

IDENTIFICATION DES RESEAUX DE PETRI : APPLICATION AU DIAGNOSTIC DES SED

SOULEIMAN OULD EL MEDHI, EDOUARD LECLERCQ, DIMITRI LEFEBVRE

GREAH - Université le Havre
25 rue P. Lebon, 76058 Le Havre Cedex
{soupleiman6 ; edouard.leclercq ; dimitri.lefebvre}@univ-lehavre.fr

RESUME : *Le diagnostic des systèmes à événements discrets peut être réalisé à l'aide de modèles par automates finis ou par réseaux de Petri. Mais les modèles de référence sont difficiles à obtenir en pratique car ils supposent une connaissance a priori des dysfonctionnements et une analyse exhaustive des modes dégradés. Ce travail s'intéresse à l'identification des modèles de référence par réseaux de Petri, obtenus à partir des séquences d'alarmes enregistrées sur le système. Un algorithme d'apprentissage en ligne est présenté. Il permet d'actualiser le modèle en fonction de chaque nouvel événement enregistré. L'apprentissage est supervisé et basé sur la rétro-propagation d'une erreur associée à la causalité des événements. Une procédure d'élagage permet de réduire la taille des modèles obtenus.*

MOTS - CLES : *réseaux de Petri, identification de modèles, graphes d'état, diagnostic*

1. INTRODUCTION

La détection et le diagnostic des défauts sont des enjeux très importants pour les systèmes industriels et les réseaux de communication. Parmi les approches existantes, les méthodes de diagnostic par comparaison avec un modèle de référence ont été largement étudiées et utilisées dans de nombreuses applications. Dans le cas des systèmes à événements discrets (SED), des modèles de référence par automates ou réseaux de Petri (RdP) sont utilisés pour réaliser le diagnostic et analyser la propagation des défauts (Alcaraz-Mejia *et al.*, 2003 ; Sampath *et al.*, 1995 ; Ushio *et al.*, 1998 ; Lefebvre *et al.*, 2005). Ces modèles sont difficiles à construire car ils nécessitent une connaissance exhaustive des défauts qui affectent le système, et il existe peu de méthodes de synthèse permettant d'adapter le modèle en fonction des nouveaux défauts apparus au fil du temps. C'est pourquoi nous nous intéressons au problème de la synthèse et de l'identification des modèles événementiels en vue du diagnostic (Bulitko *et al.*, 2005 ; Bourdeaud'huy *et al.*, 2004 ; Giu *et al.*, 2005 ; Meda-Campana *et al.*, 2002 ; Meda-Campana *et al.*, 2003).

Cette étude est un enjeu majeur dans les systèmes de propagation d'alarmes comme ceux utilisés dans les réseaux de télécommunication car elle permet de mieux expliquer les causes des alarmes, elle permet de supprimer les alarmes redondantes et enfin elle permet de prévoir l'apparition d'alarmes sévères mettant en péril l'intégrité du système (Mannila *et al.*, 1997).

Notre objectif final est de proposer une méthode systématique pour construire des modèles de référence de fonctionnement et de dysfonctionnement en fonction des séquences d'alarmes enregistrées sur le système. Les

contributions de cet article concernent l'identification des modèles de référence par RdP interprétés dont les transitions correspondent aux alarmes de la séquence. Dans nos travaux précédents (Ould El Mehdi *et al.*, 2006 ; 2007) nous avons exploité la causalité induite par l'ordre d'apparition des événements pour construire le modèle. Dans cet article, nous montrons comment le modèle construit peut être utilisé pour détecter et diagnostiquer les défauts et réaliser ainsi l'analyse des séquences d'alarmes.

L'article est composé de 4 parties. La seconde partie concerne des rappels sur la modélisation des SED par réseaux de Petri. La troisième partie présente l'algorithme que nous proposons pour construire systématiquement des modèles par réseaux de Petri capables de reproduire des séquences d'événements. Enfin la quatrième partie illustre la méthode pour la modélisation et le diagnostic d'un système à deux réservoirs.

2. MODELISATION DES SED PAR RESEAUX DE PETRI

2.1. Réseaux de Petri ordinaires

Un réseau de Petri (RdP) ordinaire est un graphe orienté biparti défini par le quadruplet $\langle P, T, W_{PR}, W_{PO} \rangle$ tel que P, T sont deux ensembles finis de sommets dénommés places et transitions avec $|P| = n, |T| = p$ et $P \cap T = \emptyset$. $W_{PR} = (w_{PR_{ij}}) \in \{0, 1\}^{n \times p}$ avec $w_{PR_{ij}} = \text{Pre}(P_i, T_j)$ où $\text{Pre}(P_i, T_j)$ est le poids de l'arc dirigé de P_i vers T_j , W_{PO} est l'application d'incidence avant et $W_{PO} = (w_{PO_{ij}}) \in \{0, 1\}^{n \times p}$ avec $w_{PO_{ij}} = \text{Post}(P_i, T_j)$, où $\text{Post}(P_i, T_j)$ est le poids de l'arc dirigé de T_j vers P_i , W_{PO} est l'application d'incidence arrière (David *et al.*, 1992). La matrice d'incidence W du réseau est définie par $W = W_{PO} - W_{PR} \in \{-1, 0, 1\}^{n \times p}$. Le marquage M du RdP est

l'application de l'ensemble des places P vers l'ensemble des entiers naturels Z^+ associant à chaque place $P_i \in P$ un entier $M(P_i)$ appelé marquage de P_i , et $M_I \in (Z^+)^n$ représente le marquage initial du réseau de Petri. Une séquence de tirs $\sigma = T_i T_j \dots T_k$ est définie comme une série ordonnée de transitions qui sont franchies lorsque le RdP évolue du marquage M au marquage M' tel que la relation (1) soit satisfaite :

$$M \xrightarrow{T_i} M_1 \xrightarrow{T_j} M_2 \rightarrow \dots \rightarrow M' \quad (1)$$

Une séquence σ peut être représentée par son vecteur caractéristique (i.e. vecteur de Parikh) $X = (x_j) \in (Z^+)^q$ où x_j représente le nombre d'occurrences du franchissement de la transition T_j dans la séquence σ (David *et al.*, 1992). Le marquage M' obtenu à partir du marquage initial M par réalisation de la séquence σ est donné par l'équation (2):

$$\Delta M = M' - M = W.X \quad (2)$$

Le graphe d'atteignabilité $R(\text{RdP}, M_I)$ est l'ensemble des marquages M tels qu'il existe une séquence σ permettant d'obtenir M à partir de M_I , que l'on notera $M_I [\sigma > M$. Une séquence σ est dite réalisable à partir de M_I s'il existe un couple de marquages $(M, M') \in R(\text{RdP}, M_I)$ tel que $M [\sigma > M'$.

2.2. Relations de propagation dans un RdP

L'analyse structurelle des modèles par RdP peut être obtenue en étudiant les chemins dirigés qui caractérisent la propagation des événements (Lefebvre *et al.*, 2005). Un chemin dirigé (CD) existe de la transition T_k à la transition T_j si une marque peut se déplacer en franchissant T_k puis T_j . Le rang de la relation de propagation entre T_k et T_j correspond à la longueur du chemin dirigé de taille minimale (i.e. le nombre de places intermédiaires dans le plus court CD) entre T_k et T_j . Quand il n'existe pas de CD entre les 2 transitions, la longueur est par convention égale à l'infini. La longueur du CD de taille minimale allant de T_k à T_j dans l'intervalle $I = [r_{\min}, r_{\max}] \cup \{\infty\}$ est caractérisée par l'équation (3):

$$CD(T_j, T_k, I) = \min\{r \in I, B_j^T ((W_{PR})^T . W_{PO})^r . B_k \neq 0\} \quad (3)$$

où : $B_j = (\delta_{jk}) \in \{0,1\}^p$, $k=1,2..p$, $C_i = (\delta_{ik}) \in \{0,1\}^n$, $k=1,2..n$, et δ représente le symbole de Kronecker (Lefebvre *et al.*, 2005). Le chemin dirigé de la place P_k à la transition T_j , celui de T_k à P_j , et celui de P_k à P_j sont définis de la même manière. Le rang des relations de propagation entre les éléments du RdP (places et transitions) dans l'intervalle $I = [r_{\min}, r_{\max}] \cup \{\infty\}$ est caractérisé par la matrice CD (RdP, I) définie par l'équation (4) et par la table 1 (Lefebvre *et al.*, 2005):

$$CD(RdP, I) = \begin{pmatrix} CD_{PP}(P_i, P_k, I) & CD_{PT}(P_i, T_k, I) \\ CD_{TP}(T_j, P_k, I) & CD_{TT}(T_j, T_k, I) \end{pmatrix} \in I^{(n+p) \times (n+p)} \quad (4)$$

Dans la suite de ce travail, on admettra $I = [0, \min(n, p)] \cup \{\infty\}$ car le rang des relations de propagation ne peut pas excéder le nombre de places ou transitions d'un RdP (Lefebvre *et al.*, 2005). Par ailleurs nous avons montré que le rang des relations de propagation est une distance entre les transitions et entre les places du RdP (Lefebvre *et al.*, 2005). Dans un souci de simplification, on omettra I .

CD	Rang des relations de propagation
$CD_{PP}(P_i, P_k, I)$	$\min\{r \in I, C_i^T (W_{PO} . (W_{PR})^T)^r . C_k \neq 0\}$
$CD_{PT}(P_i, T_k, I)$	$\min\{r \in I, C_i^T (W_{PO} . (W_{PR})^T)^r . W_{PO} . B_k \neq 0\}$
$CD_{TP}(T_j, P_k, I)$	$\min\{r \in I, B_j^T (W_{PR})^T . (W_{PO} . (W_{PR})^T)^r . C_k \neq 0\}$
$CD_{TT}(T_j, T_k, I)$	$\min\{r \in I, B_j^T ((W_{PR})^T . W_{PO})^r . B_k \neq 0\}$

Tableau 1 : Caractérisation des chemins dirigés

3. IDENTIFICATION DES RESEAUX DE PETRI

Une difficulté importante des méthodes de diagnostic par RdP est la détermination a priori d'un modèle de référence complet (fonctionnel ou dysfonctionnel). Cette détermination doit être suffisamment répétable pour permettre de construire le modèle de référence de manière systématique sans recourir à une connaissance experte du système pour chaque cas d'étude. Les réseaux de Petri sont bien adaptés pour la modélisation mais il n'existe pas de méthode systématique pour construire le modèle de référence à partir des séquences d'événements enregistrées sur le système étudié.

Plusieurs travaux existent cependant qui exploitent les séquences d'alarmes à l'aide d'une analyse statistique. Une méthode basée sur la recherche d'ensembles d'événements relativement proches temporellement et suivant un ordre partiel donné (parallèle ou série), nommés épisodes est présentée par Mannila (Mannila *et al.*, 1997). Cette recherche s'effectue en appliquant une fenêtre temporelle. L'utilisateur doit définir quelle est la durée maximale qui peut séparer deux événements considérés comme appartenant au même épisode. Le but de cette méthode est de définir des relations d'implication entre événements en fonction des fréquences d'occurrence des épisodes et de la relation d'ordre. Le modèle ainsi obtenu permet d'analyser les séquences d'alarmes et de prédire les alarmes futures les plus probables. Une approche similaire a été proposée par (Dousson *et al.*, 1999) qui introduit des modèles de chronique permettant de généraliser les relations d'ordre entre événements et de s'affranchir de l'utilisation d'une fenêtre temporelle. Ce modèle est connu sous la forme d'un graphe de contraintes temporelles que doivent satisfaire les événements d'un même épisode. Une autre méthode, enfin, proposée par (Bouché, 2005) effectue la recherche de signatures de séquences particulières en s'appuyant sur les processus de Poisson et sur les chaînes de Markov.

Le problème que nous allons étudier concerne la synthèse par apprentissage et l'identification de modèles à partir de séquences d'événements enregistrées sur le SED étudié. L'ordre est supposé total, induit par la chronologie des événements présents dans la séquence. Le modèle construit est un RdP interprété. Ses transitions et ses places seront associées respectivement aux événements et aux variables d'états caractéristiques du système considéré. Ces méthodes permettent d'obtenir des modèles de taille plus réduite que les méthodes de synthèse analytique (Ould El Mehdi *et al.*, 2006). Dans cet article, nous montrons aussi comment le modèle construit est utilisé pour détecter et diagnostiquer les défauts et réaliser ainsi l'analyse des séquences d'alarmes.

3.1. Notations

On définit une séquence ordonnée d'événements (défauts ou événements normaux) de taille K comme un vecteur ligne $S = [s(k)]$ de dimension K dont les éléments sont des nombres entiers qui représentent les événements dans l'ordre de leur apparition (par exemple, $S = [1\ 2\ 5\ 4]$ signifie que la succession d'événements $s(1) = e_1$, $s(2) = e_2$, $s(3) = e_5$, et $s(4) = e_4$ a été enregistrée et que l'événement e_1 est suivi de e_2 puis de e_5 et de e_4 selon la relation d'ordre). Le symbole 0 représente l'absence de causalité qui peut traduire une interruption d'enregistrement (par exemple, $S = [1\ 2\ 0\ 5\ 4]$ signifie qu'il n'y a pas de causalité entre les deux événements e_2 et e_5). L'entier p représente le nombre d'événements distincts présents dans la séquence de taille K (par exemple, pour $S = [1\ 2\ 0\ 5\ 4]$ $K = 5$ et $p = 4$). A chaque événement $s(k) = e_i$, $e_i \neq 0$, on associe la transition T_i dans le modèle RdP.

A partir de S nous considérons $A = (a_{ij}) \in (\mathbb{Z}^+ \cup \{\infty\})^{p \times p}$ la matrice de causalité entre les événements, obtenue à partir de S . L'élément a_{ij} correspond à la longueur du plus petit chemin dirigé, non interrompu, entre les événements e_j et e_i (i.e. pas de 0 entre e_j et e_i) présents dans la séquence S . Nous définissons aussi la matrice $B = (b_{ij}) \in \{0, 1\}^{p \times p}$ obtenue à partir de A telle que $b_{ij} = 1$ si $a_{ij} = 1$ et $b_{ij} = 0$ sinon. L'ensemble des causalités directes est complètement représenté par la matrice B . De plus, les relations de causalité directes (matrice B) sont suffisantes pour reconstruire les relations de rang supérieur à 1 (matrice A). Les matrices A et B sont définies de façon unique à partir d'une séquence S donnée.

Dans nos travaux précédent (Ould El Mehdi *et al.*, 2007) nous avons montré que pour toute séquence d'alarmes S munie d'un ordre total, il existe un RdP ordinaire $\langle P, T, W_{PR}, W_{PO} \rangle$ avec $|T| = p$, $|P| = n$ et un marquage initial M_I qui réalise S . Nous avons aussi montré que p et n vérifient la relation (5):

$$p \geq n \geq \text{Max}(a_{ij}) \quad (5)$$

L'algorithme d'identification que nous proposons construit un RdP capable de réaliser la séquence S . Le

principe de la méthode proposée est de faire coïncider la matrice $CD_{TT} = 1$ de ce RdP avec la matrice B construite à partir de S . Le RdP obtenu réalise S mais il est important de noter que ce modèle peut aussi réaliser d'autres séquences. En effet, l'algorithme proposé apprend la structure d'incidence du RdP mais en aucun cas les séquences d'état (qui dépendent de cette structure mais aussi du marquage initial). Cet algorithme identifie la relation d'ordre induite par la séquence d'événements considérée, et est inspiré des méthodes connexionnistes (Ould El Mehdi *et al.*, 2006). Il est basé sur un apprentissage supervisé par rétro-propagation du gradient qui apprend les relations de causalité entre événements. Ce choix est justifié parce que les matrices d'incidence amont et aval peuvent être interprétées comme des matrices de pondérations (binaires) d'un réseau connexionniste et parce que le dimensionnement du RdP en nombre de places peut être interprété comme le dimensionnement de la couche cachée d'un RdN (à l'aide d'une méthode d'élagage). L'algorithme construit (a) la matrice B de propagation directe des événements, (b) le RdP possédant un nombre minimal de places à partir de cette matrice B . Les couches d'entrée et de sortie du réseau de neurones correspondent toutes les deux aux p transitions du modèle et la couche cachée correspond aux n places du modèle (figure 1). Cette méthode apprend l'ensemble des relations de propagation directes entre les événements (i.e. entre les couples entrée/sortie du système).

3.2. Construction dynamique de B

la matrice B est une matrice de dimension $p \times p$. Pour une meilleure compréhension, les p événements sont renommés selon leur ordre d'apparition dans la séquence. La matrice B est initialisée à zéro ($B = (0)$) au premier événement e_1 non nul dans la séquence S . S est parcourue jusqu'au second événement e_2 non nul. Si e_2 suit directement e_1 alors $b_{21} = 1$ sinon B est une matrice nulle de dimension 2. On continue à parcourir la séquence. Dans le cas d'événements déjà présents, les éléments de la matrice B sont réactualisés, et à chaque nouvel événement rencontré, la dimension de B est augmentée d'une ligne et d'une colonne. Par exemple, si $S = [0\ 0\ 1\ 0\ 2\ 3\ 0\ 1\ 2\ 3\ 2\ 3]$, $s(3) = e_1$, et $B = (0)$. La lecture de S se poursuit, $s(5) = e_2$, et $B = B_1$ (car les 2 événements e_1 et e_2 ne se succèdent pas directement) puis $s(6) = e_3$, et $B = B_2$. Enfin, après lecture complète de la séquence on obtient $B = B_3$ (équation (6)).

$$B_1 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, B_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, B_3 = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad (6)$$

3.3. Identification du réseau de Petri

L'algorithme d'apprentissage calcule les matrices d'incidences W_{PR} et W_{PO} . La matrice des poids Q entre les éléments de la couche d'entrée et ceux de la couche cachée correspond aux connexions entre les transitions et les places (i.e. matrice d'incidence arrière W_{PR}) et la

matrice des poids V entre les éléments de la couche cachée et ceux de la couche de sortie correspond aux connexions entre les places et les transitions (i.e. la transposée de la matrice d'incidence avant W_{PO}). On définit la fonction de seuillage $Br(x)$ par $Br(x) = 0$ si $x < 0,5$ et $Br(x) = 1$ sinon. Le rôle de cette fonction est de transformer les matrices de poids Q et V en matrices d'incidence avant et arrière : $W_{PO} = Br(Q)$ et $W_{PR} = (Br(V))^T$. Le réseau est entraîné avec des couples entrée / sortie (x_k, b_k) , $k = 1, \dots, p$, grâce à un apprentissage supervisé. L'ensemble d'apprentissage est issu des relations de propagation de rang 1 de la matrice B obtenue à partir de la séquence d'événements enregistrée. L'ensemble des vecteurs d'apprentissage utilisés pour la couche d'entrée $\{x_k\}$, $k = 1, \dots, p$, contient les p vecteurs de Parikh associés aux événements e_k : $x_k = (0, \dots, 0, 1_k, 0, \dots, 0)^T$. L'ensemble des vecteurs d'apprentissage utilisés pour la couche de sortie $\{b_k\}$ $k = 1, \dots, p$, contient les p colonnes de la matrice B . L'adaptation des matrices Q et V est obtenue en comparant, pour chaque vecteur d'entrée x_k , la sortie du réseau y_k avec b_k et en calculant l'erreur quadratique moyenne ε définie par l'équation (7).

$$\varepsilon = \frac{1}{p^2} \sum_{i=1}^p \sum_{b=1}^p (y_{ib} - b_{ib})^2 \quad (7)$$

L'objectif de la phase d'apprentissage est de diminuer l'erreur ε en modifiant les matrices Q et V . Ces matrices évoluent vers les valeurs exactes des matrices d'incidence. En effet, la répétition du calcul et de la rétro-propagation de l'erreur ε pour tous les éléments de l'ensemble d'apprentissage permet de diminuer progressivement les erreurs présentes dans les relations de causalité de rang 1. La solution est obtenue lorsque l'erreur quadratique est égale à 0. Dans ce cas le minimum de la fonction coût est un minimum global (car $\varepsilon \geq 0$). Les matrices Q et V sont déterminées en calculant le vecteur de sortie $Y=V.Z$ du réseau, avec $Z = Q.X$. L'adaptation de Q et V est obtenue avec les équations (8) dans lesquelles le paramètre η représente le taux d'apprentissage.

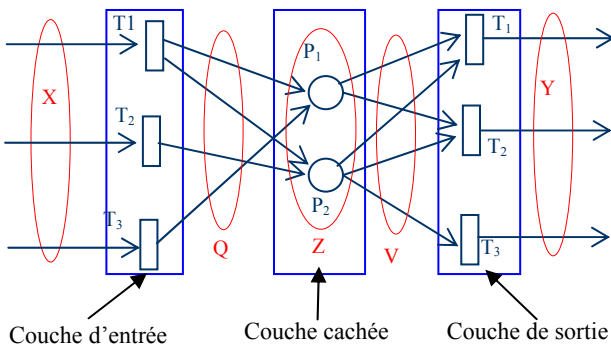


Figure 1 : Structure utilisée pour l'apprentissage

$$\Delta v_{ij} = -\eta \frac{\partial \varepsilon}{\partial v_{ij}} = -\eta \sum_{b=1}^p (y_{ib} - b_{ib}) * z_{bj}$$

$$\Delta q_{lk} = -\eta \frac{\partial \varepsilon}{\partial q_{lk}} = -\eta \sum_{i=1}^p v_{li} * \left(\sum_{b=1}^p (y_{ib} - b_{ib}) * x_{bk} \right) \quad (8)$$

Les éléments des matrices Q et V sont des nombres réels, mais la solution recherchée est, quant à elle, une paire de matrices de nombres booléens dans le cas des RdP ordinaires. Pour surmonter cette difficulté, l'algorithme d'apprentissage comporte deux boucles. La première boucle modifie les valeurs de Q et V selon l'équation (8). Chaque passage dans cette boucle est appelé « itération ». Lorsque l'erreur ε se stabilise (convergence vers un minimum local) ou lorsqu'un nombre maximal d'itérations est atteint, les éléments des matrices Q et V sont tronqués en utilisant la fonction $Br(\cdot)$ de façon à obtenir le RdP ordinaire « le plus proche » (i.e. $W_{PO} = Br(Q)$ et $W_{PR} = (Br(V))^T$) du résultat obtenu par la boucle de rétro-propagation. L'erreur ε est recalculée en utilisant les matrices d'incidence obtenues. Si cette erreur est nulle le processus s'arrête, sinon une nouvelle phase d'adaptation débute en utilisant les valeurs entières précédemment obtenues pour réinitialiser les matrices de poids. La seconde boucle correspond donc aux réinitialisations successives de Q et V , chaque passage dans cette boucle est appelé « époque ».

Dans certain cas, l'algorithme du gradient converge vers un minimum local différent du minimum global, tel que les valeurs obtenues après troncature restent à l'intérieur du puits d'attraction de ce minimum. Dans ce cas, la réinitialisation décrite précédemment ne parvient pas à extraire le modèle du puits d'attraction et l'erreur quadratique après seuillage se stabilise à une valeur non nulle. Une réinitialisation aléatoire partielle de Q et V est alors proposée : pour chaque élément inexact de la matrice CD_{TT} , la colonne correspondante de V et la ligne correspondante de Q sont modifiées aléatoirement.

L'algorithme d'apprentissage est présenté ci-dessous :

Entrées : $p, n, limite_epo, limite_it$
 E, S : ensembles de données d'apprentissage

Sorties : matrices d'incidence : W_{PR}, W_{PO}

Début

Initialisation aléatoire des matrices Q et V
 $err_av \leftarrow 1$

Tant que $limite_epo$ n'est pas éteinte
 et RdP n'est pas trouvé

Tant que ($err > error_seuil$)

et ($limite_it$ n'est pas éteinte)

$Y \leftarrow V * Q * E$

$err \leftarrow sum (Y - S)^2 / p^2$

$\Delta V \leftarrow -\eta * (Y - S) * X^T$

$\Delta Q \leftarrow -\eta * (Y - S) * V * E^T$

Fin Tant que

Troncature des matrices réelles Q et V

$Y \leftarrow V * Q * E$

Si ($Y > 0$) = S
 - RdP trouvé
Si $err_av \leq err$ (augmentation de l'erreur)
 Si $Y(i,j) <> S(i,j)$
 modification aléatoire :
 de la ligne i de V
 et la colonne j de Q
 Fin Si
Fin Si
 $err_av \leftarrow err$
Fin Tant que
Fin

L'algorithme d'apprentissage est illustré à l'aide de la séquence triviale $S = [1\ 2\ 3\ 0\ 1\ 2\ 3\ 2\ 0\ 1\ 3\dots]$. La séquence S permet de construire la matrice B des chemins dirigés de rang 1 entre les événements e_1, e_2 et e_3 :

$$B = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

En exécutant l'algorithme d'apprentissage, à partir de la matrice B , la boucle de rétro-propagation converge, au bout de 742 itérations, vers des matrices Q et V dont la troncature conduit à une structure de RdP dont les chemins dirigés de rang 1 correspondent à $CD_{TT}(RdP_1, [0, 1])$. A la fin de la première époque l'erreur vérifie $\varepsilon = 0.111$. Pour la deuxième époque, la boucle de rétro-propagation converge au bout de 123 itérations vers un RdP dont les chemins dirigés de rang 1 correspondent à $CD_{TT}(RdP_2, [0, 1])$. A la fin de la seconde époque l'erreur est nulle et l'algorithme s'arrête.

$$CD_{TT}(RdP_1, [0,1]) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix},$$

$$CD_{TT}(RdP_2, [0,1]) = B = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

Pour cet exemple, les matrices d'incidence obtenues sont données par :

$$W_{PR} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \quad W_{RO} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Pour réduire la dimension du modèle RdP obtenu, l'algorithme d'apprentissage est combiné avec une méthode d'élagage (Ould El Mehdi *et al.*, 2007) qui supprime progressivement des éléments dans la couche cachée. Le nombre initial d'éléments dans cette couche est choisi égal à p et lorsque l'algorithme converge vers un modèle acceptable, l'apprentissage est réitéré avec un nombre inférieur d'éléments dans la couche cachée. Le RdP de taille minimale est le dernier obtenu pour lequel la convergence est assurée.

4. APPLICATION AU DIAGNOSTIC D'UN SYSTEME DE CUVES

4.1. Présentation du système

Le système considéré est constitué de 2 réservoirs cylindriques R_1 et R_2 , reliés entre eux par 2 conduites C_1 et C_2 (figure 2). La conduite C_1 est située au fond des réservoirs (référence des niveaux d'eau), la conduite C_2 est située à une hauteur de 50 cm. La pompe P_1 est commandée en tout-ou-rien (TOR). La dynamique de la pompe est rapide, le transitoire peut être négligé. On peut donc supposer que le débit de P_1 vérifie $QP_1 = 0$ (pompe arrêtée) ou $QP_1 = D$ (pompe alimentée). La vanne TOR V_1 est également commandée. La pompe P_2 et la vanne V_2 sont considérées comme des entrées de perturbation. Ces entrées peuvent agir en TOR ou en continu (dans ce cas le débit de P_2 est limité à $2D$ et le débit de V_2 est limité à la valeur maximale donnée par la loi de Torricelli). Les équations du système sont données par les équations (9) et (10) :

$$\begin{aligned} S\dot{h}_1 &= QP_1 - QV_1 - QV_3 - QV_4 \\ S\dot{h}_2 &= QP_2 - QV_2 + QV_3 + QV_4 \end{aligned} \quad (9)$$

et :

$$\begin{aligned} QV_1 &= A.(2g.h_1(t))^{1/2}.V_1 \\ QV_2 &= A.(2g.h_2(t))^{1/2}.V_2 \\ QV_3 &= A.(2g./|h_1(t)-h_2(t)|)^{1/2} \\ QV_4 &= A.(2g./|\max(h_1(t),0.5)-\max(h_2(t),0.5)|)^{1/2} \end{aligned} \quad (10)$$

avec $S = 0.0154\ m^2$ la section des deux réservoirs (R_1 et R_2), $D = 10^4\ m^3/s$ le débit de la pompe P_1 , $A = 3.6\ 10^{-5}\ m^2$, la section des 2 conduits, et $g = 9.81\ m/s^2$ la constante de gravité.

Nous considérons les capteurs de niveau m_2, mm_2, M_1 , et MM_1 . Ces capteurs sont considérés comme des détecteurs TOR. Dans un souci de simplicité, et lorsqu'aucune ambiguïté n'est possible, nous noterons de la même manière l'état des capteurs de niveau et la valeur numérique du seuil (par exemple, le capteur $m_2 = 1$ si $h_2(t) \geq m_2$ et $m_2 = 0$ sinon). Enfin nous désignons respectivement par $\uparrow m_2$ et $\downarrow m_2$ les événements qui correspondent aux fronts montants et descendants sur l'état du capteur m_2 . Les mêmes notations sont adoptées pour les 3 autres capteurs de niveau mm_2, M_1 , et MM_1 , pour les 2 actionneurs P_1 et V_1 et pour les 2 entrées de perturbation P_2 et V_2 .

La commande est définie par les instructions suivantes. Si $h_1(t) \geq M_1$ alors arrêter la pompe P_1 . Si $h_1(t) \geq MM_1$ alors arrêter la pompe P_1 et ouvrir la vanne V_1 . Si $h_2(t) < m_2$ alors fermer la vanne V_1 . Si $h_2(t) < mm_2$ alors fermer la vanne V_1 et démarrer la pompe P_1 .

Différents défauts peuvent être considérés, des défauts capteurs : chaque capteur peut rester bloqué sur la position 1 ou 0 quel que soit le niveau d'eau dans les

reliée qu'à une seule place amont et une seule place aval) (Ould El Mehdi *et al.*, 2006). Nous obtenons ce graphe d'état en dupliquant les transitions qui correspondent à des synchronisations, cette méthode permet de conserver la matrice B du réseau (figure 4). Certains événements sont représentés par plus d'une transition selon les sous-séquences réalisées : un même événement est représenté par une certaine transition dans le cas d'un fonctionnement normal et par une autre dans le cas de dysfonctionnement. On peut constater que $CD_{TT}(GE, [0, 1]) = B(S_3)$

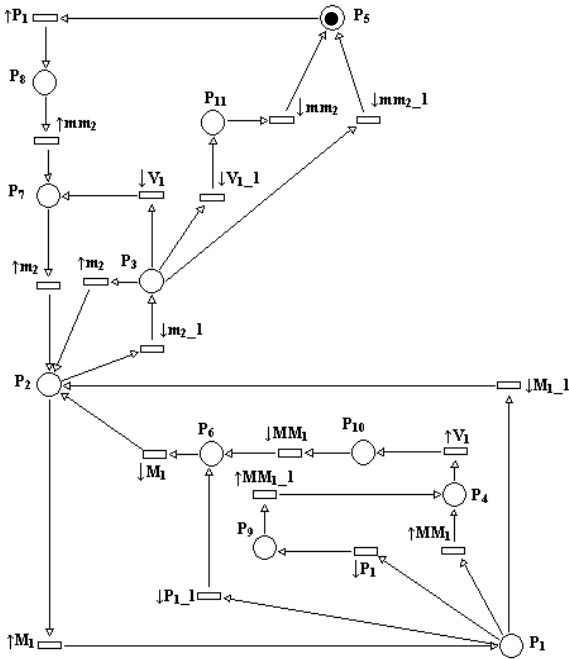


Figure 4 : Modèle de fonctionnement GE avec 11 places et 18 transitions

4.4. Application au diagnostic

Les modèles obtenus sont utiles pour détecter les défauts capteurs et actionneurs. Dans le cas du système à deux cuves, 12 défauts doivent être détectés. A titre d'illustration, le système est simulé avec un défaut sur le capteur M_1 qui reste bloqué sur la position 1 de 20000 UT jusqu'à 50000 UT et un défaut sur la pompe P_1 qui reste en fonctionnement pendant la même période. La séquence S_4 est obtenue en simulant le système sur 500000 UT.

$S_4 = [9\ 8\ 3\ 1\ 10\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 4\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 1\ 2\ 4\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 4\ 3\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 4\ 3\ 4\ 3\ 1\ 2\ 1\ 2\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 4\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 4\ 3\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 4\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 1\ 2\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 4\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 4\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 1\ 2\ 1\ 5\ 11\ 7\ 2\ 4\ 12\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 1\ 5\ 11\ 7\ 2\ 0\ 9\ 8\ 3\ 1\ 10\ 5\ 11\ 7\ 2\ 4\ 12\ 3\ 4\ 6\ 9\ 8\ 3\ 1\ 10\ 2\ 1\ 5\ 11\ 7\ 2\ 1\ 2\ 1\ 5\ 7\ 5\ 7\ 2\ 1\ 2\ 4\ 12\ 3\ 0\ 9\ 8\ 3\ 1\ 10\ 5\ 11\ 7\ 2\ 4$

12 3 4 6 9 8 3 1 10 2 1 2 1 5 11 7 2 4 12 3 1 5 11 7 4 12 6 8 3 4 3 4 3 4 3 4 2 3 4 3].

Le graphe d'état GE ne permet pas de réaliser la séquence S_4 (les causalités entre e_7 et e_4 , entre e_6 et e_8 , entre e_4 et e_2 et entre e_2 et e_3 , ne sont pas prises en compte par GE). Le défaut peut donc être détecté dès l'apparition de la sous-séquence [...7 4 ...].

Cependant il convient de noter que le retard à la détection peut être important (par exemple [...7 4...]) est observé à l'instant 35 395 UT alors que le défaut apparaît à l'instant 20 000 UT !). Ce retard n'est pas surprenant car seules les conséquences du défaut sont observées. La prise en compte des aspects temporels sera davantage étudiée dans nos travaux futurs. Il faut aussi noter que le modèle GE peut réaliser des sous séquences non incluses dans S_3 qui ne correspondent à aucune séquence « réelle ».

La séquence S_4 peut aussi être utilisée pour construire un modèle de dysfonctionnement qui permet de compléter le diagnostic en discernant les défauts. Le modèle de dysfonctionnement $GE_{dysfonct.}$ déduit de S_4 est présenté sur la figure 5. Le modèle de dysfonctionnement complet (incluant les 12 défauts considérés) contient 12 places et 47 transitions (non présenté pour des raisons de lisibilité).

5. CONCLUSION

Dans cet article, un algorithme d'identification a été proposé pour construire des modèles RdP permettant de reproduire une séquence d'événements donnée. La méthode permet d'adapter le modèle en ligne dans le cas où les événements sont enregistrés au fil du temps sans connaissance a priori sur le système étudié et sous l'hypothèse d'une relation d'ordre totale entre les événements. Elle présente, de ce point de vue, un intérêt certain car elle permet de construire en temps réel un modèle qui évolue en fonction des informations fournies par l'instrumentation. Dans le cas des systèmes soumis à des défauts multiples et fréquents, elle permet de construire systématiquement un modèle de dysfonctionnement du système mais ce modèle dépend fortement de la taille et de la qualité de la séquence d'événements enregistrée.

Nos travaux ont été guidés essentiellement par le souci d'obtenir des modèles de faible dimension. D'autres contraintes induites par les nécessités de l'observation et par les aspects temporels pourront être rajoutées de manière à construire par apprentissage des modèles plus adaptés au diagnostic précoce.

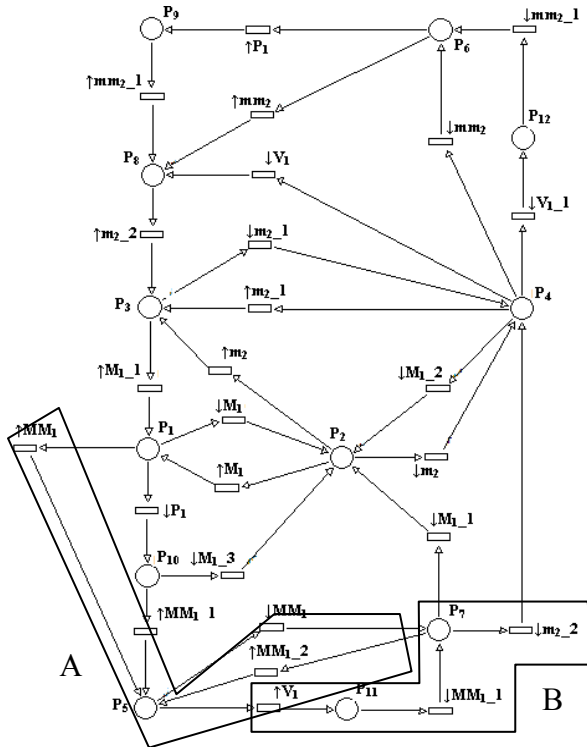


Figure 5 : Modèle de dysfonctionnement $GE_{dysfonct.}$ avec 12 places et 26 transitions incluant deux défauts

REFERENCES

Alcaraz-Mejia M., Lopez-Mellado E., Ramirez-Trevino A., Rivera-Rangel I., Petri net based fault diagnosis of DES. *Proc. IEEE-SMC03*, pp. 4730-4735, Washington, USA, 2003.

Bouché P., Une approche stochastique de modélisation des séquences d'événements discrets pour le diagnostic des systèmes dynamiques, Thèse de doctorat de l'Université d'Aix – Marseille III, 2005.

Bourdeaud'huy T., Yim P., Synthèse de réseaux de Petri à partir d'exigences, *Actes de la 5me conf. Modélisation et Simulation*, pages 413-420, Nantes, France, september 2004.

Bulitko, V. et D. C. Wilkins. Machine Learning for Time Interval Petri Nets. *In Proceedings of the 18th Australian Joint Conference on Artificial Intelligence, Lecture Notes in Artificial Intelligence*, Sydney, Australia, pp. 959-965. Springer-Verlag, 2005.

David R., Alla H., *Petri nets and grafcet – tools for modelling discrete events systems*, Prentice Hall. London, 1992.

Dousson C, Thang Vu Duong: Discovering Chronicles with Numerical Time Constraints from Alarm Logs for Monitoring Dynamic Systems. *IJCAI 1999*: 620-626

Lefebvre D. Delherm C., Diagnosis with causality relationships and directed paths in PN models. *IFAC World Congress 05*, Prague, Czech Republic, 2005.

Lefebvre D., Delherm C., Fault detection and isolation of discrete event systems with Petri net models, *IEEE – TASE*, Vol. 4, nu. 1, pp. 114 – 118, January 2007.

Lefebvre D. , Immediate diagnosis of faulty behaviours with Petri net models, *IFAC World Congress 05*, Prague, Czech Republic, 2005.

Mannila H., Toivonen H., Inkeri Verkamo A., Discovery of frequent episodes in event sequences, *Data mining and knowledge discovery*, vol. 1, pp. 259-289, 1997.

Meda-Campana M.E., Lopez-Mellado E. Incremental synthesis of Petri net models for identification of discrete event systems. *In Proc. 41th IEEE-CDC*, pp. 805-810, Las Vegas, Nevada, 2002.

Meda-Campana M.E., Lopez-Mellado E., Required event sequences for identification of discrete event systems. *In Proc. 42th IEEE-CDC*, pp. 3778-3783, Maui, Hawaii, 2003.

Ould El Mehdi S., Leclercq E., Lefebvre D., Synthèse de modèles par réseaux de Petri pour le diagnostic des systèmes à événements discrets, *Congrès Lambda Mu 15*, Lille, 10-12 octobre, 2006.

Ould El Mehdi S., Leclercq E., Lefebvre D., Apprentissage hors ligne et en ligne de modèles par réseaux de Petri pour le diagnostic des SED, soumis à *QUALITA 2007*, session invitée « Diagnostic des systèmes complexes », Tanger, Maroc, 2007.

Sampath M., Sengupta R., Lafortune S., Sinnamohideen K., Teneketzi D., Diagnosibility of discrete event systems. *IEEE-TAC*, vol. 40, no.9, pp. 1555- 1575, 1995.

Ushio T., Onishi I., Okuda K., Fault detection based on Petri net models with faulty behaviours. *IEEE – SMC*, pp 113-118, 1998.