

# Diagnostic des systèmes à événements discrets avec forçage : approche multi-modèles

Asmae El Ghadouali  
Laboratoire LIST, FSTT,  
Université Abdelmalek Essaâdi,  
Email: elghadouali.asmae-etu@uae.ac.ma

Oulaid Kamach  
Laboratoire LTI, ENSAT,  
Université Abdelmalek Essaâdi,  
Email: okamach@ensat.ac.ma

Benaïssa Amami  
Laboratoire LIST, FSTT,  
Université Abdelmalek Essaâdi,  
Email: b\_benaïssa@hotmail.com

**Résumé**—Le travail présenté dans cet article porte sur le diagnostic des systèmes à événements discrets (SED) en tenant compte des différents modes de fonctionnement possibles. L'approche adoptée est une approche multi-modèles où chaque modèle décrit un système dans un mode de fonctionnement donné où plusieurs défaillances peuvent survenir. Nous considérons deux types de modes de fonctionnement : un mode nominal et un mode dégradé. L'activation et la désactivation d'un mode sont tributaires de l'occurrence d'un événement de commutation. Cet événement peut être un événement de panne (incontrôlable) ou un événement de réparation (contrôlable et observable). Nous supposons que les événements de défaillance (pannes des ressources engagées) sont des événements inobservables. Pour assurer la commutation entre les différents modes de fonctionnement en présence de ces événements inobservables, nous proposons une nouvelle architecture de diagnostic en se basant sur le principe de coordination, de forçage des événements contrôlables et de comparaison. Nous élaborons un algorithme pour déterminer d'une part le mode actif dans lequel le système se trouve et d'autre part l'occurrence et le type des défaillances.

Les mots clés : Systèmes à événements discrets, Automate, Gestion des modes de fonctionnement, Changement de modes, multi-modèles, Diagnostic.

## I. INTRODUCTION

Vu que les systèmes à événements discrets peuvent générer des séquences indésirables, plusieurs auteurs se sont penchés sur le problème de diagnostic. Il existe un grand nombre de méthodes de diagnostic [1]–[5]. Elles se basent sur un modèle du comportement normal et/ou défaillant du système et se distinguent selon différents critères : la dynamique du procédé à diagnostiquer (discret, continu ou hybride), l'implémentation du diagnostic en ligne et/ou hors ligne, la complexité du procédé, la nature de l'information disponible (qualitative et/ou quantitative), sa profondeur (analytique ou heuristique) et sa distribution (centralisée ou décentralisée). Leur principe général repose sur une comparaison entre les données observées au cours du fonctionnement du système et les connaissances sur son comportement nominal et ses comportements de défaillance. Cependant, rares sont ceux qui ont pris en considération la gestion des modes de fonctionnement

[6] dans un environnement incertain. Nos travaux apportent les premières démarches formelles introduisant le concept de gestion des modes pour répondre au problème de diagnostic des SED.

Un tel système représente différents modes de fonctionnement selon l'engagement et la configuration de ses ressources. Un mode de fonctionnement correspond à des besoins et donc des commandes différentes, et à des ressources différentes. On parle de modèle à typologie variable ou de systèmes multi-modèles. Des travaux de [7] ont été les premiers à proposer pour intégrer à l'étude du mode de fonctionnement nominal la prise en compte des comportements admissibles du système après une défaillance critique. Ces travaux ne se limitent qu'à deux configurations (nominale et dégradée) et à l'occurrence d'une seule défaillance à la fois. L'extension supprimant cette limitation a été proposée dans les travaux de [8], [9], où plusieurs défaillances peuvent survenir. Pour assurer la commutation entre les différents modes de fonctionnements, une approche algébrique a été proposée dans les travaux de [10], [11].

Dans l'approche multi-modèles, le système commute d'un mode nominal à un mode dégradé à l'occurrence d'une défaillance. L'événement de réparation conduit au retour vers le mode nominal. Dans notre démarche, nous supposons que les événements de défaillance sont des événements inobservables. Pour identifier et localiser un événement inobservable, il existe plusieurs structures de prise de décision dépendant de la distribution de l'information disponible sur le procédé : centralisée [1], décentralisée [3], [4] ou distribuée [5]. L'approche adoptée est de structure multi-modèles [8]. Pour cela, nous proposons une nouvelle architecture de prise de décision en tenant compte des différents modes de fonctionnement possibles d'un système physique. Cette architecture se base sur trois concepts : le coordinateur, le superviseur et le comparateur. Pour faciliter l'application de notre démarche de diagnostic, nous élaborons un seul algorithme qui doit avoir une vue globale du système à diagnostiquer. Notre algorithme détermine d'une part le mode actif et d'autre part le diagnostic et le type de la défaillance. Notre objectif est d'assurer la commutation entre les différents modes de fonctionnement en présence des événements de défaillance.

Dans cet article, nous étudions, d'abord, la décision locale prise par un diagnostiqueur défini pour chaque mode de



Dans notre démarche, nous considérons deux types de modes : un mode nominal dans lequel le système remplit parfaitement sa tâche et un mode dégradé dans lequel le système poursuit sa tâche en dépit d'une défaillance. En effet, chaque système n'admet qu'un seul mode de bon fonctionnement (mode nominal); par contre, il peut avoir plusieurs modes dégradés. Dans le cas où il y a plusieurs défaillances à diagnostiquer dans la séquence d'activation  $S_a$ , nous utilisons la thermologie  $F_i$  pour indiquer la présence d'une défaillance  $f_i$ . Dans ce cas, le diagnostiqueur prend une décision de plusieurs fautes détectées dépendant du mode dégradé.

Tout d'abord, nous notons l'ensemble des traces générées dans le modèle  $G_{M_i}$  ayant comme origine l'état de départ  $q_{M_i, départ}$  et comme dernier événement de commutation  $\sigma_{M_i, M_j}$ , par l'expression  $L(G_{M_i}, \sigma_{M_i, M_j})$ , formellement :

$$L(G_{M_i}, \sigma_{M_i, M_j}) = \{S \in L(G_{M_i}) \mid \delta_{M_i, et}(q_{M_i, 0}, S\sigma_{M_i, M_j})!\}$$

**Définition 2.** Le diagnostiqueur actif  $Diag_{M_i}$  du mode  $M_i$  prend une décision si l'événement courant de la séquence d'activation appartient à l'alphabet du mode  $M_i$ . Cette décision peut être égale à: "NF" (pas de faute), " $F_i$ " (faute) et "ND" (pas de décision), telle que :

$$Diag_{M_i} : P_{M_i}(\Sigma_{global}^*) \rightarrow \{NF, F_i, ND\}$$

$$Diag_{M_i}(\sigma) =$$

$$\left\{ \begin{array}{l} NF \quad \text{si } P_{M_i}(\sigma) \neq \varepsilon \text{ et } i = 1 \text{ et } \sigma \in \Sigma_{M_i} \setminus \bigcup_{l \neq i} \Sigma_{M_l} \\ \quad \text{si } P_{M_i}(\sigma) \neq \varepsilon \text{ et } i = 1 \text{ et } \exists j \neq i \\ \quad \text{tel que } \sigma \in \Sigma_{M_i}^{\leftarrow} \cap \Sigma_{M_j}^{\rightarrow} \\ \quad \text{si } P_{M_i}(\sigma) \neq \varepsilon \text{ et } i = 1 \text{ et } \exists j \neq i \\ \quad \text{tel que } \sigma \in \Sigma_{M_i} \cap \Sigma_{M_j} \text{ et } s\sigma \notin L(G_{M_i}, \sigma_{M_i, M_j}) \\ F_i \quad \text{si } P_{M_i}(\sigma) \neq \varepsilon \text{ et } i \neq 1 \text{ et } \sigma \in \Sigma_{M_i} \setminus \bigcup_{l \neq i} \Sigma_{M_l} \\ \quad \text{si } P_{M_i}(\sigma) \neq \varepsilon \text{ et } i \neq 1 \text{ et } \exists j \neq i \\ \quad \text{tel que } \sigma \in \Sigma_{M_i}^{\leftarrow} \cap \Sigma_{M_j}^{\rightarrow} \\ \quad \text{si } P_{M_i}(\sigma) \neq \varepsilon \text{ et } i \neq 1 \text{ et } \exists j \neq i \\ \quad \text{tel que } \sigma \in \Sigma_{M_i} \cap \Sigma_{M_j} \text{ et } s\sigma \notin L(G_{M_i}, \sigma_{M_i, M_j}) \\ ND \quad \text{sinon} \end{array} \right. \quad (2)$$

Avec,  $\Sigma_{M_i}^{\leftarrow}$  l'ensemble des événements de commutation activant le mode  $M_i$ ,

$\Sigma_{M_j}^{\rightarrow}$  l'ensemble des événements de commutation désactivant le mode  $M_j$ ,

$\Sigma_{M_i} \setminus \Sigma_{M_j}$  est l'ensemble des éléments de  $\Sigma_{M_i}$  qui n'appartiennent pas à  $\Sigma_{M_j}$

et  $s$  est une séquence d'événements qui précède l'événement  $\sigma$  dans la séquence d'activation  $S_a$ , où  $s \in L_{S_a}$ .

D'après cette définition, le diagnostiqueur  $Diag_{M_i}$  du mode traité  $M_i$  peut prendre une décision locale (pas de faute NF, faute F) lorsque :

- l'événement généré appartient seulement à l'alphabet du mode  $M_i$ .
- l'événement généré est un événement de commutation activant le mode  $M_j$ .
- l'événement généré appartient à l'alphabet du mode  $M_i$  et l'alphabet d'un autre mode  $M_j$ . Mais, le suivi des

événements générés ne conduit pas à l'occurrence d'un événement de commutation activant le mode  $M_j$ .

Pour faciliter l'application de cette définition, nous proposons l'algorithme 1. Il reçoit en entrée l'ensemble des modes de fonctionnement  $M = \{M_1, M_2, \dots, M_n\}$  et l'événement courant  $\sigma$  généré par le procédé. Notre algorithme donne comme résultat final, la décision de chaque diagnostiqueur  $Diag_{M_i}(\sigma)$  du mode  $M_i$ .

---

**Algorithme 1** Algorithme de la prise de décision locale de diagnostic

---

**ENTRÉES:**  $\sigma$  {/\* L'événement généré par le procédé \*/}  
 $M$  {/\* Ensemble des modes \*/}  
 $M_i \in M$  {/\* Le mode de fonctionnement traité \*/}

**SORTIES:**  $Diag_{M_i}()$  {/\* Décision locale de diagnostiqueur  $Diag_{M_i}$  du mode  $M_i$  \*/}  
 $M_j \in M$  {/\* Le mode qui peut être activé après le mode  $M_i$  \*/}

**pour tout**  $\sigma$  généré **faire**  
  **si**  $P_{M_i}(\sigma) = \varepsilon$  **alors**  
     $Diag_{M_i}(\sigma) = ND$   
  **sinon si**  $\sigma \in \Sigma_{M_i} \setminus \bigcup_{l \neq i} \Sigma_{M_l}$  **alors**  
    **si**  $i = 1$  **alors**  
       $Diag_{M_i}(\sigma) = NF$   
    **sinon**  
       $Diag_{M_i}(\sigma) = F_i$   
    **fin si**  
  **sinon si**  $\exists j \neq i$  tel que  $\sigma \in \Sigma_{M_i} \cap \Sigma_{M_j}$  et  $s\sigma \notin L(G_{M_i}, \sigma_{M_i, M_j})$  **alors**  
    **si**  $i = 1$  **alors**  
       $Diag_{M_i}(\sigma) = NF$   
    **sinon**  
       $Diag_{M_i}(\sigma) = F_i$   
    **fin si**  
  **sinon si**  $\exists j \neq i$  tel que  $\sigma \in \Sigma_{M_i}^{\leftarrow} \cap \Sigma_{M_j}^{\rightarrow}$  **alors**  
    **si**  $i = 1$  **alors**  
       $Diag_{M_i}(\sigma) = NF$   
    **sinon**  
       $Diag_{M_i}(\sigma) = F_i$   
    **sinon**  
       $Diag_{M_i}(\sigma) = ND$   
    **fin si**  
  **fin pour**  
 Retourner  $Diag_{M_i}(\sigma)$

---

## B. Coordination

Dans notre architecture de diagnostic multi-modèles, nous utilisons un coordinateur pour fusionner les différentes décisions transmises par les diagnostiqueurs et prendre la décision finale sur l'état du système. Il reçoit en entrée les décisions des diagnostiqueurs  $Diag_{M_i}(\sigma)$ , avec  $i \in \{1, 2, \dots, n\}$ , avec  $\sigma$  l'événement courant généré par le procédé. Nous définissons un coordinateur comme suit :

**Définition 3.** Soit  $Coord = (M_i, Diag_{M_i}(\sigma))$  un coordinateur déterminant le mode courant  $M_i$  avec sa décision finale  $Diag_{M_i}(\sigma)$  lorsqu'un événement  $\sigma$  est généré.

Formellement :

$$Coord : \{NF, F, ND\}^{card(M)} \rightarrow \{(M_1, NF), (M_k, F_k), ND\}$$

$$Coord(Diag_{M_1}(\sigma), \dots, Diag_{M_k}(\sigma), \dots, Diag_{M_n}(\sigma)) =$$

$$\begin{cases} (M_1, NF) & \text{si aucun } Diag_{M_i}(\sigma) \text{ ne dit } F \\ & \text{et } Diag_{M_1}(\sigma) = NF \\ (M_k, F_k) & \text{si aucun } Diag_{M_i}(\sigma) \text{ ne dit } NF \\ & \text{et } Diag_{M_k}(\sigma) = F_k \\ ND & \text{si tous les } Diag_{M_i}(\sigma) \text{ disent } ND \end{cases} \quad (3)$$

### C. Forçage & Comparaison

Dans le cas d'une ambiguïté<sup>1</sup> de la prise de décision, nous proposons d'ajouter un superviseur. Ce dernier se base sur le concept d'événement forcé [12]. Ce superviseur ne conserve plus sa fonction originelle d'autorisation/interdiction d'événements générés par le procédé, il force certains événements à se produire dans le procédé. Le superviseur est alors considéré comme ayant la maîtrise des événements forcés qu'il génère (les événements contrôlables) alors qu'il n'a pas celle des événements générés par le processus (les événements incontrôlables). Dans notre approche de diagnostic multi-modèles, le superviseur a l'habilité de forcer les événements contrôlables de commutation (les événements de réparation) à l'entrée du procédé.

L'alphabet  $\Sigma_{global}^{\leftrightarrow}$  des événements de commutation devra donc être partitionné de la manière suivante:  $\Sigma_{global}^{\leftrightarrow} = \{\Sigma_{global}^{\leftrightarrow}\}^c \cup \{\Sigma_{global}^{\leftrightarrow}\}^{uc}$ , où  $\{\Sigma_{global}^{\leftrightarrow}\}^c$  est l'ensemble des événements de commutation contrôlables et  $\{\Sigma_{global}^{\leftrightarrow}\}^{uc}$  l'ensemble des événements de commutation incontrôlables.

Le but de superviseur est d'assurer le suivi de diagnostic après une ambiguïté de la prise de décision. Le superviseur est actif si le coordinateur est inapte à prendre une décision sûre lors de l'occurrence d'un événement commun entre deux modes de fonctionnement. Il force un événement contrôlable de commutation à l'entrée du procédé pour rester au mode précédemment actif. Le superviseur est accompagné par un comparateur. Ce dernier compare l'événement généré, par le procédé après le forçage, avec l'événement forcé. L'égalité de ces deux événements confirme la fonction du superviseur (le forçage). L'objectif de cette comparaison est de déterminer le mode actif et d'avoir une décision sûre de diagnostic.

L'activation d'un superviseur (resp. comparateur) est présentée par une fonction d'activation  $F_{Sup}$  (resp.  $F_{Comp}$ ). L'ajout du superviseur et du comparateur est illustré dans la nouvelle architecture présentée dans la figure 3.

**Définition 4.** Soit  $F_{Sup}$  la fonction d'activation du superviseur. Cette fonction est définie comme suit :

$$F_{Sup} = \begin{cases} 1 & \text{si } Coord = ND \\ 0 & \text{sinon} \end{cases} \quad (4)$$

<sup>1</sup>le cas où le coordinateur ne prend pas une décision sûre

**Définition 5.** La fonction d'activation du comparateur  $F_{Comp}$  est définie comme suit :

$$F_{Comp} = \begin{cases} 1 & \text{si le superviseur est actif, } F_{Sup} = 1 \\ 0 & \text{sinon} \end{cases} \quad (5)$$

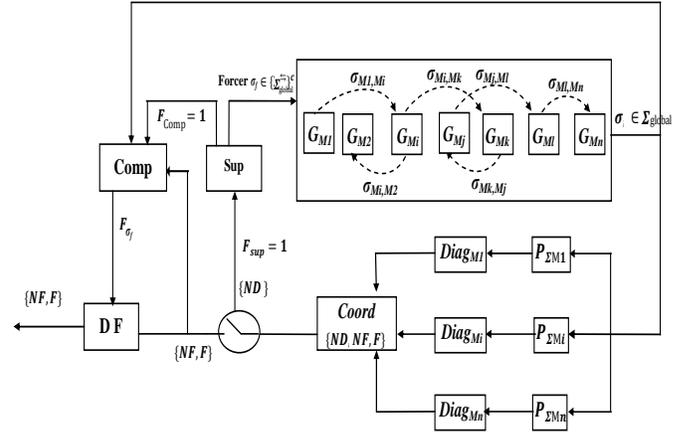


Fig. 3. Architecture proposée pour diagnostiquer un système multi-modèles à l'aide d'un coordinateur, d'un superviseur et d'un comparateur.

### D. Décision finale de diagnostic

D'après la nouvelle architecture illustrée dans la figure 3, la décision finale de diagnostic prend la valeur de coordinateur si sa décision est sûre. Sinon, un superviseur intervient par le forçage d'un événement contrôlable de commutation lors de l'occurrence d'un événement commun entre deux modes de fonctionnement. Par conséquent, le comparateur est actif afin de confirmer ou d'annuler le forçage proposé. Si l'événement généré est égal à l'événement forcé alors la décision finale est celle d'événement forcé. Sinon, le forçage est annulé car l'événement forcé n'existe pas parmi les événements possibles générés après l'événement commun. Dans ce cas, le système est toujours dans le mode précédemment actif et la décision finale sera celle du diagnostiqueur du mode actif. Si l'événement généré est encore un événement commun entre le mode forcé et un autre mode, le coordinateur ne prend pas une décision sûre. Alors, le procédé reste toujours dans le mode forcé jusqu'à l'occurrence d'un événement qui n'appartient pas à l'alphabet du mode forcé. Donc, il faut toujours attendre la réponse du procédé pour confirmer le mode actif et la décision finale de diagnostic.

**Définition 6.** Soit  $F_{\sigma_f}$  la fonction qui confirme le forçage d'événement  $\sigma_f \in \{\Sigma_{global}^{\leftrightarrow}\}^c$ . Sa valeur dépend du résultat de la comparaison entre l'événement généré  $\sigma$  et l'événement forcé  $\sigma_f$ .

$$F_{\sigma_f} = \begin{cases} 1 & \text{si } F_{Comp} = 1 \text{ et } \sigma = \sigma_f \\ 0 & \text{sinon} \end{cases} \quad (6)$$

Pour faciliter l'utilisation de notre approche, nous proposons l'algorithme 2. Il reçoit en entrée l'ensemble des modes

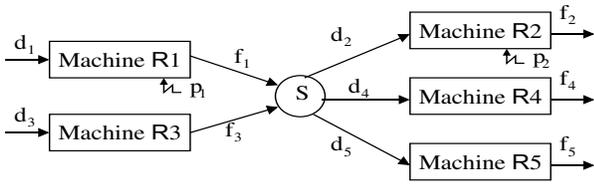


Fig. 4. Système manufacturier à cinq machines et un stock intermédiaire

de fonctionnement  $M = \{M_1, M_2, \dots, M_n\}$  et l'événement courant  $\sigma$  généré par le procédé. Notre algorithme donne comme résultat la décision finale de diagnostic, notée  $DF(\sigma)$ , à l'occurrence d'un événement  $\sigma$  généré par le procédé.

**Algorithme 2** Algorithme permettant la prise de décision finale de diagnostic

**ENTRÉES:**  $\sigma$  *{/\* l'événement courant généré par le procédé \*/}*

$M$  *{/\* Ensemble des modes \*/}*

**SORTIES:**  $DF(\sigma)$  *{/\* Décision finale de diagnostic \*/}*

$F_{Sup} = 0, F_{Comp} = 0$  *{/\* Initialement \*/}*

$M_i, M_j \in M$  *{/\* Deux modes de fonctionnement \*/}*

**pour tout**  $\sigma$  généré **faire**

**si**  $Coord(Diag_1(\sigma), \dots, Diag_n(\sigma)) \in \{NF, F\}$  **alors**

$DF(\sigma) = Coord(Diag_1(\sigma), \dots, Diag_n(\sigma));$

**sinon**

$F_{Sup} = 1;$

Forcer  $\sigma_f \in (\Sigma_{M_i}^{\leftarrow} \cap \Sigma_{M_j}^{\rightarrow})^c$  tel que  $P_{M_i}(\sigma) \neq \varepsilon$  et  $P_{M_j}(\sigma) \neq \varepsilon;$

$F_{Comp} = 1;$

Attendre un événement généré  $\sigma'$  par le procédé;

**si**  $\sigma' = \sigma_f$  **alors**

$F_{\sigma_f} = 1;$

$DF(\sigma) = Coord(Diag_1(\sigma_f), \dots, Diag_n(\sigma_f));$

$F_{Sup} = 0;$

$F_{Comp} = 0;$

**sinon**

$F_{\sigma_f} = 0;$

$DF(\sigma) = (M_i, Diag_{M_i}(\sigma));$

$F_{Sup} = 0;$

$F_{Comp} = 0;$

**fin si**

**fin si**

**fin pour**

Retourner  $DF(\sigma)$

#### IV. EXEMPLE D'APPLICATION

Dans cette section, nous traitons à titre d'exemple un système manufacturier (voir la figure 4), composé de cinq machines et stock intermédiaire : trois machines identiques  $R_1, R_2, R_4$ , deux machines redondantes  $R_3$  et  $R_5$ , et d'un stock intermédiaire  $S$ . Le système doit opérer même en cas de panne. C'est la raison de la présence de la machine  $R_3$  qui remplace la machine  $R_1$  en cas de défaillance.

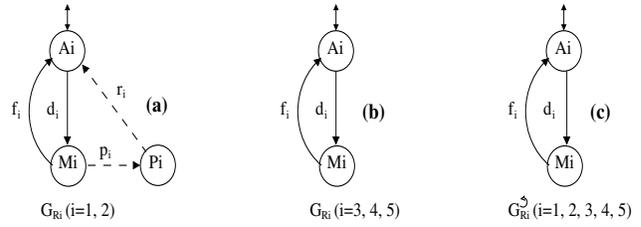


Fig. 5. (a) Modèle automate pour les deux machines  $R_1$  et  $R_2$ , (b) Modèle automate pour les machines  $R_3, R_4$  et  $R_5$

Dans ce système, les machines fonctionnent de façon indépendante. Chaque machine  $R_i$  prend des pièces brutes en amont (modélisé par l'événement  $d_i$ ) et dépose des pièces usinées en aval (symbolisé par l'occurrence de l'événement  $f_i$ ). Une machine  $R_i$  en état arrêt est labélisée par  $Ai$  et en état marche par  $Mi$ . Lorsque la machine  $R_1$  (resp.  $R_2$ ) est en marche, l'occurrence de l'événement  $p_1$  (resp.  $p_2$ ) conduit la machine dans un état de panne  $P1$  (resp.  $P2$ ). Dans cet exemple, on ne considéra pas la panne des machines  $R_3, R_4$  et  $R_5$ . La machine  $R_3$  (resp.  $R_5$ ) remplace la machine  $R_1$  (resp.  $R_2$ ) en cas de défaillance. Le modèle automate est donné conformément à la figure 5(a) pour la machine  $R_1$  et  $R_2$ , et conformément à la figure 5(b) pour les machines  $R_3, R_4$  et  $R_5$ . Les flèches en pointillés représentent les événements de commutation.

Le système peut fonctionner dans quatre modes  $M = \{N, D1, D2, D3\}$ . Dans le mode nominal  $N$  se sont les machines  $R_1, R_2$  et  $R_4$  qui produisent. Lorsque la machine  $R_1$  est en marche, l'occurrence de l'événement  $p_1$  conduit la machine dans un état de panne. Alors, une commutation du système du mode nominal  $N$  vers un mode dégradé  $D1$  a eu lieu. L'occurrence de l'événement de réparation  $r_1$  implique le retour du système dans le mode nominal  $N$ . Dans le premier mode dégradé  $D1$  correspond à l'utilisation de la machine  $R_3$  à la place de la machine  $R_1$  qui est en panne, alors que la machine  $R_2$  est en bon état. Dans le deuxième mode dégradé  $D2$ , c'est l'inverse : la machine  $R_1$  est en bon état alors que la machine  $R_2$  en panne est remplacée par la machine  $R_5$ . Cependant, la machine  $R_5$  est plus rapide que la machine  $R_2$ , il n'est donc plus nécessaire d'utiliser deux machines  $R_5$  et  $R_4$  en même temps pour consommer les pièces du stock  $S$ . De ce fait, seule la machine  $R_5$  reste active pour produire. La machine  $R_4$  est donc stoppée. Enfin, dans le troisième mode dégradé  $D3$  les deux machines  $R_1$  et  $R_2$  en panne sont remplacées par les machines  $R_3$  et  $R_5$ . Les différentes commutations possibles entre ces modes de fonctionnement sont illustrées dans la figure 6.

Le modèle automate du mode nominal  $G_N$  et des modes dégradés  $G_{D1}, G_{D2}$  et  $G_{D3}$  du procédé, représentés dans la figure 7, sont obtenus par le produit synchrone des automates de fonctionnement nominal  $G_{R_i^\circ}$  des ressources engagées dans le mode étudié. L'automate  $G_{R_i^\circ}$  de la figure 5(c) est construit par la suppression des événements de commutation  $\{p_1, r_1, p_2, r_2\}$  depuis l'automate  $G_{R_i}$ .

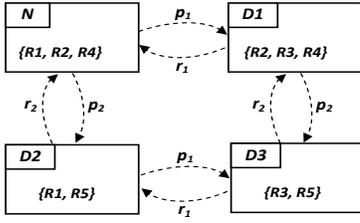


Fig. 6. Les commutations possibles entre les quatre modes d'un système manufacturier à cinq machines.

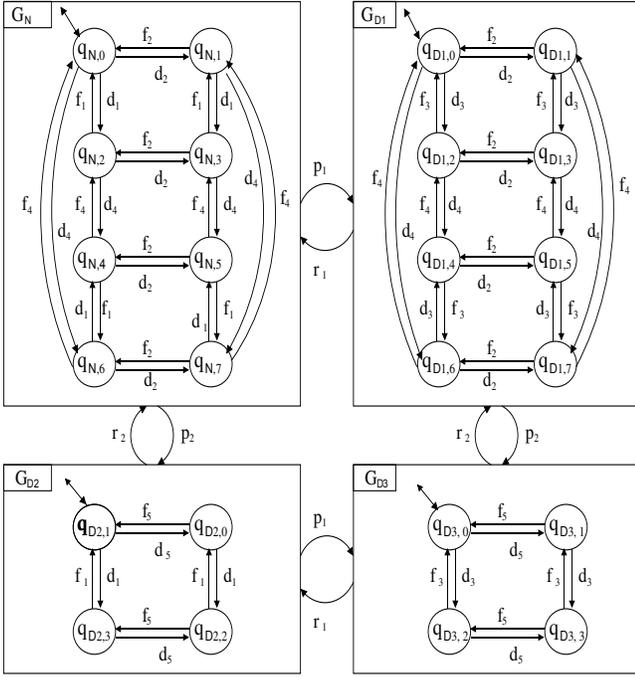


Fig. 7. Le modèle automate du mode nominal  $G_N$  et des modes dégradés  $G_{D1}$ ,  $G_{D2}$  et  $G_{D3}$  du procédé

L'alphabet global  $\Sigma_{global}$  est scindé en cinq alphabets suivants :  $\Sigma_N = \{d_1, f_1, d_2, f_2, d_4, f_4\}$  l'alphabet du mode nominal,  $\Sigma_{D1} = \{d_2, f_2, d_3, f_3, d_4, f_4\}$  l'alphabet du mode dégradé  $D1$ ,  $\Sigma_{D2} = \{d_1, f_1, d_5, f_5\}$  l'alphabet du mode dégradé  $D2$ ,  $\Sigma_{D3} = \{d_3, f_3, d_5, f_5\}$  l'alphabet du mode dégradé  $D3$  et  $\Sigma_{global} = \{p_1, r_1, p_2, r_2\}$  l'alphabet des événements de commutation.

L'alphabet  $\Sigma_{global}$  est partitionné de la manière suivante :  $\Sigma_{global} = \{\Sigma_{global}^c\} \cup \{\Sigma_{global}^{uc}\}$ , où  $\{\Sigma_{global}^c\} = \{r_1, r_2\}$  est l'ensemble des événements de commutation contrôlables (les événements de réparation) et  $\{\Sigma_{global}^{uc}\} = \{p_1, p_2\}$  l'ensemble des événements de commutation incontrôlables (les événements de panne).

Pour chaque mode  $M_i \in \{N, D1, D2, D3\}$ , nous utilisons une fonction de projection  $P_{M_i}$  (1) afin de sélectionner tous les événements observables par ce mode. Nous appliquons l'algorithme 1 pour déterminer la décision de chaque diagnostiqueur  $Diag_{M_i}$  à l'occurrence d'un événement  $\sigma$  généré par le procédé. Le coordinateur fusionne les différentes décisions

transmises par les diagnostiqueurs et prend la décision finale sur l'état du système. Si le coordinateur ne peut pas prendre une décision sûre (différente de  $NF$  et de  $F_i$ ), un superviseur intervient par le forçage d'un événement contrôlable de commutation de l'ensemble  $\{\Sigma_{global}^c\}^c = \{r_1, r_2\}$ . Dans ce cas, le comparateur est actif ( $F_{Comp} = 1$ ) afin de confirmer ou d'annuler le forçage.

En supposant que le système se trouve initialement dans le mode nominal, nous considérons l'exemple de la séquence d'activation  $d_1 d_2$  générée par le procédé. Tout d'abord, nous appliquons la fonction de projection de chaque mode à l'occurrence de l'événement  $d_1$ . En effet,  $d_1$  est un événement commun entre les deux modes  $N$  et  $D2$  (car  $P_N(d_1) \neq \varepsilon$  et  $P_{D2}(d_1) \neq \varepsilon$ ). Mais, le suivi des événements générés depuis le mode initial, l'événement  $d_1$ , ne conduit pas à l'occurrence d'un événement de commutation (panne  $p_2$ ) activant le mode  $D2$ . Nous appliquons l'algorithme 1 pour déterminer la décision de chaque diagnostiqueur. Le coordinateur prend une décision sûre  $NF$  et le mode actif est le mode Nominal. À l'occurrence de l'événement  $d_2$ , le coordinateur ne peut pas prendre une décision sûre de diagnostic car  $d_2$  est un événement commun entre les deux modes de fonctionnement  $N$  et  $D1$ . Dans ce cas, le superviseur force l'événement contrôlable de commutation  $\sigma_{D1,N} = r_1$  (car  $P_N(d_2) \neq \varepsilon$  et  $P_{D1}(d_2) \neq \varepsilon$ ). Or, le procédé génère un événement  $d_4$  qui est différent de l'événement forcé  $r_1$ . Alors, le forçage est annulé et le système est toujours dans le mode précédemment actif (le mode nominal). Nous concluons qu'aucune défaillance ( $p_1$  ou  $p_2$ ) ne s'est produite dans la séquence d'activation  $d_1 d_2$ . En effet, les événements possibles à générer après cette séquence dans le modèle nominal sont :  $f_1, p_1, f_2$  et  $d_4$ . Donc, effectivement, l'événement  $r_1$  est impossible de se produire après  $d_1 d_2$ . Alors, il faut attendre la réponse du procédé pour confirmer le mode actif et la décision finale de diagnostic.

Nous présentons, par la suite, l'exemple d'un suivi d'événements générés par le procédé d'un système manufacturier de quatre mode de fonctionnement (voir le tableau 1). Nous considérons l'exemple de la séquence d'activation  $S = d_1 d_2 p_1 d_3 d_4 f_4 f_3 p_2 d_5 f_5 r_1 d_1 r_2 d_2$ . Nous appliquons notre algorithme de diagnostic 2 pour déterminer le mode actif et sa décision de diagnostic à l'occurrence de chaque événement généré par le procédé. D'après cet exemple, le système a commencé par le mode nominal  $N$  en passant par le mode dégradé  $\{D1, D3, D2\}$  pour revenir à nouveau au mode nominal  $N$ . Alors, le mode actif est le mode nominal  $N$ . Or, l'activation du mode nominal est faite par l'occurrence de la séquence de commutation  $p_1 p_2 r_1 r_2$ . Nous concluons que le système génère deux événements de panne  $\{p_1, p_2\}$  et deux événements de réparation  $\{r_1, r_2\}$ . À l'aide de la démarche proposée, nous pouvons détecter les fautes affectées aux ressources engagées dans un système multi-modèles de quatre modes de fonctionnement.

## V. CONCLUSION

Dans cet article, nous avons présenté une démarche, complètement définie pour le diagnostic et le changement de

TABLE I  
RÉSULTAT DE NOTRE ALGORITHME DE DIAGNOSTIC APPLIQUÉ AU  
SYSTÈME MANUFACTURIER À CINQ MACHINES

$\sigma$	$P_N$	$P_{D1}$	$P_{D2}$	$P_{D3}$	$Diag_N$	$Diag_{D1}$	$Diag_{D2}$	$Diag_{D3}$	$Coord$	$DF$
$d_1$	$d_1$	$\epsilon$	$d_1$	$\epsilon$	$NF$	$ND$	$ND$	$ND$	$(N,NF)$	$(N,NF)$
$d_2$	$d_2$	$\epsilon$	$\epsilon$	$\epsilon$	$ND$	$ND$	$ND$	$ND$	$(ND)$	-
$F_{Sup} = 1$ ; $Sup$ force l'événement $\sigma_{D1,N} = r_1$ ; $F_{Comp} = 1$										
Attendre la réponse du procédé										
$p_1$	$\epsilon$	$\epsilon$	$\epsilon$	$\epsilon$	$ND$	$ND$	$ND$	$ND$	$(ND)$	-
$d_3$	l'événement généré $d_3 \neq r_1 \Rightarrow$ le forçage est annulé; $F_{r_1} = 0 \Rightarrow DF(d_3) = (N,NF)$									
$d_3$	$\epsilon$	$d_3$	$\epsilon$	$d_3$	$ND$	$ND$	$ND$	$ND$	$(ND)$	-
$F_{Sup} = 1$ ; $Sup$ force l'événement $\sigma_{D3,D1} = r_2$ ; $F_{Comp} = 1$										
Attendre la réponse du procédé										
$d_4$	l'événement généré $d_4 \neq r_2 \Rightarrow$ le forçage est annulé; $F_{r_2} = 0 \Rightarrow DF(d_4) = (D1,F1)$									
$d_4$	$d_4$	$d_4$	$\epsilon$	$\epsilon$	$ND$	$F1$	$ND$	$ND$	$(D1,F1)$	$(D1,F1)$
$f_4$	$f_4$	$f_4$	$\epsilon$	$\epsilon$	$ND$	$F1$	$ND$	$ND$	$(D1,F1)$	$(D1,F1)$
$f_3$	$\epsilon$	$f_3$	$\epsilon$	$f_3$	$ND$	$ND$	$ND$	$ND$	$(ND)$	-
$F_{Sup} = 1$ ; $Sup$ force l'événement $\sigma_{D3,D1} = r_2$ ; $F_{Comp} = 1$										
Attendre la réponse du procédé										
$p_2$	$\epsilon$	$\epsilon$	$\epsilon$	$\epsilon$	$ND$	$ND$	$ND$	$ND$	$(ND)$	-
$d_5$	l'événement généré $d_5 \neq r_2 \Rightarrow$ le forçage est annulé; $F_{r_2} = 0 \Rightarrow DF(f_3) = (D1,F1)$									
$d_5$	$\epsilon$	$\epsilon$	$d_5$	$d_5$	$ND$	$ND$	$ND$	$F_1F_2$	$(D3,F_1F_2)$	$(D3,F_1F_2)$
$f_5$	$\epsilon$	$\epsilon$	$f_5$	$f_5$	$ND$	$ND$	$ND$	$F_1F_2$	$(D3,F_1F_2)$	$(D3,F_1F_2)$
$r_1$	$r_1$	$\epsilon$	$r_1$	$\epsilon$	$NF$	$ND$	$F_2$	$ND$	$(D2,F_2)$	$(D2,F_2)$
$d_1$	$d_1$	$\epsilon$	$d_1$	$\epsilon$	$ND$	$ND$	$F_2$	$ND$	$(D2,F_2)$	$(D2,F_2)$
$r_2$	$r_2$	$r_2$	$\epsilon$	$\epsilon$	$NF$	$ND$	$ND$	$ND$	$(N,NF)$	$(N,NF)$
$d_2$	$d_2$	$d_2$	$\epsilon$	$\epsilon$	$ND$	$ND$	$ND$	$ND$	$(ND)$	-
$F_{Sup} = 1$ ; $Sup$ force l'événement $\sigma_{D1,N} = r_1$ ; $F_{Comp} = 1$										
Attendre la réponse du procédé										

mode, qui permet d'assurer la commutation entre les modes de fonctionnement en présence des événements de défaillance. Nous avons proposé une nouvelle architecture de prise de décision de diagnostic d'un système multi-modèles à l'aide des éléments bien définis (un diagnostiqueur, un coordinateur, un superviseur et un comparateur). Pour faciliter l'application de notre démarche, nous avons proposé un algorithme permettant d'estimer l'état courant du système ainsi que les occurrences des pannes inobservables des ressources engagées dans chaque mode de fonctionnement.

## REFERENCES

- [1] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete event systems", *IEEE Transactions on Automatic Control*, vol. 40, pp. 1555-1575, 1995.
- [2] E. Boutleux. *Diagnostic et suivi d'évolution de l'état d'un système par reconnaissance des formes floues. Application au modèle du réseau téléphonique français*. PhD thesis, Université de Technologie de Compiègne, Lille, France, July 1996.
- [3] R. Debouk, S. Lafortune, and D. Teneketzis. Coordinated decentralized protocols for failure diagnosis of discrete event systems. *Discrete Event Dynamic Systems : Theory and Applications*, vol. 10, pp. 33-86, 2000.
- [4] Y. Pencolé and M.-O. Cordier, "A formal framework for the decentralized diagnosis of large scale discrete event systems and its application to telecommunication networks", *Artificial Intelligence (AIJ)*, vol. 164, pp. 121-170, 2005.
- [5] R. Su, W. Wonham, "Global and local consistencies in distributed fault diagnosis for discrete event systems", *IEEE Transactions on Automatic Control*, vol. 50, pp. 1923-1935, 2005.
- [6] N. Dangoumau, A. Toguyéni, E. Craye, "Functional and behavioral modelling for dependability in automated production systems", *Journal of engineering manufacture*, vol. 216, pp. 389-405, 2002.
- [7] M. Noureifath, "Extension de la théorie de la supervision à la surveillance et à la commande des systèmes à événement discret : application à la sécurité opérationnelle des systèmes de production", Thèse de Doctorat : Institut National des Sciences Appliquées de Lyon, France.
- [8] O. Kamach, L. Piétraç and E. Niel, "Multi-model approach to discrete events systems : application to operating mode management", *Mathematics and Computers in Simulation*, vol. 70, pp. 394-407, 2005.

- [9] G. Faraut, L. Piétraç and E. Niel, "Formal Approach to Multimodal Control Design: Application to Mode Switching", *IEEE Transactions on Industrial Informatics*, vol. 5, pp. 443-453, 2009.
- [10] A. El ghadouali, O. Kamach and B. Amami, "Identification of Compatible States in Switching Mode", *International Journal of Computer Applications*, vol. 72, pp. 42-45, 2013.
- [11] A. El ghadouali, O. Kamach and B. Amami, "Approche algébrique pour la gestion des modes de fonctionnement", *de Journal Européen des Systèmes Automatisées*, vol. 48, pp. 547-568, 2014.
- [12] B. A. Brandin. *Supervisory control of Discrete Event Systems with forcible events*. PhD thesis, Université Toronto, 1989.