

Contributions à la synthèse de commande des systèmes à évènements discrets: nouvelle modélisation des états interdits et application à un atelier flexible

Maen Atli

▶ **To cite this version:**

Maen Atli. Contributions à la synthèse de commande des systèmes à évènements discrets: nouvelle modélisation des états interdits et application à un atelier flexible. Autre. Université de Lorraine, 2012. Français. NNT : 2012LORR0412 . tel-01750418

HAL Id: tel-01750418
https://hal.univ-lorraine.fr/tel-01750418

Submitted on 29 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4
Code de la Propriété Intellectuelle. articles L 335.2- L 335.10
http://www.cfcopies.com/V2/leg/leg_droi.php
http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm

Contributions à la synthèse de commande des systèmes à événements discrets : nouvelle modélisation des états interdits et application à un atelier flexible

Thèse

Soutenue le 27 septembre 2012

Pour l'obtention du

Doctorat de l'Université de Lorraine

(Spécialité : Automatique, traitement du signal et génie informatique)

Par

Maen ATLI

Composition du jury

Rapporteurs :

M. ALLA Hassane, Professeur des Universités, Universités à Université Joseph Fourier - Grenoble

M. CHU Chengbin, Professeur des Universités, l'école centrale de Paris

Examinateurs :

M. CHATELET Eric, Professeur des Universités, Université de Technologie de Troyes

M. PETIN Jean-François, Professeur des Universités, l'école nationale supérieure d'électricité et de mécanique - Nancy

M. SAVA Alexandre, Maître de conférences HDR, l'école nationale d'ingénieurs de Metz

M. ACHOUR Zied, Maître de conférences, Membre Invité, l'université de Lorraine

Directeur de thèse :

M. ADJALLAH Kondo H., Professeur des Universités, l'école nationale d'ingénieurs de Metz

A contribution to control synthesis of Discrete Event systems: New model of forbidden states (applied on a flexible workshop)

A manufacturing system may be represented by *Discrete Event System* (DES). Apart from planning (where people work with ratios of products fabricated per week or per day), any modelling could be based on the concepts of event and activities. An event corresponds to a state change. An activity is a black-box summarizing what is occurring between two events. When using Petri Nets, events are associated with *transitions*, and activities with *places*. Our work proposes a supervisory synthesis for *Discrete Event System* modelled by a class of Petri Net called *Marked Graph*. The objective of this synthesis is to build a control law that enforces the system to respect a set of given specifications. To model these specifications, we propose new mathematical formulas called *Marking Exclusion Constraint* (*MEC*). This model is our first contribution. The Second main contribution of my thesis is to synthesize a computationally efficient technique to build a supervisor that enforces the system to respect the constraints by avoiding a set of forbidden states modelled by *MEC* specifications. We extend this synthesis technique to solve the problem in the presence of uncontrollable events and unobservable events. Sometimes in order to study the performance aspects, we must take in consideration the time data. Thus we address control synthesis for *Timed Discrete Event Systems* under MEC specifications by using *Timed Petri Nets*.

Keywords: Discrete event systems, Petri Net, Supervisory synthesis, Specifications, Marking Exclusion Constraints, Flexible workshop.

Contributions à la synthèse de commande des systèmes à événements discrets: nouvelle modélisation des états interdits et application à un atelier flexible

Un Système de production peut être représenté par les systèmes à événements discrets. En dehors de la planification (où les gens travaillent avec des ratios de produits fabriqués par semaine ou par jour), la modélisation pourrait être basée sur les concepts d'événement et d'activités. Un événement correspond à un changement d'état. Une activité est une boîte noire d'encapsulation de ce qui se passe entre deux événements. En utilisant les réseaux de Petri (RdP), les événements sont représentés par les transitions, et les activités par les lieux. Notre travail propose une synthèse de commande par supervision pour les systèmes d'événements discrets modélisés par une classe de réseaux de Petri appelé graphe d'événements. L'objective de cette thèse est de concevoir un superviseur capable d'aider à améliorer la performance de système et de protéger le système en respectant des spécifications données par le fabricant ou le client selon les besoins et les conditions de travail. Pour modéliser ces spécifications, nous avons proposé un nouveau modèle mathématique de contrainte, appelé *Contrainte d'Exclusion de Marquage* (CEM). La deuxième contribution principale de ma thèse est de synthétiser une technique efficace et simple pour construire un superviseur qui impose le système de respecter des contraintes en évitant l'ensemble des états interdits modélisé par CEM. Nous avons également développé cette synthèse pour résoudre le problème d'existence des événements incontrôlables et des événements inobservables. Parfois, afin d'étudier les aspects liés à la performance, nous devons prendre le temps en considération. Donc, nous avons résolu aussi le problème des événements temporisé en utilisant RdP temporisés soumis à CEM.

Mots clés : Systèmes à événements discrets, Réseaux de Petri, Synthèses de commande, spécifications, Contrainte d'Exclusion de Marquages, Atelier flexible.

A la mémoire de mon père

A la mémoire des martyrs syriens

A la mémoire de mon ami Hassane Ismael

Je tiens à exprimer mes remerciements les plus sincères à M. Kondo H. ADJALLH, Professeur à l'ecole nationale d'ingénieurs de Metz, qui m'a suivi et guidé tout au long de ce travail. Je tiens à lui exprimer ma gratitude et mes remerciements les plus sincères pour ses conseils et le temps qu'il a consacré à diriger cette thèse.

J'exprime aussi toute ma reconnaissance à M. Alexandre SAVA, Maître de conférences HDR à l'ecole nationale d'ingénieurs de Metz pour m'encadrer et pour son soutien tout le long de de thèse.

Je suis également très reconnaissant envers M. Zied ACHOUR, Maître de conférences à l'université de Lorraine, qui m'a encadré aussi dans cette thèse, pour sa disponibilité, sa patience et ses précieux conseils.

Ma gratitude et mes remerciements vont ensuite aux membres du jury :

M. ALLA Hassane, Professeur des Universités à Université Joseph Fourier - Grenoble

M. CHU Chengbin, Professeur des Universités à l'école centrale Paris

M. CHATELET Eric, Professeur des Universités à *Université de Technologie de Troyes*

M. PETIN Jean-François, Professeur des Universités à l'école nationale supérieure d'électricité et de mécanique – Nancy

qui m'ont honoré en apportant leur expérience pour l'évaluation de cette thèse et en acceptant de participer au jury.

Je remercie vivement tous les membres du laboratoire LGIPM et en particulier Zhouhang WANG, Akram ZOUGGARI, Houssam CHOUIKHI, Abede Rahmane BENSMAINE, Hamza BOUDHAR et M. Mohammed Dahane, pour les encouragements et le soutien reçu durant cette thèse.

Je n'oublierai pas à adresser les remerciements les plus distingués à mes amis de toujours en Syrie, Hassane Ibrahim, Ahmad Lolo, Houssame Maaz, Ahmad Hazouri, Abed El-Rahmane Jebara, Abed El-Rahmane Hardan, Zaki Dakhel, Hassane El Jassem, Ahamed Kassim, Taher Bido, Abed Allah El Zaeem, Amjad Hijab.

Mes remerciements et ma gratitude vont à ma chère mère, mes sœurs et mes frères qui ont toujours eu confiance en moi, pour m'avoir encouragé et soutenu durant tous les moments difficiles qui se passent dans mon pays, La Syrie.

Enfin, un remerciement très particulier à celle qui a tout quitté pour me rejoindre en France et qui m'a donné le courage pour surmonter les moments difficiles durant cette thèse, pour son amour, son soutien et sa confiance.

Table des matières

Introduction générale 1

Chapitre 1 5

1.1 Introduction 6

1.2 Modélisation des Système à événements discrets 9

1.2.1 Langage formel et les automates à états finis 11
1.2.2 Les automates 13
1.2.3 Les réseaux de Petri 17
1.2.4 Graphe d'événements 24

1.3 Conclusion : 24

2 *Chapitre 2* 27

2.1 Introduction 28

2.2 Spécifications 28

2.2.1 Problème d'états interdits (PEI) 29
2.2.2 Problème de transition d'états interdits (PTEI) 29

2.3 Modèles de Contraintes 30

2.3.1 Contraintes linéaires 30
2.3.2 Contraintes de type prédicat 32

2.4 SdC par supervision 34

2.4.1 Synthèse de superviseur basée sur langage formel et automate 35
2.4.2 Synthèse de superviseur basée sur les réseaux de Petri 39
2.4.3 Approches de SdC et existence de transitions inobservables 55

2.5 Conclusion 61

3 *Chapitre 3* 63

3.1 Introduction 64

3.2 Problème d'état interdit 64

3.3 Contrainte d'Exclusion de Marquage et ses extensions 65

3.3.1 Contrainte d'Exclusion de Marquage CEM 65
3.3.2 Contrainte d'Exclusion de Marquage de type Ou CEM-Ou 68
3.3.3 Contrainte d'Exclusion de Marquage de type Et CEM-Et 69
3.3.4 Contrainte d'Exclusion de Marquage Mutuelle CEM-M 71
3.3.5 Classe de contraintes de CEM C-CEM 72

3.4 Puissance de modélisation de la Contrainte d'Exclusion de Marquage 72

3.4.1 CEM versus contraintes de type prédicat 72
3.4.2 CEM versus les contraintes linéaires 73
3.4.3 CEM versus CGEM 74

3.5 Déterminer la classe minimale 74

3.6 Exemple d'un réseau ferroviaire 77

3.7 Conclusion 80

4 Chapitre 4 83

4.1 Introduction 84

4.2 Notions de base pour le Graphe d'événements contrôlable 84

4.2.1 Graphe d'événements contrôlable 85

4.2.2 Superviseur optimal 85

4.3 Loi de commande des graphes d'événements totalement contrôlable 86

4.3.1 Système soumis à des spécifications de type CEM 86

4.3.2 Système soumis à des spécifications de type CEM-Ou 89

4.3.3 Système soumis à des spécifications de type CEM-Et 91

4.3.4 Système soumis à des spécifications de type CEM-M 93

4.4 Notions de base pour les graphes d'événements partiellement contrôlables 97

4.4.1 Ensemble de marquages dangereux 97

4.4.2 Superviseur optimal pour les graphes d'événements partiellement contrôlables 98

4.4.3 Notions de base 98

4.5 Loi de contrôle des graphes d'événements partiellement contrôlable ... 101

4.5.1 Système soumis à des spécifications de type CEM 102

4.5.2 Système soumis à des spécifications de type CEM-Ou 105

4.5.3 Système soumis à des spécifications de type CEM-Et 107

4.5.4 Système soumis à des spécifications de type CEM-M 109

4.6 Algorithme de construction du superviseur 113

4.7 Exemple illustratif 114

4.8 Conclusion 117

5 *Chapitre 5* 119

5.1 Introduction 120

5.2 SdC des graphes d'événements partiellement observable 120

5.2.1 Système soumis à des spécifications de type CEM 121

5.2.2 Système soumis à des spécifications de type CEM-Ou 127

5.2.3 Système soumis à des spécifications de type CEM-Et 128

5.2.4 Système soumis à des spécifications de type CEM-M 128

5.2.5 La loi de commande dans le cas général 129

5.3 Exemple illustratif 130

5.4 Conclusion 131

6 *Chapitre 6* 133

6.1 Introduction 134

6.2 Graphes d'événements P-temporisés 135

6.3 SdC de graphes d'événements P-temporisés soumis à CEM 135

6.3.1 Analyse de la loi de commande à l'état initial 136

6.3.2 Analyse de la loi de commande en franchissant des transitions d'influence 140

6.4 SdC pour des graphes d'événements P-temporisés soumis à des CEM-Ou/CEM-Et 146

6.5 Exemple illustratif 148

6.6 Conclusion 151
***7 Chapitre 7*..... 153**
7.1 Introduction..... 154
7.2 Modèle logique 158
7.3 Les spécifications..... 161
7.4 Le superviseur 162
7.5 Conclusion 171
***8 Conclusion* 172**
***9 Références*..... 174**

Introduction générale

La complexité des systèmes de production rend indispensable le développement des techniques appropriées pour garantir le respect des spécifications tels que les contraintes techniques et la sûreté de fonctionnement du système de production. La commande d'un système de production comprend deux aspects : le premier aspect concerne la partie matérielle (capteurs et actionneurs) et le deuxième concerne la partie synthèse des commandes qui utilise les informations captées par les capteurs pour élaborer des ordres envoyés aux actionneurs suivant des objectifs précis. Le développement de cette synthèse a deux avantages : le premier est l'amélioration de la performance de système en simplifiant l'environnement d'intégration des deux aspects, et le deuxième est la réduction des composants (capteurs et actionneurs) constitutifs du système et en conséquence les coûts de leur commande. Des fois l'expérience de l'ingénieur de conception n'est pas suffisante pour construire le superviseur qui correspond aux critères définies par le client dans le cahier de charge. Pour cela il faut synthétiser des approches pour l'aider à construire ce superviseur.

Notre étude recouvre les systèmes à événements discrets, où on s'intéresse au *Début* et à la *Fin* de chaque événement. Ces systèmes sont de plus en plus présents dans de nombreux secteurs d'activité, tels que les systèmes manufacturier, de télécommunication, aéronautique, etc. Ces systèmes nécessitent la modélisation et l'analyse des tâches complexes qui sont souvent critiques.

L'objective de cette thèse est de concevoir un superviseur capable d'améliorer la performance et la protection du système, en respectant les spécifications données par le fabricant ou le client selon les objectifs et les conditions de travail.

Avant de réaliser cet objectif, nous avons premièrement besoin d'outils pour modéliser le système (graphiquement ou mathématiquement ou les deux ensemble), tels que réseaux de files d'attente, l'algèbre Max Plus, les chaines de Markov, les automates à états finis, les réseaux de Petri, etc. Le deuxième besoin est de représenter mathématiquement les objectifs de commande (spécifications). Pour répondre à ce besoin, nous rappelons les modèles de contraintes déjà développés dans la littérature et nous proposons un nouveau modèle de base capable de représenter le problème *d'états interdits*. Finalement nous utilisons des expressions mathématiques pour déterminer la loi de contrôle qui sera le cœur du superviseur.

Le superviseur et le système à superviser forment un système supervisé où tous les deux travaillent en boucle fermée pour satisfaire un objectif désiré. Le système à superviser est appelé le procédé qui génère des événements pas forcément désirés. Le rôle de superviseur est d'assurer que ces événements soient admissibles. Ce rôle se traduit par des ordres envoyés au procédé. Les ordres n'obligent pas le procédé à générer les événements mais ils interdisent seulement l'occurrence de certains événements pour maintenir le système sur un ensemble de trajectoires admissibles.

Dans notre cas d'étude, nous nous intéressons aux systèmes de production. Ce type de système obéit à des règles opérationnelles ou à des algorithmes. Leur état évolue en réponse à l'occurrence d'*événements* discrets asynchrones dans le temps (arrivée d'un produit, arrivée d'un ordre d'exécution ou achèvement d'une tâche).

Notre travail est organisé en 7 chapitres. Nous commençons dans le premier chapitre par l'étude bibliographie des systèmes à événements discrets SEDs, leur définition, leurs utilités, leurs nécessités et les modèles mathématiques et graphiques. Nous nous intéressons particulièrement aux automates à états finis et aux réseaux de Petri.

L'objet du deuxième chapitre comprend un état de l'art correspondant à la synthèse de commande pour les SEDs. Dans ce chapitre, nous présentons dans un premier temps les principales méthodes citées dans la littérature utilisant les automates et les RdPs. D'autres approches sont aussi exposées pour résoudre le problème de l'existence des événements incontrôlables et ensuite des événements inobservables. Dans notre approche, nous utilisons le RdP pour modéliser le procédé et son superviseur en même temps.

Notre première contribution de cette thèse est présentée dans le chapitre trois. Cette contribution est un nouveau modèle de spécification appelé *Contrainte d'Exclusion de Marquage* (*CEM*) pour représenter d'une manière logique le problème d'état interdit. Dans ce chapitre, nous montrons l'intérêt de ce modèle en comparant sa puissance avec les autres modèles développés dans la littérature. Plusieurs extensions de ce modèle sont aussi définies. Finalement nous présentons des règles pour simplifier le nombre de contraintes.

La deuxième contribution est présentée dans le quatrième chapitre, où nous proposons une approche pour synthétiser une loi de commande par supervision pour les SEDs modélisés par *Graphes d'événements* soumis aux spécifications représentées par CEM. Cette technique est basée sur l'observation et le control simultanés des

mouvements des jetons entrants et sortants de chaque place. Dans un premier temps nous étudions le cas où le système est totalement contrôlable. Par la suite, nous abordons le cas des systèmes partiellement contrôlables, où il faut garantir que le superviseur décrit par cette synthèse est capable de restreindre le moins possible le comportement du procédé. Nous appellons cela *optimalité*. A la fin de ce chapitre, nous exposons un algorithme pour développer le superviseur qui travaille en boucle fermée avec le procédé dans le cas où il existe des événements incontrôlables.

Dans le cinquième chapitre, nous continuons à développer la synthèse précédente pour résoudre le problème des événements inobservables. Dans cette phase, nous distinguons pour chaque place, quatre ensemble d'événements dont l'occurrence permet la mise à jour de quatre fonctions qui génèrent la loi de contrôle. La loi de contrôle est binaire. Nous montrons que cette loi est maximale permissive. L'algorithme proposé à la fin de ce chapitre est l'algorithme général pour résoudre le problème de contrôlabilité et observabilité.

Parfois afin d'étudier les performances des modèles de réseaux de Petri, il faut intégrer la durée des activités dans le modèle. Pour modéliser ces systèmes, on utilise les réseaux de Petri temporisés. Dans le chapitre six, nous proposons une synthèse de commande pour les systèmes modélisés par un réseau de Petri où le temps est associé aux places. Par la construction des tables de contrôle au fil du temps, nous analysons la loi de contrôle et en conséquence nous générons des règles de franchissement pour concevoir le superviseur approprié.

Le dernier chapitre est un exemple illustratif d'un atelier flexible de machine d'usinage sur lequel nous appliquons nos contributions de cette thèse. Les résultats obtenus montrent l'applicabilité de notre étude.

Chapitre 1

Systèmes à Evénements Discrets SED

Résumé :

Dans ce chapitre, nous rappelons les notions de base sur les Systèmes à Evènements Discrets (SED), les définitions et les modèles existants pour les représenter. Tout d'abord nous présentons la définition du problème de SED. Ensuite nous exposons des outils existants pour la modélisation et l'analyse des SED. Nous accordons une attention particulière aux automates à états finis et aux réseaux de Petri RdP.

1.1 Introduction

Le vocable “Systèmes à événements discrets” est relativement récent et recouvre un champ de recherches encore assez diverses. Ce domaine est actuellement très actif, et fait l’objet de nombreuses manifestations scientifiques (il existe déjà un journal spécialisé : Journal of Discrete Event Dynamic Systems : Theory and Applications).

Les systèmes à événements discrets (SED) sont des systèmes dynamiques dont l’espace des états est discret et dont l’évolution est conforme à l’occurrence d’événements physiques à des intervalles de temps éventuellement irréguliers ou inconnus. La trajectoire des états est constante par morceaux. Ces systèmes sont une abstraction utile pour décrire et modéliser de nombreux *systèmes* conçus par l’homme tels que :

les systèmes de production et ateliers flexibles : les méthodes de production manufacturière modernes se sont orientées vers des ateliers organisés en réseaux de machines multi-tâches à la place des anciennes “lignes de transfert” rigides de machines mono-tâche. Cette organisation, qui permet la production simultanée de petites ou moyennes séries en respectant des ratios de production, conduit à des systèmes plus souples mais beaucoup plus difficiles à gérer.

les systèmes informatiques : on vise là tout type de système, des plus microscopiques comme une “puce” spécialisée en traitement de signal, jusqu’aux plus macroscopiques comme un réseau d’ordinateurs communiquant par satellites.

Réseaux de communication et de transport : téléphone ou réseau ferroviaire sont des exemples de situation où des messages ou des trains doivent emprunter les mêmes voies sans se télescoper ou entrer en collision.

Planification de tâches : la gestion d’un chantier de construction ou d’un projet quelconque nécessite l’articulation de tâches dont certaines peuvent se dérouler en parallèle, c’est-à-dire sans ordre préétabli, alors que d’autres supportent des contraintes de précédence. Ces tâches peuvent faire appel à des ressources communes (hommes, machines).

Tous ces systèmes obéissent à des règles opérationnelles, ou à algorithmes, dont les transformations ont lieu en réponse à l’occurrence d’*événements* discrets asynchrones dans le temps (arrivée d'une sollicitation, d’un ordre d’exécution ou achèvement d'une tâche).

Considérons l’exemple illustré dans Figure 1. 1. L’évolution du niveau d’un stock est déterminée par l’occurrence les événements “arrivée d’une pièce” dans le stock et

“départ d’une pièce” du stock. Cet exemple est représentatif pour les systèmes pour lesquels :

- les grandeurs[1], telles que le nombre de produits dans un stock, le nombre de processeurs en activité, le nombre de clients, ..., sont discrètes

- l'évolution est conditionnée par l'occurrence d'événements à "certains instants", tels que la fin d'exécution d'une tâche, le franchissement d'un seuil devant entraîner une action, l'arrivée d'un produit, d'un client, la défaillance d'un dispositif, ...

Alors, de cet exemple, nous constatons que l’espace d’état est un ensemble discret (fini ou infini) où l’état change seulement à certains instants du temps, de façon instantanée suite à l’occurrence des événements.

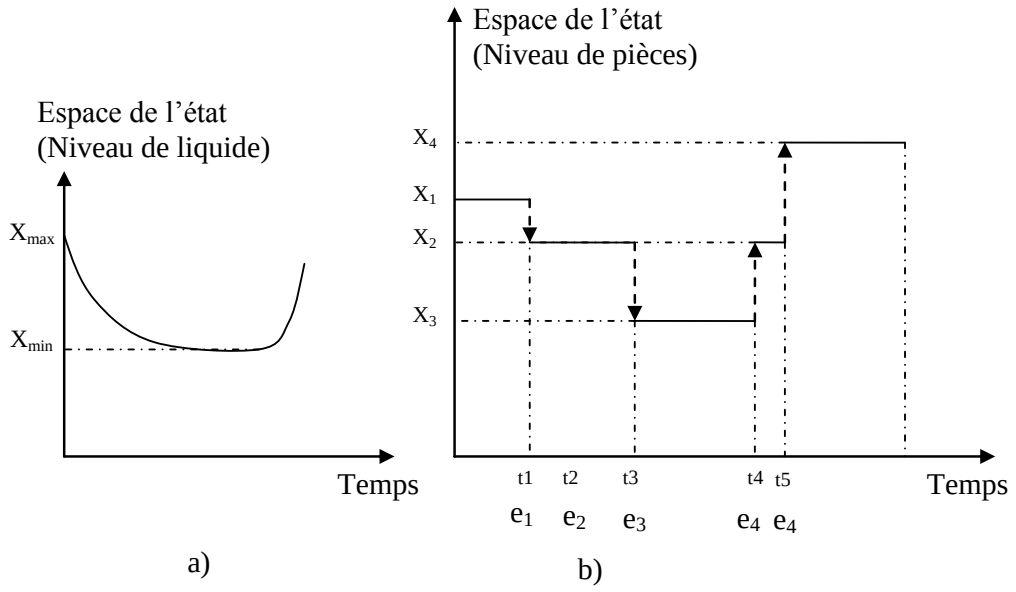


Figure 1. 1, ***Comparaison des trajectoires modéliser le niveau de stockage d’un atelier pour***
a) système continu b) Système à événements discrets

On peut remarquer aussi plusieurs caractéristiques propres pour ce système :

- l’ensemble des états (espace de l’état) $X=\{X_1, X_2, X_3, X_4\}$.
- l’ensemble d’événement $\Sigma=\{e_1, e_2, e_3, e_4\}$.
- l’espace de temps est discret $\tau=\{\tau_1, \tau_2, \tau_3, \tau_4\}$.
- chaque événement est associé à un certain moment.
- un même événement (e_4) peut se produit à des moments différents (τ_4 et τ_5).

[1] Des grandeurs telles que la position, la vitesse, l'accélération, le niveau, la pression, la température, le débit, la tension, le courant, ..., sont des variables continues, dans le sens qu'elles peuvent prendre n'importe quelle valeur dans R lorsque le temps, lui-même "continu", évolue.

- la séquence des états et le temps associé à ces états forment la trajectoire.
- un même événement (e_4) peut conduire à des états différents (X_1 et X_4).
- des événements différents (e_1 et e_4) peuvent conduire à un même état (X_2).
- des événements (e_2) peuvent se produire et être inactif pour le système.

La plupart de ces systèmes présente les caractéristiques communes suivantes :

Parallélisme : de nombreux événements peuvent se dérouler simultanément et indépendamment dans diverses parties du système.

Synchronisation : l'accomplissement de certains événements nécessite la disponibilité simultanée de plusieurs ressources ou la vérification simultanée de plusieurs conditions. La fin d'un événement entraîne l'apparition simultanée de plusieurs autres événements. Par exemple, pour qu'une conversation téléphonique ait lieu, il faut qu'une ligne soit disponible pour acheminer l'appel et que les deux interlocuteurs aient décroché. La fin de la conversation marque la libération de la ligne, et le fait que les deux interlocuteurs peuvent désormais vaquer à d'autres occupations. De telles considérations peuvent être reprises par exemple à propos d'un atelier flexible.

Concurrence : certains événements excluent l'apparition simultanée d'autres événements. Par exemple, une machine ne peut travailler que sur une seule pièce à la fois.

Ce qui suit est une liste de problèmes-types soulevés par la conception, la réalisation, la conduite ou la gestion, l'optimisation de ces systèmes :

Spécification *: avant de concevoir un système, il faut dire ce qu'on veut lui faire faire, quelle doit être sa réponse dans un certain nombre de situations, etc. il se pose donc déjà un problème de langage dans lequel ces desiderata trouveront une expression précise.*

Conception *: une fois spécifié le comportement fonctionnel du système, il faut le concevoir, notamment du point de vue de son architecture.*

Validation logique *: il faut ensuite vérifier que le système ainsi conçu répond bien aux spécifications désirées, et qu'il n'engendre pas d'autres comportements indésirables.*

Evaluation de performance *: à cette étape, la notion de temps, jusque-là, absente intervient. On cherche alors à répondre à des questions du type :*

combien d'événements d'un type donné se produisent en une heure, à quelle date se produira le n-ième événement, etc. ?

***Ordonnancement** : pour établir les politiques de priorité, de routage, etc., destinées à résoudre les problèmes posés par les phénomènes de concurrence.*

***Optimisation** : l'ordonnancement peut être un premier (levier) d'une optimisation de performance. Mais dès le choix de l'architecture, on risque d'avoir éventuellement engagé de façon assez déterminante la performance du système. A posteriori, on peut essayer d'améliorer les performances en dupliquant certaines ressources mais ceci n'a généralement de sens que sous une contrainte de budget donné, ou de coût à minimiser.*

1.2 Modélisation des Système à événements discrets

Un modèle est une approximation, une vue partielle plus ou moins abstraite de la réalité, selon un point de vue établi pour un objectif donné. Il peut être exprimé par des relations mathématiques, des représentations graphiques, des symboles, des lettres, …

L'objectif d'une modélisation peut être de *comprendre :*

1. *le comportement dynamique du système,*
2. *de comparer des configurations,*
3. *d'évaluer différentes stratégies de pilotage,*
4. *d'évaluer et d'optimiser des performances.*

Les domaines d'application sont divers. Sont listés ci-dessous quelques classes d'applications et quelques exemples de problèmes typiques rattachés à ces classes :

- Systèmes de flux de production

  - équilibrage de lignes d'assemblage,
  - conception de systèmes de transfert entre des postes,
  - dimensionnement des stocks d'un atelier,
  - comparaison de pilotage,
  - évaluation de la charge prévisionnelle de ressources,
  - étude de la synchronisation entre les réceptions des pièces et l'assemblage.

- Flux logistiques et systèmes de transport

  - conception et dimensionnement d'entrepôts,
  - dimensionnement d'une flotte de camions,
  - étude de procédures de contrôle des flux de véhicules en circulation.

- Production des services

  - étude de transactions bancaires,
  - gestion de restaurants,
  - comparaisons de politiques de maintenance des avions.

- Systèmes informatiques et télécommunications

  - évaluation de protocoles de gestion des transactions de données,
  - étude de la file d'attente mémoire d'un serveur,
  - étude des comportements des utilisateurs,
  - conception et dimensionnement de hubs («concentrateurs »).

- Autres classes d'applications

  - domaine militaire (support logistique, coordination des opérations),
  - gestion d'hôpitaux (personnel, lits, service d'urgence, …),
  - contrôle d'installation nucléaire, prévision de la météo,...

Différents vues du comportement d'un système peuvent être considérés selon l'application envisagée. Par conséquent, différents outils de modélisation et d'analyse des SED ont été développés. Il existe trois niveaux d'abstractions pour la modélisation et l'analyse des SED :

1. logiques
2. temporisés
3. stochastiques.

Le choix du niveau d'abstraction approprié dépend des objectifs de l'analyse.

Dans certaines applications, on s'intéresse au comportement logique du système (on néglige l'information du temps), pour assurer qu'un *ordre d'événements* précis se produit afin de satisfaire un ensemble de spécifications. Par exemple : premier arrivé premier servi pour un système de production. On peut aussi s'intéresser à chercher si un état (ou un ensemble d'états) est atteint ou non. Dans ce contexte, l'instant d'occurrence des événements est inutile, et il est suffisant de modéliser le comportement non temporisé du système en utilisant des outils de modélisation comme les automates ou les réseaux de Petri autonomes.

Dans d'autres applications, le temps est essentiel et doit être pris en compte explicitement par le modèle. Ces modèles sont appelés temporisés. Il s'agit des modèles où le temps est déterministe comme les réseaux de Petri temporels et les automates temporisés, et les modèles stochastiques où le temps est aléatoire comme les chaines de Markov et les réseaux de files d'attente.

Parmi l'ensemble des modèles existants, les réseaux de Petri (RdP) sont l'outil choisi pour le travail présenté dans ce mémoire, car ils possèdent un intérêt indéniable grâce à leur capacité à modéliser et à analyser les comportements des systèmes parallèles et distribués, synchronisés et communicants. Par leur relation avec les machines à états, les réseaux de Petri permettent des représentations faciles à comprendre et donc à manipuler, à la fois pour la création des modèles et pour leur analyse. Ils peuvent être utilisés pour la vérification des propriétés.

Dans la suite, nous allons rappeler brièvement quelques outils utilisés pour modéliser les Systèmes à événements discrets.

1.2.1 Langage formel et les automates à états finis

Un des moyens formels pour étudier le comportement logique des SED est basé sur la théorie des langages et des automates.

L'aspect temporel du système est ignoré, seuls l'ordre et l'identité des événements générés par le système sont pris en compte. Par conséquent, un modèle logique où seulement les séquences d'événements sont manipulées est parfaitement adapté à ce type de problème.

1.2.1.1 Langage formel

Ces langages permettent de traiter mathématiquement les problèmes relatifs aux SED, essentiellement d'un point de vue logique. Les événements sont désignés par des symboles. L'ensemble de ces symboles constitue un alphabet fini, à partir duquel sont formés les séquences d'événements appelés *mots*. L'ensemble de ces mots constitue le langage du système *L*.

Alors, un langage *L* défini sur un alphabet Σ, est un ensemble de mots (séquences), formés à partir des symboles (événements) dans Σ. La longueur d'un mot s est notée $|s|$, représente le nombre d'événements qui le composent. Notons que Σ^* est l'ensemble de tous les mots possible sur un alphabet Σ. Puisque le symbole représente un événement, nous ne distinguons pas les événements de leurs symboles.

Par exemple, en considérant un alphabet $\Sigma=\{a, b, c\}$, nous pouvons définir les langages L_1 et L_2 où :

$L_1=\{\varepsilon, a, aa, aba\} \subseteq \Sigma^*$, L_2={tous les mots possibles de longueur finie qui commencent par le symbole *a* et termine par le symbole *c* } $\subseteq \Sigma^*$= {*ac*, *abc*, *aac*, *aabc*, *aaabc*,…}.

Le langage L_1 est constitué de 4 mots, un de ces mots est le mot vide ε ayant la longueur $|\varepsilon|=\phi$. Par contre la longueur du mot *aba* est | *aba* |=3.

Le langage L_2 est composé d'un nombre infini de mots, en conséquence il ne peut pas être complètement décrit par une énumération. L'ensemble 2^{Σ}, est l'ensemble de tous sous-ensembles de Σ. Par exemple, soit Σ={*a*,*b*}, alors 2^{Σ}={φ, {a}, {b}, {a, b}}.

L'enchaînement *uv* de deux séquences *u* et *v* est une nouvelle séquence qui définit la suite des symboles de *u* immédiatement suivis par les symboles de *v*. La séquence *u* est dite préfixe de *uv*. Le symbole vide *ε* représente l'élément identité: $u\varepsilon = \varepsilon u = u$ pour tout mot *u*.

Le préfixe-clôture d'un langage *L* est le langage contenant tous les préfixes des séquences de *L*. Notons que $\overline{L}$ est le préfixe-clôture du langage *L*.

$$\overline{L} = \{s_1 \in \Sigma^* \mid \exists\, s_2 \in \Sigma^*, s_1 s_2 \in L\}$$

Par exemple, le préfixe-clôture du langage *L* = {*ab*} est définie par $\overline{L}$ = {*ε*, *a*, *ab*}. Par définition, nous pouvons remarquer que $L \subseteq \overline{L}$ .Tout langage égal à son préfixe-clôture, est qualifié de préfixe-clos, c'est-à-dire, *L* est préfixe-clos si $L = \overline{L}$. Par exemple, le langage *L* = {*ab*} n'est pas préfixe-clos, mais le langage $\overline{L}$ = {*ε*, *a*, *ab*} est préfixe-clos.

1.2.1.2 Langage régulier

Le langage régulier peut être représenté par une expression régulière définie à partir d'expressions élémentaires et des opérations d'enchainement. Une expression régulière sur un alphabet *Σ* est une expression dont les opérandes sont des symboles de Σ, et dont les opérateurs sont pris dans l'ensemble {+, ., *}. Les opérateurs +, . et * sont interprétés respectivement comme les opérateurs d'union, de concaténation et de fermeture itérative sur des langages. Par exemple l'expression $a+b^*$ est une expression régulière. Ainsi, en général, plusieurs expressions régulières correspondent à un même langage régulier. Par exemple, soit Σ={*a*, *b*, *c*}, l'expression régulière (*a*+*b*).*c* définit le langage L={*a*, *b*, *ac*, *bc*}. Le langage régulier n'est pas nécessairement un langage fini, mais en revanche, tout langage fini est un langage régulier car il peut être décrit par une expression régulière.

1.2.1.3 Opérations sur les langages

Considérons deux langages L_1 et L_2.

- *L'intersection* de L_1 et L_2, noté $L_1 \cap L_2$, définie sur Σ, désigne le langage contenant tous les mots qui appartiennent à la fois à L_1 et à L_2. Formellement :

$$L_1 \cap L_2 = \{s \mid s \in L_1 \wedge s \in L_2\}$$

- *L'union* de L_1 et L_2, noté $L_1 \cup L_2$, définie sur Σ, désigne le langage contenant toutes les mots qui sont contenus soit dans L_1, ou soit dans L_2. Formellement :

$$L_1 \cup L_2 = \{s \mid s \in L_1 \vee s \in L_2\}$$

- *La concaténation* de L_1 et L_2 définie sur Σ, par :

$$L_1 . L_2 = \{s \mid s = uv, u \in L_1 \wedge v \in L_2\}$$

Le comportement d'un SED peut donc être modélisé par un langage préfixe-clos sur un alphabet Σ. Cette modélisation par des langages du comportement du SED est indépendante du modèle utilisé pour le SED lui-même. Celui-ci devrait être représenté simplement par un générateur de langage. Les automates sont les générateurs de langages utilisés par Ramadge et Wonham dans leurs travaux (Ramadge & Wonham, 1987- a), (Ramadge & Wonham, 1987-b) et (Ramadge & Wonham, 1989) qui ont posé les bases de la théorie du superviseur des systèmes à événements discrets.

1.2.2 Les automates

Un automate est une machine qui a des entrées et des sorties discrètes, et qui réagit à une modification de ses entrées en changeant ses sorties. Cette machine est constituée d'états et de transitions associées à des événements. Le changement de sortie (passer d'un état à un autre état) dépend du mot (séquence) fourni en entrée. Les automates à états finis sont des automates dont le nombre d'états est fini. Il est dit *déterministe* dans le sens où depuis tout état, il n'existe pas plusieurs arcs de sortie qui soient associées à un même symbole et qui conduisent à des états différents.

Considérons le diagramme états-transitions de Figure 1. 2, où les nœuds représentent les états et les arcs associés à des symboles de l'alphabet représentent les transitions entre les états. L'ensemble des états $Q = \{q_0, q_1, q_2, q_3\}$. L'état initial q_0 est représenté par un arc entrant. Les états finaux (ou marqués) sont représentés par des doubles cercles, ici : $Q_m = \{q_0, q_2\}$. L'ensemble des symboles associés aux arcs est

l'ensemble des événements (alphabet) Σ = {*a*, *b*, *c*, *d*}. Les arcs dans ce diagramme modélisent une représentation graphique de la fonction de transition $\delta : Q \times \Sigma \rightarrow Q$:

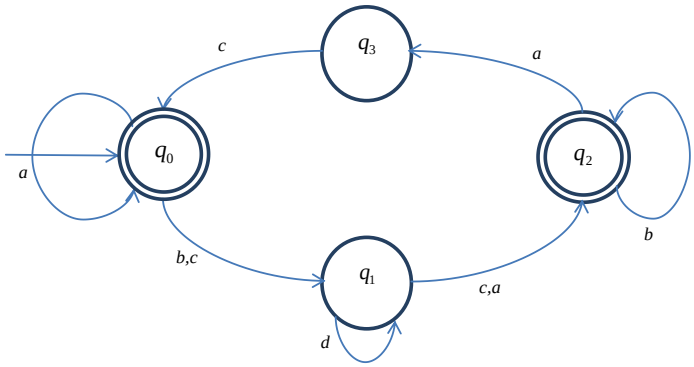


Figure 1. 2, ***Automate***

$\delta(q_0, a) = q_0$ $\quad$ $\delta(q_0, b) = \delta(q_0, b) = q_1$
$\delta(q_1, d) = q_1$ $\quad$ $\delta(q_1, c) = \delta(q_1, a) = q_2$
$\delta(q_2, b) = q_2$ $\quad$ $\delta(q_2, a) = q_3$
$\delta(q_3, c) = q_0$

On peut remarquer que :

- L'occurrence d'un événement ne change pas forcement d'état $\delta(q_0, a) = q_0$.
- A partir d'un état, on peut avoir deux événements différents dont l'occurrence mène à un même état ; $\delta(q_0, b) = \delta(q_0, c) = q_1$.
- La fonction δ est une fonction partielle dans son domaine $Q \times \Sigma$ puisqu'on n'a pas besoin d'avoir une transition pour chaque événement de Σ à partir de chaque état ; par exemple les instances $\delta(q_0, d)$ n'sont pas définies.

Un automate *A* est un quintuplet $A=(Q, \Sigma, \delta, q_0, Q_m)$ et peut être formellement défini comme suit :

- *Q est l'ensemble fini des états.*
- *Σ est un ensemble fini de symboles d'entrée (ensemble des événements), appelé alphabet d'entrée.*
- *δ est la fonction de transition d'états de $Q \times \Sigma$ vers Q qui associe un état d'arrivée à un état de départ et à un symbole d'entrée.*
- *$q_0 \in Q$ est l'état initial.*
- *$Q_m \subseteq Q$ est l'ensemble des états marqués.*

Définition 1. 1 : Un état $q \in Q$ est dit accessible s'il existe une chaîne $s \in \Sigma^*$ telle que $q = \delta(q, s)$. Par extension, un automate A est dit accessible si tous ses états sont accessibles.

Définition 1. 2 : Un état $q \in Q$ est dit co-accessible s'il existe une chaîne $s \in \Sigma^*$ telle que $\delta(q_0, s) \in Q_m$. Par extension, un automate A est dit co-accessible si tous ses états sont co-accessibles.

Définition 1. 3 : Le langage généré par un automate $A = (Q, \Sigma, \delta, q_0, Q_m)$ est

$$L(A) = \{s \in \Sigma^* \mid \delta(q_0, s) \text{ est défini}\}$$

Le langage marqué par l'automate A est

$$L_m(A) = \{s \in \Sigma^* \mid \delta(q_0, s) \in Q_m\}$$

Le langage généré par un automate est toujours préfixe-clos $\overline{L}(A) = L(A)$.

Le langage $L_m(A) \subseteq L(A)$ puisque $Q_m \subseteq Q$.

Définition 1. 4 : Les deux automates A_1 *et* A_2 sont dits équivalents *si* $L(A_1) = L(A_2)$ *et* $L_m(A_1) = L_m(A_2)$.

1.2.2.1 Composition synchrone de deux automates et produits

Ces opérations modélisent le comportement d'un système composé de sous-systèmes évoluant simultanément. Dans un souci de clarté, nous présentons cette opération pour deux automates.

Soit donc deux automates A_1 et A_2 définis par :

$A_1=(\Sigma_1, Q_1, \delta_1, q_{01}, Q_{m1})$ et $A_2=(\Sigma_2, Q_2, \delta_2, q_{02}, Q_{m2})$

La composition synchrone de A_1 et A_2 est l'automate:

$A_1 \,//\, A_2 = (\Sigma, Q, \delta, q_0, Q_m)$ où :

$$Q = Q_1 \times Q_2$$

$$\Sigma = \Sigma_1 \cup \Sigma_2$$

$$q = (q_{01}, q_{02})$$

Pour tout $q=(q_1, q_2) \in Q$ et pour tout $e \in \Sigma$, la fonction δ est définie de la façon suivante:

$$\delta((q_1,q_2),e)=\begin{cases}(\delta_1(q_1,e),\delta_2(q_2,e)) \text{ si } e\in\Psi_1(q_1)\cap\Psi_2(q_2)\\ (\delta_1(q_1,e),q_2) \text{ si } e\in\Psi_1(q_1)\setminus\Sigma_2\\ (q_1,\delta_2(q_2,e)) \text{ si } e\in\Psi_2(q_2)\setminus\Sigma_1\\ \text{non défini sinon}\end{cases}$$

où Ψ_1 et Ψ_2 sont les fonctions d'événements réalisables correspondants respectivement à A_1 et A_2. Par exemple $\Psi(q)$ est l'ensemble des événements réalisables à partir de l'état q.

Donc, les événements communs à A_1 et A_2 ne peuvent être exécutés que si les deux automates peuvent l'exécuter simultanément. Chaque automate exécute par ailleurs ses propres événements indépendamment de l'autre automate.

Contrairement à la composition synchrone, dans le produit de deux automates, seuls les événements communs peuvent être exécutés.

Considérons l'exemple dans Figure 1. 3, où on a 2 automates pour modéliser le fonctionnement du distributeur de jeton.

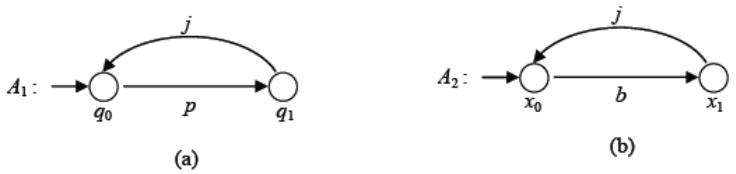


Figure 1. 3, ***Deux automates modélisant des distributeurs de jetons***

L'automate A_1 de la figure 1.3a est construit sur un alphabet de 2 symboles {*p*, *j*}. Nous pouvons remarquer que l'automate A_1 reconnaît l'ensemble des séquences telles que *p* se produit en premier et telles que *p* et *j* se produisent alternativement. Supposons que le symbole « *p* » modélise l'événement « introduction d'une *p*ièce » dans la machine et que « *j* » modélise l'événement « libération d'un *j*eton ». Alors, A_1 modélise la contrainte de fonctionnement que le jeton est délivré seulement si une pièce a préalablement été introduit dans la machine (nous supposons que les événements *p* et *j* apparaissent toujours alternativement). L'automate A_2 de la figure 1.3b est construit sur un alphabet de 2 symboles {*b*, *j*}. Le symbole « *b* » modélise l'événement « pression d'un *b*outon par l'utilisateur » et le symbole « *j* » a la même signification que dans A_1 (libération d'un *j*eton). Ce modèle exprime la contrainte qu'un jeton est délivré seulement si l'utilisateur a préalablement appuyé sur le bouton. A partir de A_1 et A_2, l'opération de *composition synchrone* permet d'obtenir un modèle global de notre machine. L'automate *M*, composé synchrone de A_1 et A_2 est représenté dans la Figure 1. 4. Ce modèle permet d'exprimer la conjonction des deux contraintes de fonctionnement définies par les automates A_1 et par A_2.

Dans l'automate M, un état correspond à un couple (q, x) où q est un état de A_1 et x est un état de A_2. L'état initial (q_0, x_0) est l'état correspondant aux états initiaux de A_1 et de A_2. L'automate M est construit sur un alphabet qui est la réunion des alphabets de A_1 et A_2, c'est-à-dire : $\{p, b, j\}$, où le symbole j est le seul événement commun aux alphabets de A_1 et A_2.

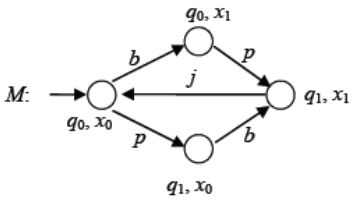


Figure 1. 4, ***l'automate modélise la composition synchrone de deux automates***

1.2.3 Les réseaux de Petri

Dès leur création en 1962 par Carl Adam Petri, les réseaux de Petri constituent un outil très approprié pour étudier les systèmes à événements discrets en raison de la puissance de modélisation et de leurs propriétés mathématiques. Les modèles obtenus, leur représentation graphique de la structure du système étudié, permettent une analyse des propriétés et par conséquent une validation de leurs spécifications. Les réseaux de Petri ont, par rapport aux automates, l'avantage d'être un modèle beaucoup plus général, bénéficiant de structures beaucoup plus riches, s'adaptant parfaitement à la description de différents types de SED. Les réseaux de Petri (RdP), qui sont l'outil choisi dans le travail présenté dans ce mémoire, possèdent un intérêt fondamental indéniable puisqu'ils permettent de modéliser et de maîtriser les comportements des systèmes parallèles et distribués, synchronisés et communicants. Par leur relation avec les machines à états, les réseaux de Petri constituent des représentations faciles à comprendre et donc à manipuler, à la fois pour la création des modèles et pour leur analyse.

Dans la suite, nous présentons d'abord dans cette section les concepts de base des réseaux de Petri et les notations correspondantes, utilisées dans la suite de cette thèse. Pour plus de détail sur les réseaux de Petri, le lecteur pourra consulter par exemple (Murata, 1989).

1.2.3.1 Notions de Base

Un Réseau de Petri (voir Figure 1. 5) est un graphe orienté composé de deux types de nœuds, de places et de transitions. Des arcs relient certaines places à certaines transitions ou certaines transitions à certaines places. Un arc ne peut pas relier deux nœuds de même type.

L'ensemble fini de places, $P = \{p_1, p_2, p_3, ..., p_m\}$, symbolisées par des cercles et représentant des conditions :

- Une ressource du système (ex. : une machine, un stock, un convoyeur,…).
- L'état d'une ressource du système (ex. : machine libre, stock vide, convoyeur en panne, …).
- …

L'ensemble fini de transitions, $T = \{t_1, t_2, t_3, ..., t_n\}$, symbolisées par des traits et représentant l'ensemble des événements (les actions se déroulant dans le système) dont l'occurrence provoque la modification de l'état du système.

- Un ensemble fini d'événements associés à chaque transition.
- Un ensemble fini d'arcs orientés qui assurent la liaison d'une place vers une transition ou d'une transition vers une place.

À l'intérieur des **places,** un nombre (positif ou nul) de **marques** ou **jetons** peut indiquer : par exemple, une condition relié à cette place est vraie (place marquée) ou fausse, ou la quantité de ressource en stock qui est représenté par cette place, etc. L'état d'un RdP est donné par le nombre de jetons dans chaque place.

Définition 1. 5 : De manière plus formelle, un RdP peut être défini comme un 4-uplet (P, T, P_{ost}, P_{re}):

- P et T sont respectivement l'ensemble fini et non vide de places et de transitions.
- P_{ost} est *l'application* d'incidence arrière, $P_{ost} : (T \times P) \rightarrow N^+$.
- P_{re} est l'application d'incidence avant, $P_{re}: (P \times T) \rightarrow N^+$.

Un arc est de la forme (p_i, t_j) ou (t_j, p_i), et le poids relatif à un arc est un entier positif. Si les poids des arcs sont tous égaux à 1, alors le réseau de Petri est dit ordinaire. Si l'arc $(p, t) \in A$, alors on dit que p (respectivement t) est une place d'entrée de la transition t (respectivement transition de sortie de la place p). Si une place p est à la fois place d'entrée et de sortie d'une même transition t, alors elle est dite impure, sinon elle est pure. Un RdP pur est un RdP qui ne contient aucune place impure, sinon il est impur. Dans la suite, nous utiliserons les notations suivantes :

- $p°$ (respectivement $°p$) désigne l'ensemble des transitions de sortie (respectivement d'entrée) de la place p.

- $t°$ (respectivement $°t$) désigne l'ensemble des places de sortie (respectivement d'entrée) de la transition t.

1.2.3.2 Matrice d'incidence

Au lieu de *Pré* et *Post* en général nous utiliserons la notation C qui est une matrice d'incidence calculable à partir de l'ensemble *Pré* et *Post* comme ci-dessous :

$$C = P_{ost} - P_{ré} = C^{+} - C^{-}$$

$$C^{-}[i,j] = Pre(p_i, t_j)$$

$$C^{+}[i,j] = Post(p_i, t_j)$$

Considérons l'exemple dans la figure 1.5, où la matrice d'incidence C de ce réseau est définie comme suit :

Arcs : $T \rightarrow P$, *matrice d'incidence arrière* C^{+}

	t_1	t_2	t_3	t_4	t_5
p_1	0	0	0	0	1
p_2	1	0	0	0	0
p_3	0	1	0	0	0
p_4	0	0	1	1	0

Arcs : $P \rightarrow T$, *matrice d'incidence avant* C^{-}

	t_1	t_2	t_3	t_4	t_5
p_1	1	1	0	0	0
p_2	0	0	1	0	0
p_3	0	0	0	1	0
p_4	0	0	0	0	1

Ainsi, la matrice d'incidence est :

$$C = C^{+} - C^{-} = \begin{bmatrix} -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 1 & -1 \end{bmatrix}$$

1.2.3.3 Vecteur de Marquage

Le vecteur de marquage M d'un réseau de Petri, est un vecteur représentant l'état courant de ce réseau par l'existence ou l'absence de jeton dans les places.

Le marquage initial M_0 (voir Figure 1. 6) décrit la distribution de jetons dans les places, au moment avant tout franchissement d'une transition (au début). Dans la suite et pour simplifier, on utilise le terme "marquage" au lieu de "vecteur de marquage". Tous les marquages accessibles à partir du marquage initial sont représentés par le *graphe de marquage* (ou le graphe d'atteignabilité) de ce système.

Le graphe des marquages d'un réseau est un graphe orienté dont les nœuds sont les marquages, et chaque arc relie un marquage à un autre qui est immédiatement accessible par une transition.

Considérons un RdP avec un marquage initial M_0. L'ensemble de marquages atteignables est noté $R(N, M_0)$.

$P=\{p_1, p_2, p_3, p_4\}$

$T=\{t_1, t_2, t_3, t_4\}$

$M_0=[1\ 0\ 0\ 0]^t$

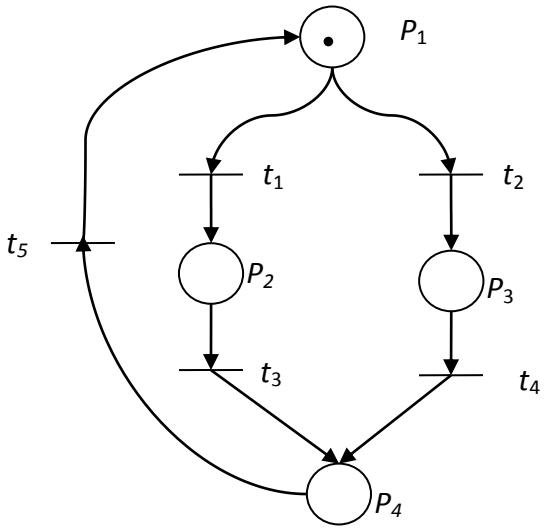


Figure 1. 5, ***l'exemple de figure 1.4 modélisé par un Réseau de Petri***

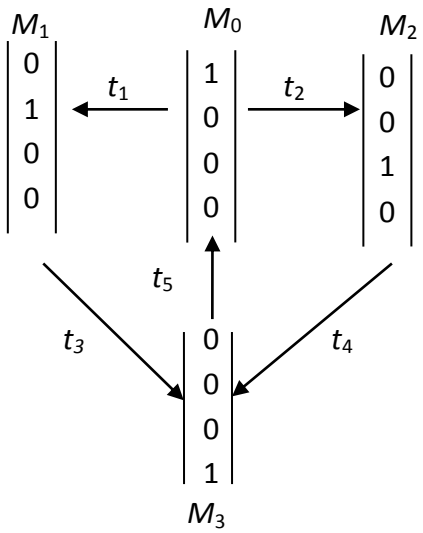


Figure 1. 6, ***Le graphe de marquage du réseau de figure 1.4.***

A partir de RdP, on peut facilement déduire automate par construction du graphe des marquages.

1.2.3.4 Règles de franchissement

La notation $M[t>$ indique que la transition t est franchissable à partir du marquage M.

On dit qu'une transition *t* est franchissable, si chaque place *p* en entrée contient un nombre de jetons supérieur ou égal à la value (poids) de l'arc qui la relie à la transition *t*. C'est-à-dire :

$$M \geq C_t^-$$

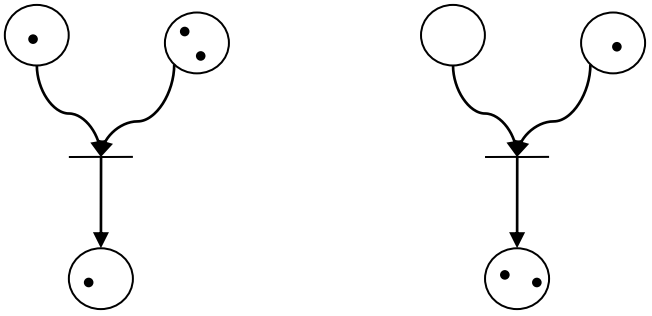
où C_t^- est la *t*-ième colonne de C^-.

Considérons le RdP illustré dans la figure 1.5, à partir du marquage initial M_0, les transitions franchissables sont (t_1 et t_2).

Le franchissement d'une transition permet d'atteindre un nouveau marquage *M*' à partir de *M*:

$$M[t>M'$$

$$M' = M + C_t$$



a) Avant de Franchissement de la transition

(La transition est franchissable)

b) Après de Franchissement de la transition

(La transition n'est plus franchissable)

Figure 1. 7, ***Franchissement d'une transition.***

Le vecteur C_t peut être calculé comme suit :

$C_t = C.[S]$

où [S] est le vecteur de séquence (événement). Considérant le même exemple de figure 1.5, en supposant que la séquence S $=t_1t_3$ est franchie, alors l'état de système sera amené de M_0 à M_3, où M_3 peut être défini comme ci-dessous :

$$M_3 = M_0 + \begin{bmatrix} -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 1 & -1 \end{bmatrix} [1\ 0\ 1\ 0\ 0]=[0\ 0\ 0\ 1]^t$$

1.2.3.5 Quelques propriétés des réseaux de Petri

Dans cette section, nous présentons les propriétés des réseaux de Petri ordinaires qui sont utiles dans cette thèse.

Transition vivante : Une transition t_j est *vivante* (voir Figure 1. 8-a) pour un marquage initial M_0 si pour tout marquage accessible $M_i \in (M_0,R)$, il existe une séquence de franchissement S qui contient la transition t_j, à partir de M_i. Par rapport à cette propriété, une transition peut être:

1. Non-vivante : si la transition t n'appartient à aucune séquence franchissable à partir de M_0.
2. L1-vivante : si la transition t, peut être franchie au moins une fois dans une des séquences franchissables à partir de M_0.
3. L2-vivante : si la transition t, peut être franchie au moins k fois dans des séquences franchissables à partir de M_0.
4. L3-vivante: si la transition t peut être franchissable infiniment dans des séquences franchissables à partir de M_0.
5. L4-vivante *ou* totalement vivante : si la transition t est L1-vivante pour tous les marquages M atteignables à partir de M_0.

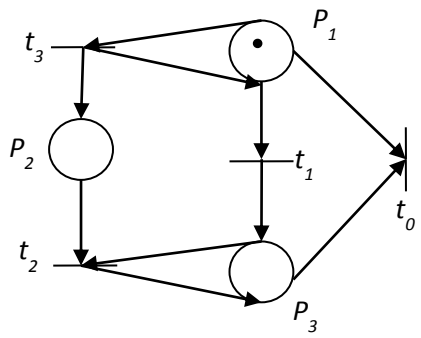
RdP vivant : Un RdP est *vivant* pour un marquage initial M_0 si toutes ses transitions sont vivantes pour M_0.

Blocage : un *blocage* (voir Figure 1. 8-c) est un marquage tel qu'aucune transition ne peut être validée.

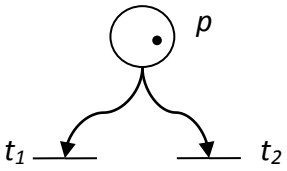
RdP borné : Une place p_i est dite *bornée* pour un marquage initial M_0 s'il existe un nombre entier naturel k, tel que pour tout marquage accessible à partir de M_0, le nombre de marques dans p_i est inférieur ou égal à k. si $k = 1$ le RdP est dit **sauf** (voir Figure 1. 8-d).

Conflit structurel (choix libre): Un *conflit structurel* (voir Figure 1. 8-b) correspond à un ensemble d'au moins 2 transitions t_1 et t_2 qui ont une place d'entrée en commun p_i.

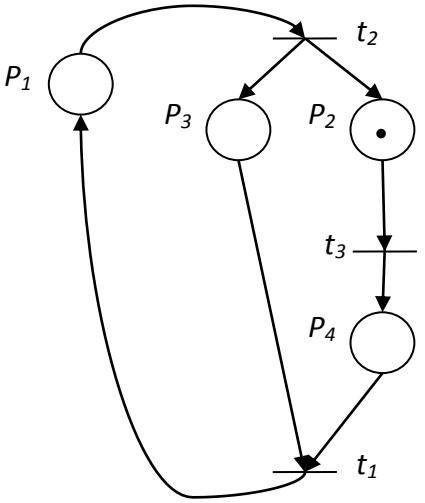
RdP réversible : Un RdP est réversible (voir Figure 1. 8-d) si, à partir de n'importe quel état atteignable M, il existe une séquence franchissable qui permet de revenir à M_0.



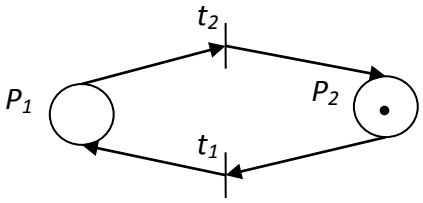
a) La transition t_0 (respectivement t_1, t_2 et t_3) est non-vivante (respectivement L1-vivante, L2-vivante et L3-vivante)



b) Conflit



c) Blocage



d) RdP Sauf et réversible

Figure 1. 8, ***Exemples de RdPs illustrant plusieurs propriétés.***

Invariant : La matrice d'incidence *C* nous permet de définir deux types d'invariance p-invariants et t-invariants.

- ***P-invariants*** **:** Un *p*-invariant d'un RdP est un vecteur *X* d'entiers non négatifs de dimension *card*(*P*) = *n* et qui vérifie

$$X^t.C = 0$$

où *C* est la matrice d'incidence du RdP.

En utilisant l'équation d'état des RdP, une propriété importante des p-invariants découle de leur définition, à savoir qu'un p-invariant *X* vérifie

$$\forall M \in R(N, M_0) : X^t.M = X^t.M_0$$

Ce qui signifie que la somme des marquages des places pondérés par le vecteur *X* est constante, quelle que soit l'évolution du RdP.

- ***T-invariants*** **:** Un *t*-invariant est un vecteur *Y* d'entiers positifs de dimension *card*(*T*) = *m* qui vérifie

$$C.Y^t = 0$$

Soit *S* une séquence franchissable à partir de M_0, telle que $\vec{S}$ est un *t*-invariant. Alors, en franchissant *S*, on revient au marquage initial. Cette propriété découle directement de la définition des *t*-invariants et de l'équation d'état. On peut dire que *S* est une séquence cyclique.

1.2.4 Graphe d'événements

Définition 1. 6 : un *graphe d'événements* est un RdP où chaque place a exactement une seule transition d'entrée et une transition de sortie. Autrement, ce type ne peut pas avoir un *conflit*. Mathématiquement, cette classe se décrit : $\forall p \in P : |p^\circ| = |^\circ p| = 1$.

Dans notre synthèse de superviseur, nous nous intéressons à cette classe qu'il est convenable de modéliser les lignes de productions (atelier flexible de machines à commande numérique, …).

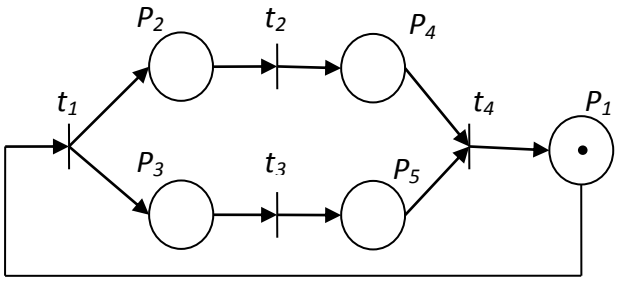


Figure 1. 9, ***Graphe d'événements***

Définition 1. 7: Un chemin π est un ensemble des nœuds se joignant successivement par des arcs.

Définition 1. 8: Le chemin de marquage $M(\pi)$ est la somme de jetons existant dans ce chemin π.

Les propriétés des graphes d'événements seront utilisées dans notre synthèse de commande dans chapitre 4 ,5 et 6.

1.3 Conclusion :

Dans ce chapitre, nous avons présenté la définition de SEDs, leurs types, leurs caractéristiques, les problèmes-types soulevés par la conception, la réalisation, la conduite ou la gestion et l'optimisation de ces systèmes. Ensuite, nous avons rappelé

des outils existants pour les modéliser et analyser leur comportement. Nous nous sommes intéressés en particulaire aux automates à états finis et aux réseaux de Petri, où nous avons présenté les définitions et les formalismes que nous allons utiliser plus tard dans les chapitres suivants.

Dans le chapitre qui suit, nous allons présenter les fondements de la théorie de la commande par supervision des Systèmes à Evénements Discrets (SED) tels que les résultats de Ramadge et Wonham (R/W) qui utilisent les automates pour modéliser les SEDs, et les approches utilisant les RdP pour modéliser le système et son superviseur.

Chapitre 2

Synthèse de commande pour les Systèmes à Evénements Discrets SED:

Résumé :

Dans ce chapitre, nous résumons une étude bibliographie de la théorie de la supervision des Systèmes à Evénements Discrets. Nous présentons tout d'abord les résultats fondamentaux de Ramadge et Wonham. Ces auteurs considèrent que le SED à superviser évolue en boucle fermée avec le superviseur qui, en fonction des séquences d'événements générées par le SED, intervient pour interdire des événements contrôlables pour garantir un comportement désiré. Leurs travaux se basent sur les langages formels pour modéliser un SED à superviser et le superviseur lui-même. Ces résultats ont constitué une base importante pour la majorité des travaux postérieurs dans ce domaine. Puis nous nous intéressons essentiellement aux travaux de recherche en modélisant le système et le superviseur à l'aide des réseaux de Petri.

2.1 Introduction

En incluant le problème de la contrôlabilité et de l'observabilité, nous présentons également une étude bibliographique concernant les travaux de synthèse de commande par superviseur pour les systèmes à événements discrets. L'approche de Ramadge et Wonham (Ramadge & Wonham, 1987- a), (Ramadge & Wonham, 1987-b) et (Ramadge & Wonham, 1989), que l'on considère une base importante pour la majorité des travaux postérieurs dans le domaine de la synthèse de commande, est premièrement montrée.

Modéliser un système et/ou des spécifications par RdP est généralement plus pratique que par un automate. L'inconvénient majeur de la synthèse de commande modélisée par les automates est l'explosion combinatoire du nombre d'états quand il s'agit d'étudier des systèmes de taille réelle. C'est pourquoi dans ce chapitre, nous nous intéressons à rappeler des travaux synthétisant un superviseur en utilisant le RdP. Dans ces approches, le procédé et le superviseur sont tous les deux modélisés par des RdPs. Les spécifications imposées sur le comportement du système sont modélisées mathématiquement, ensuite, leurs affects seront décrits par l'influence du superviseur couplé au système à superviser en restreignant le comportement de ce système.

Ce chapitre est organisé comme suit : tout d'abord nous présentons la signification des spécifications par lesquelles on décrit les comportements désirés et interdits. Des travaux présentent des approches transférant les spécifications aux modèles mathématiques. Nous discutons en particulier le problème d'état interdit, où nous révisons certains modèles mathématiques, surtout les modèles de (D. Corona, 2004), (Giua A. a.-C., 1993), (Holloway & Krogh, 1990), traitant ce problème. Puis, nous rappelons les travaux de base sur la synthèse de commande de (Ramadge & Wonham, 1987- a), (Ramadge & Wonham, 1987-b) et (Ramadge & Wonham, 1989). Ensuite, les approches de synthèse de commande basées sur les RdP sont présentées. Sous ce titre, nous distinguons plusieurs axes de recherches par rapport le modèle de spécifications et par rapport à la nature des événements : contrôlables et observables.

Dans la suite, nous utilisons l'abréviation SdC pour Synthèse de Commande.

2.2 Spécifications

Introduisant la problématique de la SdC, les contraintes sont des outils pour modéliser les spécifications, le système à superviser devant les respecter pour éviter les états "illégaux", i.e. ces spécifications nous permettent d'introduire les comportements désirés d'une manière mathématique. Selon la nature des spécifications, les problèmes de SdC peuvent être classés en deux catégories : le problème d'états interdits (PEI), et le problème de transition d'états interdits.

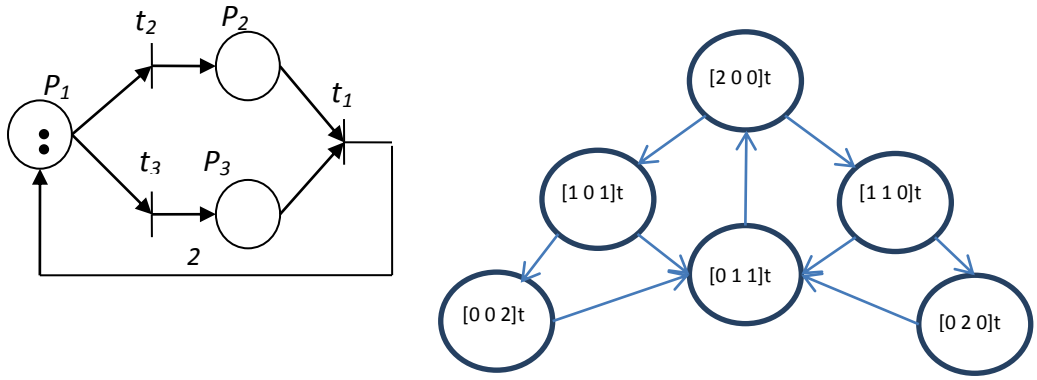
2.2.1 Problème d'états interdits (PEI)

Dans la synthèse de superviseur, en utilisant les RdPs, le problème d'états interdits est l'un des problèmes les plus modélisés. Dans ce problème, s'agit de supprimer un état ou un groupe d'états du graphe de marquages du RdP, pour arriver à construire un graphe correspondant seulement aux comportements autorisés.

Considérons le RdP représenté dans la Figure 2. 1-a, dont le graphe de marquages est donné dans la Figure 2. 1-b. L'ensemble des marquages interdits est {[0 2 0] et [0 0 2]}. Le problème d'états interdits (PEI) vise à contrôler le franchissement des transitions de telle manière que les marquages interdits ne soient pas atteignables. Formellement, le problème d'états interdits est une question de supervision à base de RdP où le comportement de système doit satisfaire les conditions suivantes :

1. $M_0 \in$ Comportement désiré
2. $\forall\, M \in$ Comportement désiré, $M\,[\delta > M' \in$ Comportement désiré.

où δ est une séquence de transitions franchies.



a) RdP *b) le* graphe d'atteignabilité

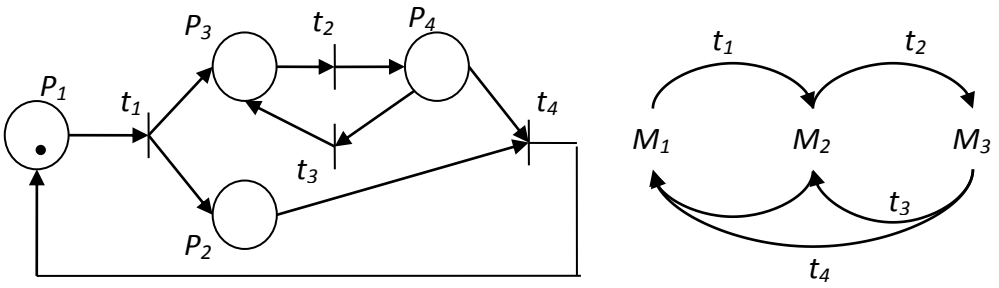
Figure 2. 1, ***Problème d'états interdits PEI***

2.2.2 Problème de transition d'états interdits (PTEI)

Le Problème de transitions d'états interdits est un problème plus général que celui des états interdits. Nous considérons dans l'exemple de la Figure 2. 2, le langage légal défini par $(t_1t_2t_4)^*$. D'après le graphe de marquages, on souhaite interdire le franchissement de la transition t_3 si la place P_4 est marquée, autrement dit, il faut interdire le franchissement de t_3 à partir de M_3. Alors tous les marquages M_1, M_2 et M_3 sont des marquages admissibles, par contre le franchissement d'une transition à partir d'un état est interdit. Notons que dans un PTEI, il est possible que des franchissements

de transitions entre des marquages admissibles soient interdits, car ils violent le comportement désiré.

Il est utile de rappeler que le problème d'états interdits est bien un problème de transitions d'états interdits, mais l'inverse n'est pas toujours vraie, car tous les franchissements de transitions conduisant à un état interdit dans un PEI sont interdits.



a) RdP *b) le* graphe d'atteignabilité

Figure 2. 2, ***Problème de transition d'états interdits***

2.3 Modèles de Contraintes

Les deux problèmes PEI et PTEI peuvent être représentés par plusieurs types de contraintes. Avant de présenter les approches les plus utilisées et les plus proches de notre contribution, nous rappelons quelques définitions qui sont utilisées dans ces approches et dans notre thèse.

Définition 2. 1: Une place critique q est une place concernée au minimum par une contrainte. L'ensemble de ces places est formulé par Q_{mc}, où m_c est le nombre des places critiques.

Définition 2. 2: L'ensemble des marquages interdits M_f est défini par tous les marquages qui ne respectent pas les contraintes.

Définition 2. 3: L'ensemble des marquages légaux M_L est défini par tous les marquages qui respectent les contraintes.

$$M_L=R(N,M_0)\backslash M_f \qquad \textbf{(2. 1)}$$

Dans notre état-de-l'art, nous distinguons deux principaux types de modèles de contraintes : les contraintes linéaires et les contraintes logiques.

2.3.1 Contraintes linéaires

Sous la dénomination des contraintes linéaires, on distingue les contraintes d'égalité et les contraintes d'inégalité.

2.3.1.1 Contraintes d'égalité

Ce type de contrainte est décrit comme suit (Yamalidou, Moody, Lemmon, & Antsaklis, 1996):

$$\sum_{i=1}^{m_c} M_i = k$$

L'équation ci-dessus signifie que toutes les places concernées (places critiques) doivent former une place invariante, autrement dit, la somme de tous les marquages des places concernées doit être invariante.

2.3.1.2 Contrainte d'inégalité de forme (inférieur ou égale à)

Ce type de contrainte est décrit comme suit (Yamalidou, Moody, Lemmon, & Antsaklis, 1996):

$$\sum_{i=1}^{m_c} M_i \leq k$$

Il signifie que la somme des jetons dans les places concernées (places critiques) ne doit pas dépasser le nombre entier *k*.

2.3.1.3 Contrainte d'inégalité de forme (supérieur ou égale à)

Dans certains cas, un ensemble des places doit garder au minium un nombre entier *k* de jetons. Ce cas est décrit par (Yamalidou, Moody, Lemmon, & Antsaklis, 1996):

$$\sum_{i=1}^{m_c} M_i \geq k$$

C'est-à-dire que la somme de jetons dans cet ensemble est toujours au moins égale à *k*.

2.3.1.4 Contraintes Généralisées d'Exclusion Mutuelles CGEM

Une Contrainte Généralisée d'Exclusion Mutuelle (CGEM) est une condition limitant la somme pondérée des marques dans un ensemble de places critiques (Guia, 1992). L'auteur de (Guia, 1992) a discuté la puissance de modélisation de cette contrainte. Il a prouvé que pour une classe de RdP, tout état interdit peut être exprimé par CGEM.

Définition 2. 4: Une CGEM est une condition qui limite la somme pondérée de marques dans un ensemble de places. Ce type de contraintes permet de traduire naturellement l'exclusion mutuelle dans l'utilisation de ressources communes

partagées entre différents processus. Ces contraintes sont décrites par des inégalités linéaires sur le marquage de certaines places pour définir l'ensemble des états admissibles. Formellement, chaque CGEM est définie par une paire ($\vec{w}$, *k*) telle que :

$\vec{w}$ est une vecteur d'entiers et *k* est une constante entière :

$$M_{(\vec{w},k)}=\{M\in R(N, M_0)\backslash \vec{w}.M\leq k\}$$

où: $M_{(\vec{w},k)}$ est l'ensemble légal de marquages décrit par une CGEM. Dans le cas où le système est soumis à un nombre τ de CGEMs, l'ensemble légal totale de marquages est l'intersection des ensembles légaux par rapport à chaque contrainte, qui est donné par :

$$M_{(\vec{w},k)}=\bigcap_{i=1}^{\tau} M_{(\vec{w}_i,k_i)}$$

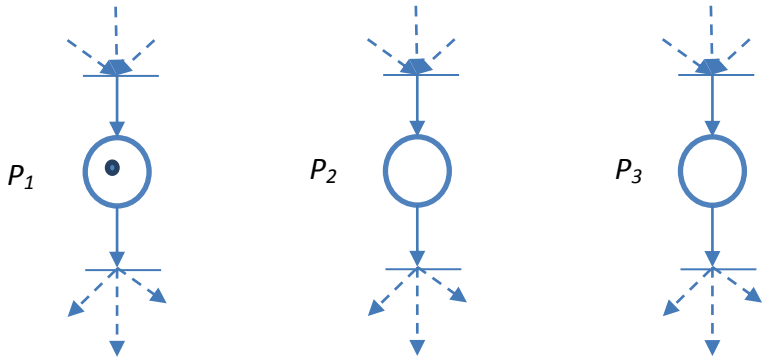
Lorsque *w*=1, la CGEM devient une contrainte d'inégalité de forme *inférieur ou égale à*.

Les états décrits par une CGEM, sont les états admissibles. Dans (Giua A. a.-C., 1993), les auteurs ont montré que dans le cas général il n'est pas toujours possible de modéliser le problème d'états interdits par CGEM, mais dans le cas où le RdP est sauf et conservatif, cela est toujours possible.

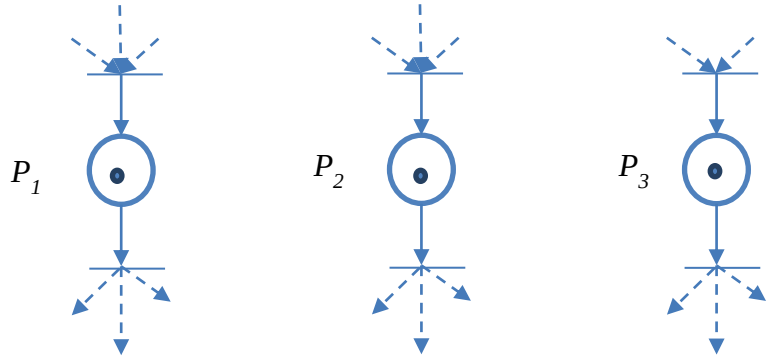
2.3.2 Contraintes de type prédicat

Sous ce titre, nous rappelons les modèles de contraintes proposés dans (Holloway & Krogh, 1990). Ils ont considéré un graphe d'événements sauf et conservatif et ont présenté trois types de contraintes : le premier modèle concerne une place critique, tandis que le second concerne un ensemble des places critiques et le dernier est une classe de contraintes.

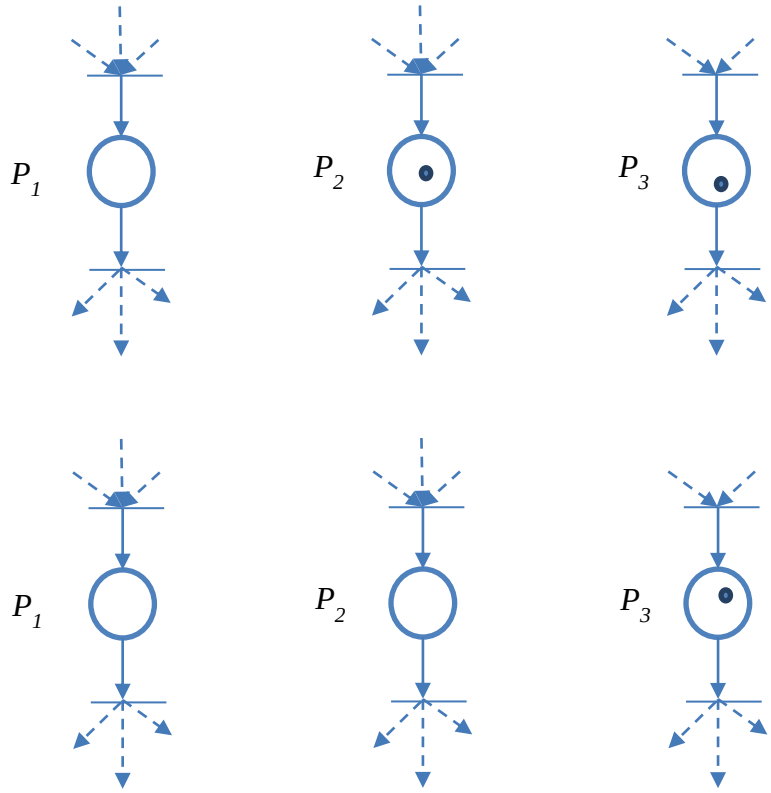
Définition 2. 5: Une *condition de place* pour un graphe d'événements, est une contrainte d'états interdits, appliquée à une place critique $p\in P$ où le marquage $M\in R(N,M_0)$ satisfait une condition de place *p* si $M(p)=1$, voir la figure 2.3 -a.



a) Etat interdit lorsque $M(p_1)=1$



b) Etat interdit lorsque $M=[1\ \ 1\ \ 1]^t$



c) Ensemble d'états interdits lorsque $M=\{M_1, M_2\}$, où $M_1=[0\ \ 1\ \ 1]^t$ et $M_2=[0\ \ 0\ \ 1]^t$

Figure 2. 3, *modèles de contraintes de type de prédicat*

Définition 2. 6: *Une condition de type ensemble* pour un graphe d'événements, est une contrainte d'états interdits, appliquée sur un ensemble des place critiques $\mathcal{B} \subseteq P$ où le marquage $M \in R(N,M_0)$ satisfait une condition de type ensemble B si $M(p)=1 \ \forall \ p \in \mathcal{B}$, voir la figure 2.3 -b.

Définition 2. 7: pour un graphe d'événements, est une contrainte d'état interdit définie par une disjonction de conditions de type ensemble $\mathfrak{I} \subseteq 2^P$ où le marquage $M \in R(N,M_0)$ satisfait une condition de type classe $\mathfrak{I}$ s'il satisfait certains conditions de type ensemble des $B \in \mathfrak{I}$, voir la figure 2.3 -c.

2.4 SdC par supervision

En général, dans la littérature, les auteurs ne font pas une distinction claire, entre la définition de contrôleur (synthèse par contrôleur) et le superviseur (synthèse par superviseur). Mais on peut localiser chaque définition pour identifier chaque concept.

***La Commande par Contrôleur*:** il déclenche l'exécution d'un ensemble d'opérations en donnant des ordres aux actionneurs. Il peut être constitué :

- d'un ensemble d'opérations correspondant à une séquence d'opérations pour la réalisation d'un produit ou d'un service.
- d'un ensemble d'opérations exécutées afin de restaurer la fonctionnalité du procédé offerte durant l'exécution normale.
- d'un ensemble d'actions avec un degré élevé de priorité engagées afin de protéger les opérateurs humains et prévenir les évolutions catastrophiques.
- d'un ensemble d'opérations de test, de réglage ou de maintenances exécutées afin de garder le procédé dans son état opérationnel.

***La commande par Superviseur* :** il calcule et met à jour les paramètres de la séquence de commande à exécuter en tenant compte de l'état du système de commande et de l'état du procédé. Le rôle du superviseur est d'interdire l'occurrence d'événements critiques dans le procédé. Il ne peut en aucun cas forcer des événements à se produire (comme le contrôleur). i.e., le superviseur ne peut que restreindre le fonctionnement du procédé. Il inclut les opérations normales et anormales (inattendues) **:**

- Durant une opération normale, le superviseur prend des décisions pour lever l'indécision dans le système de commande (ordonnancement temps-réel, optimisation, mise à jour de la commande, et remplacement d'une loi de commande par une autre).

- Quand un signe de défaillance (opération anormale) est identifié, le superviseur prend toutes les décisions nécessaires pour permettre au système de reprendre son fonctionnement normal (ré-ordonnancement, actions de recouvrement, procédures d'urgence, etc.).

Dans notre travail, nous nous intéressons au concept de superviseur.

Afin de présenter les travaux de SdC par superviseur (ou parfois de contrôleur, dans cette section le concept de contrôleur et celui de superviseur sont confondus). On classifie cette synthèse en deux axes :

Le premier axe : Synthèse de superviseur basé sur le langage formel.

Le deuxième axe : Synthèse de superviseur basé sur le réseau de Petri.

2.4.1 Synthèse de superviseur basée sur langage formel et automate

La théorie de la supervision des systèmes à événement discrets a été initiée par les travaux de Ramadge et Wonham (Ramadge & Wonham, 1987- a), (Ramadge & Wonham, 1987-b) et (Ramadge & Wonham, 1989). Dans leur approche, des modèles logiques (négligeant le temps) sont utilisés pour décrire le comportement d'un système. Un procédé est considéré comme SED s'il évolue spontanément en générant des événements.

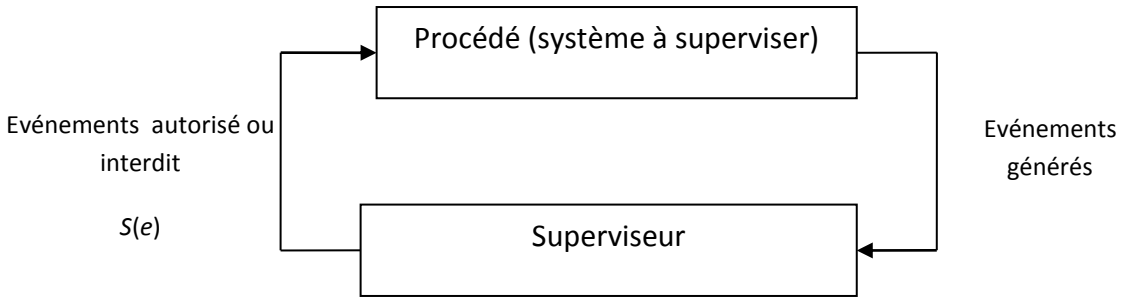


Figure 2. 4, ***Schéma de supervision***

Dans ce schéma de la Figure 2. 4, un procédé est couplé à un superviseur. Les entrées du superviseur sont les sorties du procédé, et vice versa. Le rôle du superviseur est d'autoriser ou d'interdire l'occurrence d'événements dans le procédé. Ces événements sont dits contrôlables. L'ensemble des événements contrôlables est noté Σ_c. Par contre, il ne peut en aucun cas inhiber l'occurrence d'événements dits incontrôlables (par exemple la panne de la machine). L'ensemble des événements incontrôlable est noté Σ_u. Par ce rôle, le superviseur peut changer le comportement du procédé. Ce type de supervision est appelé contrôle par retour d'événement. Etant donnés un procédé et un ensemble de spécifications logiques de fonctionnement,

l'objectif de cette théorie est de synthétiser un superviseur tel que le fonctionnement du procédé couplé au superviseur respecte les spécifications.

2.4.1.1 Superviseur par automate

Dans leurs travaux le système à superviser (Procédé *G*) est modélisé par un automate *G*. Son comportement est donc donné par le langage *L*(*G*) sur l'alphabet Σ. Une hypothèse de cette approche est qu'il est impossible de générer simultanément deux événements indépendants par le procédé. Le comportement de *G* peut être insatisfaisant dans le sens où il ne respecte pas certaines propriétés appelées *objectifs de contrôle* (*Spécification ou Contrainte*). Il est donc nécessaire de restreindre ce comportement dans le but d'assurer ces objectifs. Ce fait est achevé par un superviseur qui va décider l'ensemble des événements qui doivent être interdits pour rester dans l'ensemble des comportements décrits par l'objectif. Superviser un système consiste alors à ajouter des contraintes supplémentaires pour assurer le fonctionnement du système (procédé), afin d'éviter les comportements non désirés.

Un système couplé à un superviseur peut être perçu comme un système qui reçoit en entrée à chaque étape une liste d'événements interdits $\Sigma_{interdit}$, afin qu'ils ne soient pas exécutés dans l'étape suivante. Les événements générés à sa sortie $\Sigma_{généré}$ à l'état actuel (*i*) sont donc le résultat d'exclusion de l'ensemble $\Sigma_{interdit}$ à l'état précédent (*i*-1) de l'ensemble des événements Σ. Ce procédé est appelé *procédé supervisé* (*S/G*) (Figure 2. 5).

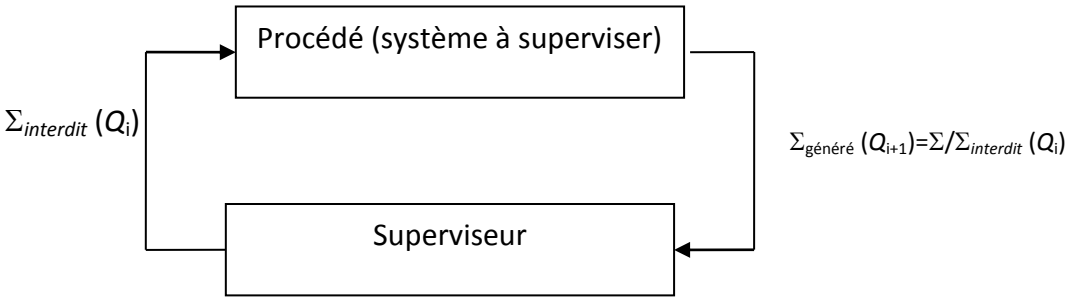


Figure 2. 5, ***Schéma de supervision : les événements interdits $\Sigma_{interdit}$ avec l'incidence des étapes.***

De manière générale, considérons que Σ est l'ensemble des événements d'un procédé. Cet ensemble est peut être subdivisé à deux sous-ensembles : $\Sigma = \Sigma_c \cup \Sigma_u$, où Σ_c (Σ_u respectivement) définie l'ensemble d'événements contrôlables (incontrôlables respectivement) sur Σ. Au contraire des événements contrôlables, les événements incontrôlables ne peuvent pas être interdits par la supervision, i.e. $\Sigma_{interdit} \cap \Sigma_u = \varphi$.

Note : le superviseur peut définir une liste des événements autorisés $\Sigma_{autorisé}$ au lieu d'une d'événements interdits $\Sigma_{interdit}$, où : $\Sigma_{autorisé} = \Sigma \ / \ \Sigma_{interdit}$.

2.4.1.2 Langage de système supervisé

Considérons deux langages L et L_m définis sur l'ensemble des événements Σ, où L est l'ensemble des chaînes que le SED peut générer et $L_m \subseteq L$ est le langage marqué utilisé pour représenter les fins d'opérations. L est toujours préfixe-clos, i.e., $L = \overline{L}$. Considérons l'automate $G = (Q, \Sigma, \delta, Q_0, Q_m)$ tel que $L(G) = L$ et $L_m(G) = L_m$. L'objectif de la commande par supervision est de concevoir un superviseur S afin de le coupler en boucle fermée avec le procédé G pour assurer le comportement désiré. La fonction de transition de G peut être commandée par S dans le sens où les événements contrôlables Σ_c de G peuvent être autorisés ou inhibés par S. Formellement, un superviseur S est une fonction $S : L(G) \rightarrow 2^{\Sigma}$.

Pour toute séquence $s \in L(G)$, $S(s) \cap \Gamma(\delta(Q_0, s))$ est l'ensemble des événements autorisés par le superviseur S et que G peut exécuter à partir de son état courant $\delta(Q_0, s)$. i.e., G ne peut pas exécuter un événement appartenant à l'ensemble des événements possibles $\Gamma(\delta(Q_0, s)$ à partir de son état courant $\delta(Q_0, s)$ si cet événement n'est pas aussi contenu dans $S(s)$.

Le système supervisé S/G est un SED dont on peut déterminer son langage généré ainsi que son langage marqué. Ces deux langages sont des sous-ensembles de $L(G)$ et $L_m(G)$ contenant les séquences possibles en présence de S. Formellement, ceci est défini comme suit.

Définition 2. 8: Langage généré et langage marqué par S/G.

Le langage généré par S/G est défini comme suit :

- $\varepsilon \in L(S/G)$ *et*
- $[(s \in L(S/G))$ *et* $(s\sigma \in L(G))$ *et* $(\sigma \in S(s))] \Leftrightarrow [s\sigma \in L(S/G)]$.

Le langage marqué par S/G est défini comme suit :

$$L_m(S/G) := L(S/G) \cap L_m(G)$$

Dans les modèles où les états marqués correspondent à des fins de tâches, ce langage revêt de l'importance dans la mesure où il caractérise les séquences intéressantes qui permettent d'accomplir des tâches tout en respectant les contraintes de supervision. On dira que le superviseur est non-bloquant pour G si toute séquence du système contrôlé peut toujours être complétée pour atteindre un état marqué, c'est-à-dire :

$$L_m(S/G) \subseteq \overline{L_m(S/G)} \subseteq L(S/G) \subseteq L(G)$$

2.4.1.3 Langage et contrôlabilité

Considérons un procédé *G* et une spécification S_{pec}. On souhaite synthétiser un superviseur *S* de façon à ce que le système en boucle fermé *S*/G, respecte la spécification S_{pec}, i.e., il nous faut trouver le langage $L_D = L(G) \cap L(S_{spec})$, que nous appelons *fonctionnement désiré* qui correspond à l'ensemble des séquences qui peuvent être générées par le procédé et qui sont tolérées par la spécification. S'il y a des événements incontrôlables Σ_u, il n'est pas toujours possible de restreindre le comportement d'un procédé par la supervision. L'existence d'un superviseur *S* tel que $L(S/G) = L_D$ réside dans le concept de contrôlabilité.

Définition 2. 9: Un langage *K* est dit contrôlable par rapport à un langage *L*(*G*) si :

$$\overline{K}\,\Sigma_u \cap L(G) \subseteq \overline{K}$$

où $\overline{K}$ représente le préfixe-clôture de langage de spécification et *L*(*G*) le langage de procédé.

Autrement, si pour toutes séquences $s \in K$ et pour tout événement incontrôlable $e \in \Sigma_u$, la séquence $s.e \in L(G)$, cela implique que cette séquence appartient aussi à *K*. On appelle ce langage *K* contrôlable par rapport à un langage *L*(*G*).

Ce problème a été traité dans (Ramadge & Wonham, 1987- a), (Ramadge & Wonham, 1987-b) et (Ramadge & Wonham, 1989). Pareillement, (Kumar, 1992) présente une approche qui définit un langage contrôlable maximal permissif, où la permissivité est définie dans le sens d'éviter un ensemble le plus petit possible en incluant les états interdits. Dans son approche (Kumar, 1992), Kumar présente un algorithme pour trouver le langage contrôlable *K* qui garantit le respect des spécifications S_{pec}. Dans cette approche, lorsqu'un procédé comporte *n* états et une spécification comporte *m* états, alors le superviseur défini par cette approche, comporte au maximum *n*.*m* états.

2.4.1.4 Conclusion de l'approche de Ramadge et Wonham

Au niveau mondial, les travaux de Ramadge et Wonham sont considérés comme une base pour plusieurs groupes de recherche. Dans cette approche, seul le système à contrôler (procédé) peut générer des événements alors que le superviseur a seulement un rôle d'inhibition de certains événements. Par conséquent, le système est seul maître du cadencement des événements.

La taille du modèle décrit dans leur approche augmente fortement avec la taille du système considéré, parce qu'il est basé sur des automates à états et sur un point de vue centralisé. Des approches ultérieures permettant de réduire cette taille par la remise en cause du point de vue centralisé : point de vue hiérarchique, modulaire sous

observation partielle ou décentralisé (Lin & Wonham, 1990), (Gaudin, 2004) et (Takai, Ushio, & Kodama, 1995).

Une fois la commande obtenue, une implantation est nécessaire. Là aussi, différents travaux proposent des interprétations sous forme de schémas *Ladder Diagram* ou de *Grafcet* (Charbonnier, Alla, & David, 1999) et (Kattan, 2004).

L'aspect des contraintes temporelles des travaux de Ramadge et Wonham est étendu dans différents approches comme (Brandin & Wonham, 1994) et (Sava, 2001).

2.4.2 Synthèse de superviseur basée sur les réseaux de Petri

L'inconvénient majeur de la SdC modélisée par les automates est l'explosion combinatoire du nombre d'états quand il s'agit d'étudier des systèmes de taille réelle. La lisibilité du modèle obtenu est également réduite. Les RdP nous permettent de pallier à cet inconvénient. Grâce à leur facilité à fournir une représentation graphique intuitive explicite des comportements tels que le parallélisme et le partage des ressources. Plusieurs approches ont utilisé les RdP dans ce domaine, parfois pour traiter le problème de la contrôlabilité et parfois pour attaquer le problème de l'observabilité. Dans ces approches, tous les deux (procédé et le superviseur) sont modélisés par des RdPs. Les spécifications sont modélisées mathématiquement, ensuite, leurs affects seront décrits par l'influence du superviseur couplé au système à superviser en restreignant le comportement de ce système.

Nous montrons dans la suite, des travaux importants qui utilisent les RdPs comme un outil de modélisation de SED, et en même temps un outil pour modéliser le superviseur. Nous classifions aussi ces approches en trois catégories :

1- Approches dédiées aux spécifications de type CGEM : (Guia, 1992), (Giua A. a.-C., 1993), (Giua & Seatzu, 2002), (Yamalidou, Moody, Lemmon, & Antsaklis, 1996), (Moody, Lemmon, & Antsaklis, 1996), (Moody & Antsaklis, 1998), (Basile F. , Recalde, Chiacchio, & Silva, 2000) et (Basile F. a., 2001), (Basile F. C., 2006), (Basile F. A., 2007) et (Basile F. , Recalde, Chiacchio, & Silva, 2009).

2- Approches basée sur la théorie de régions : (Ghaffari A. a., 2002) et (Achour Z. , 2005).

3- Approches basées sur des propriétés structurelles des RdP, on trouve ces approches traitant notamment le problème de superviseur pour les graphes d'événements : (Holloway & Krogh, 1990), (Holloway, Guan, & Zhang, 1996), (Boel, Ben-Naoum, & Breusegem, 1995) ,(Ghaffari A. X., 2003) et (Achour Z. a., 2004).

On peut aussi classifier la SdC par rapport la nature des événements :

1- Les systèmes totalement contrôlables
2- Les systèmes partiellement contrôlables
3- Les systèmes partiellement observables.

Dans ce chapitre, nous présentons les travaux qui traitent le problème de contrôlabilité et ensuite le problème d'observabilité et qui sont les plus appropriés par rapport aux résultats proposés dans ce mémoire.

2.4.2.1 Approches basées sur des CGEM

Tous les travaux qui suivent, sont basés sur des approches utilisant CGEM pour modéliser le PEI. Parmi les approches de commande, nous rappellerons celle proposé par (Guia, 1992), (Giua A. a.-C., 1993), (Giua & Seatzu, 2002), (Yamalidou, Moody, Lemmon, & Antsaklis, 1996), (Moody, Lemmon, & Antsaklis, 1996), (Moody & Antsaklis, 1998), (Basile F. , Recalde, Chiacchio, & Silva, 2000) et (Basile F. a., 2001), (Basile F. C., 2006), (Basile F. A., 2007) et (Basile F. , Recalde, Chiacchio, & Silva, 2009).

2.4.2.1.1 Approche de Giua et al, place mointeur

Considérons un système modélisé par un réseau de Petri (N, M_0) avec un ensemble de places P et une matrice d'incidence C. Le système est soumis à une spécification modélisé par une CGEM $(\vec{w}, k)$. Un moniteur qui permet de respecter cette contrainte est une place S à ajouter à N telle que le réseau résultant (N^s, M_0^s) a comme ensemble de places $P \cup \{S\}$ et une matrice d'incidence C_s définie par :

$$\boldsymbol{C_S = \begin{bmatrix} C \\ -W^t . C \end{bmatrix}} \qquad \textbf{(2.2)}$$

Le marquage initial de (N^s, M_0^s) est donné par :

$$\boldsymbol{M_{S0} = \begin{bmatrix} M_0 \\ k - w^t . M_0 \end{bmatrix}} \qquad \textbf{(2.3)}$$

Le marquage initial M_0 est supposé satisfaire la contrainte $(\vec{w}, k)$. Donc, cette place supplémentaire a des arcs sortants vers (respectivement entrants depuis) toutes les transitions d'entrée (respectivement de sortie) de toute place p du réseau appartenant au support de $\vec{w}$ et telle que $\vec{w}[p] > 0$.

Ce sens est inversé pour les places avec des poids négatifs. Les poids des arcs dépendent des coefficients de $\vec{w}$ et de la matrice d'incidence C. Un problème de SdC ou les spécifications sont modélisées par des CGEMs peut être donc résolu en ajoutant autant de moniteurs que de contraintes CGEM. Il est toutefois possible d'identifier a

priori les contraintes *redondantes* par rapport au réseau (i.e. toujours satisfaites) ou *équivalentes* entre elles (i.e. délimitant un même ensemble d'états admissibles).

Considérons le modèle de la Figure 2. 6, où toutes les transitions sont contrôlables. Supposons que les places p_2 et p_{22} matérialisent deux opérations qui se font sur la même machine et que cette machine peut exécuter une tâche à la fois. La contrainte de partage de la ressource machine peut se traduire par la condition suivante:

$$M(p_2)+M(p_{22})\leq 1$$

Il s'agit bien d'une contrainte de type CGEM avec $\vec{w}$=[0 1 0 0 1 0] et $k=1$.

Cette contrainte peut être satisfaite par une place supplémentaire S telle que $M_0(S)$=1 et $C_s(S)$=(-1 1 0 0 –1 1) comme il est montré dans la Figure 2. 7.

2.4.2.1.2 Approche basée sur des invariants de places

A la suite des travaux proposées dans (Guia, 1992), dans lequel les auteurs présentent une méthode de synthèse de superviseur maximum permissif, les auteurs de cette approche (Yamalidou, Moody, Lemmon, & Antsaklis, 1996), (Moody, Lemmon, & Antsaklis, 1996) ont développé un superviseur basé sur des invariants de places pour répondre à un ensemble des spécifications de type CGEM. Dans le principe, il s'agit d'ajouter des moniteurs (tels qu'ils sont définis par la section précédent) au système à contrôler pour construire des invariants de place. Un invariant de place *X* est un vecteur d'entiers, et il peut être calculé par l'équation $X^t.C=0$. La somme pondérée des marquages des places dans le support de *X* est alors toujours constante. Ainsi, à chaque contrainte $(\vec{w},k)$ est ajoutée une variable d'écart m_s telle que :

$$W^T.M \leq k \Rightarrow W^T.M + m_s = k$$

Cette variable d'écart représente un moniteur *S* qui va garantir que le produit $W^T.M$ ne dépassera jamais k. Considérons n_c le nombre de contraintes CGEM, C_s la matrice d'incidence des places moniteurs et M_{s_0} le vecteur de leurs marquages initiaux. Alors C_s doit satisfaire:

$$[\boldsymbol{W^t} \quad \boldsymbol{I}]\begin{bmatrix}\boldsymbol{C}\\ \boldsymbol{C_s}\end{bmatrix} = \boldsymbol{0} \qquad \textbf{(2.4)}$$

où $W = [w_1 \dots w_{mc}]$ est la matrice des contraintes CGEM.

La résolution de cette équation nous permet de décrire les attributs des arcs qui connectent les places moniteurs au réseau :

$$\boldsymbol{Cs = -W^t.C}$$

Le marquage initial des places de contrôle est donné par l'équation suivante:

$$\boldsymbol{M_{s0} = K - W^t . M_0} \qquad \textbf{(2.5)}$$

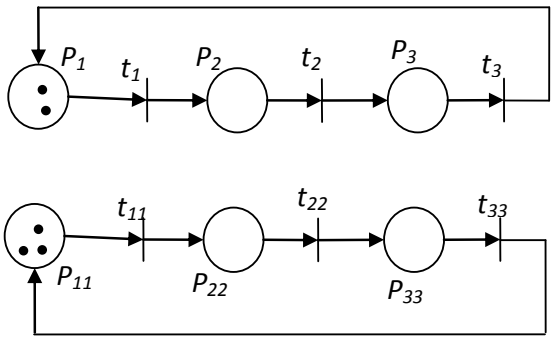


Figure 2. 6, ***modèle RdP à superviser***

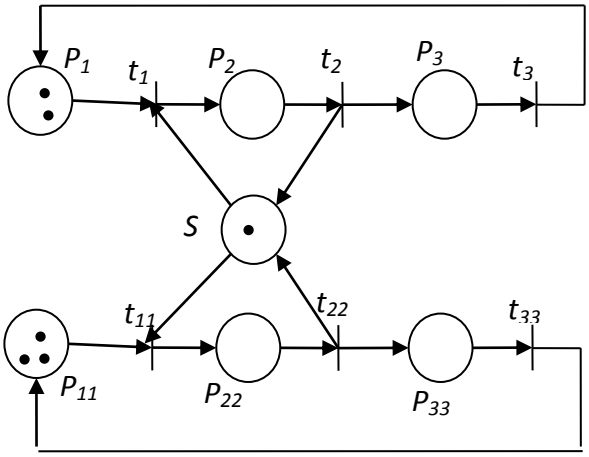


Figure 2. 7, ***modèle RdP supervisé***

Cette méthode est illustrée par le problème du chat et de la souris, voir Figure 2. 8, ce problème est présenté dans (Ramadge & Wonham, 1987- a), (Ramadge & Wonham, 1987-b) et (Ramadge & Wonham, 1989). Ce problème implique un labyrinthe de cinq chambres où le chat et la souris peuvent circuler. Les chambres se connectent par des portes par lesquelles les animaux peuvent passer.

L'objectif est de contrôler l'ouverture des portes pour que le chat et la souris ne se trouvent pas simultanément dans la même chambre. Le contrôleur doit être maximum permissif dans le sens où il doit garantir que les animaux aient le maximum de liberté dans leurs mouvements. Le modèle RdP de ce problème est montré dans Figure 2. 9. La matrice d'incidence du procédé est:

$$C = \begin{bmatrix} -1 & 0 & 1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \end{bmatrix}$$

La présence de jeton dans une place indique que l'animal correspondant est dans la chambre concernée par cette place. On suppose que le marquage initial est :

$$M_0 = [0\,0\,1\,0\,0\,0\,0\,0\,0\,1]$$

La spécification de ce problème est traduite par un ensemble de CGEMs :

$M(p_1)+M(p_6)\leq 1$

$M(p_2)+M(p_7)\leq 1$

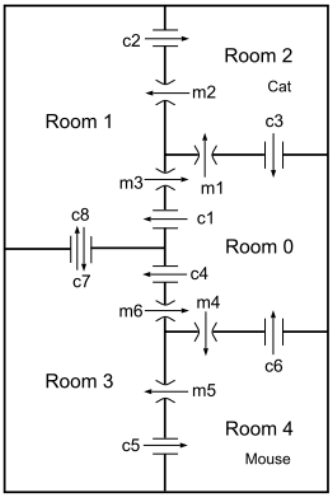


Figure 2. 8, ***le problème du chat et de la souris***

$M(p_3)+M(p_8)\leq 1$

$M(p_4)+M(p_9)\leq 1$

$M(p_5)+M(p_{10})\leq 1$

$$W = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$K = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$C_s = -W^T.C = \begin{bmatrix} 1 & 0 & -1 & 1 & 0 & -1 & 0 & 0 & 1 & 0 & -1 & 1 & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 1 & 0 & 0 & -1 & 1 & 0 \end{bmatrix}$$

Le marquage initial des places de contrôles est comme suit :

$$M_{s_0} = K - W^T.M_0 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Le modèle final supervisé montré dans Figure 2. 10, est développé dans (Yamalidou, Moody, Lemmon, & Antsaklis, 1996), (Moody, Lemmon, & Antsaklis, 1996).

L'ensemble des places moniteurs ainsi déterminé permet un contrôle maximum permissif. Le contrôleur (superviseur) fonctionne en parallèle avec le procédé où il surveille le procédé pour produire des actions en respectant le comportement désiré. Si la place de contrôle n'est pas marquée, alors la transition correspondante est bloquée.

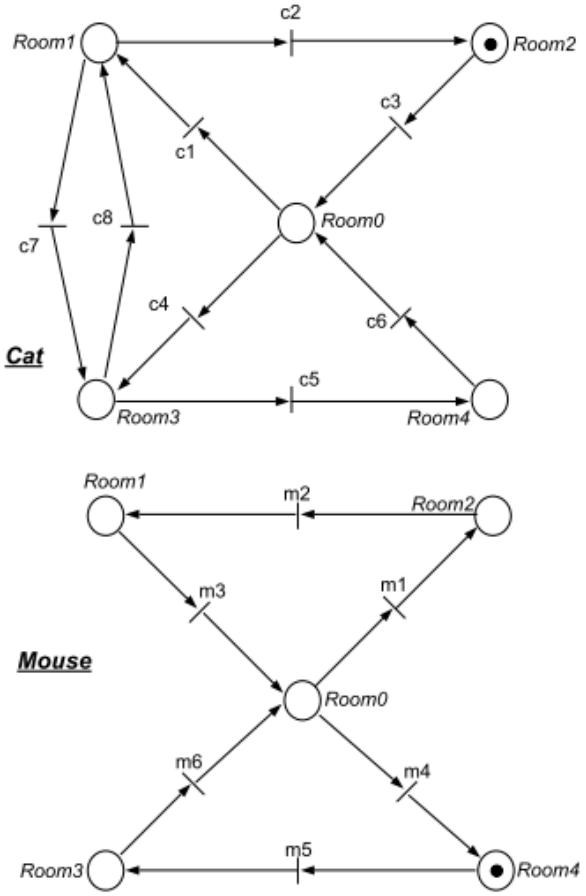


Figure 2. 9, ***Le modèle RdP du problème du chat et de la souris***

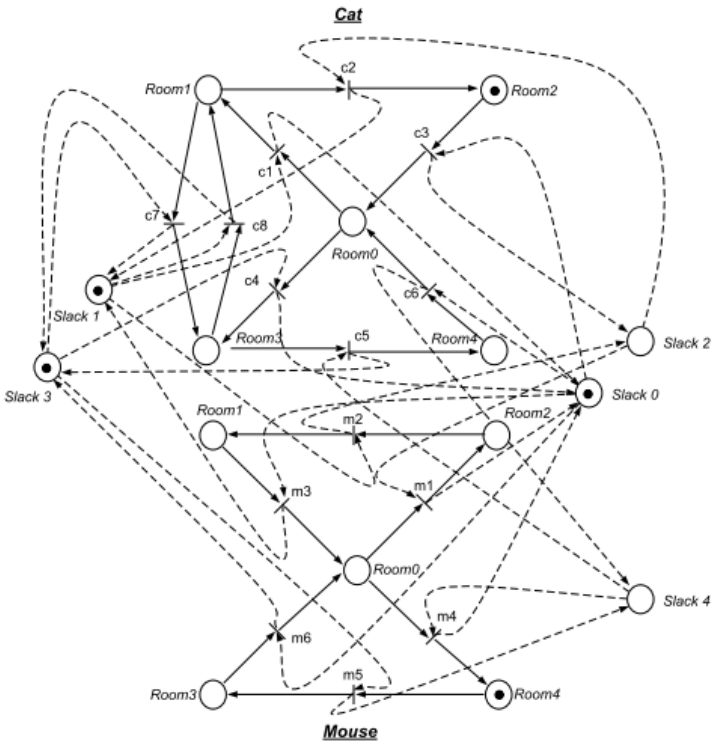


Figure 2. 10, ***Le modèle RdP supervisé du problème du chat et de la souris***

2.4.2.1.3 Approche basée sur les invariants de places et le problème d'incontrôlabilité

Le modèle proposé dans (Moody, Lemmon, & Antsaklis, 1996) ne traite pas le cas où il existe des événements incontrôlables. Ce problème est développé dans leur travail (Moody & Antsaklis, 2000).

En présence de transitions incontrôlables, un contrôleur *acceptable* ne doit pas tenter de les contrôler. Une condition suffisante pour avoir un contrôleur acceptable est la suivante:

$$W^T.C_s^u \leq 0$$

où C_s^u est la sous-matrice d'incidence représentant la partie incontrôlable. Autrement dit, il suffit qu'aucune des places du contrôleur n'ait d'arcs sortants vers les transitions incontrôlables du procédé (Moody & Antsaklis, 2000). Une contrainte $(\vec{w},k)$ qui peut être satisfaite par une place de contrôle qui ne contrôle pas les transitions non contrôlables est dite *admissible*. Dans le cas contraire, la contrainte $(\vec{w},k)$ doit être transformée en une contrainte $(\vec{w'},k')$ telle que:

- $(\vec{w'},k')$ est admissible;
- $(\vec{w},k)$ couvre $(\vec{w'},k')$, i.e. $W'^T.M \leq k' \Rightarrow W^T.M \leq k$

Les auteurs proposent la transformation des contraintes linéaires suivantes :

$$\vec{w'} = \vec{r_1} + r_2\vec{w}$$
$$k' = r_2(k+1)-1$$

où $\vec{r_1}$ est un vecteur de dimension $\text{card}(P)$ (le nombre des places du réseau non contrôlé) tel que $\vec{r_1}^T M \geq 0$ pour tout marquage atteignable M, et r_2 est un entier strictement positif.

$(\vec{w'},k')$ doit bien entendu satisfaire la condition d'admissibilité, i.e.

$$(\vec{r_1} + r_2\vec{w})^T C_s^u \leq 0$$

et de non-négativité du marquage initial, i.e.

$$\vec{r_1}^T M_0 + r_2(\vec{w}^T M_0 - k - 1) \leq -1.$$

$\vec{r_1}$ et r_2 peuvent donc être déterminés par la résolution du programme linéaire en nombres entiers.

2.4.2.1.4 Approche de Basile

L'approche de Basile (Basile F. , Recalde, Chiacchio, & Silva, 2000) synthétise un contrôleur (superviseur) RdP pour résoudre le problème d'états interdits avec des spécifications modélisées par des CGEMs. La méthode propose d'associer deux fonctions z_o et z_c à chaque transition, où le premier traduit le coût d'observation, pendant que le coût de contrôle est représenté par z_c. Les auteurs discutent deux cas :

1- ils ont considéré que les coûts dépendent du nombre d'arcs en amont et en aval de chaque transition. Par exemple, pour une place de contrôle qui a un arc en amont avec un poids $\omega_c^-(p,t)$, le coût de désactivation de cette transition est donné par $\omega_c^-(p,t)$ z_c, par contre si cette place a un arc en aval avec un poids $\omega_c^+(p,t)$, le coût qui désactive cette transition est donné par $\omega_c^+(p,t)$ z_o.

Ils ont prouvé que la politique de contrôle optimale peut être obtenue en résolvant un problème de programmation linéaire en nombre entier dont la fonction objectif est décrite en fonction de ces coûts.

2- les auteurs considèrent que les coûts d'observation et de contrôle associés à une transition représentent les coûts d'installation de matériels associés à cette transition (capteur, actionneurs).

Ensuite, ils ont prouvé que la politique de contrôle la plus optimale est obtenue en résolvant un problème de programmation entière avec une fonction objectif non linéaire décrite en utilisant ces coûts.

2.4.2.2 Approches utilisant la théorie des régions

La technique de synthèse de superviseur proposée dans (Badouel, Bernardinello, & Darondeau, 1995) et (Ghaffari A. a., 2002), se base sur la théorie des régions. En plus, les spécifications présentées sont de type CGEM.

Les auteurs de (Ghaffari A. a., 2002), présentent une méthode en utilisant la théorie des régions pour déterminer les places de contrôle, à ajouter au procédé pour restreindre son graphe d'accessibilité à un sous-ensemble de marquages admissibles en supprimant les marquages interdits. Dans cette approche, les auteurs se basent sur une modélisation du procédé par un RdP borné, ordinaire ou généralisé. Les spécifications sont de types PEI ou PTEI. Les auteurs déterminent formellement à l'aide de la théorie des régions un ensemble de places de contrôle à ajouter au modèle RdP du procédé. Cet ensemble de places de contrôle constitue le superviseur qui permet

d'assurer le respect des spécifications. Cette méthode consiste en deux phases principales :

Première phase : détermination de l'ensemble des marquages admissibles à travers les étapes suivantes:

1. Déterminer l'ensemble des marquages interdits
2. Générer le graphe des marquages partiels
3. Déterminer l'ensemble des marquages dangereux
4. Rechercher l'ensemble des marquages autorisés
5. Rechercher le comportement autorisé du procédé.

Deuxième phase : détermination des places de contrôle en utilisant la théorie des régions pour les ajouter au RdP. Pour déterminer ces places, les auteurs s'appuient sur l'équation de marquage et certaines propriétés d'un graphe fini comme suit :

1. équation de cycle
2. conditions d'atteignabilité d'un marquage quelconque depuis le marquage initial
3. conditions de séparations d'événements
4. conditions de séparations d'états.

Soient $R(N, M_0)$ le modèle RdP du procédé contrôlé et $R_c(N, M_0)$ le graphe de marquages admissibles respectant les spécifications. Pour déterminer l'ensemble des marquages admissibles les auteurs proposent l'algorithme suivant :

1. *Déterminer l'ensemble des marquages interdits.*
2. *Générer le graphe d'atteignabilité partiel* $R_p(N, M_0)$.
3. *Déterminer l'ensemble des marquages dangereux* M_D.
4. *Éliminer les nœuds appartenant à* M_D *du graphe* $R_p(N, M_0)$.
5. *Si R_c est co-atteignable, alors FIN. Sinon aller à 6.*
6. *Construire le plus grand sous-graphe co-atteignable* $CA(R)$.
7. *Calculer l'ensemble des marquages dangereux.*
8. *Éliminer de R_c les nœuds de* M_B. *Aller à 5.*

avec R le graphe ainsi obtenu, où le graphe d'atteignabilité partiel est un graphe d'atteignabilité où les successeurs des états interdits ne sont pas générés. On a noté par M_B l'ensemble de marquages bloquants.

La commande par supervision doit restreindre le comportement du système commandé au graphe R_c en inhibant toute transition contrôlable qui mène à un marquage bloquant ou interdit. Le graphe d'atteignabilité obtenu correspond au

comportement vivant maximal permissif du système commandé. Réciproquement, les contraintes d'états interdits et de vivacité sont satisfaites par tous les marquages de R_c, et aucun marquage ne mène de façon *incontrôlable* à l'extérieur de R_c. On en conclut que le graphe R_c obtenu constitue le comportement admissible vivant maximal.

Pour le problème de SdC, basée sur les RdP, les modèles RdP du système à contrôler et le graphe R_c sont donnés. La théorie des régions est utilisée pour déterminer les places de contrôle à ajouter au modèle RdP initial du procédé à contrôler.

Considérons une place de contrôle p_c. Pour le RdP du système à superviser augmenté de p_c, chaque marquage du graphe R_c doit rester atteignable. Ceci implique que p_c doit vérifier *les conditions d'atteignabilité* :

$$M_c(p_c) = M_0(p_c) + C_c(p_c, .)\vec{\Gamma}_M \geq 0 \quad \forall M \in R_c$$

où C_c, M_c et Γ_M sont respectivement la matrice d'incidence du réseau contrôlé, le marquage du réseau contrôlé et un chemin non-orienté de R reliant M_0 à M.

D'autres part, p_c doit satisfaire les équations de cycle correspondant aux cycles de R_c, donc l'application de la condition *d'atteignabilité* le long d'un cycle non orienté quelconque δ conduit à la relation suivante :

$$\sum_{t \in T} C_c(p_c, t)\vec{\delta}[t] = 0, \forall\, \delta \in \Delta.$$

où $\vec{\delta}[t]$ désigne la somme algébrique de toutes les occurrences de t dans δ *et* Δ l'ensemble des cycles d'élémentaires du graphe R_c. $\vec{\delta}$ est le vecteur d'occurrence de δ. Cette équation est appelée *l'équation de cycle.*

Quant aux événements à séparer, pour obtenir δ à partir du graphe de marquage initial, seules les transitions menant d'un marquage admissible à un marquage non admissible doivent être inhibées. Par conséquent, chaque nouvelle place p_c doit interdire une transition contrôlable t qui mène d'un marquage admissible M à un état interdit M'.

Cette interdiction se fait par l'ajout de la condition suivante appelée *condition de séparation* :

$$M'_c(p_c) = M_{c0}(p_c) + C_c(p_c, .)\vec{\Gamma}_M + C_c(p_c, t) < 0.$$

Ainsi, pour chaque transition contrôlable qui mène à un état interdit, on résout le système composé des équations précédentes pour déterminer une nouvelle place de contrôle p_c.

Les auteurs ont aussi montré qu'un comportement souhaité R, sous-graphe du graphe d'atteignabilité d'un RdP borné (N, M_0), peut être réalisé en ajoutant des places de contrôle pures à (N, M_0) si et seulement si il existe une solution $(M_{c0}(p_c), C_c(p_c, .))$ pour le système linéaire incluant toutes les conditions *d'atteignabilité (de cycle et de séparation* pour chaque transition à inhiber.

Comme il n'est pas toujours possible de déterminer un contrôleur RdP avec uniquement des places de contrôle pures, les auteurs ont aussi développé une approche permettant de déterminer un contrôleur RdP avec des places de contrôle impures. Cette approche est aussi basée sur la théorie des régions.

On considère l'exemple dans Figure 2. 11, et son graphe de marquage est montré dans Figure 2. 12, où on souhaite interdire le franchissement de la transition t_3 si la place p_4 est marquée.

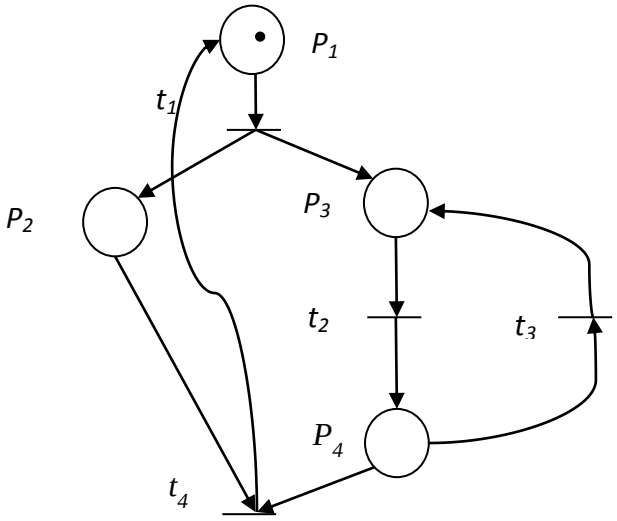


Figure 2. 11, ***Un RdP***

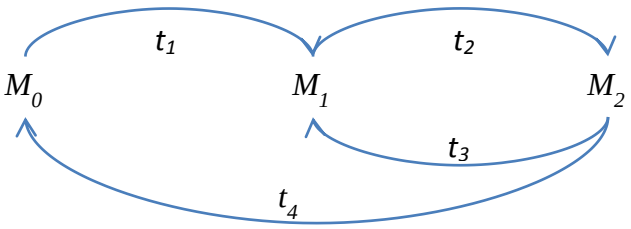


Figure 2. 12, ***Graphe de marquage correspondant au réseau de la Figure 2.11***

$$M_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \qquad M_1 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \qquad M_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

La solution du graphe de la Figure 2. 13 est comme suivante :

Conditions d'atteignabilité :

$M_0(p_c) \geq 0$

$M_1(p_c) \geq 0 \Rightarrow M_0(p_c) + C(p_c,t_1) \geq 0$

$M_2(p_c) \geq 0 \Rightarrow M_0(p_c) + C(p_c,t_1) + C(p_c,t_2) \geq 0$

LEéquations de cycles :

$C(p_c,t_1) + C(p_c,t_2) + C(p_c,t_4) = 0$

Condition de séparation :

$M_0(p_c) + C(p_c,t_1) + C(p_c,t_2) + C(p_c,t_3) < 0$

En trouvant une solution pour les précédentes conditions, on trouve la place de contrôle ayant :

$M_0(p_c)=1$, $C(p_c,t_1)=0$, $C(p_c,t_2)=-1$, $C(p_c,t_3)=-1$ et $C(p_c,t_4)=1$

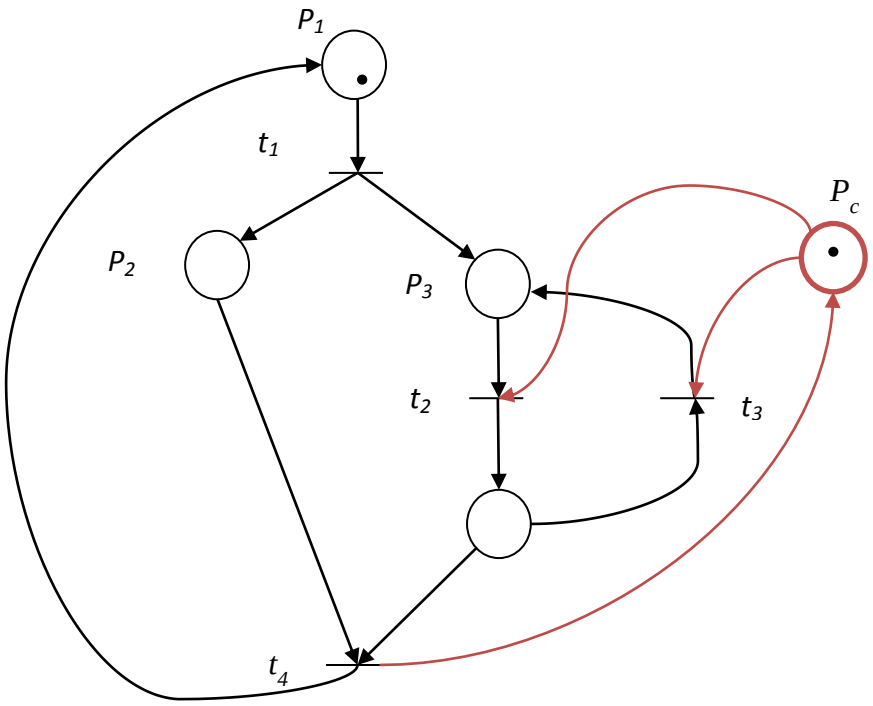


Figure 2. 13, ***Un RdP supervisé***

2.4.2.3 Synthétiser de superviseur basé sur des propriétés structurelles

On s'intéresse dans ce paragraphe aux approches donnant une synthèse de superviseur basée sur des propriétés structurelles des RdPs. L'approche de (Holloway & Krogh, 1990)et (Holloway, Guan, & Zhang, 1996) fait partie de cette catégorie des approches, lesquelles sont plus proches de notre SdC par supervision. Leur approche concerne les prédicats linéaires définis sur les modèles de type graphe d'événement sauf et vivant (cyclique). Boel *et al* ont proposé dans (Boel, Ben-Naoum, & Breusegem, 1995) une approche similaire qui concerne les graphes d'état. Une généralisation de ces approches aux classes de graphes d'événements et graphes d'états est proposée par (Holloway, Guan, & Zhang, 1996).

Holloway et son équipe ont présenté une technique de synthèse de contrôle (superviseur) maximal permissif en boucle fermée en garantissant qu'aucun des états interdits ne sont atteignables. Le contrôle se traduit en interdisant les transitions contrôlables quand il faut. Les mêmes auteurs ont développé une approche qui comprend les RdP non sauf (Holloway, Guan, & Zhang, 1996). Les inconvénients de cette approche sont que les expressions logiques associées aux transitions sont souvent complexes. La condition de franchissement d'une transition contrôlable correspond à la présence de jetons dans les places d'entrée et d'un jeton dans les places de contrôle ajoutée au réseau original. Ce rôle peut être interprété comme un commutateur binaire qui permet de passer ou non les jetons dans les places d'entrée aux places de sortie.

Par la suite, nous rappelons l'approche (Holloway & Krogh, 1990). Certains concepts présentés dans ces travaux seront utilisés par la technique de SdC que nous proposons dans ce mémoire.

Définition 2. 10: Un graphe d'événements sauf contrôlé cyclique est un réseau de Petri contrôlé *N : (P, T, A, U, E) où:*

1. Chaque place a une seule transition d'entrée et une seule transition de sortie.
2. Le poids de tous les arcs est égal à 1.
3. Le marquage des places est binaire
4. Chaque place est comprise dans un circuit élémentaire

L'ensemble des transitions est formé de deux sous-ensembles $T = T^u \cup T^c$:

- T^c l'ensemble des transitions contrôlables.
- T^u l'ensemble des transitions incontrôlables.

Les transitions contrôlables correspondent généralement à des commandes d'actionneur et ils sont représentées par des rectangles vides tandis que les transitions incontrôlables qui correspondent généralement à des signaux de capteur sont représentées par des barres.

Pour un réseau de Petri *N : (P, T, A, C)*, *R* (*N*, M_0) et $R^u(N, M_0)$ désigne respectivement l'ensemble des marquages atteignables partir du marquage initial et l'ensemble des marquages atteignables de façon incontrôlable partir du marquage initial.

Les auteurs proposent un contrôleur qui estime les prochains marquages atteignables et interdit le franchissement des transitions dont le franchissement mènera à un marquage interdit. Pour estimer les prochains marquages atteignables les auteurs proposent une méthode qui exploite les propriétés structurelles du graphe d'évènements contrôlé cyclique. En effet, dans un graphe d'évènement contrôlé cyclique, chaque place est précédée par une transition contrôlable. Ainsi le marquage des places concernées par les prédicats peut être contrôlé par les transitions contrôlables en amont. On parle alors de chemin d'influence.

Définition 2. 11: Un chemin d'influence γ_t est un chemin ($p_0, p_1, \ldots, p_n$) tel que:

1. $t = {}^\circ p_0$;
2. $\forall i \in]1, n]$, ${}^\circ p_i \notin T^c$;
3. L'une des deux conditions suivantes est satisfaite:
   - p_n° est une transition contrôlable,
   - il existe $0 \leq j \leq n$ tel que $p_n^\circ = {}^\circ p_j$.

Ainsi, le franchissement de la transition contrôlable *t* influe sur le marquage de toutes places dans le chemin γ_t.

Définition 2. 12: $\pi = (p_0, p_1, \ldots, p_n)$ est un chemin de précédence de *p* si :

1. π est un préfixe d'un chemin d'influence *de p*
2. et $p = p_n$.

Un chemin de précédence est compris dans un chemin d'influence. Notons t_π, la transition d'entrée d'un chemin de précédence π, par définition t_π est dite une transition contrôlable.

Le jeton apporté au chemin π par le franchissement de t_π atteindra inévitablement la place *p* si tous les autres chemins d'influence de *p* sont marqués. Par conséquent, connaissant le marquage de tous les chemins d'influence de *p*, il est possible de

connaitre le marquage atteignable par la place *p* de façon incontrôlable à partir d'un marquage *M*.

Pour un marquage *M* et un chemin de précédence π, soit le prédicat défini par :

$$\Lambda_{\pi}(M) = \begin{cases} 1 & \text{si } \exists p \in \pi, M(p) \neq 0 \\ 0 & \text{sinon} \end{cases}$$

Notons $\Pi_C(p)$, l'ensemble des chemins de précédence de la place p. Le théorème suivant a été démontré dans (Holloway & Krogh, 1990).

Théorème: Soit une place p d'un réseau N, le marquage $M \in R(N, M_0)$ est un marquage interdit si et seulement si

$$\Lambda_p(M) = \bigwedge_{\pi \in \Pi_p} \Lambda_{\pi}(M) = 1$$

où ⋀ est l'opérateur booléen de conjonction.

Par conséquent, le fonctionnement du contrôleur consiste à éviter d'atteindre le marquage M tel que $\Lambda_p(M) = 1$. De ce fait, le franchissement des transitions d'entrés des chemins d'influence est régularisé par les places de contrôle à l'entrée des transitions d'influence.

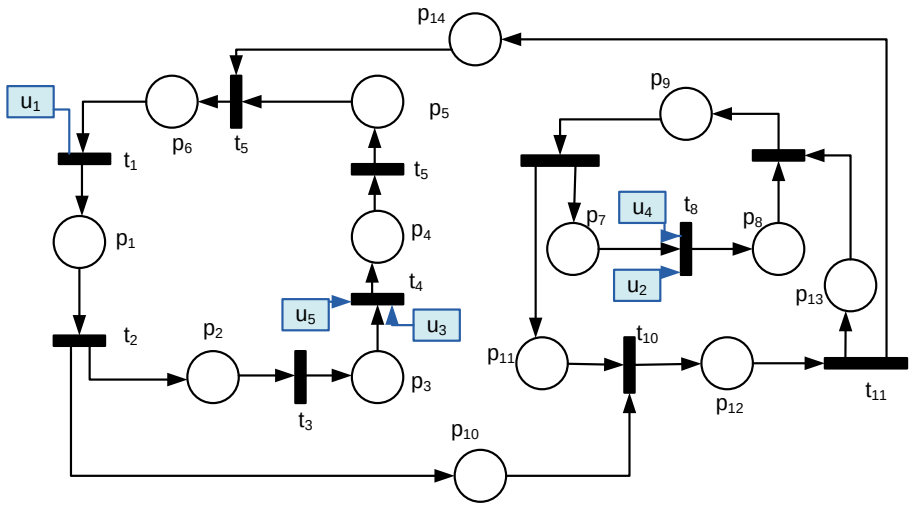


Figure 2. 14, ***Exemple d'application***

Exemple : Considérons l'exemple de la Figure 2. 14. Il s'agit d'un graphe d'évènements soumis à la spécification d'ensemble $F = \{p_5, p_{12}\}$ ainsi tout marquage M tel que $M(p_5) = M(p_{12}) = 1$ est interdit. Le marquage du réseau de la Figure 2. 14, pour p_5 on a $\Lambda_{p5}(M) = 1$ et $\Delta_{p5}(M) = 0$ et pour p_{12} on a $\Lambda_{p12}(M) = 0$, $\Delta_{p12}(M) = 1$ donc

$\Lambda_F(M) = 0$ et $\Delta_F(M) = 1$. Ainsi pour satisfaire la spécification *F*, $M(u_1) = M(u_3) = M(u_5) = 1$.

La complexité de cette approche a été discutée dans (Krogh & Holloway, 1991). Les auteurs ont montré que l'algorithme de synthèse est de complexité polynomial en nombre de transitions dans le graphe d'événement et en nombre de prédicats. Dans (Krogh & Holloway, 1991), il a également été prouvé qu'une spécification quelconque de type états interdits peut être transformée en un ensemble de prédicats linéaires. Cependant, le nombre de prédicats peut être exponentiel en la taille du réseau. Une généralisation de ces approches aux classes de graphes d'événements a été proposée par (Holloway, Guan, & Zhang, 1996).

2.4.3 Approches de SdC et existence de transitions inobservables

La commande par supervision des RdP partiellement observables a été étudiée dans (Giua & Seatzu, 2002), (Moody, Lemmon, & Antsaklis, 1996), (Moody & Antsaklis, 2000), (Stremersch, 2000), (Achour Z. a., 2004), et (Achour Z. a., 2008).

Les approches (Moody, Lemmon, & Antsaklis, 1996), (Moody & Antsaklis, 1998), (Moody & Antsaklis, 2000) et (Stremersch, 2000) présentent des approches algébriques intéressantes mais ne garantissent pas toujours l'optimalité. Notre approche dans ce chapitre propose une SdC avec un marquage initial connu contrairement à celle présentée dans (Giua & Seatzu, 2002), où les auteurs considèrent le cas où le marquage initial est inconnu et toutes les transitions sont observables.

La méthode proposée par (Achour, et al., 2005) présente une méthode en utilisant la théorie de régions, valable pour tout modèle RdP généralisé et borné. L'approche proposée construit dans un premier temps, le comportement admissible non bloquant maximum permissif, puis s'appuie sur l'utilisation de la théorie des régions pour la synthèse de contrôleurs RdP. L'ensemble des transitions est subdivisé en deux sous-ensembles disjoints : T_c l'ensemble des transitions contrôlables et T_{uc} celui des transitions incontrôlables. L'approche (Achour Z. a., 2004) est une extension de SdC proposée par (Ghaffari A. a., 2002).

Une technique différente, basée sur des propriétés structurelles des graphes d'événements est proposée dans (Achour Z. a., 2004), (Achour Z. a., 2008), où il se base sur le chemin de marquages.

2.4.3.1 Approche de Giua et Seatzu

L'approche de (Giua & Seatzu, 2002) développe un algorithme pour l'estimation de l'état d'un réseau dont le marquage ne peut pas être directement observé à cause des trois hypothèses suivantes :

- La structure du réseau *N* est connue.
- Toutes les transitions sont observables.
- Le marquage initial est inconnu.

L'algorithme d'estimation des marquages d'un réseau de places/transitions est basé sur l'observation des événements en utilisant un graphe de recouvrement observateur et un vecteur d'erreur d'estimation relatif à chaque place du réseau. Les auteurs ont aussi montré que cette estimation, générée par l'observateur, pouvait être utilisée pour la synthèse de contrôleurs. L'algorithme proposé par Giua et Seatzu interdit le franchissement des transitions menant en dehors du comportement admissible. Cet algorithme n'est pas nécessairement optimal dans le sens où une transition admissible dans le comportement désiré peut être inhibée.

L'ensemble $M(w)$ représente l'ensemble des états consistants par rapport à une séquence de tirs w observée. Formellement, $M(w) = \{M \mid \exists\, M', M'\ [w\rangle\ M\}$.

Dans la suite de cette partie, on note par M_w le marquage M tel que $M_0\ [w\rangle\ M$ et par $\vec{0}_m$ ($\vec{1}_m$) un vecteur $m \times 1$ de zéro (de *un*) où m est le nombre de place du réseau.

Pour une évolution du réseau donnée $M_{w0}\ [t_{\alpha 1}\rangle\ M_{w1}\ [t_{\alpha 2}\rangle$ …, l'algorithme suivant est utilisé pour déterminer le marquage estimé μ_{wi} relatif au marquage actuel M_{wi}. Cet algorithme est basé sur l'observation de la séquence de tirs $w_i = t_{\alpha 1}\, t_{\alpha 2} \ldots\, t_{\alpha i}$.

1. *Soit le marquage estimé initial* $\mu_{w0} = \vec{0}_m$ *et* $w = w_0$ (*séquence vide*).
2. *Soit* $i = 1$.
3. *Attendre le franchissement de* $t_{\alpha i}$
4. *Mettre à jour le marquage estimé* $\mu_{wi-1} = \mu'_{wi}$ *avec* $\mu'_{wi}(p) = max\{\mu_{wi-1}(p), C^{-}(p, t_{\alpha i})\}$.
5. *Soit* $\mu_{wi} = \mu'_{wi} + C(\ .\ , t_{\alpha i})$.
6. *Soit* $i = i + 1$.
7. *Aller en* 3.

FIN

Les auteurs ont établi un certain nombre d'hypothèses et définitions :

- Le marquage estimé est borné : $\mu_w = M_w$
- On définit l'erreur de place : $e_p(M_w, \mu_w) \equiv M_w(p) - \mu_w(p)$ noté aussi e_p.

- On définit l'erreur d'estimation $e(M_w, \mu_w) \equiv \vec{1} . (M_w - \mu_w)$

Les deux fonctions d'erreur ainsi définies, ne sont pas croissantes :

$e_p(M_{wt}, \mu_{wt}) = e_p(M_w, \mu'_{wt}) \leq e_p(M_w, \mu_w)$

$e(M_{wt}, \mu_{wt}) = e(M_w, \mu'_{wt}) \leq e(M_w, \mu_w)$

L'ensemble des marquages consistants par rapport à la séquence de tirs observée *w* est alors :

$M(w) = \{M \mid \exists\, M', M' [w\rangle M\} \equiv \{M \mid M \geq \mu_w\}$

Le graphe de recouvrement observateur est ensuite défini et représente simultanément l'ensemble des marquages atteignables du réseau avec l'erreur d'estimation en utilisant l'algorithme précédent. Le marquage initial est représenté par un rectangle avec des angles arrondis. Les marquages dont le vecteur d'erreur d'estimation = $\vec{0}_m$ sont représentés par des rectangles foncés.

Exemple : Sur la figure 2.15, un réseau de Petri est représenté ainsi que son graphe de recouvrement observateur. Chaque nœud de ce graphe représente le marquage réel M_w et l'erreur d'estimation $e_p(M_w, \mu_w) \equiv M_w - \mu_w$.

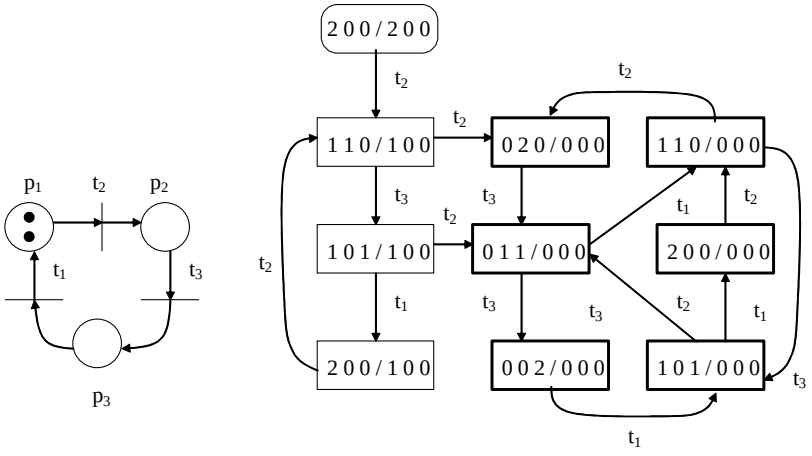


Figure 2. 15, ***Exemple d'un RdP et de son graphe de recouvrement observateur***

Dans une seconde partie, Giua et Seatzu introduisent la notion d'estimation de marquage avec macro-marquage. Dans ce cas, les auteurs supposent que le marquage $M-$ (dit marquage de départ) est connu. Après avoir évolué d'une manière inobservable, le réseau atteint un marquage M_0 (dit marquage initial) à partir duquel, le franchissement des transitions devient observable. Il est clair que $M_0 \in R(N, M-)$.

L'ensemble des marquages consistants par rapport à la séquence de tirs observée *w* devient alors :

$M(w \mid M\text{–}) = \{M \mid \exists\, M_0 \in R(N, M\text{–}),\ M_0\ [w\rangle\ M\}$.

Le macro-marquage est défini comme suit :

Définition 2. 13 : Considérons que l'ensemble des places *P* d'un réseau peut être écrit comme une union de *r*+1 sous-ensembles *:* $P = P_0 \cup P_1 \cup \ldots \cup P_r$ tel que $P_0 \cap P_1 = \varnothing$, $\forall\ j > 0$. Le nombre de jetons contenu dans P_j ($j > 0$) est connu et est égal à b_j. Cependant, le nombre de jetons dans P_0 est inconnu. Soit v_j le vecteur caractéristique relatif à chaque P_j, *c-à-d,* $v_j(p)=1$ *si* $p \in P_j$ *sinon* $v_j(p)=0$. *Soit* $V = [\vec{v}_1,\ldots,\vec{v}_r]$ *et* $\vec{b} = [b_1,\ldots,b_r]$. *Le* macro-marquage $V(V, \vec{b}) = \{M \mid V^T . M = \vec{b}\}$

Pour une évolution du réseau donnée $M_{w0}\ [t_{\alpha 1}\rangle\ M_{w1}\ [t_{\alpha 2}\rangle$ …, l'algorithme suivant décrit ci-après est utilisé pour déterminer le marquage estimé μ_{wi} ainsi que la borne B_{wi} relatif au marquage actuel M_{wi}. Cet algorithme est basé sur l'observation de la séquence de tirs $w_i = t_{\alpha 1}\, t_{\alpha 2} \ldots\, t_{\alpha i}$ et du macro-marquage initial $V(V, \vec{b})$.

1. *Soit le marquage estimé initial* $\mu_{w0} = \vec{0}_m$ *et* $w = w_0$ (*séquence vide*).
2. *Soit* $B_{w0} = \vec{b}$.
3. *Soit* $i = 1$.
4. *Attendre le franchissement de* $t_{\alpha i}$
5. *Mettre à jour le marquage estimé* $\mu_{wi-1} = \mu'_{wi}$ *avec* $\mu'_{wi}(p) = max\{\mu_{wi-1}(p), C^-(p, t_{\alpha i})\}$.
6. *Soit* $\mu_{wi} = \mu'_{wi} + C(\ .\ , t_{\alpha i})$.
7. *Soit* $B_{wi} = B_{wi-1} - V^T . (\mu'_{wi} - \mu_{wi-1})$.
8. *Soit* $i = i + 1$.
9. *Aller en* 4.

FIN

L'ensemble des marquages consistants par rapport à la séquence de tirs observée *w* devient alors :

$M(w \mid V, \vec{b}) = \{M \mid V^T . M = V^T . \mu_w + B_w,\ M \geq \mu_w\}$.

La politique de contrôle utilisée est définie ci-dessous :

Une transition *t* est inhibée après l'observation de la séquence de tirs *w*, si et seulement si il existe un marquage admissible *M* consistant par rapport à *w* tel que le franchissement de *t* à partir de *M* mène à un état interdit.

L'ensemble des états admissibles est défini par des Contraintes Généralisées d'Exclusion Mutuelle.

Exemple : Considérons le réseau de Petri de la Figure 2. 15 , avec un marquage initial $M_0 = [1,1,1]^t$ supposé inconnu. Soit l'ensemble des états interdits $F = \{M \mid M(p_1)>2\}$. Le macro-marquage initial $M(p_1) + M(p_2) + M(p_3) = 3$. La figure 1.10 représente le graphe de recouvrement observateur du réseau. Chaque nœud du graphe est noté par $(M/e/B)$ où M est le marquage réel, e est l'erreur d'estimation et B la borne résultante.

Pour des raisons de simplification, l'évolution du graphe de la Figure 2. 15 est arrêtée à chaque nœud où le vecteur d'erreur d'estimation = $\vec{0}_m$.

Les transitions en trait interrompu sont les transitions inhibées par le contrôleur. En réalité, ces transitions ont été inhibées parce qu'il existe un marquage consistant par rapport à l'observation et à partir duquel, le franchissement de la transition mène à un état interdit. Nous pouvons rapidement voir que dans tous ces nœuds :

$V^T . \mu + B = V^T . (M - e) + B = M(p_1) - e_{p1} + B = 3>2$ ce qui viole la condition $M(p_1) \leq 2$

2.4.3.2 Approche proposée par Achour

Achour a développé les travaux de Gaffari pour inclure le problème de l'observabilité (Achour Z. a., 2004), et en exploitant des propriétés structurelles des graphes d'événements, il a synthétisé une loi de commande pour le problème d'états interdits caractérisé par un ensemble de Contraintes Généralisées d'Exclusion Mutuelle (CGEM) avec la prise en compte de la contrainte d'observabilité des événements. Il a conçu une loi de commande basée sur la séparation de l'observation et de la commande, voir Figure 2. 16.

Le rôle de l'observateur est de caractériser tous les marquages courants possibles en se basant sur la séquence des transitions observables. Le rôle du superviseur est de déterminer les transitions contrôlables à interdire après le franchissement de toute transition observable. Le module de commande est activé suite à l'occurrence d'une transition observable selon la politique suivante :

- Déterminer tous les marquages courants possibles par l'observateur.
- Identifier les transitions contrôlables à interdire par le superviseur.

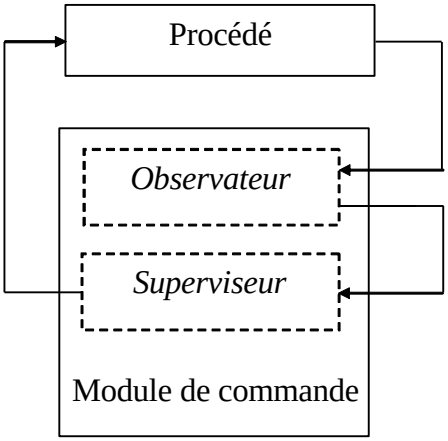


Figure 2. 16, ***La structure de la commande par supervision***

Définition 2. 14: Il a défini le vecteur *X* pour représenter le nombre de fois de franchissement d'une transition observable $t \in T_o$, ou le nombre minimal possible de fois de franchissement d'une transition inobservable $t \in T_{uo}$:

$$X(t) = \begin{cases} \sigma(t) & \forall t \in T_o \\ \left(max_{t' \in To}(\sigma(t') - d(M_0, t, t'))\right)^+ & \forall t \in T_{uo} \end{cases}$$

où

$(y)^+ = max\{0, y\}$.

La politique de commande optimale proposé dans son approche est donnée par l'algorithme ci-dessous :

1. Construire le réseau d'observation (N^o, M_0)
2. $M^o = M_0$
3. Calculer $G(w, M^o)$ pour toute *CGEM* (w, k). *Si* $G(w, M^o) > k$ pour au moins une *CGEM,* alors pas de solution de commande et Fin algorithme.
4. Pour toute transition contrôlable *t* franchissable dans le réseau d'observation,
   - *4.1.* Déterminer le marquage observateur $M^o({}^o t)$
   - *4.2*. Calculer $G(w, M^o({}^o t))$ pour toute CGEM (w, k)
   - *4.3.* Interdire *t* si $G(w, M^o({}^o t)) > k$ pour toute *CGEM* (w, k)
5. Attendre le franchissement d'une transition observable *t*
6. Mettre à jour le marquage du réseau d'observation ainsi que le marquage observateur M^o
7. Aller à l'étape *4.*

Des notions utilisées dans son approche, seront utilisées aussi dans notre approche, et seront présentées plus en détail dans les chapitres 4 et 5.

2.4.3.3 Autre travaux sur l'observabilité

L'observabilité est une propriété fondamentale des systèmes à événements discrets. L'estimation des états d'un procédé et la commande par supervision des systèmes partiellement observables sont les deux grandes thématiques liées à l'observabilité des transitions les plus étudiés dans la littérature.

Pour les systèmes modélisés par des automates à états finis, Ramadge (Ramadge, 1986) était le premier à montrer comment un observateur pouvait être déterminé pour les systèmes partiellement observables. Les auteurs de (Caines, Greiner, & Wang, 1988) ont montré qu'il est possible d'utiliser la séquence observée pour déterminer l'ensemble des états équivalents (non distinguables). L'approche est basée sur la construction d'un arbre d'observation contenant l'ensemble des marquages non distinguables. Özveren et Willsky (Ozveren & Willsky, 1990) ont proposé une approche pour la détermination de l'observateur qui permet de reconstruire un automate à états finis après l'observation d'une séquence finie d'événements. Özveren et Willsky ont aussi montré qu'un observateur peut avoir un nombre exponentiel d'états.

Li et Wonham (Li & Wonham, 1988) et (Li & Wonham, 1993) ainsi que Takai et *al.* L (Takai, Ushio, & Kodama, 1995) se sont intéressés au contrôle par retour d'état des systèmes partiellement observables. Les auteurs ont proposé des conditions nécessaires et suffisantes pour l'existence d'un contrôle par retour d'état optimal. Moody and Antsaklis (Moody & Antsaklis, 1998) et (Moody & Antsaklis, 2000) ont proposé une méthode de SdC pour les modèles réseaux de Petri avec des transitions incontrôlables et des transitions inobservables. Leur approche consiste à modifier l'ensemble des marquages admissibles quand ce dernier ne respecte pas les conditions d'incontrôlabilité et d'inobservabilité.

2.5 Conclusion

Dans ce chapitre, nous avons présenté deux axes principaux : le premier axe présente des modèles de spécifications, le deuxième axe présent des approches pour synthétiser la commande.

En général, on a vu que la problématique des spécifications peut être traduite en deux formes : le problème d'états interdits et le problème de transition d'état interdit.

La théorie de la supervision des SED a été initiée par Ramadge et Wonham, et elle est basée sur des concepts classiques du domaine de la commande des systèmes automatiques à variables continues. L'approche de Ramadge et Wonham se base sur les langages formels pour la modélisation et la synthèse de superviseurs de SED. L'objectif d'un problème de supervision est la synthèse d'une entité externe au système, appelé superviseur, capable de lui faire respecter un certain comportement

désiré en tenant compte de ses possibilités physiques. Le superviseur agit par inhibition d'événements dits contrôlables et doit tolérer tous ceux qui sont incontrôlables. Quand le but est réalisé en contraignant le comportement du système le minimum possible, le superviseur est alors dit maximum permissif.

Plusieurs approches se sont développées par la suite pour adapter les premiers résultats au contexte des réseaux de Petri. Les réseaux de Petri se distinguent par leur puissance de modélisation, la capacité des modèles et les nombreuses propriétés structurelles et comportementales qui facilitent l'analyse des modèles RdP. Mais, leur utilisation comme générateurs de langages n'est pas sans difficultés, surtout au niveau de la décidabilité de certaines propriétés comme la contrôllabilité.

L'utilisation des réseaux de Petri s'est révélée beaucoup plus avantageuse dans la résolution de problèmes de type états interdits. Leurs propriétés mathématiques et structurelles sont alors d'un grand intérêt pratique. Pour ces problèmes, l'objectif est de construire un contrôleur qui interdit au système contrôlé un ensemble d'états. Le contrôleur agit selon une politique de contrôle par retour d'état. Cela implique que la décision de contrôle dépend uniquement de l'état courant du système et non de l'historique de son évolution.

Nous avons classifié les approches qui utilisent les RdPs pour modéliser le système et le superviseur à trois catégories : La première regroupe les approches qui utilisent le CGEM et les invariants de places, la deuxième rassemble les approches basées sur la théorie des régions et la troisième concerne les approches exploitant les propriétés structurelles des RdPs. Egalement, nous avons catégorisé ces approches par rapport à la nature des événements en deux axes : le premier axe concerne la contrôlabilité et le deuxième axe l'observabilité.

Dans le chapitre suivant, nous allons proposer un nouveau modèle pour représenter le problème d'états interdits, et donnerons les motivations correspondantes.

Chapitre 3

Contraintes d'Exclusion de Marquage

Résumé :

Dans ce chapitre nous proposons un modèle de spécification qui interdit l'existence d'un nombre de jetons dans une place. Nous appelons ce modèle Contrainte d'Exclusion de Marquage CEM. Cependant pour plus d'une place critique, nous présentons d'autres extensions CEM-Où, CEM-Et et CEM-M. Nous proposons également une comparaison des contraintes d'exclusion de marquage avec des typologies de contraintes largement utilisés dans la littérature, tels que les Contraintes Généralisées d'Exclusion Mutuelle (Guia, 1992) et les contraintes de type prédicat (Holloway & Krogh, 1990). Les résultats de cette partie ont fait l'objet des publications (Atli, July 2010), (Atli, September 2010), (Atli, April 2010).

3.1 Introduction

Dans le chapitre précédent, nous avons exposé un état de l'art des techniques de SdC basées sur les RdP. Par la suite, nous présentons la première contribution de nos travaux de recherche, qui consiste dans la formalisation d'un modèle de contraintes, appelé *Contraintes d'Exclusion de Marquage*. Nous présentons d'abord le concept, les définitions et les extensions de ce modèle. Ensuite, nous comparons sa puissance de modélisation par rapport à des modèles de contraintes existant dans la littérature.

Nous organisons ce chapitre comme suit : dans la section 2, nous proposons notre modèle de contrainte CEM et ses variantes. Puis nous comparons ce modèle avec les autres dans la section 3. Un caractère très important est discuté dans section 4 pour diminuer le nombre de CEMs. Enfin, un exemple d'illustration est présenté dans la dernière section.

3.2 Problème d'état interdit

Nous montrons ici les motivations d'un nouveau modèle, et comment nous le représentons mathématiquement.

Le sens du mot *état* pour le procédé correspond au marquage dans le modèle RdP. Un état est défini par le nombre de jetons dans chaque place du RdP. L'idée est de traiter le marquage de chaque place individuellement pour décrire chaque état interdit.

Définition 3. 1: Le marquage d'un RdP est défini par le nombre de jetons existant dans les places à un moment donné. Ainsi ce marquage peut être représenté de manière logique par la conjonction des marquages de chaque place du réseau:

$$M \in R(N, M_0): M = M(p_1) \wedge \ldots \wedge M(p_m), M(p_i) = k_i \in \aleph$$

Dans l'exemple de la figure 3.1, l'état courant de ce réseau est défini comme suit :

$$M = \{M(p_1) = 1 \wedge M(p_2) = 2 \wedge M(p_3) = 0 \wedge M(p_4) = 3 \wedge M(p_5) = 2\}$$

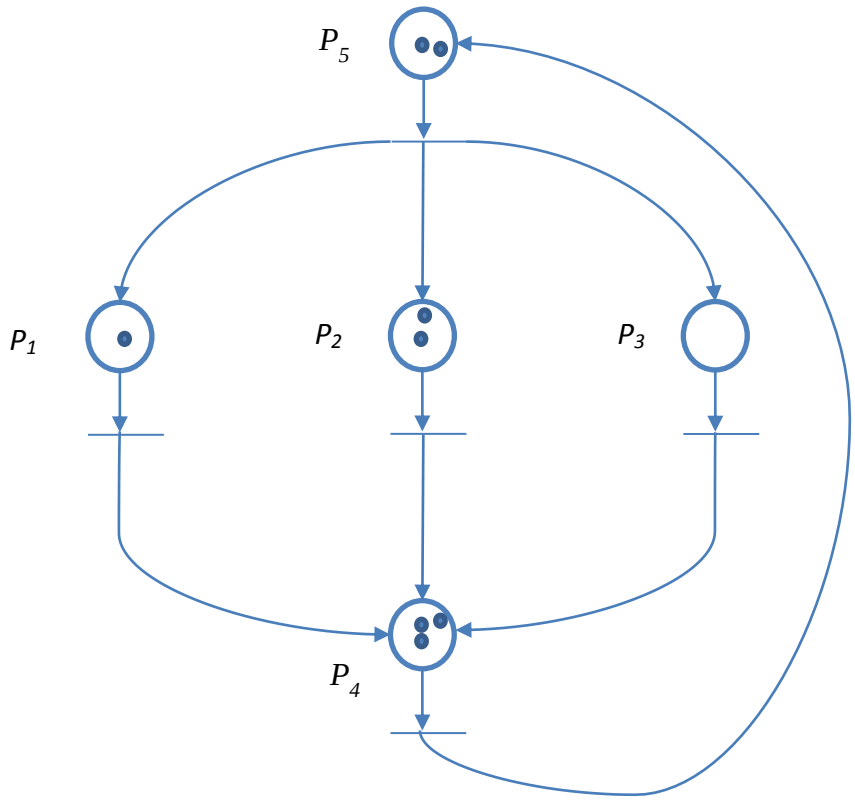


Figure 3. 1, ***Etat d'un procédé représenté par le marquage de chaque place regroupé par la relation logique Et***

3.3 Contrainte d'Exclusion de Marquage et ses extensions

Dans cette section, nous présentons notre modèle pour représenter le PEI. Premièrement nous proposons le modèle pour une spécification appliquée sur une place critique, ensuite nous étendons ce modèle à plusieurs places critiques.

3.3.1 Contrainte d'Exclusion de Marquage CEM

Définition 3. 2: Le nombre *k* pour une place critique *q* est un nombre entier de jetons, et appelé nombre propre de *q*.

Définition 3. 3: Une *Contrainte d'Exclusion de Marquage* CEM, est une condition concernant juste une place critique, où il faut exclure l'existence d'un nombre défini des jetons dans cette place.

Définition 3. 4 : L'ensemble des marquages interdits M_f décrit par une CEM pour une place critique est donné par:

$$\boldsymbol{M_f(q,k) = \{M \in R(N, M_0), q \in Q_{mc}: M(q) = k\}} \qquad \textbf{(3.1)}$$

Définition 3. 5: L'ensemble des marquages légaux décrit par *Contrainte d'Exclusion de Marquage* CEM est donné par l'expression suivante :

$$\boldsymbol{M_L(q,k) = \{M \in R(N,M_0), q \in Q_{mc}: M(q) \neq k\}} \quad \textbf{(3.2)}$$

Cet ensemble est l'union de deux sous-ensembles. Le premier *Up*(*q*,*k*) est l'ensemble de *marquages supérieurs* de la place critique *q,* par exemple le niveau de stock de sécurité, la demande minimale pour passer un ordre, etc.

$$\boldsymbol{Up(q,k) \equiv M(q) > k} \quad \textbf{(3.3)}$$

Le deuxième noté *Dn*(*q*) est l'ensemble de *marquages inférieurs* de la place critique *q*, par exemple : la capacité d'un stockage, la capacité de machine, etc.

$$\boldsymbol{Dn(q,k) \equiv M(q) < k} \quad \textbf{(3.4)}$$

$M_L(q,k) = Up(q,k) \cup Dn(q,k)$

$\forall M \in M_L(q,k) \Rightarrow M \in \{Up(q,k) \cup Dn(q,k)\} \Rightarrow$

$M \in \{M(q) > k\} \vee M \in \{M(q,k) < k\} \Rightarrow$

$M(q,k) \neq k \equiv M(q,k) > k \vee M(q,k) < k$

$$\boldsymbol{M(q,k) \neq k \equiv M(q,k) \geq k+1 \vee M(q,k) \leq k-1} \quad \textbf{(3.5)}$$

Propriété 3. 1: Pour un RdP ordinaire soumis à une CEM, *M(q,k)*≠*k*, si l'état initial est $M_0 \in Up(q,k)$, il est impossible que $M_0 \in Up(q,k)[\delta > M' \in Dn(q,k)$ sans passer par un marquage interdit ; où δ est une séquence de franchissements. Pareillement, si l'état initial est $M_0 \in Dn(q,k)$, il est impossible que $M_0 \in Dn(q,k)[\delta > M' \in Up(q,k)$ sans passer par un marquage interdit.

Propriété 3. 2: Pour un RdP soumis à une CEM, $M(q,k) \neq k$, $M_0 \in Up(q,k)$, si le poids d'un arc de sortie $\omega(q°)^i = \alpha > 1$, il est possible que $M_0 \in Up(q,k)[\delta > M' \in Dn(q,k)$ sans passer par un marquage interdit (voir Figure 3. 2).

Propriété 3. 3: Pour un RdP soumis à une CEM, $M(q,k) \neq k$, $M_0 \in Dn(q,k)$, si le poids d'un arc d'entrée $\omega(°q)^i = \alpha > 1$, il est possible que $M_0 \in Dn(q,k)[\delta > M' \in Up(q,k)$ sans passer par un marquage interdit (voir Figure 3. 3).

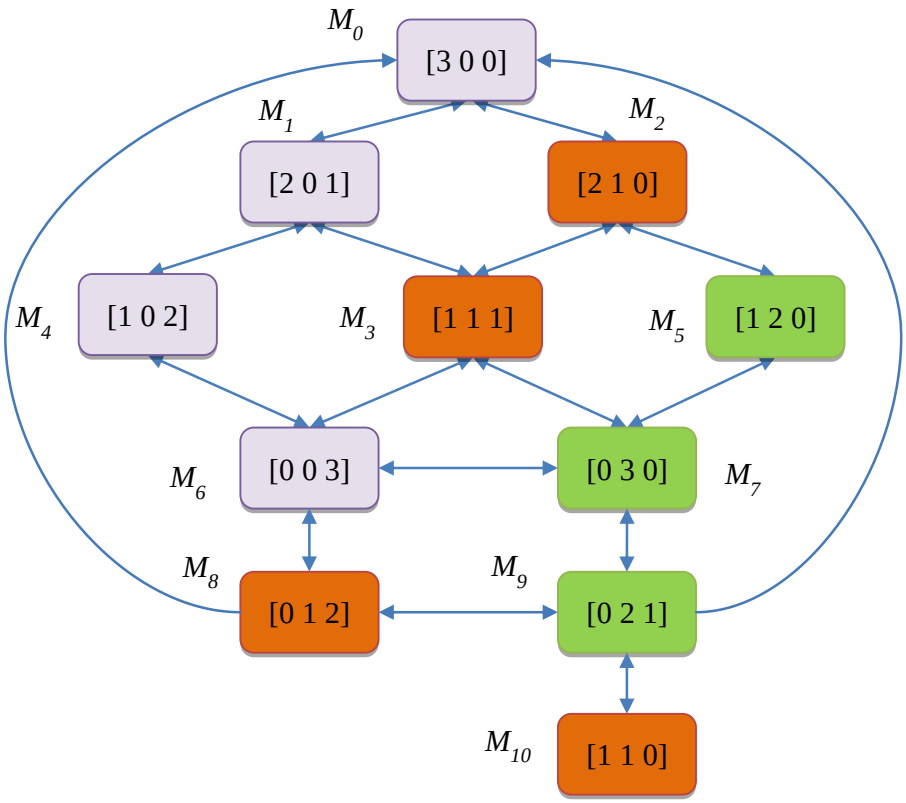


Figure 3. 2, ***Graphe des marquages avec une CEM : $M(p_1)\neq1$, $M_L=Up(p_1,1)\cup Dn(p_1,1)$, $Up(p_1,1)=\{M_5, M_7, M_9\}$, $Dn(p_1,1)=\{M_0, M_1, M_4, M_6\}$, $M_f=\{M_2, M_3, M_8, M_{10}\}$***

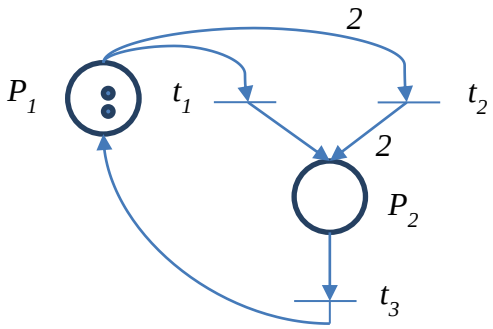


Figure 3. 3, ***Un RdP avec $M_0=[2\ 0]$***

On prend l'exemple dans la Figure 3. 3, si ce RdP est soumis à une CEM : $M(p_1)\neq1$, $M_0\in Up(p_1,1)$: cette place peut passer de $Up(p_1,1)$ à $Dn(p_1,1)$ sans passer par le marquage $M(p_1)=1$, $M_0\in Dn(p_1,1)[\sigma(t_2)>M'\in Up(p_1,1)$. Pareillement si ce réseau est soumis à une CEM : $M(p_2)\neq1$.

Pour plus d'une place critique, une CEM est une contrainte de base par laquelle nous modélisons le problème d'états interdits PEI. Trois extensions sont proposées : CEM-Ou, CEM-Et et CEM-M.

3.3.2 Contrainte d'Exclusion de Marquage de type Ou CEM-Ou

Définition 3. 6: La *Contrainte d'Exclusion de Marquage* de type *Ou*, est une condition concernant plus d'une place critique (deux ou plus), où il faut exclure les marquages pour lesquels chacune des places critiques q_i contient son nombre propre des jetons k_i.

Définition 3. 7 : Les marquages interdits M_f décrits par une CEM-Ou pour un ensemble des places critiques sont formalisés par :

$$\boldsymbol{M_{f(Ou)}(Q_{mc}, K) = \left\{ M \in R(N, M_0), q_i \in Q_{mc}: \bigwedge_{q_i \in Q_{mc}} M(q_i) = k_i \right\}}$$

où $K = \begin{bmatrix} k_1 \\ \dots \\ \dots \\ k_{mc} \end{bmatrix}$ est le vecteur de nombres propres.

Définition 3. 8: L'ensemble des marquages légaux décrit par une CEM-Ou est donné par l'expression suivante :

$$\boldsymbol{M_{L(Ou)}(Q_{mc}, K) = \left\{ \begin{array}{c} M \in R(N, M_0), q_i \in Q_{mc}: \bigvee_{q_i \in Q_{mc}} M(q_i) \neq k_i \\ M(q_i) \neq k_i \equiv M(q_i) \leq k_i - 1 \vee M(q_i) \geq k_i + 1 \end{array} \right\}}$$

Propriété 3. 4: Une CEM-Ou décrit exactement un marquage interdit si et seulement si le nombre de places critiques m_c est égale au nombre de places m (m_c=m).

Par exemple considérons le RdP donné dans la Figure 3. 2. CEM-Ou: $M(p_1)\neq 1 \vee M(p_2)\neq 1 \vee M(p_3)\neq 1$, désigne le marquage interdit $M_f=\{M_3\}$.

Propriété 3. 5 : Une CEM-Ou décrit un ensemble des marquages interdits si et seulement si le nombre de places critiques m_c est inférieur du nombre de places m (m_c<m).

Par exemple considérons le RdP dans la Figure 3. 2. CEM-Ou: $M(p_1)\neq 1 \vee M(p_2)\neq 1$, désigne les marquage interdits $M_f=\{M_3, M_{10}\}$.

Propriété 3. 6: Pour un RdP soumis à une CEM-Ou, tous les marquages pour lesquels il existe au minimum une place critique q_s ne contenant pas son nombre propre k_s, sont des marquages légaux.

Pour un RdP soumis à une CEM-Ou, tant qu'une place critique ne contient pas son nombre propre, chacune des autres places critiques peut garder un nombre de jetons égale à son nombre propre.

Par exemple : $M'=[M(q_1)=k_1 \dots M(q_s)\neq k_s \dots M(q_{mc})=k_{mc}]^t \notin M_f$.

Propriété 3. 7: Une CEM-Ou pour un ensemble des places critiques Q_{mc}, peut être représentée par un ensemble de contraintes linéaires liées par la relation logique *Ou*, le nombre de ces contraintes Linéaires est dénoté N où $N=2*m_c$.

Preuve :

Chaque place $q \in Q_{mc}$ est soumise à une CEM $\Rightarrow$ le nombre de CEM égale à m_c et chaque CEM est représentée par deux contraintes Linéaires $\Rightarrow$ le nombre total de contraintes Linéaires N égale à $2*m_c$.

3.3.3 Contrainte d'Exclusion de Marquage de type Et CEM-Et

Définition 3. 9: La *Contrainte d'Exclusion de Marquage* de type *Et*, est une condition concernant plus d'une place critique (deux ou plus), où il faut exclure tous les marquages pour lesquels au minimum une des places critiques q_i contient son nombre propre des jetons k_i.

Définition 3. 10 : Les marquages interdits M_f décrits par une CEM-Et pour un ensemble des places critiques sont formalisés :

$$\boldsymbol{M_{f(Et)}(Q_{mc}, K) = \left\{ M \in R(N, M_0), q_i \in Q_{mc} : \bigvee_{q_i \in Q_{mc}} M(q_i) = k_i \right\}}$$

Définition 3. 11: L'ensemble des marquages légaux décrit par une CEM-Et est donné par l'expression suivante:

$$\boldsymbol{M_{L(Et)}(Q_{mc}, K) = \left\{ \begin{array}{c} M \in R(N, M_0), q_i \in Q_{mc} : \bigwedge_{q_i \in Q_{mc}} M(q_i) \neq k_i \\ M(q_i) \neq k_i \equiv M(q_i) = M(q_i) \leq k_i - 1 \vee M(q_i) \geq k_i + 1 \end{array} \right\}}$$

Propriété 3. 8: Une CEM-Et pour m_c des places critiques Q_{mc}, peut être représentée par un ensemble de sous-ensembles de contraintes d'inégalité liées par la relation logique *Ou*, le nombre de ces sous-ensembles est 2^{mc}. Chacun de ces sous-ensembles se compose de m_c contraintes d'inégalité liées par la relation logique *Et*. Ainsi le nombre total de contraintes d'inégalité pour représenter une CEM-Et avec m_c des places critiques est : $N= m_c*2^{mc}$.

Preuve :

$$M_{L(Ou)}(Q_{mc}, K) = \bigwedge_{q_i \in Q_{mc}} M(q_i, k_i) \neq k_i \equiv$$

$M_{(q_1)} \neq k_1 \wedge \ldots \wedge M_{(q_i)} \neq k_i \wedge \ldots \wedge M_{(q_{mc})} \neq k_{mc} \equiv$

$(M_{(q_1)} \leq k_1 - 1 \vee M_{(q_1)} \geq k_1 + 1) \wedge \ldots \wedge (M_{(q_{mc})} \leq k_{mc} - 1 \vee M_{(q_{mc})} \geq k_{mc} + 1) \equiv$

$(M_{(q_1)} \leq k_1 - 1 \wedge \ldots \wedge M_{(q_i)} \leq k_i - 1 \wedge \ldots \wedge M_{(q_{mc})} \leq k_{mc} - 1) \vee$

……..

$(M_{(q_1)} \geq k_1 + 1 \wedge \ldots \wedge M_{(q_i)} \geq k_i + 1 \wedge \ldots \wedge M_{(q_{mc})} \geq k_{mc} + 1)$

□

Une CEM-Et peut être représentée par une matrice où :

1. le nombre de colonnes égale à m_c
2. le nombre de lignes égale à 2^{mc}
3. la relation entre les éléments de chaque ligne est la relation logique *Et*
4. la relation entre les lignes est la relation logique *Ou* .

Propriété 3. 9: L'ensemble des marquages légaux décrits par une CEM-Où pour un ensemble de places critiques Q_{mc} et un vecteur de nombres propres *K* contenant l'ensemble des marquages légaux décrit par une CEM-Et pour le même Q_{mc} et le même vecteur de nombres propres *K*.

Preuve :

$\forall q_i \in Q_{mc}$:

$$M_{L(Ou)}(Q_{mc}, K) = \left\{ \bigvee_{q_i \in Q_{mc}} M(q_i) \neq k_i \right\} = \{M(q_1) \neq k_1 \vee \ldots \vee M(q_{mc}) \neq k_{mc}\}$$

$$M_{L(Et)}(Q_{mc}, K) = \left\{ \bigwedge_{q_i \in Q_{mc}} M(q_i) \neq k_i \right\} = \{M(q_1) \neq k_1 \wedge \ldots \wedge M(q_{mc}) \neq k_{mc}\}$$

$\Rightarrow M_{L(Et)}(Q_{mc}, K) \subseteq M_{L(Ou)}(Q_{mc}, K)$

□

Ce type de contrainte est utile par exemple pour représenter la résonance des éléments électriques ou mécaniques. Où il est interdit que la fréquence de cet élément soit égale à sa fréquence de résonance pour éviter l'effondrement de cet élément (tels que ce qui s'est passé pour la Basse-Chaîne à Angers, France (1850); le Pont égyptien (Saint-Pétersbourg, 1905) et le pont de Takoma (USA, 1940)).

Considérons l'exemple dans la Figure 3. 4, où un sous-graphe d'événements représente la situation actuelle de la fréquence d'un élément ou système. Un jeton représente une unité de fréquence. Les jetons dans la place p_1 représente la fréquence actuelle de cet élément, alors la spécification se traduit en interdisant que le nombre des jetons n'égale pas à la fréquence de résonance de cet élément.

La contrainte CEM-Et qui est décrite comme suivante, modélise cette spécification :

$$M_L(Q_{mc}, K) = \{M(p_1) \neq 1 \wedge M(p_2) \neq 2 \wedge M(p_3) \neq 0 \wedge M(p_4) \neq 3\}$$

On remarque que la fréquence de chaque élément peut être inférieure ou supérieure de sa fréquence de résonance propre.

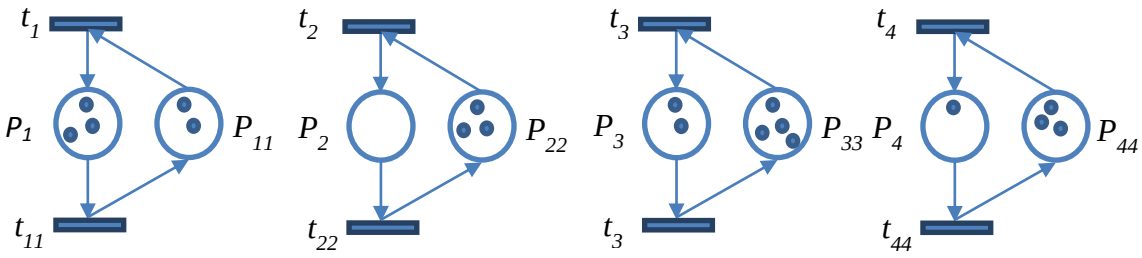


Figure 3. 4, ***Exemple de modéliser la fréquence d'éléments***

3.3.4 Contrainte d'Exclusion de Marquage Mutuelle CEM-M

Définition 3. 12: La *Contrainte d'Exclusion de Marquage Mutuelle*, est une condition concernant plus d'une place critique (deux ou plus), où la somme de jetons dans ces places ne doit pas égaler le nombre *k*.

Définition 3. 13 : Les marquages interdits M_f décrits par une CEM-Et pour un ensemble des places critiques sont formalisés par :

$$\boldsymbol{M_{f(CEM-M)}(Q_{mc}, k) = \left\{ M \in R(N, M_0), q_i \in Q_{mc} : \sum_{i=1}^{mc} M(q_i) = k \right\}}$$

Définition 3. 14: L'ensemble des marquages légaux décrit par une CEM-M est donné par l'expression suivante:

$$\boldsymbol{M_{L(CEM-M)}(Q_{mc}, k)} = \left\{ \begin{array}{c} \boldsymbol{M \in R(N, M_0), q_i \in Q_{mc}: \sum_{i=1}^{mc} M(q_i) \neq k} \\ \boldsymbol{\sum_{i=1}^{mc} M(q_i) \neq k \equiv \sum_{i=1}^{mc} M(q_i) \leq k-1 \vee \sum_{i=1}^{mc} M(q_i) \geq k+1} \end{array} \right\}$$

Propriété 3. 10: Une CEM-M consiste en deux extensions. La première est de type "*inférieur de*" (elle est appelée *la somme mutuelle de marquages inférieurs SMMI*(Q_{mc}, *k*)). La deuxième est de type "*supérieur de*" (elle est appelée *la somme mutuelle de marquages supérieurs SMMS*(Q_{mc}, *k*)).

3.3.5 Classe de contraintes de CEM C-CEM

Définition 3. 15 : Une Classe de contraintes est un ensemble de contraintes appliquées sur un système.

Une *Classe* de contraintes est satisfaite si toutes les contraintes sont satisfaites.

3.4 Puissance de modélisation de la Contrainte d'Exclusion de Marquage

Dans cette section, nous comparons la puissance de la CEM avec les deux modèles proposés par (Holloway & Krogh, 1990) et (Guia, 1992). Contrairement à toutes les deux approches, notre modèle est applicable sur tous les RdPs, sauf ou non, cyclique ou non.

3.4.1 CEM versus contraintes de type prédicat

Nous comparons notre approche à celle proposée par (Holloway & Krogh, 1990).

- Une *condition de place* (définition 2.5), interdit juste la place concernée d'être marquée. Également une CEM est appliqué juste sur une place, par contre la CEM interdit la place critique d'avoir *k* jetons. Autrement dit, la *condition de place* est un cas particulier de CEM où *k*=1. Nous pouvons la ré-formaliser en utilisant la définition CEM :

$$M_L(q, 1) = \{M \in R(N, M_0), M(q) \neq 1\}$$

- Pareillement, *la condition de type ensemble* (définition 2.6), modélise un marquage interdit où toutes les places critiques sont marquées. Encore, la CEM-Ou définit un marquage interdit pour lequel simultanément chaque place critique q_i contient son nombre propre des jetons k_i. Ainsi, *la condition de type ensemble* est un cas particulier de CEM-Ou sous une seule condition

où tous les nombres propres k_1= k_2=…= k_i=…= k_{mc}=1. L'*ensemble de conditions* peuvent être formalisé sous la forme de CEM-Ou :

$$M_L(Q_{mc}, K) = \left\{ M \in R(N, M_0), q_i \in Q_{mc} : \bigvee_{q_i \in Q_{mc}} M(q_i) \neq 1 \right\}$$

$$K = \begin{bmatrix} 1 \\ \dots \\ \dots \\ 1 \end{bmatrix}$$

3.4.2 CEM versus les contraintes linéaires

- Chaque CEM peut être modélisée par deux types de contraintes Linéaires (*supérieur* de ou *inférieur de*) reliées par la relation *Ou*.
- Chaque CEM-Ou sur m_c places critiques, peut être représentée par 2*m_c des contraintes Linéaires (*supérieur* de ou *inférieur de*) reliées par la relation *Ou*, voir la propriété 3.7.
- Chaque CEM-Et sur m_c places critiques, peut être représentée par m_c*2^{mc} des contraintes Linéaires (*supérieur* de ou *inférieur de*) reliées par la relation *Ou* et *Et*, voir la propriété 3.8.

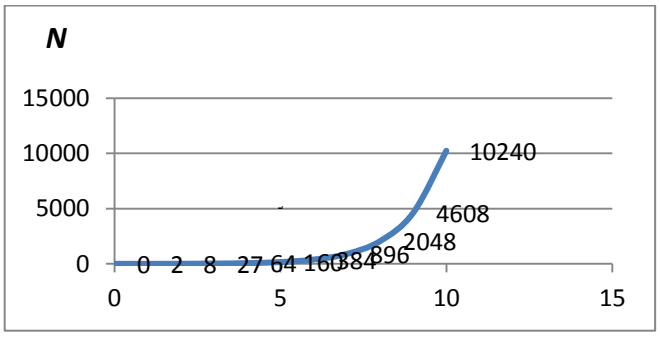


Figure 3. 5, ***Relation entre CEM-Et et les contraintes Linéaires en fonction du nombre des places critiques***

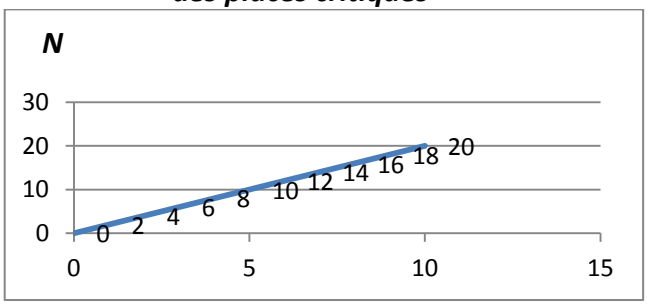


Figure 3. 6, ***Relation entre CEM-Ou et les contraintes Linéaires en fonction du nombre des places critiques***

3.4.3 CEM versus CGEM

Afin de trouver une approche entre la CEM et la CGEM proposée par (Guia, 1992), nous les comparons comme suit :

- En général, la CGEM ne permet pas de modéliser l'ensemble des spécifications sur le marquage de places lorsque le réseau n'est pas sauf et conservative. Compte tenu de la granularité des contraintes CEM, elles ont une puissance de modélisation plus importante (voir l'exemple proposé à la fin de ce chapitre).

- Puisque la CGEM est considérée comme un type des contraintes Linéaires, donc les points de comparaison dans la section précédente sont considérés aussi comme des points de comparaison entre CEM et CGEM.

- *Une CGEM modélise l'ensemble* SMMI(Q_{mc}, k).

3.5 Déterminer la classe minimale

Dans cette section, nous illustrons un caractère important, qui consiste à minimiser le nombre de contraintes dans une classe de CEM en utilisant les caractéristiques algébriques. Nous utilisons cette propriété pour réduire le nombre de contraintes, lorsque les nombres propres des places critiques ont certaines valeurs particulières.

Propriété 3. 11: Pour un RdP et $q_s \in Q_{mc}$ soumis à une classe des contraintes où :

$$\bigwedge_{q_s \in Qmc} M(q_s) \neq k_j, j \in \{1, 2, 3, \dots, J\}$$

Si $k_1 = k_2 - 1 = k_3 - 2 = \dots = k_{J-1} - (J-2) = k_J - (J-1)$,

la classe équivalente de cette place critique q_s est:

$$\boldsymbol{M_L(q_s, K) = \{M \in R(N, M_0), M(q_s) \leq k_1 - 1 \vee M(q_s) \geq k_J + 1\}}$$

$$\boldsymbol{K = \begin{bmatrix} k_1 \\ \dots \\ k_j \\ \dots \\ k_J \end{bmatrix}}$$

Preuve:

$M(q_s, K) \in M_L$: $M_L = \{[0, \infty[\setminus \{k_1, k_2, \dots, k_i, \dots, k_J\}\} = \{[0, \infty[\setminus \{k_1, k_1+1, \dots, k_1+(J-1)\}\} = \{[0, \infty[\setminus [k_1, k_1+(J-1)]\} = \{0 \leq M(q_s) \leq k_1 - 1\} \vee \{k_j + 1 \leq M(q_s) < \infty\} = M(q_s) \leq k_1 - 1 \vee M(q_s) \geq k_1 + 1$.

Propriété 3. 12: Considérons qu'un ensemble de places critiques a les mêmes nombres propres par rapport à plusieurs CEM tels que : $Q_{mc(1)} \supseteq \ldots \supseteq Q_{mc(j)} \supseteq \ldots \supseteq Q_{mc(J)}$, et $K_1 \supseteq \ldots \supseteq K_j \supseteq \ldots \supseteq K_J$ sont soumis à une classe de contraintes $M_f = M_{f(1)} \wedge \ldots \wedge M_{f(j)} \wedge \ldots \wedge M_{f(J)}$, où :

$$M_f(Q_{mc(1)}, K_1) = \bigwedge_{q_i \in Q_{mc(1)}} M(q_i) = k_i$$

$$M_f(Q_{mc(2)}, K_2) = \bigwedge_{q_i \in Q_{mc(2)}} M(q_i) = k_i$$

……

$$M_f(Q_{mc(j)}, K_j) = \bigwedge_{q_i \in Q_{mc(j)}} M(q_i) = k_i$$

……

$$M_f(Q_{mc(J)}, K_J) = \bigwedge_{q_i \in Q_{mc(J)}} M(q_i) = k_i$$

Alors $M_f(Q_{mc(1)}, K_1) \equiv M_f$.

Preuve:

$$M_f = \left\{ \bigwedge_{q_i \in Q_{mc(1)}} M(q_i) = k_i \right\} \wedge \ldots \wedge \left\{ \bigwedge_{q_i \in Q_{mc(j)}} M(q_i) = k_i \right\} \wedge \ldots \wedge \left\{ \bigwedge_{q_i \in Q_{mc(J)}} M(q_i) = k_i \right\}$$

$M_f = \{M(q_1)=k_1 \wedge M(q_2)=k_2 \wedge \ldots \wedge M(q_{mc(J)})=k_{mc(J)} \wedge \ldots \wedge M(q_{mc(j)})=k_{mc(j)} \wedge \ldots \wedge M(q_{mc(1)})=k_{mc(1)}\}$

$\wedge \ldots \wedge$

$\{M(q_1)=k_1 \wedge M(q_2)=k_2 \wedge \ldots \wedge M(q_{mc(J)})=k_{mc(J)} \wedge \ldots \wedge M(q_{mc(j)})= k_{mc(j)}\}$

$\wedge \ldots \wedge$

$\{M(q_1)=k_1 \wedge M(q_2)=k_2 \wedge \ldots \wedge M(q_{mc(J)})=k_{mc(J)}\} \equiv$

$\{M(q_1)=k_1 \wedge M(q_2)=k_2 \wedge \ldots \wedge M(q_{mc(J)})=k_{mc(J)} \wedge \ldots \wedge M(q_{mc(j)})= k_{mc(j)} \wedge \ldots \wedge M(q_{mc(1)})=k_{mc(1)}\}$

$$\equiv \left\{ \bigwedge_{q_i \in Q_{mc(1)}} M(q_i) = k_i \right\} = M_{f(1)}(Q_{mc(1)}, K_1)$$

□

Propriété 3. 13: Considérons qu'un ensemble de places critiques a les mêmes nombres propres par rapport plusieurs CEM, tels que : $Q_{mc(1)} \supseteq \ldots \supseteq Q_{mc(j)} \supseteq \ldots \supseteq Q_{mc(J)}$ et $K_1 \supseteq \ldots \supseteq K_j \supseteq \ldots \supseteq K_J$, sont soumis à une classe de contraintes $M_f = M_{f(1)} \wedge \ldots \wedge M_{f(j)} \wedge \ldots \wedge M_{f(J)}$, où :

$M_f(Q_{mc(1)}, K_1) = \bigvee_{q_i \in Q_{mc(1)}} M(q_i) = k_i$

$M_f(Q_{mc(2)}, K_2) = \bigvee_{q_i \in Q_{mc(2)}} M(q_i) = k_i$

……

$M_f(Q_{mc(j)}, K_j) = \bigvee_{q_i \in Q_{mc(j)}} M(q_i) = k_i$

……

$M_f(Q_{mc(J)}, K_J) = \bigvee_{q_i \in Q_{mc(J)}} M(q_i) = k_i$

Alors $M_f(Q_{mc(1)}, K_1) \equiv M_f$.

Preuve:

$$M_f = \left\{ \bigvee_{q_i \in Q_{mc(1)}} M(q_i) = k_i \right\} \wedge \ldots \wedge \left\{ \bigvee_{q_i \in Q_{mc(j)}} M(q_i) = k_i \right\} \wedge \ldots \wedge \left\{ \bigvee_{q_i \in Q_{mc(J)}} M(q_i) = k_i \right\}$$

$M_f = \{M(q_1)=k_1 \vee M(q_2)=k_2 \vee \ldots \vee M(q_{mc(J)})=k_{mc(J)} \vee \ldots \vee M(q_{mc(j)})=k_{mc(j)} \vee \ldots \vee M(q_{mc(1)})=k_{mc(1)}\} \wedge \ldots \wedge$

$\{M(q_1)=k_1 \vee M(q_2)=k_2 \vee \ldots \vee M(q_{mc(J)})=k_{mc(J)} \vee \ldots \vee M(q_{mc(j)})= k_{mc(j)}\}$

$\wedge \ldots \wedge$

$\{M(q_1)=k_1 \vee M(q_2)=k_2 \vee \ldots \vee M(q_{mc(J)})=k_{mc(J)}\} \equiv$

$\{M(q_1)=k_1 \vee M(q_2)=k_2 \vee \ldots \vee M(q_{mc(J)})=k_{mc(J)} \vee \ldots \vee M(q_{mc(j)})= k_{mc(j)} \vee \ldots \vee M(q_{mc(1)})=k_{mc(1)}\}$

$$\equiv \left\{ \bigvee_{q_i \in Q_{mc(1)}} M(q_i) = k_i \right\} = M_f(Q_{mc(1)}, K_1)$$

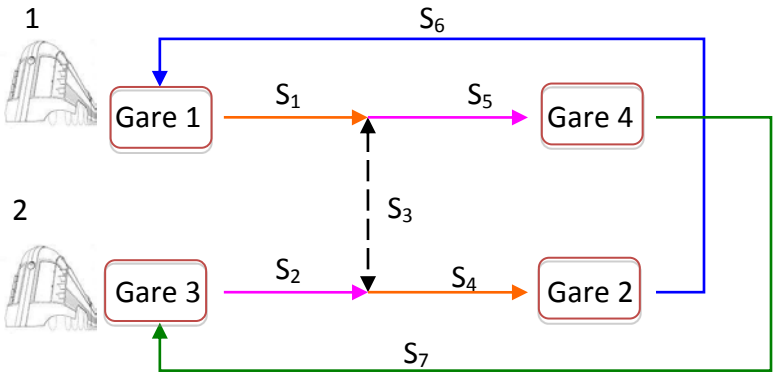
□

3.6 Exemple d'un réseau ferroviaire

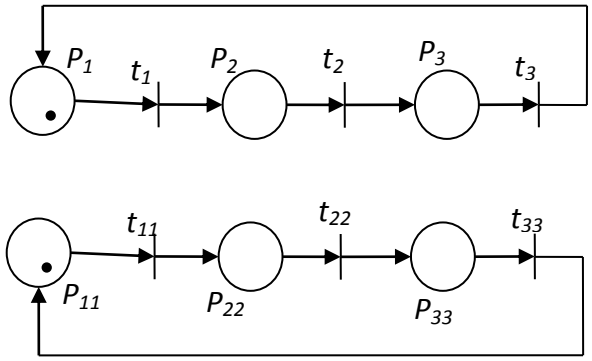
Cas 1) Considérons l'exemple montré dans la figure 3.7-a. Un réseau ferroviaire consiste de quatre gares, sept sections de vois, deux train et deux lignes. Le train 1 part depuis la gare 1 vers la gare 2 par les sections (S_1, S_3, S_4), par contre il rentre à la gare 1 directement par la section S_6. Pour le trajet du train 2, son départ est depuis la gare 3, et le terminus est à la gare 4, en passant dans les sections (S_2, S_3, S_5), et le retour est par la section S_7. Nous supposons que les sections (S_1, S_2, S_4, S_5, S_6, S_7) ont un sens unique, par contre la section S_3 à deux sens (aller-retour). La seule contrainte qui existe dans ce réseau, est d'éviter le passage simultané pour les deux trains dans la section S_3.

Le modèle RdP de cet exemple est montré dans la Figure 3. 8-b, où :

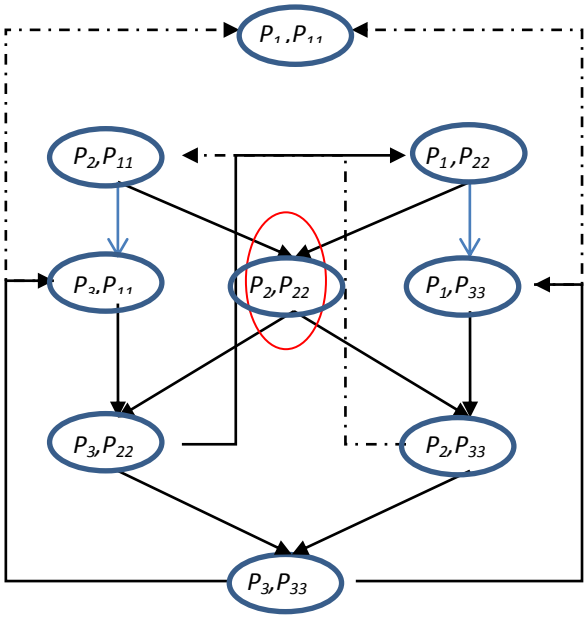
p_1 Le train 1 est dans la gare 1.
p_2 Le train 1 est dans la section S_3.
p_3 Le train 1 est dans la gare 2.
p_{11} Le train 2 est dans la gare 3.
p_{22} Le train 2 est dans la section S_3.
p_{33} Le train 2 est dans la gare 4.
t_1 Le train 1 dépasse la section S_1 et il entre dans la section S_3.
t_2 Le train 1 dépasse la section S_3.
t_3 Le train 1 quitte la gare 3 et il arrive à la gare 1 par la section S_6.
t_{11} Le train 2 dépasse la section S_2 et il entre dans la section S_3.
t_{22} Le train 1 dépasse la section S_3.
t_{33} Le train 1 quitte la gare 4 et il arrive à la gare 3 par la section S_7.



a) un réseau ferroviaire avec 2 trains



b) Un modèle RdP



c) le graphe d'atteignabilité

Figure 3. 7, ***Exemple d'un réseau ferroviaire avec un train dans chaque ligne***

Ce modèle est sauf (parce que le nombre maximum de jetons dans chaque place est inférieur ou égale un) et conservatif (tous les marquages sont accessibles à partir de M_0 et il existe des séquences de franchissement de ces marquages à M_0). Alors, l'état interdit est lorsque les deux places p_2 et p_{22} sont marqués simultanément.

- ✓ Cette spécification est modélisée par une seule CGEM :

$$M(p_2)+M(p_{22})\leq 1 \qquad (3.7)$$

- ✓ Ainsi, elle peut être modélisée par un ensemble de conditions (modèle de Holloway) :

$M(p)=1$ pour p_2 et p_{22} : $M(p_2)=1 \wedge M(p_{22})=1$

Cas 2) Considérons maintenant que ce réseau ferroviaire est utilisé par 4 trains comme montré dans la Figure 3. 8 : Initialement, il y a 2 trains dans la gare 1, et 2 dans la gare 2. Dans ce cas, les deux trains 1 et 2, peuvent passer dans la section S_3 successivement sans prendre en considération qui passe en premier, pareillement pour les trains 3 et 4. En revanche, le train 3 ne peut pas passer dans la section S_3 s'il existe au minimum un des deux trains 1 et 2, dans cette section, et vice-versa.

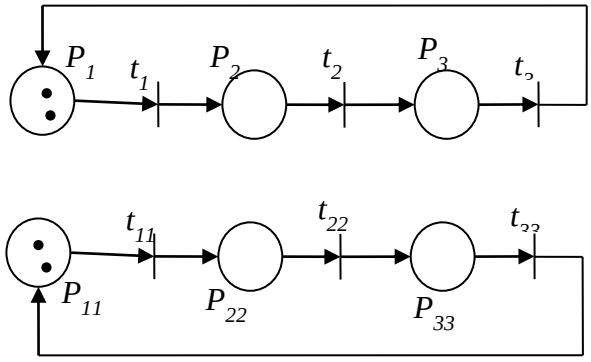


Figure 3. 8, ***Exemple d'un réseau ferroviaire avec deux trains dans chaque ligne***

- ✓ Il n'est plus possible de modéliser la spécification par la même CGEM, car une spécification de type CGEM ne permet pas de distinguer le cas où les trains vont dans le même sens du cas où les trains vont dans le sens opposé dans la section S_7.

Ainsi, le modèle de type prédicat ne traite pas cette spécification, puisque il traite juste les places sauf.

Pour modéliser cette spécification par CEM :

Il faut exclure les marquages:

M_1=[$M(p_2)$=1 $M(p_{22})$=1]
M_2=[$M(p_2)$=1 $M(p_{22})$=2]
M_3=[$M(p_2)$=2 $M(p_{22})$=1]
M_4=[$M(p_2)$=2 $M(p_{22})$=2]

Ces marquages peuvent être représentés par une classe de contraintes CEM :

$C\text{-}CEM$=CEM-Ou$_{(1)}\wedge$ CEM-Ou$_{(2)}\wedge$ CEM-Ou$_{(3)}\wedge$ CEM-Ou$_{(4)}$

CEM-Ou$_{(1)}$= $M(p_2)\neq1\vee M(p_{22})\neq1$
CEM-Ou$_{(2)}$= $M(p_2)\neq1\vee M(p_{22})\neq2$
CEM-Ou$_{(3)}$= $M(p_2)\neq2\vee M(p_{22})\neq1$
CEM-Ou$_{(4)}$= $M(p_2)\neq2\vee M(p_{22})\neq2$

$C\text{-}CEM\equiv\{M(p_2)\neq1\vee M(p_{22})\neq1\}\wedge\{M(p_2)\neq1\vee M(p_{22})\neq2\}\wedge\{M(p_2)\neq2\vee M(p_{22})\neq1\}\wedge$
$\{M(p_2)\neq2\vee M(p_{22})\neq2\}$

$\equiv\{M(p_2)\neq1\wedge M(p_2)\neq2\}\vee\{M(p_{22})\neq1\wedge M(p_{22})\neq2\}$

D'après la Propriété 3. 12:

$C\text{-}CEM\equiv\{M(p_2)\leq0\vee M(p_2)\geq3\}\vee\{M(p_{22})\leq0\vee M(p_{22})\geq3\}$

Puisque dans cet exemple, il n'existe que deux trains dans chaque ligne:

$C\text{-}CEM\equiv M(p_2)\leq0\vee M(p_{22})\leq0\equiv Dn(M(p_2),1)\cup Dn(M(p_{22}),1)$

Autrement dit, l'ensemble des marquages légaux est l'union de tous les deux marquages inférieurs de p_2 et p_{22} quand k_2=k_{22}=1.

3.7 Conclusion

Dans ce chapitre, nous avons présenté un nouveau modèle de contrainte *Contrainte d'Exclusion de Marquage*. Ce modèle apporte une puissance de modélisation complémentaire par rapport aux modèles de contraintes existantes dans la littérature pour représenter le problème d'états interdits PEI. Nous avons vu qu'une CEM concerne une place critique où elle rassemble les formes des contraintes linéaires avec les contraintes logiques.

Les extensions de ce modèle ont été aussi proposées pour traiter le cas de plusieurs places critiques, tels que : CEM-Ou, CEM-Et, CEM-M et C-CEM. Nous avons également donné des règles de réduction du nombre de contraintes dans certains cas.

Nous avons montré à partir d'un exemple de réseau ferroviaire que parfois les autres modèles ne modélisent pas le PEI, par contre le modèle CEM le peut.

Dans le chapitre suivant, nous allons discuter la SdC pour construire un superviseur capable de restreindre le comportement du système en respectant des spécifications modélisées par CEM ou une de ses extensions.

Chapitre 4

SdC de Graphes d'événements partiellement Contrôlable

Résumé :

Ce chapitre présente notre deuxième contribution, où nous proposons une solution pour résoudre le problème d'états interdits PEI pour les Graphes d'événements partiellement contrôlables soumis à des contraintes de type CEM. L'objectif est de synthétiser un superviseur qui garantit le respect du comportement désiré par le système en boucle fermée. Le superviseur observe l'évolution du système à chaque franchissement, et garantit le comportement désiré en contrôlant le franchissement des transitions contrôlables. Par contre dans le cas où il existe quelques transitions incontrôlable, le superviseur reçoit les informations sur le franchissement de toutes les transitions et ne contrôle que les transitions contrôlables. Une attention particulière est accordée à la permissivité de la loi de contrôle. La nouveauté dans ce chapitre se traduit par les lois et les fonctions qui observent, calculent et contrôlent le franchissement de la transition de sortie de la place critique. Les résultats de cette partie font l'objet des publications (Atli, July 2010) et (Atli, April 2010).

4.1 Introduction

Dans le chapitre précédent, nous avons proposé et formalisé un nouveau type de spécification que nous appelons CEM. Dans ce chapitre, nous développons une technique de SdC par supervision pour les systèmes à événements discrets modélisés par le *graphe d'événements* soumis à des *contraintes d'exclusion de marquage*.

La technique de SdC que nous proposons pour garantir le respect des spécifications de type CEM, nécessite de contrôler l'arrivée et le départ des jetons dans les places critiques. Ceci est une différence importante par rapport aux approches basées sur des spécifications de type CGEM, où on s'intéresse seulement au contrôle des jetons entrants dans les places critiques (Ghaffari A. X., 2003).

Ce chapitre est organisé comme suit : dans la section 2, nous présentons les notions de base qui sont utilisées pour le calcul de la loi de commande pour des graphes d'événements totalement contrôlables. Puis dans la section 3, nous proposons la loi de contrôle pour ce cas. Le cas où il y a des transitions incontrôlables est traité dans les sections 4 et 5. A la fin de ce chapitre, nous appliquons l'algorithme sur un exemple.

4.2 Notions de base pour le Graphe d'événements contrôlable

Les différents graphes d'atteignabilité que nous allons utiliser dans cette section sont illustrés comme suit:

$R(N,M_0)$ graphes d'atteignabilité (marquages) à partir de l'état initial

$R^c(N,M_0)$ graphes d'atteignabilité (marquages) à partir de l'état initial sous control

$R_1^c(N,M_0)$ graphes d'atteignabilité (marquages) à partir de l'état initial en franchissant juste une transition contrôlable

Notre approche de SdC calcule $R_1^c(N,M_0)$, c'est-à- dire que à chaque franchissement d'une transition contrôlable, notre approche met à jours le graphe d'atteignabilité qui respecte les spécifications.

L'approche que nous proposons est dédiée aux systèmes à événements discrets modélisés par des graphes d'événements. Alors l'évolution de marquage pour chaque place suit l'équation suivante :

$$\boldsymbol{M}[\boldsymbol{t} > \boldsymbol{M}', \boldsymbol{t} \in \{\boldsymbol{p}^\circ, {}^\circ\boldsymbol{p}\}: \boldsymbol{M}'(\boldsymbol{p}) = \boldsymbol{M}(\boldsymbol{p}) + \boldsymbol{\sigma}({}^\circ\boldsymbol{p}) - \boldsymbol{\sigma}(\boldsymbol{p}^\circ) \quad \textbf{(4.1)}$$

4.2.1 Graphe d'événements contrôlable

On dit qu'un graphe d'événements est contrôlable lorsque toutes les transitions dans ce graphe sont contrôlables : $\nexists t \in T_{uc}$. L'ensemble des marquages légaux M_L dans ce cas est l'ensemble de tous les marquages respectant toutes les spécifications:

$$M_L = R(N, M_0) \backslash M_f \qquad (4.2)$$

Par conséquent, l'ensemble de marquages légaux est obtenu en éliminant les marquages interdits de l'ensemble de marquages atteignables.

On dit qu'une transition est franchissable si chaque place des places d'entrée de cette transition a au minimum un jeton.

$$\mathbf{1}.(t \in T) = \begin{cases} \mathbf{1}\ (t\ est\ franchissable), si\ M(p) \geq \mathbf{1}, \forall p \in {}^{\circ}t \\ \mathbf{0}, \qquad\qquad\qquad\qquad sinon \end{cases} \qquad (IV.\ 1)$$

La loi du contrôle est modélisée par une fonction de contrôle binaire. Une transitions contrôlable est franchissable si elle est validée à la fois par le marquage du RdP et à la fois par la fonction de contrôle.

$$U(t \in T_c) = \begin{cases} \mathbf{1}, & t\ est\ validé \\ \mathbf{0}, & t\ n'est\ pas\ validé \end{cases} \qquad (IV.\ 2)$$

U_1 est la loi de contrôle qui permet le franchissement de toutes les transitions contrôlables validées par le marquage du RdP.
U_0 signifie que toutes les transitions contrôlables sont inhibées.

4.2.2 Superviseur optimal

L'objectif de notre approche est de construire un superviseur optimal, ici l'optimalité est définie par la capacité de superviseur à restreindre le moins possible le comportement du système. Cette optimalité est représentée dans le domaine de commande par la permissivité du superviseur.

Définition 4. 1: Un superviseur est optimal si et seulement si les deux conditions suivantes sont validées :

1. Aucun état interdit, $M \in M_f$, n'est atteignable

$$\forall t \in T, M \in M_L : M[t > M' \in M_f \Longrightarrow U(t) = 0$$

2. Aucun état légal $M \in M_L$ ne doit être interdit :

$$\forall M \in M_L, \nexists M' \in M_L\ tels\ que\ M \in M_L[t > M'\ et\ U(t) = 0$$

$$ou$$

$$\forall M, M' \in M_L\ et\ M[t > M'\ alors\ U(t) = 1$$

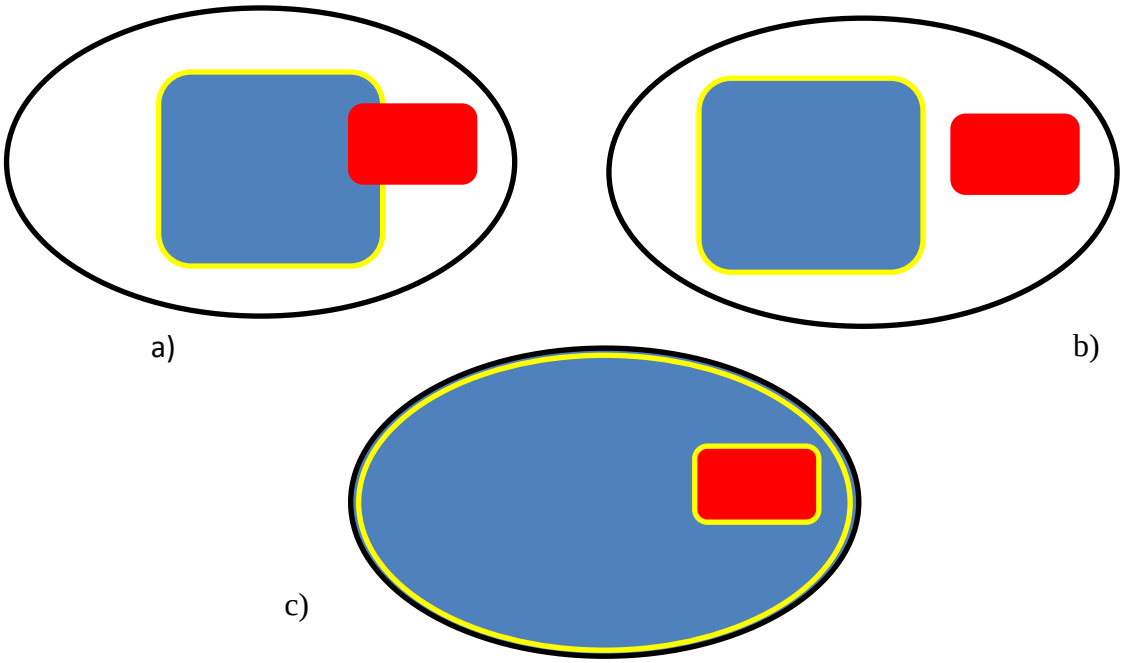


Figure 4. 1, *Optimalité sur un graphe d'événements contrôlable*

Pour expliquer le sens de l'optimalité, nous présentons la figure 4.1, où : la région entourée par la ligne noir représente $R(N,M_0)$, la région en rouge représente les marquages interdits M_f et celle en bleu et entourée en jaune représente $R^c(N,M_0)$.

Le superviseur en figure 4.1-a ne respecte pas les spécifications, par contre dans figure 4.1-b il respecte mais il n'est pas optimal. Juste celui dans la figure 4.1-c, est le superviseur optimal.

4.3 Loi de commande des graphes d'événements totalement contrôlable

Dans cette section, nous considérons un graphe d'événements totalement contrôlable pas nécessairement borné ou cyclique, soumis à des spécifications modélisés par CEM, CEM-Ou, CEM-Et ou CEM-M.

Par la suite, nous présentons la loi de contrôle proposé pour chacun de ces cas.

4.3.1 Système soumis à des spécifications de type CEM

Nous avons montré dans le chapitre 3, que les marquages légaux consistent en deux ensembles, les marquages supérieurs $Up(q,k)$ et les marquages inférieurs $Dn(q,k)$:

$$M_L(q,k) = Up(q,k) \cup Dn(q,k)$$

$$\forall M \in M_L \Rightarrow M \in \{Up(q,k) \cup Dn(q,k)\} \Rightarrow$$

$M \in \{M(q) > k\} \vee M \in \{M(q) < k\} \Rightarrow$

$M(q) \neq k \equiv M(q) > k \vee M(q) < k$

$M(q) \neq k \equiv M(q) \geq k + 1 \vee M(q) \leq k - 1$

Donc, le rôle de superviseur est d'empêcher le franchissement d'une transition contrôlable s'il conduit à un marquage interdit :

$M(q) \leq k - 1$
$(k - 1) - M(q) \geq 0$

$$(\boldsymbol{k-1}) - \boldsymbol{M_0(q)} - \boldsymbol{\sigma(^\circ q)} + \boldsymbol{\sigma(q^\circ)} \geq \mathbf{0} \qquad \textbf{(4.3)}$$

$M(q) \geq k + 1$
$-(k + 1) + M(q) \geq 0$

$$-(\boldsymbol{k+1}) + \boldsymbol{M_0(q)} + \boldsymbol{\sigma(^\circ q)} - \boldsymbol{\sigma(q^\circ)} \geq \mathbf{0} \qquad \textbf{(4.4)}$$

Le superviseur autorise le franchissement d'une transition contrôlable si au minimum une de deux inégalités (4.3) et (4.4) est satisfaite (validée).

Nous dénotons si une égalité est validée par: |(4.3)|=1, |(4.4)|=1

Sinon elle est dénotée par: |(4.3)|=0, |(4.4)|=0

□

Théorème 4. 1 : La loi de contrôle pour un graphe d'événements totalement contrôlable qui respecte une spécification de CEM est:

$$\boldsymbol{U(t \in Tc)} = \begin{cases} \mathbf{1}, & |(\mathbf{4.3})| \vee |(\mathbf{4.4})| = \mathbf{1} \\ \mathbf{0}, & \boldsymbol{sinon} \end{cases} \qquad \textbf{(IV. 3)}$$

Preuve :

Nous prouvons que les états interdits ne sont pas accessibles :

$M_0 \in M_L \Rightarrow$ soit $M_0(q) \in Up(q,k)$ ou $M_0(q) \in Dn(q,k)$.

D'abord nous discutons $M_0(q) \in Up(q,k)$, $M_0(q) \geq k+1$: si $M_0 \in M_L[\delta > M' \in M_f$, où M_f est définie par la définition 3.4$\Rightarrow M'(q) = k \Rightarrow M'(q) = M_0(q) + \sigma(^\circ q) - \sigma(q^\circ) = k \Rightarrow \sigma(^\circ p) < \sigma(p^\circ)$

Puisque notre approche présente SdC pour R_1^c (N, M_0)

le franchissement de q° conduit $M_0 \in M_L[\delta > M' \in M_f \Rightarrow \delta = q^\circ$, donc $\sigma(q^\circ)=1$ et $\sigma(^\circ q)=0$, autrement dit, il faut franchir q° pour atteindre un marquage interdit, mais ce

franchissement est interdit par la loi (*IV*.3) parce que les deux inégalités sont invalidées :

(4.3) $\Rightarrow$ (*k*-1)-*k*=-1, (4.3) n'est pas satisfaite (invalidée), donc |(4.3)|=0.
(4.4) $\Rightarrow$-(*k*+1)+*k*=-1, (4.4) est non plus (invalidée), donc |(4.4)|=0.

De même façon, nous discutons le cas où $M_0(q) \in Dn(q,k)$.

□

De cette preuve, nous pouvons présenter les corollaires suivants:

Corollaire 4. 1: $\forall q \in Q_{mc}$ soumis à une CEM, et $M(q) \in Dn(q,k)$, si $M(q) \in Dn(q,k)[\delta > M'(q) \in M_f \Rightarrow \delta = °q$.

Corollaire 4. 2: $\forall q \in Q_{mc}$ soumis à une CEM, et $M(q) \in Dn(q,k) \Rightarrow$ l'inégalité (4.3) est validée |(4.3)|=1, par contre l'inégalité (4.4) est invalidée |(4.4)|=0.

Propriété 4. 1: Pour une CEM, Si $M_0(q) \in Dn(q,k) \Rightarrow \boldsymbol{R_1^c}(N,M_0) \subseteq Dn(q,k)$.

Corollaire 4.3: $\forall q \in Q_{mc}$ soumis à une CEM, et $M(q) \in Up(q,k)$, si $M(q) \in Up(q,k)[\delta > M'(q) \in M_f \Rightarrow \delta = q°$.

Corollaire 4. 4: $\forall q \in Q_{mc}$ soumis à une CEM, et $M(q) \in Up(q,k) \Rightarrow$ l'inégalité (4.3) est invalidée |(4.3)|=0, par contre l'inégalité (4.4) est validée |(4.4)|=1.

Propriété 4. 2: Pour une CEM, Si $M_0(q) \in Up(q,k) \Rightarrow \boldsymbol{R_1^c}(N,M_0) \subseteq Up(q,k)$.

Théorème 4. 2: Pour un graphe d'événements contrôlables soumis à une CEM, la loi de contrôle décrite par l'expression IV.3, est maximale permissive.

Preuve :

Pour être maximale permissive, il faut valider les deux conditions (**la permissivité du** superviseur.

Définition 4. 1).

1. Aucun état interdit, $M \in M_f$, n'est atteignable

Cette condition est validée par le théorème 4.1.

2. Aucun état légal $M \in M_L$ ne doit être interdit

$$\forall M, M' \in M_L \; et \; M[t > M' \; alors \; U(t) = 1$$

Puisque les expressions (4.3), respectivement (4.4) représente $Dn(q,k)$, respectivement $Up(q,k)$, et puisque $M_L = Dn(q,k) \cup Up(q,k) \Rightarrow$ si $M \in M_L$, i.e., soit $M' \in Dn(q,k)$ ou $M' \in Up(q,k) \Rightarrow$ soit l'inégalité (4.3) est validée ou l'inégalité (4.4) est validée, et dans les deux cas, la loi de contrôle est satisfaite, et le franchissement est toujours satisfait.

Les deux conditions de la définition 4.1, sont validées $\Rightarrow$ la loi de contrôle IV.4 est maximale permissive.

□

Exemple : Nous considérons l'exemple dans la figure 4.2, la place p_1 est soumise à une CEM : $M(p_1) \neq 1$. Nous supposons que la transition t_1 va franchir à partir de M_0 :

$Up(p_1,1)=\{M_0, M_3\}$, $Dn(p_1,1)=\{M_2\}$ et $M_f=\{M_1\}$.

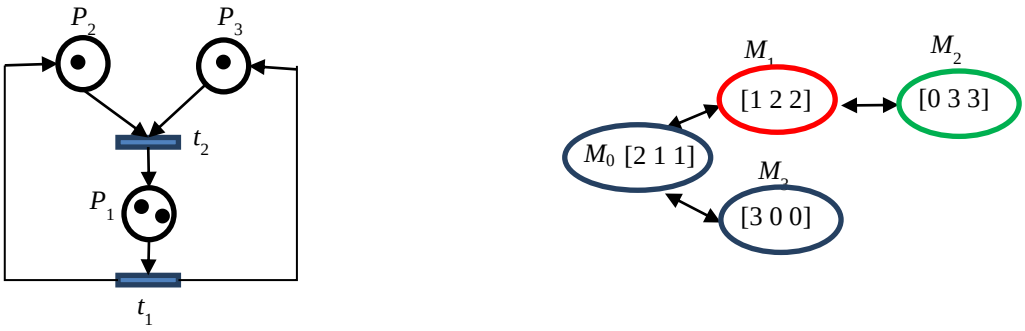
Nous calculons les deux inégalités (4.3) et (4.4), avec $\sigma(t_1=q°)=1$, $M_0(p_1)=2$, $\sigma(t_2=°q)=1$et $k=1$.

L'inégalité (4.3) : (1-1)-2-0+1=-1, cette inégalité est invalidée ($|(4.3)|=0$).

L'inégalité (4.4) : -(1+1)+2-0-1=-1, cette inégalité est aussi invalidée ($|(4.4)|=0$).

$\Rightarrow |(4.3)| \vee |(4.3)| = 0$

Par conséquent, le superviseur va interdire cette transition à franchir à partir de M_0.



a) Graphe d'événements b) Graphe de marquages

Figure 4. 2, *Exemple d'un graphe d'événements soumis à une CEM $M(p_1) \neq 1$*

4.3.2 Système soumis à des spécifications de type CEM-Ou

Nous considérons dans cette section un graphe d'événements contrôlables soumis à une CEM-Ou avec m_c places critiques. Chaque place critique q_i est soumise à une CEM $M(q_i) \neq k_i$, ces contraintes CEMs sont liées par la relation *Ou*. Ainsi, la loi de

contrôle dans ce cas, est composée de m_c lois de contrôle décrites dans IV.3, liées par la relation *Ou* comme suit :

Théorème 4. 3: La loi de contrôle pour un graphe d'événements totalement contrôlables qui respecte une spécification de CEM-Ou est:

$$\boldsymbol{U(t \in Tc)} = \begin{cases} \mathbf{1}, & \bigvee_{q_i \in Q_{mc}} \{|(\mathbf{4.3})| \vee |(\mathbf{4.4})|\} = \mathbf{1} \\ \mathbf{0}, & \boldsymbol{sinon} \end{cases} \qquad (\boldsymbol{IV.4})$$

Preuve :

Nous prouvons que les états interdits ne sont pas accessibles.

D'après la définition 3.7,

$$M_f = \begin{bmatrix} k_1 \\ . \\ . \\ . \\ k_{\mathrm{mc}} \end{bmatrix}$$

Chaque place critique p_i contient son nombre propre k_i, autrement dit que pour chaque place critique les deux inégalités (4.3) et (4.4) sont invalidées :

q	k	(4.3)	(4.4)	0/1	(4.3)$\vee$(4.4)
q_1	k_1	-1	-1	0$\vee$0	0
q_i	k_i	-1	-1	0$\vee$0	0
q_{mc}	k_{mc}	-1	-1	0$\vee$0	0
IV.4					0

Alors, la valeur logique du marquage interdit est fausse en fonction de l'expression IV.4, i.e., la loi de contrôle est 0, c'est-à-dire le franchissement de la transition qui conduit à ce marquage sera empêché, en conséquence il est impossible d'atteindre un marquage interdit sous la loi de contrôle IV.5.

□

De ce théorème, nous tirons le corollaire suivant :

Corollaire 4. 5: Afin d'assurer que le marquage M appartienne à $M_{L(Ou)}$, il suffit de valider au moins une égalité (4.3) ou (4.4) pour juste une place critique $q_s \in Q_{mc}$.

Théorème 4. 4: Pour un graphe d'événements contrôlables soumis à une CEM, la loi de contrôle décrite par l'expression IV.4, est maximale permissive.

Preuve :

Pour être maximale permissive, il faut valider les deux conditions (**la permissivité du** superviseur.

Définition 4. 1).

1. Aucun état interdit, $M \in M_f$, n'est atteignable

Cette condition est validée par le théorème 4.3.

2. Aucun état légal $M' \in M_L$ ne doit être interdit

Nous supposons que $\exists M' \in M_L : M \in M_L[t > M'$. Nous allons chercher si le franchissement de *t* peut être interdit par cette loi ou non.

L'expression suivante décrit l'ensemble des marquages légaux M_L :

$$\bigvee_{q_i \in Q_{mc}} \{|(\mathbf{4.3})|\mathrm{V}|(\mathbf{4.4})|\}$$

Tous les marquages qui appartiennent à cet ensemble valident cette expression :

Alors $\forall M' \in M_L$:

$$\bigvee_{q_i \in Q_{mc}} \{|(4.3)|\mathrm{V}|(4.4)|\} = 1$$

Selon la loi de contrôle IV.4 : il est impossible d'interdire le franchissement de $t \Rightarrow \nexists M' \in M_L : M \in M_L[t > M' \; et \; U(t) = 0$

Les deux conditions de la définition 4.1, sont validées $\Rightarrow$ la loi de contrôle IV.4 est maximal permissive.

□

4.3.3 Système soumis à des spécifications de type CEM-Et

Nous considérons dans cette section un graphe d'événements contrôlables soumis à une CEM-Et avec m_c places critiques. Chaque place critique q_i est soumise à une CEM $M(q_i) \neq k_i$, ces contraintes CEMs sont liées par la relation *Et*. Ainsi, la loi de contrôle dans ce cas, est composée de m_c sous lois de contrôle décrite par IV.3, liées par la relation *Et* comme suit :

Théorème 4. 5: La loi de contrôle pour un graphe d'événements totalement contrôlables qui respectent une spécification de CEM-Et est:

$$\boldsymbol{U(t \in Tc)} = \begin{cases} \mathbf{1}, & \bigwedge_{q_i \in Q_{mc}} \{|(\mathbf{4.3})|\mathrm{V}|(\mathbf{4.4})|\} = \mathbf{1} \\ \mathbf{0}, & \boldsymbol{sinon} \end{cases} \qquad (\boldsymbol{IV.5})$$

Preuve :

Nous prouvons que les états interdits ne sont pas accessibles :

D'après la définition 2.10, le système atteint un marquage interdit si une place critique a son nombre propre k_i.

Voyons l'expression IV.5, nous trouvons que s'il existe une place ayant son nombre propre (les deux inégalités de cette place sont invalidées), cette expression sera aussi invalidée, en conséquence le franchissement de la transition qui conduit à ce marquage sera interdit.

□

De ce théorème, nous concluons le corollaire suivant :

Corollaire 4. 6: Afin d'assurer que le marquage M est légal, $M \in M_{L(Et)}(Q_{mc}, K)$, il est nécessaire que chaque place critique $q_s \in Q_{mc}$ valide au moins une égalité (4.3) ou (4.4).

Théorème 4. 6: Pour un graphe d'événements contrôlable soumis à une CEM, la loi de contrôle décrie par l'expression IV.4, est maximale permissive.

Preuve :

Pour être maximale permissive, il faut valider les deux conditions (**la permissivité du** superviseur.

Définition 4. 1).

1 Aucun état interdit, $M \in M_f$, n'est atteignable

Cette condition est validée par le théorème 4.5.

2 Aucun état légal $M' \in M_L$ ne doit être interdit.

Nous supposons que$\exists M' \in M_L : M \in M_L[t > M'$, nous allons chercher si le franchissement de t peut être interdit par cette loi ou non.

L'expression suivante décrit l'ensemble des marquages légaux M_L :

$$\bigwedge_{q_i \in Q_{mc}} \{|(\mathbf{4.3})|\vee|(\mathbf{4.4})|\}$$

Tous les marquages qui appartiennent à cet ensemble valident cette expression :

Alors $\forall M' \in M_L$:

$$\bigwedge_{q_i \in Q_{mc}} \{|(\mathbf{4.3})|\vee|(\mathbf{4.4})|\} = 1$$

Selon la loi de contrôle IV.4 : le franchissement de t n'est pas interdit $\Rightarrow \nexists M' \in M_L : M \in M_L[\sigma(t) > M' \; et \; U(t) = 0$.

Les deux conditions de la définition 4.1, sont validées $\Rightarrow$ la loi de contrôle IV.5 est maximale permissive.

□

4.3.4 Système soumis à des spécifications de type CEM-M

Nous considérons dans cette section un graphe d'événements contrôlables soumis à une CEM-M avec m_c places critiques.

L'ensemble des marquages légaux, modélisé par une CEM-M est défini dans 3.13:

$$M_{L(CEM-M)}(Q_{mc}, k) = \sum_{i=1}^{mc} M(q_i) \neq k \equiv \left(\sum_{i=1}^{mc} M(q_i) \leq k-1\right) \vee \left(\sum_{i=1}^{mc} M(q_i) \geq k+1\right)$$

- $\left(\sum_{i=1}^{mc} \boldsymbol{M(q_i)} \leq \boldsymbol{k-1}\right) \Leftrightarrow \left((\boldsymbol{k-1}) - \sum_{i=1}^{mc} \boldsymbol{M(q_i)} \geq \boldsymbol{0}\right) \Rightarrow$

$$\left((\boldsymbol{k-1}) - \sum_{i=1}^{mc} \boldsymbol{M_0(q_i)} - \sum_{i=1}^{mc} \boldsymbol{\sigma(^\circ q_i)} + \sum_{i=1}^{mc} \boldsymbol{\sigma(q_i^\circ)} \geq \boldsymbol{0}\right) \qquad (\mathbf{4.5})$$

- $\left(\sum_{i=1}^{mc} \boldsymbol{M(q_i)} \geq \boldsymbol{k+1}\right) \Leftrightarrow \left(-(\boldsymbol{k+1}) + \sum_{i=1}^{mc} \boldsymbol{M(q_i)} \geq \boldsymbol{0}\right) \Rightarrow$

$$\left(-(\boldsymbol{k+1}) + \sum_{i=1}^{mc} \boldsymbol{M_0(q_i)} + \sum_{i=1}^{mc} \boldsymbol{\sigma(^\circ q_i)} - \sum_{i=1}^{mc} \boldsymbol{\sigma(q_i^\circ)} \geq \boldsymbol{0}\right) \qquad (\mathbf{4.6})$$

□

Théorème 4. 7: La loi de contrôle pour un graphe d'événements totalement contrôlables qui respectent une spécification de CEM-M est:

$$\boldsymbol{U(t \in Tc)} = \begin{cases} \mathbf{1}, & |(\mathbf{4.5})|\vee|(\mathbf{4.6})| = \mathbf{1} \\ \mathbf{0}, & \boldsymbol{sinon} \end{cases} \qquad (IV.\ 6)$$

Preuve :

Nous prouvons que les états interdits ne sont pas accessibles.

$M_0 \in M_L \Rightarrow$ soit $M_0 \in SMMS(Q_{mc}, k)$ ou $M_0 \in SMMI(Q_{mc}, k)$.

- D'abord nous discutons $M_0 \in SMMS(Q_{mc}, k)$:

$$\sum_{i=1}^{mc} \boldsymbol{M_0(q_i) \geq k + 1}$$

Si $M_0 \in M_L[\delta > M' \in M_f$, où M_f est définit par la définition 3.8 $\Rightarrow$

$$\sum_{i=1}^{mc} \boldsymbol{M'(q_i) = k}$$

$$\sum_{i=1}^{mc} \boldsymbol{M'(q_i) = \sum_{i=1}^{mc} M_0(q_i) + \sum_{i=1}^{mc} \sigma(^\circ q_i) - \sum_{i=1}^{mc} \sigma(q^\circ{}_i) = k}$$

$$\sum_{i=1}^{mc} \boldsymbol{\sigma(^\circ q_i) - \sum_{i=1}^{mc} \sigma(q^\circ{}_i) < 0}$$

$\Rightarrow$ Si $M_0 \in SMMS(Q_{mc}, k)$, pour atteindre un marquage interdit, il faut que le somme du nombre de fois de franchissement des transitions d'entrées soit inférieur à celui des transitions de sortie.

A. Considérons le cas où :

$$\sum_{i=1}^{mc} \boldsymbol{M_0(q_i) = k + 1}$$

Alors, pour atteindre un marquage interdit, il faut que la somme du nombre de fois de franchissement des transitions d'entrées soit égale à zéro, par contre celle des transitions de sortie doit être égale à un.

$$\sum_{i=1}^{mc} \sigma(q^\circ{}_i) = 1 \ et \sum_{i=1}^{mc} \sigma(^\circ q_i) = 0$$

autrement dit, il faut franchir $q°$ pour atteindre un marquage interdit, mais ce franchissement est interdit par la loi (*IV*.6), parce que les deux inégalités sont invalidées :

(4.5) n'est pas satisfaite (invalidée), donc |(4.5)|=0.

(4.6) non plus (invalidée), donc |(4.6)|=0.

B. Considérons le cas où :

$$\sum_{i=1}^{mc} \boldsymbol{M_0}(\boldsymbol{q_i}) = \boldsymbol{l} > \boldsymbol{k} + \mathbf{1}$$

$$\sum_{i=1}^{mc} \boldsymbol{\sigma}(°\boldsymbol{q_i}) - \sum_{i=1}^{mc} \boldsymbol{\sigma}(\boldsymbol{q}°_i) = \boldsymbol{k} - \boldsymbol{l}$$

Pour atteindre un marquage interdit, il faut que la différence entre la somme du nombre de fois de franchissement des transitions d'entrées et celui des transitions de sortie est égal de *k-l*.

Autrement dit, il faut franchir $q°$ plus que $°q$ pour atteindre un marquage interdit, mais ce franchissement est interdit par la loi (*IV*.6), parce que les deux inégalités sont invalidées :

(4.5) n'est pas satisfaite (invalidée), donc |(4.5)|=0.

(4.6) non plus (invalidée), donc |(4.6)|=0.

- De la même façon, nous discutons le cas où $\boldsymbol{M_0} \in SMMI(Q_{mc}, k)$.

□

Théorème 4. 8: Pour un graphe d'événements contrôlables soumis à une CEM-M, la loi de contrôle décrie par l'expression IV.6, est maximale permissive.

Preuve :

Pour être maximale permissive, il faut valider les deux conditions (**la permissivité du** superviseur.

Définition 4. 1).

1. Aucun état interdit, $M \in M_f$, n'est atteignable

Cette condition est validée par le théorème 4.7.

2. Aucun état légal $M \in M_L$ ne doit être interdit.

$M \in M_L[t > M' \in M_L$, $\forall$t : il faut prouver qu'il est toujours $U(t)=1$.

Nous supposons que $\exists M' \in M_L : M \in M_L[t > M'$, nous allons chercher si le franchissement de *t* peut être interdit par cette loi ou non.

L'expression suivante décrit l'ensemble des marquages légaux M_L :

$$|(\mathbf{4.5})|\vee|(\mathbf{4.6})|$$

Tous les marquages qui appartiennent à cet ensemble valident cette expression :

Alors $\forall M' \in M_L$:

$|(\mathbf{4.5})|\vee|(\mathbf{4.6})|$**=1**

Selon la loi de contrôle IV.6 : il est impossible d'interdire le franchissement de $t \Rightarrow \nexists M' \in M_L : M \in M_L[\sigma(t) > M' \; et \; U(t) = 0$.

Les deux conditions de la définition 4.1, sont validées $\Rightarrow$ la loi de contrôle IV.6 est maximale permissive.

□

Nous pouvons déduire les corollaires suivants:

Corollaire 4.7: $\forall Q_{mc}$ soumis à une CEM-M, $M \in SMMI(Q_{mc}, k)$, si $M[\delta > M'(q) \in M_f \Rightarrow \delta = °q_i$.

Corollaire 4. 8: $\forall Q_{mc}$ soumis à une CEM-M, et $M \in SMMI(Q_{mc}, k) \Rightarrow$ l'inégalité (4.5) est validée |(4.5)|=1, par contre l'inégalité (4.6) est invalidée |(4.6)|=0.

Propriété 4. 3: Pour une CEM-M, si $M_0 \in SMMI(Q_{mc}, k) \Rightarrow \boldsymbol{R_1^c}(N, M_0) \subseteq SMMI(Q_{mc}, k)$.

Corollaire 4.9: $\forall Q_{mc}$ soumis à une CEM-M, et $M \in SMMS(Q_{mc}, k)$, si $M(q)[\delta > M'(q) \in M_f \Rightarrow \delta = q_i°$.

Corollaire 4. 10: $\forall Q_{mc}$ soumis à une CEM-M, et $M \in SMMS(Q_{mc}, k) \Rightarrow$ l'inégalité (4.6) est invalidée |(4.5)|=0, par contre l'inégalité (4.6) est validée |(4.6)|=1.

Propriété 4. 4: Pour une CEM-M, Si $M_0 \in SMMS(Q_{mc}, k) \Rightarrow \boldsymbol{R_1^c}(N, M_0) \subseteq SMMS(Q_{mc}, k)$.

4.4 Notions de base pour les graphes d'événements partiellement contrôlables

L'approche développée dans cette section, propose une technique de SdC par supervision destinée aux graphes d'événements partiellement contrôlables. L'ensemble des transitions est partitionné en deux sous-ensembles $T=T_c \cup T_{uc}$: Le premier est l'ensemble des transitions contrôlables T_c (représenté par des rectangles), le second est l'ensemble des transitions incontrôlables T_{uc} (représenté par une ligne). Le superviseur peut empêcher le franchissement d'une transition contrôlable quand il faut, par contre il est impossible d'empêcher le franchissement d'une transition incontrôlable.

En raison de la présence de transitions incontrôlables, tous les marquages qui ne sont pas interdits, ne peuvent être considérés comme des marquages admis. Une conséquence directe de l'existence de transitions incontrôlables est l'existence des marquages dangereux M_D.

4.4.1 Ensemble de marquages dangereux

Définition 4. 2: L'ensemble de marquages dangereux M_D, est l'ensemble de marquages légaux à partir desquels les marquages interdits $M \in M_f$ sont accessibles d'une manière incontrôlable. Voir Figure 4. 3.

$$M_D=\{M \in M_L / M[\delta > M', M' \in M_f, \delta \subseteq T_{uc}\} \quad (4.7)$$

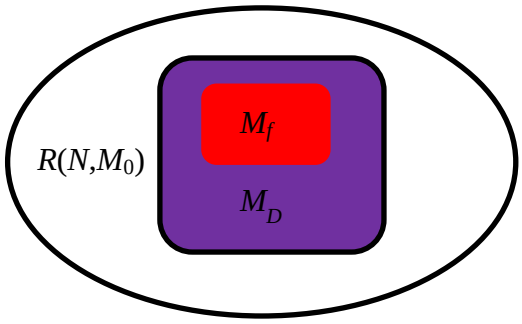


Figure 4. 3, $R(N,M_0)$, M_f, M_D et M_a

Pour un ensemble des marquages interdits M_f, en présence des transitions incontrôlables, il est nécessaire de restreindre le comportement du système en évitant non seulement tous les marquages interdits M_f, mais aussi l'ensemble des marquages dangereux M_D.

Définition 4. 3: L'ensemble de marquages admissibles, est un sous-ensemble de marquages légaux qui ne sont ni interdits, ni dangereux.

$$M_a = R(N,M_0) \setminus (M_f \cup M_D) \quad (4.8)$$

$M_a \subseteq M_L$. M_a est l'ensemble maximal des marquages où il existe une loi de contrôle qui restreint le comportement du système aux trajectoires désirées.

Définition 4. 4: $R^{uc}(N,M_0)$ est un graphe d'atteignabilité (marquages) à partir de l'état initial en franchissant des transitions incontrôlables.

4.4.2 Superviseur optimal pour les graphes d'événements partiellement contrôlables

A la présence de transitions incontrôlable, le sens de l'optimalité se traduit par la capacité du superviseur à restreindre le comportement du système pour accorder l'ensemble de marquages admissibles. Nous présentons la définition de superviseur maximal permissif à l'existence de transitions incontrôlables par la suite :

Définition 4. 5: Un superviseur est optimal à l'existence de transitions incontrôlables si et seulement si les deux conditions suivantes sont validées :

1. Aucun état interdit ou dangereux, $M \in (M_D \cup M_f)$, n'est atteignable

$$\forall t \in T_c, M \in M_a : M[t > M' \in (M_D \cup M_f) \Longrightarrow U(t \in T_c) = 0$$

2. Aucun état admissible $M \in M_a$ ne doit être interdit

$$\nexists M' \in M_a \; tels \; que \; M \in M_a[t > M' \; et \; U(t \in T_c) = 0$$

$$ou$$

$$\forall M, M' \in M_a \; et \; M[t > M' \; alors \; U(t \in T_c) = 1$$

□

4.4.3 Notions de base

Des définitions utilisées dans cette section pour le graphe d'événements, ont étés introduites dans les travaux de (Giua A. a.-C., 1993) et (Holloway & Krogh, 1990).

Définition 4. 6: Un nœud (Giua A. a.-C., 1993) représente soit une transition $t \in T$ ou une place $p \in P$.

Définition 4. 7: Un chemin (Holloway & Krogh, 1990) est un ensemble des nœuds se joignant par des arcs.

Définition 4. 8: Le chemin (Holloway & Krogh, 1990) d'influence d'une place critique $\pi(q)$, est un chemin qui commence par une transition contrôlable $t \in T_c$, et termine par

la transition d'entrée d'une place critique °q où toutes les autres transitions dans ce chemin sont incontrôlables. Cette transition contrôlable est appelée la transition d'influence.

Définition 4. 9: La zone d'influence de place critique $Z(q)$ (Giua A. a.-C., 1993) est une sous-réseau contenant tous les chemins d'influence de cette place.

Définition 4. 10: Nous formalisons l'ensemble de transitions d'influence d'une place critique par:

$$\boldsymbol{\Gamma(q) = \{t \in (Z(q) \cap T_c)\}}$$

Définition 4. 11: La distance de jetons entre la transitions d'entrée d'une place critique °q et une transition$\boldsymbol{t \in (Z(q))}$, $td(M, t \in T_c, {}^{\circ}q)$ est la somme des jetons existant entre $\boldsymbol{t \in T_c}$ **et** $\boldsymbol{{}^{\circ}q}$ au marquage M.

$$\boldsymbol{td(M, t \in T_c, {}^{\circ}q) = \sum M(p_i),\ p_i \in \pi(q)}$$

Corollaire 4. 11: L'évolution de $td(M, t' \in T_c, {}^{\circ}q)$ à suite du franchissement $M[t>M'$, est:

$$\boldsymbol{td(M', t', {}^{\circ}q) = \begin{cases} td(M, t', {}^{\circ}q) + 1 & si\ t = t' \\ td(M, t', {}^{\circ}q) - 1 & si\ t = {}^{\circ}q \\ td(M, t', {}^{\circ}q) & sinon \end{cases}} \qquad (IV.7)$$

Définition 4. 12: $D(q,M)$ (Giua A. a.-C., 1993) dénote le nombre maximal de fois que °q peut être franchie au marquage M, sans franchir les transitions d'influence. Autrement dit, $D(q,M)$ dénote le nombre maximal de jetons que peut atteindre q d'une manière incontrôlable sous U_{zero}.

$$\boldsymbol{D(q, M) = \min td(M, t \in \Gamma(q), {}^{\circ}q)} \qquad \textbf{(4.9)}$$

Définition 4. 13: $Y(q,M)$ (Ghaffari A. X., 2003), dénote le nombre maximal de jetons que la place critique q peut avoir par une manière incontrôlable sous la loi de contrôle U_{zero}.

$$\boldsymbol{Y(q, M) = M(q) + D(q, M)} \qquad \textbf{(4.10)}$$

$$\boldsymbol{Y(q, M) = M_0(q) + D(q, M) + \sigma({}^{\circ}q) - \sigma(q^{\circ})} \qquad \textbf{(4.11)}$$

Définition 4. 14: Soit $L(q)$, la zone d'entrée indirecte d'une place critique q, contenant tous les chemins qui précédent la transition de sortie q° et commencent par une transition contrôlable et terminent par la transition de sortie q°, en excluant les chemins contenant q.

Définition 4. 15: Soit $\lambda(q)$, l'ensemble des transitions contrôlables dans $L(q)$.

$$\boldsymbol{\lambda(q) = \{\, t \in (L(q) \cap T_c)\}}$$

Dans certains cas, même si la transition de sortie de la place critique est incontrôlable $t \in T_{uc}$, les transitions $t \in \lambda(q)$ peuvent contrôler le nombre maximal du franchissement de $q°$:

$$\boldsymbol{min_{t\in\lambda(q)}\big(\sigma(t) + td(M_0, q°, t)\big)} \qquad \textbf{(4.12)}$$

Exemple: Considérons l'exemple dans la Figure 4. 4, selon les définitions de cette section:

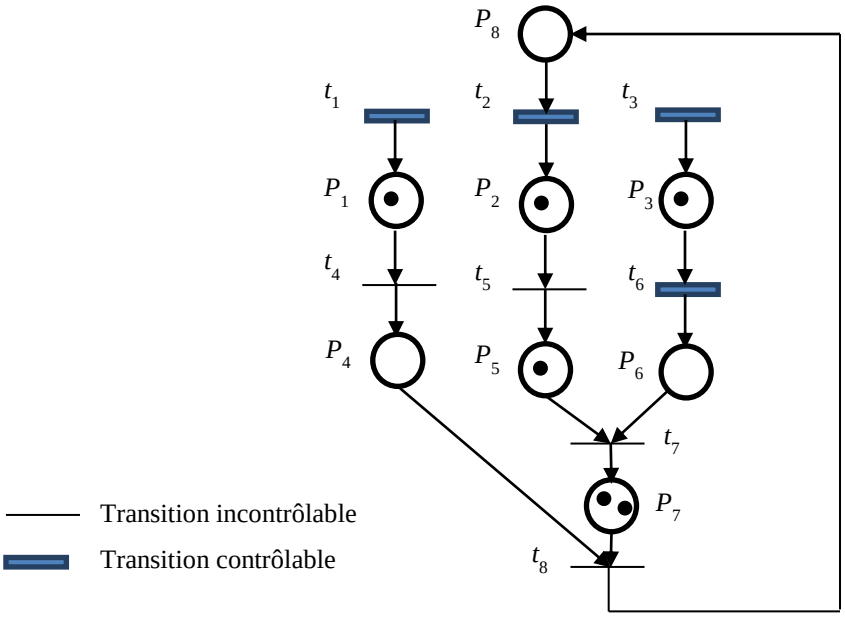


Figure 4. 4, *Graphe d'événements partiellement contrôlable avec une place critique p_7*, et M_0=[1 1 1 0 1 0 2 0]^t

La place critique p_7

La zone d'influence : Z(p_7)={ t_2, p_2, t_5, p_5, t_6, p_6, t_7}

Transition d'entrée $t_7 \in T_{uc}$

Transition de sortie $t_8 \in T_{uc}$

Les chemins d'influence :

π_1(p_7)={t_2, p_2, t_5, p_5, t_7}
π_2(p_7)={t_6, p_6, t_7}

Les transitions d'influence : Γ(p_7)={ t_2, t_6}

Les distances de jetons :

$td(M_0, t_2, t_7)=2$
$td(M_0, t_6, t_7)=0$
$D(p_7, M_0)=0$
$Y(p_7, M_0)=2$

La zone $L(p_7)= \{t_1, t_4, p_1, p_4\}$

L'ensemble $\lambda(p_7)=\{t_1\}$

4.5 Loi de contrôle des graphes d'événements partiellement contrôlable

Dans tous ce que nous avons mentionnés antérieurement, les deux inégalités (4.3) et (4.4) sont inapplicables quand les transitions d'entrée et de sortie sont incontrôlables. C'est pourquoi nous avons besoin de développer la loi de contrôle pour être adaptée à ce problème. Dans ce sens nous proposons les deux fonctions suivantes:

Définition 4. 16 : La fonction $S(°q)$ détermine le nombre maximal de fois que la transitions $°q$ peut être franchie.

$$\boldsymbol{S(°q)} = \begin{cases} \boldsymbol{\sigma(°q)} & \forall °\boldsymbol{q} \in \boldsymbol{T_c} \\ \boldsymbol{D(q, M) + \sigma(°q)} & \forall °\boldsymbol{q} \in \boldsymbol{T_{uc}} \end{cases} \qquad (\boldsymbol{IV.8})$$

Nous replaçons $\sigma(\boldsymbol{°q})$ par $S(°q)$ dans l'inégalité (4.3) :

$$\boldsymbol{(k-1) - M_0(q) - S(°q) + \sigma(q°) \geq 0} \qquad \textbf{(4.13)}$$

Définition 4. 17: La fonction $E(q°)$ caractérise le contrôle du nombre de franchissement d'une transition de sortie de la place critique $q°$.

$$\boldsymbol{E(q°)} = \begin{cases} \boldsymbol{Défini} \begin{cases} \boldsymbol{\sigma(q°)} & \forall \boldsymbol{q°} \in \boldsymbol{T_c} \\ \boldsymbol{min_{t\in\lambda(q)}(\sigma(t) + td(M_0, q°, t))} & \forall \boldsymbol{q°} \in \boldsymbol{T_{uc}} \wedge \boldsymbol{\lambda(q)} \neq \{\emptyset\} \end{cases} \\ \boldsymbol{Indéfini} \quad \forall \boldsymbol{q°} \in \boldsymbol{T_{uc}} \wedge \boldsymbol{\lambda(q)} = \{\emptyset\} \end{cases} \qquad (\boldsymbol{IV.9})$$

Si $E(q°)$ est indéfinie, alors on ne peut pas contrôler le franchissement de $q°$.

Nous replaçons $\sigma(\boldsymbol{q°})$ par $E(q°)$ dans l'inégalité (4.4):

$$\boldsymbol{1.\{E(q°)\ défini\}.(-(k+1) + M_0(q) + \sigma(°q) - E(q°) \geq 0)} \qquad \textbf{(4.14)}$$

où :

$$\mathbf{1}.\{q^{\circ}\ \boldsymbol{définie}\} = \begin{cases} \mathbf{1}, & \boldsymbol{si\ E(q^{\circ})définie} \\ \mathbf{0}, & \boldsymbol{sinon} \end{cases} \qquad \textbf{(IV.10)}$$

Propriété 4. 5: Nous ne replaçons pas σ(*q*°) par *E*(*q*°) dans l'inégalité (4.13), parce que le rôle de σ(*q*°) dans cette inégalité est de calculer le nombre de jetons existant dans *q* et pas le nombre de jetons qui peuvent atteindre *q,* et n'est pas de les contrôler.

Propriété 4. 6: Nous ne replaçons pas σ(°*q*) par *S*(°*q*) dans (4.14), parce que le rôle de σ(°*q*) dans cette inégalité est de calculer le nombre de jetons existant dans *q* et pas le nombre de jetons qui peuvent atteindre *q,* et n'est pas de les contrôler.

Propriété 4. 7: Pour les graphes d'événements partiellement contrôlables, l'ensemble $R_1^c(N,M_0)$ désigne les marquages atteignables à partir de l'état initial en franchissant juste une transition contrôlable et toutes les transitions incontrôlables franchissables.

4.5.1 Système soumis à des spécifications de type CEM

Nous considérons dans cette section un graphe d'événements partiellement contrôlable soumis à une CEM.

Théorème 4. 9: La loi de contrôle pour un graphe d'événements partiellement contrôlable qui respecte une spécification de CEM est:

$$\boldsymbol{U(t \in Tc)} = \begin{cases} \mathbf{1}, & |(\mathbf{4.13})|\vee|(\mathbf{4.14})| = \mathbf{1} \\ \mathbf{0}, & \boldsymbol{sinon} \end{cases} \qquad \textbf{(IV.11)}$$

Preuve :

Nous prouvons que les états interdits ne sont pas accessibles en utilisant cette loi. Cette preuve consiste en deux parties :

1. S'il existe un marquage interdit accessible sous cette loi de contrôle à partir d'un marquage admissible, alors cette loi n'est pas suffisante : $\exists M' \in M_f$: $M \in M_a[t > M' \in M_f$ où $(t \in T_c)$,i.e. $M'(q)=k \Rightarrow$La loi de contrôle (IV.11) est n'est pas satisfaite.

   Il faut prouver que :

   $M \in M_a[t > M' \in M_f$ où $(t \in T_c) \Rightarrow U(t)=0$

2. S'il existe un marquage dangereux accessible sous cette loi de contrôle à partir d'un marquage admissible, alors cette loi n'est pas suffisante : $\exists M' \in M_D$: $M \in M_a[t > M' \in M_D$ où $(t \in T_c)$, *i.e.* $M'(q)[\delta > M''(q)=k$ où δ une séquence de franchissements incontrôlables $\Rightarrow$ La loi de contrôle (IV.11) est n'est pas satisfaite.

Il faut prouver que:

$M \in M_a[t > M' \in M_D$ où $(t \in T_c) \Rightarrow U(t)=0$

Le premier cas : $M \in M_a[t > M' \in M_f$ où $(t \in T_c)$ i.e. $M'(q)=k$. Pour ce cas, le marquage interdit est accessible directement de M_a sans passer par M_D. Deux possibilités peuvent être discutées :

- **D'abord nous discutons $M(q) \in Up(q,k)$:**

  Selon Corollaire 4.3$\Rightarrow t=q°$.

  Selon Corollaire 4. 4$\Rightarrow$ nous cherchons la validité de l'inégalité de (4.14) :

$$\mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q}°)\ \boldsymbol{défini}\}.(-(\boldsymbol{k}+\mathbf{1})+\boldsymbol{M_0}(\boldsymbol{q})+\boldsymbol{\sigma}(°\boldsymbol{q})-\boldsymbol{E}(\boldsymbol{q}°)\geq \mathbf{0})$$

La fonction $E(q°)$ est définie et parce que $q° \in T_c \Rightarrow E(q°)=\sigma(q°)$

$$-(\boldsymbol{k}+\mathbf{1})+\boldsymbol{M_0}(\boldsymbol{q})+\boldsymbol{\sigma}(°\boldsymbol{q})-\boldsymbol{\sigma}(\boldsymbol{q}°)\geq \mathbf{0}$$

$M'(q)= M_0(q)+\sigma(°q)-\sigma(q°)=k \in M_f$

$-(k+1)+k \not\geq 0$

Alors la valeur logique de (4.14) est fausse, et celle de (4.13) est déjà fausse$\Rightarrow$La valeur de l'expression (IV.11) est fausse$\Rightarrow\sigma(t \in T_c)=0$, i.e. le franchissement de $t \in T_c$ est interdit, en conséquence la loi de contrôle décrite par (IV.11) empêche le système d'atteindre un marquage interdit.

$$\left.\begin{matrix}(4.13)\ invalidée\\(4.14)\ inavlidée\end{matrix}\right\} \Longrightarrow |(4.13)| \vee |(4.14)| = 0 \Longrightarrow U(t)=0$$

- **Ensuite nous discutons $M(q) \in Dn(q,k)$:**

  Selon Corollaire 4. 1$\Rightarrow t=°q \in T_c$.

  Selon Corollaire 4. 2$\Rightarrow$ nous cherchons la validité de l'inégalité de (4.13) :

$$(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(\boldsymbol{q})-\boldsymbol{S}(°\boldsymbol{q})+\boldsymbol{\sigma}(\boldsymbol{q}°)\geq \mathbf{0}$$

La fonction $S(°q)=\sigma(°q)$

$$(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(\boldsymbol{q})-\boldsymbol{\sigma}(°\boldsymbol{q})+\boldsymbol{\sigma}(\boldsymbol{q}°)\geq \mathbf{0}$$

$M'(q)= M_0(q)+\sigma(°q)-\sigma(q°)=k \in M_f$

$$(k-1)-k \not\geq 0$$

Alors la valeur logique de (4.13) est fausse, et celle de (4.14) est déjà fausse⇒ La valeur de l'expression (IV.11) est fausse⇒$U(t\in T_c)$=0, i.e. le franchissement de $t\in T_c$ est interdit, en conséquence la loi de contrôle décrite par (IV.11) empêche le système d'atteindre un marquage interdit.

$$\left.\begin{matrix}(4.13)\ invalidée\\(4.14)\ inavlidée\end{matrix}\right\} \Longrightarrow |(4.13)| \vee |(4.14)| = 0 \Longrightarrow U(t) = 0$$

Le deuxième cas : $M\in M_a[\sigma(t\in T_c)>M'\in M_D$, *i.e.* $M'(q)[\delta>M''(q)=k$; le marquage interdit est accessible de M_a en passant par M_D. Deux possibilités peuvent être discutées :

- **D'abord nous discutons *M(q)∈Up(q,k)* :**

Selon Corollaire 4.3⇒ nous cherchons la validité de l'inégalité (4.14) :

$$\mathbf{1}.\{\boldsymbol{E(q^\circ)\ défini}\}.(-(\boldsymbol{k}+\mathbf{1})+\boldsymbol{M_0(q)}+\boldsymbol{\sigma(^\circ q)}-\boldsymbol{E(q^\circ)}\geq\mathbf{0})$$

Deux possibilités :

a. $E(q°)$ est définie⇒ $\lambda(q)\neq\{\phi\}$.

Selon Corollaire 4.3⇒ le franchissement de $q°$ conduit à un marquage interdit, mais dans ce cas la fonction $E(q°)$ prend ce rôle, parce que $q°$ est incontrôlable. Il existe un franchissement d'une transition $t\in\lambda(q)$ qui conduit à un marquage dangereux.

$M'(q)= M_0(q)+\sigma(^\circ q)-E(q^\circ)=k\in(M_f\cup M_D)$

$$-(k+1)+k\not\geq 0$$

Alors la valeur logique de l'inégalité (4.14) est fausse, et celle de (4.13) est déjà fausse ⇒La valeur de l'expression (IV.11) est fausse⇒$\sigma(t\in\lambda(q))$=0, i.e. le franchissement de $t\in\lambda(q)$ est interdit, en conséquence la loi de contrôle décrie par (IV.11) empêche le système à atteindre à un marquage interdit.

$$\left.\begin{matrix}(4.13)\ invalidée\\(4.14)\ inavlidée\end{matrix}\right\} \Longrightarrow |(4.13)| \vee |(4.14)| = 0 \Longrightarrow U(t) = 0$$

b. $E(q°)$ est indéfinie⇒ $\lambda(q)=\{\phi\}$.

Il n'existe pas un moyen pour contrôler $q°$.

- **Ensuite nous discutons *M(q)∈Dn(q,k)* :**

Selon Corollaire 4. 2⇒ nous cherchons la validité de l'inégalité (4.13) :

$$(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0(q)}-\boldsymbol{S(^\circ q)}+\boldsymbol{\sigma(q^\circ)}\geq\mathbf{0}$$

Selon Corollaire 4. 1⇒ le franchissement de °q conduit à un marquage interdit, mais dans ce cas la fonction S(°q) prend ce rôle, parce que °q est incontrôlable. Il existe un franchissement d'une transition $t\in\Gamma(q)$ qui conduit à un marquage dangereux.

$$M'(q)= M_0(q) + \sigma(^\circ q) - \sigma(q^\circ) = k \in (M_f \cup M_D)$$

$$(k-1) - k \ngeq 0$$

Alors la valeur logique de (4.13) est fausse, et celle de (4.14) est déjà fausse ⇒ La valeur de l'expression (IV.11) est fausse ⇒$U(t\in\Gamma(q))$=0, i.e. le franchissement de $t\in\Gamma(q)$ est interdit, en conséquence la loi de contrôle décrite par (IV.11) empêche le système d'atteindre à un marquage interdit.

$$\left.\begin{matrix}(4.13)\ invalidée\\(4.14)\ inavlidée\end{matrix}\right\} \Longrightarrow |(4.13)| \vee |(4.14)| = 0 \Longrightarrow U(t) = 0$$

□

Théorème 4. 10: Pour un graphe d'événements partiellement contrôlable soumis à une CEM, la loi de contrôle décrite par l'expression IV.11, est maximale permissive.

Preuve :

Il faut montrer que les relations suivantes sont toujours vraies (Définition 4. 5).

1. Aucun état interdit ou dangereux, $M\in(M_D \cup M_f)$, n'est atteignable

$$\forall t \in T, M \in M_a: M[\sigma(t) > M' \in (M_D \cup M_f\ ) \implies U(t) = 0$$

Cette condition est déjà prouvée dans (Théorème 4. 10).

2. Aucun état admissible $M\in M_a$ ne doit être interdit.

Nous supposons que: $\exists M' \in M_a: M \in M_a[\sigma(t) > M'\ o\grave{u}\ U(t) = 0$

$U(t)$ est interdit ⇒ la valeur logique de l'expression (IV.11) est égale à zéro⇒

$|(4.13)|\vee|(4.14)|$=0⇒$M'\notin M_a$ (car les deux inégalités décrient l'ensemble de marquages admissible M_a), mais ceci contredit l'hypothèse, alors :

$$\nexists M' \in M_a: M \in M_a[\sigma(t) > M'\ et\ U(t) = 0$$

4.5.2 Système soumis à des spécifications de type CEM-Ou

Nous considérons dans cette section un graphe d'événements partiellement contrôlable soumis à une CEM-Ou avec m_c places critiques. Chaque place critique q_i est soumise à une CEM séparément $M(q_i)\neq k_i$, ces contraintes CEMs sont liées par la

relation *Ou*. Ainsi, la loi de contrôle dans ce cas, consiste en m_c sous lois de contrôle liées par la relation *Ou* comme suit :

Théorème 4. 11: La loi de contrôle pour un graphe d'événements partiellement contrôlable qui respecte une spécification de CEM-Ou est:

$$\boldsymbol{U(t \in Tc)} = \begin{cases} \mathbf{1}, & \bigvee_{q_i \in Q_{mc}} \{|(\mathbf{4.13})|\mathrm{V}|(\mathbf{4.14})|\} = \mathbf{1} \\ \mathbf{0}, & \boldsymbol{sinon} \end{cases} \qquad (\boldsymbol{IV.12})$$

Preuve :

Nous prouvons que les états interdits M_f et les états dangereux M_D ne sont pas accessibles.

Il faut prouver que les deux expressions suivantes sont toujours vraies:

1- $M \in M_a[\sigma(t \in T_c) > M' \in M_f \Rightarrow U(t)=0$

2- $M \in M_a[\sigma(t \in T_c) > M' \in M_D \Rightarrow U(t)=0$

Le premier cas : $M \in M_a[t > M' \in M_f$ où $(t \in T_c)$. D'après la Définition 3.7

$$M_f = \begin{bmatrix} k_1 \\ . \\ k_s \\ . \\ k_{\mathrm{mc}} \end{bmatrix}$$

Supposons que chaque place critique q_i a son nombre k_i propre sauf une place q_s :

$$M' = \begin{bmatrix} k_1 \\ . \\ \neq k_s \\ . \\ k_{\mathrm{mc}} \end{bmatrix} \in M_a$$

Pour une contrainte CEM-Ou, un état n'est pas interdit si au moins une des places critiques ne contient pas un nombre de jetons égal à son nombre propre, la question dans ce cas se transforme en problème d'une place critique q_s.

La preuve dans ce cas est similaire au premier cas dans la preuve du théorème 4.3.

Le deuxième cas : $M \in M_a[t \in > M' \in M_D$ où $(t \in T_c)$; le marquage interdit est accessible de M_a en passant par M_D.

Le problème dans ce cas se transforme en problème d'une place critique q_s.

Nous avons montré dans le deuxième cas, dans la preuve du théorème 4.9, que pour chaque place critique dans un cas dangereux, les deux inégalités (4.13) et (3.14) ne sont pas satisfaites.

Aussi longtemps qu'il existe une place critique qui n'atteint pas un cas dangereux, la loi de contrôle (IV.12) est satisfaite. Tant que cette place tombe dans un cas dangereux, la preuve dans ce cas est similaire au deuxième cas dans la preuve du théorème précédent.

□

Théorème 4. 12: Pour un graphe d'événements partiellement contrôlable soumis à une CEM-Ou, la loi de contrôle décrite par l'expression IV.12, est maximale permissive.

Preuve :

Il faut montrer que les relations suivantes sont toujours vraies (Définition 4. 5).

1. Aucun état interdit ou dangereux, $M \in (M_D \cup M_f)$, n'est atteignable

$$\forall t \in T, M \in M_a : M[\sigma(t) > M' \in (M_D \cup M_f) \implies U(t) = 0$$

   Cette condition est déjà prouvée (Théorème 4. 10).

2. Aucun état admissible $M \in M_a$ ne doit être interdit.

   Nous supposons que: $\exists M' \in M_a : M \in M_a[\sigma(t) > M' \; et \; U(t) = 0$

   $U(t)$ est interdit$\Rightarrow$ la valeur logique de l'expression (IV.12) égale à zéro$\Rightarrow$

   $M' \notin M_a$ (car les deux inégalités décrivent l'ensemble de marquages admissibles M_a, mais ceci contredit l'hypothèse, et prouve le résultat du théorème :

$$\nexists M' \in M_a : M \in M_a[\sigma(t) > M' \; où \; U(t) = 0$$

4.5.3 Système soumis à des spécifications de type CEM-Et

Nous considérons dans cette section un graphe d'événements contrôlables soumis à une CEM-Et avec m_c places critiques. Chaque place critique q_i est soumise à une CEM $M(q_i) \neq k_i$, ces contraintes CEMs sont liées par la relation *Et*. Ainsi, la loi de contrôle dans ce cas, est composée de m_c sous-lois de contrôle liées par la relation *Et*.

Théorème 4. 13: La loi de contrôle pour un graphe d'événements partiellement contrôlable soumis à une CEM-Et, est :

$$\boldsymbol{U(t \in Tc)} = \begin{cases} \mathbf{1}, & \bigwedge_{q_i \in Q_{mc}} \{|(\mathbf{4.13})| \vee |(\mathbf{4.14})|\} = \mathbf{1} \\ \mathbf{0}, & \boldsymbol{sinon} \end{cases} \qquad (\boldsymbol{IV.13})$$

Preuve :

Nous prouvons que les états interdits M_f et les états dangereux M_D ne sont pas accessibles.

Il faut montrer que les relations suivantes sont toujours vraies :

1- $M \in M_a[\sigma(t \in T_c) > M' \in M_f \Rightarrow U(t)=0$

2- $M \in M_a[\sigma(t \in T_c) > M' \in M_D \Rightarrow U(t)=0$

Le premier cas : $M \in M_a[\sigma(t \in T_c) > M' \in M_f$.

D'après la définition 3.10 : le marquage est un marquage interdit par rapport à une contrainte CEM-Et si une place critique a son nombre propre, dans ce cas la preuve est similaire au premier cas dans la preuve du théorème 4.11.

Le deuxième cas : $M \in M_a[\sigma(t \in T_c) > M' \in M_D$; le marquage interdit est accessible de M_a en passant par M_D.

Le marquage est un marquage interdit si une place critique a un nombre de jetons correspondant à un marquage dangereux. Dans ce cas la preuve est similaire au deuxième cas dans la preuve du théorème 4.11.

□

Théorème 4. 14: Pour un graphe d'événements partiellement contrôlable soumis à une CEM-Et, la loi de contrôle décrite par l'expression IV.13, est maximale permissive.

Preuve :

Il faut montrer que les relations suivantes sont toujours vraies (Définition 4. 5):

1. Aucun état interdit ou dangereux, $M \in (M_D \cup M_f)$, n'est atteignable

   $$\forall t \in T, M \in M_a : M[\sigma(t) > M' \in (M_D \cup M_f) \implies U(t) = 0$$

   Cette condition est déjà prouvée dans (Théorème 4. 10).

2. Aucun état admissible $M \in M_a$ ne doit être interdit.

   Nous supposons que: $\exists M' \in M_a : M \in M_a[\sigma(t) > M'$ où $U(t) = 0$

$U(t)$ est interdit⇒ la valeur logique de l'expression (IV.13) égale à zéro⇒

$M' \notin M_a$ (car les deux inégalités décrient l'ensemble de marquages admissible M_a), mais ceci contredit l'hypothèse, et prouve le résultat du théorème :

$$\nexists M' \in M_a : M \in M_a [\sigma(t) > M' \text{ où } U(t) = 0$$

□

4.5.4 Système soumis à des spécifications de type CEM-M

Nous considérons dans cette section un graphe d'événements partiellement contrôlable soumis à une CEM-M avec m_c places critiques.

$$M_{L(CEM-M)} = \sum_{i=1}^{mc} M(q_i) \neq k \supseteq M_a \equiv \left(\sum_{i=1}^{mc} Y(q_i) \leq k-1 \right) \vee \left(\sum_{i=1}^{mc} M(q_i) \geq k_i + 1 \right)$$

$$\blacksquare \left(\sum_{i=1}^{mc} \boldsymbol{Y(q_i) \leq k-1} \right) \Leftrightarrow \left(\boldsymbol{(k-1) - \sum_{i=1}^{mc} Y(q_i) \geq 0} \right) \Rightarrow$$

$$\left(\boldsymbol{(k-1) - \sum_{i=1}^{mc} M_0(q_i) - \sum_{i=1}^{mc} S(^\circ q_i) + \sum_{i=1}^{mc} \sigma(q_i^\circ) \geq 0} \right) \qquad \boldsymbol{(4.15)}$$

$$\blacksquare \left(\sum_{i=1}^{mc} \boldsymbol{M(q_i) \geq k+1} \right) \Leftrightarrow \left(\boldsymbol{-(k+1) + \sum_{i=1}^{mc} M(q_i) \geq 0} \right) \Rightarrow$$

$$\left(\boldsymbol{-(k+1) + \begin{bmatrix} 1.\{E(q_1^\circ)\} \\ \dots \\ 1.\{E(q_i^\circ)\} \\ \dots \\ 1.\{E(q_{mc}^\circ)\} \end{bmatrix}^t . \begin{bmatrix} M_0(q_1) + \sigma(^\circ q_1) - E(q_1^\circ) \\ \dots \\ M_0(q_i) + \sigma(^\circ q_i) - E(q_i^\circ) \\ \dots \\ M_0(q_{mc}) + \sigma(^\circ q_{mc}) - E(q_{mc}^\circ) \end{bmatrix} \geq 0} \right) \qquad \boldsymbol{(4.16)}$$

Théorème 4. 15: La loi de contrôle pour un graphe d'événements partiellement contrôlable soumis à une CEM-M, est :

$$\boldsymbol{U(t \in Tc)} \begin{cases} \boldsymbol{1}, & \boldsymbol{|(4.15)| \vee |(4.16)| = 1} \\ \boldsymbol{0} & \boldsymbol{sinon} \end{cases} \qquad (\boldsymbol{IV.14})$$

Preuve :

Nous prouvons que les états interdits ne sont pas accessibles.

Il faut prouver qu'il est toujours :

1- $M \in M_a[\sigma(t \in T_c) > M' \in M_f \Rightarrow U(t)=0$

2- $M \in M_a[\sigma(t \in T_c) > M' \in M_D \Rightarrow U(t)=0$

Le premier cas : $M \in M_a[\sigma(t \in T_c) > M' \in M_f$.

soit $M \in SMMS(Q_{mc}, k)$ ou $M \in SMMI(Q_{mc}, k)$.

- **D'abord nous discutons $M \in SMMS(Q_{mc}, k)$:**

  Selon Corollaire 4.3$\Rightarrow$ $t=q_i^\circ$.

  Selon Corollaire 4. 4$\Rightarrow$ nous cherchons la validité de l'inégalité (4.16):

$$\left(-(k+1) + \begin{bmatrix} 1.\{E(q_1^\circ)\} \\ \dots \\ 1.\{E(q_i^\circ)\} \\ \dots \\ 1.\{E(q_{mc}^\circ)\} \end{bmatrix}^t . \begin{bmatrix} M_0(q_1) + \sigma(^\circ q_1) - E(q_1^\circ) \\ \dots \\ M_0(q_i) + \sigma(^\circ q_i) - E(q_i^\circ) \\ \dots \\ M_0(q_{mc}) + \sigma(^\circ q_{mc}) - E(q_{mc}^\circ) \end{bmatrix} \geq 0 \right)$$

La fonction $E(q^\circ)$ est définie pour toutes les places critiques parce que $q_i^\circ \in T_c \Rightarrow$ $E(q_i^\circ)=\sigma(q_i^\circ)$:

L'inégalité (4.16) prend la forme de (4.6) :

$$\left(-(k+1) + \sum_{i=1}^{mc} M_0(q_i) + \sum_{i=1}^{mc} \sigma(^\circ q_i) - \sum_{i=1}^{mc} \sigma(q_i^\circ) \geq 0 \right)$$

C'est déjà prouvé dans la preuve du théorème (4.7).

- **Ensuite nous discutons $M \in SMMI(Q_{mc}, k)$:**

  Selon Corollaire 4. 1$\Rightarrow$ $t=^\circ q_i$.

  Selon Corollaire 4. 2$\Rightarrow$ nous cherchons la validité de l'inégalité (4.15) :

$$\left((k-1) - \sum_{i=1}^{mc} M_0(q_i) - \sum_{i=1}^{mc} S(^\circ q_i) + \sum_{i=1}^{mc} \sigma(q_i^\circ) \geq 0 \right)$$

La fonction $S(^\circ q_i)=\sigma(^\circ q_i)$, l'inégalité (4.15) prend la forme de (4.5) :

$$\left((k-1) - \sum_{i=1}^{mc} M_0(q_i) - \sum_{i=1}^{mc} \sigma(^\circ q_i) + \sum_{i=1}^{mc} \sigma(q_i^\circ) \geq 0 \right)$$

C'est déjà prouvé dans la preuve du théorème (4.7).

Le deuxième cas : $M \in M_a[\sigma(t \in T_c) > M' \in M_D$, le marquage interdit est accessible de M_a en passant par M_D.

Deux possibilités peuvent être discutées :

- **D'abord nous discutons $M \in SMMS(Q_{mc}, k)$:**

Selon Corollaire 4. 4⇒ nous cherchons la validité de l'inégalité (4.16) :

$$\left(-(k+1) + \begin{bmatrix} 1.\{E(q_1^\circ)\} \\ \dots \\ 1.\{E(q_i^\circ)\} \\ \dots \\ 1.\{E(q_{mc}^\circ)\} \end{bmatrix}^t . \begin{bmatrix} M_0(q_1) + \sigma(^\circ q_1) - E(q_1^\circ) \\ \dots \\ M_0(q_i) + \sigma(^\circ q_i) - E(q_i^\circ) \\ \dots \\ M_0(q_{mc}) + \sigma(^\circ q_{mc}) - E(q_{mc}^\circ) \end{bmatrix} \geq 0 \right)$$

Deux possibilités :

a. $E(q^\circ)$ est définie pour toutes les places critiques⇒ $\lambda(q_i) \neq \{\phi\}$.

Selon Corollaire 4.3 ⇒ le franchissement de q_i° conduit à un marquage interdit, mais dans ce cas la fonction $E(q_i^\circ)$ prend ce rôle, parce que q_i° est incontrôlable. Il existe un franchissement d'une transition $t \in \lambda(q_i)$ qui conduit à un marquage dangereux.

$$\left(-(k+1) + \sum_{i=1}^{mc} M_0(q_i) + \sum_{i=1}^{mc} \sigma(^\circ q_i) - \sum_{i=1}^{mc} E(q_i{}^\circ) \geq 0 \right)$$

$$\sum_{i=1}^{mc} M'(q_i) = \sum_{i=1}^{mc} M_0(q_i) + \sum_{i=1}^{mc} \sigma(^\circ q_i) - \sum_{i=1}^{mc} E(q_i{}^\circ) = k$$

$$-(k+1) + k \not\geq 0$$

Alors la valeur logique de (4.16) est fausse, et celle de (4.15) est déjà fausse⇒la valeur de l'expression (IV.14) est fausse⇒$U(t \in \lambda(q_i))$=0, i.e. le franchissement de $t \in \lambda(q_i)$ est interdit, en conséquence la loi de contrôle décrie par (IV.16) empêche le système à atteindre à un marquage interdit.

$$\left.\begin{matrix} (4.15)\ invalidée) \\ (4.16)\ inavlidée) \end{matrix}\right\} \Longrightarrow |(4.15)| \vee |(4.16)| = 0 \Longrightarrow U(t) = 0$$

$E(q_i^\circ)$ est indéfinie pour certains places critiques ou pour toutes les places critiques :

Il n'existe pas un moyen de contrôler $q°$.

- **Ensuite nous discutons $M \in SMMI(Q_{mc}, k)$:**

Selon Corollaire 4. 2$\Rightarrow$ nous cherchons la validité de l'inégalité (4.13) :

$$\left((k-1) - \sum_{i=1}^{mc} M_0(q_i) - \sum_{i=1}^{mc} S(°q_i) + \sum_{i=1}^{mc} \sigma(q_i°) \geq 0 \right)$$

Selon Corollaire 4. 1$\Rightarrow$ le franchissement de $°q_i$ conduit à un marquage interdit, mais dans ce cas la fonction $S(°q_i)$ prend ce rôle, parce que $°q_i$ est incontrôlable. Il existe un franchissement d'une transition $t \in \Gamma(q_i)$ qui conduit à un marquage dangereux.

$$\sum_{i=1}^{mc} M'(q_i) = \sum_{i=1}^{mc} M_0(q_i) + \sum_{i=1}^{mc} S(°q_i) - \sum_{i=1}^{mc} \sigma(q_i°) = k$$

$$(k-1) - k \not\geq 0$$

Alors la valeur logique de (4.15) est fausse, et celle de (4.16) est déjà fausse$\Rightarrow$ La valeur de l'expression (IV.14) est fausse$\Rightarrow U(t \in \Gamma(q_i))$=0, i.e. le franchissement de $t \in \Gamma(q_i)$ est interdit, en conséquence la loi de contrôle décrite par (IV.14) empêche le système d'atteindre un marquage interdit.

$$\left.\begin{matrix} (4.15)\ invalidée \\ (4.16)\ inavlidée \end{matrix}\right\} \Longrightarrow |(4.15)| \vee |(4.16)| = 0 \Longrightarrow U(t) = 0$$

□

Théorème 4. 16: Pour un graphe d'événements partiellement contrôlable soumis à une CEM-M, la loi de contrôle décrite par l'expression IV.14, est maximale permissive.

Preuve :

Pour être maximale permissive, il faut valider les deux conditions (**la permissivité du** superviseur.

Définition 4. 1).

1. Aucun état interdit ou dangereux, $M \in (M_D \cup M_f)$, n'est atteignable

Cette condition est validée par le théorème 4.15.

2. Aucun état légal $M \in M_L$ ne doit être interdit.

Nous supposons que: $\exists M' \in M_a : M \in M_a[\sigma(t) > M' \text{ où } U(t) = 0$

$U(t)$ est interdit $\Rightarrow$ la valeur logique de l'expression (IV.14) est égale à zéro $\Rightarrow$

$M' \notin M_a$ (car les deux inégalités décrivent l'ensemble de marquages admissibles M_a), mais ceci contredit l'hypothèse, et prouve le résultat du théorème:

$\nexists M' \in M_a : M \in M_a[\sigma(t) > M' \text{ où } U(t) = 0$

□

4.6 Algorithme de construction du superviseur

Dans cette partie du travail, nous proposons l'algorithme pour construire un superviseur pour les graphes d'événements partiellement contrôlables soumis aux spécifications modélisées par CEM, CEM-Où, CEM-Et ou CEM-M :

Hors ligne :

1. Définir l'ensemble de places critiques Q_{mc}
2. Pour chaque $q_i \in Q_{mc}$:
    a. Le marquage initial $M_0(q_i)$, la transition d'entrée $°q$ et la transition de sortie $q°$
    b. Définir $Z(q_i), L(q_i)$
    c. Définir $\lambda(q_i), \Gamma(q_i)$
    d. Calculer $td(M_0, t \in \lambda(q_i), q°)$, $td(M_0, t \in \Gamma(q_i), °q)$
    e. Calculer $E(q_i°)$, $S(°q_i)$
3. Vérifier si l'état initial du système appartient aux états admissibles ou non. Si la contrainte est de type :
    a. CEM, Calculer (IV.11)
    b. CEM-Ou, Calculer (IV.12)
    c. CEM-Et, Calculer (IV.13)
    d. CEM-M, Calculer (IV.14)
4. Si $M_0 \in (M_f \cup M_D)$: le problème d'état interdit n'a pas de solution.

En ligne :

1. Pour chaque franchissement $t \in T_{uc}$:
    - Si $t = q°$ respectivement $t = °q$, mettre à jour $E(q_i°)$ respectivement $S(°q_i)$
    - Sinon, faire rien
2. Avant de franchissement $t \in T_c$:

- Si $t \in \lambda(q_i)$, respectivement $t \in \Gamma(q_i)$, mettre à jour $E(q_i°)$ respectivement $S(°q_i)$
- Si la contrainte est de type :
  a. CEM, Calculer (IV.11)
  b. CEM-Ou, Calculer (IV.12)
  c. CEM-Et, Calculer (IV.13)
  d. CEM-M, Calculer (IV.14)

4.7 Exemple illustratif

Dans la figure 4.4, nous considérons que le réseau est soumis séparément à une CEM, ensuite à une CEM-Ou, puis à une CEM-Et, et à la fin à une CEM-M comme suit :

CEM-Ou: $\mathrm{M}_{\mathrm{L(Ou)}} = M(q_3) \neq 2 \vee M(q_7) \neq 1$

CEM-Et: $\mathrm{M}_{\mathrm{L(Et)}} = M(q_3) \neq 2 \ \wedge \ M(q_7) \neq 1$

CEM-M: $\mathrm{M}_{\mathrm{L(CEM-M)}} = M(q_3) + \ M(q_7) \neq 3$

Nous étudions pour l'exemple, la possibilité de franchir la séquence t_1, t_6, t_3, t_3, t_7

Hors ligne :

1. Définir l'ensemble de places critiques Q_{mc}
   Q_{mc}={p_3,p_7}

2. Pour chaque $q_i \in Q_{mc}$:
   a. Le marquage initial, la transition d'entrée et la transition de sortie de chaque place critique
      $M_0(p_3)$=1, $°q_1$=t_3, $q_1^\circ = t_6$
      $M_0(p_7)$=2, $°q_7$=t_7, $q_7^\circ = t_8$
   b. Définir $\lambda(q_i)$,$\Gamma(q_i)$
      $\lambda(p_3)$={ϕ}, $\Gamma(p_3)$= {ϕ}
      $\lambda(\ p_7)$= {t_1}, $\Gamma(\ p_7)$= {t_2, t_6}
   c. Calculer $td(M_0, t \in \lambda(q_i), q°)$, $td(M_0, t \in \Gamma(q_i), °q)$
      $td(M_0\ , t_1, t_8)$=1, $td(M_0\ , t_2, t_7)$=2, $td(M_0\ , t_6, t_7)$=0
   d. Calculer $E(q_i°)$, $S(°q_i)$
      $E(t_8)$=1, $S(t_7)$=0

En ligne :

Pour CEM-Ou:

t	$\Gamma(\ p_7)$	$\lambda(\ p_7)$	$S(t_7)$	$E(t_8)$	$\sigma(\ t_7)$	$\sigma(t_8)$	|(4.13)|	|(4.14)|	|(4.13)|∨ |(4.14)|
t_1	∉	∈	0	2	0	0	0	0	0

t	$\sigma(t_3)$	$\sigma(\ t_6)$	|(4.13)|	|(4.14)|	|(4.13)|∨ |(4.14)|
t_1	0	0	1	0	1

|(4.13)|∨ |(4.14)|	|(4.13)|∨ |(4.14)|	*IV.12*	σ(*t*)
0	1	1	σ(t_1)=1

t	Γ(p_7)	λ(p_7)	*S*(t_7)	*E*(t_8)	σ(t_7)	σ(t_8)	|(4.13)|	|(4.14)|	|(4.13)|∨ |(4.14)|
t_6	∈	∉	1	2	0	0	0	0	0

t	σ(t_3)	σ(t_6)	|(4.13)|	|(4.14)|	|(4.13)|∨ |(4.14)|
t_6	0	1	1	0	1

|(4.13)|∨ |(4.14)|	|(4.13)|∨ |(4.14)|	*IV.12*	σ(*t*)
0	1	1	σ(t_6)=1

t	Γ(p_7)	λ(p_7)	*S*(t_7)	*E*(t_8)	σ(t_7)	σ(t_8)	|(4.13)|	|(4.14)|	|(4.13)|∨ |(4.14)|
t_3	∉	∉	1	2	0	0	0	0	0

t	σ(t_3)	σ(t_6)	|(4.13)|	|(4.14)|	|(4.13)|∨ |(4.14)|
t_3	1	1	1	0	1

|(4.13)|∨ |(4.14)|	|(4.13)|∨ |(4.14)|	*IV.12*	σ(*t*)
0	1	1	σ(t_3)=1

t	Γ(p_7)	λ(p_7)	*S*(t_7)	*E*(t_8)	σ(t_7)	σ(t_8)	|(4.13)|	|(4.14)|	|(4.13)|∨ |(4.14)|
t_3	∉	∉	1	2	0	0	0	0	0

t	σ(t_3)	σ(t_6)	|(4.13)|	|(4.14)|	|(4.13)|∨ |(4.14)|
t_3	2	1	0	0	0

|(4.13)|∨ |(4.14)|	|(4.13)|∨ |(4.14)|	*IV.12*	σ(*t*)
0	0	0	σ(t_3)=0

Nous remarquons que le superviseur empêche t_3 d'être franchie à ce moment. Si on veut franchir la séquence : t_1, t_6, t_3, t_6, $_{2*}t_7$, $_{2*}t_3$

t	Γ(p_7)	λ(p_7)	*S*(t_7)	*E*(t_8)	σ(t_7)	σ(t_8)	|(4.13)|	|(4.14)|	|(4.13)|∨

									|(4.14)
t_6	$\in$	$\notin$	2	2	0	0	0	0	0

t	$\sigma(t_3)$	$\sigma(t_6)$	|(4.13)|	|(4.14)|	|(4.13)|$\vee$ |(4.14)|
t_6	$\notin$	2	1	0	1

|(4.13)|$\vee$ |(4.14)|	|(4.13)|$\vee$ |(4.14)|	*IV.12*	$\sigma(t)$
0	1	1	$\sigma(t_6)$=1

t	$\Gamma(p_7)$	$\lambda(p_7)$	$S(t_7)$	$E(t_8)$	$\sigma(t_7)$	$\sigma(t_8)$	|(4.13)|	|(4.14)|	|(4.13)|$\vee$ |(4.14)|
${}_{2*}t_7$	$\notin$	$\notin$	0	2	2	0	0	1	1

t	$\sigma(t_3)$	$\sigma(t_6)$	|(4.13)|	|(4.14)|	|(4.13)|$\vee$ |(4.14)|
${}_{2*}t_7$	0	2	1	0	1

|(4.13)|$\vee$ |(4.14)|	|(4.13)|$\vee$ |(4.14)|	*IV.12*	$\sigma(t)$
1	1	1	$\sigma({}_{2*}t_7)$=1

t	$\Gamma(p_7)$	$\lambda(p_7)$	$S(t_7)$	$E(t_8)$	$\sigma(t_7)$	$\sigma(t_8)$	|(4.13)|	|(4.14)|	|(4.13)|$\vee$ |(4.14)|
${}_{2*}t_3$	$\notin$	$\notin$	1	2	2	0	0	1	1

t	$\sigma(t_3)$	$\sigma(t_6)$	|(4.13)|	|(4.14)|	|(4.13)|$\vee$ |(4.14)|
${}_{2*}t_3$	2	2	0	0	0

|(4.13)|$\vee$ |(4.14)|	|(4.13)|$\vee$ |(4.14)|	*IV.12*	$\sigma(t)$
0	1	0	$\sigma({}_{2*}t_3)$=1

Pour CEM-Et:

Nous vérifions l'état initial pour voir s'il est admissible ou non.

M_0	$S(t_7)$	$E(t_8)$	$\sigma(t_7)$	$\sigma(t_8)$	|(4.13)|	|(4.14)|	|(4.13)|$\vee$ |(4.14)|
p_7	0	1	0	0	0	0	0

M_0	$\sigma(t_3)$	$\sigma(t_6)$	|(4.13)|	|(4.14)|	|(4.13)|$\vee$ |(4.14)|
P_3	0	0	1	0	1

|(4.13)|∨ |(4.14)|	|(4.13)|∨ |(4.14)|	*IV.13*	M_0
0	1	0	M_f

Alors, le franchissement de t_1 est interdit ; en plus l'état initial appartient à l'ensemble des états interdits.

Pour CEM-M:

Nous vérifions l'état initial pour voir s'il est admissible ou non.

$$\left((\boldsymbol{k}-\mathbf{1})-\sum_{i=1}^{mc}\boldsymbol{M_0}(\boldsymbol{q_i})-\sum_{i=1}^{mc}\boldsymbol{S}(^{\circ}\boldsymbol{q_i})+\sum_{i=1}^{mc}\boldsymbol{\sigma}(\boldsymbol{q_i}^{\circ})\geq\mathbf{0}\right) \qquad (\mathbf{3.19})$$

(3-1)-(1+2)-(0+0)+(0+0)=-1⇒|(3.19)|=0

$$\left(-(\boldsymbol{k}+\mathbf{1})+\begin{bmatrix}\mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q_1^{\circ}})\}\\ \dots\\ \mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q_i^{\circ}})\}\\ \dots\\ \mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q_{mc}^{\circ}})\}\end{bmatrix}^{\boldsymbol{t}}.\begin{bmatrix}\boldsymbol{M_0}(\boldsymbol{q_1})+\boldsymbol{\sigma}(^{\circ}\boldsymbol{q_1})-\boldsymbol{E}(\boldsymbol{q_1^{\circ}})\\ \dots\\ \boldsymbol{M_0}(\boldsymbol{q_i})+\boldsymbol{\sigma}(^{\circ}\boldsymbol{q_i})-\boldsymbol{E}(\boldsymbol{q_i^{\circ}})\\ \dots\\ \boldsymbol{M_0}(\boldsymbol{q_{mc}})+\boldsymbol{\sigma}(^{\circ}\boldsymbol{q_{mc}})-\boldsymbol{E}(\boldsymbol{q_{mc}^{\circ}})\end{bmatrix}\geq\mathbf{0}\right) \qquad (\mathbf{3.20})$$

$$-(3+1)+\begin{bmatrix}\mathbf{1}\\ \mathbf{1}\end{bmatrix}^{\boldsymbol{t}}.\begin{bmatrix}\mathbf{1}+\mathbf{0}-\mathbf{0}\\ \mathbf{2}+\mathbf{0}-\mathbf{1}\end{bmatrix}=-\mathbf{2}\Rightarrow|(3.20)|=0$$

⇒|(IV.14)|=0⇒ l'état initial appartient à l'ensemble des états interdits.

4.8 Conclusion

Dans ce chapitre, nous avons proposé une technique de SdC par supervision pour les systèmes à événements discrets, modélisés par le graphe d'événements. Cette technique est capable de restreindre le comportement de système pour respecter des spécifications de type d'état interdit qui est modélisé par CEM.

Nous avons également formalisé une solution pour résoudre le problème des transitions incontrôlables. En présence de ce types de transitions, l'objectif du superviseur est d'éviter les marquages interdits et dangereux. L'ensemble de marquage dangereux et des marquages légaux à partir desquels le système attient un marquage interdit en franchissant juste des transitions incontrôlables. Pour résoudre ce problème, nous avons distingué deux types de transitions contrôlables Γ(q) et λ(q). En utilisant ces types, nous avons développé deux fonctions pour calculer la loi de contrôle. La technique proposée dans ce chapitre nécessite de contrôler et d'observer l'arrivée et le départ des jetons des places critiques.

Nous avons montré que cette technique produit un superviseur maximal permissif, où il n'interdit que l'accès aux marquages interdits ou dangereux.

Dans le chapitre suivant, nous allons améliorer cette technique pour résoudre le problème des transitions inobservables. Dans ce cas, la technique sera plus complexe où nous allons proposer des fonctions pour estimer la distribution des jetons dans les places critiques.

Chapitre 5

SdC des Graphes d'événements partialement Observable

Résumé :

Dans ce chapitre, nous proposons une approche de SdC par supervision des graphes d'événements en présence de transitions inobservables et/ou incontrôlables pour résoudre le problème d'états interdits, modélisé par CEM. Notre approche repose sur les définitions et les concepts présentés dans le chapitre 4. Le superviseur proposé permet d'observer puis de contrôler le système. L'observation consiste à estimer l'emplacement des jetons et donc l'état du système. Cette étape est basée sur l'utilisation de la fonction d'observation proposée dans (Achour Z. a., 2004), et sur les fonctions S, E et la nouvelle fonction F. A la fin de ce chapitre nous présentons un algorithme pour construire le superviseur maximum permissif si ce dernier existe. Pour chaque place critique, l'algorithme divise l'ensemble de transitions observables en quatre sous-ensembles, et en observant leurs franchissements, l'algorithme met à jour les quatre fonctions, ces fonctions seront utilisées dans la loi de commande pour empêcher le franchissement de transitions quand il faut. Les résultats de cette partie ont fait l'objet de la publication (Atli, Septembre 2010).

5.1 Introduction

En présence de transitions inobservables, la synthèse des contrôleurs RdP devient une tâche plus difficile puisque l'état réel du système devient partiellement connu. L'approche de SdC par superviseur pour les graphes d'événements partiellement contrôlable proposée dans le chapitre précédent et publiée dans (Atli , et al., July 2010), (Atli, et al., April 2010), est améliorée dans ce chapitre pour inclure le problème d'inobservabilité.

Rappelons que l'ensemble des transitions $T = T_c \cup T_{uc}$ est formé de deux sous-ensembles ; T_c l'ensemble des *transitions contrôlables* et T_u l'ensemble des *transitions incontrôlables*. Le superviseur peut interdire le franchissement des transitions contrôlables mais pas celui des transitions incontrôlables. L'ensemble $T_{uc} = T_f \cup T_{uo}$ et aussi constitué de *transitions libres* T_f et de *transitions inobservables* T_{uo}. Il est possible qu'un procédé puisse avoir certains états atteignables à partir de transitions dont le franchissement ne peut pas être observé par le superviseur. Ces transitions sont dites inobservables. On note $T_o = T_c \cup T_f$ l'ensemble des transitions observables.

L'approche présentée dans ce chapitre traite la SdC des Graphes d'événements en présence de transitions inobservables et/ou incontrôlables pour résoudre le problème d'états interdits modélisé par des *Contraintes d'Exclusion de Marquages*.

Ce chapitre est organisé comme suit. Dans la section 2, nous proposons la SdC. Puis dans la section 3, nous illustrons cette synthèse sur un exemple, et nous terminons le chapitre par une conclusion.

5.2 SdC des graphes d'événements partiellement observable

L'approche développée dans cette section traite la SdC pour les graphes d'événements partiellement observables.

Le superviseur peut empêcher le franchissement des transitions contrôlables T_c quand il le faut, par contre il est impossible d'interdire le franchissement des transitions incontrôlables T_{uc}. En même temps, il ne peut qu'observer le franchissement des transitions observables T_o, contrairement aux franchissements des transitions inobservables T_{uo}.

Remarque 5. 1 : Il faut que l'état initial M_0 respect les deux conditions suivantes,

- $M_0 \notin M_f$ où M_f représente l'ensemble des marquages interdits,
- $M_0 \notin M_D$ où M_D représente l'ensemble des marquages dangereux.

Sinon, le problème d'état interdit n'a pas une solution.

Définition 5. 1: Soit $R^{uo}(N, M_0)$ l'ensemble des marquages atteignables à partir du marquage M_0 en ne franchissant que des transitions inobservable.

Pour un marquage interdit $M_f \in R(N,M_0)$, le manque d'information sur le franchissement des transitions inobservables rend plus difficile la détermination du marquage courant. Par conséquent, la SdC sera plus complexe. L'utilisation des fonctions *S* et *E* proposées dans le chapitre 3 n'est plus suffisante pour calculer le marquage courant. En présence de transitions inobservables, il est nécessaire de restreindre le comportement du système en évitant l'ensemble des marquages interdits et en plus l'ensemble de marquages dangereux décrits par la Définition 4.5.

Clairement les propriétés suivantes sont valables pour un réseau de Petri partiellement observable :

$M_a \subseteq M_L$, $M_D \neq \varphi$, $M_a = R(N,M_0) \backslash (M_f \cup M_D)$

Dans la suite, nous allons tout d'abord analyser la loi de commande pour une CEM. Dans ce sens, nous allons discuter séparément la loi de commande pour l'ensemble des marquages inférieurs $Dn(q, k)$ et l'ensemble des marquages supérieurs $Up(q, k)$. Ensuite nous analyserons cette loi pour toutes les autres extensions de CEM.

5.2.1 Système soumis à des spécifications de type CEM

Considérons un graphe d'événements partiellement observable soumis à une CEM. L'ensemble des marquages admissibles est :

$$M_L(q,k) = \{M \in R(N,M_0), M(q) \neq k\}$$

Nous avons vu que cet ensemble est constitué de deux sous-ensembles $Dn(q, k)$ et $Up(q, k)$. Nous allons tout d'abord analyser la loi de commande lorsque $M_0 \in Dn(q, k)$ puis $M_0 \in Up(q, k)$.

5.2.1.1 L'état initial appartient à l'ensemble des marquages inférieurs

Dans cette section nous considérons un graphe d'événements partiellement observable soumis à une CEM, où $M_0 \in Dn(q,k)$. Nous proposons quelques définitions pour une analyse structurelle des graphes d'événements.

Définition 5. 2: $Z^{out}(q)$ dénote la zone de sortie d'une place critique *q* contenant tous les chemins qui suivirent cette place et se termine par des transitions observables.

Dans la figure 5.1, $Z^{out}(p_7)=\{t_8, t_9, t_{10}, t_{11}, p_8, p_9, p_{10}\}$

Définition 5. 3: $\psi(q)$ dénote l'ensemble des transitions observables dans $Z^{out}(q)$. Formellement:

$$\Psi(q) = \{t \in Z^{out}(q) \cap T_o\}$$

Dans la figure 5.1, $\psi(p_7)$={ t_{10}, t_{11}}, $\psi(p_{11})$={t_{12}}, $\psi(p_9)$={t_{10}}.

□

D'après le corollaire 4.2, le marquage initial valide l'inégalité (4.13) et ne valide pas l'inégalité (4.14).

$$(k-1) - M_0(q) - S(^\circ q) + \sigma(q^\circ) \geq 0$$

Si $q^\circ \in T_{uo} \Rightarrow \sigma(q^\circ)$ est indéfini, autrement dit, on ne peut plus l'utiliser dans (4.13), mais on peut le replacer par la fonction $X(q^\circ)$ proposé par (Achour Z. a., 2004) :

$$\boldsymbol{X(q^\circ) = \begin{cases} \sigma(q^\circ) & \forall q^\circ \in T_o \\ \left(max_{t \in \Psi(q)}\left(\sigma(t) - td(M_0, q^\circ, t)\right)\right)^+ & \forall q^\circ \in T_{uo} \end{cases}}$$

Alors en présence de transitions inobservables, l'inégalité (4.13) prend la forme suivante :

$$\boldsymbol{(k-1) - M_0(q) - S(^\circ q) + X(q^\circ) \geq 0} \qquad \textbf{(5.1)}$$

Corollaire 5. 1: L'inégalité (5.1) modélise l'ensemble des marquages $M_{(5.1)}$ inclus dans l'ensemble de marquage admissibles M_a, et en même temps dans l'ensemble des marquages inférieurs à q.

$$M_{(5.1)} \equiv Dn(q, k) \backslash (M_f \cup M_D) \subseteq M_a$$

5.2.1.2 L'état initial appartient à l'ensemble de marquages supérieurs

Dans cette section nous considérons un graphe d'événements partiellement observable soumis à une CEM, où $M_0 \in Up(q,k)$. Nous proposons quelques définitions pour une analyse structurelle des graphes d'événements.

Définition 5. 4: $L^{out}(q)$ dénote la zone de sortie indirecte d'une place critique q, contenant tous les chemins qui suivent la transition d'entrée de cette place et se terminent par des transitions observables, en excluant tous les chemins contenant q.

Dans la figure 5.1, $L^{out}(p_6)$={p_{11}, t_{12}}

Définition 5. 5: Soit l'ensemble $\beta(q)$, l'ensemble des transitions observables dans $L^{out}(q)$, :

$$\beta(q) = \{t \in (L^{out}(q) \cap T_o)\}$$

Dans la figure 5.1, $\beta(p_{11})$={t_{10},t_{11}}, $\beta(p_6)$={t_{12}}, $\beta(p_9)$={ t_{11}}, $\beta(p_{10})$={ϕ}.

D'après le corollaire (4.4), le marquage initial valide l'inégalité (4.14) et ne valide pas l'inégalité (4.13).

$$\mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q}^{\circ})\ \boldsymbol{d\acute{e}fini}\}.(-(\boldsymbol{k}+\mathbf{1})+\boldsymbol{M_0}(\boldsymbol{q})+\boldsymbol{\sigma}(^{\circ}\boldsymbol{q})-\boldsymbol{E}(\boldsymbol{q}^{\circ})\geq\mathbf{0})$$

Si $^{\circ}q \in T_{uo} \Rightarrow \sigma(^{\circ}q)$ est indéfini, autrement dit, on ne peut plus l'utiliser dans (4.14). Dans certains cas, nous pouvons tout de même estimer le nombre minimal de franchissements de cette transition inobservable.

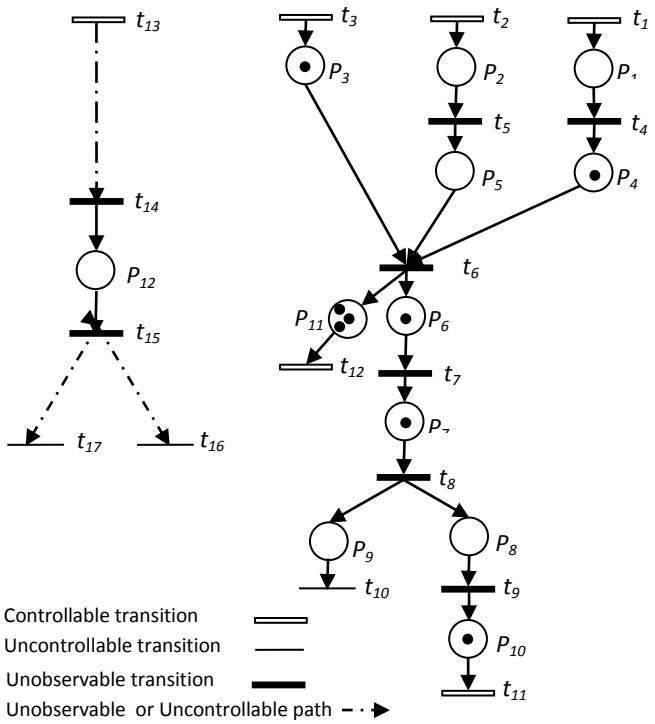


Figure 5. 1, ***Un graphe d'événements partiellement observable***

Définition 5. 6: Nous définissions la fonction $F(^{\circ}q)$ qui estime le nombre minimal de franchissements de $^{\circ}q$. Nous la formulons :

$$\boldsymbol{F}(^{\circ}\boldsymbol{q})=\begin{cases}\boldsymbol{D\acute{e}fini}\begin{cases}\boldsymbol{\sigma}(^{\circ}\boldsymbol{q}) & \forall^{\circ}\boldsymbol{q}\in\boldsymbol{T_o}\\ \left(\max_{\boldsymbol{t}\in\boldsymbol{\beta}(\boldsymbol{q})}\left(\boldsymbol{\sigma}(\boldsymbol{t})-\boldsymbol{td}(\boldsymbol{M_0},^{\circ}\boldsymbol{q},\boldsymbol{t})\right)\right)^{+} & \forall^{\circ}\boldsymbol{q}\in\boldsymbol{T_{uo}}\wedge\boldsymbol{\beta}(\boldsymbol{q})\neq\{\emptyset\}\end{cases}\\ \boldsymbol{Ind\acute{e}fini} \quad \forall^{\circ}\boldsymbol{q}\in\boldsymbol{T_{uo}}\wedge\boldsymbol{\beta}(\boldsymbol{q})=\{\emptyset\}\end{cases}$$

Preuve :

Il existe deux possibilités à discuter :

- $\forall\, ^{\circ}q \in T_o$: $\sigma(^{\circ}q)$ est défini, et on peut détecter directement le franchissement de $^{\circ}q$
- $\forall\, ^{\circ}q \in T_{uo}$: Nous avons deux cas,

1. Si $\beta(q)\neq\{\phi\}$, le nombre minimale de franchissements de $^\circ q$ peut être détecté comme suit :

   - $\forall t\in\beta(q)$ et $\sigma(t)<d(M_0,^\circ q,\ t)$: Le nombre de franchissements de $^\circ q$ ne peuvent pas être estimés.

   - $\forall t\in\beta(q)$ et $\sigma(t)>d(M_0,\ ^\circ q,\ t)$: Le nombre de franchissements de $^\circ q$ *est* au moins égale à $e=\sigma(t)-d(M_0,\ ^\circ q,\ t)$,

   - $\forall t_1,t_2\in\beta(q),\sigma(t_1)>d(M_0,^\circ q,t_1),\sigma(t_2)>d(M_0,^\circ q,t_2)$ et $e_1=\sigma(t_1)-d(M_0,^\circ q,t_1)< e_2=\sigma(t_2)-d(M_0,\ ^\circ q,\ t_2)$: $^\circ q$ *a été franchi au minimum* $e_2=max\ (e_1,\ e_2)$.

2. Si $^\circ q\in T_{uo}\wedge\beta(q)=\{\phi\}$, il n'y a aucun moyen d'estimer le franchissement de$^\circ q$.

Alors en présence de transitions inobservables, l'inégalité (4.14) devient :

$$\mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q}^\circ)\ \boldsymbol{d\acute{e}fini}\wedge\boldsymbol{F}(^\circ\boldsymbol{q})\ \boldsymbol{d\acute{e}fini}\}.(-(\boldsymbol{k}+\mathbf{1})+\boldsymbol{M_0}(\boldsymbol{q})+\boldsymbol{F}(^\circ\boldsymbol{q})-\boldsymbol{E}(\boldsymbol{q}^\circ)\geq\mathbf{0}) \quad \textbf{(5.2)}$$

Corollaire 5. 2: L'inégalité (5.2) modélise l'ensemble des marquages $M_{(5.2)}$ qui appartient à l'ensemble des marquages admissibles M_a, et en même temps à l'ensemble de marquages supérieurs à q.

$$\boldsymbol{M_{(5.1)}\equiv Up(q,\ k)\setminus(M_f\cup M_D)\subseteq M_a}$$

Corollaire 5. 3: Les deux inégalités (5.1) et (5.2) modélisent l'ensemble des marquages admissibles pour un graphe d'événements partiellement observable.

$$\boldsymbol{M_{(5.1)}\cup M_{(5.2)}=M_a}$$

Propriété 5. 1: Si les deux fonctions $\boldsymbol{F}(^\circ\boldsymbol{q})\ \boldsymbol{et}\ \boldsymbol{E}(\boldsymbol{q}^\circ)$ sont indéfinies, alors la valeur logique de l'inégalité (5.2) est zéro, |(4.2)|=0.

Théorème 5. 1: La loi de contrôle pour un graphe d'événements totalement observables qui respecte une spécification de CEM est:

$$\boldsymbol{U}(\boldsymbol{t}\in\boldsymbol{Tc})=\begin{cases}\mathbf{1}, & |(\mathbf{5.1})|\vee|(\mathbf{5.2})|=\mathbf{1}\\ \mathbf{0}, & \boldsymbol{sinon}\end{cases} \qquad \textbf{(}\boldsymbol{V}\textbf{.1)}$$

Preuve :

La preuve peut se faire en deux parties,

1. S'il existe un marquage interdit accessible sous cette loi de commande à partir d'un marquage admissible, alors cette loi n'est pas suffisante : $\exists M'\in M_f$: $M\in M_a[\sigma(t\in T_c)>M'\in M_f$ i.e. $M'(q)=k\Rightarrow$ La loi de commande (V.1) est vainne.

2. S'il existe un marquage dangereux accessible sous cette loi de contrôle à partir d'un marquage admissible, alors cette loi n'est pas suffisante : $\exists M'\in M_D$:

$M \in M_a[\sigma(t \in T_c) > M' \in M_D$, *i.e.* $M'(q)[\delta > M''(q) = k$, où δ est une séquence de franchissements inobservables et /ou incontrôlables $\Rightarrow$ La loi de contrôle (*V.*1) est vainne.

Dans ces deux cas, s'il existe un tel marquage, alors la loi de commande décrie en (*V.*1) est inutile.

Le premier cas : $M \in M_a[\sigma(t \in T_c) > M' \in M_f$ i.e. $M'(q) = k$, pour ce cas, le marquage interdit est accessible directement de M_a sans passer par M_D. Deux possibilités peuvent être discutées :

- ***M*(*q*)∈*Up*(*q*) :**

  Selon le Corollaire 4.3 $\Rightarrow t = q° \in T_c$.

  Selon Corollaire 4. 4$\Rightarrow$ nous cherchons la validité d'inégalité (5.2) :

$$1.\{E(q°)\ défini \wedge F(°q)\ défini\}.(-(k+1) + M_0(q) + F(°q) - E(q°) \geq 0)$$

La fonction $E(q°)$ est définie et $q° \in T_c \Rightarrow E(q°) = \sigma(q°)$

$$1.\{F(°q)\ défini\}.(-(k+1) + M_0(q) + F(°q) - \sigma(q°) \geq 0)$$

$$-(k+1) + M_0(q) + F(°q) - \sigma(q°) \geq 0$$

$M'(q) = M_0(q) + F(°q) - \sigma(q°) = k \in M_f$

$$-(k+1) + k \not\geq 0$$

Alors la valeur logique de (5.2) est fausse, et celle de (5.1) est aussi fausse $\Rightarrow$ La valeur de l'expression (*V.*1) est fausse $\Rightarrow$ $U(t \in T_c) = 0$, i.e. le franchissement de $t \in T_c$ est interdit, en conséquence, la loi de commande décrite par (*V.*1) empêche le système d'arriver à un marquage interdit.

$$\left.\begin{matrix}(5.1)\ invalidée) \\ (5.2)\ inavlidée)\end{matrix}\right\} \Longrightarrow |(5.1)| \vee |(5.2)| = 0 \Longrightarrow (V.1) = 0$$

- ***M*(*q*)∈*Dn*(*q*) :**

  Selon le Corollaire 4. 1$\Rightarrow t = °q \in T_c$.

  Selon le Corollaire 4. 2$\Rightarrow$ nous cherchons la validité de l'inégalité (5.1) :

$$(\boldsymbol{k-1}) - \boldsymbol{M_0(q)} - \boldsymbol{S}(°\boldsymbol{q}) + \boldsymbol{X}(\boldsymbol{q}°) \geq \boldsymbol{0}$$

La fonction $S(°q) = \sigma(°q)$

$$(\boldsymbol{k-1}) - \boldsymbol{M_0}(\boldsymbol{q}) - \boldsymbol{\sigma}(°\boldsymbol{q}) + \boldsymbol{X}(\boldsymbol{q}°) \geq \boldsymbol{0}$$

$M'(q)= M_0(q) + \sigma(°q) - X(q°) = k \in M_f$

$$(k-1) - k \not\geq 0$$

Alors la valeur logique de (5.1) est fausse, et celle de (5.2) est aussi fausse ⇒ La valeur de l'expression (V.1) est fausse ⇒ $U(t \in T_c)$=0, i.e. le franchissement de $t \in T_c$ est interdit, en conséquence, la loi de commande décrite par (V.1) empêche le système d'atteindre un marquage interdit.

$$\left.\begin{matrix}(5.1)\ invalidée\\(5.2)\ inavlidée\end{matrix}\right\} \Longrightarrow |(5.1)| \vee |(5.2)| = 0 \Longrightarrow (V.1) = 0$$

<u>Le deuxième cas :</u> $M \in M_a[\sigma(t \in T_c) > M' \in M_D$, *i.e.* $M'(q)[\delta > M''(q) = k$; le marquage interdit est accessible de M_a en passant par M_D. Deux possibilités peuvent être discutées :

- ***M*(*q*)∈*Up*(*q*) :**

Selon le Corollaire 4. 4⇒ nous cherchons la validité d'inégalité (5.2) :

$$1.\{E(q°)\ défini \wedge F(°q)\ défini\}.(-(k+1) + M_0(q) + F(°q) - E(q°) \geq 0)$$

Deux possibilités :

a. $E(q°)$ est défini ⇒ $\lambda(q) \neq \{\phi\}$.

$M'(q)= M_0(q) + F(°q) - E(q°) = k \in (M_f \cup M_D)$

$$-(k+1) + k \not\geq 0$$

Alors la valeur logique de (5.2) est fausse, et celle de (5.1) est aussi fausse ⇒ La valeur de l'expression (V.1) est fausse ⇒ $U(t \in \lambda(q))$=0, i.e. le franchissement de $t \in \lambda(q)$ est interdit, en conséquence, la loi de contrôle décrie par (V.1) empêche le système d'atteindre un marquage interdit.

$$\left.\begin{matrix}(5.1)\ invalidée\\(5.2)\ inavlidée\end{matrix}\right\} \Longrightarrow |(5.1)| \vee |(5.2)| = 0 \Longrightarrow (V.1) = 0$$

b. $E(q°)$ est indéfinie⇒ $\lambda(q)=\{\phi\}$.

Il n'existe pas un moyen de contrôler $q°$.

- ***M*(*q*)∈*Dn*(*q*) :**

Selon le Corollaire 4. 2⇒ nous cherchons la validité d'inégalité (5.1) :

$$(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(\boldsymbol{q})-\boldsymbol{S}(^{\circ}\boldsymbol{q})+\boldsymbol{X}(\boldsymbol{q}^{\circ})\geq\mathbf{0}$$

Selon le Corollaire 4. 1⇒ le franchissement de °q conduit à un marquage interdit, ⇒un franchissement d'une transition $t\in\Gamma(q)$ conduit à un marquage dangereux.

$$M'(q)= M_0(q)+\sigma(^{\circ}q)-\sigma(q^{\circ})=k\in(M_f\cup M_D)$$

$$(k-1)-k\not\geq 0$$

Alors la valeur logique de (5.1) est fausse, et celle de (5.2) est aussi fausse ⇒ La valeur de l'expression (V.1) est fausse ⇒ $U(t\in\Gamma(q))$=0, i.e. le franchissement de $t\in\Gamma(q)$ est interdit, en conséquence la loi de commande décrite par (V.1) empêche le système d'atteindre à un marquage interdit.

$$\left.\begin{matrix}(5.1)\ invalidée\\(5.2)\ inavlidée\end{matrix}\right\}\Longrightarrow|(5.1)|\vee|(5.2)|=0\Longrightarrow(V.1)=0$$

Théorème 5. 2: Pour un graphe d'événements partiellement contrôlable soumis à une CEM, la loi de commande décrite par l'expression V.1, est maximale permissive.

Preuve :

Si cette expression modélise une loi maximale permissive, il faut valider les deux conditions :

1. Aucun état interdit ou dangereux, $M\in(M_D\cup M_f)$, n'est atteignable

$$\forall t\in T, M\in M_a: M[t>M'\in\left(M_D\cup M_f\right)\Longrightarrow U(t)=0$$

Cette condition est déjà prouvée dans la preuve de théorème 5.1.

2. Aucun état admissible $M\in M_a$ ne doit être interdit.

   Nous supposons que $\exists M'\in M_a: M\in M_a[t>M'\ et\ U(t)=0.$

Le franchissement de t est interdit ⇒ la valeur logique de l'expression (V.1) est égale à zéro ⇒ |(5.1)|∨|(5.2)|=0 ⇒$M'\notin M_a$ (car les deux inégalités décrient l'ensemble de marquages admissibles M_a), mais ceci contredit l'hypothèse, et approuve le théorème, alors il n'existe pas de marquage M' où :

$$\exists M'\in M_a: M\in M_a[t>M'\ et\ U(t)=0$$

5.2.2 Système soumis à des spécifications de type CEM-Ou

Nous considérons dans cette section un graphe d'événements partiellement observable soumis à une CEM-Ou avec m_c places critiques. Chaque place critique q_i est indépendamment soumise à une CEM $M(q_i)\neq k_i$. Ces contraintes CEMs sont reliées par

la relation *Ou*. Ainsi, la loi de commande dans ce cas, consiste dans m_c sous lois de commande reliées par la relation *Ou* comme suit :

$$\boldsymbol{U(t \in Tc) = \begin{cases} 1, & \bigvee_{q_i \in Q_{mc}} \{|(5.1)|\vee|(5.2)|\} = 1 \\ 0, & sinon \end{cases}} \qquad (V.2)$$

5.2.3 Système soumis à des spécifications de type CEM-Et

Nous considérons dans cette section un graphe d'événements partiellement observable soumis à une CEM-Et avec m_c places critiques. Chaque place critique q_i est indépendamment soumise à une CEM $M(q_i) \neq k_i$. Ces contraintes CEMs sont reliées par la relation *Et*. Ainsi, la loi de commande dans ce cas, consiste dans m_c sous lois de commande reliées par la relation *Et* comme suit :

$$\boldsymbol{U(t \in Tc) = \begin{cases} 1, & \bigwedge_{q_i \in Q_{mc}} \{|(5.1)|\vee|(5.2)|\} = 1 \\ 0, & sinon \end{cases}} \qquad (V.3)$$

5.2.4 Système soumis à des spécifications de type CEM-M

Nous considérons dans cette section un graphe d'événements partiellement observable soumis à une CEM-M avec m_c places critiques.

$$M_{L(CEM-M)} = \sum_{i=1}^{mc} M(q_i) \neq k \supseteq M_a \equiv \left(\sum_{i=1}^{mc} Y(q_i) \leq k-1\right) \vee \left(\sum_{i=1}^{mc} M(q_i) \geq k_i + 1\right)$$

- $\boldsymbol{\left(\sum_{i=1}^{mc} Y(q_i) \leq k-1\right) \Leftrightarrow \left((k-1) - \sum_{i=1}^{mc} Y(q_i) \geq 0\right) \Rightarrow}$

$$\boldsymbol{\left((k-1) - \sum_{i=1}^{mc} M_0(q_i) - \sum_{i=1}^{mc} S(^\circ q_i) + \sum_{i=1}^{mc} X(q_i^\circ) \geq 0\right)} \qquad (5.3)$$

Corollaire 5. 4 : L'inégalité (5.3) modélise l'ensemble des marquages $M_{(5.3)}$ qui appartient à l'ensemble des marquages admissibles M_a, et en même temps à SMMI de Q_{mc}.

$$\boldsymbol{M_{(5.1)} \equiv SMMI(Q_{mc}, k) / (M_f \cup M_D) \subseteq M_a}$$

- $\boldsymbol{\left(\sum_{i=1}^{mc} M(q_i) \geq k+1\right) \Leftrightarrow \left(-(k+1) + \sum_{i=1}^{mc} M(q_i) \geq 0\right) \Rightarrow}$

$$\left(-(\boldsymbol{k}+\mathbf{1})+\begin{bmatrix}\mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q}_1^\circ)\wedge\boldsymbol{F}(^\circ\boldsymbol{q})\}\\ \dots\\ \mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q}_i^\circ)\wedge\boldsymbol{F}(^\circ\boldsymbol{q}_i)\}\\ \dots\\ \mathbf{1}.\{\boldsymbol{E}(\boldsymbol{q}_{mc}^\circ)\wedge\boldsymbol{F}(^\circ\boldsymbol{q}_{mc})\}\end{bmatrix}^t.\begin{bmatrix}\boldsymbol{M_0}(\boldsymbol{q_1})+\boldsymbol{\sigma}(^\circ\boldsymbol{q_1})-\boldsymbol{E}(\boldsymbol{q_1^\circ})\\ \dots\\ \boldsymbol{M_0}(\boldsymbol{q_i})+\boldsymbol{\sigma}(^\circ\boldsymbol{q_i})-\boldsymbol{E}(\boldsymbol{q_i^\circ})\\ \dots\\ \boldsymbol{M_0}(\boldsymbol{q_{mc}})+\boldsymbol{\sigma}(^\circ\boldsymbol{q_{mc}})-\boldsymbol{E}(\boldsymbol{q_{mc}^\circ})\end{bmatrix}\geq\mathbf{0}\right)\qquad(\mathbf{5.4})$$

Corollaire 5. 5: L'inégalité (5.4) modélise l'ensemble des marquages $M_{(5.4)}$ inclus dans l'ensemble des marquages admissibles M_a, et en même temps dans SMMS de Q_{mc}.

$$\boldsymbol{M_{(4.1)}\equiv \text{SMMS}(Q_{mc}, k)\setminus (M_f\cup M_D)\subseteq M_a}$$

Alors, la loi de commande pour ce cas est :

$$\boldsymbol{U(t\in Tc)}\begin{cases}\mathbf{1}, & |(\mathbf{5.3})|\vee|(\mathbf{5.4})|=\mathbf{1}\\ \mathbf{0} & \boldsymbol{sinon}\end{cases}\qquad(\boldsymbol{V.4})$$

5.2.5 La loi de commande dans le cas général

L'algorithme suivant présente la SdC pour les graphes d'événements partiellement observables et/ou contrôlables soumis à des spécifications de type CEM, CEM-Où, CEM-Et, CEM-M ou une classe de CEMs C-CEM:

Hors ligne :

1. Définir l'ensemble des places critiques Q_{mc}
2. Pour chaque $q_i\in Q_{mc}$:
   a. Identifier le marquage initial $M_0(q_i)$, la ou les transitions d'entrée $^\circ q_i$ et la ou les transitions de sortie q_i°
   b. Définir $Z(q_i)$, $Z^{out}(q_i)$, $L(q_i)$ et $L^{out}(q_i)$
   c. Définir $\Gamma(q_i)$, $\Psi(q_i)$, $\lambda(q_i)$ et $\beta(q_i)$
   d. Calculer $td(M_0, t\in\Gamma(q_i), {}^\circ q)$, $td(M_0, q^\circ, t\in\Psi(qi))$, $td(M_0, t\in\lambda(q_i), q^\circ)$ et $td(M_0, {}^\circ q, t\in\beta(qi))$
   e. Calculer $S(^\circ q_i)$, $X(q_i^\circ)$, $E(q_i^\circ)$ et $F(^\circ q_i)$
3. Vérifier si l'état initial du système appartient à un état admissible ou non. Si la contrainte est de type :
   a. CEM, Calculer (V.1)
   b. CEM-Ou, Calculer (V.2)
   c. CEM-Et, Calculer (V.3)
   d. CEM-M, Calculer (V.4)
4. Si $M_0\in(M_f\cup M_D)$: le problème d'état interdit n'a pas une solution.

En ligne :

5. Pour chaque franchissement $t\in T_o$:
   Si $t\in\Gamma(q_i)$, mettre à jour $S(^\circ q_i)$.
   Si $t\in\lambda(q_i)$, mettre à jour $E(q_i^\circ)$.
   Si $t\in\psi(q_i)$, mettre à jour $X(q_i^\circ)$.
   Si $t\in\beta(q_i)$, mettre à jour $F(^\circ q_i)$.

6. Avant le franchissement de $t \in T_c$, si la contrainte est de type :
    a. CEM, Calculer (V.1)
    b. CEM-Ou, Calculer (V.2)
    c. CEM-Et, Calculer (V.3)
    d. CEM-M, Calculer (V.4)
7. Sinon, ne fait rien

5.3 Exemple illustratif

Considérons le réseau de la figure 5.1 et une contrainte CEM : $M(p_6) \neq 2$. Considérons le franchissement de t_2 à partir de M_0.

Nous appliquons l'algorithme de la section précédente :

1. Définir l'ensemble des places critiques Q_{mc}
   $Q_{mc}=\{p_6\}$

2. Pour chaque $q_i \in Q_{mc}$:
    a. Identifier le marquage initial $M_0(q_i)$, la ou les transitions d'entrée $^\circ q_i$ et la ou les transitions de sortie q_i°

       $M_0(p_6)=1$, $^\circ p_6 = t_6$, $p_6^\circ = t_7$

    b. Définir $Z(q_i)$, $Z^{out}(q_i)$, $L(q_i)$ et $L^{out}(q_i)$
       $Z(p_6)=\{t_1, t_2, t_3, t_4, t_5, t_6, p_1, p_2, p_3, p_4, p_5\}$
       $Z^{out}(p_6) =\{t_7, t_8, t_9, t_{10}, t_{11}, p_7, p_8, p_9, p_{10}\}$
       $L(p_6) =\{\phi\}$
       $L^{out}(p_6) =\{t_{12}, p_{11}\}$

    c. Définir $\Gamma(q_i)$, $\Psi(q_i)$, $\lambda(q_i)$ et $\beta(q_i)$
       $\Gamma(p_6)=\{t_1, t_2, t_3\}$
       $\Psi(p_6)=\{t_{10}, t_{11}\}$
       $\lambda(p_6)=\{\phi\}$
       $\beta(p_6)=\{t_{12}\}$

    d. Calculer $td(M_0, t \in \Gamma(q_i), {}^\circ q)$, $td(M_0, q^\circ, t \in \Psi(qi))$, $td(M_0, t \in \lambda(q_i), q^\circ)$ et $td(M_0, {}^\circ q, t \in \beta(qi))$
       $td(M_0, t \in \Gamma(p_6), t_6)$:
          $td(M_0, t_1, t_6) = 1$
          $td(M_0, t_2, t_6) = 0$
          $td(M_0, t_3, t_6) = 1$
       $td(M_0, t_7, t \in \Psi(p_6))$,
          $td(M_0, t_7, t_{10}) = 1$
          $td(M_0, t_7, t_{11}) = 2$
       $td(M_0, t_6, t \in \beta(p_6))$
          $td(M_0, t_6, t_{12}) = 3$

e. Calculer $S(°q_i)$, $X(q_i°)$, $E(q_i°)$ et $F(°q_i)$
$S(t_6)=0$
$X(t_7)=0$
$E(t_7)$ est indéfinie
$F(t_6)=0$

3. Vérifier si l'état initial du système est un état admissible ou non. Si la contrainte est de type :
   i. CEM, Calculer ($V.1$)

M_0	$S(t_6)$	$E(t_7)$	$X(t_7)$	$F(t_6)$	|(4.1)|	|(4.2)|	|(4.1)|∨ |(4.2)|
p_6	0	X	0	0	1	X	1

$M_0 \notin (M_f \cup M_D)$

En ligne :

1. Pour chaque franchissement $t \in T_o$:

   $t_2 \in T_o$ et $t_2 \in \Gamma(p_6) \Rightarrow S(t_6)=1$

2. Avant le franchissement de $t \in T_c$, $t_2 \in T_c$, si la contrainte est de type :

   i. CEM, Calculer ($V.1$)

t	$\Gamma(p_6)$	$\Psi(p_6)$	$S(t_6)$	$X(t_7)$	|(4.1)|	|(4.2)|	|(4.1)|∨ |(4.2)|
t_2	$\in$	$\notin$	1	0	0	0	0

Le système ne peut pas franchir t_2 jusqu'à ce que la fonction $X(t_6)$ soit mise à jour en franchissant t_{10} ou t_{11}.

5.4 Conclusion

Dans ce chapitre, nous avons présenté un algorithme pour calculer la loi de control pour les systèmes à événements discrets partiellement observables modélisés par des graphes d'événements.

En présence de ce types de transitions incontrôlables, l'objectif du superviseur est d'éviter les marquages interdits ou dangereux. L'ensemble des marquages dangereux est un ensemble de marquages légaux à partir desquels le système atteint un marquage interdit en franchissant juste des transitions incontrôlables. Pour résoudre ce problème, nous avons distingué deux types de transitions contrôlables, et deux types de transitions observables. Pour chaque place critique, nous avons définit ces types : deux sortes pour la transition d'entrée et deux types pour la transition de sortie. Ensuite, nous avons développé quatre fonctions pour estimer la distribution de

jetons dans les places critiques. Par ces fonctions, nous avons proposé les lois de contrôle adaptables à chaque type de CEM.

Nous avons vu que la technique proposé dans ce chapitre, estime et contrôle les mouvements des jetons arrivants et sortants des places critiques. Nous avons montré que cette technique produit un superviseur maximal permissif, où il n'interdit que l'accès aux marquages interdits ou dangereux.

Dans le chapitre suivant, nous allons développer une technique pour analyser une loi de contrôle pour des systèmes temporisés où l'aspect de temps joue un rôle très important.

Chapitre 6

SdC de Graphes d'événements Place-Temporisée

Résumé :

Ce chapitre développe une approche de SdC pour les Systèmes à Evénements Discrets modélisés par des graphes d'événements temporisés où le temps est associé aux places. Cette approche résout le problème d'états interdits caractérisé par CEM. Une technique efficace pour construire le superviseur est présentée en prenant en considération la nature incontrôlable et / ou inobservable de certains événements. Le temps rend notre politique de commande plus permissive en permettant au système d'atteindre des états supplémentaires considérés comme des états dangereux sans la prise en compte du temps. Cette méthode suppose que le marquage initial soit connu. A partir de ce marquage, nous construirons les tables de temps qui montrent la chronologie de la distribution des jetons sur toutes les places. Ces tables sont indispensables pour construire ensuite les Tables de Loi de Commande TLC. En analysant les informations dans ces tables, le superviseur peut déterminer le moment de son intervention. Les résultats de cette partie font l'objet de la publication (Atli, et al., March 2011).

6.1 Introduction

Il est parfois utile d'intégrer la durée des activités dans le modèle de réseaux de Petri afin d'étudier les performances de certains systèmes. Dans les approches précédentes, aucune durée n'est associée au franchissement de transitions et/ou au temps de séjour des jetons dans les places. Cependant il existe certains systèmes à événements discrets dont l'évolution se base sur le temps. Les deux principales extensions du temps dans les réseaux de Petri sont les réseaux de Petri temporels et les réseaux de Petri temporisés. Dans un réseau de Petri temporel, une transition peut être franchie dans un intervalle de temps. Dans les réseaux de Petri temporisés, la transition est franchie après un temps de séjour associé aux places (les réseaux de Petri place-temporisés RdP-PTs), ou aux transitions (les réseaux de Petri transition-temporisés RdP-TTs). Le temps peut être une variable déterministe, aléatoire modélisé par des fonctions de densité de probabilité, ou floue décrite par des intervalles flous.

La première approche des RdP-TTs a été proposée par (Ramchandani, 1974). Le modèle proposé associe à chaque transition un paramètre temporel, dont la sémantique correspond à la durée de tir de cette transition. Plus tard, un modèle de RdP-PTs a été présenté par (Sifakis, 1980), où un paramètre temporel est associé à chaque place. La sémantique de ce paramètre correspond au temps de séjour minimum d'un jeton dans une place (temps d'indisponibilité). Les deux sous-classes RdP-PT et RdP-TT sont expressément équivalentes.

Compte tenu de cette équivalence notre approche est basée sur les RdP-PTs, et les résultats peuvent être appliqués sur RdP-TTs. Nous considérons une évolution du réseau à vitesse maximale c-à-d que le franchissement des transitions se produit au plus tôt, et tous les jetons disponibles franchissent à ce moment, autrement un seul état est accessible en fonction du temps depuis un état donné (Brahms, 1983) et (David & Alla, 1992).

Ce chapitre est organisé comme suit : Dans la deuxième section, nous rappelons la définition des graphes d'événements place-temporisés. Puis dans la troisième section, nous exposons une analyse de loi de contrôle pour des graphes d'événements place-temporisés soumis à spécification de type CEM en utilisant la méthode proposé dans (Achour Z. a., 2008). Dans cette section, nous distinguons deux cas : le premier cas est l'analyse de cette loi à partir de l'état initial sans franchir des transitions d'influence, le deuxième cas est l'analyse de cette loi, en franchissant des transitions d'influence. Ensuite et dans la quatrième section, nous développons cet analyse pour résoudre le problème d'état interdit modélisé par CEM-Et ou CEM-Ou. A la fin de ce chapitre, nous présentons un exemple réel pour montrer l'applicabilité de notre méthode.

6.2 Graphes d'événements P-temporisés

Un graphe d'événements P-temporisé est une classe de RdP-PTs. Dans ce chapitre nous associons le paramètre *temps* aux places. Un graphe d'événements P-temporisé est défini tel que (N, H, ϑ), où N représente un graphe d'événement normal. H est la fonction de délais associés à chaque place $H=\{h_i : i=1, 2, \ldots, m\}$ où h_i représente le paramètre *temps* associé à la place p_i (le temps dont le jeton arrivant à cette place, a besoin d'être disponible). θ_j spécifie le moment de franchissement d'une transition t_i.

Un jeton arrivant à la place p_i au moment θ_j devient disponible après h_i unité de temps. Une transition sera valable s'il existe au moins un jeton disponible dans toutes les places d'entrée de cette transition. Alors un jeton dans une place p_i peut avoir deux cas possibles : disponible ou non, et le marquage curent $M=M_d+M_{ud}$ consiste en deux sous-marquages, M_d comprend tous les jetons disponibles, et M_{ud} comprend ceux non disponibles.

Règle 6. 1: Une transition t est franchissable si $M_d(pi) \geq Pre(p_i, t)$.

6.3 SdC de graphes d'événements P-temporisés soumis à CEM

Le travail présenté dans ce chapitre représente une extension des travaux présentés dans les chapitres précédents en intégrant l'aspect temps. Les états interdits sont modélisés par des CEM et le modèle est représenté par un graphe d'événement P-temporisé. La SdC est basée sur l'approche proposée dans (Achour Z. a., 2008) où le pire cas est noté $Y(q, M, \theta)$ et représente le nombre de jetons que la place critique q peut avoir au moment θ à partir d'un marquage connu M et en ne franchissant que des transitions incontrôlables. Le marquage de la place q ; $M(q)$ est influencé par le chemin de marquage $\pi(q)$ tel que $\pi(q) = t_\pi^c p_1 t_2 p_2 \ldots p_n t_{n+1} q$ où t_c représente la seule transition contrôlable (une transition d'influence) dans le chemin π.

Lorsqu'on considère les graphes d'événements P-temporisés qui évoluent à vitesse maximale, les jetons ne peuvent plus rester indéfiniment dans les places. Un jeton ne peut pas passer à une autre place avant un certain délai. Par conséquence, le comportement du système est plus contraint que celui des graphes d'événements normaux. Certains marquages dangereux ne sont plus considérés lorsqu'on construit le superviseur, car ils ne sont plus atteints.

Ainsi, en utilisant l'information sur le temps, cela nous permet de construire des superviseurs plus permissifs.

$$Y(q,M,\theta)=Y^{+}(q,M,\theta)-Y^{-}(q,M,\theta) \qquad \textbf{(6. 1)}$$

où $Y^+(q,M,\theta)$ (*resp.* $Y^-(q,M,\theta)$) représente le nombre de jetons arrivant (respectivement quittant) la place critique q au moment θ, tel que:

$$Y^+(q,M,\theta)= M(q)+M(p_n)_{\ if\ \theta\geq h(\pi n)}+\ldots+M(p_1)_{\ if\ \theta\geq h(\pi 1)}+\sigma(t_1,\ \theta-e-h_{(\pi 1)}) \qquad \textbf{(6. 2)}$$

$$Y^-(q,M,\theta)= M(q)_{\ if\ \theta\geq h(q)}+ M(p_n)_{\ if\ \theta\geq h(\pi n)+h(q)}+\ldots+M(p_1)_{\ if\ \theta\geq h(\pi 1)+h(q)}+\sigma(t_1,\ \theta-e-(h_{(\pi 1)}+h_{(q)})) \qquad \textbf{(6. 3)}$$

π_i représente le chemin $\pi(pi\rightarrow^{\circ}q)$, $\sigma(t_i,\tau)$ est le nombre de franchissements de t_i pendant la période τ, et e est le moment de franchir la transition d'influence.

Corollaire 6. 6 : la loi de commande pour un graphe d'événements P-temporisé soumis à une CEM est donnée tel que,

$$\boldsymbol{U(t\in Tc)=\begin{cases}1, & Y(q,M,\theta)\neq k\\ 0, & sinon\end{cases}} \qquad (VI.1)$$

Le superviseur n'interviendra pas dans le comportement du système si ce comportement respecte toujours la condition (*VI.*1), sinon le superviseur interdira le franchissement de la transition d'influence ou de la transition de sortie de la place critique.

6.3.1 Analyse de la loi de commande à l'état initial

Nous analysons ici deux cas d'études pour montrer que la loi de commande (VI.1) n'est pas suffisante pour assurer le respect des spécifications modélisées par des CEMs.

Cas d'étude 6. 1 : Considérons l'exemple représenté à la Figure 6. , où une place critique est soumise à une CEM : $M(p_3)\neq 2$. Les équations (6.1) et (6.2) prennent les formes suivantes :

$Y^+(p_3,M_0,\theta)= M(p_3)+M(p_1)_{\ if\ \theta\geq 4}+M(p_2)_{\ if\ \theta\geq 2}+\sigma(t_1,\ \theta-4)$

$Y^-(p_3,M_0,\theta)= M(p_3)_{\ if\ \theta\geq 3}+ M(p_1)_{\ if\ \theta\geq 7}+M(p_2)_{\ if\ \theta\geq 5}+\sigma(t_1,\ \theta-7)$

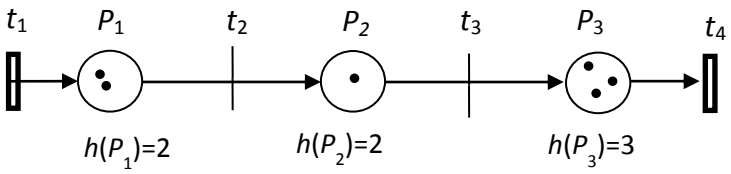


Figure 6. 1*, Un graphe d'événements P-temporisé soumis à une* CEM : $M(p_3)\neq 2$

Nous analysons les deux équations en utilisant des tables de temps pendant la période $[0,\ h_{(\pi)}+h_{(q)}=7]$.

Table 6. 1, $Y^+(p_3,M_0,\theta)$

θ	$M(p_3)$	$M(p_1)$ *if* $\theta\geq4$	$M(p_2)$ *if* $\theta\geq2$	$Y^+(p_3,M_0,\theta)$
0	3	0	0	3
1	3	0	0	3
2	3	0	1	4
3	3	0	1	4
4	3	2	1	6
5	3	2	1	6
6	3	2	1	6
7	3	2	1	6

Table 6. 2,$Y(p_3,M_0,\theta)$

θ	$M(p_3)$ *if* $\theta\geq3$	$M(p_1)$ *if* $\theta\geq7$	$M(p_2)$ *if* $\theta\geq5$	$Y(p_3,M_0,\theta)$
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	3	0	0	3
4	3	0	0	3
5	3	0	1	4
6	3	0	1	4
7	3	2	1	6

Table 6. 3, Table de Loi de Commande TLC pour p_3

θ	$Y^+(p_3,M_0,\theta)$	$Y(p_3,M_0,\theta)$	$Y(p_3,M_0,\theta)$	$U_{(t4)}$
0	3	0	3	1
1	3	0	3	1
2	4	0	4	1
3	4	3	1	1
4	6	3	3	1
5	6	4	2	0
6	6	4	2	0
7	6	6	0	1

En analysant la Table 6. 3, nous trouvons :

- Pendant $\theta \in[0,4]$, le superviseur permet le franchissement de t_4 le plus tôt possible (quand il y aura des jetons valables dans p_3).
- Quand θ=5, 6, le superviseur intervient parce que la loi de commande est fausse.

La Table 6. 4, montre les différents comportements du système (sans intervention du superviseur, et avec son intervention). En analysant cette table nous trouvons que :

- Sans le superviseur : le système arrive à l'état interdit quand θ=5, 6

- Avec le superviseur : En interdisant le franchissement de t_4 quand θ=5, 6. A ces moments il existe dans p_3 deux types de jetons (des jetons disponibles, des jetons non disponibles).

- Malgré que le superviseur permet le franchissement de t_4 à θ = 4, mais il n'existe aucun jeton disponible dans p_3 pour passer.

Table 6. 4, Le comportement de système

	Sans superviseur			Sous superviseur		
θ	p_3	p_2	p_1	p_3	p_2	p_1
0	3	1	2	3	1	2
1	3	1	2	3	1	2
2	3+1=4	2	0	3+1=4	2	0
3	4-3=1	2	0	4-3=1	2	0
4	1+2=3	0	0	1+2=3	0	0
5	3-1=2	0	0	1+2=3	0	0
6	2	0	0	1+2=3	0	0
7	0	0	0	0	0	0

Cas d'étude 2 : Considérons l'exemple de la figure 6.2, où p_3 est soumise à la même CEM $M(p_3)\neq 2$.

$Y^+(p_3,M_0,\theta)= M(p_3)+M(p_1)\ _{if\ \theta\geq 4}+M(p_2)\ _{if\ \theta\geq 2}+\sigma(t_1,\ \theta\text{-}4)$

$Y^-(p_3,M_0,\theta)= M(p_3)\ _{if\ \theta\geq 1}+ M(p_1)\ _{if\ \theta\geq 5}+M(p_2)\ _{if\ \theta\geq 3}++\sigma(t_1,\ \theta\text{-}5)$

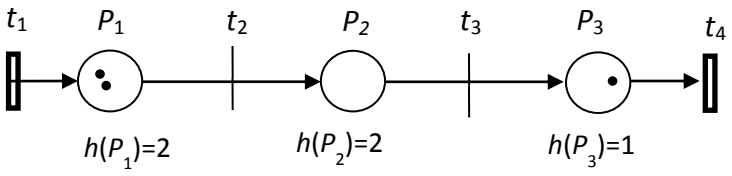


Figure 6. 2, *Un graphe d'événements P-temporisé soumis à une CEM* : *M(p_3)≠2*

Nous répétons les mêmes étapes et mêmes tables pendant la période [0, $h_{(\pi)}$+$h_{(q)}$=5] pour montrer que la loi de commande (*VI*.1) est toute seule insuffisante pour déterminer le comportement admissible.

Table 6. 5, $Y^+(p_3,M_0,\theta)$

θ	$M(p_3)$	$M(p_1)$ *if* $\theta\geq4$	$M(p_2)$ *if* $\theta\geq2$	$Y^+(p_3,M_0,\theta)$
0	1	0	0	1
1	1	0	0	1
2	1	0	0	1
3	1	0	0	1
4	1	2	0	3
5	1	2	0	3

Table 6. 6, $Y^-(p_3,M_0,\theta)$

θ	$M(p_3)$ *if* $\theta\geq1$	$M(p_1)$ *if* $\theta\geq5$	$M(p_2)$ *if* $\theta\geq3$	$Y^-(p_3,M_0,\theta)$
0	0	0	0	0
1	1	0	0	1
2	1	0	0	1
3	1	0	0	1
4	1	0	0	1
5	1	2	0	3

Table 6. 7, Table de Loi de commande TLC pour p_3

θ	$Y^+(p_3,M_0,\theta)$	$Y^-(p_3,M_0,\theta)$	$Y(q,M,\theta)$	$U_{(t4)}$
0	1	0	1	1
1	1	1	0	1
2	1	1	0	1
3	1	1	0	1
4	3	1	2	0
5	3	3	0	1

Malgré que la valeur logique de la loi de commande est vrai à θ=1, le superviseur doit interdire le franchissement de t_4 à ce moment, sinon à θ=4 la somme des jetons dans p_3 *sera* égale à 2. Autrement, le superviseur doit intervenir à θ=1, pour éviter d'atteindre un état interdit à θ=4. C'est bien montré dans la Table 6. 8.

Table 6. 8, Le comportement de système

	Sans Supervisor			Sous Supervisor		
θ	p_3	p_2	p_1	p_3	p_2	p_1
0	1	0	2	1	0	2
1	0	0	2	1	0	2
2	0	2	0	1	2	0
3	0	2	0	1	2	0
4	2	0	0	1+2=3	0	0
5	0	0	0	0	0	0

Par conséquence, nous avons vu que le superviseur doit intervenir seulement selon la loi de Commande (*VI*.1), et que cette loi toute seule est insuffisante. C'est pourquoi nous a avons besoin de décrire le protocole qui détermine les conditions sous lesquelles le superviseur doit intervenir :

Règle 6. 2 : Le rôle du superviseur est d'observer l'évolution du système et d'interdire le franchissement des transitions quand il le faut. Le superviseur TLC construit pendant l'intervalle [0, $h(\pi)+h(q)$], ensuite le superviseur interdit le franchissement de la transition $q^\circ \in T_c$ si au moins une de deux conditions suivantes est validée :

1. Au moment θ_i, si la valeur logique de loi de Commande (*VI*.1) est fausse.
2. Au moment θ_j, si les trois conditions suivantes sont validées :
   a. Il existe au moins un jeton disponible dans q à θ_j
   b. La valeur logique de la loi de commande (*VI*.1) est vraie à θ_j.
   c. $Y(q,M,\theta_j)< k$, où $\theta_i>\theta_j$, $\theta_j \in [0, \theta_i [$

   θ_j: instant avant θ_i lorsque la valeur de loi de commande (*VI*.1) est fausse.

Reconsidérons les deux cas d'études suivants :

- Premier cas : Table 6. 3, le superviseur interdit le franchissement de t_4 quand θ=5, 6, par contre cette transition est franchissable à θ=4car $Y(p_3,M_0,4)$=3>2.
- Deuxième cas : Table 6. 7, le superviseur interdit le franchissement de t_4 quand θ=4 (car la valeur logique de (*VI*.1) est fausse), c'est pareille pour θ=1 car les trois conditions sont validées (il y a un jeton disponible dans p_3 à θ=1, la valeur de (*VI*.1) est vraie et $Y(p_3,M_0,1)$=1<2).

6.3.2 Analyse de la loi de commande en franchissant des transitions d'influence

Nous étudions ici dans deux cas de figures l'effet du franchissement des transitions d'influence, et déterminons en conséquence le moment d'intervention du superviseur.

Cas d'étude 3: Considérons l'exemple dans Figure 6. , où un graphe d'événements P-temporisé est soumis à une CEM : $M(p_3)\neq 2$, et on veut franchir t_1 au moment e=1.

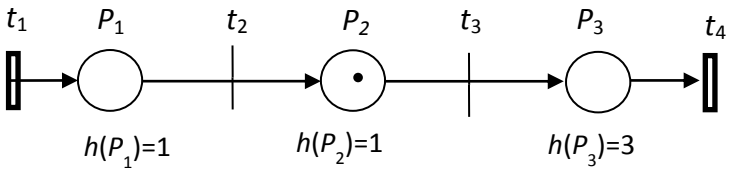


Figure 6. 3, *Un graphe d'événements P-temporisé soumis à une CEM* : $M(p_3)\neq 2$

Les deux équations (6.1) et (6.2) prennent les formes :

$Y^+(p_3,M_0,\theta)= M(p_3)+M(p_1)_{\ if\ \theta\geq 2}+M(p_2)_{\ if\ \theta\geq 1}+\sigma(t_1,\theta\text{-}3)$
$Y^-(p_3,M_0,\theta)= M(p_3)_{\ if\ \theta\geq 3}+ M(p_1)_{\ if\ \theta\geq 5}+M(p_2)_{\ if\ \theta\geq 4}+\sigma(t_1,\theta\text{-}6)$

Nous construisons les tables pendant la période $[e,e+ h_{(\pi 1)}+h_{(q)}]=[1, 6]$

Remarque 6. 1 : nous pouvons replacer la notion $\sigma(t_1,\theta\text{-}3)$ par $\sigma(t_1)_{\ if\ \theta\geq 3}$, pareillement pour $\sigma(t_1,\theta\text{-}6)$ par $\sigma(t_1)_{\ if\ \theta\geq 6}$.

Table 6. 9,$Y^+(p_3,M_0,\theta)$

θ	$M(p_3)$	$M(p_1)_{\ if\ \theta\geq 2}$	$M(p_2)_{\ if\ \theta\geq 1}$	$\sigma(t_1)_{\ if\ \theta\geq 3}$	$Y^+(p_3,M_0,\theta)$
1	0	0	1	0	1
2	0	0	1	0	1
3	0	0	1	1	2
4	0	0	1	1	2
5	0	0	1	1	2
6	0	0	1	1	2

Table 6. 10, $Y^-(p_3,M_0,\theta)$

θ	$M(p_3)_{\ if\ \theta\geq 3}$	$M(p_1)_{\ if\ \theta\geq 5}$	$M(p_2)_{\ if\ \theta\geq 4}$	$\sigma(t_1)_{\ if\ \theta\geq 6}$	$Y^-(p_3,M_0,\theta)$
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	1
5	0	0	1	0	1
6	3	0	1	1	2

Table 6. 11, Table de Loi de Commande TLC pour p_3

θ	$Y^+(p_3,M_0,\theta)$	$Y^-(p_3,M_0,\theta)$	$Y(p_3,M_0,\theta)$	$U_{(t4)}$
1	1	0	1	1
2	1	0	1	1
3	2	0	2	0
4	2	1	1	1
5	2	1	1	1
6	2	2	0	1

La loi de commande (VI.1) est fausse à θ=1, et dans 2 unités de temps (θ=3) le marquage interdit sera atteint, c'est pourquoi le superviseur doit intervenir avant cet événement. Le seul moyen d'éviter l'accès à ce marquage interdit est d'interdire t_1 à θ=1.

Cas d'étude 4: Considérons l'exemple dans Figure 6., où un graphe d'événements temporisé soumis à une CEM : $M(p_3)\neq 2$, et on veut franchir t_1 au moment e=1.

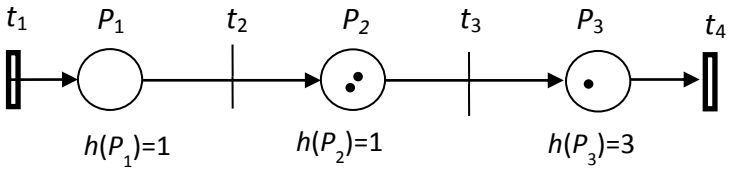


Figure 6.4, *Un graphe d'événements P-temporisé soumis à une CEM* : $M(p_3)\neq 2$

Les deux équations (6.1) et (6.2) prennent les formes :

$Y^+(p_3,M_0,\theta)= M(p_3)+M(p_1)_{\ if\ \theta\geq 2}+M(p_2)_{\ if\ \theta\geq 1}+\sigma(t_1)_{\ \theta\geq e+\ 2}$

$Y^-(p_3,M_0,\theta)= M(p_3)_{\ if\ \theta\geq 3}+ M(p_1)_{\ if\ \theta\geq 5}+M(p_2)_{\ if\ \theta\geq 4}+\sigma(t_1)_{\ \theta\geq e+\ 5}$

Table 6. 12, $Y^+(p_3,M_0,\theta)$

θ	$M(p_3)$	$M(p_1)_{\ if\ \theta\geq 2}$	$M(p_2)_{\ if\ \theta\geq 1}$	$\sigma(t_1)_{\ if\ \theta\geq 3}$	$Y^+(p_3,M_0,\theta)$
1	1	0	2	0	3
2	1	1	2	0	4
3	1	1	2	1	5
4	1	1	2	1	5
5	1	1	2	1	5
6	1	1	2	1	5

Table 6. 13, $Y^-(p_3,M_0,\theta)$

θ	$M(p_3)_{\ if\ \theta\geq 3}$	$M(p_1)_{\ if\ \theta\geq 5}$	$M(p_2)_{\ if\ \theta\geq 3}$	$\sigma(t_1)_{\ if\ \theta\geq 6}$	$Y^-(p_3,M_0,\theta)$
1	0	0	0	0	0
2	0	0	0	0	0
3	1	0	2	0	3
4	1	0	2	0	3
5	1	1	2	0	4
6	1	1	2	1	5

Table 6. 14, Table de Loi de Commande TLC pour p_3

θ	$Y^+(p_3,M_0,\theta)$	$Y^-(p_3,M_0,\theta)$	$Y(p_3,M_0,\theta)$	$U_{(t4)}$
1	3	0	3	1
2	4	0	4	1
3	5	3	2	0
4	5	3	2	0
5	5	4	1	1
6	5	5	0	1

Clairement, en analysant la Table 6. 14, nous remarquons que la loi de commande à θ=3, 4 est fausse, c'est-à-dire qu'il faut interdire le franchissement de t_1 à θ=1. Un deuxième choix est possible : ça consiste à interdire le franchissement de q° à θ=3, 4, car $Y(p_3,M_0,2)>k$, i.e., il est possible de garder les jetons disponibles dans p_3 pour avoir un marquage de p_3 supérieur à k à θ=3, 4. Contrairement au cas de figure 5.3, le superviseur n'avait qu'un seul choix parce que $Y(p_3,M_0,2)<k$.

Alors, dans ce cas de Figure 6.3, le superviseur a les deux choix suivants :

1. Interdire le franchissement de t_1 à θ=1
2. Permettre le franchissement de t_1 à θ=1, à condition que le franchissement de t_4 sera interdit à θ=3 et θ=4.

La Table 6. 15, nous montre les différents comportements du système avec et sans le superviseur.

Table 6. 15, Le comportement de système

	Sans superviseur			Sous superviseur (deuxième choix)		
θ	p_3	p_2	p_1	p_3	p_2	p_1
1	1+2=3	1	1	1+2=3	1	1
2	1+2+1=4	1	0	1+2+1=4	1	0
3	1+1=2	0	0	2+1+1=4	0	0
4	1+1=2	0	0	2+1+1=4	0	0
5	1	0	0	1	0	0
6	0	0	0	0	0	0

Nous résumons la loi de commande d'une transition d'influence par deux règles :

Règle 6. 3 : Permettre le franchissement de la transition d'influence à $\theta=e$, si toutes les valeurs de la loi de commande (*VI.*1) dans TLC est vrai pendant [e, $e+ h_{(\pi 1)}+h_{(q)}$].

Règle 6. 4 : Lors du franchissement d'une transition d'influence à $\theta=e$, et si les deux conditions suivantes sont validées :

1. Il existe au moins une valeur fausse dans TLC pendant [e, $e+ h_{(\pi 1)}+h_{(q)}$] à θ_i
2. $Y(q,M, \theta_i -1)> Y(q,M, \theta_i)$

Le superviseur a deux choix :

A. Soit, le superviseur empêche le franchissement de la transition d'influence à $\theta=e$

B. Ou, le superviseur permet le franchissement de la transition d'influence à θ=*e*, à condition d'interdire le franchissement de *q*° à θ_i

Exemple illustratif :

Considérons le graphe d'événements dans Figure 6. , avec M_0=[1 2 1 2 1]^t et la CEM: $M(p_5)\neq 2$.

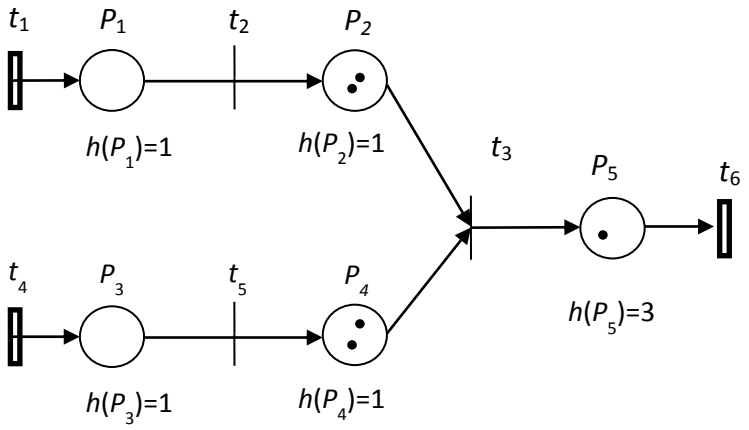


Figure 6. 5, ***Un graphe d'événements temporisé soumis à une CEM* : $M(p_5)\neq 2$**

Dans cet exemple, la place critique a deux chemins d'influence (π_1: t_1,p_1,t_2,p_2) and (π_2: t_4,p_3,t_5,p_4). Les places p_2 et p_3 sont les places d'entrée de $^\circ q \in T_{uc}$:

$$(p_q)_i=\{p_q \in P / (p_q)_i = {}^\circ({}^\circ q)\}$$

Nous analysons la chronologie de t_3:

$\sigma(t_3,M_0,\theta)= min\ (Y(p_2,M_0,\theta), Y(p_4,M_0,\theta))$

Les jetons arrivant dans p_5 par le franchissant de t_3, vont quitter cette place dans un délai $h(p_5)$:

$\sigma(t_6,M_0,\theta)= \min\ (Y(p_2,M_0,\theta- h(p_5)), Y(p_4,M_0,\theta- h(p_5)))$

$Y^+(p_5,M_0,\theta)=M(p_5)+ \sigma(t_3,M_0,\theta)$

$Y^-(p_5,M_0,\theta)= M(p_5)+ \sigma(t_6,M_0,\theta)$

$Y(p_5,M_0,\theta)= Y^+(p_5,M_0,\theta)-Y^-(p_5,M_0,\theta)$

Table 6. 16, $Y(p_2,M_0,\theta)$

θ	$M(p_2)$ *if* $\theta \geq 1$	$M(p_1)$ *if* $\theta \geq 3$	$Y(p_2,M_0,\theta)$	$Y(p_2,M_0,\theta - h(p_5))$
0	0	0	0	0
1	2	0	2	0
2	2	0	2	0
3	2	1	3	2
4	2	1	3	2
5	2	1	3	3

Table 6. 17, $Y(p_4,M_0,\theta)$

θ	$M(p_4)$ *if* $\theta \geq 3$	$M(p_3)$ *if* $\theta \geq 4$	$Y(p_4,M_0,\theta)$	$Y(p_4,M_0,\theta - h(p_5))$
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	2	0	2	0
4	2	1	3	0
5	2	1	3	2
6	2	1	3	3

Table 6. 18, $\sigma(t_3,M_0,\theta)$ et $\sigma(t_6,M_0,\theta)$

θ	$Y(p_2,M_0,\theta)$	$Y(p_4,M_0,\theta)$	$\sigma(t_3,M_0,\theta)$	$\sigma(t_6,M_0,\theta)$
0	0	0	0	0
1	2	0	0	0
2	2	0	0	0
3	3	2	2	0
4	3	3	3	0
5	3	3	3	2
6	3	3	3	3

Table 6. 19, $Y^+(p_5,M_0,\theta)$

θ	$M(p_5)$	$\sigma(t_3,M_0,\theta)$	$Y(p_5,M_0,\theta)$
0	1	0	1
1	1	0	1
2	1	0	1
3	1	2	3
4	1	3	4

Table 5. 20, $Y(p_5,M_0,\theta)$

θ	$M(p_5)$ *if* $\theta \geq 2$	$\sigma(t_6,M_0,\theta)$	$Y(p_5,M_0,\theta)$
0	0	0	0
1	0	0	0
2	1	0	1
3	1	0	1
4	1	0	1
5	1	2	3
6	1	3	4
7	1	3	4

Table 6. 21, TLC de p_5

θ	$Y(p_5,M_0,\theta)$	$Y(p_5,M_0,\theta)$	$Y(p_5,M_0,\theta)$	$U_{(t5)}$
0	1	0	1	1
1	1	0	1	1
2	1	1	0	1
3	3	1	2	0
4	4	1	3	1
5	4	3	3	1
6	4	4	1	1
7	4	4	0	1

D'après la Règle 6. 2, le franchissement de t_6 est interdit à θ=3 parce que la valeur de la loi de commande est fausse.

D'après la Règle 6. 2, le franchissement de t_6 est interdit à θ=2 parce que :

A. Il y a un jeton disponible dans p_5 à θ=2
B. La valeur logique est vrai à θ=2
C. $Y(p_5,M_0,2)< 2$

6.4 SdC pour des graphes d'événements P-temporisés soumis à des CEM-Ou/CEM-Et

Nous proposons dans cette partie un algorithme pour construire un superviseur pour des graphes d'événements P-temporisés soumis à des CEM-Où ou CEM-Et, à condition que les transitions de sortie des places critiques soient contrôlables.

Algorithme :

A. Au marquage initial M_0 :

1. Pour chaque $q_i \in Q_{mc}$, construire les tables de temps pendant *max* $[0, h_{(\pi)} + h_{(qi)}]$

Table 6. 22, $Y^+(q_i, M_0, \theta)$

θ	$M(q_i)$	....	$Y^+(q_i, M_0, \theta)$
0			
...			
max ($h_{(\pi)} + h_{(qi)}$)			

Table 6. 23, $Y^-(q_i, M_0, \theta)$

θ	$M(q_i)_{if \theta \geq h(q)}$	....	$Y^-(q_i, M_0, \theta)$
0			
...			
max ($h_{(\pi)} + h_{(qi)}$)			

2. Pour chaque $q_i \in Q_{mc}$, construire TLC $[0, h_{(\pi)} + h_{(qi)}]$ en appliquant la Règle 6. 2.

Table 6. 24, TLC

θ	$Y^+(q_i, M_0, \theta)$	$Y^-(q_i, M_0, \theta)$	$Y(q_i, M_0, \theta)$	U_i
0				
...				
max ($h_{(\pi)} + h_{(qi)}$)				

Construire TLC-CEM-Ou si la spécification est CEM-Ou, et TLC-CEM-Et si la spécification est CEM-Et.

Table 6. 25, CLT-CEM-Ou

θ	U_1	....	U_{mc}	$U = U_1 \vee \ldots \vee U_{mc}$
0				
...				
max($h_{(\pi)} + h_{(qi)}$)				

Table 6. 26, CLT-CEM-Et

θ	U_1	....	U_{mc}	$U = U_1 \wedge \ldots \wedge U_{mc}$
0				
...				
max ($h_{(\pi)} + h_{(qi)}$)				

i. Tous les franchissements de transitions contrôlables sont admissibles si toutes les valeurs de loi de commande dans CLT pendant *max* ($h_{(\pi)} + h_{(qi)}$) sont vraies (Règle 6. 2).

ii. Si la loi de commande prend une valeur zéro à ϑ:

- Pour CEM-Ou : il est suffisant d'appliquer la Règle 6. 2 pour une place critique ϑ.

- Pour CEM-Et : il faut appliquer la Règle 6. 2 pour toutes les places critiques ϑ.

B. Au franchissement d'une transition d'influence $t \in \Gamma(q_i)$ au moment e :

1. Pour chaque $q_i \in Q_{mc}$, construire les tables de temps pendant *max* $[e, e + h_{(\pi 1)} + h_{(q)}]$

2. Pour chaque $q_i \in Q_{mc}$, construire TLC $[e, e + h_{(\pi 1)} + h_{(q)}]$ en appliquant Règle 6. 2

3. Tous les franchissements de transitions contrôlables sont admissibles si toutes les valeurs de la loi de commande dans CLT pendant *max* ($[e, e + h_{(\pi 1)} + h_{(q)}]$) sont vrai (Règle 6. 3).

4. Si loi de commande prend une valeur zéro à ϑ :

i. Pour CEM-Ou : il est suffisant d'appliquer la Règle 6. 2 pour une place critique

ii. Pour CEM-Et : il faut appliquer la Règle 6. 4 pour toutes les places critiques.

6.5 Exemple illustratif

Considérons la ligne de production de la Figure 6. Cette ligne se compose de deux cellules. La première est constituée d'un réservoir de liquide avec une capacité illimitée et une pompe qui remplit les bouteilles par un litre de liquide. La seconde est un convoyeur pour déplacer les bouteilles d'une capacité d'un litre.

Le modèle de graphe d'événements P-temporisé pour cet exemple est donné dans la Figure 6. . La Table 6. 27, nous donne la description des places, des transitions et les délais. Le marquage initial est : $M_0 = [3\,1\,0\,3\,1\,0\,0]$.

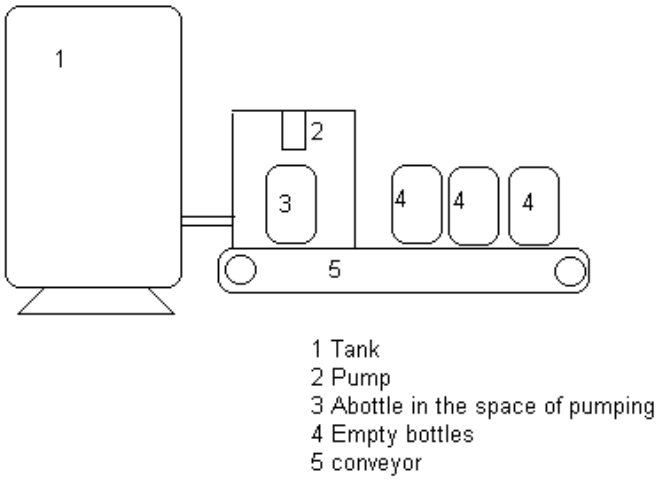


Figure 6. 6, ***Machine de remplissage de bouteilles avec du liquide***

Table 6. 27, Interprétation du graphe d'événement P-temporisé

Place	Interpretation	Delai
p_1	Bouteilles vides sont sur le convoyeur	Non temporisé
p_2	Bouteille vide est exactement dans l'espace de remplissage	2 unités de temps
p_3	Capacité de l'espace de remplissage	Non temporisé
p_4	Réservoir	Non temporisé
p_5	Un litre de liquide est prêt à être rempli	1 unité de temps
p_6	Capacité de la pompe	Non temporisé
p_7	Remplissage du liquide	2 unités de temps
Transition		
t_1	Arriver d'une nouvelle bouteille au convoyeur	Non temporisé
t_2	Arriver d'une bouteille à l'espace de remplissage	Non temporisé
t_3	sortie d'une bouteille de l'espace de remplissage	Non temporisé
t_4	Remplir le réservoir avec du liquide	Non temporisé
t_5	Vider un litre du réservoir	Non temporisé
t_6	La pompe est prête à fonctionner	Non temporisé
t_7	Terminer le travail de la pompe	Non temporisé

Pendant le fonctionnement de ce système, il est nécessaire de spécifier que la pompe ne fonctionne pas s'il n'existe pas de bouteille dans l'espace de remplissage. En revanche il est possible d'avoir une bouteille dans cet espace même s'il n'y a plus de liquide. Nous traduisons cette spécification de fonctionnement en langage de graphes d'événements : tous les marquages ayant un jeton dans p_7 et zéro dans p_2 sont des marquages interdits, dans la figure 6.7.

$$M_f = \{M \in R(N, M_0) / M(p_2) \neq 1 \wedge M(p_7) \neq 0\}$$

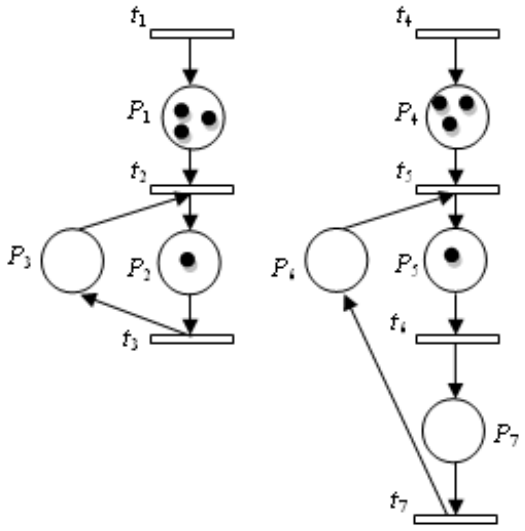


Figure 6. 7, ***Un graphe d'événements P-temporisé pour l'exemple de figure 5.6***

Table 6. 28, $Y^+(p_2, M_0, \theta)$

θ	$M(p_2)$	$Y^+(p_2, M_0, \theta)$
0	1	1
1	1	1
2	1	1

Table 6. 29, $Y^-(p_2, M_0, \theta)$

θ	$M(p_2)_{if\theta \geq 2}$	$Y^-(p_2, M_0, \theta)$
0	0	0
1	0	0
2	1	1

Table 6. 30, TLC pour p_2

θ	$Y^+(p_2, M_0, \theta)$	$Y^-(p_2, M_0, \theta)$	$Y(p_2, M_0, \theta)$	U_1
0	1	0	1	1
1	1	0	1	1
2	1	1	0	0

Table 6. 31, $Y^+(p_7, M_0, \theta)$

θ	$M(p_5)_{if\theta \geq 1}$	$M(p_7)$	$Y^+(p_7, M_0, \theta)$
0	0	0	0
1	1	0	1
2	1	0	1
3	1	0	1

Table 6. 32, $Y(p_7,M_0,\theta)$

θ	$M(p_5)_{if\theta\geq 3}$	$M(p_7)_{if\theta\geq 2}$	$Y^-(p_7,M_0,\theta)$
0	0	0	0
1	0	0	0
2	0	0	0
3	1	0	1

Table 6. 33, TLC pour p_7

θ	$Y^+(p_7,M_0,\theta)$	$Y^-(p_7,M_0,\theta)$	$Y(p_7,M_0,\theta)$	U_2
0	0	0	0	1
1	1	0	1	0
2	1	0	1	0
3	1	1	0	1

Table 6. 34, TLC-MEC-Ou

θ	U_1	U_2	$U=U_1\vee U_2$
0	1	1	1
1	1	0	1
2	0	0	0
3	0	1	1

Dans Table 6. 34, nous voyons que la spécification n'est pas respectée à $\theta=2$, Il y aura à ce moment un litre de liquide qui sera pompé dans le vide. En conséquence, le superviseur doit mettre en œuvre de nombreuses procédures à ce moment d'après la Règle 6. 2.

Le superviseur va empêcher de franchir de t_3 à $\theta=2$, et va le permettre au moment suivant à $\theta=3$.

6.6 Conclusion

Dans ce chapitre, nous avons proposé une SdC par superviseur pour les systèmes à événements discrets modélisés par des graphes d'événements temporisés soumis aux spécifications décrites par des CEM. Nous avons traité le problème en tenant compte des événements incontrôlables et/ou inobservables. Grâce à la donnée temps, ces deux types d'information deviennent similaires et le marquage réel devient connu.

Notre politique suppose que le marquage initial est connu pour construire les tables de temps qui montrent la chronologie de distribution des jetons dans les places. D'après ces tables, nous construisons un autre type de tables, appelé tables de loi de commande TLC. En analysant ces tables, le superviseur intervient sur le comportement du système quand il le faut selon quatre règles pour contrôler les transitions

d’influence ou de sortie. Une extension de ce travail a été proposée pour résoudre le problème de CEM-Ou ou CEM-Et.

Nous avons traité une classe spécifique d’événements temporisés. Des travaux futurs dans ce domaine peuvent être développés, tel que : le cas où on néglige la vitesse maximale, les réseaux de Petri T-temporisé, les réseaux de Petri temporel, etc.

Chapitre 7

Atelier flexible d'usinage

Résumé :

Dans ce chapitre nous illustrions notre travail par un exemple d'un atelier flexible d'usinage, constitué de machines-outils à commande numérique. Nous modélisons l'atelier par un graphe d'événements, ensuite nous décrivons les spécifications par des CEM et en fin nous construisons le superviseur en utilisant l'algorithme présenté dans les chapitres précédents.

7.1 Introduction

Pour illustrer la capacité des CEM et pour tester notre SdC, nous présentons un exemple d'un atelier flexible de type *Jobshop* où les pièces à fabriquer ne circulent pas dans le même sens de parcours des machines et en plus les pièces ne passent pas forcement sur toutes les machines. Dans cet exemple il existe un stock pour les pièces, noté *Stock*, trois machines M_1, M_2 et M_3, trois robots R_1, R_2 et R_3, six convoyeurs S_1, S_2, S_3, S_4, S_5 et S_6, et trois pré-stock *Pré-*S_1, *Pré-*S_2 et *Pré-*S_3:

M_1 : Tour à commande numérique

M_2 : Fraiseuse à commande numérique

M_3: Poste automatique d'assemblage

R_1, R_2: pour recharger et décharger les machines M_1 et M_2

R_3: pour décharger les machines M_3

Pré-S_1, *Pré-*S_2 et *Pré-*S_3: pour déposer les palettes à l'entrer les machines M_1, M_2 et M_3.

S_1 : Convoyeur pour transporter les palettes du *stock* vers M_1

S_2 : Convoyeur pour transporter les palettes du *stock* vers M_2

S_3 : Convoyeur pour transporter les palettes de M_1 vers M_3

S_4 : Convoyeur pour transporter les palettes de M_2 vers M_3

S_5 : Convoyeur pour transporter les palettes dans les deux sens entre M_1 et M_2

S_6 : Convoyeur pour transporter les palettes vides de M_3 vers le stock.

La disposition des éléments de cet atelier est présentée dans Figure 7. 1.

L'atelier fabrique deux types de pièces *Pièce1 et Pièce2.* Les pièces sont réalisées par usinage d'un brut obtenu par moulage. Chaque pièce doit subir un certain nombre d'opérations sur les machines dans un ordre précis.

Pièce1 : $M_1 \rightarrow M_2$

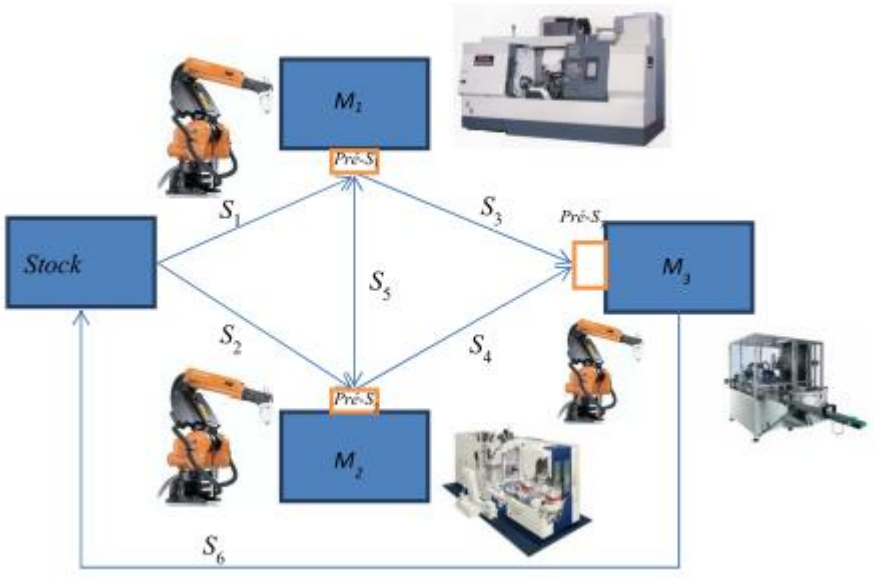
La *Pièce1* est composée de trois entités D_1, D_2 et D_3:

Pour réaliser l'entité D_1 : Alésage sur M_1.
Pour réaliser l'entité D_2 : Centrage et perçage sur M_2.
Pour réaliser l'entité D_3 : Fraisage finition sur M_2.

Pièce2 : $M_2 \rightarrow M_1$

La *Pièce2* est composée de trois entités D_4, D_5 , D_6 et D_7:

Pour réaliser l'entité D_4 : Centrage et perçage sur M_2.
Pour réaliser l'entité D_5 : Alésage sur M_2.
Pour réaliser l'entité D_6 : Chanfrein sur M_1.
Pour réaliser l'entité D_7 : Tournage salgnée sur M_1.



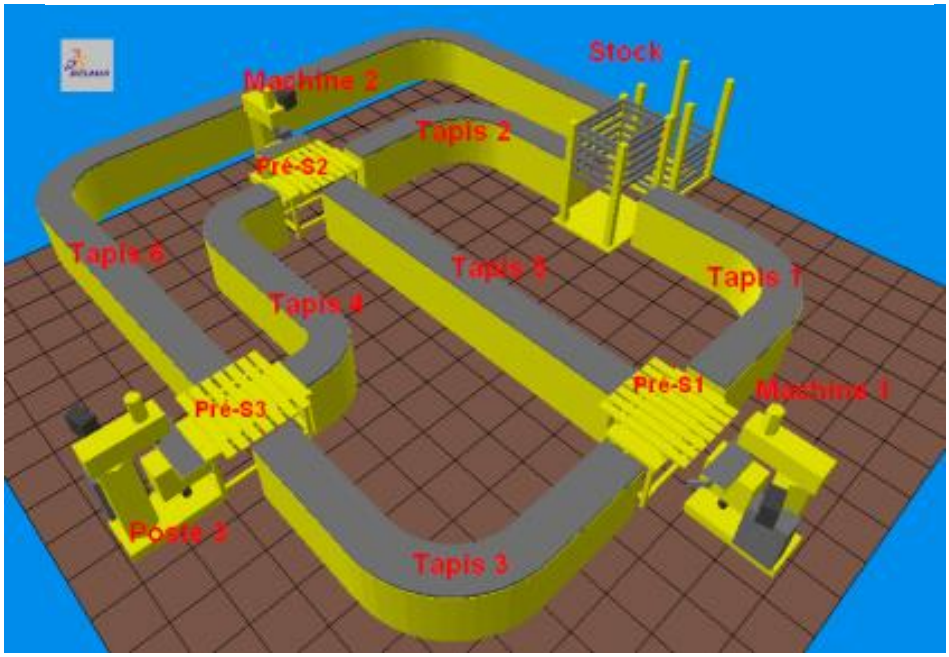
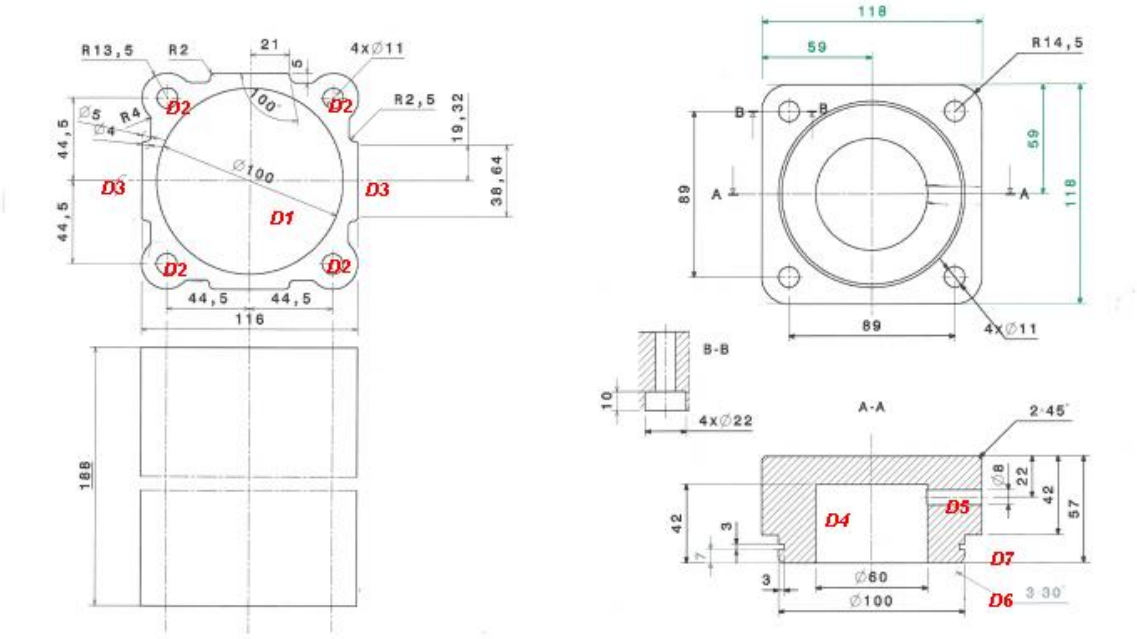


Figure 7. 1, ***Exemple d'un atelier flexible***



Pièce1 *Pièce2*

Figure 7. 2, *les pièces 1 et 2*

L'ordonnancement des pièces sur les machines est défini comme suit :

M_1 : *Pièce1*→2 *Pièce2*

M_2 : 2 *Pièce2* → *Pièce1*

Nous devons respecter des ratios de production 1 : 2 ce qui veut dire que nous produisons simultanément une *Pièce1* et deux *Pièce2.*

Nous supposons que les pièces sont portées par des palettes dédiées à type de pièce et qu'une palette porte seulement une pièce. Lorsqu'une pièce a subi les opérations requises, elle est déchargée de la palette et entre dans le stock de M_3 pour l'assemblage. Les palettes vides entrent au stock pour recharger de nouvelles pièces. Dans le stock, il existe deux palettes pour les pièces de type *Pièce1* et quatre palettes pour les pièces de type *Pièce2.*

La figure 7.3 présente les différents niveaux de commande de cet atelier :

Le premier niveau :

1. Chaque machine est pilotée par un contrôleur de machine à commande numérique.
2. Le poste trois et les tapis sont pilotés par un automate programmable
3. Chaque robot est piloté par un contrôleur propre.

Les contrôleurs des machines à commande numérique sont programmés par des langages spécifiques tels que le G-codes ou le M-codes, le ladder et le Grafcet pour les automates programmables. Dans ce niveau, la commande génère les événements pour produire les opérations voulus.

Le deuxième niveau :

Un superviseur synchronise les événements produits sur tous les composants de cet atelier, pour respecter les spécifications définies.

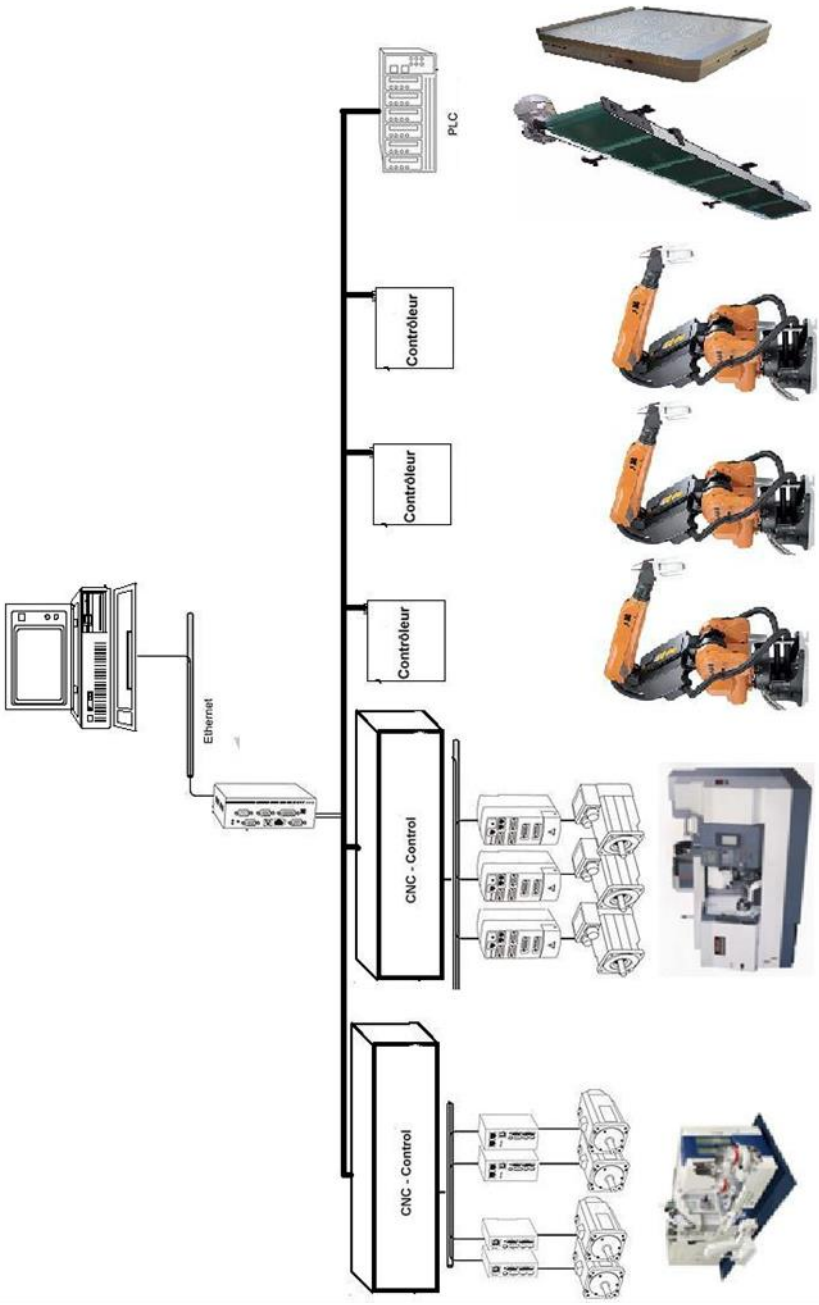


Figure 7. 3, ***Les niveaux de commande de l'atelier***

7.2 Modèle logique

Les tableaux suivants représentent les significations des transitions et des places pour le modèle logique dans la figure 7.4:

Pièce1	
p_{11}	S_1
p_{12}	*Pré*-S_1
p_{13}	Robot R_1 (*Pré*-$S_1 \rightarrow M_1$)
p_{14}	Machine M_1
p_{15}	Robot R_1 ($M_1 \rightarrow$ *Pré*-S_1)
p_{16}	*Pré*-S_1
p_{17}	S_5
p_{18}	*Pré*-S_2
p_{19}	Robot R_2 (*Pré*-$S_2 \rightarrow M_2$)
p_{110}	Machine M_2
p_{111}	Robot R_2 ($M_2 \rightarrow$ *Pré*-S_2)
p_{112}	*Pré*-S_2
p_{113}	S_4
p_{114}	*Pré*-S_3
p_{115}	Robot R_3 (*Pré*-$S_3 \rightarrow M_2$)
p_{116}	Pièce dans sa position dans le poste M_3
p_{117}	S_6
p_{118}	Stock
p_1	Assemblage
t_{11}	Pièce sort du stock et entre dans S_1
t_{12}	Pièce sort de S_1 et entre dans *Pré*-S_1
t_{13}	Robot R_1 porte la pièce de *Pré*-S_1
t_{14}	Robot R_1 met la pièce dans M_1 et le début de l'usinage sur M_1
t_{15}	La fin de l'usinage sur M_1 et Robot R_1 porte la pièce de M_1
t_{16}	Robot R_1 met la pièce dans *Pré*-S_1
t_{17}	Pièce sort de *Pré*-S_1et entre dans S_5
t_{18}	Pièce sort de S_5 et entre dans *Pré*-S_2
t_{19}	Robot R_2 porte la pièce de *Pré*-S_2
t_{110}	Robot R_2 met la pièce dans M_2 et le début de l'usinage sur M_2
t_{111}	La fin de l'usinage sur M_2 et Robot R_2 porte la pièce de M_2
t_{112}	Robot R_2 met la pièce dans *Pré*-S_2
t_{113}	Pièce sort de *Pré*-S_2 et entre dans S_4
t_{114}	Pièce sort de S_4 et entre dans *Pré*-S_3
t_{115}	Robot R_3 porte la pièce de *Pré*-S_3 et la met dans M_3, la palette vide entre dans S_6
t_{116}	Robot R_3 met la pièce dans M_3
t_{117}	La palette vide entre dans le stock
t_1	le début de l'assemblage sur M_3
t_2	La fin de l'assemblage sur M_3

Pièce2	*Pièce2*	
p_{21}	p_{31}	S_2
p_{22}	p_{32}	*Pré*-S_2
p_{23}	p_{33}	Robot R_2 (*Pré*-$S_2 \rightarrow M_2$)
p_{24}	p_{34}	Machine M_2
p_{25}	p_{35}	Robot R_2 ($M_2 \rightarrow$ *Pré*-S_2)
p_{26}	p_{36}	*Pré*-S_2
p_{27}	p_{37}	S_5
p_{28}	p_{38}	*Pré*-S_1
p_{29}	p_{39}	Robot R_1 (*Pré*-$S_1 \rightarrow M_1$)
P_{210}	p_{310}	Machine M_1
p_{211}	p_{311}	Robot R_1 ($M_1 \rightarrow$ *Pré*-S_1)
p_{212}	p_{312}	*Pré*-S_1
p_{213}	p_{313}	S_3
p_{214}	p_{314}	*Pré*-S_3
p_{215}	p_{315}	Robot R_3 (*Pré*-$S_3 \rightarrow M_2$)
p_{216}	p_{316}	Pièce le poste M_3
p_{217}	p_{317}	S_6
p_{218}	p_{318}	Stock
p_1	p_1	Assemblage
t_{21}	t_{31}	Pièce sort du stock et entre dans S_2
t_{22}	t_{32}	Pièce sort de S_2 et entre dans *Pré*-S_2
t_{23}	t_{33}	Robot R_2 porte la pièce de *Pré*-S_2
t_{24}	t_{34}	Robot R_2 met la pièce dans M_2 et le début de l'usinage sur M_2
t_{25}	t_{35}	La fin de l'usinage sur M_2 et Robot R_2 porte la pièce de M_2
t_{26}	t_{36}	Robot R_2 met la pièce dans *Pré*-S_2
t_{27}	t_{37}	Pièce sort de *Pré*-S_2 et entre dans S_5
t_{28}	t_{38}	Pièce sort de S_5 et entre dans *Pré*-S_1
t_{29}	t_{39}	Robot R_1 porte la pièce de *Pré*-S_1
t_{210}	t_{310}	Robot R_1 met la pièce dans M_1 et le début de l'usinage sur M_1
t_{211}	t_{311}	La fin de l'usinage sur M_1 et Robot R_1 porte la pièce de M_1
t_{212}	t_{312}	Robot R_1 met la pièce dans *Pré*-S_1
t_{213}	t_{313}	Pièce sort de *Pré*-S_1 et entre dans S_3
t_{214}	t_{314}	Pièce sort de S_3 et entre dans *Pré*-S_3
t_{215}	t_{315}	Robot R_3 porte la pièce de *Pré*-S_3 et la met dans sa position dans M_3, la palette vide entre dans S_6
t_{216}	t_{316}	Robot R_3 met la pièce dans M_3
t_{217}	t_{317}	La palette vide entre dans le stock
t_1	t_1	le début de l'assemblage sur M_3
t_2	t_2	La fin de l'assemblage sur M_3

Les places p_2, p_3, p_4 et p_5 garantient qu'une pièce de type *Pièce2* passe alternativement dans les places p_{21} et p_{31}.

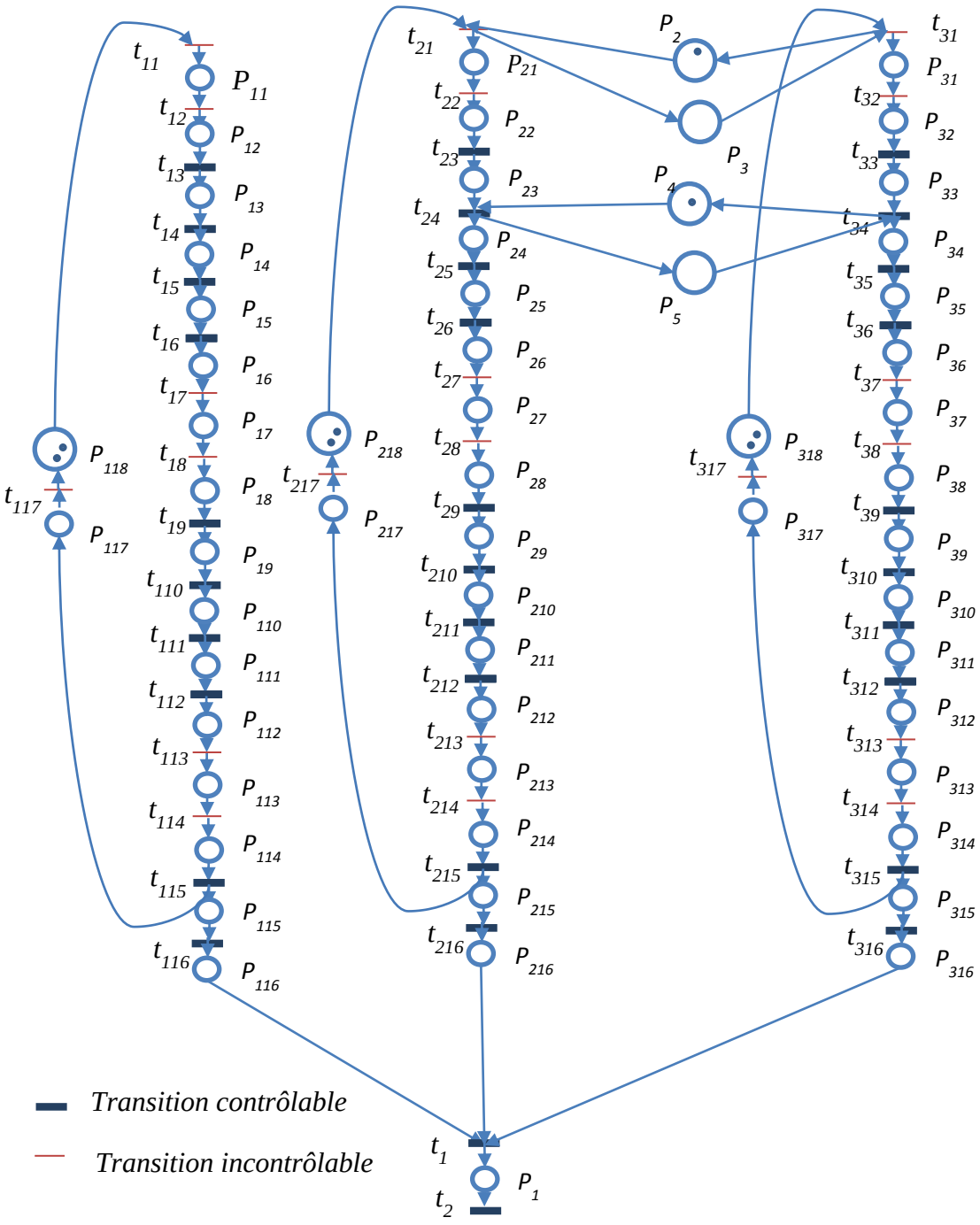


Figure 7. 4, ***Le modèle logique de l'atelier flexible***

7.3 Les spécifications

Robots :

Robot R_1 : ce robot ne transporte qu'une pièce à la fois.

$Q_{mc1}=\{p_{13}, p_{15}, p_{29}, p_{211}, p_{39}, p_{311}\}$

$SMMI(Q_{mc1}, 2)\equiv M(p_{13})+M(p_{15})+M(p_{29})+ M(p_{211})+ M(p_{39})+ M(p_{311})\leq 1$

Robot R_2 : ce robot ne transporte qu'une pièce à la fois.

$Q_{mc2}=\{p_{19}, p_{111}, p_{23}, p_{33}, p_{25}, p_{35}\}$

$SMMI(Q_{mc2}, 2)\equiv M(p_{19})+M(p_{111})+M(p_{23})+ M(p_{33})+ M(p_{25})+ M(p_{35})\leq 1$

Robot R_3 : ce robot peut transporter une pièce de type pièce 1 ou deux de type pièce 2.

$Q_{mc3}=\{ p_{115}, p_{215}, p_{315} \}$

$C_1\equiv\{M(p_{115})\neq 1\vee M(p_{215})\neq 1\}\wedge\{M(p_{115})\neq 1\vee M(p_{215})\neq 2\}\wedge\{M(p_{115})\neq 2\vee M(p_{215})\neq 1\}\wedge\{M(p_{115})\neq 2 \vee M(p_{215})\neq 2\}\wedge\{M(p_{115})\neq 1\vee M(p_{315})\neq 1\}\wedge\{M(p_{115})\neq 1\vee M(p_{315})\neq 2\}\wedge\{M(p_{115})\neq 2\vee M(p_{315})\neq 1\}\wedge\{ M(p_{115})\neq 2\vee M(p_{315})\neq 2\}$

$\equiv\{M(p_{115})\neq 1\wedge M(p_{115})\neq 2\}\vee(\{ M(p_{215})\neq 1\wedge M(p_{215})\neq 2\}\wedge\{ M(p_{315})\neq 1\wedge M(p_{315})\neq 2\})$

D'après les règles des simplifications:

$C_1\equiv\{M(p_{115})\leq 0\vee M(p_{115})\geq 3\}\vee(\{M(p_{215}) \leq 0\vee M(p_{215}) \geq 3\}\wedge\{M(p_{315}) \leq 0\vee M(p_{315}) \geq 3\})$

Puisque dans cet exemple, il n'existe que deux jetons dans chaque sous-réseau:

$C_1\equiv M(p_{115})\leq 0\vee(M(p_{215}) \leq 0\wedge M(p_{315}) \leq 0)$

Machines :

Machine M_1 : Cette machine n'usine qu'une pièce à la fois.

$Q_{mc4}=\{ p_{14}, p_{210}, p_{310} \}$

$SMMI(Q_{mc4}, 2)\equiv M(p_{14})+M(p_{210})+M(p_{310})+ M(p_{33})\leq 1$

Machine M_2 : Cette machine peut usiner une pièce de type pièce 1 ou deux pièces de type pièce 2.

$Q_{mc5}=\{ p_{110}, p_{24}, p_{34} \}$

$C_2 \equiv \{M(p_{110}) \neq 1 \vee M(p_{24}) \neq 1\} \wedge \{M(p_{110}) \neq 1 \vee M(p_{24}) \neq 2\} \wedge \{M(p_{110}) \neq 2 \vee M(p_{24}) \neq 1\} \wedge \{M(p_{110}) \neq 2 \vee M(p_{24}) \neq 2\} \wedge$
$\{M(p_{110}) \neq 1 \vee M(p_{34}) \neq 1\} \wedge \{M(p_{110}) \neq 1 \vee M(p_{34}) \neq 2\} \wedge \{M(p_{110}) \neq 2 \vee M(p_{34}) \neq 1\} \wedge \{M(p_{110}) \neq 2 \vee M(p_{34}) \neq 2\} \equiv$

$\equiv \{M(p_{110}) \neq 1 \wedge M(p_{110}) \neq 2\} \vee (\{ M(p_{24}) \neq 1 \wedge M(p_{24}) \neq 2\} \wedge \{ M(p_{34}) \neq 1 \wedge M(p_{34}) \neq 2\})$

D'après les règles des simplifications :

$C_2 \equiv \{M(p_{110}) \leq 0 \vee M(p_{110}) \geq 3\} \vee (\{M(p_{24}) \leq 0 \vee M(p_{24}) \geq 3\} \wedge \{M(p_{34}) \leq 0 \vee M(p_{34}) \geq 3\})$

Puisque dans cet exemple, il n'existe que deux jetons dans chaque sous-réseau:

$C_2 \equiv M(p_{110}) \leq 0 \vee (M(p_{24}) \leq 0 \wedge M(p_{34}) \leq 0)$

Tapis de transport :

Tapis S_5 :

$Q_{mc6} = \{ p_{17}, p_{27}, p_{37} \}$

$C_3 \equiv \{M(p_{17}) \neq 1 \vee M(p_{27}) \neq 1\} \wedge \{M(p_{17}) \neq 1 \vee M(p_{27}) \neq 2\} \wedge \{M(p_{17}) \neq 2 \vee M(p_{27}) \neq 1\} \wedge \{M(p_{17}) \neq 2 \vee M(p_{27}) \neq 2\} \wedge$
$\{M(p_{17}) \neq 1 \vee M(p_{37}) \neq 1\} \wedge \{M(p_{17}) \neq 1 \vee M(p_{37}) \neq 2\} \wedge \{M(p_{17}) \neq 2 \vee M(p_{37}) \neq 1\} \wedge \{M(p_{17}) \neq 2 \vee M(p_{37}) \neq 2\} \equiv$

$\equiv \{M(p_{17}) \neq 1 \wedge M(p_{17}) \neq 2\} \vee (\{ M(p_{27}) \neq 1 \wedge M(p_{27}) \neq 2\} \wedge \{ M(p_{37}) \neq 1 \wedge M(p_{37}) \neq 2\})$

D'après les règles des simplifications::

$C_3 \equiv \{M(p_{17}) \leq 0 \vee M(p_{17}) \geq 3\} \vee (\{M(p_{27}) \leq 0 \vee M(p_{27}) \geq 3\} \wedge \{M(p_{37}) \leq 0 \vee M(p_{37}) \geq 3\})$

Puisque dans cet exemple, il n'existe que deux jetons dans chaque sous-réseau:

$C_3 \equiv M(p_{17}) \leq 0 \vee (M(p_{27}) \leq 0 \wedge M(p_{37}) \leq 0)$

7.4 Le superviseur

Cet exemple est modélisé par un réseau partiellement contrôlable, pour cela nous utilisons l'algorithme du chapitre 4.

Hors ligne :

1. Définir l'ensemble de places critiques Q_{mc}

$Q_{mc}= Q_{mc1} \cup Q_{mc2} \cup Q_{mc3} \cup Q_{mc4} \cup Q_{mc5} \cup Q_{mc6}$

Q_{mc}={ p_{13}, p_{15}, p_{29}, p_{211}, p_{39}, p_{311}, p_{19}, p_{111}, p_{23}, p_{33}, p_{25}, p_{35}, p_{115}, p_{215}, p_{315}, p_{14}, p_{210}, p_{310}, p_{110}, p_{24}, p_{34}, p_{17}, p_{27}, p_{37}}

2. Pour chaque $q_i \in Q_{mc}$:

   a. Le marquage initial de la place critique q ($M_0(q_i)$), la transition d'entrée °q et la transition de sortie q°

q	$M(q)$	°q	q°
p_{13}	0	t_{13}	t_{14}
p_{15}	0	t_{15}	t_{16}
p_{29}	0	t_{29}	t_{210}
p_{211}	0	t_{211}	t_{212}
p_{39}	0	t_{39}	t_{310}
p_{311}	0	t_{311}	t_{312}
p_{19}	0	t_{19}	t_{110}
p_{111}	0	t_{111}	t_{112}
p_{23}	0	t_{23}	t_{24}
p_{33}	0	t_{33}	t_{34}
p_{25}	0	t_{25}	t_{26}
p_{35}	0	t_{35}	t_{36}
p_{115}	0	t_{115}	t_{116}
p_{215}	0	t_{215}	t_{216}
p_{315}	0	t_{315}	t_{316}
p_{14}	0	t_{13}	t_{14}
p_{210}	0	t_{210}	t_{211}
p_{310}	0	t_{310}	t_{311}
p_{33}	0	t_{33}	t_{34}
p_{110}	0	t_{110}	t_{111}
p_{24}	0	t_{24}	t_{25}
p_{34}	0	t_{34}	t_{35}
p_{17}	0	$t_{17} \in T_{uc}$	$t_{18} \in T_{uc}$
p_{27}	0	$t_{27} \in T_{uc}$	$t_{28} \in T_{uc}$
p_{37}	0	$t_{37} \in T_{uc}$	$t_{38} \in T_{uc}$

   b. Définir $\lambda(q_i)$,$\Gamma(q_i)$

q	$\Gamma(q_i)$	$\lambda(q_i)$
p_{13}	t_{13}	t_{14}
p_{15}	t_{15}	t_{16}
p_{29}	t_{29}	t_{210}
p_{211}	t_{211}	t_{212}
p_{39}	t_{39}	t_{310}

p_{311}	t_{311}	t_{312}
p_{19}	t_{19}	t_{110}
p_{111}	t_{111}	t_{112}
p_{23}	t_{23}	t_{24}
p_{33}	t_{33}	t_{34}
p_{25}	t_{25}	t_{26}
p_{35}	t_{35}	t_{36}
p_{115}	t_{115}	t_{116}
p_{215}	t_{215}	t_{216}
p_{315}	t_{315}	t_{316}
p_{14}	t_{13}	t_{14}
p_{210}	t_{210}	t_{211}
p_{310}	t_{310}	t_{311}
p_{33}	t_{33}	t_{34}
p_{110}	t_{110}	t_{111}
p_{24}	t_{24}	t_{25}
p_{34}	t_{34}	t_{35}
p_{17}	t_{16}	ϕ
p_{27}	t_{26}	ϕ
p_{37}	t_{36}	ϕ

c. Calculer $td(M_0, t \in \lambda(q_i), q^\circ)$, $td(M_0, t \in \Gamma(q_i), {}^\circ q)$

q	$td(M_0, t \in \Gamma(q_i), {}^\circ q)$
p_{17}	0
p_{27}	0
p_{37}	0

d. Calculer $E(q_i^\circ)$, $S({}^\circ q_i)$

q	$S({}^\circ q_i)$	$E(q_i^\circ)$
p_{13}	0	0
p_{15}	0	0
p_{29}	0	0
p_{211}	0	0
p_{39}	0	0
p_{311}	0	0
p_{19}	0	0
p_{111}	0	0
p_{23}	0	0
p_{33}	0	0
p_{25}	0	0
p_{35}	0	0
p_{115}	0	0
p_{215}	0	0
p_{315}	0	0

p_{14}	0	0
p_{210}	0	0
p_{310}	0	0
p_{33}	0	0
p_{110}	0	0
p_{24}	0	0
p_{34}	0	0
p_{17}	0	indéfinie
p_{27}	0	indéfinie
p_{37}	0	indéfinie

3. Vérifier si l'état initial du système est un état admissible ou non.

- Pour Q_{mc1}={p_{13}, p_{15}, p_{29}, p_{211}, p_{39}, p_{311}}, $SMMI(Q_{mc1}, 2)$

Nous vérifions l'inégalité (4.15)

$$\left((\boldsymbol{k}-\mathbf{1})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{M_0(q_i)}-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{S(^\circ q_i)}+\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{\sigma(q_i^\circ)}\geq\mathbf{0}\right)$$

$$(\mathbf{2}-\mathbf{1})-\mathbf{0}-\mathbf{0}+\mathbf{0}=\mathbf{1}\ >0$$

L'inégalité (4.15) est validée, alors l'état initial de l'atelier est un état admissible par rapport à la première spécification.

- Pour Q_{mc2}={p_{19}, p_{111}, p_{23}, p_{33}, p_{25}, p_{35}}, $SMMI(Q_{mc2}, 2)$

Nous vérifions l'inégalité (4.15)

$$\left((\boldsymbol{k}-\mathbf{1})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{M_0(q_i)}-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{S(^\circ q_i)}+\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{\sigma(q_i^\circ)}\geq\mathbf{0}\right)$$

$$(\mathbf{2}-\mathbf{1})-\mathbf{0}-\mathbf{0}+\mathbf{0}=\mathbf{1}\ >0$$

L'inégalité (4.15) est validée, alors l'état initial de l'atelier est un état admissible par rapport à la deuxième spécification.

- Pour Q_{mc3}={ p_{115}, p_{215}, p_{315} }, $C_1 \equiv M(p_{115}) \leq 0 \vee (M(p_{215}) \leq 0 \wedge M(p_{315}) \leq 0)$

Nous vérifions l'inégalité (4.13) pour chaque place critique :

$$(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0(q)}-\boldsymbol{S(^\circ q)}+\boldsymbol{\sigma(q^\circ)}\geq\mathbf{0}$$

Pour p_{115} : $(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(p_{115})-\boldsymbol{S(t_{15})}+\boldsymbol{\sigma(t_{16})}=(\mathbf{1}-\mathbf{1})-\mathbf{0}-\mathbf{0}+\mathbf{0}=\mathbf{0}\Longrightarrow |4.13|=1$

Pour p_{215} : $(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(p_{215})-\boldsymbol{S(t_{25})}+\boldsymbol{\sigma(t_{26})}=(\mathbf{1}-\mathbf{1})-\mathbf{0}-\mathbf{0}+\mathbf{0}=\mathbf{0}\Longrightarrow |4.13|=1$

Pour p_{315} : $(\boldsymbol{k-1}) - \boldsymbol{M_0}(p_{315}) - \boldsymbol{S}(\boldsymbol{t_{35}}) + \boldsymbol{\sigma}(\boldsymbol{t_{36}}) = (\boldsymbol{1-1}) - \boldsymbol{0} - \boldsymbol{0} + \boldsymbol{0} = \boldsymbol{0} \Longrightarrow |4.13| = 1$

$$1 \vee (1 \wedge 1) = 1$$

Alors l'état initial de l'atelier n'est pas un état interdit par rapport à la troisième spécification.

- Pour Q_{mc4}={ p_{14}, p_{210}, p_{310} }, $SMMI(Q_{mc4}, 2)$

Nous vérifions l'inégalité (4.15)

$$\left((\boldsymbol{k-1}) - \sum_{\boldsymbol{i=1}}^{\boldsymbol{mc}} \boldsymbol{M_0}(\boldsymbol{q_i}) - \sum_{\boldsymbol{i=1}}^{\boldsymbol{mc}} \boldsymbol{S}(^{\circ}\boldsymbol{q_i}) + \sum_{\boldsymbol{i=1}}^{\boldsymbol{mc}} \boldsymbol{\sigma}(\boldsymbol{q_i}^{\circ}) \geq \boldsymbol{0} \right)$$

$$(\boldsymbol{2-1}) - \boldsymbol{0} - \boldsymbol{0} + \boldsymbol{0} = \boldsymbol{1} > 0$$

L'inégalité (4.15) est validée, alors l'état initial de l'atelier est un état admissible par rapport à la quatrième spécification.

- Pour Q_{mc5}={ p_{110}, p_{24}, p_{34} }, $C_2 \equiv M(p_{110}) \leq 0 \vee M(p_{24}) \leq 0 \vee M(p_{34}) \leq 0$

Nous calculons l'inégalité (4.13) pour chaque place critique :

$$(\boldsymbol{k-1}) - \boldsymbol{M_0}(\boldsymbol{q}) - \boldsymbol{S}(^{\circ}\boldsymbol{q}) + \boldsymbol{\sigma}(\boldsymbol{q}^{\circ}) \geq \boldsymbol{0}$$

Pour p_{110} : $(\boldsymbol{k-1}) - \boldsymbol{M_0}(p_{110}) - \boldsymbol{S}(\boldsymbol{t_{110}}) + \boldsymbol{\sigma}(\boldsymbol{t_{111}}) = (\boldsymbol{1-1}) - \boldsymbol{0} - \boldsymbol{0} + \boldsymbol{0} = \boldsymbol{0} \Longrightarrow |4.13| = 1$

Pour p_{24} : $(\boldsymbol{k-1}) - \boldsymbol{M_0}(p_{24}) - \boldsymbol{S}(\boldsymbol{t_{24}}) + \boldsymbol{\sigma}(\boldsymbol{t_{25}}) = (\boldsymbol{1-1}) - \boldsymbol{0} - \boldsymbol{0} + \boldsymbol{0} = \boldsymbol{0} \Longrightarrow |4.13| = 1$

Pour p_{34} : $(\boldsymbol{k-1}) - \boldsymbol{M_0}(p_{34}) - \boldsymbol{S}(\boldsymbol{t_{34}}) + \boldsymbol{\sigma}(\boldsymbol{t_{35}}) = (\boldsymbol{1-1}) - \boldsymbol{0} - \boldsymbol{0} + \boldsymbol{0} = \boldsymbol{0} \Longrightarrow |4.13| = 1$

$$1 \vee (1 \wedge 1) = 1$$

L'état initial de l'atelier est un état admissible par rapport à la cinquième spécification.

- Pour Q_{mc6}={ p_{17}, p_{27}, p_{37} }, $C_3 \equiv M(p_{17}) \leq 0 \vee M(p_{27}) \leq 0 \vee M(p_{37}) \leq 0$

Nous vérifions l'inégalité (4.13) pour chaque place critique :

$$(\boldsymbol{k-1}) - \boldsymbol{M_0}(\boldsymbol{q}) - \boldsymbol{S}(^{\circ}\boldsymbol{q}) + \boldsymbol{\sigma}(\boldsymbol{q}^{\circ}) \geq \boldsymbol{0}$$

Pour p_{17} : $(\boldsymbol{k-1}) - \boldsymbol{M_0}(p_{17}) - \boldsymbol{S}(\boldsymbol{t_{17}}) + \boldsymbol{\sigma}(\boldsymbol{t_{18}}) = (\boldsymbol{1-1}) - \boldsymbol{0} - \boldsymbol{0} + \boldsymbol{0} = \boldsymbol{0} \Longrightarrow |4.13| = 1$

Pour p_{27} : $(\boldsymbol{k-1}) - \boldsymbol{M_0}(p_{27}) - \boldsymbol{S}(\boldsymbol{t_{27}}) + \boldsymbol{\sigma}(\boldsymbol{t_{28}}) = (\boldsymbol{1-1}) - \boldsymbol{0} - \boldsymbol{0} + \boldsymbol{0} = \boldsymbol{0} \Longrightarrow |4.13| = 1$

Pour p_{37} : $(\boldsymbol{k-1}) - \boldsymbol{M_0}(p_{37}) - \boldsymbol{S}(\boldsymbol{t_{37}}) + \boldsymbol{\sigma}(\boldsymbol{t_{38}}) = (\boldsymbol{1-1}) - \boldsymbol{0} - \boldsymbol{0} + \boldsymbol{0} = \boldsymbol{0} \Longrightarrow |4.13| = 1$

$1 \vee (1 \wedge 1) = 1$

L'état initial de l'atelier est un état admissible par rapport à la sixième spécification.

4. $M_0 \in (M_f \cup M_D)$: on continue le calcule en ligne.

En ligne :

Si on veut franchir la séquence $\delta = t_{11} \rightarrow t_{12} \rightarrow t_{13} \rightarrow t_{14} \rightarrow t_{15} \rightarrow t_{16} \rightarrow t_{21} \rightarrow t_{22} \rightarrow t_{23} \rightarrow t_{24} \rightarrow t_{25} \rightarrow t_{26}$…

Pour le franchissement de t_{11}:

$t \in T_{uc}$:

- Ne fait rien

Pour le franchissement de t_{12}:

$t \in T_{uc}$:

Ne fait rien

Pour le franchissement de t_{13}:

$t \in T_c$:

a. $t \in \Gamma(p_{13})$, mettre à jour $S(^\circ q_i)$

$S(t_{13})=1$

b. la contrainte est de type $SMMI(Q_{mc1}, 2)$:

Nous calculons l'inégalité (4.15)

$$\left((\boldsymbol{k-1}) - \sum_{\boldsymbol{i=1}}^{\boldsymbol{mc}} \boldsymbol{M_0}(\boldsymbol{q_i}) - \sum_{\boldsymbol{i=1}}^{\boldsymbol{mc}} \boldsymbol{S}(^\circ \boldsymbol{q_i}) + \sum_{\boldsymbol{i=1}}^{\boldsymbol{mc}} \boldsymbol{\sigma}(\boldsymbol{q_i}^\circ) \geq \boldsymbol{0} \right)$$

$\boldsymbol{(2-1)-0-1+0=0} \quad \Rightarrow |(4.15)|=1$, t_{13} est franchissable.

Pour le franchissement de t_{14}:

$t \in T_c$:

a. $t \in \Gamma(p_{14})$, mettre à jour $S(^\circ q_i)$

$S(t_{14})$=1

b. la contrainte est de type *SMMI*(Q_{mc4}, 2):

Nous calculons l'inégalité (4.15)

$$\left((\boldsymbol{k}-\mathbf{1})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{M_0}(\boldsymbol{q_i})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{S}(^{\circ}\boldsymbol{q_i})+\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{\sigma}(\boldsymbol{q_i}^{\circ})\geq\mathbf{0}\right)$$

$(\mathbf{2}-\mathbf{1})-\mathbf{0}-\mathbf{1}+\mathbf{0}=\mathbf{0}$ $\Rightarrow$ |(4.15)|=1, t_{14} est franchissable.

Pour le franchissement de t_{15}:

$t\in T_c$:

a. $t\in\Gamma(p_{15})$, mettre à jour $S(^{\circ}q_i)$

$S(t_{15})$=1

b. la contrainte est de type *SMMI*(Q_{mc1}, 2):

Nous calculons l'inégalité (4.15)

$$\left((\boldsymbol{k}-\mathbf{1})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{M_0}(\boldsymbol{q_i})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{S}(^{\circ}\boldsymbol{q_i})+\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{\sigma}(\boldsymbol{q_i}^{\circ})\geq\mathbf{0}\right)$$

$(\mathbf{2}-\mathbf{1})-\mathbf{0}-\mathbf{1}+\mathbf{0}=\mathbf{0}$ $\Rightarrow$ |(4.15)|=1, t_{15} est franchissable.

Pour le franchissement de t_{16}:

$t\in T_c$:

a. $t\in\Gamma(p_{17})$, mettre à jour $S(^{\circ}q_i)$

$S(t_{17})$=1

b. la contrainte est de type C_3::

Pour p_{17} : $(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(p_{17})-\boldsymbol{S}(\boldsymbol{t_{17}})+\boldsymbol{\sigma}(\boldsymbol{t_{18}})=(\mathbf{1}-\mathbf{1})-\mathbf{0}-\mathbf{1}+\mathbf{0}=-\mathbf{1}\Longrightarrow|4.13|=0$

Pour p_{27} : $(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(p_{27})-\boldsymbol{S}(\boldsymbol{t_{27}})+\boldsymbol{\sigma}(\boldsymbol{t_{28}})=(\mathbf{1}-\mathbf{1})-\mathbf{0}-\mathbf{0}+\mathbf{0}=\mathbf{0}\Longrightarrow|4.13|=1$

Pour p_{37} : $(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(p_{37})-\boldsymbol{S}(\boldsymbol{t_{37}})+\boldsymbol{\sigma}(\boldsymbol{t_{38}})=(\mathbf{1}-\mathbf{1})-\mathbf{0}-\mathbf{0}+\mathbf{0}=\mathbf{0}\Longrightarrow|4.13|=1$

$0 \vee (1 \wedge 1) = 1$

Alors t_{16} est franchissable.

Pour le franchissement de t_{21}:

$t \in T_{uc}$:

Ne fait rien

Pour le franchissement de t_{22}:

$t \in T_{uc}$:

Ne fait rien

Pour le franchissement de t_{23}:

$t \in T_c$:

a. $t \in \Gamma(p_{23})$, mettre à jour $S(^\circ q_i)$

   $S(t_{23})=1$

b. la contrainte est de type $SMMI(Q_{mc2}, 2)$:

Nous calculons l'inégalité (4.15)

$$\left((\boldsymbol{k}-\mathbf{1}) - \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}} \boldsymbol{M_0}(\boldsymbol{q_i}) - \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}} \boldsymbol{S}(^\circ \boldsymbol{q_i}) + \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}} \boldsymbol{\sigma}(\boldsymbol{q_i}^\circ) \geq \mathbf{0} \right)$$

$(\mathbf{2}-\mathbf{1}) - \mathbf{0} - \mathbf{1} + \mathbf{0} = \mathbf{0}$ $\Rightarrow$ |(4.15)|=1, t_{23} est franchissable.

Pour le franchissement de t_{24}:

$t \in T_c$:

a. $t \in \Gamma(\ p_{24})$, mettre à jour $S(^\circ q_i)$

   $S(t_{24})=1$

b. la contrainte est de type $SMMI(Q_{mc5}, 2)$:

Nous calculons l'inégalité (4.15)

$$\left((\boldsymbol{k}-\mathbf{1})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{M_0}(\boldsymbol{q_i})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{S}(^{\circ}\boldsymbol{q_i})+\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{\sigma}(\boldsymbol{q_i}^{\circ})\geq\mathbf{0}\right)$$

$(\mathbf{2}-\mathbf{1})-\mathbf{0}-\mathbf{1}+\mathbf{0}=\mathbf{0}$ $\Rightarrow$ |(4.15)|=1, t_{24} est franchissable.

Pour le franchissement de t_{25}:

$t\in T_c$:

a. $t\in\Gamma(p_{25})$, mettre à jour $S(^{\circ}q_i)$

$S(t_{25})=1$

b. la contrainte est de type $SMMI(Q_{mc2}, 2)$:

Nous calculons l'inégalité (4.15)

$$\left((\boldsymbol{k}-\mathbf{1})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{M_0}(\boldsymbol{q_i})-\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{S}(^{\circ}\boldsymbol{q_i})+\sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{mc}}\boldsymbol{\sigma}(\boldsymbol{q_i}^{\circ})\geq\mathbf{0}\right)$$

$(\mathbf{2}-\mathbf{1})-\mathbf{0}-\mathbf{1}+\mathbf{0}=\mathbf{0}$ $\Rightarrow$ |(4.15)|=1, t_{25} est franchissable.

Pour le franchissement de t_{26}:

$t\in T_c$:

a. $t\in\Gamma(p_{27})$, mettre à jour $S(^{\circ}q_i)$

$S(t_{27})=1$

b. la contrainte est de type C_3:

Pour p_{17} : $(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(p_{17})-\boldsymbol{S}(\boldsymbol{t_{17}})+\boldsymbol{\sigma}(\boldsymbol{t_{18}})=(\mathbf{1}-\mathbf{1})-\mathbf{0}-\mathbf{1}+\mathbf{0}=-\mathbf{1}\Longrightarrow |4.13|=0$

Pour p_{27} : $(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(p_{27})-\boldsymbol{S}(\boldsymbol{t_{27}})+\boldsymbol{\sigma}(\boldsymbol{t_{28}})=(\mathbf{1}-\mathbf{1})-\mathbf{0}-\mathbf{1}+\mathbf{0}=-\mathbf{1}\Longrightarrow |4.13|=0$

Pour p_{37} : $(\boldsymbol{k}-\mathbf{1})-\boldsymbol{M_0}(p_{37})-\boldsymbol{S}(\boldsymbol{t_{37}})+\boldsymbol{\sigma}(\boldsymbol{t_{38}})=(\mathbf{1}-\mathbf{1})-\mathbf{0}-\mathbf{0}+\mathbf{0}=\mathbf{0}\Longrightarrow |4.13|=1$

$0\vee(0\wedge1)=0$

Alors t_{16} n'est pas franchissable.

C'est-à-dire que le robot R_2 ne doit pas mettre la *pièce*2 usinée par M_2 dans *Pré-*S_2 jusqu'à ce que la *pièce*1 usinée par M_1 quitte le tapis $S_{5\ modélisé}$ par le franchissant de t_{18}.

7.5 Conclusion

Dans cette partie, nous avons étudié un atelier flexible constitué de machines-outils à commande numérique, considéré comme un système à événements discrets SED. Nous avons considéré dans cet atelier que chaque machine est commandée par un contrôleur propre pour générer des événements qui seraient traduits en langage machine pour produire les opérations demandées. Le rôle de notre superviseur est de synchroniser les événements pour toutes les machines et les composants de cet atelier à fin de respecter les spécifications définies.

Nous avons construit le superviseur en nous basant sur les algorithmes présentés dans cette thèse. La totalité des algorithmes a été programmée en utilisant le Visual Basic et ces algorithmes pourront être implémentés sur la plateforme supervision du laboratoire.

Conclusion

Les travaux que nous avons menés dans le cadre de ma thèse de doctorat représentent une contribution au domaine de la synthèse de la commande des systèmes à événements discrets. Nous avons cherché à résoudre le problème d'état interdit par rapport à un nouveau modèle de spécifications en garantissant la permissivité de la loi de contrôle générée.

Nous avons principalement quatre contributions. La première est de proposer un nouveau modèle pour représenter le problème d'état interdit. La deuxième contribution est de développer une technique de synthèse de la commande pour les systèmes modélisés par le graphe d'événements en présence des transitions incontrôlables. La troisième est d'éteindre cette technique de synthèse pour traiter le problème en présence des transitions inobservables. La quatrième contribution présente une loi de contrôle pour traiter le cas particulier où le système à commander est modélisé par un réseaux de Petri P-temporisé.

Ainsi, dans le troisième chapitre, nous avons introduit un nouveau modèle de spécifications où l'élément de base est la Contrainte d'Exclusion de Marquage (CEM) pour représenter le problème d'état interdit. Nous avons montré qu'une CEM concerne une place critique et qu'elle rassemble des expressions linéaires et logiques. Pour plusieurs places critiques, nous avons proposés 4 extensions principales CEM-Ou, CEM-Et, CEM-M et C-CEM. Nous avons comparé dans le même chapitre la puissance de modélisation de ce modèle avec les autres modèles existants.

Ensuite, dans les chapitres quatre et cinq, nous avons présenté une technique de synthèse de commande pour les systèmes à événements discrets modélisés par l'outil graphe d'événements, en présence des transitions incontrôlables et inobservables. La nouveauté de cette technique est de contrôler et d'observer les jetons qui arrivent aux places critique et ceux qui quittent ces places. Pour cela, nous avons classifié l'ensemble de transitions par rapport chaque place critique à quatre ensembles. En utilisant ces ensembles, nous avons développé deux fonctions en plus de deux fonctions déjà proposées dans la littérature. Les quatre fonctions sont utilisées pour construire une loi de commande capable à estimer la distribution des jetons et contrôler leurs mouvements en respectant les spécifications modélisées par CEM.

Dans le sixième chapitre, nous avons pris en compte l'information temporelle sur l'occurrence des événements tout en considérant la présence des transitions incontrôlables et/ou inobservables. Nous avons analysé la distribution des jetons au fil du temps. En utilisant cette distribution, nous avons mis les bases d'une approche de synthèse de commande pour les systèmes à événements discrets temporisés soumis aux contraintes de type CEM.

La modélisation de problème d'état interdit ainsi que les approches de synthèse de la commande développées dans ce mémoire traitent une classe particulière de réseaux de Petri (Graphe d'événements). Il serait intéressant de poursuivre et généraliser cette approche aux autres classes de réseaux de Petri.

Il serait intéressant dans la suite des travaux d'introduire la contrainte temporelle et dans la modélisation du procédé et dans les spécifications. Il apparaît essentiel d'avoir des méthodes qui permettent de modéliser et analyser les phénomènes temporels dans les systèmes à événements discrets pour pouvoir prouver le respect des contraintes de temps. L'une des perspectives de cette thèse, serait d'étendre les approches proposées pour la prise en compte des contraintes temporelles.

La modélisation et la méthode de synthèse actuelle, ne tiennent pas compte des situations de blocage pouvant se produire en boucle fermée. Les lois de commande déterminées dans cette thèse permettent au procédé de respecter les spécifications de type CEM-Ou, CEM-Et et CEM-M. L'occurrence d'un événement contrôlable et autorisé par le superviseur mais peut dans certains cas provoquer une situation de blocage. Il est intéressant d'étendre ces travaux pour éviter ce type de situations.

Références

Achour, Z. (2005). *Contribution à la synthèse des controleurs des systèmes à événements discrets partiellement observables.* Metz: Thèse, Université de Paul verlaine.

Achour, Z. a. (2004). Supervisory control of partially observable marked graphs. *Automatic Control, 49*(11), 2007--2011.

Achour, Z. a. (2008). On the supervisory control of marked graphs. *Industrial Engineering and Engineering Management* (pp. 2122--2126). singapore: IEEE.

Atli, M. A. (2011). Control Law for Timed Marhed Graph Constrained by Marking Exclusion Constraint OR/AND. *Automatica si Calculatoire (Iasi)*, 75-92.

Atli, M., Achour, Z., Sava, A., & Rezg, N. (September 2010). Supervisory Control of Partially Observable Marked Graph Based on Marking Exclusion Constraint. *5th Conference on Management and Control of Production Logistics* (pp. 64-69). Coimbra, Portugal: IFAC.

Atli, M., Adjallah, K., Achour, Z., Sava, A., & Rezg, N. (March 2011). Supervisory control of timed-place marked graph based on Marking Exclusion Constraint. *International Conference on Communications, Computing and Control Applications (CCCA)* (pp. 1-8). Tunis: IEEE.

Atli, M., Sava, A., Achour, Z., & Rezg, N. (July 2010). Control synthesis for partially controlled marked graph subject to marking exclusion constraints. *Large Scale Complex Systems Theory and Applications. 9.* Lille -France: IFAC.

Atli, M., Sava, A., Achour, Z., Adjallah, K., & Rezg, N. (April 2010). The Control Policy of Marking Exclusion Constraint applying on Marked Graph Partially controllable. *4ème Congrès International Francophone de Mécanique Avancée - CIFMA* (pp. 1-9). Aleppo: University of Aleppo.

Badouel, E., Bernardinello, L., & Darondeau, P. (1995). Polynomial algorithms for the synthesis of bounded nets. *TAPSOFT'95: Theory and Practice of Software Development*, pp 364-378.

Basile, F. a. (2001). Deadlock recovery of Petri net models controlled using observers. *8th IEEE International Conference on Emerging Technologies and Factory Automation* (pp. 441--449). IEEE.

Basile, F. A. (2007). An optimization approach to Petri net monitor design. *Automatic Control, 52*(2), 306--311.

Basile, F. C. (2006). Suboptimal supervisory control of Petri nets in presence of uncontrollable transitions via monitor places. *Automatica, 42*(6), 995--1004.

Basile, F., Recalde, L., Chiacchio, P., & Silva, M. (2000). Closed-loop live Petri net supervisors for generalized mutual exclusion constraints. *5th International Workshop on Discrete Event Systems (WODES 00),* (pp. 169--180). Gent, B, Aug.

Basile, F., Recalde, L., Chiacchio, P., & Silva, M. (2009). Closed-loop Live Marked Graphs under Generalized Mutual Exclusion Constraint Enforcement. *Discrete Event Dynamic Systems, 19*(1), 1-30.

Boel, R., Ben-Naoum, L., & Breusegem, V. V. (1995). On forbidden state problems for a class of controlled Petri nets. *IEEE Transactions on Automatic Control, 40*(10), 1717-1731.

Brahms, G. (1983). *Réseaux de Petri: théorie et pratique. Book, vol 1 & 2.* Paris: Masson.

Brandin, B., & Wonham, W. (1994). Supervisory control of timed discrete-event systems. *IEEE Transactions on Automatic Control, 39*(2), 329-342.

Caines, P., Greiner, R., & Wang, S. (1988). Dynamical logic observers for finite automata. *Proceedings of the 27th IEEE Conference on Decision and Control* (pp. 226--233). IEEE.

Charbonnier, F., Alla, H., & David, R. (1999). Discrete-event dynamic systems. *IEEE Transactions on Control Systems Technology, 7*(2), 175-187.

D. Corona, A. G. (2004). Marking estimation of Petri nets with silent transitions. *IEEE Trans. on Automatic Control, 1*, 966--971.

David, R., & Alla, H. (1992). *Petri Nets and Grafcet: tools for modelling discrete event systems.* Book, Prentice Hall.

Gaudin, B. (2004). *Synthèse de contrôlleurs sur des systèmes à événements discrets structurés, thèse.*

Ghaffari, A. a. (2002). Algebraic and geometric characterization of Petri net controllers using the theory of regions. *Sixth International Workshop on Discrete Event Systems* (pp. 219--224). IEEE.

Ghaffari, A. X. (2003). Feedback control logic for forbidden-state problems of marked graphs: application to a real manufacturing system. *IEEE Transactions on Automatic Control, 48*(1), 18--29.

Giua, A. a.-C. (1993). Petri net supervisors for generalized mutual exclusion constraints. *12th IFAC world congress.* Sideny, Australia: IFAC.

Giua, A., & Seatzu, C. (2002, Septembre). Obervabitity of place/transition nets . *IEEE Transactions on Automatic Control, 47*(9), 1424-1437.

Guia, A. (1992). *Petri nets as discrete event models for supervisory control.* NY, USA: Rensselaer Polytechnic Institute.

Holloway, & Krogh. (1990). Synthesis of feedback control logic for a class of controlled Petri nets. *IEEE Transactions on Automatic Control, 35*(5), 514--523.

Holloway, L., Guan, X., & Zhang, L. (1996). A Generalization of State Avoidance Policies for Controlled Petri Nets. *IEEE Transactions on Automatic Control, 41*(6), 804--816.

Kattan, B. (2004). *Synthèe structurelle d'un contrôlleur basèe sur le Grafcet, thèse.* Universit{\'e} Joseph Fourier. Grenoble: Joseph Fourier.

Krogh, B. H., & Holloway, L. E. (1991). Synthesis of feedback control logic for discrete manufacturing systems. *Automatica, 27*, 641-651.

Krogh, H. a. (1992). On closed-loop liveness of discrete-event systems under maximally permissive control. *IEEE Transactions on Automatic Control, 37*(5), 692--697.

Kumar, R. (1992). *Supervisory synthesis techniques for discrete event dynamical systems.* Usa, Texas: University of Texas at Austin.

Li, & Wonham. (1988). Controllability and observability in the state-feedback control of discrete-event systems. *IEEE*, (pp. 203--208).

Li, Y., & Wonham, W. (1993). Control of vector discrete-event systems. i. the base model. *Automatic Control, IEEE Transactions on, 38*(8), 1214-1227.

Lin, F., & Wonham, W. (1990). Decentralized control and coordination of discrete-event systems with partial observation. *Automatic Control, IEEE Transactions on, 35*(12), 1330-1337.

Moody, J. O., & Antsaklis, P. J. (1998). *Supervisory Control of Discrete Event Systems Using Petri Nets.* Kluwer Academic Publishers.

Moody, J. O., Lemmon, M., & Antsaklis, P. J. (1996). Supervisory control of Petri nets with uncontrollable/unobservable transitions. *35th IEEE CDC.* Kobe, Japan.

Moody, J., & Antsaklis, P. (2000). Petri net supervisors for DES with uncontrollable and unobservable transitions. *Automatic Control, IEEE Transactions on, 45*(3), 462-476.

Murata, T. (1989). Petri nets: Properties, analysis and applications. *Proceedings of the IEEE, 77*(4), 541--580.

Ozveren, C., & Willsky, A. (1990). Observability of discrete event dynamic systems. *Automatic Control, IEEE Transactions on, 35*(7), 797-806.

Ramadge. (1986). Observability of discrete event systems. *25th IEEE Conference on Decision and Control. 25*, pp. 1108-1112. IEEE.

Ramadge, P., & Wonham, W. (1987- a). Modular feedback logic for discrete event systems. *SIAM Journal on Control and Optimization, 25*, 1202.

Ramadge, P., & Wonham, W. (1987-b). Supervisory control of a class of discrete event processes. *SIAM Journal on Control and Optimization, 25*, 206.

Ramadge, P., & Wonham, W. (1989). The Control of Discrete Event Systems. *Proceeding of IEEE, 77*(1), Vol 77, 81--98.

Ramchandani, C. (1974). Analysis of asynchronous concurrent systems by timed Petri nets. *Massachusetts Institute of Technology*.

Sava, A. (2001). *Sur la synthèse de la commande des systèmes à événements discrets temporisés, thèse.* Institut National Polytechnique de Grenoble.

Sifakis, J. (1980). Use of Petri nets for performance evaluation. *Acta Cybernetica, 4*(1978), 185-202.

Stremersch, G. (2000). *Linear algebraic design of supervisors for partially observed Petri nets, Book.* Citeseer.

Takai, S., Ushio, T., & Kodama, S. (1995). Static-state feedback control of discrete-event systems under partial observation. *Automatic Control, IEEE Transactions on, 40*(11), 1950-1954.

Yamalidou, K., Moody, J., Lemmon, M., & Antsaklis, P. (1996). Feedback control of petri nets based on place invariants* 1. *Automatica, 32*(1), 15-28.