $\ell(t_f) = \sigma_f$. Alors, selon la Définition 3, r possède une transition

$t_f \in T$ étiquetée par σ_f et le déclenchement de t_f dans \mathcal{N}_f ajoute un jeton à la place p_d ; ainsi, puisqu'aucune transition de \mathcal{N}_f ne retire un jeton de la place p_d lors du déclenchement, il s'ensuit que la condition B est vraie.

($B \implies A$). Supposons que le marquage atteignable \vec{m}'_d ait plus de jetons dans la composante p_d que \vec{m}_d , c'est-à-dire $m_d(p_d) < m'_d(p_d)$. Alors, selon la Définition 3, il existe une transition $t_f \in T$ dans r étiquetée par σ_f ; ainsi, il s'ensuit que la condition A est vraie. \square

Nous sommes maintenant en mesure de montrer comment le FDPN peut être utilisé pour décider de l'occurrence future d'un événement de faute à partir d'un marquage donné $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$. Cette décision est prise avec les nœuds terminaux du RT $\Psi_{\vec{m}_d}$ du \mathcal{N}_f à partir d'un marquage fautif équivalent $\vec{m}_d \in M_{d,\vec{m}}$. Tout d'abord, nous présentons une condition nécessaire et suffisante qui utilise les nœuds terminaux de $\Psi_{\vec{m}_d}$ pour décider si un ou plusieurs jetons seront finalement ajoutés au composant $m_d(p_d)$, ce qui équivaut à dire que l'événement de faute se produira inévitablement à partir de \vec{m} , comme suit.

Théorème 2 (Confirmation d'un comportement erroné). *Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ une LPN, $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$ un marquage accessible de \mathcal{N}_l , Σ_{uo} l'ensemble des événements non observables, $\sigma_f \in \Sigma_{uo}$ l'événement de faute, $\mathcal{N}_f = (P_d, T, Pre_d, Post_d, \vec{m}_{d,0}, \Sigma, \ell)$ le FDPN correspondant et $\vec{m}_d \in M_{d,\vec{m}}$ tout marquage fautif de \vec{m} . Alors $A \iff B$, où :*

- $A : (\exists k \in \mathbb{Z}_+^*)(\forall r \in T^*)(\forall \vec{m}' \in \mathcal{RC}(\mathcal{N}_l))(((\vec{m}[r]\vec{m}') \wedge (|r| \geq k) \vee ((\forall t \in T)(\neg(\vec{m}'[t]))))) \implies ((\exists t_f \in T)(\ell(t_f) = \sigma_f \wedge (t_f \in r)))$; et
- $B : (\forall q \in Q_T(\Psi_{\vec{m}_d}))(\forall \vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f))((\vec{m}'_d = \mu(q)) \implies (m_d(p_d) < m'_d(p_d)))$.

Proof. Selon la Définition 3 et Lemma 1, la condition A est équivalent à la condition C , où

$$C : (\exists k \in \mathbb{Z}_+^*)(\forall r \in T^*)(\forall \vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f))(((\vec{m}_d[r]\vec{m}'_d) \wedge (|r| \geq k) \vee ((\forall t \in T)(\neg(\vec{m}'_d[t]))))) \implies (m_d(p_d) < m'_d(p_d)).$$

Ainsi, prouver $A \iff B$ est équivalent à prouver que $C \iff B$.

($C \implies B$). Soit $k \in \mathbb{Z}_+^*$ un entier positif qui satisfait à la condition C . En outre, faisons en sorte que $q \in Q_T(\Psi_{\vec{m}_d})$ soit un nœud terminal et $\vec{m}'_d = \mu(q)$ son marquage atteignable correspondant. Remarquons que q est soit un nœud de blocage, soit un nœud répétitif.

Considérons tout d'abord que q est un nœud de blocage. Alors, \vec{m}'_d correspond aux marquages qui peuvent être atteints à partir de \vec{m}_d , mais qui ne peuvent permettre aucune transition par la suite. Ainsi, selon la condition C , il s'ensuit que $m_d(p_d) < m'_d(p_d)$.

Considérons maintenant que q est un nœud répétitif. Remarquons que q est un nœud répétitif car il existe un nœud en amont q^R tel que $\mu(q^R) = \vec{m}'_d$. Cela implique que \vec{m}'_d peut être atteint après le déclenchement d'une séquence de transition plus longue que k , qui est formée par le déclenchement de la séquence de transition qui mène de \vec{m}_d à \vec{m}'_d et le déclenchement répétitif de la séquence de transition entre q^R et q ; par conséquent, conformément à la condition C , $m_d(p_d) < m'_d(p_d)$.

($B \implies C$). Supposons que la condition B soit vraie, et que $k \in \mathbb{Z}_+^*$ soit la hauteur de $\Psi_{\vec{m}_d}$, c'est-à-dire la longueur du plus long chemin possible dans le RT. Alors, toute séquence de transition $r \in T^*$ telle que $\vec{m}_d[r]$ et $|r| \geq k+1$ correspond à un chemin dans $\Psi_{\vec{m}_d}$ qui passe par un nœud répétitif de $\Psi_{\vec{m}_d}$; Par conséquent, conformément à la condition B , le déclenchement de r conduit de \vec{m}_d à un marquage comportant plus de jetons dans la composante p_d que \vec{m}_d . Si $|r| < k+1$, mais que le marquage atteint après r ne permet aucune transition, alors il correspond à un chemin qui mène à un nœud de blocage, ce qui, selon la condition B , implique

également que le déclenchement de r mène de \vec{m}_d à un marquage avec plus de jetons dans la composante p_d que \vec{m}_d ; et donc, la condition C est vraie. \square

En d'autres termes, l'événement de faute se produira inévitablement à partir de \vec{m} si, et seulement si, tous les nœuds terminaux de $\Psi_{\vec{m}_d}$ ont plus de jetons dans le composant p_d que \vec{m}_d .

Similaire au Théorème 2, nous établissons également une condition nécessaire et suffisante pour utiliser $\Psi_{\vec{m}_d}$ afin de déterminer si l'événement de faute ne se produira pas à partir de \vec{m} , comme suit.

Théorème 3 (Confirmation d'un comportement normal). *Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN, $\vec{m} \in \mathcal{RC}(\mathcal{N}_l)$ un marquage accessible de \mathcal{N}_l , Σ_{uo} l'ensemble des événements non observables, $\sigma_f \in \Sigma_{uo}$ l'événement de faute, $\mathcal{N}_f = (P_d, T, Pre_d, Post_d, \vec{m}_{d,0}, \Sigma, \ell)$ le FDPN correspondant et $\vec{m}_d \in M_{d,\vec{m}}$ tout marquage fautif de \vec{m} . Alors $A \iff B$, où :*

- $A : (\forall r \in T^*)((\vec{m}[r]) \implies ((\forall t_f \in T)((\ell(t_f) \neq \sigma_f) \vee (t_f \notin r))));$
- $B : (\forall q \in Q_T(\Psi_{\vec{m}_d}))(\forall \vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f))((\vec{m}'_d = \mu(q)) \implies (m_d(p_d) = m'_d(p_d))).$

Proof. D'après la contraposée du Lemme 1, pour toute séquence de transition $r \in T^*$ et tout marquage $\vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f)$ tel que $\vec{m}_d[r]\vec{m}'_d$, la bi-implication suivante est vraie :

$$((\forall t_f \in T)((\ell(t_f) \neq \sigma_f) \vee (t_f \notin r))) \iff (m_d(p_d) \geq m'_d(p_d)).$$

Puisqu'aucune transition de \mathcal{N}_f ne retire de jetons de la place p_d lors de l'exécution, il s'ensuit que $m_d(p_d) \leq m'_d(p_d)$; par conséquent, la bi-implication ci-dessus est équivalente à :

$$((\forall t_f \in T)((\ell(t_f) \neq \sigma_f) \vee (t_f \notin r))) \iff (m_d(p_d) = m'_d(p_d)).$$

D'après la bi-implication ci-dessus et la Définition 3, la condition A est équivalente à la condition C , où

$$C : (\forall r \in T^*)(\forall \vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f))((\vec{m}_d[r]\vec{m}'_d) \implies (m_d(p_d) = m'_d(p_d))).$$

Ainsi, prouver que $A \iff B$ est équivalent à prouver que $C \iff B$.

$(C \implies B)$. Si B est faux, alors il existe un nœud terminal $q \in Q_T(\Psi_{\vec{m}_d})$ étiqueté par un marquage accessible avec plus de jetons dans p_d que \vec{m}_d . Remarquons que l'existence de q implique qu'il existe une séquence de transitions $r \in T^*$ dont le déclenchement conduit de \vec{m}_d à un marquage comportant plus de jetons dans p_d que \vec{m}_d , ce qui implique que C est faux.

$(B \implies C)$. Si C est faux, alors il existe une séquence $r \in T^*$ dont le tir à partir de \vec{m}_d conduit à un marquage $\vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f)$ avec plus de jetons dans p_d que \vec{m}_d . L'existence de r et de \vec{m}'_d implique qu'il existe un nœud $q \in Q$ étiqueté par un marquage accessible qui a plus de jetons dans p_d que \vec{m}_d . Enfin, puisque les transitions de \mathcal{N}_f ne suppriment pas de jetons de p_d , il existe un chemin dans $\Psi_{\vec{m}_d}$ qui mène de q à un nœud terminal $q_T \in Q_T(\Psi_{\vec{m}_d})$ qui a également plus de jetons dans p_d que \vec{m}_d , ce qui implique que B est faux. \square

Exemple 1. *Considérons le LPN de la Figure 1(a) et son FDPN correspondant de la Figure 1(b). D'autre part, nous considérons le marquage accessible $\vec{m}_1 = 3p_2 + p_6$ du LPN, le marquage fautif équivalent $\vec{m}_{d,1} = \vec{m}_1$ tel que $\vec{m}_{d,1} \in M_{d,\vec{m}_1}$, et le RT $\Psi_{\vec{m}_{d,1}}$ représenté dans la Figure 2 (a). Notons que tous les nœuds terminaux de $\Psi_{\vec{m}_{d,1}}$, à savoir q_9, q_{12}, q_{14} , et q_{15} , sont étiquetés par des marquages qui ont un jeton de plus dans la composante p_d par rapport à $\vec{m}_{d,1}$; ainsi, selon le Théorème 2, l'événement de faute se produira forcément dans le futur après l'obtention du marquage \vec{m}_1 .*

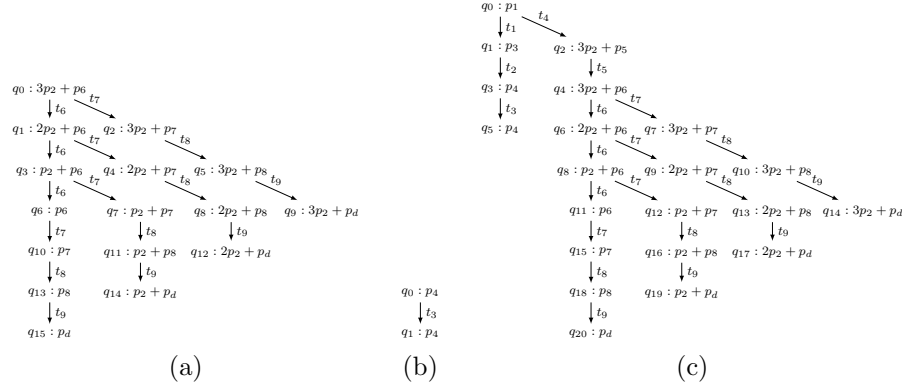


Figure 2: Les RTs du FDPN de la Figure 1 à partir des marquages $\vec{m}_{d,1} = 3p_2 + p_6$ (a), $\vec{m}_{d,2} = p_4$ (b), et $\vec{m}_{d,0} = p_1$ (c).

Considérons maintenant le marquage $\vec{m}_2 = p_4$, son marquage fautif équivalent $\vec{m}_{d,2} = \vec{m}_2$, et RT $\Psi_{\vec{m}_{d,2}}$ représenté par la Figure 2(b). Notons que ce RT ne possède qu'un seul nœud terminal q_1 qui est également étiqueté par $\vec{m}_{d,2}$. En d'autres mots, tous les nœuds terminaux de $\Psi_{\vec{m}_{d,2}}$ ont le même nombre de jeton que $\vec{m}_{d,2}$; ainsi, selon le Théorème 3, l'événement de faute ne surviendra pas suite au marquage \vec{m}_2 . \square

4.3 Pronostic en ligne des fautes

Dans cette section, nous présentons un algorithme permettant de décider en ligne de l'état du pronostiqueur au fur et à mesure que des événements sont observés pendant le fonctionnement du système. Tout d'abord, nous utilisons les Théorèmes 1, 2, et 3 pour établir une relation entre les états de pronostic de n'importe quel LPN et les marquages de base du BRG correspondant au FDPN, comme suit.

Définition 5 (États du pronostiqueur à partir des marquages de base). Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN, $\mathcal{N}_f = (P_d, T, Pre_d, Post_d, \vec{m}_{d,0}, \Sigma, \ell)$ le FDPN correspondant et $\mathcal{BG} = (M_G, A_G)$ le BRG de \mathcal{N}_f . Les états du pronostiqueur à partir des marquages de base sont donnés par l'application $\mathcal{P}_B : M_G \rightarrow \{F, N, I\}$, où, pour un marquage de base $\vec{m}_{d,b} \in M_G$, les trois états possibles du pronostiqueur sont définis comme suit :

- $\mathcal{P}_B(\vec{m}_{d,b}) = F$ si $(\forall q \in Q_T(\Psi_{\vec{m}_{d,b}}))(\forall \vec{m}' \in \mathcal{RC}(\mathcal{N}_f))((\vec{m}' = \mu(q)) \implies (m_{d,b}(p_d) < m'(p_d)))$;
- $\mathcal{P}_B(\vec{m}_{d,b}) = N$ si $(\forall q \in Q_T(\Psi_{\vec{m}_{d,b}}))(\forall \vec{m}' \in \mathcal{RC}(\mathcal{N}_f))((\vec{m}' = \mu(q)) \implies (m_{d,b}(p_d) = m'(p_d)))$; et
- $\mathcal{P}_B(\vec{m}_{d,b}) = I$ si $\mathcal{P}_B(\vec{m}_{d,b}) \neq F$ et $\mathcal{P}_B(\vec{m}_{d,b}) \neq N$. \square

Comme le montre le Théorème 4, la Définition 5 peut être utilisée avec l'ensemble des marquages de base de \mathcal{N}_f compatibles avec s_o pour décider de l'état de pronostic $\mathcal{P}(s_o)$ pour toute séquence d'événements observables donnée $s_o \in L_o(\mathcal{N}_l)$.

Théorème 4 (Pronostic de fautes à l'aide de marquages de base). *Soit $\mathcal{N}_l = (P, T, Pre, Post, \vec{m}_0, \Sigma, \ell)$ un LPN, Σ_{uo} l'ensemble des événements non observables, $\sigma_f \in \Sigma_{uo}$ l'événement de faute, et $\mathcal{N}_f = (P_d, T, Pre_d, Post_d, \vec{m}_{d,0}, \Sigma, \ell)$ le FDPN correspondant. Alors, l'état du pronostiqueur pour une séquence d'événements observables donnée $s_o \in L_o(\mathcal{N}_l)$ peut être déterminé comme suit :*

- $\mathcal{P}(s_o) = F$ si et seulement si $(\forall \vec{m}_{d,b} \in \mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o))(\mathcal{P}_B(\vec{m}_{d,b}) = F)$;
- $\mathcal{P}(s_o) = N$ si et seulement si $(\forall \vec{m}_{d,b} \in \mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o))(\mathcal{P}_B(\vec{m}_{d,b}) = N)$; et
- $\mathcal{P}(s_o) = I$ si et seulement si $\mathcal{P}(s_o) \neq F$ et $\mathcal{P}(s_o) \neq N$.

Proof. Soit $s_o \in L_o(\mathcal{N}_l)$ une séquence d'événements observables. Nous prouvons chaque bi-implication séparément.

$(\mathcal{P}(s_o) = F)$. Selon la Définition 2, $\mathcal{P}(s_o) = F$ si, et seulement si, l'événement fautif se produira inévitablement après avoir observé s_o . D'autre part, selon la relation entre \mathcal{N}_l et \mathcal{N}_f décrite par la Définition 3 et le Lemme 1, $\mathcal{P}(s_o) = F$ si et seulement si chaque marquage $\vec{m}_d \in C_o(\mathcal{N}_f, s_o)$ compatible avec des marquages accessibles $\vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f)$ avec plus de jetons dans p_d que \vec{m}_d après le franchissement de séquences de transitions suffisamment longues ou de séquences de transitions conduisant à un marquage bloquant.

Selon le Théorème 1, tous les marquages $\vec{m}_d \in C_o(\mathcal{N}_f, s_o)$ consistent avec s_o peuvent être atteints à partir de marquages de base $\vec{m}_{d,b} \in \mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ après le franchissement de transitions inobservables ; par conséquent, la bi-implication susmentionnée est équivalente à l'affirmation que $\mathcal{P}(s_o) = F$ si, et seulement si, chaque marquage de base $\vec{m}_{d,b} \in \mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ atteint les marquages $\vec{m}'_d \in \mathcal{RC}(\mathcal{N}_f)$ avec plus de jetons dans p_d que $\vec{m}_{d,b}$ après les franchissements de séquences de transitions suffisamment longues ou de séquences de transition qui conduisent à des marquages bloquants. D'après le Théorème 2, l'affirmation précédente est vraie pour tout marquage de base $\vec{m}_{d,b} \in \mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ si et seulement si tous les nœuds terminaux des RTs $\Psi_{\vec{m}_{d,b}}$ sont étiquetés par des marquages accessibles avec plus de jetons dans p_d que $\vec{m}_{d,b}$. Enfin, l'assertion susmentionnée est vraie si, et seulement si, tous les états de pronostic des marquages de base de $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ sont F , ce qui est équivalent à la première bi-implication de ce théorème.

$(\mathcal{P}(s_o) = N)$. La preuve de la seconde bi-implication de ce théorème est similaire à celle de la première ; cependant, nous considérons la seconde bi-implication de la Définition 2 au lieu de la première, le Théorème 3 au lieu du Théorème 2, et que les états pronostiqués des marquages de base sont tous égaux à N au lieu de F .

$(\mathcal{P}(s_o) = I)$. Puisque les deux premières bi-implications de ce théorème sont vraies, il s'ensuit que $\mathcal{P}(s_o) = I$ si et seulement si $\mathcal{P}(s_o) \neq F$ et $\mathcal{P}(s_o) \neq N$. \square

En d'autres termes, étant donné une séquence d'événements $s_o \in L_o(\mathcal{N}_l)$ et son ensemble correspondant de marquages de base $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$, l'état du pronostiqueur est : (i) F , si tous les états pronostiqués des marquages de base de $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ sont également F ; (ii) N , si tous les états pronostiqués des marquages de base de $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ sont N ; ou (iii) I , si aucune des conditions ci-dessus n'est vraie.

Afin d'utiliser le Théorème 4 pour effectuer le pronostic de faute en ligne, nous utilisons l'Algorithme 1. Il comporte une phase hors ligne (Étapes 1 à 4) et une phase en ligne (Étapes 5 à 16).

Dans la phase hors ligne, l'Algorithme 1 calcule la séquence d'événements observée initiale $s_o = \varepsilon$ et l'ensemble initial de marquages de base $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ en tant qu'ensemble ne comportant que le marquage initial du FDPN (étapes 1 et 2). Dans la suite, l'algorithme calcule

Algorithme 1: Pronostic de défaillance en ligne à l'aide du BRGs.

Entrée : FDPN $\mathcal{N}_f = (P_d, T_d, Pre_d, Post_d, \vec{m}_{d,0}, \Sigma_d, \ell_d)$.

```

1   $s_o \leftarrow \varepsilon$ ;
2   $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o) \leftarrow \{\vec{m}_{d,0}\}$ ;
3  Construire  $\mathcal{BG} = (M_G, A_G)$  le BRG de  $\mathcal{N}_f$  conformément à la Définition 1;
4  pour chaque  $\vec{m}_{d,b} \in M_G$ , calculer  $\mathcal{P}_B(\vec{m}_{d,b})$  conformément à la Définition 5;
5  tant que le pronostiqueur fonctionne faire
6      si pour tout  $\vec{m}_{d,b} \in \mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ ,  $\mathcal{P}_B(\vec{m}_{d,b}) = F$  alors
7          | poser  $\mathcal{P}(s_o) \leftarrow F$ ;
8      sinon si pour tout  $\vec{m}_{d,b} \in \mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o)$ ,  $\mathcal{P}_B(\vec{m}_{d,b}) = N$  alors
9          |  $\mathcal{P}(s_o) \leftarrow N$ ;
10     sinon
11         |  $\mathcal{P}(s_o) \leftarrow I$ ;
12     fin
13     attendre l'observation d'un événement  $\sigma_o \in \Sigma_o$ ;
14     calculer  $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o\sigma_o) = \{\vec{m}'_{d,b} \in M_G : (\exists \vec{m}_{d,b} \in \mathcal{M}_{seq,d}(\vec{m}_{d,0}, s_o))[(\vec{m}_{d,b}, \sigma_o, \vec{m}'_{d,b}) \in A_G]\}$ ;
15      $s_o \leftarrow s_o\sigma_o$ ;
16 fin
```

également le BRG correspondant du FDPN et l'état du pronostic pour chaque marquage de base du BRG (Étapes 3-4).

Après les phases hors ligne de l'Algorithme 1, l'algorithme commence sa phase en ligne en décidant de l'état du pronostiqueur $\mathcal{P}(s_o)$ avec les Étapes 6-12, qui effectuent les vérifications décrites dans le Théorème 4. Dans la suite, les Étapes 13-15 attendent une nouvelle observation d'événement $\sigma_o \in \Sigma_o$, ce qui amène l'algorithme à utiliser le BRG pour mettre à jour l'ensemble actuel de marquage de base du FDPN et la séquence d'événements observée.

Exemple 2. Utilisons l'algorithme 1 pour effectuer le pronostic de défaut en ligne du LPN de la Fig. 1 (a), dont le FDPN est représenté sur la Fig. 1 (b). L'algorithme 1 calcule d'abord s_o comme ε , et l'ensemble initial de marquage de base $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, \varepsilon)$ comme $\{p_1\}$ (étapes 1-2). Dans la suite, l'algorithme 1 utilise les étapes 3-4 pour calculer le BRG correspondant du FDPN et l'état du pronostiqueur pour chaque marquage de base du BRG. La figure 3 montre le BRG obtenu et l'état du pronostic pour chaque nœud.

Après l'étape 4, l'algorithme entame sa phase en ligne, qui commence par l'utilisation du théorème 4 pour décider de l'état du pronostiqueur $\mathcal{P}(\varepsilon)$ (étape 6-12). Comme le montre la figure 3, l'état pronostiqué du marquage de base p_1 est $\mathcal{P}_B(p_1) = I$. Cet état de pronostic a été décidé hors ligne par l'étape 4, qui a vérifié que le RT Ψ_{p_1} représenté dans la figure 2 (c) a, à la fois, des nœuds terminaux avec plus de jetons dans p_d que dans p_1 et des nœuds terminaux avec le même nombre de jetons dans p_1 . Ainsi, aucune des conditions des étapes 6 et 8 n'étant vraie, cela amène l'algorithme à utiliser l'étape 11 pour décider que $\mathcal{P}(\varepsilon) = I$.

Considérons maintenant que la séquence d'événements aa ait été observée. En utilisant le BRG l'algorithme 1 utilise 2 itérations des étapes 13-15 pour mettre à jour s_o en aa et $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, aa)$ en $\{p_3+p_7\}$. Remarquons que le pronostic de l'état p_3+p_7 est $\mathcal{P}_B(p_3+p_7) = F$; ainsi, la condition de l'étape 6 est vraie, ce qui amène l'Algorithme 1 à utiliser l'étape 7 pour décider que $\mathcal{P}(aa) = F$.

Enfin, considérons que la séquence d'événements ab^4 ait été observée au lieu de aa . L'algorithme 1 utilise 5 itérations des étapes 13-15 pour mettre à jour s_o en ab^4 et $\mathcal{M}_{seq,d}(\vec{m}_{d,0}, ab^4)$ en $\{p_4\}$. Remarquons que l'état du pronostic de p_4 est $\mathcal{P}_B(p_4) = N$; ainsi, la condition de l'étape 8 est vraie, ce qui amène l'algorithme 1 à utiliser l'étape 9 pour décider que $\mathcal{P}(ab^4) = N$. \square

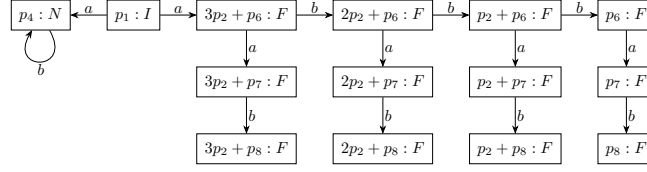


Figure 3: Le BRG du FDPN de la Figure 2, où chaque état est étiqueté par la décision de son pronostic.

4.4 Analyse de la complexité algorithmique

L'Algorithme 1 nécessite le calcul des RTs pour chaque marquage de base du BRG correspondant au FDPN. Bien que ce calcul puisse devenir important pour certains LPNs, il est effectué pendant sa phase hors ligne. Par conséquent, la phase en ligne de l'Algorithme 1 ne nécessite pas le calcul des arbres comme dans [8]. En effet, la phase en ligne peut obtenir l'état du pronostiqueur directement à partir des marquages de base compatibles avec la séquence d'événements observée. En outre, le système BRG peut être utilisé pour calculer l'ensemble des marquages de base cohérents de manière efficace. Par conséquent, la complexité de calcul en ligne de l'Algorithme 1 est proportionnel à l'estimation des marquages de base cohérents avec la séquence observée. Cette estimation faite à l'aide du BRG, a donc une complexité en $O(|M_G| \times n_T)$, où $|M_G|$ est le nombre de nœuds du BRG et n_T est le nombre de transitions. Il est important de noter que la complexité obtenue ici est meilleure que celle de l'approche de [8], car le calcul de la complexité des arbres de couverture est un problème ouvert, ce qui empêche de calculer la complexité de l'approche de [8].

5 Conclusion

Dans cet article, nous proposons une nouvelle approche pour le pronostic des fautes de SEDs modélisés par des LPNs bornés. En s'appuyant sur les concepts de marquage de base et de RTs, nous présentons des conditions nécessaires et suffisantes qui garantissent que le pronostic de fautes est effectué correctement. Contrairement à [8], l'approche présentée ici est spécifique au pronostic de fautes de LPNs bornés. Elle est donc moins générale mais d'une meilleure complexité car les calculs les plus complexes sont effectués hors ligne. Elle est donc mieux appropriée pour les applications temps réel.

Notons que cette approche de prédiction d'une faute, n'est pas spécifique au pronostic. Elle pourrait être généralisée à la prédiction de n'importe quel événement donné que l'utilisateur voudrait surveiller.

En perspective de ce travail, nous souhaitons encore améliorer l'efficacité de cette approche, en la basant sur la construction unique d'un seul arbre d'accessibilité utilisable quel que soit le marquage de base donné. D'autre part, nous souhaitons identifier les classes de LPNs non bornés dont l'arbre de couverture pourrait être calculé hors ligne et utilisé sur le même principe que les RTs des LPNs bornés. En outre, nous désirons étendre cette approche afin de traiter le pronostic des LPNs temporisés et stochastiques.

References

- [1] Rabah Ammour, Edouard Leclercq, Eric Sanlaville, and Dimitri Lefebvre. Fault prognosis of timed stochastic discrete event systems with bounded estimation error. *Automatica*, 82:35–41, 2017.
- [2] Rabah Ammour, Edouard Leclercq, Eric Sanlaville, and Dimitri Lefebvre. Faults prognosis using partially observed stochastic Petri-nets: an incremental approach. *Discrete Event Dynamic Systems*, 28:247–267, 2018.
- [3] Raphael Julio Barcelos and João Carlos Basilio. Disjunctive fault prediction of decentralized discrete event systems: Verification, predictor design and k copredictability. *Automatica*, 148:110769, 2023.
- [4] Maria Paola Cabasino, Alessandro Giua, Marco Pocci, and Carla Seatzu. Discrete event diagnosis using labeled Petri nets. an application to manufacturing systems. *Control Engineering Practice*, 19(9):989–1001, 2011.
- [5] Jun Chen and Ratnesh Kumar. Stochastic failure prognosability of discrete event systems. *IEEE Transactions on Automatic Control*, 60(6):1570–1581, 2014.
- [6] Jun Chen and Ratnesh Kumar. Stochastic failure prognosis of discrete event systems. *IEEE Transactions on Automatic Control*, 67(10):5487–5492, 2021.
- [7] Amira Chouchane and Mohamed Ghazel. Fault-prognosability, K-step prognosis and K-step predictive diagnosis in partially observed Petri nets by means of algebraic techniques. *Automatica*, 162:111513, 2024.
- [8] Braian I de Freitas, Armand Toguyeni, and João C Basilio. Online fault prognosis of labeled Petri nets. *IEEE Control Systems Letters*, 2025.
- [9] Sahika Genc and Stéphane Lafortune. Predictability of event occurrences in partially-observed discrete-event systems. *Automatica*, 45(2):301–311, 2009.
- [10] Redouane Kanazy, Samir Chafik, and Eric Niel. Prognosis of failure events based on labeled temporal Petri nets. *Advances in Science, Technology and Engineering Systems Journal*, 5(3):432–441, 2020.
- [11] Ratnesh Kumar and Shigemasa Takai. Decentralized prognosis of failures in discrete event systems. *IEEE Transactions on Automatic Control*, 55(1):48–59, 2009.
- [12] Dimitri Lefebvre. Fault diagnosis and prognosis with partially observed Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(10):1413–1424, 2014.
- [13] Ziyue Ma, Xiang Yin, and Zhiwu Li. Marking predictability and prediction in labeled Petri nets. *IEEE Transactions on Automatic Control*, 66(8):3608–3623, 2020.
- [14] T. Murata. Petri nets: properties, analysis, and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.
- [15] Leonardo PM Santoro, Marcos V Moreira, and João C Basilio. Computation of minimal diagnosis bases of discrete-event systems using verifiers. *Automatica*, 77:93–102, 2017.
- [16] Ana TY Watanabe, Renan Sebem, Andre B Leal, and Marcelo da S Hounsell. Fault prognosis of discrete event systems: An overview. *Annual Reviews in Control*, 51:100–110, 2021.
- [17] Wenqing Wu, Xiang Yin, and Shaoyuan Li. Verification of coprognosability in decentralized fault prognosis of labeled Petri nets. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 4845–4850. IEEE, 2018.
- [18] Xiang Yin. Verification of prognosability for labeled Petri nets. *IEEE Transactions on Automatic Control*, 63(6):1828–1834, 2017.
- [19] Dan You, ShouGuang Wang, and Carla Seatzu. Verification of fault-predictability in labeled Petri nets using predictor graphs. *IEEE Transactions on Automatic Control*, 64(10):4353–4360, 2019.