Théorie du contrôle par supervision : Approche multi-modèle des modes de fonctionnement

Oulaid KAMACH, Laurent PIETRAC, Eric NIEL

Laboratoire d'Automatique Industrielle de Lyon
25, Avenue, jean Capelle, INSA de Lyon
69100, France

oulaid.kamach@insa-lyon.fr
http://www-lai.insa-lyon.fr

Résumé— **L'objectif principal de cet article consiste, d'une part à gérer la commande dans différents modes de fonctionnement et d'autre part à montrer l'applicabilité de notre proposition. Il s'agit de la multi-modélisation et la synthèse de contrôleurs. L'exemple retenu correspond à un système proposé par le groupe de travail "COSED" (Commande Opérationnelle des Systèmes à Evènements Discrets) devenu "INCOS" du "GdR MARS". Les modes de fonctionnement envisagés pour ce système sont le mode nominal et un mode dégradé. Dans le mode nominal, le système remplit parfaitement son office de production pour lequel, il a été conçu. Au cours du fonctionnement nominal, l'occurrence d'un événement d'exception (ou de défaillance) conduira le système en fonctionnement dégradé dans lequel il peut accomplir sa tâche en dépit de la défaillance considérée. Notre approche est une approche multi-modèle. En effet pour chacun des deux modes de fonctionnement sera associé un modèle correspondant. Ainsi, des contraintes de fonctionnement (spécifications) seront imposées sur chacun des deux modèles obtenus. En suite les deux modèles ainsi que leurs spécifications correspondantes seront étendus pour la caractérisation d'un mécanisme de commutation permettant le passage d'un modèle à l'autre. L'étape de synthèse sera basée sur ces différents modèles étendus.**

Mots-clés— **SED, automates à états, synthèse de contrôleurs, gestion des modes de fonctionnement.**

I. Introduction

Dans cet article nous allons présenter une proposition d'extension de la théorie de contrôle par supervision et une méthode d'analyse permet la prise en compte des modes de fonctionnement dégradés dûs à une panne d'un composant du système. Après une rapide présentation de l'approche classique utilisée dans cette théorie, nous justifierons le besoin d'extensions nécessaires à cette prise en compte. Nous présenterons ensuite notre approche à partir de l'étude d'un système proposé par le groupe "COSED" pour l'étude des méthodes de conception de la commande des Systèmes à Événements Discrets (SED) [5], [1], [2].

Nous n'allons présenter ici que des concepts généraux de l'approche classique, le lecteur désirant plus de détails pourra se référer à [3], [4], [9]. La théorie de la supervision des SED a été initiée par les travaux de Ramadge et Wonham. Cette approche repose sur la séparation claire entre le modèle du procédé et celui du spécification. Le procédé est considéré comme un générateur d'événements. Le superviseur observe les événements et son rôle consiste à interdire ou autoriser certains événements. L'ensemble des événements, appelé alphabet du procédé Σ est scindé en deux ensembles, l'ensemble des événements contrôlables Σ_c et l'ensemble des événements incontrôlables Σ_{uc} : $\Sigma = \Sigma_c \cup \Sigma_{uc}$. Un événement est dit contrôlable si son occurrence peut être interdite par le superviseur. En revanche l'occurrence d'un événement incontrôlable ne peut jamais être interdite par le superviseur. Ainsi l'association procédé-superviseur permet d'obtenir le langage désiré. Par définition celui-ci correspond aux trajectoires d'événements générées par le procédé respectant des spécifications imposées par le cahier des charges. Dans l'approche classique tous les événements sont observables et le temps n'intervient pas. L'étude se place donc à un niveau qualitatif : des modèles logiques seront exploités pour décrire le comportement du procédé. Lors de l'étude d'un système, le modèle du procédé est obtenu par le produit synchrone des modèles des composants de ce procédé. Le modèle des spécifications est obtenu par composition parallèle des automates de chaque spécification.

Généralement, dans le cadre de la théorie de contrôle par supervision, la majorité des approches existantes reposent sur l'utilisation d'un modèle unique représentant le procédé global. En effet, dans l'approche centralisée [3], [7], le procédé est modélisé par un modèle unique (RdP ou automate à état, etc.). Cette stratégie rend souvent la synthèse de contrôleurs très complexe et ne permet pas la prise en compte les différents modes de fonctionnement. Des extensions ont été proposées pour pallier au problème de la complexité, mais toujours en exploitant un seul modèle du procédé. En guise d'exemple, l'approche décentralisée [8], [10], [6], considère un seul modèle, l'alphabet, dit aussi l'ensembles des événements ou des transitions, est partitionné en plusieurs sous-alphabets.

Ces approches permettent de remédier au problème de la complexité, mais néanmoins elles s'appuient sur des concepts mathématiques rendant la preuve difficile. Par exemple, dans l'approche décentralisée une condition nécessaire de l'existence d'un superviseur garantissant le fonctionnement désiré relève de la notion de l'observabilité : pour montrer que le fonctionnement décentralisé obtenu est optimal, il est nécessaire que la notion de normalité soit également vérifiée. Dans l'approche hiérarchique [12], [11], il faut montrer la consistance hiérarchique pour pouvoir concilier les niveaux considérés.

La prise en considération des modes de fonctionnement et de leur gestion résultent ici de la définition d'un procédé et d'un superviseur élaborés spécifiquement pour chacun des modes retenus. Cette proposition nécessite au delà de la vérification des propriétés classiques relevant de la théorie de contrôle par supervision (contrôlabilité, blocage,...) et du fait de l'alternance des modes, de suivre explicitement chacun des modèles. Pour cela nous allons adopter la structure multi-modèle souvent utilisée dans l'automatique continue [19], [13], [14]. Dans cette structure, au lieu de se focaliser sur un seul modèle, plusieurs modèles sont considérés et chaque modèle décrit le comportement du système global dans un mode de fonctionnement donné. Il sera ainsi nécessaire d'évoquer dans l'alternance des modes le suivi des modèles du procédé ainsi que le suivi des modèles des spécifications. Le premier permettant de localiser les états où des événements apparaissent ou se raccrochent, le second localisant les états de récupération.

II. L'approche multi-modèle

Dans cette section nous allons nous intéresser à l'aspet modélisation. Nous proposons d'attribuer à chaque modèle automate du procédé un mode de fonctionnement. A un instant donné un seul mode est actif, les autres modes sont inactifs. L'activation d'un mode de fonctionnement relève essentiellement de l'apparition d'un événement d'exception. Le problème majeur lors de la commutation entre les différents modèles du procédé consiste à identifier l'état d'arrivée du modèle nouvellement actif. Cet état doit être compatible avec l'état à partir duquel l'événement d'exception s'est produit. Pour étudier ce problème, nous proposons tout d'abord le cas simple ne considérant que deux modes de fonctionnement. Chaque modèle du procédé est associé à un modèle de la spécification (1). S'il y a commutation de modes, il existe un passage d'une part d'un modèle de procédé dans un mode vers un autre modèle de procédé. De même les spécifications de mode étant associées à un mode particulier, il existera un passage d'un modèle de spécification à un autre.

Le mécanisme de commutation entre les modèles des procédés et quant à lui nécessite la mise en place de canaux d'information (voir 1).

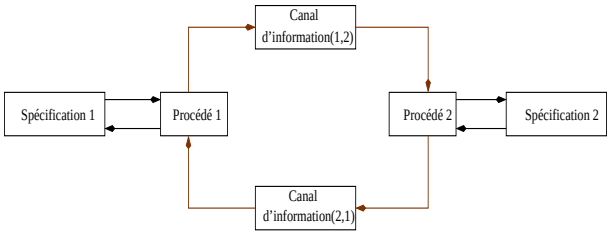

Fig. 1. structure multi-modèle

A. Commutation entre les modèles du procédé

Nous définissons $\Lambda = \{\lambda_1, \lambda_2\}$ comme l'ensemble des indices des modes de fonctionnement possibles.

Pour $\lambda \in \Lambda$, nous définissons le modèle automate $G_\lambda = (Q_\lambda, \Sigma_\lambda, \delta_\lambda, q_{0,\lambda}, Q_{m,\lambda})$ avec :

- Q_λ : ensemble fini d'état dans le mode λ ;
- Σ_λ : caractérise l'ensemble d'événements, dit aussi alphabet ;
- $\delta_\lambda : Q_\lambda \times \Sigma_\lambda \longrightarrow Q_\lambda$ est la fonction partielle de transition. Pour tout $q \in Q_\lambda$ et $\sigma \in \Sigma_\lambda$, on note par $\delta_\lambda(q, \sigma)!$ (resp. $\neg\delta_\lambda(q, \sigma)!$) si $\delta_\lambda(q, \sigma)$ existe (resp. n'existe pas). Cette fonction peut être étendue de la façon suivante $\delta_\lambda : Q_\lambda \times \Sigma_\lambda^* \longrightarrow Q_\lambda$;
- $q_{0,\lambda}$: état initial du mode λ ;
- $Q_{m,\lambda}$: ensemble des états finaux du mode λ, $Q_{m,\lambda} \subseteq Q_\lambda$.

Nous définissons Σ' comme l'ensemble des événements de commutation permettant de passer du modèle G_{λ_1} au modèle G_{λ_2} et *vice versa*. L'activation/désactivation est dûe à l'occurrence d'un événement de commutation. Dans notre cas (deux modes de fonctionnement sont possible), l'alphabet Σ' est constitué de deux événements $\alpha_{\lambda_1,\lambda_2}$ et $\alpha_{\lambda_2,\lambda_1}$. L'événement $\alpha_{\lambda_1,\lambda_2}$ (resp. $\alpha_{\lambda_2,\lambda_1}$) permet de désactiver le modèle G_{λ_1} (resp. G_{λ_2}) et d'activer le modèle G_{λ_2} (resp. G_{λ_1}).

Remarque 1: L'alphabet global Σ est scindé en trois sous-ensembles : l'alphabet Σ_{λ_1} du mode λ_1, l'ensemble des événements de commutation Σ' et l'alphabet $\Sigma_{\lambda 2}$ du mode λ_2.
$\Sigma = \Sigma_{\lambda_1} \cup \Sigma' \cup \Sigma_{\lambda_2}$

Nous considérons les hypothèses suivantes :

1. $\Sigma_{\lambda_i} \cap \Sigma' = \emptyset$ *pour* $i \in \{1, 2\}$;
2. $\Sigma_{\lambda_1} \cap \Sigma_{\lambda_2} \neq \emptyset$.

Rappelons qu'à un instant donné, un seul modèle est actif. Dans notre cas d'étude, le modèle initialement actif est G_{λ_1}, le modèle G_{λ_2} est inactif. Ce constat nous a amené à étendre les modèles G_{λ_1} et G_{λ_2}. De ce fait, nous ajoutons à chaque modèle G_λ, où $\lambda \in \{\lambda_1, \lambda_2\}$, un état particulier $q_{in,\lambda}$, appelé état inactif du modèle, inspiré du concept d'état puits proposé par [17] et de l'état non significatif proposé par [18]. Le modèle étendu $G_{\lambda,et}$ actif à un instant donné est donc le seul modèle ne se trouvant pas dans son état inactif. Nous ajoutons également l'ensemble des événements de commutation Σ' à l'alphabet Σ_{λ_1} du modèle G_{λ_1} et à l'alphabet Σ_{λ_2} du modèle G_{λ_2}. D'une manière formelle, pour chaque $\lambda \in \Lambda$ nous définissons le modèle étendu du modèle $G_\lambda = (Q_\lambda, \Sigma_\lambda, \delta_\lambda, q_{0,\lambda}, Q_{m,\lambda})$ comme suit :
$G_{\lambda,et} = (Q_{\lambda,et}, \Sigma_{\lambda,et}, \delta_{\lambda,et}, q_{0,\lambda,et}, Q_{m,\lambda,et})$ avec

- $Q_{\lambda,et} = Q_\lambda \cup \{q_{in,\lambda}\}$;
- $\Sigma_{\lambda,et} = \Sigma_\lambda \cup \Sigma'$;
- $q_{0,\lambda,et} = q_{0,\lambda}$ si $\lambda = \lambda_1$;
- $q_{0,\lambda,et} = q_{in,\lambda}$ si $\lambda = \lambda_2$;
- $Q_{m,\lambda,et} = Q_{m,\lambda}$;
- La fonction de transition étendue est définie comme suit :
  - $\forall q \in Q_\lambda$ et $\forall \sigma \in \Sigma_\lambda$ si $\delta_\lambda(q, \sigma)$ existe alors $\delta_{\lambda,et}(q, \sigma) := \delta_\lambda(q, \sigma)$;
  - $\forall q \in Q_{\lambda_i}$ à partir duquel l'événement de commutation $\alpha_{\lambda_i,\lambda_j}$ peut être généré alors $\delta_{\lambda_i,et}(q, \alpha_{\lambda_i,\lambda_j}) := q_{in,\lambda_i}$ avec $(i \neq j$ et $i, j \in \{1, 2\})$;
  - $\delta_{\lambda_j,et}(q_{in,\lambda_j}, \alpha_{\lambda_i,\lambda_j})$ avec $(i, j \in \{1, 2\}$ et $i \neq j)$ sera définie ultérieurement.

Initialement le modèle du procédé est $G_{\lambda_1,et}$. Son état initial est q_{0,λ_1}, en revanche pour le modèle $G_{\lambda_2,et}$, son état initial est q_{in,λ_2}.

Le principal objectif du mécanisme de commutation, entre les deux modèles du procédé G_{λ_1} et G_{λ_2}, est de suivre explicitement l'évolution du procédé dans différents modes de fonctionnement et de déterminer ainsi un état adéquat à partir duquel le modèle $G_{\lambda_2,et}$, (resp. $G_{\lambda_1,et}$) sera activé (resp. réactivé).
L'événement de commutation $\alpha_{\lambda_1,\lambda_2}$ peut survenir dans plusieurs états du modèle G_{λ_1}. Cet événement permettera d'inactiver le modèle G_{λ_1} et également activer le modèle G_{λ_2}, il conduit le modèle G_{λ_1} depuis un état $q \in Q_{\lambda_1}$ vers l'état inactif q_{in,λ_1}. En revanche, il conduit le modèle G_{λ_2} depuis l'état inactif q_{in,λ_2} vers un état $q \in Q_{\lambda_2}$.
Le canal d'information est une fonction permettant de déterminer l'état de départ du modèle G_{λ_2} tout en se basant sur une trace du modèle G_{λ_1}. Cette trace est définie par des séquences d'événements générés dans le modèle G_{λ_1} et menant vers un état où l'événement de commutation peut se produire.
Formellement cette fonction est définie comme suit :

$$\pi_{\lambda_i,\lambda j} : (\Sigma_{\lambda_i})^* \longrightarrow (\Sigma_{\lambda_j})^* \text{ telle que , } i,j \in \{1,2\} \; et \; i \neq j$$

$$\pi_{\lambda_i,\lambda_j}(\varepsilon) = \varepsilon$$

$$\pi_{\lambda_i,\lambda_j}(s\sigma) = \begin{cases} \pi_{\lambda_i,\lambda_j}(s)\sigma & \text{si } \sigma \in \Sigma_{\lambda_i} \cap \Sigma_{\lambda_j} \\ \pi_{\lambda_i,\lambda j}(s) & \text{si } \sigma \in \Sigma_{\lambda_i}/\Sigma_{\lambda_j} \end{cases}$$

Cette définition de la projection n'apporte aucune contrainte sur les définitions de Σ_{λ_i} et de Σ_{λ_j}. Dans le cas particulier où $\Sigma_{\lambda_j} \subseteq \Sigma_{\lambda_i}$, cette projection correspond à la projection naturelle classiquement utilisée dans la théorie de contrôle par supervision. La fonction de projection $\pi_{\lambda_1,\lambda 2}$ observe uniquement l'occurrence des événements appartenant à l'intersection $\Sigma_{\lambda_1} \cap \Sigma_{\lambda_2}$. Cela nous permet de suivre exclusivement les éléments communs entre les modèles G_{λ_1} et G_{λ_2}.
Les propositions suivantes établissent un cadre formel pour la détermination de l'état de départ du modèle $G_{\lambda_{2,et}}$ suite à l'occurrence de l'événement de commutation $\alpha_{\lambda_1,\lambda_2}$ et pour celle de l'état de retour du modèle $G_{\lambda_{1,et}}$ suite à l'occurrence de l'événement de commutation $\alpha_{\lambda_1,\lambda_2}$.

Proposition II.1: $\forall s \in L(G_{\lambda_1})$ telle que $\alpha_{\lambda_1,\lambda_2} \in suiv(s)$[1]. L'état de départ du modèle G_{λ_2} est donné comme suit :

$$\delta_{\lambda_2,et}(q_{in,\lambda_2},\, \alpha_{\lambda_1,\lambda_2}) = \delta_{\lambda_2}(q_{0,\lambda_2},\, \pi_{\lambda_1,\lambda_2}(s)) \; \blacksquare$$

Proposition II.2: $\forall s \in L(G_{\lambda_1})$, $\alpha_{\lambda_1,\lambda_2} \in suiv(s)$ et $\forall s' \in L(G_{\lambda_2}, \pi_{\lambda_1,\lambda_2}(s))$[2] telle que $\alpha_{\lambda_2,\lambda_1} \in suiv(s')$. L'état de retour du modèle G_{λ_1} est donné comme suit :

$$\delta_{\lambda_1,et}(q_{in,\lambda_1},\, \alpha_{\lambda_2,\lambda_1}) = \delta_{\lambda_1}(q_{0,\lambda_1},\, \pi_{\lambda_1,\lambda_2}(s)\pi_{\lambda_2,\lambda_1}(s')) \; \blacksquare$$

Pour de plus amples informations concernant la démonstartion, le lecteur peut se référer à [15].

B. commutation entre les modèles des spécifications

D'après la théorie de contrôle par supervision le procédé est soumis à des objectifs de contrôle (spécifications). Dans l'approche multi-modèle, nous définissons des spécifications de contrôle pour chaque mode de fonctionnement. Or les modes de fonctionnement considérés peuvent être différents, par conséquent les spécifications respectives seront distinctes. Par ailleurs, dans la section précédente, nous avons traité le mécanisme de commutation entre les différents modèles du procédé en étendant ces derniers et en déterminant les états de départ et de retour pour chaque modèle. Les spécifications devant suivre explicitement l'évolution du procédé, elles devront également être soumises au mécanisme de commutation. En effet, pour $\lambda \in \Lambda$, la spécification $S_\lambda = (X_\lambda, \Sigma_\lambda, \xi_\lambda, x_{0,\lambda}, X_{m,\lambda})$ est associée au procédé $G_\lambda = (Q_\lambda, \Sigma_\lambda, \delta_\lambda, q_{0,\lambda}, Q_{m,\lambda})$.
Initialement, le modèle de la spécification S_{λ_1} est activé, *i.e.* l'automate associé se trouve dans l'état initial x_{0,λ_1}. A l'occurrence de l'événement de commutation $\alpha_{\lambda_1,\lambda_2}$ l'automate $G_{\lambda_1,et}$ sera conduit vers son état inactif q_{in,λ_1}. Dans ce cas le modèle de la spécification S_{λ_1} sera étendu en rajoutant un état inactif x_{in,λ_1} vers lequel elle sera conduite jusqu'à l'occurrence de l'événement de commutation $\alpha_{\lambda_2,\lambda_1}$ lui permettant ainsi de rejoindre un état $x \in X_{\lambda_1}$. Comme précédemment, le problème de la commutation est posé. En effet, si le modèle du procédé G_{λ_i}, $(i \in \{1,2\})$ possède plusieurs états de départ et/ou de retour, la spécification correspondante peut posséder éventuellement plusieurs états de départ et/ou de retour afin de suivre correctement l'évolution du procédé. Notons que le mécanisme de commutation développé dans la section précédente ne s'adapte pas aux spécifications car la trace générée dans le procédé G_{λ_i} peut contenir des événements qui peuvent être interdits par le modèle de la spécification S_{λ_j}. Pour plus d'informations concernant l'étude des spécifications étendues, le lecteur peut se référer à [16].

III. Application

Nous rappelons que le système étudié est composé de trois chaînes fonctionnelles assurant un mouvement vertical, un mouvement horizontal et une aspiration. Nous appliquerons notre approche en considérant deux modes de fonctionnement, un mode de fonctionnement nominal à partir duquel un événement de panne peut survenir sur le vérin assurant le mouvement horizontal et le mode dégradé permettant de terminer la tâche en cours si cela est possible, et de mettre le système en position de sécurité. Dans cet exemple, l'ensemble des évé-

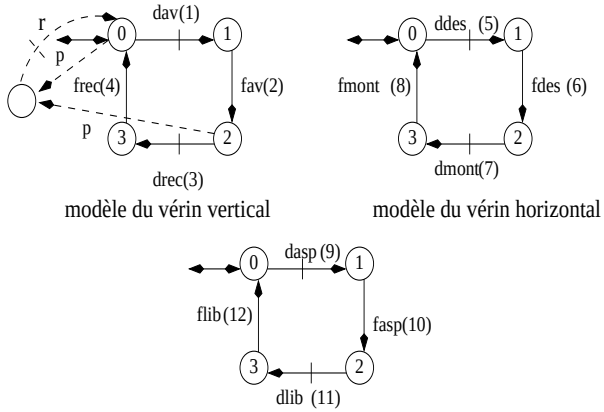

Fig. 2. modèles des composants

[1] $suiv(s)$ est l'ensemble des événements qui suivent la chaîne s
[2] $= \{u \in (\Sigma_{\lambda_j})^* / \delta_{\lambda_j}(\delta_{\lambda_j}(q_{0,\lambda_j}, \pi_{\lambda_i,\lambda_j}(s)), u)!\}$

nements de commutation est : $\Sigma' = \{r, p\}$ (r = événement de réparation, p = événement de panne), $\Sigma_n = \{dav,\ fav,\ drec,\ frec,\ ddes,\ fdes,\ dmont,\ fmont,\ dasp,\ fasp,\ dlib,\ flib\}$ et $\Sigma_d = \{ddes,\ fdes,\ dmont,\ fmont,\ dasp,\ fasp,\ dlib,\ flib\}$.

événements contrôlable	numérotation
dav : demande d'avance	1
drec : demande de recule	3
ddes : demande descente	5
dmont : demande de montée	7
dasp : demande d'aspiration	9
dlib : demande de libération	11
événements incontrôlables	numérotation
fav : fin d'avance	2
frec : fin de recule	4
fdes : fin descente	6
fmont : fin de montée	8
fasp : fin d'aspiration	10
flib : fin de libération	12

TABLE I
LISTE DES ÉVÉNEMENTS DU MODE NOMINAL

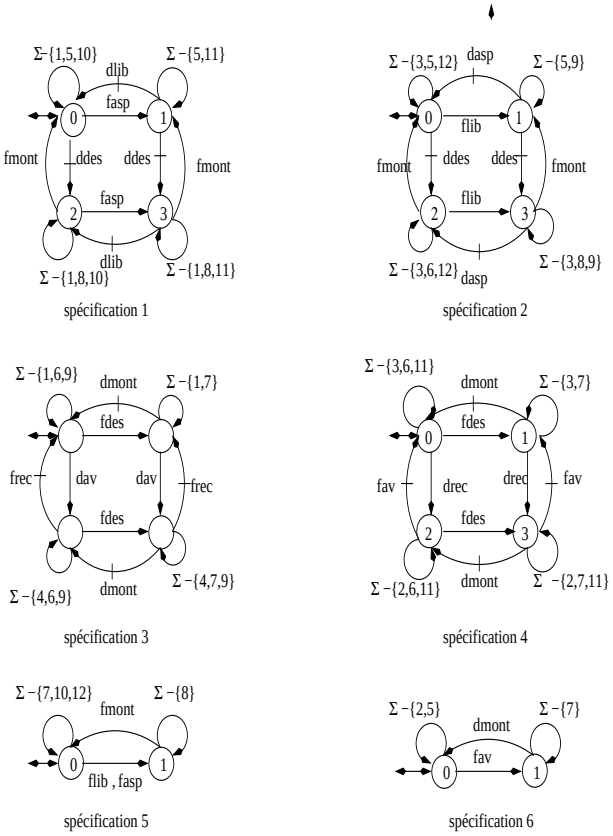

Fig. 3. automates correspondants aux contraintes du mode nominal

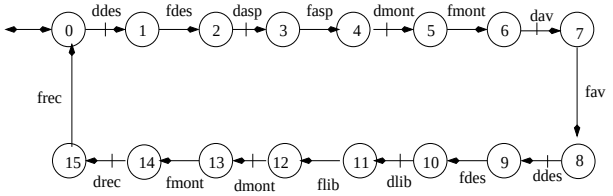

Fig. 4. contrôleur du mode nominal

A. *Mode nominal*

A.1 Modélisation du procédé G_n

Les modèles présentés ici ne seront ni expliqués ni justifiés, les numéros entre parenthèses correspondent à ceux utilisés dans le logiciel "TCT", récupérable à partir de [4]. Le modèle nominal du procédé en mode nominal G_n est obtenu par composition parallèle des trois automates de la figure 2 sans tenir compte des événements de commutation (r et p). Le modèle nominal comporte 64 états et 192 transitions, mais il n'est pas utile de le représenter.

A.2 Modélisation des spécifications S_n

Pour le mode nominal, les spécifications sont les suivantes :

1. L'avance ne peut se faire que si le vérin vertical est en position haute et si une pièce est aspirée ;

2. Le recul ne peut se faire que si le vérin vertical est en position haute et qu'une pièce a été libérée ;

3. Une pièce ne peut être saisie que si le vérin vertical est en position basse et le vérin horizontal en position rentré ;

4. Une pièce ne peut être libérée que si le vérin vertical est en position basse et le vérin horizontal est sorti ;

5. Le vérin vertical ne monte que si une pièce est aspirée ou libérée ;

6. Le vérin vertical ne descend que si le vérin horizontal est sorti.

Le modèle global S_n du spécification du mode nominal est obtenu par composition paralléle des six automates (3). Le modèle du procédé sous contrôle (figure 4) est obtenu par le produit de G_n et de S_n.

B. *Mode dégradé*

B.1 Procédé dégradé

Dans le cas de notre exemple, le mode dégradé peut être enclenché uniquement si l'événement de défaillance se produit à partir des états "0" ou "2" du modèle du vérin horizontal. Nous considérons que ce composant est alors bloqué et donc qu'il ne génère aucun événement. Il n'apparaît donc plus dans le nouveau modèle du procédé en mode dégradé. le procédé dégradé G_d est donc composé à partir des modèles des composants de mouvement vertical et d'aspiration.

B.2 Les spécifications du mode dégradé

Les spécifications du mode dégradé ne font pas partie du cahier des charges proposé par [5]. Nous considérons qu'en cas de défaillance la réaction du système doit être de se mettre dans un état non dangereux pour lui-même, la pièce et l'opérateur et en même temps de se mettre dans un état le plus proche de l'état initial du mode nominal afin de faciliter le démarrage. Ainsi la pièce doit être déposée si cela est possible. Les spécifications retenues sont les suivantes :

1. La pièce ne peut être libérée qu'en position basse ;

2. Le vérin ne peut monter que si la pièce est libérée ;
3. La pièce ne peut pas être aspirée.

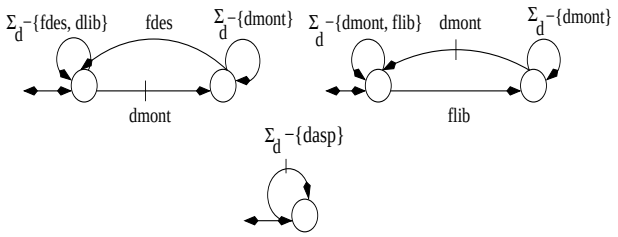

Fig. 5. modèles des spécifications du mode dégradé

Le modèle global de la spécification dégradé est obtenu par la composition paralléle de 3 automate de la figure 5.

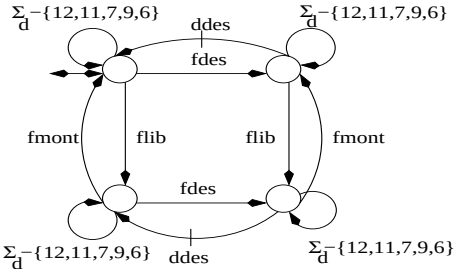

Fig. 6. spécification du mode dégradé

C. Modèles étendus

C.1 Procédé étendu

Le procédé dégradé G_d est donc composé à partir des modèles des composants de mouvement vertical et d'aspiration. Cependant les modèles G_2 (vérin vertical) et G_3 (venteuse) de la figure 2 ne peuvent pas être utilisés car lorsque la panne survient le procédé a évolué et l'état initial du procédé dégradé ne correspond plus à l'état initial du modèle nominal. La panne survenant dans l'état 2 ou l'état 0 du composant G_1 peut donc survenir entre les états 0 et 6 ou entre l'état 8 et 14 (inclus) du procédé supervisé (4). Supposons que cette panne arrive dans l'état 8. Pour déterminer l'état de départ du mode dégradé nous cherchons dans quel état se trouve chaque composant du procédé. La trace ayant permis d'atteindre cet état est : $trace = ddes.fdes.dasp.fasp.dmont.fmont.dav.fav$ L'application de la fonction de la projection sur la trace générée dans le mode nominal donne :
$\pi_{n,d}(trace) = ddes.fdes.dasp.fasp.dmont.fmont.$ Donc d'après la proposition II.1, l'état de départ du procède du mode dégradé est

$$\delta_{d,et}(q_{in,d},\, (\alpha_{n,d} = p)) = \delta_d(q_{0,d},\, \pi_{n,d}(trace)) = q_{8,d}.$$

L'exemple donné précédemment correspondait à une panne arrivant dans un état particulier du procédé sous contrôle. Que se passe-t-il si cette panne survient dans un autre état : dans les états 9 à 14 ou 0 à 6 de figure 3 ?. Le reste du procédé évolue et donc les modèles construits pour le mode dégradé en fonction de l'état 8 ne peuvent pas être utilisés à cause du changement d'état de départ. Il en est de même pour les modèles des spécifications dont l'état initial doit être cohérent avec celui du modèle du procédé.

Pour construire le procédé étendu du mode dégradé, nous appliquons la proposition II.1 sur toutes les traces du procédé sous contrôle du mode nominal et nous obtenons le modèle étendu du mode dégradé de la figure 7. L'événement de panne peut être représenté par plusieurs événements significatifs p_i. Chaque événement est associé à un et un seul état du mode dégradé. Cet représentation permet d'enlever le problème de déterminisme. Formellement

$$p = p_i \;\text{ si } \delta_{d,et}(q_{in,d}, \pi_{n,d}(s)) = q_{i,d}$$

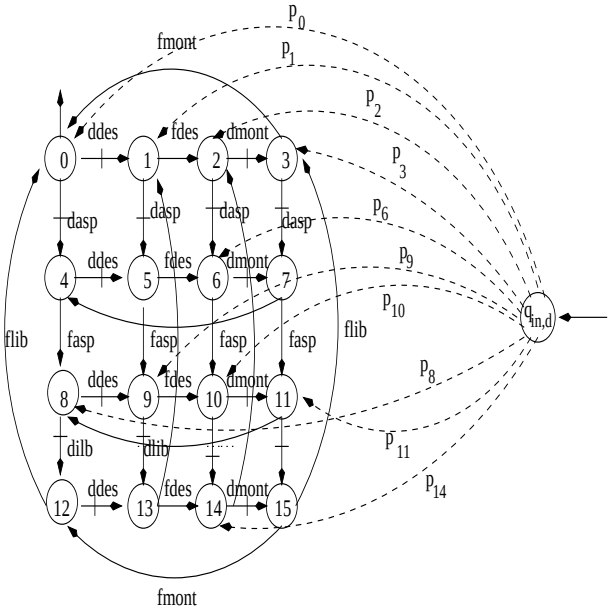

Fig. 7. modèle étendu du mode dégradé

Le modèle de la figure 7 n'est pas complet. En effet, l'objectif de cette figure est de mettre l'accent sur les états de départ du modèle à partir de l'état inactif $q_{in,d}$ (occurrence d'un événement de panne). L'événement de réparation n'apparait donc pas dans le modèle, puisque cet événement ne fait que conduire le modèle vers l'état inactif $q_{in,d}$.

C.2 Spécification étendue

Puisque les spécifications doivent suivre explicitement l'évolution du procédé, elles doivent aussi soumises au même mécanisme de commutation. La figure 6 représente le modèle automate global des spécifications du mode dégradé. Le modèle de la spécification étendue est représenté par la figure 8. Par manque de place, le mécanisme permettant de construire ce modèle ne sera pas représenté ici. Voir [16] pour plus de détails.

C.3 Contrôleur étendu du mode dégradé et nominal

La vérification de la contrôlabilité et du non-blocage du contrôleur étendu sont effectués à l'aide du logiciel "TCT". Le contrôleur résultant du mode dégradé admet 11 états et 16 transitions comme le montre la figure 9. La figure 10 représente le contrôleur étendu du mode nominal.

Dans le modèle de la figure 9 (resp. de la figure 10), les transitions de commutation sont soit entrantes soit sortantes de l'état inactif $q_{in,d}$ (reps. $q_{in,n}$). Mais pour des

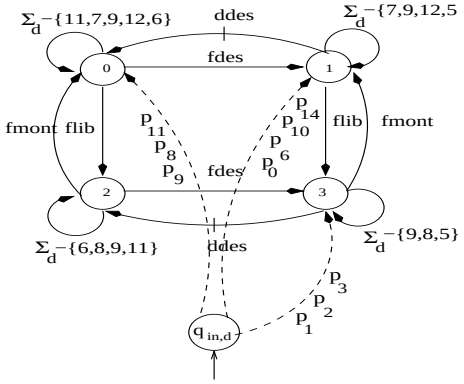

Fig. 8. spécification étendue global du mode dégradé

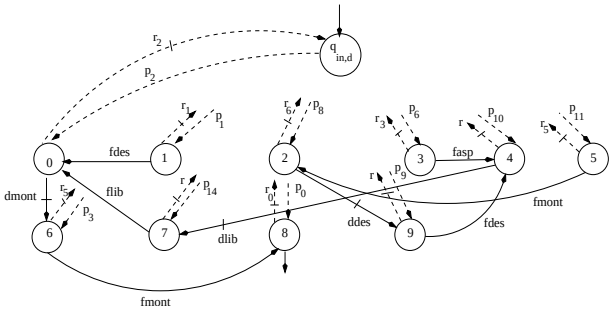

Fig. 9. contrôleur étendu du mode dégradé

raisons de lisibilité, ces transitions ne sont pas acheminées vers leurs états.

IV. Conclusion

Cet article nous a permis de présenter l'extension de la théorie de contrôle par supervision par la prise en compte de différents modes de fonctionnement. A partir d'un exemple nous avons montré l'utilisation de différents modèles du procédé et proposé un mécanisme systématique de détermination de l'état départ/retour d'un modèle. Nous avons également justifié et présenté des modèles étendus permettant la prise en compte de différents états de départ possibles dans un mode donné, évitant ainsi de construire autant de modèles que d'états de départ/retour possibles. Nous avons aussi étendu les modèles des spécifications de chaque mode de fonctionnement afin de prendre en compte le mécanisme de commutation entre les modèles des procédés et ceux des spécifications. Les contrôleurs étendus du mode nominal et dégradé sont contrôlables et non-bloquants.

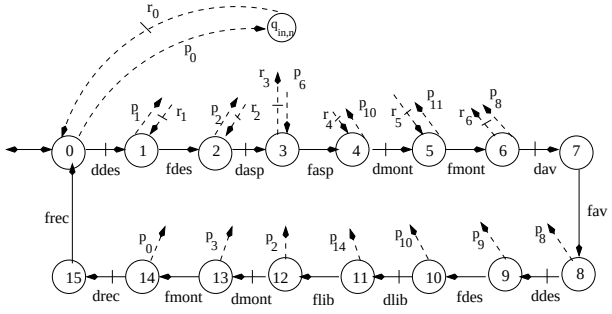

Fig. 10. contrôleur étendu du mode nominal

Références

[1] D. Gouyou, J.F. Pétin et A. Gouin. « Modèles du procédé et de ses spécifications pour la synthèse de la commande, »MSR, Metz, France, pp. 45-60, 2003.

[2] A. Philippot, A. Tajer. « Synthèse de la commande spécifiée en Grafcet : application à un préhenseur pneumatique, »MSR, Metz, France, pp. 61-75, 2003.

[3] P. Ramadge, W. Wonham. « The control of Discret Event Systems, »In Proc. of the IECE, Vol. 77, p. 81-98. 1987.

[4] W. M. Wonham. *Notes on control of discrete-event systems*, notes de cours, depertment of Electrical and Coputer Engineering, University of Toronto, http ://www.control.toronto.edu/people/profs/wonham/ 2002.

[5] site web : japura.lurpa.ens-cachan.fr/cosed.

[6] T. -S. Yoo, S. Lafortune. « Decentralized supervisory control : a new architecture with a dynamic decision fusion ruleă, »6th international workshop on discrete event systems, Saragosse, Espagne, pp. 11-17, 2-4 octobre 2002.

[7] B. H. Krogh. « Controlled Petri nets and maximally permissive feedback logic, »In Proc. 25th Annual Allerton Conference, University of Illinois at Urbana-Champaingn, p. 317-326, Oct. 1987.

[8] F. Lin, W. M. Wonham. « Decentralized supervisory control of discrete-event systems, » Information Sciences, Vol. 41, no. 2, pp. 199-224, 1988

[9] G. C. Cassandras, S. Lafortune. « Introduction to Discrete Event Systems, » 1st edn. Boston : Kluwer Academic Publishers, pp. 822, 1999.

[10] K. Rudie, W. M. Wonham. « Think globally, act locally : Decentralized supervisory control, »IEEE trans. on automat. Contr, 37(11) :1692-1708, November 1992.

[11] K. C. Wong, J. G. Thistle, R. P. Malhame et H. H. Hoang. « Supervisory Control of distributed Systems : conflict resolution. » Discrete Event Dynamics Systems, Vol. 10, pp. 131-186, 2000.

[12] H. S. Zhong. *Hierarchical control of Discrete Event systems.* Ph.D. Thesis : Departement of Electrical engineering, University of Toronto, Toronto, Canada, pp. 155, 1992.

[13] S. Talmoudi. « Multi-modéle et multi-commande neuronaux pour la conduite numerique des systmes non-linéaires et non-stationnaires, »IEEE, Cifa, Nante, France, pp. 871-876, 2002.

[14] P. Charbonnaud, F. Rotella et S. Médar. « Process Operating mode Monitoring Process : Switching Online the Right Controller, »IEEE transactions on systems, Man and Cybernetics, Part C 31(1) : pp 77-86, 2002.

[15] O. Kamach, S. Chafik, L. Pietrac et E. Niel. « Representation of reactive system with different models, »IEEE SMC, Reference TA2L4 in CDROM, Yasmine Hammamet, Tunisie. Octobre 6-9, 2002.

[16] O. Kamach, S. Chafik, L. Pietrac et E. Niel. « Multi-model approach for discrete event systems, »IEEE CESA, référence S2-R-00-0315 in CD ROM, école centrale de Lille, France, 2003

[17] M. Nourelfath. « Extension de la théorie de la supervision à la surveillance et à la commande des Systèmes à Evénements Discrets. Thèse de Doctorat : Institut National des Scineces Appliquées de Lyon, p 145, France, 1997.

[18] N. Dangoumau. « Contribution à la gestion des modes des systèmes automatisés de production. Thèse de Doctorat : Université des Sciences et Technologiques de Lille, p 181, France 2000.

[19] M. Zefran et J. Burdick. « Design of switching controllers for systems with changing dynamics, »proc. 37th conf on Decision and control, pp. 2113-2118, 1998.