Synthèse de contrôleur pour les Systèmes non-déterministes

Oulaid KAMACH, Laurent PIETRAC, Eric NIEL

Laboratoire d'Automatique Industrielle de Lyon
25, Avenue, jean Capelle, INSA de Lyon
69621, Villeurbanne France

prénom.nom@insa-lyon.fr
http://www-lai.insa-lyon.fr

Résumé— **Le travail présenté dans ce papier porte sur la synthèse de contrôleur des systèmes non-déterministes. Notre contribution se base sur la théorie de contrôle par supervision initiée par les travaux de Ramadge et Wonham. Notre objectif est d'enrichir cette théorie par l'étude de la supervision des systèmes non-déterministes. La classe de systèmes non-déterministes que nous considérons sont les systèmes à plusieurs états initiaux. L'ensemble des états initiaux caractérise un ensemble de dynamiques distinctes. Nous proposons une approche permettant de démontrer qu'il existe un superviseur unique permettant d'aboutir au langage désiré, bien que le procédé ait plusieurs états initiaux. Nous formulons ainsi les conditions nécessaires et suffisantes d'existence d'un tel superviseur.**

Mots-clés— **SED, automates à états finis, synthèse de contrôleurs, systèmes non-déterministes, systèmes déterministes, théorie de contrôle par supervision.**

I. Introduction

S'agissant de la synthèse de contrôleurs, la plupart des approches, reposant sur la théorie de contrôle par supervision, se facalisent sur l'étude de la commande des systèmes déterministes [2], [3]. Beaucoup d'attentions ont été également apportées à la commande du Systèmes à Événements Discrets (SED) sous observation partielle [7], [4], dans laquelle seulement un sous-ensemble d'événements est accessible au superviseur. Pour de tels systèmes, les conditions nécessaires et suffisantes d'existence d'un superviseur garantissant les contraintes imposées par un cahier des charges ainsi que les algorithmes de synthèse aussi bien hors-line [12], [5], [6], [8] qu'en ligne [11], ont été démontrées. Les systèmes partiellement observés peuvent néanmoins présenter un comportement non-déterministe. Ce comportement ne relève pas forcément de l'observation partielle, mais plutôt du fait que le système est préalablement non-déterministe ou bien parce que seulement un modèle partiel du système est disponible par le fait qu'une partie ou toutes ses activités internes ne sont pas modélisées. Contrairement au SED déterministe, dont leur comportement est entièrement décrit par du langage généré, la description du comportement des systèmes non-déterministes exige beaucoup plus d'amélioration et de détail. Tandis que dans le cas de déterministe, le langage désiré peut être adéquatement exprimé sous forme d'expressions régulières, ce n'est pas toujours vrai pour le système non-déterministe. En effet, pour formellement spécifier le comportement légal du système commandé, il peut être nécessaire d'étudier, au delà du langage autorisé, le degré de non-déterministe pour lequel le système commandé permet de maintenir. Dans le cadre de la théorie des automates, non-déterminisme est défini par le fait que plusieurs transitions portant la même étiquette (événement) sortent d'un même état, ou lorsque l'automate possède plusieurs états initiaux. Dans ce papier nous nous intéressons à un cas particulier des systèmes non-déterministes. Ce cas est représenté par des automates ayant plusieurs états initiaux. Chaque état initial de l'automate décrit ainsi un comportement du système, *i.e.*, il n'existe pas ainsi un comportement unique du système (ce qui est le cas pour les systèmes déterministes). Pour chacun de ces comportements, un langage désiré est associé. Nous devons alors montrer qu'il existe un superviseur unique qui permet de garantir le langage désiré quel que soit l'état initial de l'automate.

II. Theorie de contrôle par supervision

Dans la théorie de contrôle par supervision, et par analogie à l'automatique continue, le fonctionnement en boucle ouverte est caractérisé par un seul schéma-bloc contenant uniquement le modèle automate du procédé (qui évolue spontanément).
Le modèle automate déterministe est un *5-uplet* noté G pour le procédé et donné par :

$$G = (Q,\, \Sigma,\, \delta,\, q_0,\, Q_m) \text{ où } :$$

- Q : ensemble d'états, caractérisant par exemple l'ensemble des activités possibles du procédé ;
- Σ : alphabet ou ensemble d'événements générés par le procédé ;
- q_0 : état initial du procédé ;
- $\delta : Q \times \Sigma \longrightarrow Q$ est la fonction partielle de transition telle que pour un état d'arrivée ($q' \in Q$) et un état de départ ($q \in Q$) on associe un événement ($\sigma \in \Sigma$) *i.e.* $\delta(q, \sigma) = q'$. Cette fonction permet de représenter par exemple les changements possibles d'activités du procédé. Elle peut être étendue comme suit $\delta : Q \times \Sigma^* \longrightarrow Q$ telle que pour toute séquence d'événements $s \in \Sigma^*$ et pour tout $\sigma \in \Sigma$, $\delta(q, s\sigma) = \delta(\delta(q, s), \sigma)$ et $\forall q \in Q$, $\delta(q, \varepsilon) = q$;

– Q_m : ensemble d'états marqués (aboutir à un état marqué depuis l'état initial du procédé signifie l'accomplissement d'une tâche du procédé).

Deux langages formels caractérisent le comportement du procédé G : le langage généré $L(G)$, et le langage marqué $L_m(G)$. Le langage $L(G)$ est préfixe-clos c'est à dire qu'il contient toutes les chaînes d'événements générées depuis l'état initial du procédé. Le langage $L_m(G)$ est formé par l'ensemble des séquences d'événements permettant d'atteindre un état marqué depuis l'état initial. Formellement :

$$\begin{array}{rcl} L(G) & = & \{s \in \Sigma^* \mid \delta(q_0, s)!\} \\ L_m(G) & = & \{s \in \Sigma^* \mid \delta(q_0, s) \in Q_m\} \end{array}$$

Afin de garantir les contraintes imposées par un cahier des charges, un superviseur est associé au procédé. Le superviseur restreint le comportement du procédé en boucle ouverte. On parle alors de comportement en boucle fermée. Formellement un superviseur est une application définie par :

$$S : L(G) \longrightarrow \Gamma \qquad \text{avec} \qquad \Gamma = \{\gamma \in Pwr(\Sigma)^1 \mid \Sigma_u \subseteq \gamma\}$$

Pour chaque séquence d'événements s générée dans le procédé G, le superviseur fournit la liste d'événements autorisés $\gamma \in Pwr(\Sigma)$. Cette liste contient inévitablement l'ensemble des événements incontrôlables ($\Sigma_u \subseteq \gamma$) puisque ces derniers ne peuvent pas être inhibés par le superviseur. Le langage correspondant au comportement en boucle fermée, noté $L(S/G)$, se définit de la manière récursive suivante :

$$\begin{array}{l} \varepsilon \in L(S/G) \\ (s\sigma \in L(S/G)) \Leftrightarrow ((s \in L(S/G)), (s\sigma \in L(G)) \,\text{et}\, \sigma \in S(s)) \end{array}$$

L'existence du superviseur est donc assujettie au concept de contrôlabilité [1] développé ci-après.

Définition 1 (Langage contrôlable)
Soit l'automate G de langage $L(G)$ et ayant Σ_u comme ensemble d'événements incontrôlables, un sous langage $K \subseteq L(G)$ est dit contrôlable si et seulement si $\overline{K}\Sigma_u \cap L(G) \subseteq \overline{K}$ ($\overline{K}$ est le préfixe-clos de K). ♦

Cette définition est équivalente à la vérification de la proposition suivante :

$$\forall s \in \overline{K},\ \forall \sigma \in \Sigma_u \text{ si } s\sigma \in L(G) \text{ alors } s\sigma \in \overline{K}$$

En d'autres termes, pour toute séquence d'événements s appartenant à $\overline{K}$ et pour tout événement incontrôlable $\sigma \in \Sigma_u$, si la séquence d'événements appartient à $L(G)$ alors elle doit également appartenir à $\overline{K}$. Dans le cas contraire, cela signifie que le superviseur devrait interdire l'occurrence possible d'un événement incontrôlable, ce qui serait contraire à la définition même d'événements incontrôlables. A partir de cette condition de contrôlabilité, l'existence d'un superviseur peut être déterminée en fonction du théorème suivant.

[1] Pwr(Σ) est l'ensemble des sous-ensembles de Σ

Théorème 1
[Existence d'un superviseur tel que $L(S/G) = K$ [2]]
Soit K le langage désiré tel que $K \neq \emptyset$ et $K \subseteq L(G)$, il existe un superviseur S tel que le fonctionnement en boucle fermée $L(S/G)$ soit égal à K si et seulement si :

1. K est préfixe-clos ($\overline{K} = K$) ;

2. K est contrôlable par rapport à $L(G)$ ($\overline{K}\Sigma_u \cap L(G) \subseteq \overline{K}$). ♦

Le théorème ci-dessus a été établi pour le cas où les événements générés par le procédé sont tous observables par le superviseur. Dans le cas de l'observation partielle *i.e.* seulement une fraction de l'alphabet d'événements est observable, dans ce cas il s'avère nécessaire, au delà de la contrôlabilité, d'étudier l'observabilité.

Définition 2 (Observabilité)
Soit K un langage désiré tel que $K \subseteq L(G)$. K est dit observable par rapport à $L(G)$, P et Σ_{loc} (Σ_{loc} est l'ensemble d'événements observables par le superviseur S) si et seulement si :

1. Pour toutes séquences d'événements $s \in \overline{K}$, $s' \in \overline{K}$ et pour tout événement $\sigma \in \Sigma_c$ tels que $P(s) = P(s')$, $s\sigma \in \overline{K}$ et $s'\sigma \in L(G)$ alors $s'\sigma \in \overline{K}$;

2. L'inverse et vrai i.e. si $s'\sigma \in \overline{K}$ et $s\sigma \in L(G)$ alors $s\sigma \in \overline{K}$. ♦

Pour en savoir davantage, le lecteur pourra se référer aux [13], [3] pour de plus amples informations concernant la théorie de contrôle par supervision.

III. Existence et unicité d'un superviseur

A. Mise en situation

Le problème posé dans cette section consiste à montrer l'unicité d'un superviseur pour les systèmes non-déterministes. Plus précisément, supposons qu'un modèle du procédé peut admettre plusieurs états initiaux (ou états de départ), existe-il un **unique** superviseur permettant d'aboutir au langage désiré quel que soit l'état initial du procédé ? D'une manière formelle, étant donné un automate $G = (Q,\, \Sigma,\, \delta,\, Q_{\text{initiaux}},\, Q)$ possédant un ensemble d'états initiaux (Q_{initiaux}), existe-il un unique superviseur S, tel que $\forall\, q \in Q_{\text{initiaux}}$, le fonctionnement en boucle fermée soit le langage désiré $\overline{K_q}$, ($L(S/G) = \overline{K_q}$?), avec $\overline{K_q}$ langage désiré élaboré à partir de l'état initial q.

Définition 3
Soit G un modèle automate comportant plusieurs états initiaux. Pour chaque état initial q de G une dynamique particulière sera donnée par le langage $L(G_q)^2$, avec $G_q = (Q,\, \Sigma,\, \delta,\, q,\, Q_m)$ l'automate caractérisé par un seul état de départ q. ♦

⌜ *Exemple*

Soit un automate G possédant deux états initiaux q_1 et q_2 ($Q_{initiaux} = \{q_1, q_2\}$). Les langages $L(G_{q_1}) = \{s \in \Sigma^* \mid \delta(q_1, s)!\}$ et $L(G_{q_2}) = \{s \in \Sigma^* \mid \delta(q_2, s)!\}$ représentent deux dynamiques différentes du procédé G. $L(G_{q_1})$ est la

[2] $L(G_q)$ est l'ensemble de toutes les séquences d'événements partant depuis l'état q et menant vers un état du procédé G

dynamique associée à l'état q_1, $L(G_{q_2})$ est la dynamique donnée par l'état q_2. ⌟

Le problème évoqué de synthèse de contrôleur consiste à coupler le procédé G à un unique superviseur garantissant les bonnes performances quelle que soit sa dynamique courante (figure 1). Il s'agit de définir pour chaque état initial q du procédé G le langage désiré K_q. Nous devrons ensuite vérifier les conditions nécessaires et suffisantes d'existence et d'unicité d'un tel superviseur [10] .

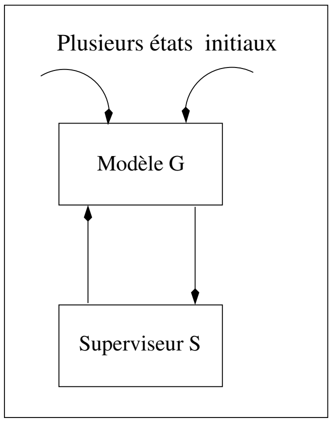

Fig. 1. Le problème de synthèse de contrôleur

B. *Théorème*

Soient $\overline{K}$ le langage désiré associé au modèle G et $\{\overline{K_q} \mid q \in Q_{\text{initiaux}}\}$ l'ensemble des langages désirés correspondant à l'ensemble d'états initiaux $q \in Q_{\text{initiaux}}$ du procédé G.

Rappelons que dans l'approche classique de la théorie de contrôle par supervision, le concept de contrôlabilité est suffisant pour démontrer l'existence d'un superviseur S qui agit sur un procédé G admettant un seul état initial [1]. Cependant, le concept de contrôlabilité s'avère insuffisant pour prouver l'unicité de S (quelque soit l'état de départ q du procédé G, $L(S/G) = \overline{K_q}$). En effet, puisque le procédé G comporte un ensemble d'états initiaux, le superviseur S est incapable d'identifier l'état à partir duquel G_i serait initialisé.

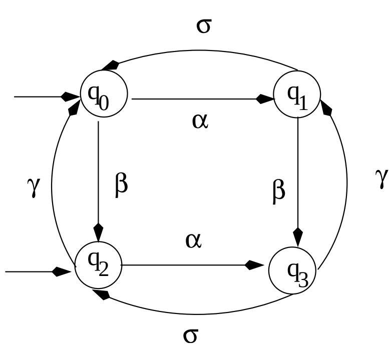

Fig. 2. Automate G possédant deux états initiaux

Pour mieux comprendre ce problème, considérons l'automate de la figure 2. Cet automate admet deux états initiaux q_0 et q_2. Supposons que les occurrences de l'événement α à partir d'états q_0 et q_2 mènent vers deux états différents, respectivement q_1 et q_3. Dans ce cas le superviseur est incapable de déterminer l'état d'arrivée après occurrence de α. De ce fait si l'événement σ est autorisé à partir de q_1 ($\alpha\sigma \in K_{q_0}$) alors l'événement σ doit être également autorisé à partir de l'état q_3. En revanche si depuis les deux états q_0 et q_2 les événements de départ sont différents, alors dans ce cas la synthèse s'effectue d'une manière classique. Le théorème suivant établit les conditions nécessaires et suffisantes que nous recherchons.

Théorème 2
Soient :

- *G un automate possédant un ensemble d'états initiaux $q \in Q_{initiaux}$,*
- *$\{\overline{K_q} \mid q \in Q_{initiaux}\}$ l'ensemble des langages désirés correspondant à chaque état initial q.* ♦

Il existe un unique superviseur S tel que $\forall q \in Q_{initiaux}$, $L(S/G) = \overline{K_q}$ si et seulement si :

C_1. *$\forall q \in Q_{initiaux}$, $\overline{K_q}$ est contrôlable par rapport à $L(G_q)$*

C_2. *$\forall s \in \overline{K_q}$ et $\forall s' \in \overline{K_{q'}}$ tels que $s = s'$, $\forall \sigma \in \Sigma_c$[3] :*
$$s\sigma \in \overline{K_q} \text{ et } s'\sigma \in L(G_{q'}) \Leftrightarrow s'\sigma \in \overline{K_{q'}}$$

Intuitivement, la condition 2 relève du fait que le superviseur S ne dispose d'aucune information pour connaître *a priori* parmi l'ensemble d'états initiaux, l'état à partir duquel le modèle G serait activé. Ainsi, s'il existe depuis deux états initiaux q_1 et q_2 un événement σ autorisé depuis l'état q_1, il devra également être autorisé depuis l'état q_2.

Théorème 3 (d'après [5] [4])
Il existe un unique superviseur S, tel que $L(S/G_{et}) = \overline{K}$ si et seulement si :

C_1'. *$\overline{K}$ est contrôlable par rapport à $L(G_{et})$;*

C_2'. *$\overline{K}$ est observable par rapport à $L(G_{et})$ et P.* ♦

Le théorème 3 a été démontré dans les références [5] [4] et donc qu'en démontrant ces équivalences, le théorème 2 est démontré.

C. *Démonstration*

Les différentes étapes de notre démonstration sont :

1. Lever le non-déterministe, par étendre le procédé G par un état particulier, appelé état inactif et ensuite par des transitions de connexion partant de l'état inactif et menant vers un état initial ;

2. Déterminer le langage désiré $\overline{K}$ du procédé G_{et} en se basant sur les langages désirés élaborés à partir de chacune d'états initiaux considérés ;

3. Appliquer le théorème classique d'existence d'un superviseur S_{et} qui est appliqué à G_{et} et K ;

4. Se ramener à S qui n'observe pas les événements de connexion. Nous montrerons alors que notre problème peut se ramener à l'étude de l'existence d'un superviseur avec observation partielle ;

La vérification de l'équivallence des premières conditions

[3] Σ_c est l'ensemble des événements contrôlables du modèle G

des théorèmes 2 et 3 revient à assimiler le comportement en boucle fermée de l'automate non-déterministe à celui de l'automate étendu. L'équivalence des secondes conditions repose sur la notion de l'observabilité.

5. Démontrer l'équivalence entre la condition C_1 du théorème. 2 et C'_1 du théorème 3 ;

6. Démontrer l'équivalence entre la condition C_2 du théorème 2 et C'_2 du théorème 3.

C.1 Étape 1 : lever le non-déterministe par l'extension du procédé G

Soient $G = (Q, \Sigma, \delta, Q_{\text{initiaux}}, Q_m)$ et $T = \{\tau_1, \tau_2, ..., \tau_n\}$ l'ensemble d'événements de connexion qui permettent d'atteindre l'ensemble d'états initiaux. La définition de l'ensemble d'événements de connexion T se base sur celle de l'ensemble d'états initiaux. Il y a autant d'événements de connexion que d'états initiaux.

Le modèle G peut être étendu de la façon suivante : $G_{et} = (Q_{et}, \Sigma_{et}, \delta_{et}, q_{in}, Q_{m,et})$ avec :

- $Q_{et} = Q \cup \{q_{in}\}$;
- $\Sigma_{et} = \Sigma \cup T$;
- $Q_{m,et} = Q_m$;
- la fonction de transition est définie comme suit :
  1. $\forall \sigma \in \Sigma$ et $\forall q \in Q$ si $\delta(q, \sigma)!$ alors $\delta_{et}(q, \sigma) := \delta(q, \sigma)$;
  2. $\forall \tau_i \in T$ et $\forall q_i \in Q_{\text{initiaux}}$, $\delta_{et}(q_{in}, \tau_i) := q_i$.

⌜ *Exemple*
Soit le modèle automate G de la figure 3.

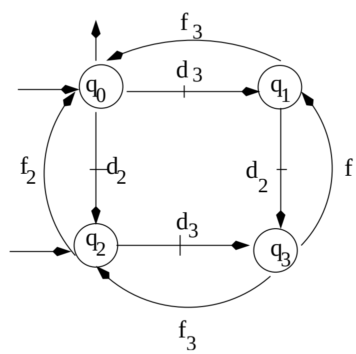

Fig. 3. Modèle automate non étendu G

D'après la figure 3, le modèle G admet deux états initiaux q_0 et q_2, nous définissons par conséquent l'ensemble d'événements de connexion $T = \{\tau_0, \tau_2\}$. Le modèle automate étendu est représenté par la figure 4. ⌟

L'automate étendu possède un seul état initial q_{in} et par conséquent une seule dynamique $L(G_{et})$. Cela permet d'appliquer aisément les méthodes classiques de la théorie de contrôle par supervision.

C.2 Étape 2 : détermination du langage désiré $\overline{K}$

Afin d'appliquer ces méthodes, nous devons tout d'abord définir le langage désiré correspondant au modèle étendu.

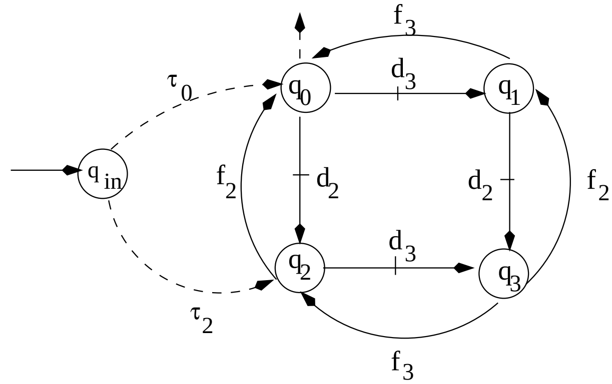

Fig. 4. Modèle automate étendu G_{et}

De ce fait, nous allons étendre également les langages désirés $\overline{K_q}$ de chaque état initial du procédé G par l'ajout de l'événement de commutation τ_q correspondant. D'une manière formelle $\forall q \in Q_{\text{initiaux}}$, le langage étendu de $\overline{K_q}$ est $\tau_q \overline{K_q}$, avec :

$$\tau_q \overline{K_q} := \{s \mid s = \tau_q s' \text{ où } s' \in \overline{K_q}\}$$

Par conséquent, le langage désiré global du modèle G_{et} est donné par :

$$\overline{\cup_{q \in Q_{\text{initiaux}}} \tau_q K_q}$$

Par simplification, nous notons désormais $\overline{\cup_{q \in Q_{\text{initiaux}}} \tau_q K_q}$ par $\overline{K}$.

C.3 Étape 3 : existence d'un superviseur unique S_{et}

Puisque l'automate G_{et} admet un seul état de départ (état initial), alors d'après [2] il existe un seul superviseur noté S_{et}, tel que $\forall q \in Q_{\text{initiaux}}$, $L(S_{et}/G_{et}) = \overline{K}$ *si et seulement si* $\overline{K}$ est contrôlable par rapport à $L(G_{et})$.

C.4 Observation partielle

Puisque l'ensemble d'événements T est inaccessible au superviseur S, nous allons définir la fonction de projection P qui permet d'observer seulement l'occurrence d'événements de l'alphabet Σ : tous les événements de T seront considérés non observables. Cette fonction est donnée comme suit :

$$\text{Soit} \quad P : \Sigma_{et}^* \longrightarrow \Sigma^* \quad \text{telle que} \quad \forall \sigma \in \Sigma \text{ et } \forall s \in \Sigma^* :$$
$$\begin{array}{lcl} P(\varepsilon) & = & \varepsilon \\ P(s\sigma) & = & \begin{cases} P(s)\sigma & \text{si} \quad \sigma \in \Sigma \\ P(s) & \text{si} \quad \sigma \in T \end{cases} \end{array}$$

Soit $S : P(\Sigma_{et}^*) \longrightarrow \Gamma := \{\gamma \in Pwr(\Sigma_{et}) : \Sigma_{et,u} \subseteq \gamma\}$[4] tel que $\forall s \in \Sigma_{et}^*$, $S(P(s)) = S_{et}(s)$. La figure 5 montre le fonctionnement en boucle fermée du superviseur S avec le procédé étendu G_{et}.

D'après la figure 5, nous constatons que l'étude de l'existence de S nous ramène au cas de l'étude de l'existence d'un superviseur observant partiellement un procédé. Les conditions d'existence de ce superviseur sont définies dans le théorème 3.
Le théorème 2 est donc vrai si les différentes conditions des deux théorèmes sont équivalentes.

[4] $\Sigma_{et,u}$ représentant l'ensemble des événements incontrôlables

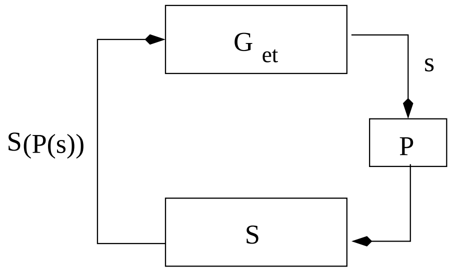

Fig. 5. G_{et} couplé avec le superviseur S via la fonction de projection P

C.5 Étape 5 : équivalence des conditions C_1' et C_1

Cette équivalence qui correspond au lemme 3, nécessite tout d'abord d'introduire deux autres équivalences, qui sont les lemmes 1 et 2.

Lemme 1

$$\overline{\cup_{q\in Q_{initiaux}} K_{q,et}} = \cup_{q\in Q_{initiaux}} \overline{K_{q,et}}$$

♦

Démonstration

Pour démonter l'égalité, nous allons démontrer deux inclusions :

1. $\overline{\cup_{q\in Q_{initiaux}} K_{q,et}} \subseteq \cup_{q\in Q_{initiaux}} \overline{K_{q,et}}$;
2. $\cup_{q\in Q_{initiaux}} \overline{K_{q,et}} \subseteq \overline{\cup_{q\in Q_{initiaux}} K_{q,et}}$.

1. $\forall q \in Q_{initiaux}$, $\overline{K_{q,et}} \subseteq \overline{\cup_{q\in Q_{initiaux}} K_{q,et}}$
$\Rightarrow \cup_{q\in Q_{initiaux}} \overline{K_{q,et}} \subseteq \overline{\cup_{q\in Q_{initiaux}} K_{q,et}}$;

2. $s \in \cup_{q\in Q_{initiaux}} \overline{K_{q,et}}$
$\Rightarrow \exists\, q \in Q_{initiaux}$ tel que $s \in \overline{K_{q,et}}$
$\Rightarrow \exists\, u \in \Sigma_{et}^*$ tel que $su \in K_{q,et}$
$\Rightarrow su \in \cup_{q\in Q_{initiaux}} K_{q,et}$
$\Rightarrow s \in \overline{\cup_{q\in Q_{initiaux}} K_{q,et}}$.

Les deux implications sont vraies donc l'égalité est vérifiée. ◇

Lemme 2

$$\forall q \in Q_{initiaux},\ \tau_q \overline{K_q} = \overline{\tau_q K_q}$$

♦

Démonstration

D'une façon similaire, nous allons montrer que $\forall q \in Q_{\text{initiaux}}$:

1. $\tau_q \overline{K_q} \subseteq \overline{\tau_q K_q}$
2. $\overline{\tau_q K_q} \subseteq \tau_q \overline{K_q}$

1. $s \in \tau_q \overline{K_q}$
$\Rightarrow s = \tau_q s_1$ avec $s_1 \in \overline{K_q}$
$\Rightarrow \exists u \in \Sigma_{et}^*$ tel que $s_1 u \in K_q$
$\Rightarrow \tau_q s_1 u \in \tau_q K_q$
$\Rightarrow s = \tau_q s_1 \in \overline{\tau_q K_q}$

2. $s \in \overline{\tau_q K_q}$
$\Rightarrow u \in \Sigma_{et}^*$ tel que $su \in \tau_q K_q$
$\Rightarrow su = \tau_q s_1 u \in \tau_q K_q$ avec $s_1 u \in K_q$
$\Rightarrow s_1 \in \overline{K_q}$
$\Rightarrow \tau_q s_1 \in \tau_q \overline{K_q}$
$\Rightarrow s = \tau_q s_1 \in \tau_q \overline{K_q}$

Les deux implications sont vraies donc l'égalité est vérifiée. ◇

Lemme 3

$\overline{K}$ est contrôlable par rapport à $L(G_{et})$ si et seulement si $\forall q \in Q_{initiaux}$, $\overline{K_q}$ est contrôlable par rapport à $L(G_q)$. ♦

Démonstration

Pour démontrer l'équivalence, nous allons démontrer deux implications :

1. $\overline{K}$ contrôlable par rapport à $L(G_{et})$
$\Rightarrow \forall q \in Q_{\text{initiaux}}$, $\overline{K_q}$ est contrôlable par rapport à $L(G_q)$
2. $\forall q \in Q_{\text{initiaux}}$, $\overline{K_q}$ contrôlable par rapport à $L(G_q)$
$\Rightarrow \overline{K}$ est contrôlable par rapport à $L(G_{et})$

1. $s \in \overline{K_q} \Rightarrow \tau_q s \in \tau_q \overline{K_q}$
$\Rightarrow \tau_q s \in \overline{\tau_q K_q}$ (lemme 2)
$\Rightarrow \tau_q s \in \cup_{q\in Q_{\text{initiaux}}} \overline{\tau_q K_q}$
$\Rightarrow s \in \overline{K}$ (lemme 2)

Soit $\sigma \in \Sigma_{et,u}$ tel que $\tau_q s\sigma \in L(G_{et})$.
$\overline{K}$ contrôlable par rapport à $L(G_{et})$
$\Rightarrow \tau_q s\sigma \in \overline{K}$
$\Rightarrow \tau_q s\sigma \in \cup_{q\in Q_{\text{initiaux}}} \overline{\tau_q K_q}$ (lemme 1)
$\Rightarrow \exists\, q \in Q_{\text{initiaux}}$ tel que $s\sigma \in \overline{K_q}$;

2. $\forall q \in Q_{\text{initiaux}}$, $\overline{K_q}$ contrôlable par rapport à $L(G_q)$
$\Rightarrow \tau_q \overline{K_q} = \overline{\tau_q K_q}$ (lemme 2) est contrôlable par rapport à $L(G_{et})$ puisque $\forall q \in Q_{\text{initiaux}}$, τ_q est supposé un événement
autorisé par défaut
$\Rightarrow \overline{\cup_{q\in Q_{\text{initiaux}}} \tau_q K_q}$ est contrôlable par rapport $L(G_{et})$

◇

C.6 Étape 6 : équivalence des conditions C_2' et C_2

Lemme 4

K est observable par rapport à $L(G_{et})$ et P si et seulement si :

1. $\forall\, s \in \overline{K_q}$ et $\forall s' \in \overline{K_{q'}}$ tels que $s = s'$, $\forall \sigma \in \Sigma_c$[5] :
$s\sigma \in \overline{K_q}$ et $s'\sigma \in L(G_{q'}) \Leftrightarrow s'\sigma \in \overline{K_{q'}}$

Démonstration

Nous allons maintenant établir une relation d'équivalence entre la condition C_2 du théorème 2 et l'observabilité de $\overline{K}$. Notons que $\overline{K}$ est observable par rapport à $L(G_{et})$ et P, ***si et seulement si*** :

1. $(\forall s, s' \in \Sigma_{et}^*)(P(s) = P(s'))(\forall \sigma \in \Sigma_{et,c}) \Rightarrow s \in \overline{K}$, $s' \in \overline{K}$ et $s\sigma \wedge s' \in \overline{K}$ et $s'\sigma \in L(G_{et})$, alors $s'\sigma \in \overline{K}$;

2. l'inverse est vrai.

Supposons que $\overline{K}$ est observable par rapport à $L(G_{et})$ et P.

Soit $s \in \overline{K_q}$ et $s' \in \overline{K_{q'}}$ tel que $s = s'$.

[5] Σ_c est l'ensemble des événements contrôlables du modèle G

Si $s\sigma \in \overline{K_q}$ et $s'\sigma \in L(G_{q'})$, montrons que $s'\sigma \in \overline{K_{q'}}$.

D'une part	$s \in \overline{K_q}$
	$\Rightarrow \tau_q s \in \tau_q \overline{K_q} = \overline{\tau_q K_q}$(lemme 2)
	$\Rightarrow \tau_q s \in \overline{\cup_{q \in Q_{initiaux}} \tau_q K_q} = \overline{K}$
D'autre part	$s' \in \overline{K_{q'}}$
	$\Rightarrow \tau_{q'} s \in \overline{\cup_{q \in Q_{initiaux}} \tau_q K_q}$
Puisque	$s = s'$ alors $P(\tau_q s) = P(\tau_{q'} s')$
Or	$s\sigma \in \overline{K_q}$
	$\Rightarrow \tau_q s\sigma \in \overline{\cup_{q \in Q_{initiaux}} \tau_q K_q} = \overline{K}$
	$\Rightarrow \tau_{q'} s'\sigma \in \overline{\cup_{q \in Q_{initiaux}} \tau_q K_q} = \overline{K}$
	(puisque $\overline{K}$ est observable)
	$\Rightarrow \tau_{q'} s'\sigma \in \cup_{q \in Q_{initiaux}} \overline{\tau_q K_q}$(lemme 1)

Il en résulte que :

$\tau_{q'} s'\sigma \in \overline{\tau_{q'} K_{q'}} = \tau_{q'} \overline{K_{q'}} \Rightarrow s'\sigma \in \overline{K_{q'}}$ (condition C_2 du théorème 2).

Maintenant supposons la condition C_2 vérifiée et montrons l'observabilité de $\overline{K}$.

Soit $s \in \overline{K}$ et $s' \in \overline{K}$ tels que $\forall \sigma \in \Sigma_{et,c}$, $s\sigma \in \overline{K}$.

Montrons que si $s'\sigma \in L(G_{et})$ alors $s'\sigma \in \overline{K}$.

D'une part	$s\sigma \in \overline{K} = \overline{\cup_{q \in Q_{initiaux}} \tau_q K_q}$
	$\Rightarrow \exists u \in \Sigma_{et}^*$ tel que $s\sigma u \in \cup_{q \in Q_{initiaux}} \tau_q K_q$
	$\Rightarrow \exists q \in Q_{initiaux}$ tel que $s\sigma u \in \tau_q K_q$
	$\Rightarrow s\sigma u = \tau_q s_1 \sigma u \in \tau_q K_q$ (avec $s = \tau_q s_1$)
	$\Rightarrow s_1 \sigma u \in K_q \Rightarrow s_1 \sigma \in \overline{K_q}$
De même	$s' \in \overline{K} = \overline{\cup_{q \in Q_{initiaux}} \tau_q K_q}$
	$\Rightarrow \exists v \in \Sigma_{et}^*$ tel que $s'v \in \cup_{q \in Q_{initiaux}} \tau_q K_q$
	$\Rightarrow \exists q' \in Q_{initiaux}$ tel que $s'v \in \tau_{q'} K_{q'}$
	$\Rightarrow s'v = \tau_{q'} s_2 v \in \tau_{q'} K_{q'}$ (avec $s' = \tau_{q'} s_2$)
	$\Rightarrow s_2 v \in K_{q'} \Rightarrow s_2 \in \overline{K_{q'}}$

On a $s_1 = s_2$ puisque $P(s) = P(s')$. Par conséquent, et selon la condition C_2 du théorème 2, $s_2\sigma \in \overline{K_{q'}}$. Donc $\tau_{q'} s_2 \sigma = s'\sigma \in \overline{\tau_{q'} K_{q'}}$ *.i.e.* $s'\sigma \in \overline{K}$. ◇

Les lemmes 1, 2, 3 et 4 nous ont donc permis de démontrer l'équivalence entre les conditions des théorèmes 2 et 3, et donc de justifier notre proposition.

IV. Conclusion

Cet article nous a permis de présenter l'extension de la théorie de contrôle par supervision par l'étude de la commande des systèmes non-déterministes. Nous avons considéré le cas où les modèles automates décrivant ces systèmes possèdent un ensemble d'états initiaux. Pour chaque état initial, un langage désiré a été élaboré et nous avons démontré qu'il existe un unique superviseur permettant de garantir le langage désiré, bien que le procédé ait plusieurs états initiaux. Nous avons levé le non-déterministe par étendre le procédé d'une part par l'ajout d'un état particulier, appelé état inactif et d'autre part par l'ajout des transitions de connexion partant de l'état inactif et menant vers un état initial. Ensuite nous avons déterminé le langage désiré global (ce langage a été défini en se basant sur l'ensemble des langages désirés élaborés pour chaque état initial) du procédé étendu et par suite nous avons appliqué le théorème classique d'existence d'un superviseur sur le procédé étendu et le langage désiré global, après on se ramene à un superviseur qui n'observe pas les événements de connexion. Nous avons montré alors que notre problème peut se ramener à l'étude d'existence d'un superviseur sous observation partielle.

Références

[1] P. Ramadge, W. Wonham. « The control of Discrete Event Systems, »In Proc IEEE, vol. 77, p. 81-98. 1989.

[2] P. Ramadge, W. Wonham. « Supervisory control of a classe of discrete event processes, »SIAM, Control Optimisation, vol. 25, n. 1, p. 206-230, 1987.

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

[4] T. -S. Yoo, S. Lafortune. « Decentralized supervisory control : a new architecture with a dynamic decision fusion rule, » 6th international Workshop on Discrete Event Systems, Saragosse, Espagne, p. 11-17, 2-4 octobre 2002.

[5] F. Lin. « Decentralized control and coordination of discrete event systems with partial observation, » IEEE Transactions on Automatic Control, vol. 35, p. 1330-1337, 1990.

[6] F. Lin. « Supervisory control of timed discrete event systems under partial observation, » IEEE Transactions on Automatic Control, vol. 40, p. 558-562, 1990.

[7] K. Rudie, W. M. Wonham. « Think globally, act locally : Decentralized supervisory control, »IEEE Transactions on Automatic Control, vol. 37, n. 11, p. 1692-1708, November 1992.

[8] 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, p. 131-186, 2000.

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

[10] O. Kamach. « Approche multi-modèle pour les systèmes à événements discrets : application à la gestion des modes de fonctionnement, »Thèse de doctorat à l'Institut National des Sciences Appliquées de Lyon, 2004.

[11] S.L. Chung, S. Lafortune et F. Lin. « Limited lookahead policies in supervisory control of discrete event systems, »IEEE Transactions on Automatic Control, vol. 37, p. 1921-1935, 1992.

[12] R. Cieslak, C. Desclaux, A. Fawaz et P. Varaiya. « Supervisory control of discrete event processes with partial observation, »IEEE Transactions on Automatic Control, vol. 33, p. 246-260, 1988.

[13] C. G. Cassandras et S. Lafortune. « Introduction to Discrete Event Systems, »Kluwer Academic Publishers, 1999.